

An Efficient Method of Fingerprint Compression based on Sparse Algorithm

Madhura Bhat¹

¹Department of CSE,
AMC Engineering College,
Bangalore

Jayashubha J²

Professor
²Department of CSE, AMC Engineering College,
Bangalore

Abstract— A new fingerprint compression technique based on sparse representation is introduced. This gives us an overcomplete and exhaustive dictionary from a set of fingerprint patches allowing one to represent them as a sparse linear combination of dictionary atoms. In this algorithm, the first step is to construct a dictionary for a predefined fingerprint image patches. For any new fingerprint image, represent its patches according to the dictionary by computing l_1 -minimization and then quantize using Lloyd's algorithm and encode using static arithmetic encoding. In this proposal, we consider effect of various factors on compression results. The algorithm is tested on a variety of fingerprint images. The experimental results demonstrate that the proposed algorithm is highly efficient when compared with several existing and competing compression techniques (JPEG, JPEG 2000, and WSQ), especially at high compression ratios. The experimental results and the simulations also demonstrate that the proposed algorithm is accurate in preserving the minutiae.

Keywords— Fingerprint compression, sparse representation, l_0/l_1 minimization, JPEG 2000, JPEG, WSQ, PSNR

I. INTRODUCTION

Identification of persons by biometric characteristics is a very important technology, because as biometric identifiers can't be shared with anyone and they represent the individual's unique bodily identity. Among many biometric recognition technologies, fingerprint recognition is very popular for personal identification due to its uniqueness, universality true, collectability and invariance [1]. Very large volumes of fingerprint images are collected and are stored every day in a wide range of applications such as forensics and identification access control. In 2005, the size of the CBI fingerprint card archive contained over 900 million items and archive size was increasing at the rate of 20 000 to 40 000 new entry cards per day [1]. Large volumes of data consume large amount of memory. Fingerprint image compression is a necessary technique to solve this issue. In general, image compression technologies can be classed into either lossless or lossy.

Lossless compression techniques allows the nearly exact original images to be reconstructed from the sparse compressed data. Many lossless image compression

technologies are used in cases where it is important that the original and the decompressed data are identical. Avoiding distortion limits their compression efficiency. When used in image compression where slight distortion is acceptable, lossless compression technologies are often employed in the output coefficients of lossy compression. Lossy compression type technologies usually transform a given image into another domain, quantize and encode using various technologies. During the last few decades, transform-based image compression technologies have been developed and extensively researched and some standards have been generated. Two of the most common options of transformation are the Discrete Cosine Transform (DCT) [5] and the most popular Discrete Wavelet Transform (DWT) [8]. The DCT-based bit encoder can be thought of as compression of a stream of 8×8 small equal sized blocks of images. This transform has been adopted in JPEG [4]. The JPEG lossy compression scheme has many advantages such as simplicity, universality and availability. However, it has a very bad performance at very low bit-rates mainly because due to the use of block-based DCT scheme.

For this reason the JPEG- expert group began to develop a new wavelet-based compression algorithm for digital still images, namely JPEG 2000 [5], [9]. The DWT-based algorithms include three steps: a DWT [6] computation of the normalized image [5], quantization of the DWT coefficients and lossless coding of the quantized coefficients [5]. The detail can be found in [2] and [4]. Compared to JPEG, JPEG 2000 [5] provides many features that support scalable and interactive access to large-sized image [3]. It also allows extraction of different resolutions [12], pixel fidelities [11], regions of interest, components [14] and etc. There are several other DWT-based algorithms [15], such as Set Partitioning in Hierarchical Trees (SPIHT) Algorithm [3]. The above algorithms are best suited for general image compression. Specifically targeted at fingerprint images, there are special compression algorithms [3]. The most common is Wavelet Scalar Quantization (WSQ) [3]. It became the FBI standard for the compression of 600 dpi fingerprint images [6]. Inspired by the WSQ algorithm [3], a few wavelet packet based [5] fingerprint compression algorithms have been developed. In addition to WSQ, there are various other algorithms for fingerprint compression, such as Contourlet Transform (CT) [11]. The above algorithms have a common and similar shortcoming, i.e without the

ability of learning. The fingerprint images are not compressed well now. They will not be compressed well later either. In this proposal, a new approach based on sparse representation is given. The proposed method has the unique ability by updating the dictionary. The process is as follows: construct a matrix whose columns represent features of the fingerprints, now referring the matrix dictionary where columns are called atoms; for a given fingerprint, divide it into small equal sized blocks called patches such that the number of pixels are equal to the size of the atoms; use the concept of sparse representation to obtain the coefficients results; later quantize the coefficients; lastly, encode them and other related information using lossless coding techniques [3].

