Estimating missing data sequences in X-ray microbeam recordings

Chao Qin and Miguel Á. Carreira-Perpiñán

EECS, School of Engineering, University of California, Merced, USA
{cqin,mcarreira-perpinan}@ucmerced.edu

Abstract

Techniques for recording the vocal tract shape during speech such as X-ray microbeam or EMA track the spatial location of pellets attached to several articulators. Limitations of the recording technology result in most utterances having sequences of frames where one or more pellets are missing. Rather than discarding such sequences, we seek to reconstruct them. We use an algorithm for recovering missing data based on learning a density model of the vocal tract shapes, and predicting missing articulator values using conditional distributions derived from this density. Our results with the Wisconsin X-ray microbeam database show we can recover long, heavily oscillatory trajectories with errors of 1 to 1.5 mm for all articulators.
Index Terms: articulatory databases, X-ray microbeam, missing data.

1. Introduction

Articulatory recording techniques such as X-rays, X-ray microbeam, electromagnetic articulography (EMA), ultrasound and magnetic resonance imaging (MRI) record a representation of part of the vocal tract during speech, and provide essential information for speech production research, speech therapy, and speech training, as well as for speech processing in the articulatory domain (e.g. for speech recognition, speech coding, speech synthesis and articulatory inversion) [1, 2, 3]. Publicly available databases such as the Wisconsin X-ray microbeam database [1] and the MOCHA database [2] have enabled much work in these areas. However, at present, obtaining good-quality recordings is expensive and difficult. The different recording technologies suffer from several problems: synchronisation with the acoustic wave, various types of interference, noise, risk to the subject, partial representation of the vocal tract, etc. The raw data recorded usually needs a heavy post-processing, some of it done manually at great cost. Since so much effort and resources are required to obtain this data, it is imperative to make the best possible use of every recording made even if it contains errors, rather than discard it and re-record it.

We focus here on one particular problem, mistracked or missing pellets, that affects techniques (such as X-ray microbeam or EMA) that are based on tracking over time the 2D or 3D positions of pellets attached to the tongue, lips and other articulators (see fig. 1). Mistracks happen for various reasons, from pellets unattaching to sensor malfunction [1, 4]. Some of the reasons depend on the recording technology. With X-ray microbeam [1], on which we focus in this paper, mistracks can happen because the microbeam looks for a pellet but is not able to find it, or because it follows the wrong pellet. This may be caused by the pellet accelerating too quickly; by shadowing from tissue, bone, teeth and fillings; or by pellets coming into close proximity. Mistracks can often be detected by the recording technology (e.g. when losing a pellet) or a posteriori (e.g. if following the wrong pellet, the values for two pellets will be nearly identical over time); but sometimes they are not detected at all. We will focus here on detected mistracks, and assume that the recording system provides a binary label indicating whether each component of the vector containing the coordinates of all pellets is present or not.

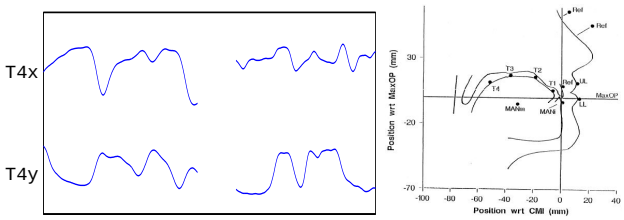

Fig. 1. Example of typical mistracks for one pellet in the Wisconsin XRMB database (pellet schematic at right). The mistrack duration is around 0.5 sec. The pellet can move drastically over this period, so one cannot simply interpolate it linearly.

With X-ray microbeam, mistracks of a given pellet occur more commonly in subsequences of 50 to 500 ms, often near the beginning of a record, after which the pellet is recaptured (a record is defined as a single continuous task interval, e.g. an utterance or an isolated word recording). Mistrack proportions in the XRMB database are small (around 1.9%), but the proportion of records containing mistracks is at least 35% [1, p. 65]. Since recording is expensive, cumbersome and (with X-ray microbeam) risky, this means that one cannot just discard records and re-record them again until they are perfectly tracked. Reconstructing the mistracks becomes a necessity. At present, the XRMB database indicates which frames are missing in each record, but provides no reconstruction.

