LEARNING DICTIONARY FOR EFFICIENT SIGNAL COMPRESSION

Afshin Abdi, Ali Payani, Faramarz Fekri

Georgia Institute of Technology

ABSTRACT

We consider the problem of learning dictionaries for data compression. Different from ordinary learning methods, the objective is to design a dictionary such that the signal has a low entropy representation in the basis of the dictionary, rather than giving a sparse or low-energy representation. To achieve this goal, we need to consider the effect of quantization on the rate-distortion curve as well as an estimation of the distributions of the coefficients. Based on this probability estimation, the coefficients are computed, quantized and then entropy-coded. As such, we have developed algorithms for different classes of dictionaries; orthonormal, union of orthonormals and general dictionaries with unit-norm atoms, to iteratively learn the dictionary and the distribution models of the coefficients. A mixture of Gaussians is adopted to estimate the probability and is updated using the expectation maximization algorithm together with the dictionary learning. Simulation results on the real seismic data show the effectiveness of the proposed algorithm compared to ordinary dictionary learning methods.

Index Terms— Dictionary Learning, Signal Compression, Low-Entropy Signal Representation

1. INTRODUCTION

Nowadays, modern sensing applications generate excessive amount of data which require a large bandwidth for transmission and a large repository to store. For example, a typical seismic survey may generate tens of terabytes of data. Therefore, compression at the sensor before transmission is emerging as a critical part in many data acquisition applications, especially in wireless systems.

Numerous algorithms have been developed for efficient signal compression. A large class of such algorithms relies on finding an appropriate transform to represent data more efficiently, in a way that discarding the 'least important' information in the transform domain has an insignificant effect on the quality of the data. Examples of such transforms include discrete cosine transform (DCT) and wavelets which have been successfully used for image compression in JPEG and JPEG2000 standards. In these methods, the dictionary is fixed and not adapted to a specific desired class of signals. In recent years, many algorithms have been proposed for designing signal-dependent (overcomplete) dictionaries, especially for the sparse representation. These methods have been shown to produce great results in applications such as denoising [1, 2], face detection [3], image classification and restoration [4, 5] and recently for image and video compression [6–8].

In this paper, the goal is to design dictionaries that are adapted to the available data and result in good compression performance, especially for the class of seismic signals. Our approach is different in the sense that the main objective is designing a dictionary and devising an algorithm to find a representation of the signal in the dictionary domain such that the representation requires less bits for compression. This is a departure from conventional dictionary learning methods, whose objective is to design dictionaries for sparse signal representation instead of directly optimizing for compression.

In the next section, we formulate the problem more rigorously and state the assumptions that help in designing a dictionary-based compression algorithm. In section III, the algorithms for different classes of dictionaries are presented and finally, through simulations, we demonstrate the effectiveness of the proposed approach.

1.1. Notations

We denote vectors by bold-faced lower-case letters and matrices by bold-faced upper-case letters. For a vector $\boldsymbol{a}$, its i-th entry is denoted by a_i and for a matrix $\boldsymbol{A}$, $\boldsymbol{a}_{:,i}$ or $\boldsymbol{a}_i$ represents the i-th column, $\boldsymbol{a}_{i,:}$ the i-th row of the matrix and $a_{i,j}$ the element at row i and column j. $\|\boldsymbol{A}\|_F$ is the Frobenius norm of matrix $\boldsymbol{A}$ and $\|\boldsymbol{a}\|_2$ is the ℓ_2-norm of vector $\boldsymbol{a}$. $\log_2$ denotes the logarithm in base 2 and $\ln$ is the natural logarithm. $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a multivariate Gaussian random variable with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

