K-SVD: ALGORITHM FOR FINGERPRINT COMPRESSION BASED ON SPARSE REPRESENTATION

Trishala K. Balsaraf
PG Student,
Department of Electronics and Telecommunication Engineering,
Shree Tulja Bhavani College of Engineering,
Tuljapur, Maharashtra, India

Mahesh N. Karanjkar
Assistant Prof.
Department of Electronics and Telecommunication Engineering,
Shree Tulja Bhavani College of Engineering,
Tuljapur, Maharashtra, India

ABSTRACT

In current years there has been an increasing interest in the study of sparse representation of signals. Using an overcomplete glossary that contains prototype signal-atoms, signals are described by sparse linear combinations of these atoms. Recognition of persons by means of biometric description is an important technology in the society, because biometric identifiers cannot be shared and they intrinsically characterize the individual's bodily distinctiveness. Among several biometric recognition technologies, fingerprint compression is very popular for personal identification. One more fingerprint compression algorithm based on sparse representation using K-SVD algorithm is introduced. In the algorithm, First we construct a dictionary for predefined fingerprint photocopy patches. For a new given fingerprint images, suggest its patches according to the dictionary by computing l^0-minimization by MP method and then quantize and encode the representation.This paper comparesdissimilarcompression standards like JPEG,JPEG-2000,WSQ,K-SVDetc. The paper show that this is effective compared with several competing compression techniques particularly at high compression ratios.

INDEX TERMS - Compression, sparse representation, JPEG, JPEG 2000, WSQ, K-SVD.

INTRODUCTION

In the world today we are identified by the various biometric characteristics such as fingerprint recognition and in this paper we explain the fingerprint recognition based on sparse representation. Because in recent years there will be growing interest in the field of sparse representations of signals. Applications that use sparse representation are many that include compression, regularization in converse problems, feature extraction, and more. Among many biometric recognition technologies, fingerprint compression is very popular for personal identification due to the uniqueness, universality, collectability and invariance [1].

Large volumes of fingerprints are collected together and stored daily in a wide range of applications, includingforensics, access controland fingerprint square measure evident from

the information of Federal Bureau of Investigation (FBI). Large volume of data requires the large amount of memory. Fingerprint compression is a key technique to solve the problem.Compared with general normal images the fingerprint images contain simpler configuration. They are only composed of ridges and valleys. In the local regions, they seem to be the same. Therefore, to solve these two problemsthe pre-processing,pre-aligned the whole image is sliced into square and non-overlapping small patches.

Generally, compression technologies can be classed into lossless and lossy. The 8×8 small block of images. This transform has been used in JPEG [4]. The JPEG compression theme has several benefits likesimplicity, catholicity and accessibility. However, it has abad performance at low bit-rates mainly due to theunderlying block-based DCT format. For this motive, asearly as 1995, the JPEG-committee begins to develop anew wavelet-based compression for still images, especially JPEG 2000[5].

Targeted at fingerprint images, there are specialcompression algorithms. The most common is WaveletScalar Quantization (WSQ)[7]. It became the FBI standardfor the compression of 500 dpi fingerprint images. Motivatedby the WSQ algorithm, a few wavelet packet basedfingerprint compression schemes such as Contour let Transform (CT) have been developed. But,these algorithms have a common shortcoming namely, without the ability of knowledge. The fingerprint images can't be compacted well now. They will not be compressed well later. In this paper, ainnovative approach based on sparse representation is given [8]. The proposed method has the ability by updating the dictionary. The effects on actual fingerprint matching or recognition are not examined. In this paper, we will take it into contemplation. In most Automatic Fingerprint identification System (AFIS), the main feature used to match two fingerprint images are minutiae (ridges endings and bifurcations). Therefore, the difference of the minutiae among pre- and post-compression is considered in the paper.

RELATED WORKS

In this section, we describe the various image compression techniques and also we compare the proposed method withexisting fingerprint compression algorithms likeJPEG,JPEG-2000,WSQ,K-SVD etc.Generally, compression technologies can be classed into lossless and lossy.

Lossless compression is a type of image compression algorithms that allows the original data to be perfectly reconstructed from the compressed data. Typical image file formats of lossless compression are like PNG or GIF;It is also often used as a component within lossy data compression technologies. Lossless compression is used where it is essential that the original and the decompressed data be identical, or where deviations from the original data could be deleterious. Lossless compression methods may be classified according to the type of data they are designed to compress.

