EFFICIENT MANIFOLD PRESERVING AUDIO SOURCE SEPARATION USING LOCALITY SENSITIVE HASHING

Minje Kim[1], *Paris Smaragdis*[1,2,3], *Gautham J. Mysore*[3]

[1]Department of Computer Science, University of Illinois at Urbana-Champaign
[2]Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign
[3]Adobe Research

minje@illinois.edu, paris@illinois.edu, gmysore@adobe.com

ABSTRACT

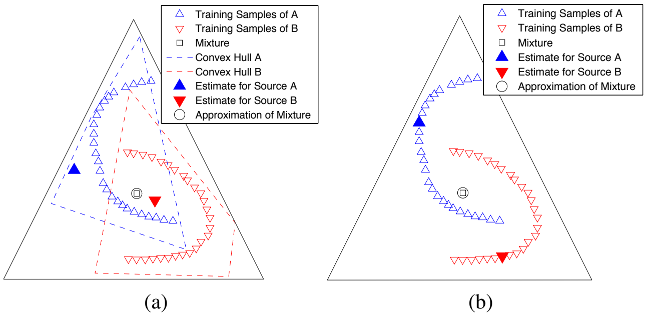
We propose an efficient technique to learn probabilistic hierarchical topic models that are designed to preserve the manifold structure of audio data. The consideration of the data manifold is important, as it has been shown to provide superior performance in certain audio applications such as source separation. However, the high computational cost of a sparse encoding step due to the requirement of a large dictionary prevents it from being used in real-world applications such as real-time speech enhancement and the analysis of big audio data. In order to achieve a substantial speed-up of this step, while still respecting the data manifold, we propose to harmonize a particular type of locality sensitive hashing with the hierarchical topic model. The proposed use of hashing can reduce the computational complexity of the sparse encoding by providing candidates of non-zero activations, where the candidate set is built based on Hamming distance. The hashing step is followed by comprehensive sparse coding that considers those candidates only, rather than the entire dictionary. Experimental results show that the proposed hashing technique can provide audio source separation results comparable to the similar system without hashing, but with significantly less and cheaper computation.

Index Terms— Locality Sensitive Hashing, Winner Take All Hashing, Source Separation, Topic Modeling

1. INTRODUCTION

Probabilistic topic models such as Probabilistic Latent Semantic Indexing (PLSI) [1] and a related non-probabilistic counterpart, Nonnegative Matrix Factorization (NMF) [2], have gained a great deal of popularity for analyzing monaural audio signals, e.g. music transcription [3], music source separation [4, 5, 6], and speech denoising [7, 8]. A common assumption that underlies those approaches is that each magnitude spectrum of the Short-Time Fourier Transform (STFT) of an audio signal is generated from a probability vector. Topic models are to discover a convex combination of topics to approximate the underlying distribution, where each topic represents a sound component, e.g. a note of music or a phoneme of speech, with the help of its additive parts-based representation.

A popular usage of such models is for the process of separating multiple sound sources from a single-channel mixture recording. In that scenario, a convex hull is first learned for each source-specific set of training spectra. Then, an unseen mixture spectrum is decomposed into individual estimates of source spectra such that they meet two criteria about convex combinations. First, each source estimate should be confined to its corresponding convex hull, which models the source. Second, the convex combination of the source estimates should approximate the mixture [9].



(a) (b)

Fig. 1. [11] A toy separation example with (a) an ordinary topic model (b) sparse PLSI. In (a) the source estimates are good solutions in the conventional sense, because they are in their corresponding convex hulls and their convex combination approximates the mixture well. However, they are not good source estimates after all since they are off the manifold of the training data and one of them even falls in the overlap. In (b) the solutions with sparse PLSI are free from those problems.

