RECONSTRUCTION OF ARTICULATORY MEASUREMENTS WITH SMOOTHED LOW-RANK MATRIX COMPLETION

Weiran Wang — TTI-Chicago — weiranwang@ttic.edu

Raman Arora — Johns Hopkins University — arora@cs.jhu.edu

Karen Livescu — TTI-Chicago — klivescu@ttic.edu

ABSTRACT

Articulatory measurements have been used in a variety of speech science and technology applications. These measurements can be obtained with a number of technologies, such as electromagnetic articulography and X-ray microbeam, typically involving pellets attached to individual articulators. Due to limitations in the recording technologies, articulatory measurements often contain missing data when individual pellets are mis-tracked, leading to relatively high rates of loss in this expensive and time-consuming data source. We present an approach to reconstructing such data, using low-rank matrix factorization techniques combined with temporal smoothness regularization, and apply it to reconstructing the missing entries in the Wisconsin X-ray microbeam database. Our algorithm alternates between two simple steps, each having a closed form as the solution of a linear system. The algorithm gives realistic reconstructions even when a majority of the frames contain missing data, improving over previous approaches to this problem in terms of both root mean squared error and phonetic recognition performance when using the reconstructions.

Index Terms— articulatory data, X-ray microbeam, missing data, matrix factorization

1. INTRODUCTION

Articulatory measurements are a valuable resource for a number of spoken language technology applications. For example, in speech synthesis they have been used to generate speech from articulation [1, 2, 3]. They have been used to train acoustic-to-articulatory inversion models with application, for example, in speech recognition [4, 5, 6, 7]. In speech recognition they have also been used for multi-view acoustic feature learning [8, 9]. There are a number of ways of simultaneously recording acoustic and articulatory data, including X-ray microbeam [10], electromagnetic articulography (EMA) [11], ultrasound [12], and magnetic resonance imaging (MRI) [13].

This research was supported by NSF grant IIS-1321015. The opinions expressed in this work are those of the authors and do not necessarily reflect the views of the funding agency.

We are mainly concerned with articulatory measurements corresponding to the spatial location of pellets attached to several articulators, as in EMA and X-ray microbeam, and we focus our efforts on data from the Universty of Wisconsin X-ray microbeam database (XRMB) [10]. Due to limitations of the recording technology, articulatory measurements often contain frames where one or more pellets' coordinates are missing. In the case of X-ray microbeam recordings, pellets are often mis-tracked for a part of an utterance for roughly 50-500ms at a time [10] (see Fig. 1 (left) for sample mis-track patterns). Since it is prohibitively expensive to record perfectly clean articulatory measurements, such mis-tracked records are left as is and only annotated as mis-tracked. The runs of missing data are sufficiently long that reconstruction via single-dimension interpolation is not feasible.

Although the overall proportion of missing data in a database may be low, the proportion of affected frames is much higher. The subset of XRMB used in this paper includes 47 speakers uttering 53 utterances. In this data set, 3.4% of the entries are missing, yet 23.6% of the frames contain at least one missing entry, and the proportions of missing data vary greatly between speakers. Overall, XRMB is reported to have about 35% affected utterances [10].

There have been several approaches applied to reconstructing the missing entries of articulatory recordings. Roweis [14] takes an approach based on probabilistic principal component analysis which employs Expectation Maximization (EM). Qin and Carreira-Perpñan [15] model the fully observed frames with Gaussian mixtures and impute the missing values based on conditional statistics of the missing dimensions given the observed dimensions.

The task can be viewed as the problem of completing a matrix from a few given entries. This is a fundamental problem with many applications in machine learning, computer vision, network engineering, and data mining. Much interest in matrix completion has been caused by recent theoretical breakthroughs in compressed sensing [16, 17], as well as by the celebrated Netflix challenge on practical prediction problems such as user ratings prediction [18, 19]. Many matrix completion approaches assume that the underlying data matrix is low-rank [16, 20, 21], as a simple way of constraining the degrees of freedom in the model.

