Optimizing dictionary learning parameters for solving Audio Inpainting problem

Vaclav Mach and Roman Ozdobinski

***Abstract*—Recovering missing or distorted audio signal samples has been recently improved by solving an Audio Inpainting problem. This paper aims to connect this problem with K-SVD dictionary learning to improve reconstruction error for missing signal insertion problem. Our aim is to adapt an initial dictionary to the reliable signal to be more accurate in missing samples estimation. This approach is based on sparse signals reconstruction and optimization problem. In the paper two staple algorithms, connection between them and emerging problems are described. We tried to find optimal parameters for efficient dictionary learning.**

***Keywords*—Audio Inpainting, Dictionary Learning, K-SVD, Orthogonal Matching Pursuit, Signal reconstruction, Sparse Representations**

I. Introduction

Since the time audio recording and transmittion have been discovered, there is always a possibility of creation an error in the signal. In times of analogue sound carriers they were sensitive to scratching a gramophone record or tearing a magnetic tape. Digital audio carriers are more resistent to these damages mainly because of Error Correction Codes but still transmitting audio signal through e.g. IP telephony can be affected by packet loss.

Many problems connected with analogue audio recordings restoration and their digitized copies have been solved in the past by various techniques. For example distorted or missing signal recover was repaired by interpolation techniques [1], samples repetition, wavelet transform [2] or neural networks [3]. IP telephony is a common example of packet loss problem where data is lost during the transmition and they have to be recovered, e.g. by [4].

The recent research in the field of sparse signals representations has brought new approach that can be utilized for this audio restoration. Methods of solving a problem called Audio Inpainting were first mentioned in [5] as an algorithm for recovering missing samples in audio signal. This framework provides segmentwise restoration of audio signals where the position of missing samples is a-priori known. Sparse representation modeling uses Orthogonal Matching Pursuit method for solving an inverse problem. As an initial dictionary for sparse signals modeling, Gabor or DCT dictionaries are used.

More sparsity and the resulting better approximation of a missing signal can be achieved by choosing suited transform function to construct the dictionary or adapting initial dictionary to the given signal. We decided to go the second way and tried to adapt an initial dictionary with the K-SVD algorithm [6]. This method is a generalization of K-means clustering process and helps the dictionary to better fit observed audio data.

In section II sparse signals modeling together with Audio Inpaing as one of sparse signals modeling application are described. The dictionary learning via K-SVD algorithm is described in section III. Section IV shows our experimental results. Software components are described in section V and optimal parameters for dictionary learning which we found out are summed up in conclusion (section VI).

II. Audio inpainting by sparsity constraints

Sparse signal representations for inpainting problems were first used in image signal processing [7] and a few years later the Audio Inpainting algorithm was introduced in [5].

For some given signal $\boldsymbol{y} \in \mathbb{R}^N$ it is known, which samples of it are reliable and which ones are distorted or missing. Therefore, we can divide the support of the vector into the two sets $\mathcal{I}^r$ and $\mathcal{I}^m$ containing the coordinates of the reliable and missing samples, respectively. As a consequence of the partitioning we find $\mathcal{I}^r \cup \mathcal{I}^m = \{1, \ldots, N\}$ and $\mathcal{I}^r \cap \mathcal{I}^m = \emptyset$. By deleting the rows with indices in $\mathcal{I}^m$ from the $N \times N$ identity matrix we obtain the matrix $\mathbf{M}^{\mathrm{r}} \in \mathbb{R}^{|\mathcal{I}^r| \times N}$ selecting the reliable samples from the signal

$$\boldsymbol{y}^{\mathrm{r}} = \mathbf{M}^{\mathrm{r}}\boldsymbol{y}. \tag{1}$$

Using methods based on sparsity we have to find an appropriate function system to represent this class of signals by a few prototype functions only. The design of such a collection of atoms into which we can efficiently expand our signal, will be the central concern of this contribution. Given a set of atoms $\{\boldsymbol{d}_j\}_{j=1}^{M}$ one can build the dictionary matrix $D \in \mathbb{R}^{N \times M}$, where the columns of the matrix are the atoms. For a given coefficient vector $\boldsymbol{c}$ we can then compute the corresponding signal as a linear combination of the dictionary atoms by the matrix multiplication

$$\boldsymbol{y} = \mathbf{D}\boldsymbol{c}. \tag{2}$$

From now on, we will assume that $\mathbf{D}$ has a full rank and that $M \geq N$ with the consequence that for a given signal $\boldsymbol{y}$ we can always find a set of coefficients such that (2) is satisfied. Such an expansion can always be computed by the pseudo-inverse of $\mathbf{D}$ denoted by $\mathbf{D}^+$, i.e. $\boldsymbol{c} = \mathbf{D}^+\boldsymbol{y}$ satisfy (2).

