THE FLEXIBLE SIGNATURE DICTIONARY

Faraz Barzideh, Karl Skretting, Kjersti Engan

Department of electrical engineering and computer science
University of Stavanger
Norway

ABSTRACT

Dictionary learning and Sparse representation of signals and images has been a hot topic for the past decade and aims to help find the sparsest representation for the signal(s) at hand. Typically, the Dictionary learning process involves finding a large number of free variables. Also, the resulting dictionary in general does not have a specific structure. In this paper we use the ideas from *Image Signature Dictionary* and *General overlapping frames* and proposed a *flexible signature dictionary*. We show that the resulting signatures capture the essence of the signal and can represent signals of their own type very well in opposed to signals of other types.

Index Terms— dictionary learning, signature dictionary, sparse representation

1. INTRODUCTION

1.1. Background

Dictionary learning and Sparse representation of signals and images have attracted much attention in signal and image processing community because of its applicability in various areas such as denoising, inpainting, compression and classification [1]. The focus of this paper is a structure imposed dictionary learning method in which the collection of free variables of the dictionary can be regarded as a signature of the training signal. A motivation for such an approach can be that signature dictionaries might be useful in a classification context, either by using the learned signatures as features, or by using the representation as features followed by a classifier.

Let x denote an $N \times 1$ signal vector and D an $N \times K$ matrix called dictionary. The columns of D are called the atoms of the dictionary. The element in row n, column k of D are denoted by $D(n,k)$. Letting $K > N$ would make the reconstruction equation, $x = Dw$ an underdetermined system of equations and D to an overcomplete or redundant dictionary. We wish to approximate x as a linear combination of a few of the atoms in D ($\hat{x} = Dw$), w is sparse $K \times 1$ coefficient vector. The solution for such problem can be formulated as:

$$w = \underset{w}{\operatorname{argmin}} \|x - Dw\|_2^2 \;\; s.t \;\; \|w\|_0 \leq s. \qquad (1)$$

Equation 1 has proven to be an NP-hard problem [2]. However, many methods have been proposed which can find a suboptimal solution for this problem. Some find the solution through greedy methods such as (Orthogonal) Matching Pursuit. Others relax the l_0-norm to l_1-norm and utilize convex optimization such as LASSO [1].

The right choice of the dictionary can lead to better representation of the signal, i.e. a more sparse representation with less residual. For finding a proper dictionary we can choose between prespecified or learned dictionaries.

Prespecified dictionaries based on known transforms like discrete cosine transform (DCT) or wavelet based methods [3, 4] usually inhabit some structure on the dictionary which sometimes allows for calculating the coefficients via fast methods and compact memory usage during computation. A problem with prespesified dictionaries is that they often are too general and hence may not produce good results for some applications.

An alternative approach is to train the dictionary for the problem at hand and hence represent a specific signal class more efficiently. Such approach can be seen in Method of Optimal Directions (MOD) by Engan et al. [5] and K-SVD by Aharon et al. [6]. More recent dictionary learning methods include online dictionary learning by Mairal et al. (ODL) [7], Recursive Least Squares Dictionary Learning Algorithm (RLS-DLA) proposed by Skretting et al. [8] and many other variants in later years.

Let X and W denote the concatenation of training vectors and sparse coefficients respectively, i.e. xs and ws are columns of X and W. For learning the dictionary, a training set representative for the application should be available. The dictionary learning task can in general be expressed as:

$$\{D, W\} = \underset{D,W}{\operatorname{argmin}} \sum_{i=1}^{M} \|x_i - Dw_i\|_2^2 \; s.t \; \|w_i\|_0 \leq s. \quad (2)$$

Where M is the number of training vectors. Equation 2 is an infeasible optimization problem and a common way to solve such a problem is to view it as nested minimization and divide it into two steps in a loop:

1. Fix D and find W using:

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

$$w_i = \text{argmin} \|x_i - Dw_i\|_2^2 \; s.t \; \|w_i\|_0 \leq s, \;\; 1 \leq i \leq M.$$

2. Fix W and update the dictionary D.

The first step is *sparse coding*. The second step is *dictionary updating* which is the focus of this paper.