The typical pattern of missing articulatory data is quite different from that in other domains such as user ratings in recommender systems, which have a very high missing data proportion. More importantly, articulatory measurements have a sequential structure: We know the time ordering of the recordings, and that the trajectories of articulators should vary smoothly over time due to physical constraints. Therefore, it is natural to combine matrix completion techniques with temporal smoothness constraints for reconstructing missing articulatory data. In the following, we present one such approach and reconstruct all missing measurements simultaneously for each speaker (without adaptation), making use of both fully observed and partially observed frames. In the remainder of the paper, we introduce our approach and give an optimization procedure, discuss closely related approaches, and demonstrate our approach in terms of reconstruction error and speech recognition using reconstructed measurements.

2. SMOOTHED LOW-RANK MATRIX COMPLETION

In the following, we denote by $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D\times N}$ the articulatory measurements over N successive frames, where each column of the matrix corresponds to the $D = 16$ dimensional articulatory measurements in a time frame. In our case there are 100 frames per second (downsampled from the original XRMB frame rate). Let $\mathbf{M} \in \mathbb{R}^{D\times N}$ be a binary matrix with $\mathbf{M}_{ij} = 1$ if $\mathbf{X}_{ij}$ is observed and 0 otherwise, for $i = 1, \ldots, D$, $j = 1, \ldots, N$.

We denote by $\odot$ the element-wise multiplication between two matrices, and by $\otimes$ the Kronecker ("outer") product. We use $\mathbf{M}^i$ ($\mathbf{M}_j$) to indicate the i-th row (j-th column) of the matrix $\mathbf{M}$, $\mathrm{diag}\,(\mathbf{v})$ the diagonal matrix with elements of vector $\mathbf{v}$ on the diagonal, and vec($\mathbf{V}$) the vector obtained by concatenating the columns of matrix $\mathbf{V}$.

2.1. Objective function

In low-rank matrix completion, we approximate the underlying data matrix $\mathbf{X}$ as the multiplication of two matrices, $\mathbf{X} \approx \mathbf{U}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{D\times k}$, $\mathbf{V} \in \mathbb{R}^{N\times k}$, and $k < \max\{D, N\}$ so that the approximation is low-rank. Equivalently, each frame is approximated as a linear combination of k basis vectors (columns of $\mathbf{U}$). On the one hand, we would like the approximation to be as close to the observed entries as possible, i.e., $\left|\mathbf{X}_{ij} - (\mathbf{U}\mathbf{V}^\top)_{ij}\right|$ should be small if $\mathbf{X}_{ij}$ is not missing. On the other hand, we want the trajectory to be smooth over time, i.e., the difference between successive frames $\|\mathbf{x}_{i+1} - \mathbf{x}_i\|$ should be small. This suggests a smoothness penalty $\sum_{j=1}^{N-1} \|\mathbf{x}_{j+1} - \mathbf{x}_j\|^2 = \mathrm{tr}\left(\mathbf{X}\mathbf{L}\mathbf{X}^\top\right)$ with

$$\mathbf{L} = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix} \tag{1}$$

But since $\mathbf{X}$ is not fully observed, we shall instead impose the smoothness penalty on the low-rank approximation. Combining the two intuitions gives the following objective function:

$$\min_{\mathbf{U},\mathbf{V}} \quad \left\|\mathbf{M} \odot (\mathbf{X} - \mathbf{U}\mathbf{V}^\top)\right\|_F^2 + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) + \gamma\,\mathrm{tr}\left(\mathbf{U}\mathbf{V}^\top\mathbf{L}\mathbf{V}\mathbf{U}^\top\right), \tag{2}$$

where $\|\cdot\|_F$ is the Frobenius norm and $\lambda, \gamma > 0$ are tradeoff parameters for the $L2$ and smoothness penalties respectively. The $L2$ term functions like a Gaussian prior on $\mathbf{U}$ and $\mathbf{V}$, and also helps avoid numerical instability as described in Section 2.2. Once the factors $(\mathbf{U}, \mathbf{V})$ are obtained by solving (2), the missing entries of $\mathbf{X}$ are filled with the corresponding entries of $\mathbf{U}\mathbf{V}^\top$.

Without the smoothness penalty (i.e., $\gamma = 0$), the above objective reduces to one that is widely used in the matrix completion and collaborative filtering literature and leads to the alternating least squares (ALS) minimization algorithm [18]. This approach has been very successful for recommender systems, where it is widely believed that there are only a few latent factors that contribute to the user ratings.