Manifold learning is particularly important when training signals are monophonic, i.e. the signal has only one pitch or utterance at a time, such as with speech. In this case the ordinary convex hull representation may result in spurious approximations since the hull encompasses all the possible combinations of topics both on and off the data manifold. For instance, speech estimates in the off-manifold area, but inside the hull, might not sound like human speech. Additionally, convex hulls from different sources can overlap with each other and degrade separation results since the hull learned from the training data of a given source could then to some degree explain other sources. This insight was addressed in [10], where a sparse non-parametric model was proposed to handle the issue (we refer to this technique as *sparse PLSI*). In sparse PLSI, each training spectrum is considered as a fixed individual topic, and only a few of them are activated to explain a source spectrum, which finally contributes to form a mixture spectrum. Figure 1 depicts this argument.

A disadvantage of this manifold consideration is the requirement of larger training data sets for robust local reconstruction. Retaining a large number of training samples, instead of the convex hulls de-

This material is based upon work partially supported by the National Science Foundation under Grant No. 1319708.

fined by their simplicial corners only, makes learning computationally heavy and demands a larger amount of memory. These issues were addressed by hierarchical topic models in [11], where a middle-layer variable was presented to divide the model into two parts: local selection (the lower level) and global approximation (the higher level). In the former part, the overcomplete dictionary elements are multiplied with the sparsely encoded middle-level selector variable to produce a set of *hyper topics*, each of which represents a participating source spectrum. The hyper topics are then combined to approximate the mixture input mimicking the audio mixing procedure. Although this model provides a more direct and convenient way to couple manifold learning and topic modeling, its sparse coding step on the selector variable still demands a lot of computation.

In this paper, we propose a technique to speed up the aforementioned manifold preserving source separation by using hashing, which is the first attempt to examine the use of hashing for audio topic models to the best of our knowledge. To this end, we adopt Winner-Take-All (WTA) hashing [12], a particular type of locality sensitive hashing [13], which has shown to be efficient in image search [14]. Similar to the original usage, we hash the dictionary elements to promptly provide a small set of candidates as a match result, so that the subsequent exhaustive search can only focus on this reduced set rather than the entire dictionary. We use Hamming distance on the hash code bits to minimize the burden introduced by this additional procedure of constructing the candidate set. On the other hand, the difference from the original hashing is that we have to associate a hash code from a mixture spectrum with multiple codes from different sources. Our key contribution is the development of a technique to do this. Another important advantage of employing the hashing technique is that its cheap fixed-point operations can extend the applicability of the topic model-based audio analysis techniques to the implementations with more restricted conditions.

2. RELATED WORK

2.1. The Hierarchical Topic Models for Manifold Preserving Source Separation

For an observation vector indexed by $t = \{1, \cdots, T\}$, the probabilities of observing the features indexed by $f \in \{1, \cdots, F\}$ is approximated by a combination of multi-layered topics, $X_{f,t} \sim \sum_{z_y,y} P(f|z_y)P_t(z_y|y)P_t(y)$ [11]. In this model $P(f|z_y)$ is for a categorical distribution over the features given a topic z_y, equivalently to that in standard topic models [1, 15] if we ignore y, the subscript indicating the hyper topics[1]. The second parameter $P_t(z_y|y)$ can be viewed in two ways. First, if we marginalize the middle-layer topics $y \in \{1, \cdots, Y\}$, then the conventional topic distribution is specified, i.e. $\sum_y P_t(z_y|y)P_t(y) = P_t(z)$. On the other hand, $P_t(z_y|y)$ can aggregate original topics into its representatives $P_t(f|y) = \sum_{z_y} P(f|z_y)P_t(z_y|y)$, which we call hyper topics.

In the hierarchical topic model $P_t(z_y|y)$ is regularized to be sparse. If for a given y only a small number of original topics z_y are activated, they tend to form a local convex hull on the data manifold where their combination, the hyper topic $P_t(f|y)$, lies on. One of the ways to impose the sparsity constraint proposed in [11] is to search for the nearest neighbors of the current estimation of hyper topics, and allow only those neighbors to be activated.

