IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

Fingerprint Compression based on Sparse Representation using Pixel Level Patch Decomposition

J. Saikrishna[1], Dr. T. Sreenivasulu Reddy[2]

M.Tech, Dept of ECE, S V University College of Engineering, Tirupati, India[1]

Associate Professor, Dept of ECE, S V University College of Engineering, Tirupati, India[2]

Abstract: Reducing memory size of image is known as Compression. In the process of compression there will be degradation in the image, such type of compression is known as Lossy compression. The counter part of losssy compression is Lossless compression, where image is not degraded and quality is retained. The classical methods to compress an image were JPEG using DCT and JPEG 2000 using DWT. These methods come under frequency domain lossy compression standards. To improve the compression ratio and retain the quality of compressed image compared with these classical methods a new method in spatial domain is proposed which is known as sparse representation using pixel level patch decomposition. The decomposition is done on the image based on the pixel values at the level of 20X20 patches and a dictionary is constructed to remove the redundancy in the image, a threshold value is fixed the patch values which are greater than the threshold values retained in the dictionary and remaining are discarded. The metrics like PSNR, MSE and compression ratio are calculated and significant improvement is observed in the proposed method.

Keywords: JPEG, JPEG2000, Sparse Representation, Fingerprint Dictionary.

I. INTRODUCTION

Biometric recognisation of any human being involves by taking fingerprint, iris and face recognization features for security monitoring purposes as fingerprint is one of the robust biometric recognisation features. Today the databases of fingerprint are prominently in very huge memory size for productive utility of memory and fast accessing system. The storage of fingerprint images should be in a small size and quality should be retained.

To address these issue image compression plays a crucial role. The classical methods of image compression are DCT-JPEG, DWT-JPEG2000 are widely used but has some limitations in retaining the quality and reducing the memory size as these compression standards comes under frequency domain lossy compression hierarchy. So to alleviate the sited problems in the classical methods we proposed a spatial domain sparse representation standard for finger print image compression. The sparse representation will play on the spatial domain of the image considering the grey level values of the pixels. The fingerprint image is taken from a database and a threshold value for each pixel I an fingerprint image is fixed the pixels which are greater than the threshold are retained and remaining are discarded and a 20X20 patches of sparse represented image is sent to the minimization problem section (MP) method which reduces the error in the patches and the huff-man code along with quantization is used to generate the binary stream of compressed image is generated which is a sparse representation compressed image output. The quality metrics like MSE, PSNR are calculated for the proposed method and compressed ratio is also found all the metrics shows significant improvement compared with classical methods like DCT-JPEG, DWT-JPEG2000.

II. METHODOLOGY

➢ **Model of Sparse representation**

Given A = [a1, a2, . . . , aN] ∈ RM×N , a new sample y ∈ RM×1, is assumed to be represented as a linear combination of few columns from the dictionary A, as shown in formula (1). This is the only prior knowledge about the dictionary in this algorithm. Later, the property can be ensured by constructing the dictionary properly

$$y = AX \qquad (1)$$

Where y ∈ RM×1, A ∈ RM×N and x = [x1, x2, . . . , x N]T ∈ RN×1.

Obviously, the system y =Ax is under determined when M < N. Then, the solution is not special. According to the theory, the representation is sparse. The solution can be obtained to resolve the following optimization problems.

$$(l^0) : \min \|x\|_0 \; s.t. Ax = y \qquad (2)$$

Solution of the optimization problem is expected to be very sparse, namely, ‖x‖0<< N. The notation ‖x‖ counts the nonzero entries in x. actually it is not a norm. However, without ambiguity, we still call it l^0-normalization. The compression of y value can be achieved by compressing the x value. Then record the locations of its non-zero entries and their magnitudes ant then later, quantize and encode the record values. This is what we can do in the next, techniques for solving the optimization problem

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

➢ Sparse Solution by Greedy (General)Algorithm:

To resolve the optimization problem l^0 directly. The problem of finding the sparsest solution of the system (2) is NP-hard. The Matching Pursuit (MP) because of its simplicity and efficiency is often used to approximately solve the l^0 problem. Many variants of the algorithm are available, offering improvements either in accuracy or and in complexity. Although the theoretical analysis of these algorithms is difficult, experiments show that they behave quite well when the number of non-zero entries is low

➢ Sparse Solution by l^1-Minimization:

It is a natural idea that the optimization problem (2) can be approximated by solving the following optimization problem

$$\left(l^p\right): \min \|x\|_p^p \ s.t. Ax = y \quad (3)$$

Obviously, the smaller p is the closer solutions of the two optimization problems l^0 and l^p are, as illustrated in Fig 1.

This is because the magnitude of x is not important when p is small. What does matter is whether x is equal to 0 or not. Therefore, p is theoretically chosen as small as possible. However, the optimization problem (3) is not convex if 0 < p < 1. It makes p = 1 the most ideal situation, namely, the following problems

$$\left(l^1\right): \min \|x\|_1 \ s.t. Ax = y \quad (4)$$

Recent developments in the field of sparse representation and compressed sensing reveal that the solution of the optimization problem (4) is almost equal to the solution of the optimization problem (2) if the optimal solution is sparse enough. The problem (4) can be effectively solved by linear programming methods

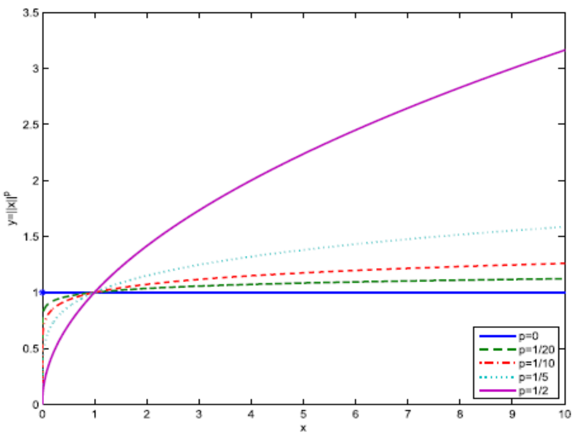


Fig.1. the behaviour of $\|x\|_p$ for various values of p. As p tends to zero, $\|x\|_p$ approaches the l^0-norm.

➢ FINGERPRINT COMPRESSION BASED ON REPRESENTATION

It is defined that the size of the dictionary is very large. The dictionary contains more information as much as possible. To obtain a dictionary with a more size, the pre-processing is essential and influenced by appearance, rotation and noise; the fingerprints of the same fingerprint may look very different. We know that each fingerprint image is to be arranged, independently of the others. The most common pre-aligned technique is to translate and rotate the fingerprint images are according to position of core point. Unfortunately, reliable detection of the core is very difficult in fingerprint images with very low quality. The core is correctly detected and the size of the dictionary may be huge in size because of the size of the whole fingerprint image is very large.

Compared with the basic images, the fingerprint images are to be in a fundamental structure. They are mixture of ridges and valleys. In local regions they are to be same. To solve these two problems, the whole image is sliced into square and is non-overlapped in to small patches. For these small patches, there are no problems of transformation and rotation. The size of the dictionary is not too large because of the small blocks are very smaller.

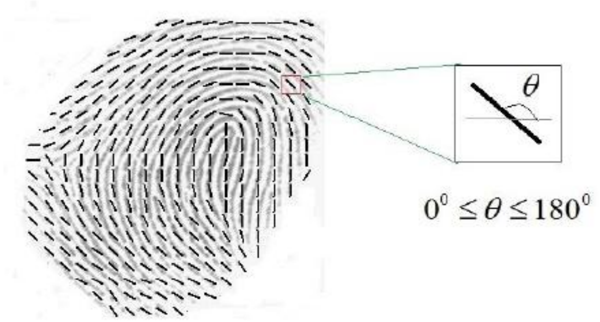


Fig.2. A fingerprint image with its corresponding orientation image computed over a square-meshed grid. Each element denotes the local orientation of the fingerprint ridges.

