TYING ROTATIONS OF COVARIANCE MATRICES VIA RIEMANNIAN SUBSPACE CLUSTERING

Yusuke Shinohara

Corporate Research and Development Center, Toshiba Corporation
1, Komukai-Toshiba-cho, Saiwai-ku, Kawasaki, 212-8582, Japan
yusuke.shinohara@toshiba.co.jp

ABSTRACT

The use of full covariance matrices in acoustic modeling is getting popular, but its huge computational burden in likelihood calculation is a major issue. Semi-tied covariance matrices are commonly used to speed-up the computation, where global or phone-based tying of transforms, or "rotations", is usually used. However, such tyings are heuristic, and not necessarily optimal. In this paper, we propose a Riemannian-geometric approach to optimally tying rotations of covariance matrices. We first introduce a tangent space of the Riemannian manifold of covariance matrices, which has an excellent distance for measuring dissimilarity between covariance matrices. We then show that covariance matrices having the same rotation to each other lie on the same subspace in the tangent space. Exploiting this property, we fit subspaces to samples (covariances) in the tangent space for finding out clusters of samples that have similar rotations, and tie them together. By doing so, an optimal tying that minimizes the sum of "distortions" of covariance matrices can be found. Experimental results on the Wall Street Journal corpus show a superior performance of the proposed tying over the conventional ones.

Index Terms— speech recognition, acoustic modeling, full covariance, rotation tying, Riemannian manifolds

1. INTRODUCTION

It is well known that acoustic models using full covariances perform better than those using diagonal covariances, given that the same number of Gaussians are used [1, 2]. However, two major issues have prevented them from being deployed in practical applications. One is the data sparsity problem; when the number of frames assigned to a mixture component is not enough, estimation of the full covariance gets unstable. The other is the huge computational cost in likelihood calculation. The former is less of a problem now than used to be as the training data is increasing rapidly, but the latter is still a big problem.

The semi-tied covariance (STC) [3] is the most common approach to mitigating this issue, and is used in many of the state-of-the-art LVCSR systems [4, 5, 6]. Here, covariance matrices are grouped into classes, and their rotations are tied within each class. The STC can be implemented as a set of tied feature transforms, which enables a fast likelihood calculation. To do so, components are usually grouped according to the phoneme that it belongs to, or globally tied. However, there is no theoretical reason to believe that such a grouping is optimal.

In this paper, we propose a Riemannian-geometric method for grouping and tying rotations of covariance matrices. The space of covariance matrices is considered as a Riemannian manifold equipped with the Fisher information metric, and the notion of distance is defined on the manifold to measure dissimilarity between covariance matrices. Because algorithms usually get complicated in Riemannian manifolds (which are curved), its tangent space (which is flat) is used as a surrogate in this work. We show that in a tangent space of the manifold, covariance matrices having the same rotation to each other lie on the same subspace. This property is exploited in clustering rotations. Specifically, a set of subspaces is fit to samples (covariances of a given acoustic model) in the tangent space for finding out clusters of samples that have similar rotations. The class of each component is then determined according to the result of this clustering, and rotations are tied within each class.

By determining the grouping of components in a theoretically grounded manner, the resultant semi-tied covariance gets closer to the original full covariance, which leads to an improved speech recognition performance.

2. RELATED WORK

There have been attempts to approximate full covariance matrices with less complex models for speeding-up likelihood calculation (and/or mitigating the data sparsity issue). These include the semi-tied covariance (STC) [3], also known as maximum-likelihood linear transform (MLLT) [7], extended maximum-likelihood linear transform (EMLLT) [8], and subspace for precision and mean (SPAM) [9]. Among them, the most commonly used one is probably the STC, which is used in many systems as described earlier. However, usually global or phone-based tying is used, which is the issue we would like to address in this work. In [10], one transform was used for each tied-state, but the large number of transforms should lead to a large computational cost, which could be an issue in practical applications. In [11, 12], Euclidean distance based clustering was used, but there is no strong reason to believe that those components that are close in the acoustic space have similar rotations. The most closely related method to ours is the maximum-likelihood tying of semi-tied transforms mentioned in [3]. However, the method was not evaluated in the paper, so a direct comparison with it is difficult. An attempt to approximately rebuild a full covariance system with a simpler system was discussed in [13]. This is somewhat similar to our approach, but they reconstructed it with a diagonal covariance system, while we do so with a semi-tied covariance system.