A hyper topic $P_t(f|y)$ can have a set of nearest neighbors $\mathcal{N}_y^t$:

$$\mathcal{N}_y^t \leftarrow \Big\{z_y : \mathcal{E}\Big[P(f|z_y)\big\|P_t(f|y)\Big] < \mathcal{E}\Big[P(f|z' \notin \mathcal{N}_y^t)\big\|P_t(f|y)\Big]\Big\}, \quad (1)$$

where cross entropy is used for the divergence measure, i.e. $\mathcal{E}[A\|B] = -\sum A_i \log B_i$. The set should be small enough to reflect the local structure as in Locally Linear Embedding (LLE) [16]. Once the neighbor set is found, the selector elements $P(z_y|y)$ with $z_y \notin \mathcal{N}_y^t$ are filled with zeros. Consequently, the hierarchical EM updates are only on the neighbor set $\mathcal{N}_y^t$:

The reduced E-step:

$$P_t(z_y, y|f) \leftarrow \frac{P(f|z_y)P_t(z_y|y)P_t(y)}{\sum_{z_y,y} P(f|z_y)P_t(z_y|y)P_t(y)} \quad \forall z_y \in \mathcal{N}_y^t \quad (2)$$

The reduced M-step:

$$\begin{aligned} P_t(z_y|y) &\leftarrow \frac{\sum_f X_{f,t}P_t(z_y, y|f)}{\sum_{f,z_y} X_{f,t}P_t(z_y, y|f)} && \forall z_y \in \mathcal{N}_y^t, \\ P_t(z_y|y) &\leftarrow 0 && \forall z_y \notin \mathcal{N}_y^t, \\ P_t(y) &\leftarrow \frac{\sum_{f,z_y} X_{f,t}P_t(z_y, y|f)}{\sum_{f,y,z_y} X_{f,t}P_t(z_y, y|f)} && \forall z_y \in \mathcal{N}_y^t. \quad (3) \end{aligned}$$

In the separation scenario, $X_{f,t}$ stands for the magnitude of Fourier spectra of a mixed signal, while the middle-layer latent variable y indicates sources. We denote the overcomplete dictionary of y-th source with $P(f|z_y)$. Note that the topic index z_y for the dictionary elements is with the source index y to distinguish dictionaries from different sources. We assume that a mixture spectrum is associated with a set of hyper topics, each of which corresponds to an estimated source spectrum. The time index t reflects the fact that we do the separation in a frame by frame manner. Finally, the separation of y-th source from t-th mixture spectrum can be done by multiplying the learned posterior to the input mixture, i.e. $X_{f,t} \sum_{z_y} P_t(z_y, y|f)$.

Computational complexity: if we ignore the nearest neighbor searching step (1) and assume the number of training specta Z_y is same for all the sources, the complexity of separating a frame in (2) and (3) is $\mathcal{O}(FZ_yY)$. With the sparse representation of matrices for $P_t(z_y|y)$, the complexity of the separation further reduces to $\mathcal{O}(FKY)$, where K is the number of neighbors smaller than Z_y. However, the nearest neighbor search cannot be ignored since its complexity $\mathcal{O}(FZ_yY)$ on the entire training samples Z_y is higher than $\mathcal{O}(FKY)$.

2.2. Winner-Take-All Hashing

The recent application of WTA hashing [12] to a big image searching task provided accurate and fast detection results [14]. As a kind of locality sensitive hashing [13], it has several unique properties: (a) similar data points tend to collide more (b) Hamming distance of hash codes approximately reflects the original distance of data. Therefore, it can be seen as a distribution on a family of hash functions $\mathcal{F}$ that takes a collection of objects, such that for two objects x and y, $\mathrm{Pr}_{h\in\mathcal{F}}[h(x) = h(y)] = \mathrm{sim}(x, y)$. $\mathrm{sim}(x, y)$ is some similarity function defined on the collection of objects [17].

WTA hashing encodes relative ordering of the elements in an input vector. Although the rank order metric can be a stable discriminative feature, it non-linearly maps data to an intractably high dimensional space. For example, the number of orders in M-combinations out of an F-dimensional vector is (# combinations) × (# orders in each combination) $= \frac{F!}{(F-M)!}$. Instead, WTA hashing produces hash codes that compactly approximate the relationships.

[1] We use the more basic PLSI model rather than Latent Dirichlet Allocation (LDA) for a clearer explanation of the proposed manifold learning ideas.

