Sparse Representation with Temporal Max-Smoothing for Acoustic Event Detection

Xugang Lu[1], *Peng Shen*[1], *Yu Tsao*[2], *Chiori Hori*[1], *Hisashi Kawai*[1]

1. National Institute of Information and Communications Technology, Japan
2. Research Center for Information Technology Innovation, Academic Sinica, Taiwan

Abstract

In order to incorporate long temporal-frequency structure for acoustic event detection, we have proposed a spectral patch based learning and representation method. The learned spectral patches were regarded as acoustic words which were further used in sparse encoding for acoustic feature representation and modeling. In our previous study, during feature encoding stage, each spectral patch was encoded independently. Considering that spectral patches taken from a time sequence should keep similar representations for neighboring patches after encoding, in this study, we propose to enhance the temporal correlation of feature representation using a temporal max-smoothing algorithm. The max-smoothing tries to pick up the maximum response in a local time window as the representative feature for detection task. We tested the new feature for automatic detection of acoustic events which were selected from lecture audio data. Experimental results showed that the temporal max-smoothing significantly improved the performance.

Index Terms: Feature learning, matching pursuit, temporal max-smoothing, acoustic event detection.

1. Introduction

In real acoustic environments, besides speech event, many other types of acoustic events exist. Detecting and locating their temporal boundaries are important for many speech technology applications. For example, in lecture speech, many acoustic events, such as background music, laugh, and applause events are recorded. For lecture speech transcription based on automatic speech recognition (ASR) technique, acoustic event detection (AED) and segmentation should be performed before speech signals are inputted to ASR systems. In multimedia data, acoustic event logs can provide efficient index for multimedia content analysis and retrieval applications [1, 2, 3]. Designing an accurate acoustic event detection (AED) algorithm is important in audio processing and applications.

In traditional modeling for AED, the frame by frame based feature representation is directly mapped to their corresponding categories via a classifier. For example, Mel frequency cepstral coefficient (MFCC) feature is modeled with a Gaussian mixture model (GMM) or support vector machine (SVM) model [4, 5, 6]. The frame based feature only encodes statistics of audio signals in a short time window (e.g., 10 or 20 ms) [3, 4, 5]. In modeling, this may cause model confusion since there exist large overlaps of the statistic distributions of the feature. Considering the discriminative information of acoustic events is encoded in structured temporal-frequency patterns, we have proposed a bag of spectral patch based feature learning algorithm for AED [7]. The bag of spectral patches was learned from a collection of temporal-frequency patches via a clustering algorithm. The learned bag of spectral patches were regarded as acoustic words for feature encoding. In encoding, a sparse representation was obtained based on the acoustic words either based on a sparse coding algorithm or a triangle k-means encoding algorithm [7, 8]. Based on the sparse representation, we trained a support vector machine (SVM) classifier for AED.

In our previous study, in feature encoding stage, each spectral patch was encoded independently based on the learned acoustic words without any consideration of the temporal correlations between spectral patches. Spectral patches extracted consecutively from a time series, their representations should show strong temporal continuity since the neighboring spectral patches have strong time correlation. Based on this consideration, we need to take temporal smooth constraint (i.e., temporal correlation) into consideration in feature encoding. In practical implementations, the temporal correlation can be regarded as a temporal smoothing on the encoded features. However, in sparse encoding, it is possible that a small variation in the spectral patches may lead to a large difference in the encoded feature space [9]. Directly applying a temporal smooth process on feature representation is not suitable. In order to make temporal smoothing not sensitive to small variations, we propose to use temporal max-smoothing on the sparse encoded features. The max-smoothing tries to pick up the maximum response of the encoding in a local time window, and ignore the small values in doing the smooth filtering. By using this process, it is possible to extract time-shift invariant feature which takes time correlation of representations into consideration.