3. ROTATION TYING

A covariance matrix can be eigen-decomposed into a rotation matrix and a diagonal matrix as

$$\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^{\top}, \tag{1}$$

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

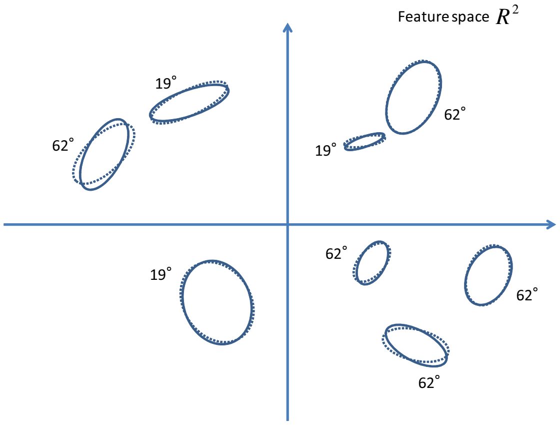

Fig. 1. *An example of rotation tying of covariance matrices, where the feature vector is two-dimensional ($n = 2$) and the number of classes is two ($K = 2$). The dotted ellipses represent Gaussians with original covariance matrices, while the solid ellipses represent Gaussians with rotation-tied (or semi-tied) covariance matrices.*

where $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times n}$ denotes the covariance matrix, $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ is the rotation matrix[1] (whose columns are the eigenvectors of $\boldsymbol{\Sigma}$), $\boldsymbol{D} \in \mathbb{R}^{n\times n}$ is the diagonal matrix (whose diagonals are the eigenvalues of $\boldsymbol{\Sigma}$), and n is the dimensionality of the feature vector. Rotation tying is a scheme where rotation matrices are shared among Gaussians (Figure 1). By using such covariance matrices, likelihood can be evaluated efficiently. Specifically, the log-likelihood of observation $\boldsymbol{o}$ for mixture component m is calculated as[2]

$$\log\mathcal{N}(\boldsymbol{o};\boldsymbol{\mu}_m,\boldsymbol{U}_k\boldsymbol{D}_m\boldsymbol{U}_k^\top) = \log\mathcal{N}(\boldsymbol{U}_k^{-1}\boldsymbol{o};\boldsymbol{U}_k^{-1}\boldsymbol{\mu}_m,\boldsymbol{D}_m), \tag{2}$$

where $\boldsymbol{\mu}_m$ and $\boldsymbol{D}_m$ are the mean and the diagonal matrix of m, and $\boldsymbol{U}_k$ is the rotation matrix of class k (to which m belongs). By storing $\boldsymbol{U}_k^{-1}\boldsymbol{\mu}_m$ in the acoustic model, the increase in computational cost compared with the diagonal covariance case is only K matrix-vector multiplications in each frame, where K is the number of rotation classes.

4. RIEMANNIAN FRAMEWORK

In this section, we briefly review the Riemannian framework of covariance computation [14, 15], which forms the foundation of our algorithm proposed in the next section.

4.1. Riemannian manifold of covariance matrices

Let $\mathcal{M}$ be the manifold of n-by-n covariance matrices, $\boldsymbol{\Sigma} = (s_{ij})$, and $\boldsymbol{\theta} = (s_{11},\ldots,s_{nn})^\top \in \mathbb{R}^{n(n+1)/2}$ be its coordinate system. The Fisher information metric is used as the Riemannian metric of the manifold as

$$g_{ij}(\boldsymbol{\theta}) = \int \frac{\partial\log p(\boldsymbol{x};\boldsymbol{\theta})}{\partial\theta_i}\frac{\partial\log p(\boldsymbol{x};\boldsymbol{\theta})}{\partial\theta_j}p(\boldsymbol{x};\boldsymbol{\theta})d\boldsymbol{x}, \tag{3}$$

where $p(\boldsymbol{x};\boldsymbol{\theta})$ is a parametric distribution of $\boldsymbol{x}$ with parameter $\boldsymbol{\theta}$. Specifically, a Gaussian distribution with a constant mean vector (zero vector), $p(\boldsymbol{x};\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{x};\boldsymbol{0},\boldsymbol{\Sigma}(\boldsymbol{\theta}))$, is used, where $\boldsymbol{\Sigma}(\boldsymbol{\theta})$ is the covariance matrix defined by parameter $\boldsymbol{\theta}$. Let $\gamma : [a,b] \to \mathcal{M}$ be a curve on $\mathcal{M}$, and its length is calculated as

$$\mathcal{L}(\gamma) = \int_a^b \parallel \dot{\gamma}(t) \parallel dt = \int_a^b \sqrt{\sum_{i,j} g_{ij}\frac{d\theta_i(t)}{dt}\frac{d\theta_j(t)}{dt}}dt. \tag{4}$$