Although using learned dictionaries has been advantageous in many applications compared to using prespecified dictionaries, they have downsides. For example, dictionary learning algorithms are not suitable for large problem sets i.e. large signal blocks. The reason is that there would be a large number of free variables in the dictionary to learn, which demands a very large training dataset in order to avoid overfitting. Another issue is lack of structure in the resulting dictionary. For some applications having a specific structure is highly desired. For example when representing image blocks, we would like to have the same corresponding representation for edges regardless of their orientation. Also when dealing with long periodic signals, shift invariant structures help limiting the number of free variables meaning that we can handle larger problems.

1.2. Proposed method in relation to previous work

Efforts have been made to overcome the shortcomings of learned dictionaries. For example imposing a specific structure on the dictionary. Among them the double sparsity method by Rubinstein et al. [9], Image-Signature-Dictionary (ISD) proposed by Aharon et al. [10], General overlapping frames by Skretting [11] and ILS-DLA by Engan et al. [12] are a few.

The proposed method is inspired from General overlapping frames by Skretting and ISD by Aharon and Elad. Similar to ISD, we want to find a signature for the training signal, reduce the number of free variables and also impose a structure on the dictionary. We want this signature to capture the essence of the signal. Our formulation grants us more flexibility than ISD, thus we call it the *Flexible Signature Dictionary* or FSD.

In the next section we derive the equations for the FSD. Then we use its similarity to MOD and introduce the mini-batch version, which we call mFSD. Section 3 shows the resulting signatures from ISD, FSD and mFSD and how they reconstruct different signal classes. Finally section 4 concludes this paper with a summary of the work and possible future extensions.

2. FLEXIBLE SIGNATURE DICTIONARY

2.1. Deriving the equations for FSD

The following section is dealing with the dictionary updating step. There are NK free variables in dictionaries learned by MOD, KSVD and many other learning techniques for a D of size $N \times K$. In this paper we want to reduce this number to Q free variables where $Q \ll NK$. We also want to imposes a structure on D similar to prespecified dictionaries while learning the free variables. For doing so we relate the elements of the dictionary in a linear fashion to these Q free variables.

Let q denote the *signature vector* containing Q free variables. The elements in the dictionary matrix, D, are related to the elements in q in the following way:

$$D(n,k) = \sum_{j}^{Q} g^j(n,k) q(j). \tag{3}$$

Where g^j, $1 \leq j \leq Q$ are usually *sparse* $N \times K$matrices which determine the places in D where the jth free variable is present as well as its weight. For example, for having a shift invariant structure, on a 4×8 dictionary D, the g^js will have a structure given by equation 4 where $g^j(n,k) = (g^{circ}(n,k) == j)$. Where $(a == b)$ is a logical expression evaluating to 1 if it is true and 0 if it is false.

$$g^{circ} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 3 & 4 & 5 & 6 & 7 & 8 & 1 \\ 3 & 4 & 5 & 6 & 7 & 8 & 1 & 2 \\ 4 & 5 & 6 & 7 & 8 & 1 & 2 & 3 \end{bmatrix} \tag{4}$$

The flexibility of the FSD is seen from the definition in equation 3. In ISD, part of the signature forms each dictionary atom, whereas in the formulation of equation 3, each dictionary element maybe a linear combination of several of the free variables in the signature q.

Using operator $vec(\cdot)$, which denotes a function reshaping a matrix into a vector by stacking it column by column, and defining

$$G = [vec(g^1), vec(g^2), \ldots, vec(g^Q)], \tag{5}$$

equation 3 can be written as:

$$vec(D) = Gq. \tag{6}$$

In other words, the dictionary, D, is related to the signature vector, q, through a mapping matrix, G.

Denoting $\hat{X}$ as the approximation of the training vectors, X, we can rewrite $\hat{X} = DW$ using operator $vec(\cdot)$,

$$vec(\hat{X}) = vec(DW). \tag{7}$$

The Kronecker product [13] is denoted by $\otimes$. Exploiting a property of the Kronecker product $vec(CXB^T) = (B \otimes C)vec(X)$, found in equation 2 of [13], equation 7 can be written as:

$$vec(\hat{X}) = (W^T \otimes I_N)vec(D) = (W^T \otimes I_N)Gq. \tag{8}$$

Where I_N is the identity matrix of size $N \times N$.