An expansion $\boldsymbol{c}$ is called sparse, if only few entries of the vector are different from zero. We will denote the length of the support of the coefficient vector by $\|\boldsymbol{c}\|_0$. An expansion $\boldsymbol{c}$,

V. Mach is with the Department of Telecommunications, Brno University of Technology, Brno, Czech Republic, Purkynova 118, 612 00 Brno, e-mail: (vaclav.mach@phd.feec.vutbr.cz).

R. Ozdobinski is with the Department of Telecommunications, Brno University of Technology, Brno, Czech Republic, Purkynova 118, 612 00 Brno, e-mail: (xozdob00@stud.feec.vutbr.cz).

Manuscript received November 30, 2012; revised January 6, 2013.

which has only few entries significantly different from zero, is called compressible. It is quite clear that for a given dictionary only a certain class of signals will admit sparse expansions. In the later sections we will be concerned with audio signals and for this specific class a number of different dictionaries has been proposed, among them Gabor and DCT systems [5].

As we need these dictionaries later for performance comparison, we will define them now. For $0 \leq j < N$, we define the corresponding atom in the points $0 \leq m < N$

$$d_j[m] = \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)\left(j + \frac{1}{2}\right)\right]. \tag{3}$$

The resulting transform is purely real and will admit a certain level of sparsity for audio signals due to its harmonic structure. Gabor dictionaries can be constructed as DCT atoms with additional phase information. Due to the additional parameter, there is possibly a better fit to the signal, making the coefficients sparser. For some $0 \leq j < N$, $0 \leq n < N$ and $\varphi \in (0, 2\pi)$ we define

$$g_j[m] = \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)\left(j + \frac{1}{2}\right) + \varphi\right]. \tag{4}$$

Now we want to turn to the problem of audio inpainting, i.e. we want to reconstruct the missing samples in an audio signal. The problem we have to solve is: given the reliable samples $\boldsymbol{y}^{\mathrm{r}} = \mathbf{M}^{\mathrm{r}}\boldsymbol{y}$ reconstruct the full signal $\boldsymbol{y}$ under the assumption that $\boldsymbol{y}$ can be represented by a sparse set of coefficients. Incorporating this information, we formulate the following optimization problem

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \|\boldsymbol{y}^r - \mathbf{M}^r\mathbf{D}\boldsymbol{c}\|_2 \text{ s.t. } \|\boldsymbol{c}\|_0 \leq k_0. \tag{5}$$

Solving this NP-hard non-convex optimization problem is not feasible, therefore we will use orthogonal matching pursuit (OMP), a greedy algorithm, instead [8]. In the following we will describe the steps in detail in Algorithm 1 as it is not only used for the audio inpainting but also plays a major role in the K-SVD algorithm presented in Section III. This algorithm always yields a coefficient vector with $\|\boldsymbol{c}\| \leq s$ for some user determined sparsity parameter s.

Due to the length of the signals under investigation, one usually processes the data segment-wise. Therefore, the steps explained above have to be executed for every time slice and then one re-synthesizes from these individually obtained small portions. While for some pre-specified dictionaries (such as Gabor and DCT) there exist fast transforms, it would not be feasible to perform the steps described above for the dictionaries that we will construct in Section III.

III. Dictionary Learning

A. Overview

Static prespecified dictionaries like the introduced DCT or Gabor dictionaries are efficient because the transform process can be realized in a fast way. They are usually tailored to a specific group of signals. In the following we will discuss how to flexibly construct a dictionary adapted to the signal allowing for sparser representations.