2. PROBLEM STATEMENT

In this paper, we investigate the problem of dictionary domain compression of data $\boldsymbol{X} = [\boldsymbol{x}_1\, \boldsymbol{x}_2\, \dots\, \boldsymbol{x}_N]$. It involves designing an appropriate dictionary $\boldsymbol{D}$, from a set of desired dictionaries $\mathcal{D} \subset \mathbb{R}^{n\times m}$, and an algorithm to find discrete coefficients $\boldsymbol{w}^{(q)}$ from a (possibly unknown) set $\mathcal{W} \subset \mathbb{R}^m$, such that $\boldsymbol{x} \approx \boldsymbol{D}\boldsymbol{w}^{(q)}$ and the rate to transmit $\boldsymbol{w}^{(q)}$ is as low as possible. Minimizing the total rate, R, of transmitting $\boldsymbol{W} = [\boldsymbol{w}_1\, \boldsymbol{w}_2, \dots, \boldsymbol{w}_N]$ such that the error remains less than a given threshold, ε, can be expressed as

$$\begin{aligned} &\min_{\boldsymbol{D}\in\mathcal{D},\boldsymbol{W}\in\mathcal{W}} R(\boldsymbol{W}) \\ &\text{s.t.} \quad \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W}\|_F^2 \le \varepsilon. \end{aligned} \tag{P1}$$

Alternatively, for a suitable parameter λ, we desire to

$$\min_{\boldsymbol{D}\in\mathcal{D},\boldsymbol{W}\in\mathcal{W}} \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{W}\|_F^2 + \lambda R(\boldsymbol{W}). \qquad \text{(P2)}$$

Note that we have ignored the cost of transmitting dictionary, $\boldsymbol{D}$, as it is constant and independent of the learning algorithm. In the limit, by increasing the number of data samples, the fixed cost of transmitting $\boldsymbol{D}$ becomes negligible compared to the rate of all coefficients. For example, in our simulations, it contributes to less than 10% of the total bit-rate. Or, as is the case in the memory-assisted compression [9], the dictionary is learned at both the encoder and decoder, via training the dictionary over available data in the common memory from past transmissions.

Two possible approaches to optimize the rate are

1. Directly optimizing for discrete values of $\boldsymbol{w}^{(q)}$. This requires optimizing for a discrete (finite or infinite) set $\mathcal{W}$ and a probability distribution $P(\cdot)$ on each element of $\mathcal{W}$. However, the increase in the number of parameters makes the optimization problem difficult to solve in most cases and may result in poor local minimum.
2. An alternative method would be to let the coefficients take any value, but use an (optimum) quantizer to discretize and then encode for compression.

In this paper, we focus on the second approach. The coefficients $\boldsymbol{W}\in\mathbb{R}^{m\times N}$ are found such that with an appropriate quantizer $Q(\cdot)$, $\boldsymbol{W}_q = Q(\boldsymbol{W})$, $\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{W}_q\|_F^2 \le \varepsilon$ and the rate $R(\boldsymbol{W}_q)$ is minimized.

2.1. Quantizer Design

Although it is possible to use Lloyd-Max [10, 11] approach to design a quantizer with minimum distortion, for any number of quantization bins, it does not necessarily give the minimum rate for the required distortion. On the other hand, entropy-constrained algorithms to design the quantizer are more complex to optimize and in our experiments with real data on seismic signals and at our desired SNRs (20-40 dB), they did not perform much better. Therefore, we decided to use uniform quantizers to discretize the coefficients. In addition to its simplicity, uniform quantizer has the advantages of (i)

1. The scalar uniform quantizer is asymptotically efficient under weak assumptions on the density function, i.e., its output entropy is asymptotically smaller than the rate of any other quantizer for the given distortion criteria [12].

2. The error due to the quantization is uniformly bounded for all values of $\boldsymbol{w}$, i.e., if $\boldsymbol{w}^{(q)} = Q(\boldsymbol{w})$, then $\|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{w}^{(q)}\|_2 \le \|\boldsymbol{x}-\boldsymbol{D}\boldsymbol{w}\|_2 + c$ where c is a constant that only depends on the quantization step-size δ and dictionary $\boldsymbol{D}$, independent of $\boldsymbol{w}$ and $\boldsymbol{x}$. This helps controlling the SNR of each individual block of data, $\boldsymbol{x}$.