The fundamental approach we follow is based on the following question: given that say only the tongue dorsum pellet is missing, can we reconstruct it from the location of the rest of the pellets? More generally, is there enough information in the present components of the data to predict the missing components? If the answer is yes—which it largely is in our problem, at least if not too much data is missing—then we can apply machine learning algorithms to estimate the missing data quite accurately. In addition, the method must handle time-varying missing data patterns in a transparent way. We describe such an algorithm in section 2, and report very successful experimental results with the XRMB database in section 3.

Related work. From previous work [5, 6, 7] we know that it is possible to reconstruct the entire midsagittal tongue contour with submillimetric accuracy from the positions of just 3–4 points on it. However, those papers assumed that some components were always present (the 3–4 pellets) and the rest always

missing, so the problem reduces to fitting a single mapping from the present to the missing components. This is unsatisfactory in our case because which components are present and which are missing varies from frame to frame; thus, the number of combinations of (present,missing) variables (such as missing $=$ {tongue tip, lower lip} and present $=$ {rest}) grows exponentially, and there is neither enough data nor enough computation available to fit all those mappings. Roweis [8] proposed to learn a low-dimensional manifold to represent the data and then intersect this with the constraints provided by the present values. This geometric approach is only efficient with (locally) linear manifolds.

2. Deriving mappings with varying sets of inputs and outputs from a density model

Our goal is to obtain a flexible, efficient way to construct mappings "on demand" between an arbitrary set of input and output variables. Our approach is based on [9, 10]. Call the articulatory variables $\mathbf{x} = (x_1, \ldots, x_D)$ (in our problem, $D = 16$ for the 2D positions of 8 pellets). At frame $\mathbf{x}_t$ in the utterance, let $\mathcal{P}_t$ and $\mathcal{M}_t$ be two sets of indices with $\mathcal{P}_t \cap \mathcal{M}_t = \varnothing$ and $\mathcal{P}_t \cup \mathcal{M}_t = \{1, \ldots, D\}$, indicating the present and missing variables at that frame, respectively. The idea is to encode all possible input-output relations in a master joint density $p(x_1, \ldots, x_D)$, and derive from it a mapping $\mathbf{x}_\mathcal{P} \rightarrow \mathbf{x}_\mathcal{M}$ as the mean of the conditional distribution $p(\mathbf{x}_\mathcal{M}|\mathbf{x}_\mathcal{P})$. The conditional distribution answers the question "what can we say about the values of $\mathbf{x}_\mathcal{M}$ given I know the values of $\mathbf{x}_\mathcal{P}$?".

For this to be useful, computing the conditional distributions must be done efficiently, yet they must be able to represent arbitrary nonlinear mappings. We can satisfy both needs by defining the joint density to be a Gaussian mixture with M components $p(\mathbf{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. The conditional distribution can be obtained as $p(\mathbf{x}_\mathcal{M}|\mathbf{x}_\mathcal{P}) = p(\mathbf{x})/p(\mathbf{x}_\mathcal{P})$ in terms of the joint and marginal distributions, all of which are Gaussian mixtures, and they equal (the indices assume we extract the corresponding block matrices, e.g. the marginalised variables are simply removed):