Algorithm 1 Orthogonal matching pursuit

```
Input: $\boldsymbol{y}$... signal segment,
       $\mathbf{D}$... dictionary,
       $s$... level of sparsity,
       $\epsilon$... approximation error
Initialization: $i = 0$,
                $\boldsymbol{c}$ = zero vector of length $M$
                $\Omega_0 = \emptyset$... support set
Individual Steps:
Compute pseudo-inverse of dictionary $\mathbf{D}^+$
$R = \boldsymbol{y}$
while $\|R\|_2^2 > \epsilon$ and $i \leq s$ do
    $i = i + 1$
    Choose index $j$ with maximal absolute value in $\mathbf{D}^+R$
    Update support $\Omega_i = \Omega_{i-1} \cup j$
    Add the $j$-th entry of $\mathbf{D}^+_{\Omega_i}y$ to the $j$-th entry of $\boldsymbol{c}_k$
    $R = \boldsymbol{y} - \mathbf{D}_{\Omega_i}\boldsymbol{c}_k$
end while
Output: $\boldsymbol{c}$... sparse coefficients approximating $\boldsymbol{y}$
```

This process has got two main stages: first one is to make a training set of signals from reliable segments of the input audio data. These portions of samples are selected from the input signal following user specifications. In the second stage, an adapted dictionary is obtained from the learned data. Designing a new adapted dictionary brings additional computational burden because each dictionary atom has to be compared with the training data.

B. Brief history of Dictionary Learning

The idea of dictionary learning was first introduced in 1996 by Olshausen. This method was called Maximum Likelihood Method [9], other method called Method of Optimal Directions (MOD) was introduced in 2000 by Engan et al. [10]. Another approach that used Maximum A-Posteriori Probability was introduced by Engan (1999) [11] and Murray (2001) [12]. Few years later Lesage et al. presented Unions of orthonormal bases [13].

C. The K-SVD algorithm

This algorithm was first presented by Aharon et al. [6]. It is inspired by the K-means algorithm [14] solving the vector quantization problem. Vector quantization is a process where training examples are assigned to their nearest neighbors, each example is represented with just one coefficient and given the coefficients, dictionary $\mathbf{D}$ atoms are constructed. There is an obvious relation between sparse representations and quantization. Vector quantization is an extreme sparse representation when only one atom is allowed in the signal decomposition and this coefficient value must be 0 or 1.

As a predecessor and the closest algorithm of K-SVD is MOD (Method of Optimal Directions) which updates the whole dictionary in each learning iteration. The advantage of K-SVD is that it updates just one vector (atom) in each iteration and at the same time coefficients corresponding to this atom are updated, therefore the convergence is accelerated.

The goal of this algorithm is to adapt a dictionary $\mathbf{D}$ to represent the input signal $\boldsymbol{y}_k$ more sparsely by using any pursuit algorithm that approximates an optimization problem

$$\hat{\boldsymbol{c}}_k = \arg\min_{\boldsymbol{c}_k} \|\boldsymbol{y}_k - \mathbf{D}\boldsymbol{c}_k\|_2^2 \text{ s.t. } \|\boldsymbol{c}_k\|_0 \leq S_0. \qquad (6)$$

In this case we use OMP algorithm, which is described in Section II.

The K-SVD algorithm is described below (2), details can be found in [8][6].

Algorithm 2 K-SVD

```
Initialization: $\mathbf{D}^{(0)} \in \mathbb{R}^{n \times K}$,
                $J = 1$,
                $S_{\max}$... max. sparsity of vectors $\boldsymbol{c}_i$
Repeat until convergence (stopping rule):
Sparse coding
Solve using any pursuit algorithm
    $\min_{\boldsymbol{c}_k}\{\|\boldsymbol{y}_k - \mathbf{D}\boldsymbol{c}_k\|_2^2\}$ s.t. $\|\boldsymbol{c}_k\|_0 \leq S_{\max}$
Dictionary update
    For each atom $k = 1, 2, \ldots, K$ in $\mathbf{D}^{J-1}$ update by:
Set the group of indices using updated atom
    $\omega_k = \{i | 1 \leq i \leq N, \boldsymbol{c}_T^k(i) \neq 0\}$
Compute the error matrix $\mathbf{E}_k$ by
    $\mathbf{E}_k = \mathbf{Y} - \sum_{j \neq k} \boldsymbol{d}_j \boldsymbol{c}_T^j$
Restrict $\mathbf{E}_k$ choosing only columns corresponding to $\omega_k$
  to obtain $\mathbf{E}_k^R$
Apply SVD decomposition $\mathbf{E}_k^R = \mathbf{U}\Delta\mathbf{V}^T$
Update dictionary atom $\tilde{\boldsymbol{d}}_k$ (first column of $\mathbf{U}$)
Update coefficient vector $\boldsymbol{c}_R^k$ (first column of $\mathbf{V}$ multiplied
  by $\Delta(1,1)$)
Set $J = J + 1$
```

Every time the dictionary $\mathbf{D}$ is modified, it has to be checked whether it is ℓ_2-normalized.