2.2. Optimization

The objective function is convex and quadratic in $\mathbf{U}$ if $\mathbf{V}$ is fixed and vice versa. This naturally leads to alternating optimization on the two sets of variables. Compared to [18], the added smoothness penalty term complicates the optimization, but we still have a closed-form solution for each step.

U-step For fixed $\mathbf{V}$, we compute the gradient of the objective (2) with respect to $\mathbf{U}$ and set it to zero to obtain the following linear system:

$$(\mathbf{M} \odot (\mathbf{U}\mathbf{V}^\top - \mathbf{X}))\mathbf{V} + \lambda\mathbf{U} + \gamma\mathbf{U}(\mathbf{V}^\top\mathbf{L}\mathbf{V}) = \mathbf{0}.$$

We can further decompose the linear system into a $k \times k$ system for each row i of $\mathbf{U}$:

$$\mathbf{U}^i\mathbf{V}^\top\mathrm{diag}\left(\mathbf{M}^i\right)\mathbf{V} + \lambda\mathbf{U}^i + \gamma\mathbf{U}^i(\mathbf{V}^\top\mathbf{L}\mathbf{V}) = \mathbf{X}\,\mathrm{diag}\left(\mathbf{M}^i\right)\mathbf{V},$$

so that each row of $\mathbf{U}$ can be solved in closed form as

$$\mathbf{U}^i = \mathbf{X}\,\mathrm{diag}\left(\mathbf{M}^i\right)\mathbf{V}(\mathbf{V}^\top\,\mathrm{diag}\left(\mathbf{M}^i\right)\mathbf{V} + \lambda\mathbf{I} + \gamma\mathbf{V}^\top\mathbf{L}\mathbf{V})^{-1}.$$

V-step For fixed $\mathbf{U}$, we compute the gradient of the objective (2) with respect to $\mathbf{V}$ and set it to zero to obtain the following linear system:

$$(\mathbf{M}^\top \odot (\mathbf{V}\mathbf{U}^\top - \mathbf{X}^\top))\mathbf{U} + \lambda\mathbf{V} + \gamma\mathbf{L}\mathbf{V}\mathbf{U}^\top\mathbf{U} = \mathbf{0}. \tag{3}$$

Without the smoothness penalty term, $\mathbf{V}$ can be obtained similarly to $\mathbf{U}$ by solving a $k \times k$ system for each row separately. However, the smoothness regularization couples rows of $\mathbf{V}$ together, i.e., for each row j, the above system reduces to

$$\mathbf{V}^j\mathbf{U}^\top\,\mathrm{diag}\,(\mathbf{M}_j)\,\mathbf{U} + \lambda\mathbf{V}^i + \gamma\mathbf{L}^j\mathbf{V}(\mathbf{U}^\top\mathbf{U}) = (\mathbf{X}_j)^\top\,\mathrm{diag}\,(\mathbf{M}_j)\,\mathbf{U},$$

where the last term on the left contains all rows of $\mathbf{V}$. Notice that (3) is essentially a Sylvester equation for which iterative solvers exist [22]. Alternatively, we could rewrite it as

$$(\mathbf{K} + \lambda\mathbf{I} + \gamma\mathbf{L} \otimes (\mathbf{U}^\top\mathbf{U})) \cdot \text{vec}(\mathbf{V}^\top) = \text{vec}(\mathbf{U}^\top(\mathbf{M} \odot \mathbf{X})),$$

where

$$\mathbf{K} = \begin{bmatrix} \mathbf{U}^\top \operatorname{diag}(\mathbf{M}_1)\,\mathbf{U} & & \\ & \ddots & \\ & & \mathbf{U}^\top \operatorname{diag}(\mathbf{M}_N)\,\mathbf{U} \end{bmatrix}.$$

Therefore, $\mathbf{V}$ can still be obtained in closed form by solving the above $NK \times NK$ sparse linear system thanks to the sparsity in $\mathbf{L}$.