$$
\begin{aligned}
p(\mathbf{x}_\mathcal{P}) &= \textstyle\sum_{m=1}^{M} \pi_m \mathcal{N}(\mathbf{x}_\mathcal{P}; \boldsymbol{\mu}_{m,\mathcal{P}}, \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{P}}) \\
p(\mathbf{x}_\mathcal{M}|\mathbf{x}_\mathcal{P}) &= \textstyle\sum_{m=1}^{M} \pi_{m,\mathcal{M}|\mathcal{P}} \mathcal{N}(\mathbf{x}_\mathcal{M}; \boldsymbol{\mu}_{m,\mathcal{M}|\mathcal{P}}, \boldsymbol{\Sigma}_{m,\mathcal{M}|\mathcal{P}}) \\
\pi_{m,\mathcal{M}|\mathcal{P}} &= \pi_m \mathcal{N}(\mathbf{x}_\mathcal{P}; \boldsymbol{\mu}_{m,\mathcal{P}}, \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{P}})/p(\mathbf{x}_\mathcal{P}) \\
\boldsymbol{\mu}_{m,\mathcal{M}|\mathcal{P}} &= \boldsymbol{\mu}_{m,\mathcal{M}} + \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{M}}^T \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{P}}^{-1} (\mathbf{x}_\mathcal{P} - \boldsymbol{\mu}_{m,\mathcal{P}}) \\
\boldsymbol{\Sigma}_{m,\mathcal{M}|\mathcal{P}} &= \boldsymbol{\Sigma}_{m,\mathcal{M}\mathcal{M}} - \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{M}}^T \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{P}}^{-1} \boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{M}} \\
\mathbf{f}(\mathbf{x}_\mathcal{P}) &= \mathrm{E}\{\mathbf{x}_\mathcal{M}|\mathbf{x}_\mathcal{P}\} = \textstyle\sum_{m=1}^{M} \pi_{m,\mathcal{M}|\mathcal{P}}(\mathbf{x}_\mathcal{P})\, \boldsymbol{\mu}_{m,\mathcal{M}|\mathcal{P}}(\mathbf{x}_\mathcal{P})
\end{aligned}
$$

where the last equation gives the desired mapping. We also get errorbars for the prediction from the corresponding covariance. For diagonal components $\boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{M}} = \mathbf{0}$ and $\boldsymbol{\Sigma}_{m,\mathcal{P}\mathcal{P}}$ is diagonal, so the distribution of each component is obtained by simply crossing out rows and columns, without matrix inversions. However, a full-covariance mixture requires fewer components.

In summary, the method is as follows. The joint density model is learnt offline using a complete data set $\{\mathbf{x}_n\}$ using the EM algorithm (note that EM can also deal with incomplete datasets). At each frame $\mathbf{x}_t$ we determine which components $\mathcal{M}_t$ are missing, and reconstruct them as $\mathrm{E}\{\mathbf{x}_{\mathcal{M}_t}|\mathbf{x}_{\mathcal{P}_t}\}$. Note each frame in the utterance is reconstructed independently of the others, i.e., we apply no temporal smoothing.

If the conditional distributions are unimodal for each frame, using the conditional mean gives a good reconstruction for the missing values. If at some frame there were many missing values, the latter would be poorly constrained and their distribution would likely be multimodal, but this is not the case in our data.

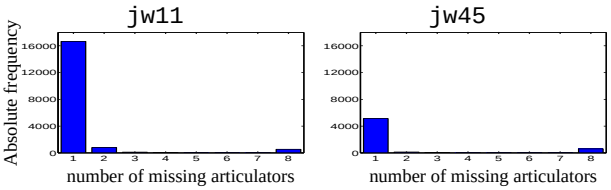

Fig. 2. Histogram of the number of missing articulators.

3. Experimental results

We use the Wisconsin X-ray microbeam database (XRMB [1]), which records, simultaneously with the acoustic wave, the positions of 8 pellets in the midsaggital plane of the vocal tract (fig. 1), sampled at 147 Hz. We use articulatory data from two speakers, `jw11` and `jw45`, with mistrack percentages of 11.32% and 3.55%, respectively. Mistracks occur most often on one articulator at a time and very rarely on multiple articulators (fig. 2). In this paper we focus on the reconstruction of single missing articulator, but our method is generally applicable to cases of multiple missing articulators.