There is no guarantee that the K-SVD algorithm can reach a global or even a local minimum. The purpose of this paper is to find basic approximated parameters for the convergence of audio signals dictionary learning using K-SVD algorithm. However, there is no possibility to check the convergence inside the algorithm, the only chance is to do it externally by comparing the results with another experiment.

IV. Experimental results

A. Optimizing K-SVD parameters

Audio inpainting presented in [5] was performed only with static dictionaries. That was the motivation for using the K-SVD algorithm to adapt the dictionary on the observed signal and therefore improve the reconstruction of missing signals. Software utilized for our experiments is specified in Section V. Audio Inpainting Toolbox also contains testing wave files with speech (sampling frequency $f_\mathrm{s} = 8\,\mathrm{kHz}$, $16\,\mathrm{kHz}$) and music ($f_\mathrm{s} = 16\,\mathrm{kHz}$). We have done several tests to obtain optimal parameters for dictionary learning and with these parameters we tried to compete our approach with static dictionaries. These tests were performed on one channel audio file *music07_16kHz.wav* with sampling frequency $f_\mathrm{s} = 16\,\mathrm{kHz}$ and length of 5 seconds.

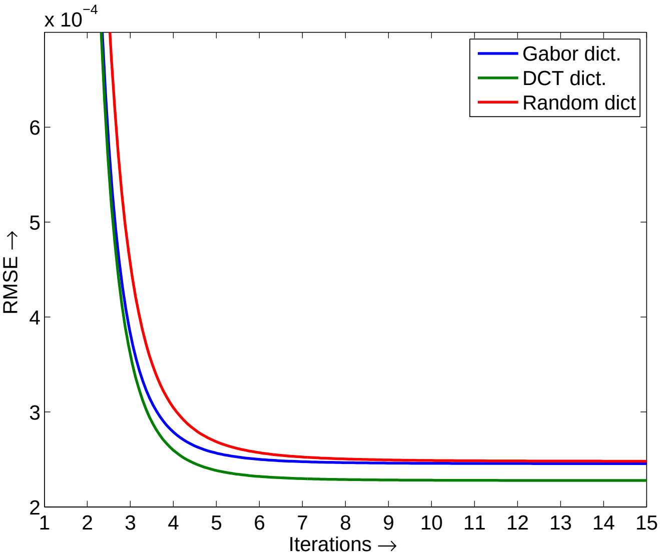

Fig. 1. RMSE according to number of iterations using different initial dictionaries.

After each iteration of dictionary learning via K-SVD algorithm, Root Mean Square Error (RMSE) is computed by

$$\mathrm{RMSE} = \sqrt{\frac{\|\mathbf{Y} - \mathbf{D}_{\mathrm{opt}}\mathbf{C}\|_F^2}{N \times M}}. \qquad (7)$$

After a few iterations RMSE settles at some value and remains unchanged. You can see in fig. 1 that four iterations are enough to reach satisfying RMSE value and after about 10 iterations RMSE is stabilized at its minimum. Because the lowest RMSE

Other experiment was focused on minimizing RMSE according to space between segments obtained from reliable samples to get the training data. If you have a short audio file and you do not have enough training segments of the signal, it has to be decided between smaller segment shift for more training data and larger segment shift for less training data. However, decreasing the segment shift is nothing but artificial enlarging the amount of training data and the samples are repeated in training segments. We got results presented in fig. 2. Using the audio file mentioned above of length 80 000 samples, we have the segment length of 256 samples, redundancy factor 3, therefore dictionary $\mathbf{D}$ has got a size of 256×768. With these parameters we can set up a shift of segments from interval $(1; 100)$. You can see that by increasing the segment shift value we get smaller RMSE during the dictionary learning process.

One of parameters of the dictionary learning explored further is the maximum number of nonzero coefficients $S_{\max}$. For $S_{\max} \in \{1, 2, 3, 4, 5\}$ dictionary learning experiments was made with focus on lowest RMSE depending on different $S_{\max}$ and therefore reaching the minimal error. In figure 3 it is obvious that after six iterations the minimal RMSE is reached by $S_{\max} = 3$ and remains minimal with very little change.