Minimizing the representation error $||X - \hat{X}||_F^2$ or equivalently $||vec(X) - vec(\hat{X})||_2^2$ with regard to q is the least

squares solution of $X = \hat{X}$ or $vec(X) = (W^T \otimes I_N)Gq$ which is found to be:

$$q = [G^T(WW^T \otimes I_N)G)]^{-1}G^T vec(XW^T). \quad (9)$$

Equation 9 is the closed form solution of the above learning problem. q is now a representation of the dictionary and after some iterations becomes the *signature* we were looking for. With this approach, the number of free variables has been reduced from NK to Q. Algorithm 1 summarizes our learning method:

Algorithm 1: FSD learning algorithm

Initialize q with a Gaussian random vector;
Get the set of training vectors X
for $i = 1$ *To* $MaxIter$ **do**
  $vec(D_i) = Gq_i$
  % Sparse coding step:
  Find W using D_i and X
  % Dictionary updating step:
  $A = WW^T$
  $B = XW^T$
  $q_i = [G^T(A \otimes I_N)G)]^{-1}G^T vec(B)$
end

2.2. Deriving the mini-batch version

As seen in algorithm 1, FSD and MOD have similar dictionary updating stages as they both alternated between sparse approximation and dictionary update stages. It is also possible to extend FSD to mini-batch version, mFSD, similar to how MOD can be extended to the mini-batch variant of RLS-DLA [8]. Doing so would make FSD capable of processing large amounts of data and also suitable for online learning.

In order to do so, we change A and B from algorithm 1 to

$$A_i = \lambda A_{i-1} + W_i W_i^T,\ A_0 = I_K \quad (10)$$

$$B_i = \lambda B_{i-1} + X_i W_i^T,\ B_0 = D_0 \quad (11)$$

where λ is a forgetting factor, X_i is the training set number i and W_i is its corresponding sparse coefficient matrix.

Algorithm 2 shows the learning process for mFSD.

Algorithm 2: mFSD learning algorithm

Initialize q with a Gaussian random vector;
$A_0 = I_K$
$vec(D_0) = Gq_0$
$B_0 = D_0$
for $l = 1$ *To* $MaxIter$ **do**
  for $i = 1$ *To* $MinibatchNo$ **do**
    Get the new mini-batch of training vector X_i
    % Sparse coding step:
    Find W_i using D_{i-1} and X_i
    % Dictionary updating step:
    $A_i = \lambda A_{i-1} + W_i W_i^T$
    $B_i = \lambda B_{i-1} + X_i W_i^T$
    $q_i = [G^T(A_i \otimes I_N)G)]^{-1}G^T vec(B_i)$
    $vec(D_i) = Gq_i$
  end
end

3. EXPERIMENTS

In this section we first show examples of signatures learned with FSD, mFSD and ISD from some synthetic signals normalized to have maximum absolute value of 1. Then we look into how these signatures reconstruct different classes of signals.

In all experiments a limit on the number of nonzero coefficients is enforced in the sparse coding step. We have implemented the ISD as described in chapter 12 of [1]. All dictionary learning implementations are performed in MATLAB by the authors. Sparse coding is done using orthogonal matching pursuit from the SPAMS software package [7].

The training vectors for each signal class are chosen by randomly extracting blocks from an example signal and adding Gaussian noise with variance 0.2. Finally these noisy blocks are concatenated to form the training matrix X. The elements of initial signatures are always chosen as random values form a Gaussian distribution with zero mean and unit variance. A shift invariant structure is used so we can compare FSD to ISD, and to visually evaluate the performance of the signature. However the FSD could also utilize other structures.

First, we show the trained signatures from FSD, mFSD and ISD for different signal classes and as can be seen from figures 1 to 3, all the methods capture the essence of their signal class which is what we hoped for. The shift invariant structure that was enforced as the structure of the dictionary is visible.

Then, the reconstruction capability of the signatures are tested on the same signal class as their training signals and also on other signals classes. In figures 4 and 5, reconstructions done by using sparse representation of noisy signals and dictionaries learned using FSD and mFSD are shown. In these experiments the signals are represented very sparsely by two different dictionaries, one trained for the same signal class, and one trained for a different signal class.