WTA hashing first defines a permutation table $\mathcal{P} \in \mathbb{R}^{L\times M}$ that has L different sets of random indices, each of which chooses among M elements. For the l-th set the position of the maximal element among M elements is encoded instead of the full ordering. Therefore, the length of hash codes is ML-bits since each permutation results in M bits, where only one bit is on to indicate the position of the maximum, e.g. $3 = 0100$ if $M = 4$, and there are L such permutations. Whenever we do this encoding for an additional permutation, at most $M-1$ new pairwise orders (maximum versus the others) are embedded in the hash code. The permutation table is fixed and shared so that the hashing results are consistent.

For example, if the first row of $\mathcal{P}^{L\times 2}$ is $[4, 2]$, then the first two bits of the hash code for a vector x = [8.8,9.9,3.3,3.4] will be 10 since the second index 2 indicates the largest element (3.4 versus 9.9). Another row in $\mathcal{P}$ will add another two bits, and so on. Note that WTA hashing results in hash codes that respect shape similarities rather than simple Euclidean distance between vectors. Also, WTA hashing is known to be robust to the additive noise.

Even though WTA hashing provides stable hash codes that can potentially replace the original features, its approximated rank orders cannot fully reflect the original error function, e.g. cross entropy in topic models. Therefore, in the proposed method, we use this hash codes only to reduce the size of the solution space. Then, the final separation is achieved from the original EM algorithm with comprehensive nearest neighbor searching in (1) and (3).

3. WINNER-TAKE-ALL HASHING FOR MANIFOLD PRESERVING SOURCE SEPARATION

As in Section 2.1, we can respect the manifold of the data during topic modeling by allowing only a small number of local neighbors to participate in the reconstruction of the hyper topics. Sparsity on the selection parameter $P_t(z_y|y)$ is critical for this procedure, but due to the possibly large number of training samples, Z_y, it primarily accounts for the computational complexity of the algorithm. One naïve approach to using hashing in order to reduce this complexity is to replace the active set of topics, i.e. the nearest neighbors $\mathcal{N}_y^t$, with the ones with lowest Hamming distance to the hyper topics. However, the mismatch between the approximated rank ordering measure and the original cross entropy can cause inaccurate results.

Instead of solely relying on Hamming distance as our distance metric, we use WTA hashing as a pre-processing step. After the hashing part reduces the search space from Z_y to N topics, we perform a K nearest neighbor search on these reduced N topics to refine the results. The key idea is to keep an up to date set of candidates $\mathcal{Z}^{N\times Y}$, whose y-th column vector holds the indices of N closest candidates to the y-th hyper topic in terms of the Hamming distance. If we set $K < N \ll Z_y$, the final estimation of $P_t(z_y|y)$ can focus only on N elements rather than the entire Z_y topics. Unless N is too small to include the K important topics as candidates, or the Hamming distance defined on the WTA hash codes is significantly different from the original distance measure, some spurious candidates included in the candidate set should not be of significant consequence. In other words, the exhaustive nearest neighbor search is able to pick out the final nearest neighbors anyway with or without hashing, but a proper hashing results can speed up this by providing good candidate solutions.