We partition the data for each speaker into training and the testing sets. They contain $50\,000$ frames randomly sampled from 49 utterances and $10\,000+$ frames from 14 utterances, respectively. All utterances are normal speech. To estimate the joint density $p(\mathbf{x})$, we explore two types of models. (1) *Nonparametric Gaussian kernel density estimate (KDE)*. We try isotropic (KDEi) and full-covariance matrices (KDEF). In KDEi, the user supplies a bandwidth σ so each covariance is $\sigma^2\mathbf{I}$. In KDEF, we estimate a full covariance matrix for each mixture component m from its 100 nearest neighbours in the training set, and multiply this by a user bandwidth σ_F so each covariance is $\sigma_F^2 \boldsymbol{\Sigma}_m$. (2) *Parametric density estimate by Gaussian mixtures (GM)*. We try GM with $M = 32$, 64 and 128 components and each with a full-covariance matrix (this gave better results than using isotropic or diagonal covariances). Each GM is trained with EM from 10 random initialisations.

Reconstruction of artificially blacked-out data. Our goal here is to quantify the reconstruction accuracy with ground truth. Given an utterance with complete articulatory measurements, we black out two channels of one articulator over the entire utterance, and then infer their positions given the remaining 7 articulators, and compare with the ground truth. We repeat this for each articulator and for several utterances.

First, we study the choice of model parameters. Although many rules exist to set the bandwidth of a KDE in an unsupervised setting, here we can set it to minimise our reconstruction error. Fig. 3 plots the effect of the KDE bandwidth. For each blacked-out articulator, we compute the RMSE by averaging over a subset of the testing set for the given σ or σ_F. Each articulator favours a slightly different σ. On average, we found σ = 1.75 (`jw11`) and 1 (`jw45`), and σ_F = 1 (`jw11`, `jw45`) to be optimal, and use these values for the rest of the experiments. The reconstruction error also varies among different articulators. In general, it is easier to reconstruct the dorsum tongue pellets `T2,T3,T4` (around 1 mm RMSE) than `T1` (the tongue tip) and the lips ($1.5+$ mm RMSE). This agrees with the observation that the latter tend to be more variable and harder to predict (`T1`) or less coupled from other articulators (lips).

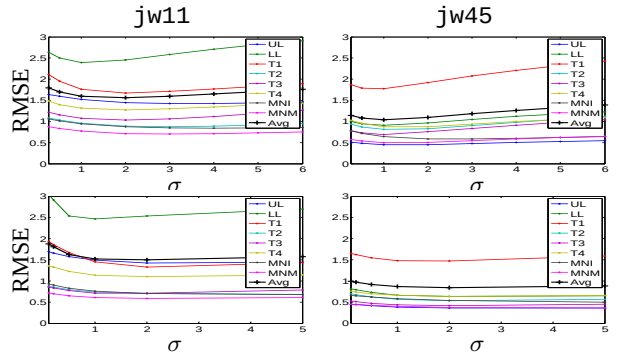

Fig. 3. Effect of the bandwidth σ on reconstructing missing articulators. *Top*: KDEi. *Bottom*: KDEF. The "Avg" curve is a weighted sum of the reconstruction error of each articulator, with weights inversely proportional to the respective error.

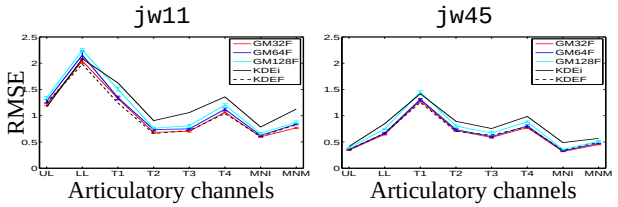

Fig. 4. Reconstruction error for each missing articulator. For the Gaussian mixtures, the (tiny) errorbars are over 10 random initialisations of EM.