The distance between two points on $\mathcal{M}$ is then defined as the length of the geodesic (the shortest curve connecting the two points), which can be calculated as

$$d(\boldsymbol{\Sigma}_1,\boldsymbol{\Sigma}_2) = \sqrt{\sum_j \log^2(\eta_j)}, \tag{5}$$

where η_j is the j-th eigenvalue of $\boldsymbol{\Sigma}_1^{-\frac{1}{2}}\boldsymbol{\Sigma}_2\boldsymbol{\Sigma}_1^{-\frac{1}{2}}$. The distance has been shown to be effective in measuring dissimilarity between covariance matrices in various tasks. See references [14, 15, 16, 17, 18] for further details.

4.2. Tangent space

The Riemannian manifold of covariance matrices defined in the previous subsection is *curved*, and computation on it is not straightforward. An alternative approach is to work on its tangent space, which is *flat* and computation is much easier. In particular, the tangent space at identity is used in this work. The tangent space allows simple and effective computation of covariance matrices, and has been successfully used, for instance, in DT-MRI (diffusion-tensor magnetic resonance imaging) [19, 20].

Mapping a point on the manifold, $\boldsymbol{\Sigma} \in \mathcal{M}$, to the tangent space is realized by the logarithm map as

$$\log(\boldsymbol{\Sigma}) = \sum_{k=1}^{\infty}\frac{(-1)^{k-1}}{k}(\boldsymbol{\Sigma}-\boldsymbol{I})^k = \boldsymbol{U}\log(\boldsymbol{D})\boldsymbol{U}^\top, \tag{6}$$

where we assumed that the covariance matrix is eigen-decomposed as $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$, and $\log(\boldsymbol{D})$ is the diagonal matrix consisting of the logarithms of the eigenvalues. For computational efficiency, we use a vectorized form of the log-covariance,

$$\boldsymbol{\xi} = \text{vec}\left(\log(\boldsymbol{\Sigma})\right) \in \mathbb{R}^{n(n+1)/2}, \tag{7}$$

where the vectorization operator is defined for matrix argument $\boldsymbol{X} = (x_{ij}) \in \mathbb{R}^{n\times n}$ as

$$\text{vec}(\boldsymbol{X}) = \left(x_{11},\ldots,x_{nn},\sqrt{2}\,x_{12},\ldots,\sqrt{2}\,x_{n-1,n}\right)^\top. \tag{8}$$

The diagonal elements are concatenated as they are, while the off-diagonal elements are multiplied with $\sqrt{2}$ to compact the duplicated elements ($x_{ij} = x_{ji}$). It is referred to as a *log-covariance vector* hereafter.

The tangent space is a Euclidean space, so standard Euclidean operations can be used. For instance, calculating a distance on the Riemannian manifold is computationally costly, as can be seen from Eq. (5), but in the tangent space it is simply calculated by the Euclidean distance between log-covariance vectors[3]. Owing to the Euclidean nature, the tangent space allows us to design simple and effective algorithms of covariance computation.

[1] Strictly speaking, it is an orthogonal matrix; although it is a slight abuse of language, we call it a rotation matrix in this paper.

[2] $\log\det\boldsymbol{U}_k$ is omitted because it is always zero for any rotation matrix.

[3] In a special case, when the covariance is diagonal, it is trivial to see that the distance in (5) and the Euclidean distance become exactly the same. A similar discussion was made in [16].

5. RIEMANNIAN SUBSPACE CLUSTERING

We propose a method for tying rotations of covariance matrices so that the sum of "distortions" (distances to the original covariance matrices) is minimized. A special property of the tangent space is exploited in our rotation tying algorithm. In this section, we first show the property, then describe our algorithm.