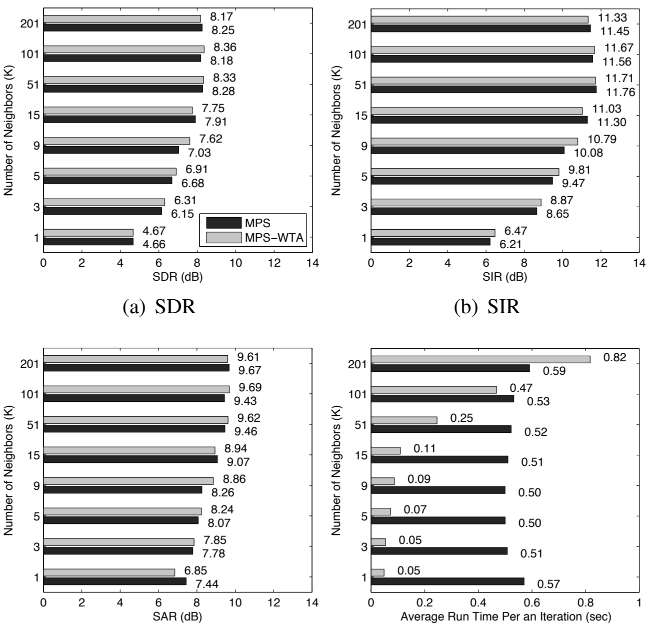
Algorithm 1 describes the separation procedure assisted by WTA hashing. We use notation $A_{:,i}$ to indicate i-th column of a matrix A. In Algorithm 1 each source has its own set of training samples that are indexed by z_y, and hashed in advance (line 4 to 6). Then, we separate each t-th mixture frame independently. In order to conduct simultaneous hash code matching on multiple source dictionaries with only an unseen mixture spectrum, at every iteration we do a tentative separation first, and then do hashing with the source estimates. First, current source estimates $P_t(f|y)$ (randomly initialized vectors at the first iteration) are hashed for a later comparison with their corresponding dictionaries (line 11). Once again, Hamming distance is used to take advantage of the lower complexity of bit-pattern comparisons (line 12). The learned N candidates per each source are used for the subsequent nearest neighbor search, which was originally on the entire samples $Z_y \gg N$ (line 13). The actual EM updates are not affected by this procedure, since they are already defined by excluding non-neighbors (line 14 to 16). The proposed harmonization of hashing and the topic model relies on the reconstruction of the normalized source spectra $P_t(f|y)$ at every EM iteration (line 15). It could be a tentative solution to the separation problem before the convergence, but at the same time it serves as a query at the next iteration to update the neighbor sets: the candidate set $\mathcal{Z}$ and the nearest neighbors $\mathcal{N}_y^t$.

Algorithm 1 Manifold preserving source separation with WTA hashing

```
 1: Initialize a permutation table $\mathcal{P} \in \mathbb{R}^{L,M}$.
 2: Initialize $P(f|z_y)$ with source-specific training spectra.
 3: Initialize $N$ and $K$ to hold the inequalities, $K < N < Z_y$.
 4: for $y \leftarrow 1$ to $Y$ and $z_y \leftarrow 1$ to $Z_y$ do
 5:     $C_{:,z_y,y} \leftarrow$ WTA_hash( $P(f|z_y)_{:,z_y}, \mathcal{P}$)
 6: end for
 7: for $t \leftarrow 1$ to $T$ do
 8:     Initialize $P_t(f|y)$, $P_t(z_y|y)$ and $P_t(y)$ with random numbers, and normalize to sum to one.
 9:     repeat
10:         for $y \leftarrow 1$ to $Y$ do
11:             $c \leftarrow$ WTA_hash( $P_t(f|y)_{:,y}, \mathcal{P}$)
12:             Find a set of $N$ candidate topics for $y$-th source, $z_y \in \mathcal{Z}_{:,y}$, with least Hamming distance to $c$: $Hamming(c, C_{:,z_y,y})$.
13:             $\mathcal{N}_y^t \leftarrow \Big\{z_y \Big| z_y \in \mathcal{Z}_{:,y},\ \mathcal{E}\Big[P(f|z_y)\big\|P_t(f|y)\Big] < \mathcal{E}\Big[P(f|z'_y \notin \mathcal{N}_y^t)\big\|P_t(f|y)\Big]\Big\}$
14:             for all $f \in \{1 \cdots F\}, z_y \in \mathcal{N}_y^t$ do
15:                 $P_t(z_y, y|f) \leftarrow \frac{P(f|z_y)P_t(z_y|y)P_t(y)}{\sum_{z_y,y} P(f|z_y)P_t(z_y|y)P_t(y)}$,
                    $P_t(z_y|y) \leftarrow \frac{\sum_f X_{f,t}P_t(z_y,y|f)}{\sum_{f,z_y} X_{f,t}P_t(z_y,y|f)}$,
                    $P_t(y) \leftarrow \frac{\sum_{f,z_y} X_{f,t}P_t(z_y,y|f)}{\sum_{f,y,z_y} X_{f,t}P_t(z_y,y|f)}$,
                    $P_t(f|y) \leftarrow \sum_{z_y \in \mathcal{N}_y^t} P(f|z_y)P_t(z_y|y)$.
16:             end for
17:         end for
18:     until Convergence
19: end for
```