However, the relative difficulty in the reconstruction does differ among the speakers, as does the absolute reconstruction error (the average RMSE differs by 0.5 mm among both speakers).

Next, we quantify the reconstruction error for individual missing articulators with each density model. Fig. 4 plots the averaged RMSE over the test set for individual missing articulators. Although by little, the GMs consistently outperform the KDEs by an average 0.2 mm. KDEF beats KDEi but requires considerably more computation. Among all GMs, the one with $M = 32$ components beats all others. All these results hold for both speakers and they are consistent with fig. 3. For example, they confirm that the lower lip for `jw11` and `T1` for `jw45` are the hardest to be reconstructed. On average, the reconstruction for all articulators is 1 to 1.5 mm for `jw11` (except for the lower lip, with a RMSE of 2.0 mm) and 0.5 to 1 mm for `jw45` (except for `T1`, with a RMSE of 1.4 mm). Thus, we conclude that a highly parsimonious Gaussian mixture (just 32 full-covariance components) can achieve a very accurate reconstruction. Recall that the measurement error in XRMB is around 0.5 mm.

Fig. 5 shows typical reconstructions of tongue pellets' trajectories for missing periods of 5 sec. The reconstructed trajectories are very close and correlated with the true ones even though the latter heavily oscillate over the missing period. This holds for all density models, although as mentioned the GM provides the best reconstruction (lowest reconstruction error and also smoothest reconstructions). KDEF is again better than KDEi and occasionally beats GM (e.g. the reconstruction of `T1` between 3.2 and 3.4 seconds). Even though we use no temporal information, discontinuous or jagged reconstructions happen only very rarely. `T1` is often more difficult to reconstruct than other tongue pellets since its motion is less coupled with them. On the other hand, it is very easy to reconstruct `T2` from `T3` or vice versa. The results are consistent among both speakers.

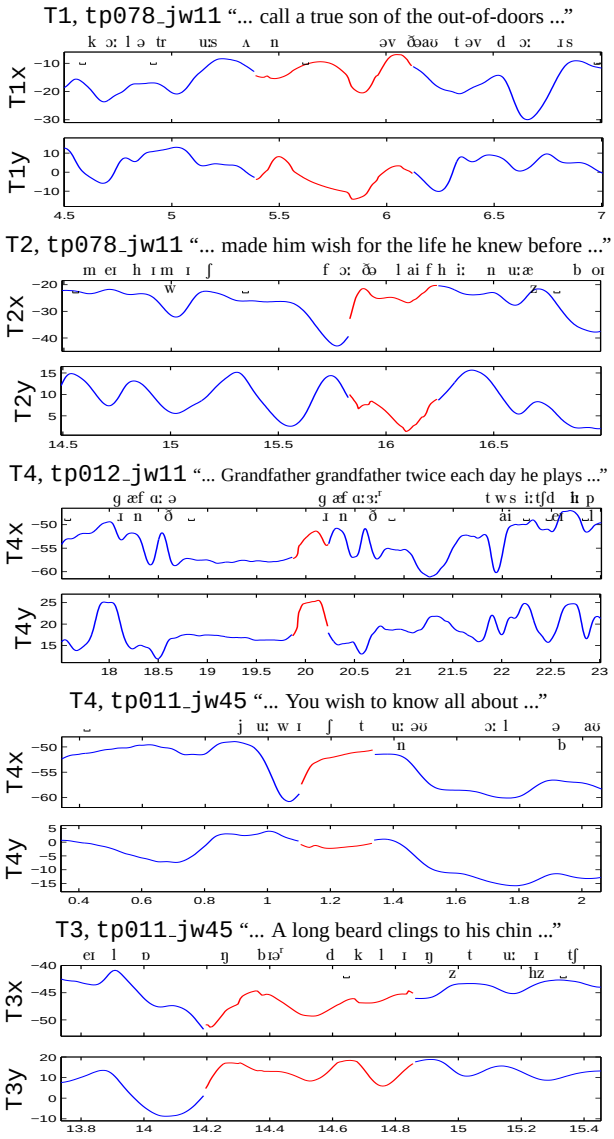

Fig. 6. Reconstruction of truly missing data with a GM (red); rest of the articulatory trajectory in blue, and phonetic labels.