The paper is organized as follows. In section 2, the acoustic dictionary learning from spectral patches, and sparse feature extraction based on the learned acoustic dictionary are first introduced. Then a temporal max-smoothing is applied on the sparse feature representations. In section 3, the model algorithm is briefly introduced. In section 4, AED experiments are carried out to test the proposed algorithm. Conclusion and discussion on future work is given in last section.

2. Feature extraction based on acoustic dictionary learning

The feature extraction processing is showed in Fig. 1. As shown in this figure, three steps are involved. The first step is to automatically learn the acoustic dictionary from a large training data set. The second step is to do sparse feature encoding for a given acoustic signal based on the learned acoustic dictionary. The third step is a temporal max-smoothing on the features as post processing. In the following subsections, these three steps are explained in details.

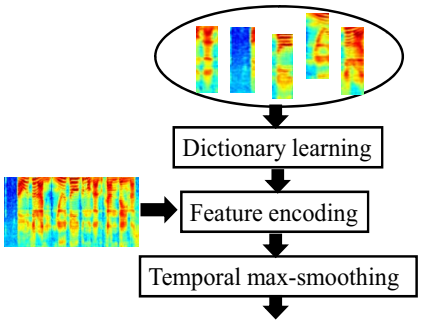

Figure 1: Acoustic dictionary learning and feature extraction.

2.1. Acoustic dictionary learning from spectral patches

The purpose of this step is to learn typical spectral patches (as acoustic words) in an acoustic dictionary. Many dictionary learning algorithms have been proposed in image processing, for example, generalized k-means based vector quantization algorithm [10], sparse coding based dictionary learning [11], matching pursuit based learning (MP) [12]. As we have discussed in introduction, invariant acoustic event pattern information is distributed in temporal-frequency structure, we learn the pattern structure from spectral patches analogy to learning image patch structure. The spectral patches are randomly extracted with a time window from speech power spectrum (Mel frequency scaled spectrum). The extracted patch is reshaped to be a long vector for acoustic dictionary learning. In addition, as shown in image processing [13], pre-processing procedures of the input vectors helped to make the learned dictionary with meaningful structures (e.g., object edges, textures). These procedures include contrast normalization and whitening. Differently from image processing for local brightness and contrast normalization, the purpose of using contrast normalization is to remove the large variations of the dynamic range caused by absolute density among patches. After contrast normalization, a zero phase component analysis (ZCA) is applied for input vector whitening. The whitening is done as follows:

$$\begin{aligned} &\hat{\mathbf{x}} = \mathbf{A}\tilde{\mathbf{x}}_c \\ &\mathbf{A} = \mathbf{V}\left(\mathbf{Z} + \varepsilon\mathbf{I}\right)^{-\frac{1}{2}}\mathbf{V}^T \\ &\mathbf{V}\mathbf{Z}\mathbf{V}^T = \mathrm{cov}\left(\tilde{\mathbf{x}}_c\right), \end{aligned} \tag{1}$$

where $\tilde{\mathbf{x}}_c$ is zero centered vector of the contrast normalized spectral patch vector (in our study, a mean and variance normalization was applied), $\hat{\mathbf{x}}$ is whitened spectral patch vector, $\mathbf{A}$ is a whitening transform matrix, $\mathbf{V}$ and $\mathbf{Z}$ are the eigen vector and eigenvalue matrix of the covariance matrix of $\tilde{\mathbf{x}}_c$, respectively. In Eq. 1, ε is a constant which is used as a low-pass filtering to remove noise effect (on small eigenvalues). The whitened vectors are used in acoustic dictionary learning. For easy and fast computation, the MP based dictionary learning was adopted in this study as:

$$\begin{aligned} &\min_{\mathbf{D},\mathbf{s}} \sum_{i=1}^{N} \left\|\hat{\mathbf{x}}_i - \mathbf{D}\mathbf{s}_i\right\|_2^2 \\ &\text{s.t.} \left\|\mathbf{d}_j\right\| = 1, j = 1, 2, ..., K \\ &\mathrm{Card}\left(\mathbf{s}_i\right) = q, i = 1, 2, ..., N, \end{aligned} \tag{2}$$

where $\mathbf{D}$ is the acoustic dictionary with K elements as $\mathbf{d}_j$ (column vector of $\mathbf{D}$), $j = 1, 2, ..., K$. $\mathbf{s}_i$ is sparse coefficient vector for the i-th sample. $\mathrm{Card}\left(\mathbf{s}_i\right) = q$ denotes only q hot elements as 1 in one coefficient vector. In our experiments, $q = 1$ was used.