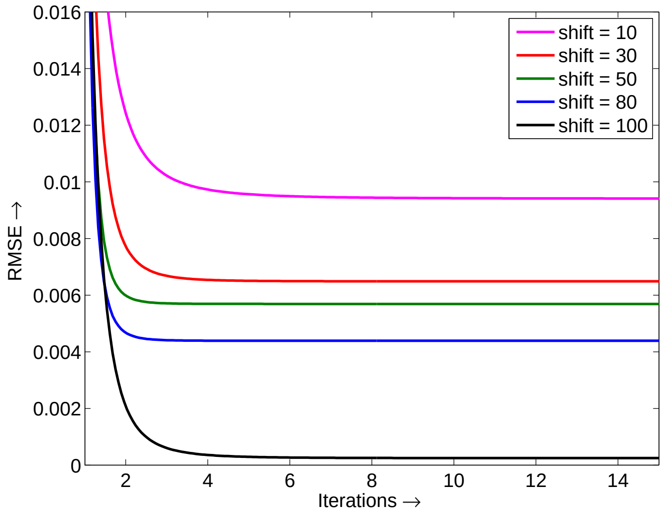

Fig. 2. RMSE according to shift of the original signal segmentation for training data.

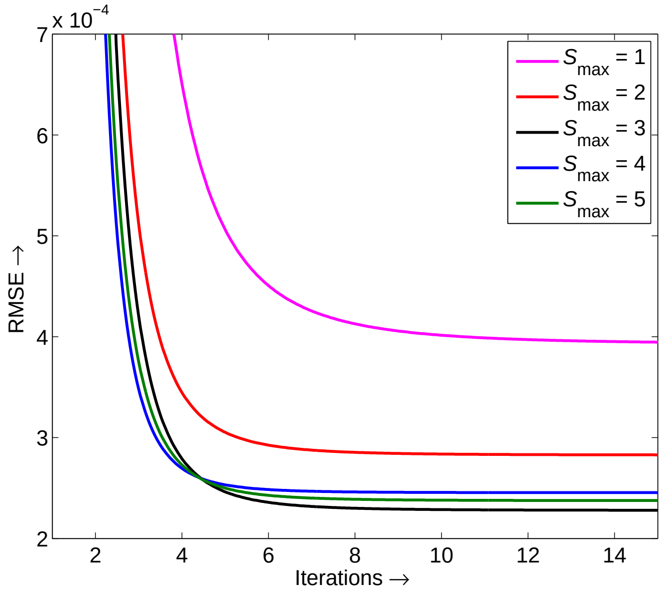

Fig. 3. RMSE according to max. number of non-zero coefficients during the dictionary learning.

Choosing the number of iterations has to be set up deliberately, since the number can be small and RMSE will remain high (the dictionary is not adapted as much as it can be) or the number can be too high and after reaching the minimum RMSE the algorithm can waste the time with new iterations or worse the RMSE can go up. That is why another experiment observing RMSE was performed with best parameters obtained above. Figure 4 shows that satisfying RMSE can be obtained with three or four iterations. This test was done for number of iterations from interval $(1; 200)$. For different signals number of iterations for settling, the RMSE can be various and during our experiments we used number of iterations of 50.

After settling the RMSE value on its minimum values were oscilating, therefore for all of the plots above, Matlab Curve Fitting Toolbox was utilized to make a trendline (approximation by quadratic function).

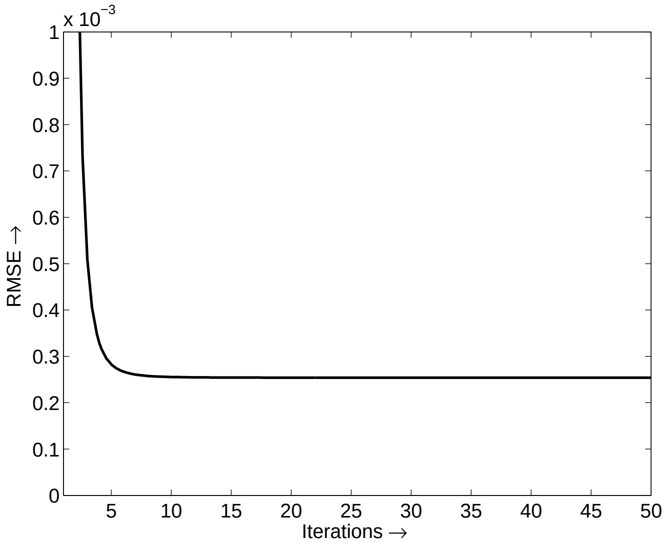

Fig. 4. RMSE according to number of iterations.