3. In quantizing a continuous random variable z with small quantization step-size, δ, the rate is $R(z^{(q)}) \approx H_d(z) - \log_2\delta$, where $H_d(.)$ is the differential entropy. Therefore, optimizing for coefficients in $\mathbb{R}^{m\times N}$ and using a uniform quantizer adds only a constant to the cost function, independent of the final solution.

2.2. Modeling the Distribution of Coefficients

The next step is estimating the probability distribution of the coefficients to design a (near optimum) codebook and compute the rate. Although using non-parametric density estimation methods gives more flexibility, it requires more information to be transmitted to have the probability model at both encoder and decoder. Furthermore, optimizing for the distribution to reduce the rate is not an easy task and it has the potential of over-fitting the distribution. Hence, for the new data, the model may not be a good representative. Therefore, parametric density estimation is the method of choice for rate optimization.

In this paper, we used Gaussian Mixture Model (GMM) to estimate the distribution of coefficients, i.e., $p(\boldsymbol{w}) = \sum_{\boldsymbol{s}} \pi(\boldsymbol{s}) p(\boldsymbol{w}|\boldsymbol{s})$ where $\pi(\boldsymbol{s})$ is the weight of sources indexed by $\boldsymbol{s}$ and $p(.|\boldsymbol{s}) \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{s}}, \boldsymbol{\Sigma}_{\boldsymbol{s}})$ is a multivariate Gaussian distribution of the same dimension as $\boldsymbol{w}$. We choose GMM mainly because the proposed algorithm will have a closed form solution and becomes less complex. Furthermore, for large enough number of sources in the mixture, GMM results in the same distribution for the *quantized* coefficients, $\boldsymbol{w}^{(q)}$.

To further reduce the complexity of the compression algorithm, we encode each coefficient separately. This implies that the underlying distributions for w_i's, used in the compression, be independent, i.e., $p(\boldsymbol{w}) = \prod_i p_i(w_i)$. This is equivalent to assuming diagonal covariance matrices in the mixture model.

2.3. Outline of the Dictionary Learning Algorithm

As simultaneously optimizing for all parameters is not an easy task, we divide the optimization problem (P1) or (P2) into three steps such that at each step, only one set of parameters are updated, while the remainings are kept fixed.

As the coefficients are encoded separately, $R(\boldsymbol{W}) = \sum_{i=1}^{N} R(\boldsymbol{w}_i) = -\sum_{i=1}^{N} \log_2 p(\boldsymbol{w}_i)$. The rate can be upper bounded as $-\log p(\boldsymbol{w}_i) \le -\log \pi(\hat{\boldsymbol{s}}_i) - \log p(\boldsymbol{w}_i|\hat{\boldsymbol{s}}_i)$, where $\hat{\boldsymbol{s}}_i$ is the maximum a posterior (MAP) estimation of source index, $\boldsymbol{s}$, from the coefficients $\boldsymbol{w}_i$. This can be interpreted as the encoder at the transmitter estimates the source that have generated the coefficients and use the specific codebook designed for that source to compress the coefficients. This helps to further simplify the optimization problem. Thus, the general steps of iterative dictionary learning for compression can be summarized as following;

1. For the current coefficients, update dictionary $\boldsymbol{D}$ to minimize the error

$$\min_{\boldsymbol{D}\in\mathcal{D}} \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{W}\|_F^2. \qquad (1)$$

2. Fixing the distribution model $\boldsymbol{\Theta} = \{(\pi(\boldsymbol{s}), \boldsymbol{\mu_s}, \boldsymbol{\Sigma_s})\}$, and dictionary $\boldsymbol{D}$, the sub-problem of finding the best compressible coefficient for each data vector $\boldsymbol{x}$ is

$$\min_{\boldsymbol{s},\boldsymbol{w}} \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{w}\|_2^2 - \lambda \left(\log \pi(\boldsymbol{s}) + \log p(\boldsymbol{w}|\boldsymbol{s})\right), \quad (2)$$

or

$$\begin{aligned} \min_{\boldsymbol{s}} \max_{\boldsymbol{w}} \ & \log \pi(\boldsymbol{s}) + \log p(\boldsymbol{w}|\boldsymbol{s}) \\ \text{s.t.} \quad & \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{w}\|_2^2 \leq \varepsilon. \end{aligned} \quad (3)$$

3. Update the distribution to fit the coefficients and further reduce the bit rate.
4. Iterate the above steps until convergence.