5.1. Property of the tangent space

The tangent space mentioned above has a special property. Namely, covariance matrices having the same rotation to each other lie on the same subspace in the tangent space. This property can be derived as follows. Assuming that covariance matrix $\boldsymbol{\Sigma}$ is eigendecomposed as $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^\top$ as before, the corresponding log-covariance matrix can be written as

$$\log(\boldsymbol{\Sigma}) = \boldsymbol{U}\log(\boldsymbol{D})\boldsymbol{U}^\top = \sum_{j=1}^{n}\log(\lambda_j)\,\boldsymbol{u}_j\boldsymbol{u}_j^\top, \qquad (9)$$

where λ_j is the j-th eigenvalue of $\boldsymbol{\Sigma}$, and $\boldsymbol{u}_j$ is the j-th eigenvector of $\boldsymbol{\Sigma}$. Let $\boldsymbol{a}_j = \mathrm{vec}(\boldsymbol{u}_j\boldsymbol{u}_j^\top)$ and we have

$$\boldsymbol{\xi} = \mathrm{vec}(\log(\boldsymbol{\Sigma})) = \sum_{j=1}^{n}\log(\lambda_j)\boldsymbol{a}_j. \qquad (10)$$

This equation shows that for any covariance matrix having the same rotation matrix $\boldsymbol{U}$, the corresponding log-covariance vector lies on the subspace spanned by $\{\boldsymbol{a}_j\}_{j=1}^n$ derived from the columns of $\boldsymbol{U}$. Hereafter, the subspace is simply referred to as "the subspace defined by rotation matrix $\boldsymbol{U}$".

Note that the subspace is invariant to permutation and negation of column vectors of $\boldsymbol{U}$. For instance, $\boldsymbol{U}_1 = [\boldsymbol{v}|\boldsymbol{w}|\ldots]$ and $\boldsymbol{U}_2 = [\boldsymbol{w}|\boldsymbol{v}|\ldots]$ represents the same subspace, and $\boldsymbol{U}_3 = [\boldsymbol{v}|\ldots]$ and $\boldsymbol{U}_4 = [-\boldsymbol{v}|\ldots]$ have the same subspace. If one tries to cluster rotation matrices directly in a naive way, permutation and negation will cause serious problems, because standard distance functions, e.g. the one with the Frobenius norm, $d_{\mathrm{F}}(\boldsymbol{U}_1, \boldsymbol{U}_2) = \parallel \boldsymbol{U}_1 - \boldsymbol{U}_2 \parallel_F$, do not return small values for $d(\boldsymbol{U}_1, \boldsymbol{U}_2)$ and $d(\boldsymbol{U}_3, \boldsymbol{U}_4)$, so it is difficult to put $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ (or $\boldsymbol{U}_3$ and $\boldsymbol{U}_4$) into the same cluster. In contrast, our method does not suffer from such issues.

5.2. Clustering

Our rotation tying technique, which we coin *Riemannian subspace clustering (RSC)*, is developed by exploiting the property of the tangent space. The basic idea is to find similar rotations by fitting subspaces to the given samples (Figure 2).

Algorithm 1 summarizes the procedure of Riemannian subspace clustering. First, a set of covariance matrices, $\{\boldsymbol{\Sigma}_i\}_{i=1}^N$, where N is the number of samples, is given. Each of them is then converted to the log-covariance vector as $\boldsymbol{\xi}_i = \mathrm{vec}\,(\log(\boldsymbol{\Sigma}_i))$. Then the set of K subspaces that best fits the log-covariance vectors is found (K is the number of rotation classes). To do this, subspace clustering [21], in particular K-planes clustering, is modified and used. Specifically, the assignment step, where each sample is assigned to the nearest subspace, and the update step, where each subspace is updated to fit the assigned samples, are iterated until convergence. Finally, rotations of those covariance matrices assigned to the same subspace are tied together.

In the assignment step, the distance from sample $\boldsymbol{\xi}$ to the subspace defined by $\boldsymbol{U}$ is calculated as $\parallel \boldsymbol{\xi} - \boldsymbol{P}(\boldsymbol{U})\,\boldsymbol{\xi} \parallel$, where $\boldsymbol{P}(\boldsymbol{U})$ is the projection matrix to this subspace. The projection matrix is defined as $\boldsymbol{P}(\boldsymbol{U}) = \boldsymbol{A}(\boldsymbol{A}^\top\boldsymbol{A})^{-1}\boldsymbol{A}^\top$, where $\boldsymbol{A} = [\boldsymbol{a}_1|\ldots|\boldsymbol{a}_n]$, and $\{\boldsymbol{a}_j\}_{j=1}^n$ are the basis vectors of the subspace. Here, from the definition, $\boldsymbol{a}_j = \mathrm{vec}(\boldsymbol{u}_j\boldsymbol{u}_j^\top)$, we can see that $\{\boldsymbol{a}_j\}_{j=1}^N$ are orthonormal basis vectors, so the projection matrix can be simplified to $\boldsymbol{P}(\boldsymbol{U}) = \boldsymbol{A}\boldsymbol{A}^\top$.