2.2. Feature encoding

The learned acoustic dictionary can be used in a bag-of-words model based feature encoding framework which is popularly used in computer vision [14]. In encoding for representation, several encoding strategies have been proposed. For example, histogram based encoding [15], sparse coding [11], and soft triangle k-means encoding [13]. We have applied the later two types of encoding methods in our previous studies [7, 8]. In approximation based encoding, any spectral patch can be regarded as a linear approximation of the learned acoustic dictionary. One of these techniques is the least absolute shrinkage and selection operator (lasso) based sparse coding. It is formulated as:

$$\begin{aligned} &\mathbf{s}^* = \arg\min_{\mathbf{s}} \tfrac{1}{2}\left\|\hat{\mathbf{x}} - \mathbf{D}\mathbf{s}\right\|_2^2 + \lambda\left|\mathbf{s}\right|_1^1 \\ &\left|\mathbf{s}\right|_1^1 = \sum_{i=1}^{K}\left|s_i\right|, \end{aligned} \tag{3}$$

where $\hat{\mathbf{x}}$ is an whitened spectral patch, $\mathbf{D}$ is the learned acoustic dictionary, $\mathbf{s}$ is the sparse coding coefficient (with K acoustic words in the learned dictionary), $|.|$ is an operator to obtain the absolute value of a variable. λ is used to control the tradeoff between the approximation accuracy and sparsity. In Eq. 3, the l_1 sparsity is defined as sum of the absolute values of the vector elements. Another feature encoding algorithm is triangle k-means encoding. It is a "soft" version of the "hard" encoding (vector quantization) used in k-means clustering. The hard encoding is defined as:

$$s_i\left(\hat{\mathbf{x}}\right) = \begin{cases} 1, \text{ for } i = \arg\min_j \left\|\hat{\mathbf{x}} - \mathbf{d}_j\right\|_2^2 \\ 0, \text{ otherwise,} \end{cases} \tag{4}$$

where only one "hot" element gives value in a vector. This "hot" value is the one which shows the highest similarity (defined as minimum Euclidian distance) when comparing the input spectral patch (whitened) with the acoustic words in the learned dictionary. In order to encode much more information, the soft version with explicit sparsity control was proposed [7, 13]. It is formulated as:

$$\begin{aligned} &s_i\left(\hat{\mathbf{x}}\right) = \max\left\{0, \alpha * \mathrm{mean}\left(\mathrm{dist}\left(\hat{\mathbf{x}}, \mathbf{d}\right)\right) - \mathrm{dist}\left(\hat{\mathbf{x}}, \mathbf{d}_i\right)\right\} \\ &\mathrm{dist}\left(\hat{\mathbf{x}}, \mathbf{d}\right) = \left\|\hat{\mathbf{x}} - \mathbf{d}\right\|_2^2, \end{aligned} \tag{5}$$

where $\mathrm{dist}\left(\hat{\mathbf{x}}, \mathbf{d}\right)$ is defined as Euclidian distance between the input spectral patch and acoustic words, mean(.) denotes an average operator, α is sparsity control parameter. As shown in [7, 13], with the triangle k-means encoding (via α controlling the representation sparsity), comparable performance could be obtained compared with many other sparse coding algorithms.