In the next section, we consider different classes of dictionaries and address each of the above subproblems.

3. LEARNING DICTIONARY FOR COMPRESSION

In this section, we consider three different classes of dictionaries: orthonormal, union of orthonormal dictionaries, and general dictionaries with unit-norm atoms.

3.1. Updating Dictionary

As mentioned earlier, at each iteration of the algorithm, the dictionary is updated to minimize the error $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W}\|_F^2$ for a fixed $\boldsymbol{W}$. Here, we briefly review the adopted methods to update the dictionary in each desired class of dictionaries.

3.1.1. Orthonormal dictionary

It is well known that the orthonormal dictionary $\boldsymbol{D}$ that minimizes $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W}\|_F^2$ for fixed $\boldsymbol{X}$ and $\boldsymbol{W}$ is given by the singular value decomposition:

Lemma 1 (see e.g. [13]). *Let $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$ be the svd of $\boldsymbol{X}\boldsymbol{W}^T$. The orthonormal dictionary that minimizes $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W}\|_F^2$ is given by $\boldsymbol{D} = \boldsymbol{U}\boldsymbol{V}^T$.*

3.1.2. Union of Orthonormal Dictionaries

Let $\boldsymbol{D} = [\boldsymbol{D}_1\, \boldsymbol{D}_2\, \ldots\, \boldsymbol{D}_L]$ be the concatenation of L orthonormal dictionaries $\boldsymbol{D}_l$. By decomposing the coefficients $\boldsymbol{W}$ accordingly, the error can be rewritten as $\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W} = \boldsymbol{X} - \sum_{l=1}^{L} \boldsymbol{D}_l\boldsymbol{W}_l$. In this case, block coordinate relaxation methods [13, 14] have been used successfully to update the sub-dictionaries, one at a time. Let $\boldsymbol{E}_l = X - \sum_{k\neq l} \boldsymbol{D}_k\boldsymbol{W}_k$. Fixing all sub-dictionaries except $\boldsymbol{D}_l$, Lemma 1 can be used to find the optimum dictionary that minimizes the error, i.e., $\boldsymbol{D}_l = \operatorname{argmin}_{\boldsymbol{D}} \|\boldsymbol{E}_l - \boldsymbol{D}\boldsymbol{W}_l\|_F^2$.

3.1.3. Dictionaries with Unit-Norm Atoms

Assume that $\mathcal{D} = \{\boldsymbol{D} \in \mathbb{R}^{n\times m} : \|\boldsymbol{d}_i\|_2 = 1,\ i = 1, 2, \ldots, m\}$. Unlike the orthonormal case, there is no closed-form solution to update $\boldsymbol{D}$. However, different algorithms such as MOD and projection [15], K-SVD [16] and Lagrange dual [17] have been proposed to find the dictionary that minimizes the error.

As the complexity of the above methods is relatively high, we propose using the following simple step-by-step algorithm to update the dictionary atoms iteratively. Note that $\boldsymbol{D}\boldsymbol{W} = \sum_j \boldsymbol{d}_j\, \boldsymbol{w}_{j,:}$. Therefore, fixing all atoms of the dictionary except $\boldsymbol{d}_i$, the error will be $\boldsymbol{E}_i = \boldsymbol{X} - \sum_{j\neq i} \boldsymbol{d}_j\boldsymbol{w}_{j,:}$ and the optimum choice for $\boldsymbol{d}_i$ for the given coefficients is

$$\begin{aligned} \boldsymbol{d}_i^* = \operatorname*{argmin}_{\boldsymbol{d}} & \|\boldsymbol{E}_i - \boldsymbol{d}\boldsymbol{w}_{i,:}\|_F^2 \\ \text{s.t.} \quad & \|\boldsymbol{d}\|_2 = 1, \end{aligned}$$

whose solution is $\boldsymbol{d}_i^* = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2}$, where $\boldsymbol{v} = \boldsymbol{E}_i\boldsymbol{w}_{i,:}^T$.