Reconstruction of truly missing data. Fig. 6 shows the reconstructions of truly missing articulators (for which we have no ground truth) using the GM with $M = 32$ components; we show the phonetic labels (which are available from the synchronised speech) to help validate the reconstruction. Overall, the reconstructed articulatory trajectories are quite smooth, and the endpoints of the reconstructed data typically match very well with the present data, even though this was not enforced (each frame is reconstructed independently). Small discontinuities do occur, likely caused by the transition from one mixture component to another. This may be improved by a better density model, or by using temporal information. Visually, the trajectories look realistic, particularly if comparing with the corresponding phonetic label of the missing data, and if we compare the same phonetic context in a case where it is missing with a case where it is present. This happens in the reconstruction of mistracks of `T4` for `tp12_jw11`: note the context grandfather, especially for `T4y`.

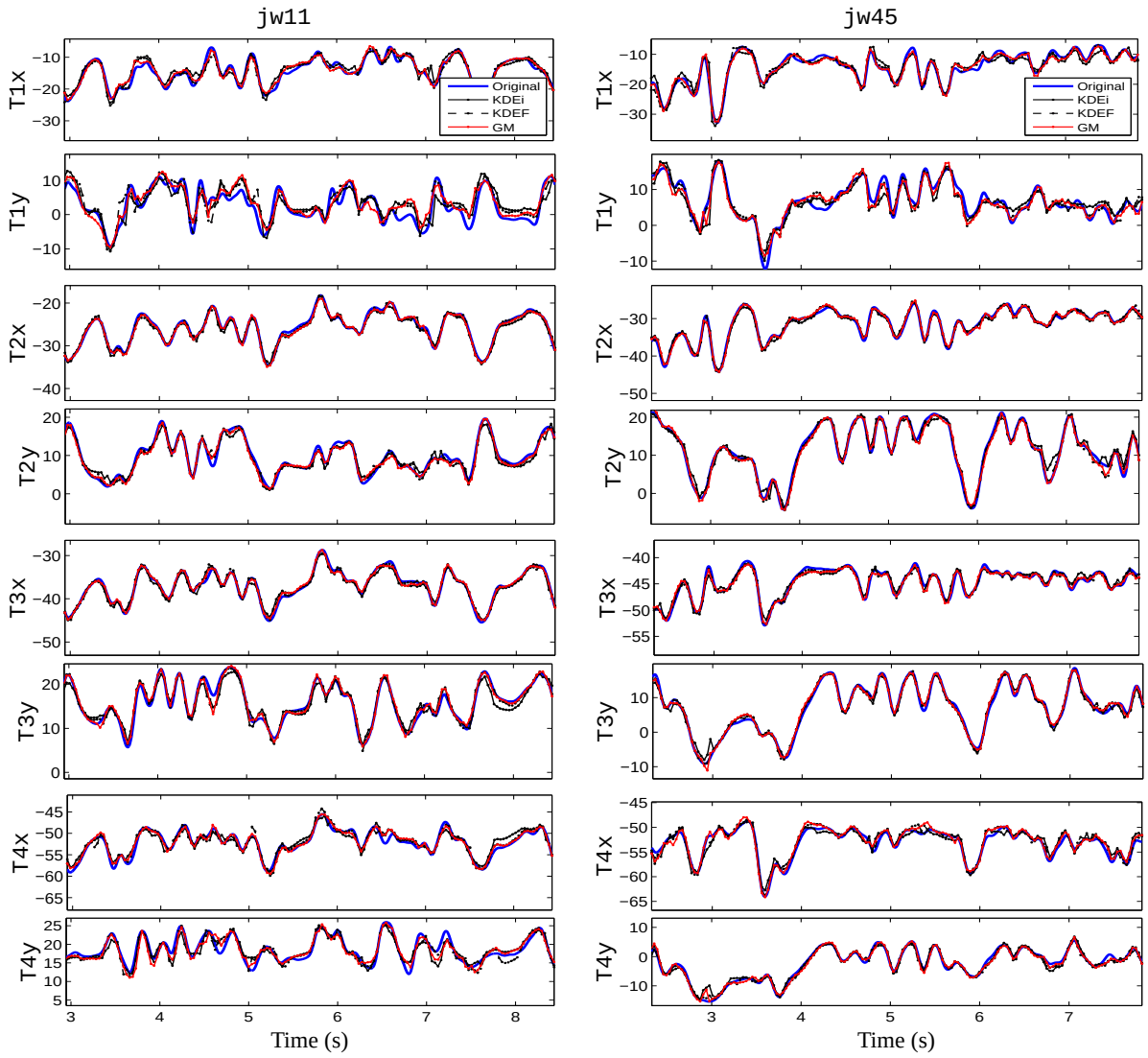

Fig. 5. Reconstruction of artificially blacked-out articulators `T1,T1,T3,T4` (top to bottom) for the utterance `tp011`.

4. Conclusion

We have extended an algorithm for missing data reconstruction and applied it to recovering missing pellet tracks in X-ray microbeam recordings, where the pellets are missing over extended periods, and the subset of missing pellets changes over time. A surprisingly parsimonious density model was sufficient to produce very accurate reconstructions for most pellets, even when the trajectory oscillates drastically over the period where it is missing. One limitation of the approach is that it relies on estimating a density model of the data ahead of time using a complete dataset (with no missing values). While this is not a problem with existing, large articulatory databases, future work should address reconstruction in more challenging situations, such as (near) real-time, or where little or no complete data are available for training.