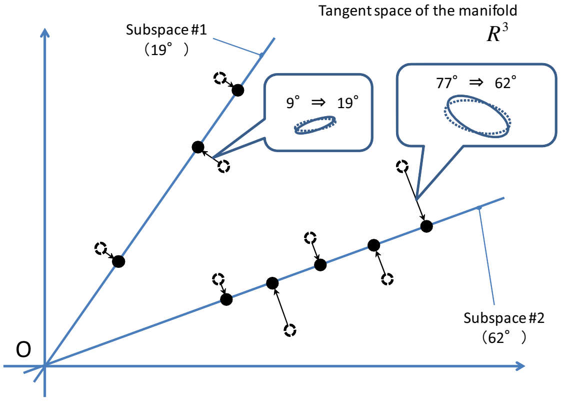

Fig. 2. *Riemannian subspace clustering corresponding to the example in Fig. 1. The set of K subspaces that best fits the given samples are found. All the samples on a subspace have the same rotation.*

The update step is carried out for each subspace as follows. The sum of squared distances from the assigned samples to the subspace, $J(\boldsymbol{U})$, defined below, is minimized with respect to the rotation matrix $\boldsymbol{U}$,

$$J(\boldsymbol{U}) = \sum_{\boldsymbol{\xi}_i\in\mathcal{S}} \parallel \boldsymbol{\xi}_i - \boldsymbol{P}(\boldsymbol{U})\,\boldsymbol{\xi}_i \parallel^2, \qquad (11)$$

where $\mathcal{S}$ is the set of samples assigned to this subspace, and $\boldsymbol{P}(\boldsymbol{U})$ is the orthogonal projection matrix derived from $\boldsymbol{U}$. An algorithm to minimize objective functions with orthogonality constrains can be found in [22], and is used to minimize $J(\boldsymbol{U})$ in this work. Specifically, a gradient decent method on the Stiefel manifold is used. To do so, we need partial derivative $\frac{\partial J}{\partial \boldsymbol{U}}$, but fortunately the projection matrix is simplified as $\boldsymbol{P}(\boldsymbol{U}) = \boldsymbol{A}\boldsymbol{A}^\top$, so the partial derivative can be calculated easily.

Because the rotation-tied covariance is a special case of the semi-tied covariance[4], acoustic models obtained via rotation-tying can be refined with the re-estimation formula developed for semi-tied covariance [3].

Algorithm 1: Riemannian Subspace Clustering

Input: A set of covariance matrices, $\{\boldsymbol{\Sigma}_i\}_{i=1}^N$, and the number of rotation classes, K

```
1. For $i = 1..N$, calculate $\boldsymbol{\xi}_i = \mathrm{vec}(\log(\boldsymbol{\Sigma}_i))$
2. For $k = 1..K$, initialize $\boldsymbol{U}_k$; for instance, randomly select
   one of $\{\boldsymbol{\Sigma}_i\}_{i=1}^N$ and use its rotation
```

[4] The semi-tied covariance matrix is written as $\boldsymbol{\Sigma}_m = \boldsymbol{H}_k\boldsymbol{D}_m\boldsymbol{H}_k^\top$, where $\boldsymbol{H}_k$ is a general linear matrix and $\boldsymbol{D}_m$ is a diagonal matrix. The rotation-tied covariance matrix is a special case where $\boldsymbol{H}_k$ is an orthogonal matrix.

```
3. Repeat until convergence
    (a) For $i = 1..N$, assign $\boldsymbol{\xi}_i$ to the nearest subspace
    (b) For $k = 1..K$, update the $k$-th subspace ($\boldsymbol{U}_k$) to fit the assigned samples
4. Output class assignments and rotations $\{\boldsymbol{U}_k\}_{k=1}^{K}$
```