In most places, the evaluation of compression performance of the algorithms is limited to Peak Signal to Noise Ratio (PSNR) evaluation. The effects on actual fingerprint matching or recognition are not at all investigated. In this proposal, we will take it into consideration [7]. In most Automatic Fingerprint identification System (AFIS) [13], the main feature used to match two fingerprint images are minutiae [14] (ridges endings and bifurcations). Hence, the difference of the minutiae between pre- and post-compression is mainly considered in the paper.

II. LITERATURE REVIEW

Early signs of its core ideas in sparse appeared in a pioneering work [10]. In that paper, the authors introduced the concept of dictionaries [14] and put forward some of the core ideas [12] which later on became highly essential in the fields such as a greedy pursuit. Later on S. S. Chen, D. Donoho and M. Saunders [15] introduced a similar pursuit technique which used l_0 -norm [3] for sparse. It is surprising that the proper [3] solution often [5] could be obtained by solving a linear convex programming task [7]. Since the two seminal works [9], researchers have contributed [7] a great deal in the field. The activity in this field is spread over various disciplines. There are already plenty of successful applications in various fields, such as face recognition [14], image denoising [12], object detection [12] and super high resolution image reconstruction [15]. In paper [17], the authors proposed a general classification algorithm [11] for object recognition based on a sparse representation computed by l_0 -minimization. At the same time, the algorithm based on sparse representation [5] has a better performance [12] than any other algorithms [4] such as nearest neighbor [4], nearest subspace [8] and linear SVM [4]; on the other hand, the a very new framework provided brand new insights into face recognition [13]: with sparsity features properly harnessed, the choice of features becomes very less important than the number of features. This phenomenon is very common in the fields of sparse representation [13]. It doesn't only exist in the face recognition application, but also appears in various other situations. In paper [4], features based on sparse and redundant representations brought on an over-complete dictionary [4], the authors designed an algorithm [2] that could remove the zero-mean white Gaussian and homogeneous Gaussian additive noise from a given image. In this paper [14], we can see that the content size of the dictionary is of extreme importance. The importance is

embodied [10] in two aspects [12]. On one hand, the dictionary must correctly reflect the content of the images; on the other hand, the dictionary is large enough [13] that the given image can be represented sparsely [5]. These two points are absolutely vital for the methods [13] based on sparse representation [3]. Sparse representation has already seen some applications in image compression [14], [11]. In paper [7], the experimental results show that the proposed algorithm has very good performance. However, its compression efficiency is consistently very lower than JPEG 2000's [15]. If more natural images are tested, this phenomenon will be more obvious [12] that the compression efficiency is lower than that of the state-of-the-art [12] compression technologies. In paper [8], the experimental results show success compared to several well-known compression techniques [3]. Anyhow, the authors emphasize that an essential pre-process [15] step for this method is an image alignment procedure [12]. It is very hard to do that in the practical application. There are other algorithms [1]–[2] for fingerprint image compression that comes under a linear model assumption. In paper [2], [21], the authors showed how to exploit the data independent nature of Independent Component Analysis (ICA) [15] to compression special images (face and fingerprint images). The experimental results of the two papers suggested that, for special class, it was not worth to use the over-complete dictionaries. In this paper, we show the fingerprints can be compressed better under an over complete and exhaustive dictionary if it is properly constructed as per standard procedures. In paper [9], the authors put forward an algorithm of fingerprint compression based on the popular Nonnegative Matrix Factorization (NMF) [2], [3]. Although NMF has some successful applications, it also has shortcomings. In some cases, non-negativity is not at all necessary. For example, in the image compression algorithm, what is considered is how to reduce the difference minutiae difference between pre- and post-compression rather than concentrating on nonnegativity [13].