TABLE I
K-SVD PARAMETERS

Segment length N	256 samples
Initial dictionary $\mathbf{D}$	Random
Number of iterations	50
Max. number of non-zero coefs. $S_{\max}$	3
Segment shift	100 samples

B. *Dictionary comparison on real signals inpainting*

Now we will show the comparism of audio inpainting results of different sound files and utilizing various dictionaries. Both static (DCT and Gabor) and trained versions of these dictionaries were used to compare reconstruction results. The redundancy of all the dictionaries is 3. The parameters of the K-SVD dictionary learning algorithm are summarized in table I. The initial dictionary for K-SVD learning was filled with random values because during our experiments we got the most satisfying dictionary learning process. In each of audio files a gap (sequence of samples with zero value) is made with the size from 1 to 240 samples and evaluation of the signal reconstruction process is computed as Signal-to-Noise Ratio (SNR) only for missing samples by

$$\mathrm{SNR}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = 10 \log \frac{\|\boldsymbol{y}(\boldsymbol{I}^{\mathrm{m}})\|_2^2}{\|\boldsymbol{y}(\boldsymbol{I}^{\mathrm{m}}) - \hat{\boldsymbol{y}}(\boldsymbol{I}^{\mathrm{m}})\|_2^2}. \tag{8}$$

Our first experiments were made with male (*male04_16kHz.wav*) and female voice (*female04_16kHz.wav*) speaking English. The hole was generated starting at $6\,000^{\text{th}}$ sample. In figure 5 you can see female voice reconstruction results. The best SNR values were obtained using Gabor dictionary and as you can see the trained dictionary (supposed to better approximate the input signal) is in some cases worse than static dictionaries.

Figure 6 shows the reconstruction results of male speech and you can see that results of different dictionaries are almost

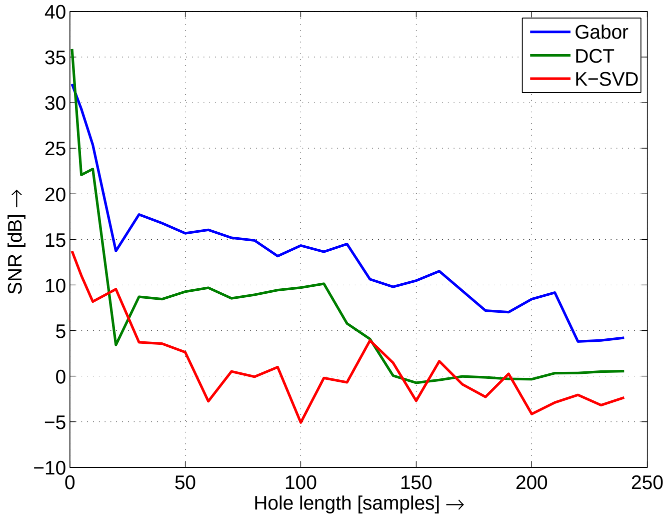

Fig. 5. Female speech missing signal reconstruction.

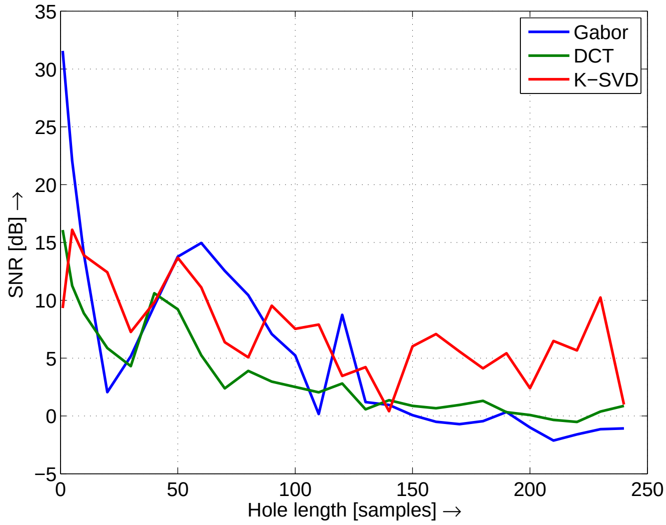

Fig. 6. Male speech missing signal reconstruction.

the same, but starting 160 samples hole length the trained dictionary overcomes static dictionaries.

Other experiments were done for audio files containing music samples. The gap was generated with $33\,000^{\text{th}}$ sample starting. First music file (*music06_16kHz.wav*) contains woman voice singing and the reconstruction error is shown in figure 7. Here, Gabor and trained dictionary are overcoming the DCT dictionary and K-SVD trained dictionary looks like to be more stable in larger gaps.

A sample of drums playing is recorded in (*music07_16kHz.wav*) file. In figure 8 you can see that all the dictionaries produce more or less the same reconstruction results in the sense of SNR.