Acknowledgments. Work funded by NSF CAREER award IIS–0754089.

5. References

[1] J. R. Westbury, *X-Ray Microbeam Speech Production Database User's Handbook V1.0*, University of Wisconsin, Jun. 1994.

[2] A. A. Wrench, "A multi-channel/multi-speaker articulatory database for continuous speech recognition research," in *Phonus 5*, Saarbrücken: Institute of Phonetics, 2000, pp. 1–13.

[3] E. Bresch, Y.-C. Kim, K. Nayak, D. Byrd, and S. Narayanan, "Seeing speech: Capturing vocal tract shaping using real-time magnetic resonance imaging," *IEEE Sig. Proc. Mag.*, May 2008.

[4] K. Richmond, "Estimating articulatory parameters from the acoustic speech signal," Ph.D. dissertation, U. Edinburgh, 2001.

[5] T. Kaburagi and M. Honda, "Determination of sagittal tongue shape from the positions of points on the tongue surface," *JASA*, vol. 96, no. 3, pp. 1356–1366, 1994.

[6] C. Qin, M. Á. Carreira-Perpiñán, K. Richmond, A. Wrench, and S. Renals, "Predicting tongue shapes from a few landmark locations," in *Proc. Interspeech*, 2008, pp. 2306–2309.

[7] C. Qin and M. Á. Carreira-Perpiñán, "Reconstructing the full tongue contour from EMA/X-Ray microbeam," in *ICASSP*, 2010.

[8] S. Roweis, "Data driven production models for speech processing," Ph.D. dissertation, California Institute of Technology, 1999.

[9] M. Á. Carreira-Perpiñán, "Reconstruction of sequential data with probabilistic models and continuity constraints," in *NIPS*, 2000.

[10] ——, "Continuous latent variable models for dimensionality reduction and sequential data reconstruction," Ph.D. dissertation, University of Sheffield, UK, 2001.