➢ Construction of the Dictionary

The dictionary can be constructed in three ways. Firstly, we construct a training set. Then, the dictionary is obtained from the set. The whole fingerprint images are cut and then into fixed-size square patches. Given these patches after the initial screening

- The first patch is added to the dictionary, which is basically in empty.
- To check whether the next patch is to be similar to the all patches in dictionary. If yes, then the next Patch is to be tested; otherwise, the patch is added into the dictionary. Then the common measures between the two patches are calculated by solving the optimization problem (5).

$$s\left(p_1, p_2\right) = \min \left\| \frac{p_1}{\|p_1\|_F^2} - t * \frac{p_2}{\|p_2\|_F^2} \right\|_F^2 \quad (5)$$

Where $\| \cdot \|_F^2$ is the Fresenius norm. P1 and P2 are the corresponding matrices of the two patches. t, a are the parameters of the optimization problem (5), is a scaling factor.

- Repeat the second step until all patches have been tested

Before the dictionary is constructed, the mean value of the every patch is to be calculated and subtracted from the

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

next patch. Then the details of the three methods are given.

- The first method is to choose fingerprint patches from the training samples at random and arrange these patches as columns of the dictionary matrix.
- The second method is the patches from the front view of a fingerprint have an orientation while the patches from the background are does not have, as shown in Fig. 2. This fact can be used to construct the dictionary. Divide the interval [00, . . . , 1800] into the equal-size intervals. Each interval is to be represented by an orientation (the middle value of each interval is chosen). Choose the same number of patches for every interval and arrange them into the dictionary.
- The third method is a training method called K-SVD. The dictionary is obtained by the iteratively solving the optimization problem (6). Y is consisted of the training patches, A is the dictionary, X is the coefficients and Xi is the i^{th} column of X. We can calculate the coefficient matrix X using MP method, which guarantees the coefficient vector Xi has no more than T non-zero elements. Then update every dictionary element based on the singular value decomposition (SVD).

$$\min_{A,X} \|Y - AX\|_F^2 \; s.t. \forall i, \|X_i\|_0 < T \qquad (6)$$

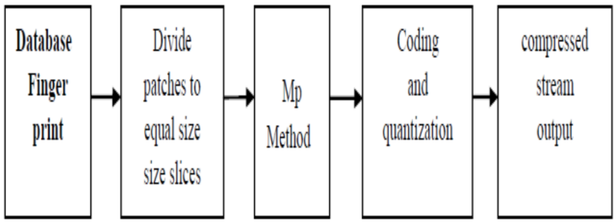
➢ Compression of a Given Fingerprint

A new fingerprint, slice is divided into square patches which have the same size with the training patches. The size of the patches has a direct impact on the compression efficiency. The algorithm becomes more efficient as the size increases. However, the computation complexity and the size of the dictionary also increase rapidly. The proper size should be chosen. How to choose this size will be give in. In addition, to make the patches fit the dictionary better, the mean of each patch needs to be calculated and subtracted from the patch. After that, compute the sparse representation for each patch by solving the l0 problem. Those coefficients whose absolute values are less than a given threshold are treated as zero. For each patch, four kinds of information need to be recorded. They are the mean value, the number about how many atoms to use, the coefficients and their locations. The tests show that many image patches require few coefficients. Consequently, compared with the use of a fixed number of coefficients, the method reduces the coding complexity and improves the compression ratio

➢ Coding and Quantization

Entropy coding of the atom number of each patch, the mean value of each patch, the coefficients and the indexes is carried out by static arithmetic coders. The atom number of each patch is separately coded. The mean value of each patch is also separately coded. The quantization of coefficients is performed using the Lloyd algorithm, learnt off-line from the coefficients which are obtained from the training set by the MP algorithm over the dictionary. The first coefficient of each block is quantized with a larger number of bits than other coefficients and entropy-coded using a separate arithmetic coder. The model for the indexes is estimated by using the source statistics obtained off-line from the training set. The first index and other indexes are coded by the same arithmetic encoder. In the following experiments, the first coefficient is quantized with 6 bits and other coefficients are quantized with 4 bits.