It can be seen that the signatures are able to reconstruct their own class of signal in a proper manner. For other signal classes, the reconstruction is poor. In fact this poor recon-

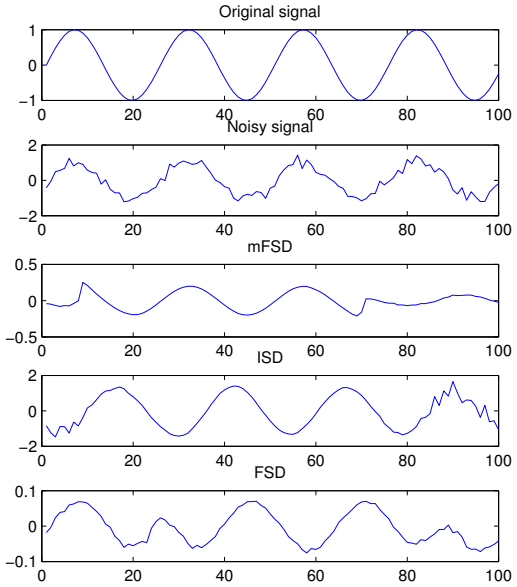

Fig. 1: Signatures learned for $\sin(t)$. Top two rows are the underlying signal and a noisy training vector. The three bottom rows are the signatures learned using mFSD, ISD and FSD.

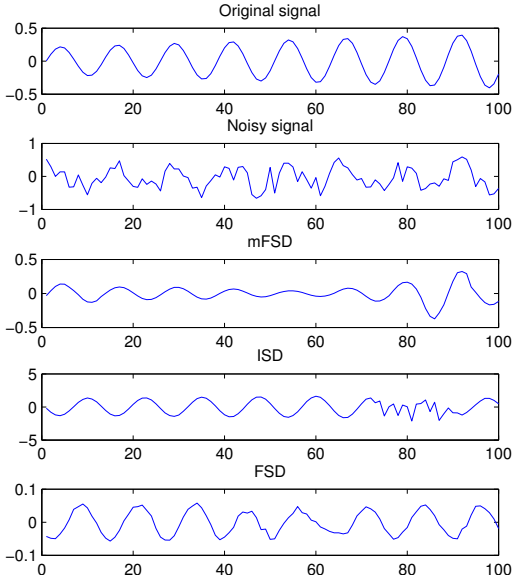

Fig. 2: Signatures learned for $t\sin(t)$. Top two rows are the underlying signal and a noisy training vector. The three bottom rows are the signatures learned using mFSD, ISD and FSD.

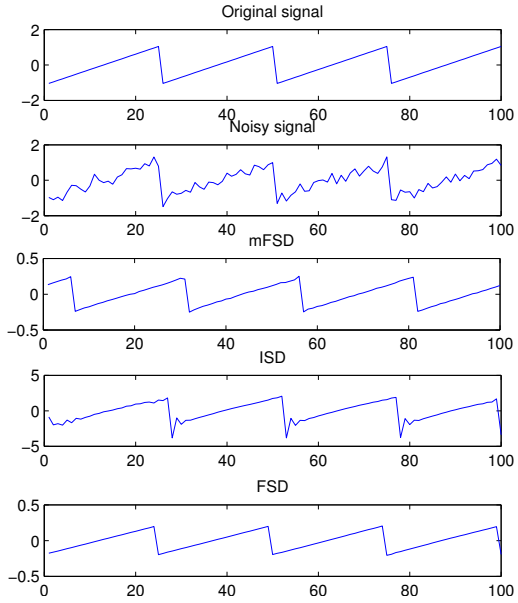

Fig. 3: Signatures learned for sawtooth signal. Top two rows are the underlying signal and a noisy training vector. The three bottom rows are the signatures learned using mFSD, ISD and FSD.

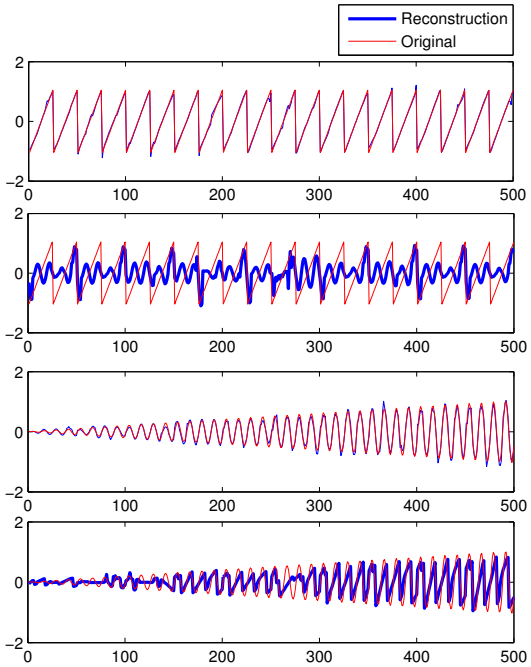

Fig. 4: Signal reconstruction using sawtooth and $t\sin(t)$ signatures from mFSD. It can be seen that each signature reconstructs its own signal very well while performs poorly on the other class

struction looks a lot like the signal of which the signature was trained on, which indicates that signatures produced by FSD and mFSD are capable of capturing the essence of the training signal, a reason for considering them for signal classification. Note that the representation was forced to be very sparse.