3.1. Computational complexity

First of all, we can ignore the complexity of the hash code generation procedure $\mathcal{O}(LMZ_yY)$ as it can be done in advance (line 4 to 6). Since EM updates (line 15) are still on the K final nearest



(a) SDR

(b) SIR

(c) SAR

(d) The average run time

Fig. 2. The average cross talk cancellation results of ten random pairs of speakers by using the comprehensive MPS and its hashing version, MPS-WTA, in terms of (a) SDR (b) SDR (c) SDR. (d) Average run time of individual iterations. We implemented the algorithms with MATLAB® and ran them in a desktop with 3.4 GHz Intel® Core™ i7 CPU and 16GB memory.

neighbors, their complexity $\mathcal{O}(FKY)$ remains same. The actual speed-up happens in line 13 where we do the cross entropy based nearest neighbor search, but now on a much reduced set with only N candidates. Therefore, line 13 runs in the order of $\mathcal{O}(FNY)$, which reduces original complexity $\mathcal{O}(FZ_yY)$ if $N < Z_y$.

Hashing introduces additional complexity, but it is still less complex than $\mathcal{O}(FNY)$. WTA hashing for Y hyper topics (line 11), the calculation of Hamming distance between the hyper topics and the training data, and construction of N candidates (line 12), run in the order of $\mathcal{O}(YLM)$, $\mathcal{O}(YLZ_y)$, and $\mathcal{O}(Z_yN)$, respectively. However, because usually we set $L < F$ and $M < N$, its complexity $\mathcal{O}(YLM)$ is lower than $\mathcal{O}(FNY)$. The Hamming distance calculation with $\mathcal{O}(YLZ_y)$ can be also disregarded thanks to the cheap bit-operations. Therefore, the complexity of each iteration for the source y is governed by $\mathcal{O}(Z_yN)$ or $\mathcal{O}(FNY)$ depending on the inequality between Z_y and FN, while neither of them is more complex than the original $\mathcal{O}(FZ_yY)$ since usually $N < Z_y$ and $N < FY$.

4. NUMERICAL EXPERIMENTS

In this section we compare the hierarchical models with or without the use of hashing: the comprehensive Manifold Preserving Separation without hashing (MPS) and the proposed Manifold Preserving Separation with WTA hashing (MPS-WTA). The comparison validates that the proposed harmonization with hashing does not significantly reduce the cross-talk cancellation (separation of a target and interference speaker) performance although it spends less time. The separation quality is measured with the common source separation metrics: Signal-to-Interference Ratio (SIR), Signal-to-Artifact Ratio (SAR), and Signal-to-Distortion Ratio (SDR), that measure the degree of separation, algorithmic artifacts, and the overall quality, respectively [18]. Throughout the experiments, the size of the candidate set N is set to be proportional to that of $\mathcal{N}_y^t$, i.e. $N = 5K$. A permutation table $\mathcal{P}$ is defined with $L = 100$ and $M = 4$, and shared among all hash executions.

For the cross-talk cancellation experiment, we first concatenate nine random sentences per a TIMIT speaker as our training data. Each training set is then transformed into a matrix using STFT with 64 ms *Hann* windowing and 32 ms overlap. We take the magnitudes of the matrix and normalize them to make sure the column vectors sum to one. A sentence per each speaker is set aside for testing. We randomly select a pair of male and female speaker for an experiment and mix their test sentences. We repeat this for ten different pairs. Depending on the random choices of speakers and the sentences, the number of the column vectors (spectra) in the training matrices varies from around 700 to 1,000, while the number of frequency bins is fixed to 513. We run both algorithms for 200 iterations, where we observe convergence. We tried different numbers of neighbors $K = \{1, 3, 5, 9, 15, 51, 101, 201\}$ to control the model complexity.