Besides, the methods based on sparse representation don't work very well in the general image compression field. Some of the reasons are as follows: the contents of the general images are so high that there is no such proper dictionary under which the given image can be represented sparsely; even if there is one such, the size of the dictionary will be too large to be computed efficiently. For instance, the deformation, rotation, translation, cropping and the noise all can make the dictionary become too extra-large. Therefore, sparse concept representation must be employed in special image compression field in where there are no such above shortcomings. The concept of fingerprint image compression is one of them. Where we will give the reasons and the accounts for the feasibility of fingerprint compression based on sparse representation.

III. MODEL OF SPARSE REPRESENTATION

a. The Model

Given $A = [n_1, n_2, \dots, n_N] \in \mathbb{R}^{M \times N}$, any new sample $y \in \mathbb{R}^{M \times 1}$, is assumed to be represented as a linear combination of

Very few columns from the dictionary A [12], as shown in formula above. This is the only prior assumed knowledge about the dictionary in the sparse algorithm. Later, it will see the property can be ensured by constructing the dictionary properly. $y = Ax$, where $y \in \mathbb{R}^{M \times 1}$, $A \in \mathbb{R}^{M \times N}$ and $x = [x_1, x_2, \dots, x_N]^T \in \mathbb{R}^{N \times 1}$.

Obviously, the result $y = Ax$ is not at all determined when $M < N$. Therefore, its solution is not unique [13]. According to the assumption, the representation is sparse [14]. A proper solution can be yielded by solving the following optimization problem:

$$(10): \min \|x\| \text{ s.t. } Ax = y.$$

Solution of the optimization problem [4] is expected to be very sparse [13], namely, $\|x\| \ll N$. The notation $\|x\|$ counts the nonzero entries in x . actually it is a norm. However, without any ambiguity, we still call it norm. In fact, the actual compression of y can be achieved by compressing x first. First, record the locations of all of its non-zero entries and their magnitudes. Second, quantize and encode them. This is what sparse will do. Next, techniques adopted for solving the optimization problem are given.

b. L1 minimization

It is a natural idea that the optimization problem can be approximated by solving the below optimization problem:

$$(1p) : \min \|x\| \text{ s.t. } Ax = y$$

where $p > 0$ and $\|x\|_p^p = \sum_{i=1}^N |x_i|^p = 1$
 $|x_i|^p$.

Obviously, the smaller p is, the closer the solutions of the two optimization problems 10 and 1 p. This is because the magnitude of x is not all important when p is very small. What does matters is whether x is equal to 0 or not. Therefore, p is chosen as small as possible. However, the optimization problem is not convex [15] if $0 < p < 1$. It makes $p = 1$ the most ideal situation, namely, the following problems.

$$(11): \min \|x\| \text{ s.t. } Ax = y.$$

Recent developments in sparse coding and compressed sensing [21]–[24] reveal that the solution of the optimization problem [13] is equal to the solution of the second optimization problem if the optimal solution is sparse enough. The above problem can be solved by linear programming methods.

IV. METHODOLOGY

Here, in this section, we give all the details about how to use sparse coding representation to compress fingerprints. This part includes how to construct the dictionary, compression of a given fingerprint, quantization [13] and coding with the analysis of the algorithm complexity.

In the above paragraphs, it is mentioned that the size of the dictionary [3] will be too large when it contains as much information as possible [16]. Therefore, to obtain a dictionary

with a modest size, the preprocessing [13] is required. Influenced by the concept of transformation [12], rotation and noise, the fingerprints of the same finger image may look very different. What we first think is that each fingerprint image is pre-aligned [14], independently of the others. The most common pre-alignment [11] technique is to translate and rotate the fingerprint [11] according to the position of the core point [6]. Unfortunately, reliable detection [6] of the core is very difficult in fingerprint images with poor quality. Even if the core is correctly detected [1], the size of the dictionary may be overlarge [14] because the size of a whole fingerprint image is way too large. Compared with ordinary general natural images, the fingerprint images have very simpler structure [12]. They are only composed of ridges and valleys. In the local regions, they look the same [7]. Hence, to solve these two problems, the whole fingerprint image is sliced into square equal sized and non-overlapping small patches [8]. For these equal sized small patches, there are no problems about transformation [13] and rotation. The size of the dictionary is not too large because the small blocks [7] are relatively smaller [7].

a. Dictionary Construction

The dictionary is obtained from the dataset image. Choose the whole fingerprint images [5], cut them into fixed-size square patches.