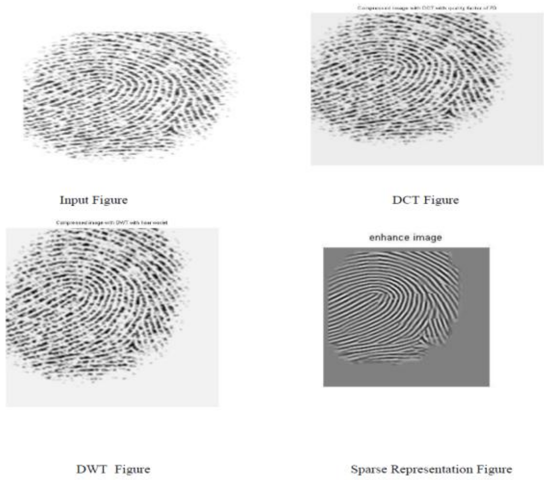
III.SYSTEM ARCHITECTURE



1) The fingerprint image of .JPG format is taken from a dataset for decomposing it into patches.
2) By using spatial domain technique, fixing a threshold for a pixel value, the pixels whose values are greater than threshold are retained and remaining are discarded.
3) The output after decomposition is 20X20 patches which are sent to reducing the error by applying minimization problem technique.
4) By applying huff-man coding and quantization the patches are coded and quantized so that the memory size is reduced.
5) The output is a lossless compressed format of input finger print image.

IV.RESULTS

Input fingerprint image is taken from a database classical methods like JPEG,JPEG200 are applied on the image for compression and the compressed output images are as shown in the figure1, figure2, and figure3, the sparse methods is applied on the same input image, the compressed output is shown in the figure 4. By observing all the compressed output sparse compressed output means the quality of the image.



Input Figure DCT Figure

DWT Figure Sparse Representation Figure

The sparse representation for image compression is recurred for input images as shown in the figures

Copyright to IJARCCE

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

Figure : 1

Figure : 2

Figure : 3

Figure : 4

Table 1: Comparison of DCT, DWT and Sprarse representation of image compression

Figures	DCT					DWT					Sparse Representation				
	input File size	Output File Size	Compression ratio	PSNR	MSE	input File size	Output File Size	Compression ratio	PSNR	MSE	input File size	Output File Size	Compression ratio	PSNR	MSE
Figure 1	75.5 KB	40.08 KB	0.531	36.9209	13.2126	75.5 KB	39.03 KB	0.516	62.7976	0.0341	75.5 KB	11.90 KB	0.157	78.0459	1.08E+04
Figure 2	75.5 KB	37.80 KB	0.501	39.2565	7.7167	75.5 KB	36.05 KB	0.477	64.771	0.0217	75.5 KB	11.40 KB	0.151	75.2185	1.15E+04
Figure 3	75.5 KB	41.01 KB	0.543	37.5977	11.3061	75.5 KB	39.09 KB	0.517	62.9095	0.0333	75.5 KB	11.80 KB	0.156	77.8371	1.08E+04
Figure 4	75.5 KB	41.60 KB	0.551	38.6563	8.8603	75.5 KB	40.20 KB	0.532	60.0129	0.0648	75.5 KB	80 KB	0.156	77.8579	1.08E+04

V. CONCLUSIONS

Compression, quality retaining of a fingerprint image is achieved by using spatial domain compression technique called sparse representation. Minimization problem is also applied to the decomposed patches which reduces the error effect on the patches. So by using this scenario a significant improvement in the quality metrics and memory metrics like PSNR, MSE and Compression ratio respectively improved compared with classical methods JPEG-DCT, JPEG2000-DWT.