The $\mathbf{U}$/$\mathbf{V}$-step of the algorithm clearly finds the unique minimum given the other set of parameters and thus decreases the overall objective. All matrices to be inverted are positive semidefinite and in fact positive definite if $\lambda > 0$. Thus using a small positive λ improves numerical stability for each step. As initialization, we fill the missing entries with zeros and compute the truncated SVD to obtain $\mathbf{U}$ and $\mathbf{V}$.

3. RELATED WORK

Roweis described his algorithm for reconstructing missing data as a modified EM algorithm for PCA [14, page 49]. The algorithm can be viewed as alternating optimization of the following objective:

$$\min_{\mathbf{X},\mathbf{U},\mathbf{V}} \left\|\mathbf{X} - \mathbf{U}\mathbf{V}^\top\right\|_F^2 \tag{4}$$

The algorithm consists of:

- generalized E-step: for fixed basis $\mathbf{U}$, compute the latent representation $\mathbf{V}$ using partially observed dimensions for each frame, and fill in the missing entries of $\mathbf{X}$ with corresponding entries in $\mathbf{U}\mathbf{V}^\top$.

- M-step: for fixed $\mathbf{V}$, compute the basis of the principal subspace $\mathbf{U}$ by solving a linear system $\mathbf{U} = \mathbf{X}\mathbf{V}(\mathbf{V}^\top\mathbf{V})^{-1}$

Our approach is similar to this one when we do not use any regularization ($\lambda = \gamma = 0$), but there are differences in the optimization parameters and error function: Our approach does not optimize over $\mathbf{X}$ while Roweis' does; and Roweis' algorithm requires filling in the missing entries in $\mathbf{X}$ at every M-step, whereas in our approach we only fill them in once at the end. However, the two approaches ultimately aim to minimize the same approximation error only at observed entries during training, and both fill in the missing entries with corresponding entries of $\mathbf{U}\mathbf{V}^\top$. In our experience, our optimization empirically converges much faster, presumably because we take into account missing entries in both steps. Finally, we find the regularization in our approach to be important for better reconstruction, whereas Roweis' approach does not include regularization.

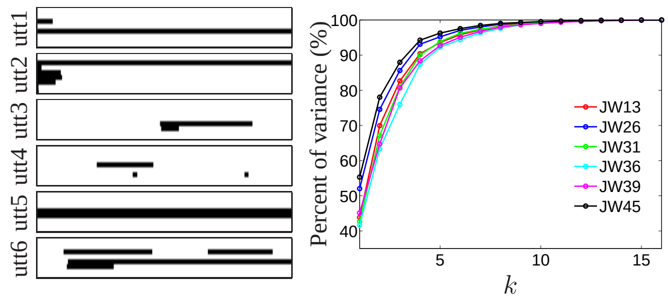

Fig. 1. XRMB data characteristics. Left: missing data patterns for 6 utterances (white/black pixels denote observed/missing entries); rows correspond to the 16 articulatory dimensions and columns correspond to time. Right: percentage of total variance contained in the first k principal components for several speakers' articulatory measurements.

In Qin and Carreira-Perpiñán's approach [15], each frame of articulatory data (16 dimensions) is modeled with a Gaussian mixture model (GMM), and the missing entries are reconstructed as the mean of the conditional distribution of missing entries given the observed entries (which is again a Gaussian mixture). The GMM parameters are learned on fully observed frames, so the potentially useful information in the large amount of partially observed frames is unused. The approach cannot succeed when few fully observed frames are available, and this is the case for several speakers in XRMB; for example, speaker JW29 has only 810 fully observed frames out of $51\,608$ frames in our data set, which is insufficient to learn an accurate Gaussian mixture model.

There are also several related matrix completion algorithms from the machine learning literature. For example, Candes and Tao [17] minimize the nuclear norm (sum of singular values) as a convex surrogate for rank. Jain *et al.* [21] directly solve for a matrix $\mathbf{X}$ that agrees on observed entries as much as possible, subject to the (nonconvex) constraint $rank(\mathbf{X}) \leq k$. Keshavan *et al.* [20] proposed initializing the solution using SVD after trimming (zeroing out rows and columns with too few entries) the input matrix, followed by a greedy minimization of the residual error.

The most important distinction between our approach and the above related work is that we explicitly model the temporal smoothness in our time series data whereas the above approaches ignore sequential structure, and would produce identical results even if the frames were shuffled.