2.3. Temporal max-smoothing

In feature encoding as introduced in section 2.2, either sparse coding or triangle k-means encoding, it can be regarded as a mapping function which maps a spectral patch to a feature vector. This mapping however is not smooth [9]. A small variation in the input spectral patch may lead to a large difference in the encoded vectors. In addition, in feature encoding, each spectral patch is encoded independently. In real applications, spectral patches taken from neighboring time locations should be encoded into representations with strong time correlation. It is not suitable to directly adding temporal smoothing on the encoded feature vectors. In order to extract time-shift invariant feature and remove large variations due to encoding methods, inspired by max-pooling in image processing [16], we propose to use a max-smoothing algorithm as a post processing on the encoded feature vectors. The max-smoothing algorithm tries to assign a

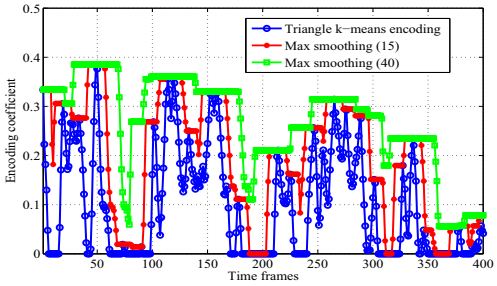

Figure 2: Temporal max-smoothing process.

feature coefficient in each time frame to the maximum response value in a local time window as:

$$s_j(\hat{\mathbf{x}}_t) = \max_{t-T_{win} \le i \le t+T_{win}} \{s_j(\hat{\mathbf{x}}_i)\} \quad j = 1, 2, ..., K, \tag{6}$$

where T_{win} is the time window for extracting several consecutive spectral patches. Fig. 2 shows the effect of the max-smooth processing. In this figure, the triangle k-means encoding method is used in feature extraction, and one dimension feature corresponding to one acoustic word is shown (as vertical axis). The max-smooth process was applied for two time windows of sizes 15 and 40 frames (16 ms for each frame). From this figure we can see that local maximum response was picked up around a local time window as representation.

3. Support vector machine modeling

In this study, a linear support vector machine (SVM) was used for acoustic event modeling. For training data pairs $(\mathbf{s}_i, l_i)$, with $i = 1, 2, ..., N$, where l_i is the label, and $\mathbf{s}_i$ is the i-th feature vector. Multi-class SVMs are built, and each SVM for each acoustic event is constructed as one-against-all with parameter $\mathbf{w}_j$ (the j-th SVM) as [17]:

$$\min_{\mathbf{w}_j} \sum_{i=1}^{N} \left(\max \left\{ 0, 1 - l_i \left(\mathbf{w}_j^T \mathbf{s}_i \right) \right\} \right)^2 + \beta \left\| \mathbf{w}_j \right\|_2^2, \tag{7}$$

where β is a parameter for classifier regularization. The classification can be done by picking up the one which gives the maximum value from all the SVMs as:

$$\hat{l} \triangleq \arg \max_{j \in \{1,2,...,M\}} \mathbf{w}_j^T \mathbf{s}, \tag{8}$$

where M is the total event number.

4. Experiments

We carried out experiments on automatic detection of acoustic events. Acoustic events were selected from audio data of TED (technology, entertainment, and design) talks. From the TED talks, besides speech, we selected other five types of acoustic event data, i.e., applause, laugh, cough, music, and background (noise) events. The acoustic segments of different events were manually labeled and transcribed. The Mel frequency filter band spectrum (40 filter bands) feature was used in acoustic dictionary learning. The spectrum was extracted for each time window with 16 ms frame length and 10 ms frame shift. In acoustic dictionary learning stage, 500,000 spectral patches were randomly chosen from audio spectrum of 50 TED talks. In testing, audio spectrum was selected from another 10 TED talks. The whole system was configured as shown in Fig. 3. In this figure, the first step was feature extraction (with or without temporal smoothing) based on a learned acoustic dictionary as discussed in section 2.1. Then the feature was normalized for SVM event modeling and classification. A hangover scheme is applied to smooth the event detection decisions.