Lossy compression technologies usually transform an image into another domain, quantize and encode its coefficients. lossy compression is the type of image encoding methods that uses inexact approximations to represent the content. These methods are used to reduce data size for storage, handling, and transmit content. The amount of data reduction achievable using lossy compression is often much higher than through lossless techniques. In lossy

transform codes, samples of image are taken, chopped into small segments, transformed into a new basis space, and quantized. The resulting quantized values are then encoded using entropy coding.There are transform-based image compression methods have been implicitly researched and some principles have appeared. Two most common options of transformation are the Discrete Cosine Transform (DCT) [2] and the Discrete Wavelet Transform (DWT)[3].

JPEG

JPEG is a commonly used technique of lossy compression for digital images, particularly for those images which are produced by digital photography. Image files that use JPEG compression are generally called "JPEG files", and are stored in variants of the JIF image format. The term "JPEG" is an acronym for the Joint Photographic Experts Group.The DCT-based encoder is use for compression of a stream of 8×8 small block of images. This alterationhas been used in JPEG [4]. The JPEG compression technique has some benefits such as simplicity, universality and availability. However, it has a bad performance at low bit-rates because of the essential block-based DCT scheme. For this reason, as early as 1995, the JPEG-committee start to develop a new wavelet-based compression model for still images, namely JPEG 2000[5][6].

JPEG 2000

JPEG 2000 (JP2) is a type of image compression and coding system. It was created by the Joint Photographic Experts Groupin 2000 with the purpose ofsuppressing their original discrete cosine transform-based JPEG standard (created in 1992) with a newly designed, wavelet-based method.Thereis a modest increase in compression performance of JPEG 2000 compared to JPEG; the main advantage of JPEG 2000 is the significant elasticity of the codestream. The DWT-based algorithms consist of three steps: a DWT computation of the normalized image, quantization of the DWT coefficients and lossless coding of the quantized coefficients. The detail can be found in [7] [12]and. Compared with JPEG, JPEG 2000 provides many qualities that maintain scalable and interactive access to large-sized image. In additionit allows extraction of different resolutions, pixel fidelities, regions of interest, components etc. There are some other DWT-based algorithms, such as Set Partitioning in Hierarchical Trees (SPIHT) Algorithm. The aim of JPEG 2000 is not only improve compression performance over JPEG but also adding features such as scalability and editability. The improvement of JPEG 2000's in compression performance relative to the original JPEG standard is actually rather modest and should not ordinarily be the primary suggestion for evaluating the design. Very low and very high compression rates are supported in JPEG 2000. The capability of the design to handle a very large range of effective bit rates is one of the strengths of JPEG 2000. For example, to decrease the number of bits for aimage below a certain amount, the advisable thing to do with the image before encoding it. That is unnecessary when using JPEG 2000, because JPEG 2000 automatically does this through its multiresolution decomposition structure.

WSQ

The Wavelet Scalar Quantization algorithm (WSQ) is a compression algorithm used for gray-scale fingerprint images. It is based on wavelet theory and has become a standard for the replace and storage of fingerprint images. WSQ was developed by the Federal Bureau of

Investigation(FBI)[7] for the compression of 500 dpi fingerprint images.The WSQ compression technique developed by the FBI and alternative entities was designed to compress fingerprint pictures between ratios of 10:1 and 20:1. At these compression ratios, sufficient friction ridge and pore detail is maintained for the needs of identification, by fingerprint matching hardware and by human latentfingerprint examiners.This compression method is chosen over standard compression method like JPEG because at the same compression ratios WSQ doesn't present the "blocking artifacts" and failure of fine-scale features that are not suitable for identification in financial environments and criminal justice.

K-SVD

K-means clustering process.K-SVD is an iterative method that alternates between sparse coding of the examples based on the existing dictionary and a process of updating the dictionary atoms to better fit the data [9]. The update of the dictionarycolumns is grouped with an update of the sparse representations, thereby accelerating convergence. The K-SVD algorithm is expandable and can work with any pursuit method (e.g., basis pursuit, FOCUSS, or matching pursuit). We evaluate this algorithm and express its results both on synthetic tests and in applications on real image data.