REFERENCES

[1] Guangqi Shao, Yanping Wu, Yong A, Xiao Liu, and Tiande Guo, Fingerprint compression based on Sparse representation, vol.23, No.2, Feb 2014

[2] D. Maltoni, D. Miao, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, 2nd ed. London, U.K.: Springer-Verlag, 2009

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

[4] 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.

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

[6] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of JPEG-2000," in Proc. IEEE Data Compress. Conf.,Mar. 2000, pp. 523–541.

[7] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," IEEE Signal Process. Mag., vol. 11, no. 5,pp. 36–58, Sep. 2001.

[8] T. Hopper, C. Brislawn, and J. Bradley, "WSQ gray-scale fingerprint image compression specification," Federal Bureau of Investigation, Criminal Justice Information Services, Washington, DC, USA, Tech. Rep. IAFIS-IC-0110-V2, Feb. 1993.

[9] C. M. Brislawn, J. N. Bradley, R. J. Onyshczak, and T. Hopper, "FBI compression standard for digitized fingerprint images," Proc. SPIE, vol. 2847, pp. 344–355, Aug. 1996.

[10] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. Circuits Syst. Video Technol., vol. 6, no. 3, pp. 243–250, Jun. 1996.

[11] R. Sudhakar, R. Karthiga, and S. Jayaraman, "Fingerprint compression using contourlet transform with modified SPIHT algorithm," IJECE Iranian J. Electr. Comput. Eng., vol. 5, no. 1, pp. 3–10, 2005.

[12] S. G. Mallat and Z. Zhang, "Matching pursuits with timefrequency dictionaries," IEEE Trans. Signal Process., vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[13] S. S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," SIAM Rev., vol. 43, no. 1, pp. 129–159, 2001.

[14] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 2, pp. 210–227, Feb. 2009.

[15] Y. Y. Zhou, T. D. Guo, and M. Wu, "Fingerprint image compression algorithm based on matrix optimization," in Proc. 6th Int. Conf. Digital Content, Multimedia Technol. Appl., 2010, pp. 14–19.

[16] A. J. Ferreira and M. A. T. Figueiredo, "Class-adapted image compression using independent component analysis," in Proc. Int. Conf. Image Process., vol. 1. 2003, pp. 625–628.

[17] A. J. Ferreira and M. A. T. Figueiredo, "On the use of independent component analysis for image compression," Signal Process., Image Commun., vol. 21, no. 5, pp. 378–389, 2006.

[18] P. Paateroand U. Tapper, "Positive matrix factorization: A nonnegative factor model with optimalutilization of error estimates of data values," Environmetrics, vol. 5, no. 1, pp. 111–126, 1994.

[19] D. D. Leeand and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," Nature, vol. 401, pp. 799–791, Oct. 1999.

[20] E. Amaldi and V. Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems," Theoretical Comput. Sci., vol. 209, nos. 1–2, pp. 237–260, 1998.

[21] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," IEEE Trans. Signal Process., vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

BIOGRAPHIES

J. Saikrishna completed his B.Tech in Electronics and Communication Engineering at Sri Venkatesa Perumal College of Engineering and Technology, JNTU-A, Puttur, India in 2010. He is pursuing his M.Tech Degree with specialization of Signal Processing (SP) at Sri Venkateswara University College of Engineering (SVUCE), S.V. University, Tirupati, India.

Dr T. Sreenivasulu Reddy received the B.Tech. Degree in ECE from Sri Venkateswara University, Tirupati, India, in 1990, and M.Eng. degree in Digital Electronics and Communication Engg from Karnatak University, Dharwad, India, in 1996 and Ph.D. degree in Radar Signal Processing from Sri Venkateswara University, Tirupati. He is currently an Associate Professor with the Department of Electronics and Communication Engineering, Sri Venkateswara University College of Engineering, Sri Venkateswara University. His research interests include Radar and image signal processing. Mr. Reddy is a Fellow of the Institution of Electronics and Telecommunication Engineers and a member of the Indian Society for Technical Education.

DOI 10.17148/IJARCCE.2015.411105