4. EXPERIMENTAL RESULTS

4.1. Data

The XRMB database [10] consists of simultaneously recorded speech and articulatory measurements from 47 American English speakers (22 males, 25 females). Each speaker's

Table 1. Missing data proportions for several speakers.

Speaker	# Frames	Missing Frames (%)	Missing Entries (%)
JW11	54880	14.4	1.9
JW15	56849	78.0	10.7
JW29	51608	98.4	13.9
JW30	54809	20.6	3.4

Table 2. Reconstruction errors (RMSE) obtained by different algorithms for artificially blacked-out data.

Source	Target	Ref	GMM	Ours ($\lambda = 0$, $\gamma = 0$)	Ours ($\lambda = 0$)	Ours ($\gamma = 0$)	Ours
JW11	JW13	17.77	5.08	1.70	1.65	1.52	1.51
	JW26	18.44	2.37	1.71	1.68	1.40	1.40
	JW31	15.71	2.48	1.85	1.81	1.59	1.58
	JW45	19.78	1.47	1.43	1.38	1.38	1.37
JW15	JW13	27.70	7.66	2.00	1.89	1.24	1.24
	JW26	29.52	17.34	2.57	2.12	1.29	1.29
	JW31	25.71	7.12	2.60	1.90	1.40	1.39
	JW45	32.05	13.41	3.10	1.83	1.37	1.36
JW29	JW13	25.51	17.25	1.97	1.81	1.84	1.63
	JW26	23.13	13.21	2.10	1.99	1.33	1.32
	JW31	21.82	14.81	1.42	1.42	1.38	1.19
	JW45	24.95	13.67	1.88	1.38	1.45	1.20
JW30	JW13	21.65	2.59	6.51	1.69	6.38	1.69
	JW26	22.42	4.83	6.64	2.13	6.51	2.10
	JW31	19.72	7.14	5.87	1.85	5.76	1.83
	JW45	25.70	2.90	1.89	1.80	1.36	1.35

recordings comprise approximately 20 minutes of read speech including multi-sentence recordings, individual sentences, isolated word sequences, and number sequences, as well as non-speech oral motor tasks. We exclude utterances corresponding to isolated words and oral motor tasks, leaving up to 53 utterances per speaker. The utterance texts are identical for all of the speakers; this is important in our evaluation, as described below. The articulatory measurements are horizontal and vertical displacements of 8 pellets on the speaker's tongue, lips, and jaw. We downsample the articulatory data from an original rate of 145.6542 Hz to 100Hz to match the frame rate of our acoustic features (mel-frequency cepstral coefficients (MFCCs) computed every 10ms).

4.2. Validation of the low-rank assumption

We first select 6 speakers with $< 1\%$ missing entries and $< 5\%$ missing frames, and plot the eigen-spectrum computed by PCA on fully observed frames for each speaker in Figure 1 (right). It is clear that the eigen-spectrum decays quickly such that the first few principal components contain most of the total variance.

4.3. Reconstructing artificially blacked-out data

We then design a mechanism for testing our algorithm and selecting hyperparameters (rank k, regularization parameters λ and γ). We follow the previous work of [14] and [15] and create artificially blacked-out entries that are held out for training, and evaluate the reconstructions by computing the errors at these ground-truth entries. We try to mimic the natural missing data pattern in XRMB by copying the patterns from one speaker to another. For example, suppose speaker JW29's data contains missing entries; then we select a different speaker, JW13, whose articulatory measurements are mostly complete, and remove entries from JW13's data corresponding to the ones missing from JW29, after linearly warping the two speakers' data to the same length. After reconstructing the artificially missing data of JW13, we evaluate the results by computing the root mean squared error (RMSE, in millimeters) of the reconstructions at those entries that are artificially blacked-out for JW13. In the following, we transfer the missing data patterns of source speakers $\{$JW11, JW15, JW29, JW30$\}$ to four target speakers $\{$JW13, JW26, JW31, JW45$\}$. Table 1 shows the proportions of missing data for the four source speakers. This problem setting is more challenging than that of [15], where several utterances from two speakers with low missing data proportions were selected and one pellet (2 out of 16 dimensions) at a time was blacked out and reconstructed. In practice, the patterns of missing data are very different between the pellets and there is much more missing data for most speakers.