Last experiment was performed with guitar playing music sample (*music11_16kHz.wav*). Figure 9 shows that for gap length from 40 to 110 K-SVD trained dictionary strongly overcomes static dictionaries of about 10 dB. Static dictionaries reconstruction results are almost the same.

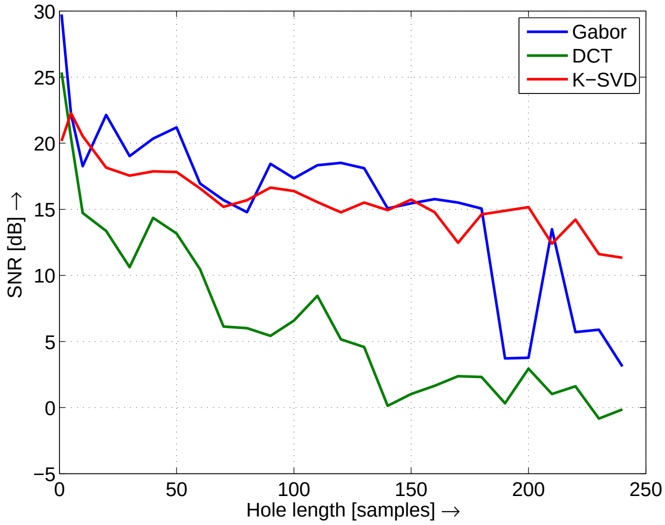

Fig. 7. A woman singing missing signal reconstruction.

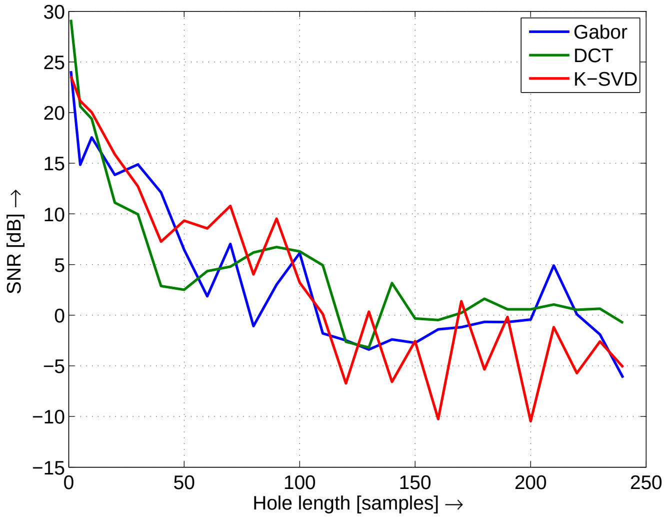

Fig. 8. Drums playing missing signal reconstruction.

V. Software

For Audio Inpainting process we used the Audio Inpaiting Toolbox (http://small-project.eu/keyresults/audio-inpainting) and the dictionary learning was done by SMALLbox v. 1.9 (http://small-project.eu/software-data/smallbox/).

These toolboxes are freely downloadable from the given links. Source m-files for reproducing our experiments were created on functions from Audio Inpainting Toolbox and can be downloaded from [15].

Files are created and may be run by using MATLAB. It has to be noted that results obtained by using random dictionary could differ because every time you run *randn()* function in MATLAB, you get a new matrix.

VI. Conclusion

In this paper, we presented a connection of two techniques to improve the reconstruction of missing audio signal infor-

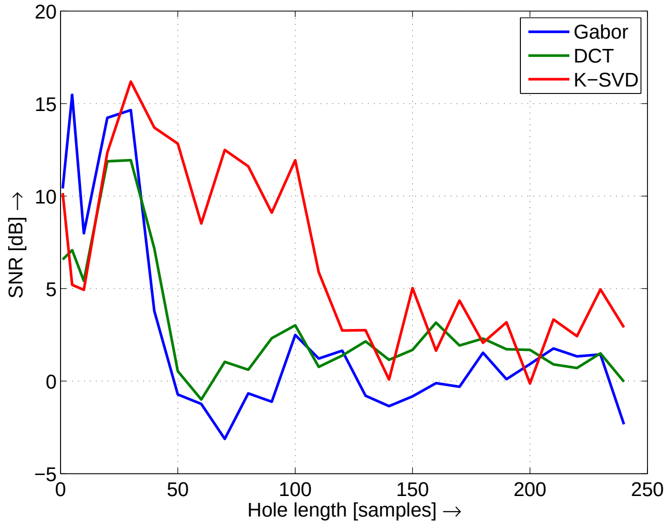

Fig. 9. A guitar playing missing signal reconstruction.

mation. Solving the Audio Inpainting problem was done by Orthogonal Matching Pursuit algorithm and adapting of the dictionary was processed by the K-SVD algorithm. Both of them are described in the text. The adapted dictionary was compared with static DCT and Gabor dictionaries.

