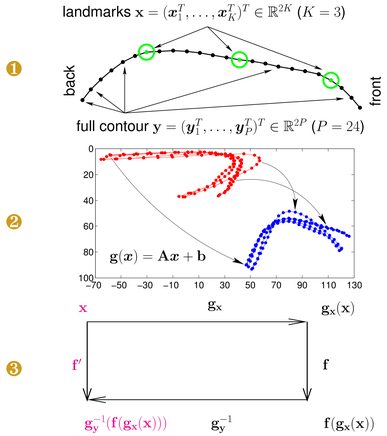
REGULARIZING AN ADAPTATION ALGORITHM FOR TONGUE SHAPE MODELS

Mohsen Farhadloo and Miguel Á. Carreira-Perpiñán.
EECS, School of Engineering, University of California, Merced.

1 Abstract

Realistic data-driven models of the tongue shape can be obtained by learning a nonlinear mapping from tongue landmarks to full contours, trained on a dataset of thousands of contours. Semiautomatic contour extraction from ultrasound takes a lot of time and effort from an expert, so practically it is preferable to adapt a reference model given just a few contours from the new speaker. However, adaptation with very few contours is unreliable and prone to overfitting. We study several forms of regularization to constrain the adaptation, and determine the optimal amount of regularization by leave-one-out cross-validation. Our results show that good accuracy models can be found reliably with no user intervention.

2 Introduction



❶ Learn a predictive mapping $\mathbf{f}$ in order to reconstruct the full tongue contour given only a few landmarks, from a dataset $\{(\mathbf{x}_n, \mathbf{y}_n)\}$ containing many contours.

❷ Given an adaptation dataset, estimate 2D-wise linear transformations $\mathbf{g}(\boldsymbol{x}) = \mathbf{A}\boldsymbol{x} + \mathbf{b}$ which are applied to each contour point separately (where $\mathbf{g}\colon \mathbb{R}^2 \to \mathbb{R}^2$).

❸ Obtain the adapted mapping $\mathbf{f}' = \mathbf{g_y}^{-1} \circ \mathbf{f} \circ \mathbf{g_x}$.

3 Predictive model of tongue shapes (Qin et al 2008)

The prediction problem: given the 2D locations of K landmarks $(\mathbf{x})$ on the tongue midsagittal contour, reconstruct the entire contour $(\mathbf{y})$, represented by P 2D points.

- Linear mapping: $\mathbf{f}(\mathbf{x}) = \mathbf{W}\mathbf{x} + \mathbf{w}$.
- Radial basis function (RBF) network: $\mathbf{f}(\mathbf{x}) = \mathbf{W}\Phi(\mathbf{x}) + \mathbf{w}$ with M Gaussian basis functions $\phi_m(\mathbf{x}) = \exp\left(-\frac{1}{2}\|(\mathbf{x} - \boldsymbol{\mu}_m)/\sigma\|^2\right)$ and width σ.

Both mappings are estimated from contours obtained from ultrasound images. This beats spline interpolation of the landmarks.

4 Adaptation with global feature transformations (Qin & Carreira-Perpiñán 2009)

Idea: apply the same invertible transformation to each $2D$ point.
Given the predictive mapping $\mathbf{f}$ and a small number N of adaptation contours $\{\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N$, we estimate an invertible linear mapping $\mathbf{g}$ that transforms new data (from the new speaker) to old data (from the old speaker):

$$\tilde{\mathbf{x}} = \mathbf{g_x}(\mathbf{x}) = \begin{pmatrix} \mathbf{A}\boldsymbol{x}_1 + \mathbf{b} \\ \dots \\ \mathbf{A}\boldsymbol{x}_K + \mathbf{b} \end{pmatrix} \qquad \tilde{\mathbf{y}} = \mathbf{g_y}(\mathbf{y}) = \begin{pmatrix} \mathbf{A}\boldsymbol{y}_1 + \mathbf{b} \\ \dots \\ \mathbf{A}\boldsymbol{y}_P + \mathbf{b} \end{pmatrix}.$$

Our optimal solution $\{\mathbf{A}, \mathbf{b}\}$ is the minimizer of the proxy error function

$$F(\mathbf{A}, \mathbf{b}) = \sum_{n=1}^N \|\mathbf{g_y}(\mathbf{y}_n) - \mathbf{f}(\mathbf{g_x}(\mathbf{x}_n))\|^2.$$

Only 6 parameters $\mathbf{A}_{2\times 2}$ and $\mathbf{b}_{2\times 1}$ to be estimated $\Rightarrow$ little data needed (small N) to achieve reasonable accuracy. But:

- The accuracy stagnates as N grows (error 0.7 mm more than the ground truth).
- It doesn't allow for different transformations in different points of the tongue.
- Minimizing $F(\mathbf{A}, \mathbf{b})$ rather than $\mathbf{E}(\mathbf{A}, \mathbf{b})$ introduces a small bias in the transformation.