Note that in the rate-constrained representation of the signal, (P1) or (P2), since the coefficients no longer minimize only the distortion term $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{W}\|_F^2$, they will not be orthogonal to the error, i.e. $\boldsymbol{E}_i\boldsymbol{w}_{i,:}^T \neq \boldsymbol{0}$ and $\boldsymbol{d}_i^*$ is usually well-defined.

3.2. Computing Coefficients

Recall that for a given source index $\boldsymbol{s}$ in the mixture model, the coefficients are found using (see (2))

$$\begin{aligned} \boldsymbol{w}^*(\boldsymbol{s}) &= \operatorname*{argmin}_{\boldsymbol{w}} \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{w}\|_2^2 - \lambda\big(\log \pi(\boldsymbol{s}) + \log p(\boldsymbol{w}|\boldsymbol{s})\big) \\ &= \boldsymbol{\mu_s} + (\boldsymbol{D}^T\boldsymbol{D} + \frac{\lambda}{2}\boldsymbol{\Sigma_s}^{-1})^{-1}\boldsymbol{D}^T(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\mu_s}), \end{aligned} \quad (4)$$

where $\boldsymbol{\mu_s}$ and $\boldsymbol{\Sigma_s}$ are the mean and covariance matrix of the multivariate Gaussian source determined by index $\boldsymbol{s}$.

The resulting minimum total cost is obtained as

$$\begin{aligned} \mathcal{J}(\boldsymbol{s}, \boldsymbol{w}^*(\boldsymbol{s})) = \lambda\Big[&(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\mu_s})^T \left(\lambda\boldsymbol{I} + 2\boldsymbol{D}\boldsymbol{\Sigma_s}\boldsymbol{D}^T\right)^{-1} \\ &(\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\mu_s}) + 0.5 \ln\det(\boldsymbol{\Sigma_s}) - \ln \pi(\boldsymbol{s})\Big]. \end{aligned} \quad (5)$$

For a given dictionary $\boldsymbol{D}$, finding the best sources $\boldsymbol{s}^*$ for (P2), which minimizes (5), requires enumerating all possible choices of $\boldsymbol{s}$ and comparing the values of the total cost function. Alternatively, for (P1), it requires an additional optimization step to find λ such that $\|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{w}^*(s)\|_2^2 \leq \varepsilon$ prior to comparing the rates for all possible choices of $\boldsymbol{s}$.

However, since the covariance matrix is assumed to be diagonal[1], for the orthonormal dictionary, the source index and optimum value of the i-th coefficient are obtained by

$$s_i^* = \operatorname*{argmin}_s \frac{(\mu_{i,s} - y_i)^2}{\lambda + 2\sigma_{i,s}^2} + \ln\frac{\sigma_{i,s}}{\pi_i(s)}, \quad (6a)$$

$$w_i^* = w_i(s_i^*) = \frac{\lambda\,\mu_{i,s_i^*} + 2\,\sigma_{i,s_i^*}^2\, y_i}{\lambda + 2\,\sigma_{i,s_i^*}^2}, \quad (6b)$$

where $\boldsymbol{y} = \boldsymbol{D}^T\boldsymbol{x}$, and $\mu_{i,s}$ and $\sigma_{i,s}$ are the mean and standard deviation of the s-th source in the GMM of the i-th coefficient.