4. CONCLUSION AND FUTURE WORK

In this paper, a flexible signature dictionary was proposed. The resulting signature captures the essence of the signal. It has less free variables than the dictionaries resulting from

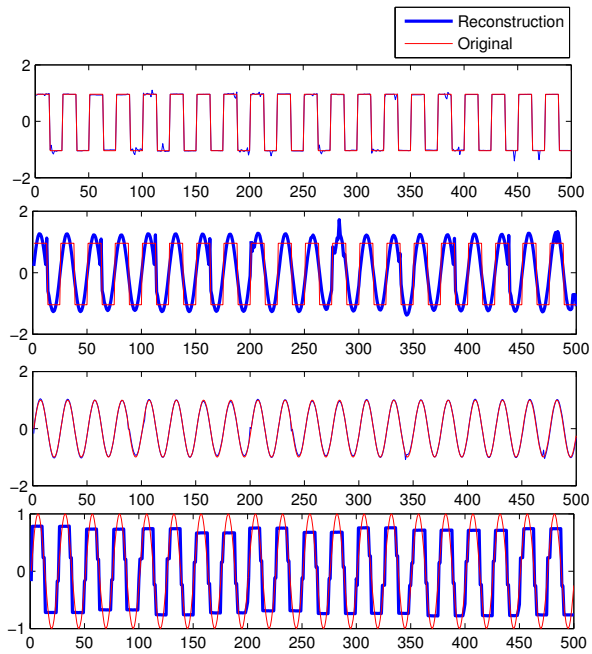

Fig. 5: Signal reconstruction using square and $\sin(t)$ signatures from FSD. It can be seen that each signature reconstructs its own signal very well while performs poorly on the other class

methods such as MOD and RLS-DLA. It also imposes a specific structure, for example a shift invariant property.Note that other classes of structures can also be enforced. The learning algorithm of the FSD has a compact algebraic formulation and the learning was easily extended to a mini-batch variant, mFSD.

By looking at the outcome of reconstruction of signals of different classes using the signatures, we showed that signal classification seems to be a possible future application for these methods. In future work we will experiment using the FSD on real world signals, and with different structures.

REFERENCES

[1] Michael Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, springer, 2010.

[2] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.

[3] R. R. Coifman and M. V. Wickerhauser, "Adapted waveform analysis as a tool for modeling, feature extraction, and denoising," *Optical Engineering*, vol. 33, no. 7, pp. 2170–2174, 1994.

[4] J. Portilla, V. Strela, M.J. Wainwright, and E.P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *Image Processing, IEEE Transactions on*, vol. 12, no. 11, pp. 1338–1351, Nov 2003.

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

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

[7] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA, 2009, ICML '09, pp. 689–696, ACM.

[8] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *Signal Processing, IEEE Transactions on*, vol. 58, no. 4, pp. 2121–2130, April 2010.

[9] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *Signal Processing, IEEE Transactions on*, vol. 58, no. 3, pp. 1553–1564, March 2010.

[10] M. Aharon and M. Elad, "Sparse and redundant modeling of image content using an image-signature-dictionary," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 228–247, 2008.

[11] Karl Skretting, *Sparse Signal Representation using Overlapping Frames*, Ph.D. thesis, University of Stavanger, 2002.

[12] K. Engan, K. Skretting, and J.H. Husøy, "Family of iterative LS-based dictionary learning algorithms,ILS-DLA, for sparse signal representation," *Digital Signal Processing*, vol. 17, no. 1, pp. 32 – 49, 2007.

[13] Charles F.Van Loan, "The ubiquitous Kronecker product," *Journal of Computational and Applied Mathematics*, vol. 123, no. 12, pp. 85 – 100, 2000, Numerical Analysis 2000. Vol. III: Linear Algebra.