We reconstruct all utterances for each target speaker at once, so all utterances share the same basis $\mathbf{U}$, while the smoothness penalty is only imposed within each utterance. We do not run our algorithm on each utterance separately as pellets are sometimes missing for entire utterances, so there is insufficient information to reconstruct these dimensions using a low-rank matrix factorization model. We select 50% of the blacked-out entries as a tuning set for hyper-parameter selection and the other 50% for testing. Hyper-parameter selection is done via grid search for rank k in $\{2,\ 4,\ 6,\ 8,\ 10,\ 12,\ 14,\ 16\}$ and λ, γ in $\{0,\ 10^{-2},\ 10^{-1},\ 1,\ 10,\ 10^{2}\}$ for our algorithm. For comparison, we have also implemented the Gaussian mixture model (GMM) of [15]. For the GMM algorithm we tune the number of Gaussian components M in $\{1,\ 2,\ 4,\ 8,\ 16,\ 32,\ 64\}$ and train with EM.

The test set RMSEs obtained for different (source, target) pairs are shown in Table 2. Results are also provided for special cases of our algorithm: no regularization at all ($\lambda = 0$, $\gamma = 0$, roughly corresponding to Roweis' approach), no L2 regularization ($\lambda = 0$), and no smoothness regularization ($\gamma = 0$). As a reference, we show the RMSE obtained by filling all missing entries with zeros, denoted Ref (this is in fact the initialization for our algorithm).

From the results it is clear that regularization (L2 or smoothness) improves performance, and the two regulariza-

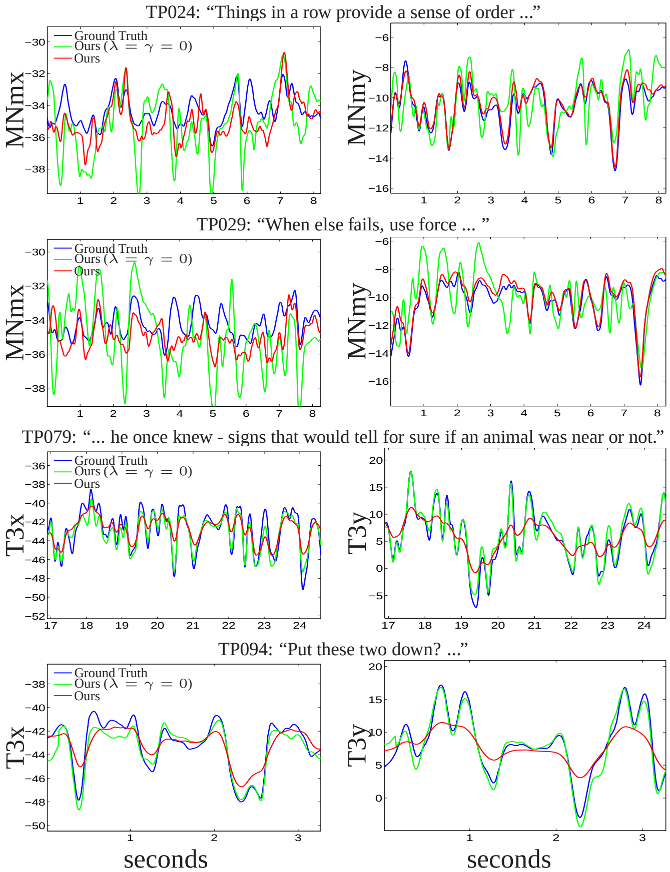

Fig. 2. Sample reconstructions of the horizontal (left) and vertical (right) coordinates of the mandibular and mid-tongue pellets. The GMM-based reconstructions are far beyond the range of the pellet locations and are not shown.

tions are complementary. When no regularization is used, the best reconstruction is obtained at a relatively low rank (4 or 6, as Roweis suggested). With regularization, even better reconstruction can be obtained by our algorithm at a higher rank. We also note that GMMs work well when the missing proportion is very low (e.g., when JW11 is the source speaker), in which case the optimal number of Gaussian components M is larger. But when most frames are missing, discarding those frames entirely loses too much information, and the GMM approach tends to select very small M and perform poorly.