For union of L orthonormal dictionaries, we can use block coordinate relaxation method to compute the coefficients efficiently. Let $\boldsymbol{D} = [\boldsymbol{D}_1\, \boldsymbol{D}_2\, \ldots\, \boldsymbol{D}_L]$, and $\boldsymbol{w}_l$ be the vector

[1] Since each coefficient is compressed separately, they should have independent probability models.

of coefficients corresponding to $\boldsymbol{D}_l$. Starting from an initial guess for $\boldsymbol{w}$ (using matching pursuit or pseudo-inverse of $\boldsymbol{D}$), at each iteration, we fix all coefficients except $\boldsymbol{w}_l$ and compute $\boldsymbol{y}_l = \boldsymbol{D}_l^T \left(\boldsymbol{x} - \sum_{k \neq l} \boldsymbol{D}_k \boldsymbol{w}_k \right)$. Then, $\boldsymbol{s}_l$ and $\boldsymbol{w}_l$ are updated by minimizing $\|\boldsymbol{y} - \boldsymbol{D}_l \boldsymbol{w}_l\|_2^2 + \lambda R(\boldsymbol{w}_l)$, given by (6).

3.3. Updating the Model

To update the models, the Expectation Maximization (EM) algorithm is exploited [18, 19]. At each iteration of the dictionary learning algorithm, few iterations of the EM algorithm is applied on each row of the coefficients to update the model.

4. SIMULATION RESULTS

We have evaluated the performance of our proposed method on real seismic traces available from [20] and [21], (referred to as DB1 and DB2 here). The data from each sensor was treated as a one-dimensional signal and was segmented into blocks of length n to create data vectors, $\boldsymbol{x}_i$s. In the simulations $n = 16$ and $n = 32$ were used. To model the distributions of the coefficients, different number of sources, K, in the Gaussian mixture model were tested and based on the results, we found out that there is no significant improvement in the performance by increase K beyond 5. So, here, only the results for $K = 5$ mixtures for each coefficient are presented.

The rate-distortion (R-D) curves of the proposed method is compared against standard DCT (for orthonormal dictionaries) and [DCT $\boldsymbol{I}$] (for over-complete dictionaries) and standard dictionary learning algorithms for sparse representation (LASSO-based optimization with different weights). To obtain the R-D curves, we ran all algorithms for different values of the parameters and used appropriate uniform quantizer to get the best compression rate for the SNR from 20 to 40 dB.

Figures 1 and 2 show the results for orthonormal dictionaries of size 32×32 on DB1 and DB2, respectively. As DB1 has more measurement noise, the performance gap (in Fig. 1) is not as significant as those of DB2 (in Fig. 2). Furthermore, since the existing sparse dictionary learning method is not specifically designed with the compression gain as the objective, it has a poor performance and sometimes, in our simulations we observed that it performs even worse than standard DCT-based methods. Similar results were obtained for dictionary size 16×16 which is omitted here.

In Fig. 3, the performance of the proposed algorithm for learning union of two orthonormal dictionaries ($n = 16$ and $m = 32$) is compared versus the [DCT $\boldsymbol{I}$] dictionary and sparse dictionary learning method, where the coefficients of [DCT $\boldsymbol{I}$] dictionary is computed via orthogonal matching pursuit [22].

5. CONCLUSION

In this paper, we have developed a dictionary learning algorithm such that the desired class of signals has a low-entropy representation in the basis of the dictionary and hence, is more suitable for compression. To do so, we first studied the effect of quantizer on the rate-distortion and concluded that for the real signals (seismic traces) and the desired range of SNR (20dB and above), the uniform quantizer performs close to optimum. To further simplify the compression algorithm, the encoding of each coefficient is done separately and in a memoryless manner. We used different Gaussian mixture models to approximate the distributions of the coefficients. Based on these assumptions, the dictionary learning algorithm consists of iteratively updating the probability models, computing compressible coefficients and updating the dictionary to reduce the distortion and rate. Simulation results showed that the proposed algorithm performs well for the compression of real seismic traces, outperforming ordinary learning methods for the sparsity and standard DCT transform.