PROPOSED SYSTEM IMPLEMENTATION

In this section, we give the details about how to use K-SVD algorithm for fingerprint compression based onspares representation. The part includes construction of the dictionary, compression of a given fingerprint, quantization and coding and analysis of the algorithm complexityas shown in figure (1).
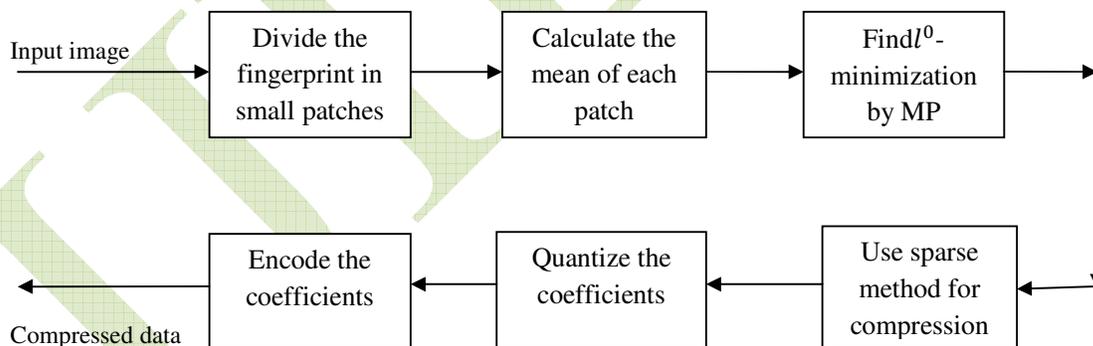
Input image → Divide the fingerprint in small patches → Calculate the mean of each patch → Find l^0-minimization by MP → Use sparse method for compression → Quantize the coefficients → Encode the coefficients → Compressed data

Fig.1. Proposed System Block Diagram.

Obtaining an overcomplete dictionary from a set of fingerprint patches allows us to represent them as a sparse linear combination of dictionary atoms.

Construction of the Dictionary

In this paper, the dictionary will be assembled in three ways. First, we construct a training set. Then, the dictionary is obtained from the set.Cut the whole fingerprint image in fixed size square patches as in figure (2).
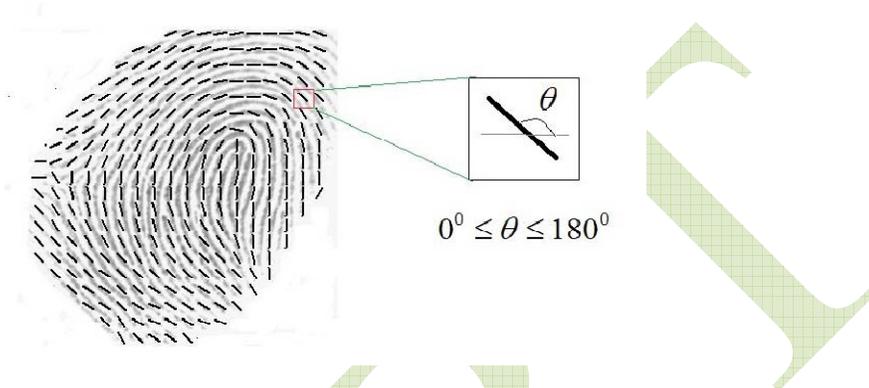

$$0^0 \leq \theta \leq 180^0$$

Fig.2.A fingerprint image with its fixed square patches

The specific process is as follows: create a base matrix whose columns characterize features of the fingerprint images, referring the matrix dictionary whose columns are called atoms; for a whole fingerprint, divide it into small blocks called patches whose number of pixels are identical to the dimension of the atoms use the method of sparse representation to obtain the coefficients then quantize the coefficients; last encode the coefficients and other related information using lossless coding methods. Greedy algorithm is used to construct training sample. Then the dictionary is obtained by the following steps.

- The first patch is added to the dictionary, which is initially empty.

- Then we check whether the next patch is adequately similar to all patches in the dictionary. If yes, then next patch is tested; otherwise, the patch is added into the dictionary. Here, the similarity measure between two patches is calculated by solving the optimization problem (1).

$$S\left(P_1, P_2\right) = \min_{t} \left\| \frac{P_1}{\left\|P_1\right\|_F^2} - t * \frac{P_2}{\left\|P_2\right\|_F^2} \right\|_F^2 \qquad (1)$$

Where $\left\|\bullet\right\|_F^2$ is the Frobeniusnorm. P_1 and P_2 are the corresponding matrices of two patches. t, a parameter ofthe optimization problem (1), is a scaling factor.