6. EXPERIMENTS

Experiments were conducted using the Wall Street Journal (WSJ) corpus to evaluate different types of covariance matrices, including diagonal, semi-tied (global, phone-based, and proposed tyings), and full. The SI-284 set (29,735 utterances) of the WSJ corpus was used to train acoustic models. Three-state, left-to-right hidden Markov models were used to represent cross-word triphones, where 2,368 tied-states were used. All the models were trained with the maximum-likelihood criterion. Mel-frequency cepstral coefficients of c_0 to c_{12}, their Δ, and $\Delta\Delta$ were used with cepstral mean normalization as features ($n = 39$). The standard WSJ 5k-word bigram was used as the language model. The November-92 set (330 utterances) was used to evaluate the word error rate for each acoustic model.

6.1. Diagonal vs. full covariance

Diagonal covariance acoustic models with 1 to 16 mixture components per state were trained in a standard mix-up process. The number of components per state was incremented by two at a time, and between each increment, four iterations of B-W re-estimation were conducted. For each of the diagonal covariance acoustic models, diagonal covariances were switched to full ones with off-diagonal elements zero, and four iterations of B-W were added to create a full covariance acoustic model. In doing so, the full covariance was smoothed with the diagonal one as proposed in [23]; i.e. the off-diagonal elements of the covariance were discounted by a scale $\frac{c}{c+\tau}$, where c is the count of frames assigned to the component, and τ is the smoothing constant (set to 100 in this work).

Table 1 shows the comparative results of the diagonal and full covariance models. For every case, the full covariance performed significantly better that the diagonal one. Though, the full covariance cannot be used in practice due to its huge computational cost. A common way to approximate the full covariance at a low cost is the semi-tied covariance, which is evaluated next.

Table 1. Word error rates (%) of diagonal and full covariance acoustic models on the November '92 set.

Covariance	1-mix	2-mix	4-mix	8-mix	16-mix
Diagonal	11.62	8.74	6.73	5.66	5.19
Full	6.89	5.51	4.50	4.20	4.00

6.2. Semi-tied covariance with different tyings

For each of the 4- and 8-mix full covariance acoustic models, an optimal tying structure was found by Riemannian subspace clustering, and rotations of covariance matrices were tied to create a semi-tied covariance acoustic model. The maximum-likelihood estimation procedure proposed by Gales [3] was then used to refine the tied-transforms, and four iterations of B-W were added to finalize the acoustic model. On the other hand, for comparison, two other semi-tied covariance acoustic models were made in a conventional way. Specifically, given the diagonal covariance acoustic model described above, semi-tied transform(s) were estimated with either global or phone-based[5] tying, and four iterations of B-W re-estimation were added.

Table 2 shows the word error rate of each acoustic model. Semi-tied covariance acoustic models fell somewhere in between the diagonal and full covariance models. The ones with global tying were significantly better than the diagonal covariance models, which shows the importance of modeling correlations between feature dimensions (i.e. rotations of covariances). The phone-based tying brought only marginal gains over the global tying, which is a similar result to the one reported in [11]. The result indicates that naive tying structures cannot bring big gains even though the number of classes is increased. In contrast, with the same number of classes, our proposed tying has brought better recognition accuracies. Further, by increasing the number of classes to 80, additional improvements were obtained with the proposed tying. The result led us to believe that optimally determined tying structures can push the semi-tied covariance closer to the full covariance, and hence to an improved recognition performance.

Table 2. Word error rates (%) on November '92 for acoustic models with different types of covariance matrices.

Covariance	Tying	4-mix	8-mix
Diagonal	-	6.73	5.66
Semi-tied	Global	5.81	5.34
Semi-tied	Phone (40)	5.75	5.31
Semi-tied	**RSC (K=40)**	**5.49**	**4.88**
Semi-tied	**RSC (K=80)**	**5.25**	**4.52**
Full	-	4.50	4.20

7. CONCLUSIONS

A method of tying rotations of full covariance matrices for fast likelihood calculation was proposed. Our contribution is twofold. First, we have shown a property of a tangent space of the Riemannian manifold of covariance matrices; namely, covariance matrices having the same rotation to each other lie on the same subspace in the tangent space. Secondly, exploiting this property, we have derived an algorithm called Riemannian subspace clustering for finding and tying clusters of covariance matrices that have similar rotations. Our tying performed significantly better than conventional tyings (global and phone-based) in experiments using the WSJ corpus. Application of the Riemannian framework is not limited to the one presented in this paper. We will apply the framework to other problems in acoustic modeling, speaker adaptation, and robustness in our future work.