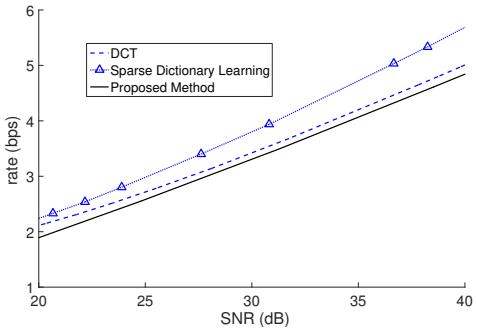

Fig. 1. R-D of dictionaries for database DB1, $n = 32$, $k = 5$

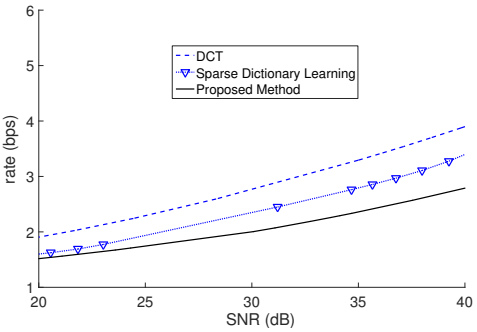

Fig. 2. R-D of dictionaries for database DB2, $n = 32$, $k = 5$

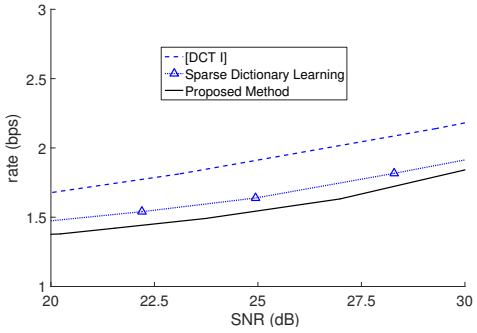

Fig. 3. R-D (per coefficient) for different dictionaries of size 16×32 and $K = 5$, applied on DB2

6. REFERENCES

[1] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec 2006.

[2] Y. M. Huang, L. Moisan, M. K. Ng, and T. Zeng, "Multiplicative noise removal via a learned dictionary," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4534–4543, Nov 2012.

[3] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 2691–2698.

[4] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *2009 IEEE 12th International Conference on Computer Vision*, Sept 2009, pp. 2272–2279.

[5] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 3501–3508.

[6] K. Skretting and K. Engan, "Image compression using learned dictionaries by RLS-DLA and compared with K-SVD," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 1517–1520.

[7] K. Kreutz-Delgado, J. Murray, B. Rao, K. Engan, T. Lee, and T. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, no. 2, pp. 349–396, Feb 2003.

[8] J. F. Murray and K. Kreutz-Delgado, "Sparse image coding using learned overcomplete dictionaries," in *Machine Learning for Signal Processing, 2004. Proceedings of the 2004 14th IEEE Signal Processing Society Workshop*, Sept 2004, pp. 579–588.

[9] A. Beirami and F. Fekri, "Memory-assisted universal source coding," in *Data Compression Conference (DCC), 2012*, April 2012, pp. 392–392.

[10] J. Max, "Quantizing for minimum distortion," *IRE Transactions on Information Theory*, vol. 6, no. 1, pp. 7–12, March 1960.

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

[12] H. Gish and J. Pierce, "Asymptotically efficient quantizing," *IEEE Transactions on Information Theory*, vol. 14, no. 5, pp. 676–683, Sep 1968.

[13] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning unions of orthonormal bases with thresholded singular value decomposition," in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 5, March 2005, pp. v/293–v/296 Vol. 5.

[14] S. Sardy, A. G. Bruce, and P. Tseng, "Block coordinate relaxation methods for nonparametric wavelet denoising," *Journal of computational and graphical statistics*, vol. 9, no. 2, pp. 361–379, 2000.

[15] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, vol. 5, 1999, pp. 2443–2446 vol.5.

[16] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov 2006.

[17] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds. MIT Press, 2007, pp. 801–808.

[18] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[19] J. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," International Computer Science Institute, Tech. Rep. TR-97-021, 1998.

[20] "UTAM seismic data library," http://utam.gg.utah.edu/SeismicData/ SeismicData.html, [Online].

[21] "USGS seismic data library," http://energy.usgs.gov/ GeochemistryGeophysics/, [Online].

[22] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec 1993.