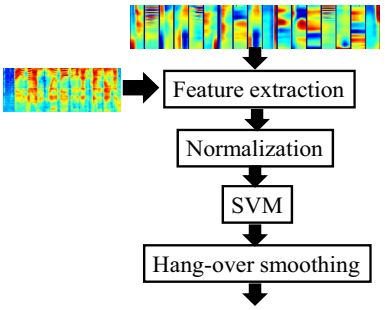

Figure 3: Configuration of the acoustic event detection system.

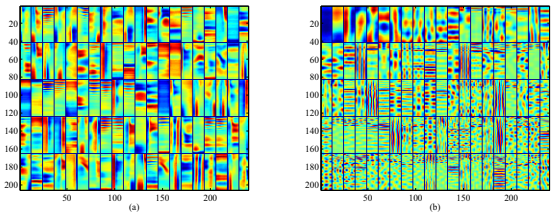

Figure 4: Acoustic dictionary learned from spectral patches with the MP algorithm (a), and with the principal component analysis algorithm (b).

Performance evaluation metrics are related to false alarm rate and hitting rate of event detection. For audio data, these metrics can be frame based, event based or class-wise event based evaluation [1, 3]. In this study, frame based evaluation is used, i.e., frame based recall (Rec), precision (Pre) and F evaluation metrics were used [5]. In the following subsections, several processing stages involved in the system are examined.

4.1. Acoustic dictionary learning

An example of the learned acoustic dictionary is shown in Fig. 4. For comparison, basis vectors based on principal component analysis (PCA) of the whitened spectral patches is also shown in this figure. In this figure, each rectangle box is one acoustic word with x-axis as time (each with 11 frames) and y-axis as frequency (Mel frequency band index, each with 40 bands). From this figure, we can see that the MP learned acoustic words encoded much more meaningful structure of acoustic spectrum than the PCA learned acoustic words, such as harmonics, time-frequency transitions. We believe that these acoustic words are representative time-frequency structures of acoustic events. In addition, the MP algorithm can learn overcomplete dictionary while the PCA can not learn a dictionary with large number of basis vectors beyond the number of feature dimensions.

4.2. Feature encoding for AED

Many factors affect the AED performance, for example, spectral patch size (PS), acoustic codeword size (CS) of the learned dictionary. In order to examine the two encoding methods discussed in section 2.2, first we fix the PS and CS in learned dictionary for AED experiments. The temporal smoothing is not applied in this stage. Feature sparsity is controlled with varying parameter λ in Eq. 3 for lasso sparse coding, and is controlled by changing parameter α for triangle k-means encoding in Eq. 5. The results are shown in tables 1 and 2. From these two

Table 1: Performance of lasso sparse coding (PS=11, CS =128) (%)

λ	0.1	0.5	0.8	0.9	1	1.5
Rec	75.53	82.85	**84.06**	83.98	83.94	82.73
Pre	77.47	84.34	**85.59**	85.51	85.47	84.24
F	76.31	83.51	**84.74**	84.66	84.62	83.40

Table 2: Performance of triangle k-means encoding (PS=11, CS=128) (%)

α	0.9	0.95	1.0	1.05	1.1	1.5
Rec	84.46	**84.68**	84.48	82.26	78.79	73.81
Pre	85.95	**86.92**	86.72	84.00	80.49	75.47
F	85.12	**85.59**	85.39	83.02	79.53	74.53

tables, we can see that feature sparsity should be kept in a certain range to obtain the best performance for these two encoding methods ($\lambda = 0.8$ for lasso sparse coding, and $\alpha = 0.95$ for triangle k-means encoding). In addition, we confirmed that the triangle k-means encoding obtained comparable or even better performance than the lasso sparse coding. Therefore, in the following experiments, the triangle k-means encoding method is used.

4.3. Temporal smoothing on sparse features

In order to enhance temporal correlation in feature representations, we did experiments to examine a temporal smooth processing on the feature vectors. We examine two temporal smoothing algorithms, one is temporal median smoothing, the other is temporal max-smoothing. In the experiments, the triangle k-means encoding with sparsity control parameter $\alpha = 0.95$ was used.