- Repeat the second step until all patches have been tested.

In the preceding paragraphs, it is mentioned that the dictionary will be constructed in three ways the first is select the patches at random and arrange them as columns of the dictionary (Random-SR).The second is to select patches according to orientations called Orientation-SR. See Figure (3), there are 100 patches with orientation 45 and size 20×20.
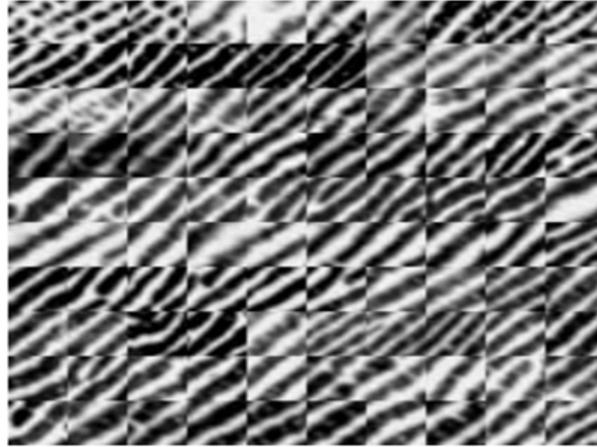

Fig.3.100 Patches with Size 20*20

The third is to prepare the dictionary by K-SVD method (K-SVD-SR in short)[9]. The dictionary is obtained by solving an optimization problem (4).Y is the training patches, A is the dictionary, X are the coefficients and X_i is the i th column of X. The dictionary for the K-SVD algorithm is constructed as follows.

Given $A = [a_1, a_2, \ldots\ldots\ldots\ldots\ldots, a_N] \in R^{(M*N)}$, any new sample $y \in R^{(M*1)}$ canbe represented as a sparse linear combination of few columns from the dictionary A, as shown in formula (2).This is the only prior knowledge about the dictionary in ouralgorithm. Later, we will see the property can be ensured byconstructing the dictionary properly.

$$Y = AX \qquad\qquad (2)$$

$$Where \ \ y \in R^{M \times 1}, A \in R^{M \times N}, \ and \ \ x = [x_1, x_2, \ldots\ldots x_N]^T \in R^{N \times 1}.$$

If M < N. and A is a full-rank matrix, an infinite number of solutions are available for the illustrationof problem, hence constraints on the solution must be set. The solution with the fewest number of nonzero coefficients is definitely an appealing representation.A proper solution canbe obtained by solving the following sparsest optimization problem:

$$(\ell^0): \qquad \min \|x\|_0 \qquad s.t. \qquad Ax = y \qquad\qquad (3)$$

Where $\|\bullet\|_0$ is the ℓ^0- norm, counting the nonzero entries of a vector.Solution of the optimization problem is very sparse if $\|x\|_0 << N$. The notation $\|x\|_0$ counts the nonzero entries in x actually it is not a norm. However, withoutuncertainty, we still call it ℓ^0-norm. In fact,

the compression of y can be achieved by compressing X. To solve the optimization problem ℓ^0 we should compute the coefficients matrix Xusing MP method, which guarantees that the coefficient Vector X_i has no more than T non-zero elements. Then, update each dictionary element based on the singular value decomposition (SVD) method.

$$\min_{A,X} \left\| Y - AX \right\|_F^2 \qquad s.t. \ \forall i \ \left\| X_i \right\|_0 < T \qquad (4)$$

In the following paper, the three kinds of dictionaries will be compared.

Compression of a Given Fingerprint

For the fingerprint compression based on sparse representation, thesize of patches and thesize of the dictionary arethe most important parameters. The size of patches has a direct impact oncompression efficiency. The larger the size is, the higher theefficiency is. However, to represent an arbitrary patch well, thesize of the dictionary needs to be sufficiently large. This causes more computational complexity.

There for the proper size of patch should be chosen. So far, there is no good way to estimate the parameter. For our proposed system implementation we can chose the 12×12, 16×16 and 20×20 sizes of patches. There are two reasons that the size 8×8 is not tested. The patches of this size are too small to contain the structure of fingerprints and it's difficult to compress such small patches at high compression ratios. From Figure(4), it is seen that the size 12×12 has better performance when the number of atoms in the dictionary is 4096.