Figure 2 shows sample reconstructions of the mandibular (MNm) and mid-tongue (T3) pellets for several utterances. In this case the reconstructions were obtained with the optimal hyperparameters (based on overall RMSE) when we reconstruct JW45's data based on the missing data patterns of JW29. In this experiment, only 1.6% of the total frames include the mandibular pellet, and the utterances shown in the figure have this pellet missing entirely; the algorithms must infer the missing entries from the few observations of this pellet and information from other pellets. In this very challenging condition, we are able to reconstruct well the rapidly oscillating trajectories with low-rank matrix factorization, while the regularized version improves over the unregularized version. For the mid-tongue pellet, which is missing for only a short duration, the unregularized algorithm works better, indicating that the smoothness regularization selected globally for all pellets is somewhat too strong for this particular pellet. However, T3 is somewhat of an outlier: looking at all of the pellets individually, it is almost always the case that our algorithm with some non-zero regularization outperforms the unregularized version, and for some pellets the smoothing and/or L2 regularization makes a very large difference.

Table 3. Phonetic error rates (PER) of recognition using the baseline features and concatenations of the baseline features with reconstructed articulatory measurements.

Method	PER (%)
Baseline (MFCCs only)	31.1
GMM	22.0
Ours ($\lambda = \gamma = 0$)	20.4
Ours	20.0

4.4. Phonetic recognition with reconstructed data

Next, we consider what effect the differences in reconstruction performance may have on downstream tasks of interest. Many have found that appending articulatory measurements to acoustic features improves speech recognition performance (e.g., [4]), and we test our reconstructions on this task.

First, we select the optimal hyperparameters for each algorithm based on the average performance on all of the above (source, target) pairs and use them to reconstruct all of the data in our XRMB data set. There is a wide range of hyper-parameter combinations at which our algorithm performs similarly well, but we use ($k = 6$, $\lambda = 1$, $\gamma = 1$) for our algorithm with full regularization and $k = 4$ for the unregularized ($\lambda = \gamma = 0$) special case. Since the performance of the GMM approach varies a great deal depending on the missing data proportion, we set M for each speaker to match the source speaker from {JW11, JW15, JW29, JW30} with the closest missing data proportion.

We use disjoint sets of 14/9/9 speakers for recognizer training/tuning/testing. The recognizer is a basic 3-state left-to-right monophone HMM-based model, where each state has a GMM observation model with 32 components. The baseline acoustic features are 13 MFCCs appended with first and second derivatives. The articulatory measurements are concatenated over a 7-frame window around each frame, and their dimensionality is then reduced with PCA. Table 3 reports the phone error rates (PER) obtained on the test speakers when using only the baseline MFCCs and when appending with reconstructed articulatory measurements produced by different methods. As expected, appending the articulatory data always improves recognition performance over the baseline (up to 11% absolute and more than 33% relative). Our smoothed low-rank reconstruction algorithm performs much better than the GMM approach and slightly better than the unregularized special case. The difference in performance between our algorithm and its unregularized version is significant at a level of $p < 0.01$ according to a Matched Pair Sentence Segment (Word Error) test [23].

5. FUTURE DIRECTIONS

We have proposed a simple algorithm for reconstructing missing articulatory measurements based on low-rank matrix completion and temporal smoothness regularization. It achieves good reconstruction error compared to previous approaches, and the reconstructed articulatory data improves the performance of a phonetic speech recognizer.

There are several natural directions for future work. First, the globally linear assumption underlying low-rank matrix completion might be unrealistic, and one can instead model the data as approximately lying on the union of multiple subspaces [14], or on a low-dimensional nonlinear manifold [24, 25]. Second, we have not used the simultaneously recorded acoustic data that is available in the XRMB data, which contains complementary information that may be useful for reconstruction. Third, our smoothness penalty can be considered to be a simple dynamic model that encourages nearby frames to be similar, and it is possible to extend it to richer dynamic models and to pellet-specific smoothing. Finally, our approach does not handle the (infrequent) case of a pellet that is missing from most or all of a speaker's data; for this purpose adaptation approaches can be considered for applying one speaker's reconstruction model to another speaker [26].

Acknowledgement

We thank Louis Goldstein for providing phonetic alignments for the data used in the recognition experiments.