4.3.1. Temporal median smoothing

A median filter was applied on each dimension of the feature coefficient for temporal smoothing. The smoothed feature vectors were used to train the SVM model for AED. The results are shown in table 3. In this table, "FRMs" represents the number of frames of a time window used in median filtering. The experiment condition is with PS =11, and CS =128. From this table, we can see that simply adding temporal smoothing on the sparse feature decreases the performance. This is due to the sparse coding is not a smooth mapping.

4.3.2. Temporal max-smoothing

We did experiments using temporal max-smoothing for feature process. The results are shown in table 4 for experiment with the same condition as in section 4.3.1. From this table, we can see that increasing the time window size for max smoothing ("MaxWin") in a certain range helps to improve the performance. The best time window for max smoothing is around 20 frames for codeword size 128 (16 ms frame length).

In our previous study, we have showed that increasing the spectral patch size in a certain range helped to improve the performance [7]. The improvement was due to learning an acoustic dictionary which could explore long temporal-frequency structure for representation. In this study, the improvement of temporal max-smoothing was caused by utilizing the strong temporal correlation in feature encoding. We further conducted experiments to examine the joint effect of the two factors (spectral patch size and time window in max-smoothing). Two time windows for max-smoothing 5 and 20 fames were tested with variation of PS to 3, 7, 11, 15, 19 frames. Code word size 1024 was used in order to examine their joint effect on large acoustic dictionary based performance. The results are drawn in Fig. 5. In this figure, the three evaluation metrics with different temporal window size for max-smoothing are shown (labeled as Rec_MaxWin#, Pre_MaxWin#, F_MaxWin#). From this figure, we can see that in small spectral PS cases, the improvement from increasing of the max-smoothing time window is large. Increasing patch size can have similar effect as increasing time window in max smoothing. But the effect is different. Putting these two factors into consideration can further improve the performance.

Table 3: Performance of temporal median filtering (PS = 11, CS =128) (%)

FRMs	1	5	10	15	20	25
Rec	**84.68**	84.66	84.04	82.56	81.00	80.65
Pre	**86.92**	86.91	86.24	84.73	83.15	82.79
F	**85.59**	85.58	84.93	83.44	81.87	81.52

Table 4: Performance of temporal max-smoothing (PS = 11, CS =128) (%)

MaxWin	1	5	10	15	20	25
Rec	84.68	85.09	85.20	85.69	**85.97**	85.33
Pre	86.92	87.36	87.47	87.96	**88.25**	87.60
F	85.59	86.02	86.12	86.61	**86.90**	86.25

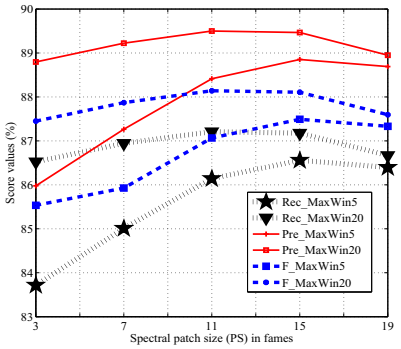

Figure 5: Join effect of PS and temporal max-smoothing.

5. Conclusion and future work

In this paper, we first compared the lasso sparse coding with a triangle k-means encoding algorithm, and confirmed that the triangle k-means encoding could obtain comparable or better results. Considering that strong temporal correlation exists in spectral patches, we further added an temporal max smoothing process to enhance the feature temporal correlation while reducing the variations due to the sparse encoding process. Experiments showed that adding temporal max-smoothing improved the performance, and could further improve the performance on large spectral patch size based encoding.

Many deep feature learning algorithms have been proposed for pattern classification [18, 19, 20]. For example, deep neural network based feature learning. We try to use acoustic dictionary learning concept for extracting features, and examine many factors in AED. From our study, we found many factors which are important for AED. We believe that if these factors are explicitly taken into consideration in deep learning architecture, much more efficient feature learning algorithm can be designed than using a fully black-box like deep neural network without consideration of specific knowledge in AED task.