5 Adaptation with local feature transformations (Qin et al 2010)

Idea: apply different invertible transformations to each $2D$ point.
We estimate invertible linear mappings $\mathbf{g_x}(\mathbf{x})$ and $\mathbf{g_y}(\mathbf{y})$:

$$\tilde{\mathbf{x}} = \mathbf{g_x}(\mathbf{x}) = \begin{pmatrix} \mathbf{A}_1^{\mathbf{x}}\boldsymbol{x}_1 + \mathbf{b}_1^{\mathbf{x}} \\ \dots \\ \mathbf{A}_K^{\mathbf{x}}\boldsymbol{x}_K + \mathbf{b}_K^{\mathbf{x}} \end{pmatrix} \qquad \tilde{\mathbf{y}} = \mathbf{g_y}(\mathbf{y}) = \begin{pmatrix} \mathbf{A}_1^{\mathbf{y}}\boldsymbol{y}_1 + \mathbf{b}_1^{\mathbf{y}} \\ \dots \\ \mathbf{A}_P^{\mathbf{y}}\boldsymbol{y}_P + \mathbf{b}_P^{\mathbf{y}} \end{pmatrix}.$$

It is easier to work with the parameters $\mathbf{C}_i^{\mathbf{y}} = (\mathbf{A}_i^{\mathbf{y}})^{-1}$, $\mathbf{d}_i^{\mathbf{y}} = -(\mathbf{A}_i^{\mathbf{y}})^{-1}\mathbf{b}_i^{\mathbf{y}}$ of $\mathbf{g_y}^{-1}$. The optimal solution $\{\mathbf{A}_1^{\mathbf{x}}, \mathbf{b}_1^{\mathbf{x}}, \dots, \mathbf{A}_K^{\mathbf{x}}, \mathbf{b}_K^{\mathbf{x}}, \mathbf{C}_1^{\mathbf{y}}, \mathbf{d}_1^{\mathbf{y}}, \dots, \mathbf{C}_P^{\mathbf{y}}, \mathbf{d}_P^{\mathbf{y}}\}$ is the minimizer of the the predictive square error function:

$$E(\mathbf{A}^{\mathbf{x}}, \mathbf{b}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}}, \mathbf{d}^{\mathbf{y}}) = \sum_{n=1}^N \left\|\mathbf{y}_n - \mathbf{g_y}^{-1}(\mathbf{f}(\mathbf{g_x}(\mathbf{x}_n)))\right\|^2.$$

There are $6(K + P)$ parameters $\Rightarrow$ more accurate adaptation possible (error only 0.1 to 0.3 mm more than the ground truth). Besides:

- There is no estimation bias: E is the ideal error to minimize, not F.
- Minimizing E is simpler now because the parameters are decoupled.
- Needs more adaptation data.
- More prone to overfitting and unreliable with less than 10 contours.

Objectives of this work:

❶ Regularize $E(\mathbf{A}^{\mathbf{x}}, \mathbf{b}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}}, \mathbf{d}^{\mathbf{y}}) + \lambda R(\mathbf{A}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}})$ to establish a spectrum between the global approach with few contours and the purely local one with many contours.

❷ Determine automatically an optimal point in the spectrum given the adaptation dataset.

ICASSP 2012 IEEE
IEEE International Conference on Acoustics, Speech, and Signal Processing
March 25-30, 2012 • Kyoto International Conference Center • Kyoto, Japan

6 Regularizers

We study three different regularizers in this paper, one that penalizes high condition number transformations, and two that encourage all transformation matrices to be identical.