Figure 2 shows the separation results. We can see that the two methods share similar degrees of separation performance. In the first three sub-figures we can see that their separation performances in terms of (a) SDR, (b) SIR, and (c) SAR, are not significantly different between the two methods. Therefore, we can conclude that MPS-WTA gives comparable crosstalk cancellation performance to MPS. It is a favorable observation for the proposed method, as it can perform the task with reduced computation as we analyzed in section 3.1, and in less average run time per an iteration as shown in Figure 2 (d). From the figures, we observe that with the proposed method, we achieve a significant speed-up by reducing the number of neighbors with a modest decrease in separation performance. For instance, by reducing the number of neighbors from 51 (above which there seems to be no gain in performance) to 5 we can triple the speed, but the separation performance decreases only by about 1.5dB. When N is set to be larger than the number of the training samples, hashing does not provide with the compact candidate set anymore, but merely increases the computation with its redundant searching (see the topmost bar in (d)). Note that the average run time cannot be a valid measure by itself since it can depend on the implementation. Instead, we believe that the computational complexity analysis in Section 3.1 justifies the speed-up theoretically.

5. CONCLUSION

In this paper we proposed to use hashing to facilitate efficient learning of a manifold from audio spectra during their decomposition. The decomposition model uses a middle-layer latent variable to learn the computationally expensive sparse encoding of overcomplete dictionaries, because the sparsity allows the only local neighbors to contribute to the solution similarly to the manifold learning techniques. The proposed hashing technique reduced the complexity by providing a set of candidates that can include the final sparse activations. In this way the proposed hashing-based topic model could achieve the separation of audio mixtures with no performance drops, but with a sensible speed-up. The proposed method was particularly useful in learning a compact representation or in quickly picking out the only relevant entries from a relatively large audio data set. Experiments on a cross-talk cancellation task showed the merit of the proposed method, showcasing both increased processing speed and comparable accuracy. We believe that the proposed method can be promising in devices with limited resources, and in analyzing big audio data.

6. REFERENCES

[1] T. Hofmann, "Probablistic latent semantic analysis," in *Proceedings of the International Conference on Uncertainty in Artificial Intelligence (UAI)*, 1999.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems (NIPS)*. 2001, vol. 13, MIT Press.

[3] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, 2003, pp. 177–180.

[4] T. O. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[5] M. Kim, J. Yoo, K. Kang, and S. Choi, "Nonnegative matrix partial co-factorization for spectral and temporal drum source separation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1192–1204, 2011.

[6] N. J. Bryan and G. J. Mysore, "An efficient posterior regularized latent variable model for interactive sound source separation," in *Proceedings of the International Conference on Machine Learning (ICML)*, Atlanta, Georgia, 2013.

[7] D. L. Sun and G. J. Mysore, "Universal speech models for speaker independent single channel source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, 2013.

[8] Z. Duan, G. J. Mysore, and P. Smaragdis, "Online plca for real-time semi-supervised source separation," in *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, 2012, pp. 34–41.

[9] B. Raj and P. Smaragdis, "Latent variable decomposition of spectrograms for single channel speaker separation," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005, pp. 17–20.

[10] P. Smaragdis, M. Shashanka, and B. Raj, "A sparse nonparametric approach for single channel separation of known sounds," in *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, BC, Canada, 2009.

[11] M. Kim and P. Smaragdis, "Manifold preserving hierarchical topic models for quantization and approximation," in *Proceedings of the International Conference on Machine Learning (ICML)*, Atlanta, Georgia, 2013.

[12] J. Yagnik, D. Strelow, D. A. Ross, and R. Lin, "The power of comparative reasoning," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011, pp. 2431–2438.

[13] P. Indyk and R. Motwani, "Approximate nearest neighbor – towards removing the curse of dimensionality," in *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*, 1998, pp. 604–613.

[14] T. Dean, M. A. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, and J. Yagnik, "Fast, accurate detection of 100,000 object classes on a single machine," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[15] D. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[16] S. T. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[17] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings of the Annual ACM Symposium on Theory of Computing (STOC)*, NY, USA, 2002.

[18] E. Vincent, C. Fevotte, and R. Gribonval, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.