- The first patch is added to the dictionary, which is initially empty [7].
- Then we check if the next patch is sufficiently similar to all patches in the dictionary [5]. If yes, the next patch is tested [6]; otherwise, the patch is added into the dictionary. Here, the similarity measure [6] between two patches is calculated by solving the optimization problem:

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

F is the Frobenius norm [6]. P_1 and P_2 are the corresponding matrices of two patches i.e p_1 and p_2 . t , a parameter of the optimization problem [5], is a scaling factor.

- Repeat the step 2 until all patches have been tested.

b. Compression

Given a new fingerprint, slice it into square patches [9] which have the same size. The size of each of the patches has a direct relationship on the compression efficiency [5].

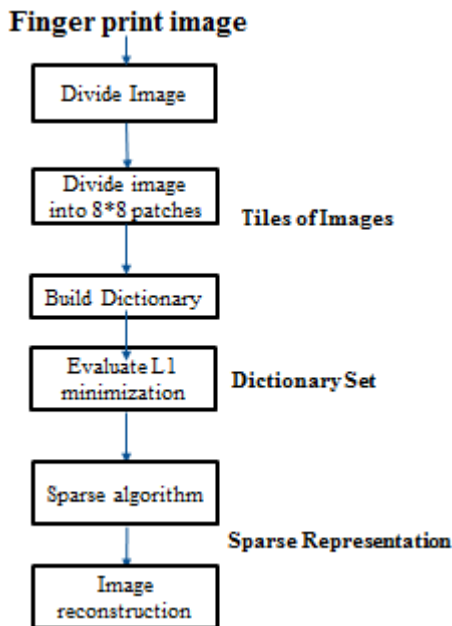
The algorithm becomes more efficient as the size increases. However, the computation complexity [4] and the size of the dictionary also increase rapidly [6]. The proper size should be chosen [7]. In addition, to make the patches fit [7] the dictionary better, the mean of each patch needs to be calculated and subtracted from the patch [3]. After that, evaluate the sparse representation for each patch by solving the 11 problem [8]. Those values whose absolute values are less than a given threshold [6] are treated as zero. For each patch, four kinds of information need to be recorded [7]. They are the mean value [9], the number about how many atoms to use, the coefficients and their locations [3]. The tests show that many image patches require few coefficients [6].

Consequently, compared with the use of a fixed number of coefficients [8], the method reduces the coding complexity and improves the compression ratio [7]

c. Encoding and Quantization

Entropy coding of each patch [6], the mean value of each patch [8], the coefficients and the indexes is carried out by static arithmetic coders [6]. The atom number of each patch is separately coded [3]. The mean value of each patch is also separately coded [5]. The quantization of coefficients is performed using the Lloyd algorithm [3], learnt off-line from the coefficients [6] which are obtained from the training dataset by over the dictionary [6]. The first coefficient of each block is quantized with a larger number of bits than other coefficients [5] and entropy-coded using a separate arithmetic coder [3]. The model for the indexes is estimated by using the source statistics [14] obtained off-line from the dataset. The first index and other remaining indexes are coded by the same arithmetic encoder. In the following experiments [6], the first coefficient is quantized with 6 bits and other coefficients are quantized [5] with 4 bits.

The system architecture is given below in Figure 1:



V. RESULT

The sparse coding algorithm is implemented using Matlab. The algorithm is tested on a dataset consisting of 80 images. The dataset consists of images which are bad, average and are good in terms of quality. The experimental results prove that the compressed images can hold most of the minutiae for all types of images. The highest compression ratio is found to be 36.4%.

The selected fingerprint image is shown in Figure 2. This fingerprint image is divided into 64 patches as shown in Figure 3. The original and the compressed fingerprint image are shown in Figure 4.