In addition, to make the patches fit the dictionary superior, the mean of each patch is to be calculated and subtracted from the patch. After that, calculate the sparse representation for each patch by solving the ℓ^0 minimisation problem. Those coefficients whose absolute values are less than a specified threshold are treated as zero.
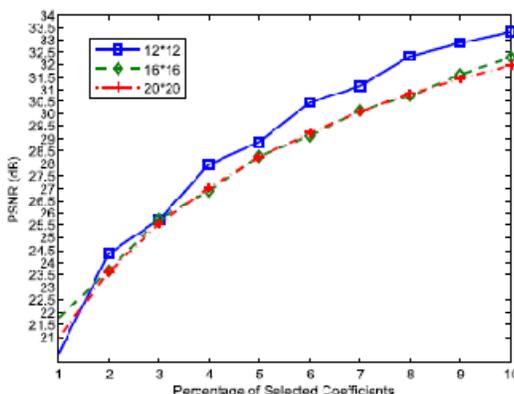

Fig.4. Average Performance ofthe Proposed Algorithms with Different Sizes of Patches.

For every patch, four kinds of information require to be recorded. They are the mean value, the number regarding how many atoms to use, the coefficients and their locations.

Quantization and Coding

Entropy coding of the atom number of each patch, the mean value of each patch, the coefficients and the indexes is calculated by static arithmetic coders [10]. The atom number of each patch is individually coded. The mean value of each patch is also individually coded. The quantization of coefficients is achieve using the Lloyd algorithm [11], The first coefficient of each block is quantized by means of a larger number of bits than other coefficients i.e. in proposed system the first coefficient is quantized with 6 bits and all other coefficients are quantized with 4 bits and entropy-codedwith a separate arithmetic coder. The model for the indexes is estimated by using the source statistics achieved off-line from the training set. The first index as well as other indexes is coded by the same arithmetic encoder.

Testing of the Algorithm Complexity

The algorithm includes two parts that is the training process and the compression process. Because the training process is off-line, simply the complexity of compression process is analyzed.

Algorithm 1 K-SVD algorithm for fingerprint compression based on sparse representation

---

1. For a given fingerprint cut it into small patches.
2. For each patch its mean is calculated and subtracted from the patch.
3. For each patch, solve the l^0-minimization problem by MP method.
4. Those coefficients whose absolute values are less than a given threshold are treated as zero. Record the remaining coefficients as well as there locations.
5. Encode the atom number of each patch, the mean value of each patch, and indexes; quantize and encode the coefficients.
6. Output the compressed stream.

Algorithm 1 summaries the entire compression process. The compressed stream doesn't include the dictionary and the information about the models. It consistsexclusively of the encoding of the atom number of each patch, the mean value of each patch, the coefficients and also the indexes. In practice, only the compressed stream needs to be transmitted to restore the fingerprint. In both encoder as well as the decoder, the dictionary, the quantization tables of the coefficients and the statistic tables for arithmetic coding need to be stored.

CONCLUSION

The different compression techniques are adapted to compress the fingerprint images are studied and compared their Performance especially at high compression ratios.New compression algorithm based on sparse representation in introduced.Due to the block-by-block processing mechanism, however, the algorithm has higher complexities. The paper shows that the K-SVD algorithm for fingerprint compression based on sparse representation

is more efficient than other compression technique such as JPEG, JPEG2000,WSQ. One of the main problems in developing compression algorithms for fingerprints resides in the needfor protection of the minutiae which are used in the identification. Our algorithmcan be able to hold most of the minutiae forcefully during the compression and reconstruction.

Further, we think the technique based on sparse representation do not work very well in the general image compression field. The reason are as follows: the content of the general images are so rich thereforethere is no proper dictionary under which the given image can be characterize sparsely; stillthere is one, the size of the dictionary may be too large to be computeefficiently. For example deformation, rotation, translation and the noise all can make the dictionary become very large. Therefore sparse representation should be employed in special image compression field in which there are no above shortcomings. The field of fingerprint image compression is one of them.

REFERENCES

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

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

[3] C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets andWavelet 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] 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.

[6] 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.

[7] 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.*

[8] Guangqi Shao, Yanping Wu, Yong A, Xiao Liu, and Tiande *Guo Fingerprint Compression Based on Sparse Representation, IEEE Trans. on Image Processing, vol. 23, no. 2, february 2014*

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

[10] K. Sayood, *Introduction to Data Compression, 3rd ed. San Mateo, CA, USA: Morgan Kaufman, 2005, pp. 81–115.*

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

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