8. REFERENCES

[1] P. Bell, *Full Covariance Modelling for Speech Recognition*, PhD thesis, Edinburgh University, 2010.

[2] P. Olsen, V. Goel, S.J. Rennie, "Discriminative training for full covariance models," in *Proc. ICASSP*, 2011.

[5] The class of each component was determined according to the central-phoneme of the triphone (i.e. the number of classes was 40).

[3] M.J.F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, 7(3):272–281, 1999.

[4] J. Schalkwyk, D. Beeferman, F. Beaufays, B. Byrne, C. Chelba, M. Cohen, M. Kamvar, B. Strope, "Google Search by Voice: A Case Study," in *Advances in Speech Recognition: Mobile Environments, Call Centers and Clinics*, Springer, pp. 61–90, 2010.

[5] H. Soltau, G. Saon, and B. Kingsbury, "The IBM Attila speech recognition toolkit", in *Proc. IEEE Spoken Language Technology Workshop*, 2010.

[6] G. Saon and J.-T. Chien, "Large-Vocabulary Continuous Speech Recognition Systems: A Look at Some Recent Advances," *IEEE Signal Processing Magazine*, 29(6):18–33, November 2012.

[7] R.A. Gopinath, "Maximum likelihood modeling with Gaussian distributions for classcation," in *Proc. ICASSP*, 1998.

[8] P. Olsen and R.A. Gopinath, "Modelling inverse covariance matrices by basis expansion," *IEEE Transactions on Speech and Audio Processing*, 12(1):37–46, 2004.

[9] S. Axelrod, V. Goel, R.A. Gopinath, P.A. Olsen, and K. Visweswariah, "Subspace constrained Gaussian mixture models for speech recognition," *IEEE Transactions on Speech and Audio Processing*, 13(6):1144–1160, 2005.

[10] M.J.F. Gales, "Maximum likelihood multiple subspace projections for hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, 10(2):37–47, 2002.

[11] K.-C. Sim, and M.J.F. Gales, "Minimum Phone Error Training of Precision Matrix Models," *IEEE T-ASLP*, 14(3):882–889, 2006.

[12] G. Saon, D. Povey, and H. Soltau, "Large margin semi-tied covariance transforms for discriminative training," in *Proc. ICASSP*, 2009.

[13] X. Cui, J. Xue, X. Chen, P.A. Olsen, P.L. Dognin, U.V. Chaudhari, J.R. Hershey, and Z. Bowen, "Hidden Markov Acoustic Modeling with Bootstrap and Restructuring for Low-Resourced Languages," *IEEE Transactions on Audio, Speech, and Language Processing.*, 20(8):2252–2264, October 2012.

[14] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *International Journal of Computer Vision*, 66(1):41–66, 2006.

[15] C. Lenglet, M. Rousson, R. Deriche, and O. Faugeras, "Statistics on the manifold of multivariate normal distributions: Theory and application to diffusion tensor MRI processing," *Journal of Mathematical Imaging and Vision*, 25(3):423–444, 2006.

[16] Y. Shinohara, T. Masuko, and M. Akamine, "Covariance clustering on Riemannian manifolds for acoustic model compression," in *Proc. ICASSP*, 2010.

[17] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on Riemannian manifolds," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(10):1713–1727, 2008.

[18] F. Porikli, A. Srivastava, P. Turaga, and A. Veeraraghavan, *Differential Geometric Methods for Shape Analysis and Activity Recognition*, Tutorial at IEEE Inernational Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

[19] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magnetic Resonance in Medicine*, 56:411–421, 2006.

[20] P. Fillard, X. Pennec, V. Arsigny, and N. Ayache, "Clinical DT-MRI estimation, smoothing and fiber tracking with log-Euclidean metrics," *IEEE Trans. Medical Imaging*, 26(11):1472–1482, 2007.

[21] R. Vidal, "Subspace Clustering," *IEEE Signal Processing Magazine*, 28(2):52–68, 2011.

[22] A. Edelman, T. Arias, and S. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1998.

[23] D. Povey, *Discriminative Training for Large Vocabulary Speech Recognition*, PhD thesis, Cambridge University, 2003.