6. REFERENCES

[1] T. Toda, A. W. Black, and K. Tokuda, "Mapping from articulatory movements to vocal tract spectrum with Gaussian mixture model for articulatory speech synthesis," *ISCA Speech Synthesis Workshop* 2004.

[2] T. Kaburagi and M. Honda, "Determination of the vocal tract spectrum from the articulatory movements based on the search of an articulatory-acoustic database," *ICSLP* 1998.

[3] Z.-H. Ling, K. Richmond, J. Yamagishi, and R.-H. Wang, "Integrating articulatory features into HMM-based parametric speech synthesis," *IEEE Trans. Audio, Speech, and Lang. Proc.* Vol. 17, No. 6, pp. 1171–1185, 2009.

[4] A. A. Wrench and K. Richmond, "Continuous speech recognition using articulatory data," *Interspeech* 2000.

[5] J. Frankel and S. King, "ASR — articulatory speech recognition," *Interspeech* 2001.

[6] F. Rudzicz, "Correcting errors in speech recognition with articulatory dynamics," *ACL* 2010.

[7] C. Canevari, L. Badino, L. Fadiga, and G. Metta, "Relevance-weighted-reconstruction of articulatory features in deep-neural-network-based acoustic-to-articulatory mapping," *Interspeech* 2013.

[8] R. Arora and K. Livescu, "Multi-View CCA-based acoustic features for phonetic recognition across speakers and domains," *ICASSP* 2013.

[9] R. Arora and K. Livescu, "Multi-view learning with supervision for transformed bottleneck features," *ICASSP* 2014.

[10] J. R. Westbury, *X-Ray Microbeam Speech Production Database User's Handbook Version 1.0*, Waisman Center on Mental Retardation & Human Development, University of Wisconsin, Madison, WI, June 1994.

[11] A. A. Wrench, "A multi-channel/multi-speaker articulatory database for continuous speech recognition research," *Phonus* 2000.

[12] C. Qin, M. Á. Carreira-Perpiñán, K. Richmond, A. Wrench, and S. Renals, "Predicting tongue shapes from a few landmark locations," *Interspeech* 2008.

[13] E. Bresch, J. Nielsen, K. Nayak, and S. Narayanan, "Synchronized and noise-robust audio recordings during realtime magnetic resonance imaging scans," *J. Acoustic Soc. Amer.*, Vol. 120, No. 4, pp. 1791–1794, 2006.

[14] S. T. Roweis, *Data Driven Production Models for Speech Processing*, Ph.D. Thesis, Cal. Inst. Of Tech., 1999.

[15] C. Qin and M. Á. Carreira-Perpiñán, "Estimating missing data sequences in X-ray microbeam recordings," *Interspeech* 2010.

[16] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations Of Computational Mathematics*, Vol. 9, No. 6, pp. 717–772, Dec. 2009.

[17] E J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Info. Theory*, Vol. 56, No. 5, pp. 2053–2080, Apr. 2010.

[18] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," *SIGKDD* 2008.

[19] R. Bell and Y. Koren, "Scalable collaborative filtering with jointly derived neighborhood interpolation weights," *ICDM* 2007.

[20] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Trans. Info. Theory*, Vol. 56, No. 6, pp. 2980–2998, 2010.

[21] P. Jain, R. Meka, and I. S. Dhillon, "Guaranteed rank minimization via singular value projection," *NIPS* 2010.

[22] D. Y. Hu and L. Reichel, "Krylov-subspace methods for the Sylvester equation," *Linear Algebra and its Applications*, Vol. 172, pp. 283–313, 1990.

[23] D. S. Pallet, W. M. Fisher, and J. G. Fiscus, "Tools for the analysis of benchmark speech recognition tests," *ICASSP* 1990.

[24] N. D. Lawrence and R. Urtasun, "Non-linear matrix factorization with Gaussian processes," *ICML* 2009.

[25] W. Wang, M. Á. Carreira-Perpiñán, and Z. Lu, "A denoising view of matrix completion," *NIPS* 2011.

[26] M. Farhadloo and M. Á. Carreira-Perpiñán, "Learning and adaptation of a tongue shape model with missing data," *ICASSP* 2012.