6. References

[1] Giannoulisy, D., Benetosx, E., Stowelly, D., Rossignolz, M., Lagrangez, M., and Plumbley, M., "Detection and Classification of Acoustic Scenes and Events: an IEEE AASP Challenge," IEEE workshop on applications of signal processing to audio and acoustics (WASPAA), 2013.

[2] Heittola, T., Mesaros, A., Eronen, A., and Virtanen, T., "Context-dependent sound event detection," EURASIP Journal on audio, speech, and music processing, vol. 1, pp. 1-13, 2013.

[3] Zhuang, X., Zhou, X., Hasegawa-johnson, M. A., and Huang, T. S., "Real-world acoustic event detection," Pattern recognition letters, vol. 31, no. 12, pp. 1543-1551, 2010.

[4] Zieger, C., "An HMM based system for acoustic event detection," Multimodel technologies for perception of humans, pp. 338-344, 2008.

[5] Temko, A., Nadeu, C., and Biel, J. I., "Acoustic Event Detection: SVM-Based System and Evaluation Setup in CLEAR'07," Multimodel technologies for perception of humans, pp. 354-363, 2008.

[6] Huang, Z., Cheng, Y., Li, K., Hautamaki, C., and Lee, C., "A Blind Segmentation Approach to Acoustic Event Detection Based on I-Vector," in Proc. Interspeech, pp. 2282–2286, 2013.

[7] Lu, X., Tsao, Y., Matsuda, S., and Hori, C., "Sparse representation based on a bag of spectral exemplars for acoustic event detection," ICASSP, Italy, 2014.

[8] Lu, X., Tsao, Y., Peng, S., and Hori, C., "Spectral Patch Based Sparse Coding for Acoustic Event Detection," ISCSLP, Singapore, 2014.

[9] Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., and Gong, Y., "Locality-constrained linear coding for image classification," in Proc. of CVPR, pp. 3360-3367, IEEE, 2010.

[10] Aharon, M., Elad, M., and Bruckstein, A., "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," IEEE Transactions on Signal Processing, 54 (11): 4311-4322, 2006.

[11] Mairal, J., Bach, F., Ponce, J., and Sapiro, G., "Online Learning for Matrix Factorization and Sparse Coding," Journal of Machine Learning Research, volume 11, pp: 19-60, 2010.

[12] Mallat, S. G., and Zhang, Z., "Matching Pursuits with Time-Frequency Dictionaries," IEEE Transactions on Signal Processing, Vol. 41, No. 12, pp. 3397-3415, 1993.

[13] Coates, A., Lee, H., and Ng, A. Y., "An analysis of single-layer networks in unsupervised feature learning," in Proc. the 14-th International Conference on AI and Statistics, 215-223, 2011.

[14] Csurka, G., Dance, C., Fan, L., Willamowski, J., and Bray, C., "Visual categorization with bags of keypoints," Workshop on statistical learning in computer vision, ECCV, pp.1-22, 2004.

[15] Sivic, J., and Zisserman, A., "Video Google: A text retrieval approach to object matching in videos," vol. 2, pp. 1470-1477, In ICCV, 2003

[16] Boureau, Y., Roux, N, Bach, F., Ponce, J., and LeCun, Y. "Ask the locals: multi-way local pooling for image recognition," In ICCV, 2011.

[17] Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., and Lin, C. J., "LIBLINEAR: A library for large linear classification," Journal of machine learning research, vol. 9, pp. 1871-1874, 2008.

[18] Bengio, Y., "Learning deep architectures for AI," Foundations and Trends in Machine Learning, 2(1): 1-127, 2009.

[19] Hinton, G. E., and Salakhutdinov, R., "Reducing the Dimensionality of Data with Neural Networks," Science, 313: 504-507, 2006.

[20] Ranzato, M. A., Huang, F. J., Boureau, Y. L., and LeCun, Y., "Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition," IEEE conference on Computer Vision and Pattern Recognition, 1-8, 2007.