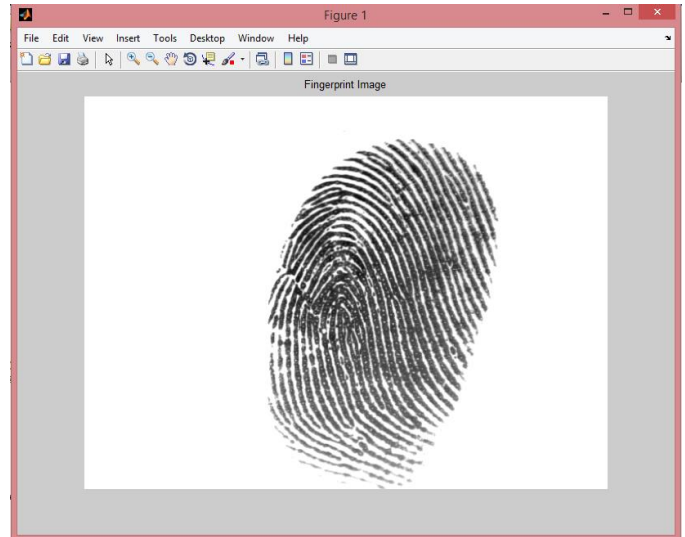


Fig.2 Fingerprint Image

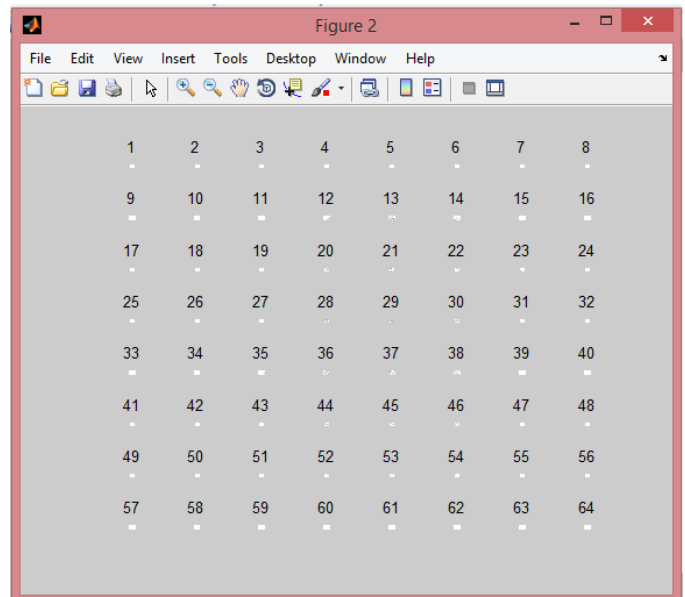


Fig. 3 Divided into 64 equal sized square patches



Fig.4 Original and compressed fingerprint images

VI. CONCLUSION

A new concept of compression algorithm adapted to fingerprint images is introduced and developed. Despite the complexity of the proposed algorithm, it compares favorably with existing more sophisticated algorithms such as WSQ, especially at high compression ratios such as 36.9%. Due to the patch-by-patch processing mechanism, the algorithm has higher complexities while implementing.

The experimental results show that the block effect of the algorithm is very less serious than that of JPEG or KSVD. One of the main challenges in developing compression algorithms for fingerprints resides in the need for preserving the minutiae which are used in the identification. The experimental results show that the algorithm can hold most of the minutiae robustly even during the compression and while reconstruction. The algorithm achieved a highest compression ratio of 36.9% with an accuracy of 92%, while preserving most of the minutiae during compression and while reconstruction.

ACKNOWLEDGMENT

The authors would like to thank the Department of CSE AMC Engineering College and the anonymous reviewers for their valuable comments and suggestions that helped greatly improve the quality of this paper.

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. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.
- [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] 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] 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.
- [9] 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.
- [10] 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.
- [11] S. G. Mallat and Z. Zhang, "Matching pursuits with timefrequency dictionaries," *IEEE Trans. Signal Process.* vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [12] S. S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
- [13] 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.
- [14] M. Elad and M. Aharon, "Image denoising via sparse and redundant representation over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [15] S. Agarwal and D. Roth, "Learning a sparse representation for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 113–127.
- [16] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
- [17] K. Skretting and K. Engan, "Image compression using learned dictionaries by RLS-DLA and compared with K-SVD," in *Proc. IEEE ICASSP*, May 2011, pp. 1517–1520.
- [18] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," *J. Vis. Commun. Image Represent.* vol. 19, no. 4, pp. 270–282, 2008.
- [19] 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.
- [20] 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.
- [21] 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.
- [22] P. Paatero and U. Tapper, "Positive matrix factorization: A nonnegative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 1, pp. 111–126, 1994.
- [23] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 799–791, Oct. 1999.
- [24] 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.
- [25] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [26] Guan qi Shao, Yanping Wu, Yong A, Xiao Liu. *Fingerprint Compression Based on Sparse Representation IEEE Transaction on Image Processing*, VOL. 23, no. 2, Feb 2014.