Our aim was to find optimal parameters for dictionary learning via K-SVD and compare these results with static dictionaries. Construction of the building blocks of the dictionary is signal dependent and there are no general rules for setting up the parameters. In most cases the trained dictionary overcomes static dictionaries but it has to be taken into account the higher computational load. However, there are cases, where the reconstruction error of trained dictionary was the worse among these three dictionaries. Using real signals we have done several tests which are presented in the paper.

Future work will be focused on construction of the dictionary atoms and moving this problem to time-frequency plane. Knowledge presented in the paper will be utilized for solving real problem of old traditional music recordings which have signal gaps surrounded in high noise level.

References

[1] W. Etter, "Restoration of a discrete-time signal segment by interpolation based on the left-sided and right-sided autoregressive parameters," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1124 –1135, may 1996.

[2] P. Rajmic and J. Klimek, "Removing crackle from an LP record via wavelet analysis," in *Proceedings of the 7th international conference on digital audio effects DAFx04*, 2004, pp. 100–103. [Online]. Available: http://dafx04.na.infn.it/WebProc/Proc/P_038.pdf

[3] G. Cocchi and A. Uncini, "Subbands audio signal recovering using neural nonlinear prediction," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, vol. 2, 2001, pp. 1289 –1292 vol.2.

[4] C. Rodbro, M. Murthi, S. Andersen, and S. Jensen, "Hidden markov model-based packet loss concealment for voice over ip," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1609 –1623, sept. 2006.

[5] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. Plumbley, "Audio inpainting," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 922 –932, March 2012.

[6] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representations," *IEEE Transactions on Signal Processing*, vol. 54, pp. 4311–4322, 2006.

[7] M. Elad, J. Starck, P. Querre, and D. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (mca)," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 340–358, 2005.

[8] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.

[9] B. Olshausen and D. Field, "Natural image statistics and efficient coding," in *Computation in Neural Systems*, vol. 7, January 1996, pp. 333–339.

[10] K. Engan, S. O. Aase, and J. H. Husy, "Multi-frame compression: theory and design," *Signal Processing*, vol. 80, no. 10, pp. 2121 – 2140, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168400000724

[11] K. Engan, B. D. Rao, and K. Kreutz-Delgado, "Frame design using FOCUSS with method of optimal directions (MOD)," in *Proc. NORSIG 1999, Tronheim, Norway*, June 1999, pp. 65–69.

[12] J. F. Murray and K. Kreutz-Delgado, "An improved focuss-based learning algorithm for solving sparse linear inverse problems," in *Conference Record of the 35rd Asilomar Conference on Signals, Systems and Computers*, November 2001.

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

[14] A. Gersho and A. Gray, *Vector Quantization and Signal Compression*, ser. The Kluwer International Series in Engineering and Computer Science. Springer-Verlag GmbH, 1992. [Online]. Available: http://books.google.cz/books?id=DwcDm6xgItUC

[15] V. Mach and R. Ozdobinski. (2012) Source codes for audio inpainting with dictionary learning. [Online]. Available: http://www.stud.feec.vutbr.cz/∼xmachv00/paper2012/

Acknowledgment

The described research was performed in laboratories supported by the SIX project; the registration number CZ.1.05/2.1.00/03.0072, the operational program Research and Development for Innovation and supported by project of Brno University of Technology NT13499-4/2012 and the project reg. no CZ.1.07/2.3.00/20.0094.

Vaclav Mach received the B.Sc at the Department of Telecommunications at the Faculty of Electrical Engineering and Communication at Brno University of Technology in 2009. He received Ing. (M.Sc) at the same department in 2011. Now he is a Ph.D student at the same faculty.

The area of his professional interests is utilizing sparse representations in audio signal processing, old recordings restoration and acoustics. He is a student member of Audio Engineering Society.

Roman Ozdobinski received the B.Sc at the Department of Telecommunications at the Faculty of Electrical Engineering and Communication at Brno University of Technology in 2012. Currently he is a master student at the same department. His interests has been concentrated in audio signal processing, mainly sparse representations.