Condition number regularizer (Cond#)

- It assumes the ideal transformation is not too far from simple translation and rotation ($Cond(\mathbf{A}) = 1$).
- Choosing a regularizer that penalizes the high-condition-number transformations.

$$\lambda\mathcal{C}(\mathbf{A}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}}) = \lambda_1\left(\textstyle\sum_{i=1}^K C(\mathbf{A}_i^{\mathbf{x}}) + \sum_{i=1}^P C(\mathbf{C}_i^{\mathbf{y}})\right)$$
$$C(\mathbf{A}) = \operatorname{tr}(\mathbf{A}^T\mathbf{A}) - D\det(\mathbf{A}^T\mathbf{A})^{1/D} \quad \text{for } \mathbf{A}_{D\times D},$$

Chain regularizer (Chain)

- In local approach in few-contour region, regularizer causes reduction in the value of point-wise RMSE and its variance, but in large-contour region, it causes increase.
- In few-contour region the global method performs better than its local counterpart.
- Choosing a regularizer that penalizes the non-similar transformations.

$$R_2(\mathbf{A}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}}) = \lambda_2(R_2(\mathbf{A}^{\mathbf{x}}) + R_2(\mathbf{C}^{\mathbf{y}}))$$
$$R_2(\mathbf{A}^{\mathbf{x}}) = \|\mathbf{A}_2^{\mathbf{x}} - \mathbf{A}_1^{\mathbf{x}}\|^2 + \|\mathbf{A}_3^{\mathbf{x}} - \mathbf{A}_2^{\mathbf{x}}\|^2 + \dots + \left\|\mathbf{A}_K^{\mathbf{x}} - \mathbf{A}_{K-1}^{\mathbf{x}}\right\|^2,$$

Variance regularizer (Var)

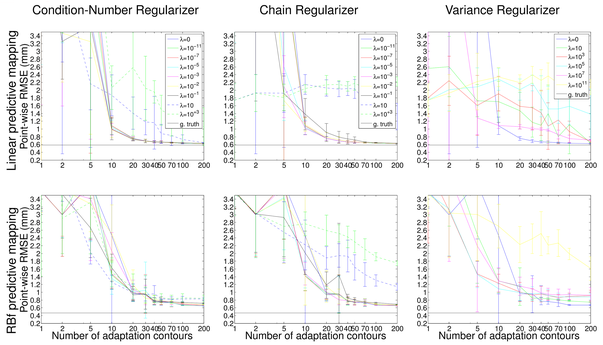
- Another regularizer that penalizes the non-similar transformations.

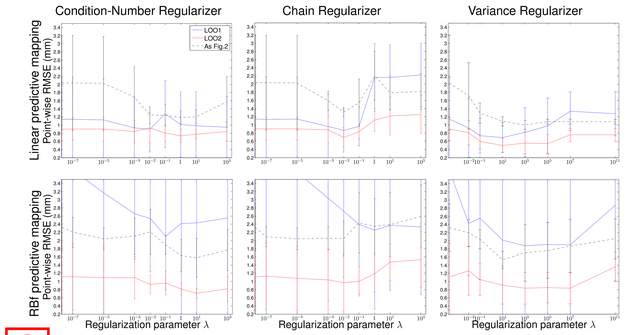
$$R_3(\mathbf{A}^{\mathbf{x}}, \mathbf{C}^{\mathbf{y}}) = \lambda_3(R_3(\mathbf{A}^{\mathbf{x}}) + R_3(\mathbf{C}^{\mathbf{y}}))$$
$$R_3(\mathbf{A}^{\mathbf{x}}) = \operatorname{tr}(\operatorname{cov}(\operatorname{vec}(\mathbf{A}_1^{\mathbf{x}}), \dots, \operatorname{vec}(\mathbf{A}_K^{\mathbf{x}})))$$

Leave-One-Out cross validation

In a real situation one may only have access to a few contours that should be used for adaptation and also for finding the best values of the related parameters. A practical method in this situation is LOO in which each adaptation contour is set aside for validating the learned model using the $N - 1$ remaining contours and then the results would be averaged over these N experiments. In our experiments the tuning parameter λ is one of the λ_1, λ_2 or λ_3 that we introduced. We find the value for λ by looking for it in a particular range of values.

7 Experimental results





- With Chain and Var regularizers it is possible to find a value for λ which gives small error both in few and largecontour regions.
- We can modulate between purely local ($\lambda = 0$) and purely global ($\lambda \to \infty$) feature transformation methods for Chain and Var but not the Cond#.
- Both Chain and Var seem to switch continuously from the local case (very low error) to the global one (stagnation error).
- By comparing lower envelopes, for linear adaptation Chain and Var are better than the Cond# and for RBF adaptation all regularizers are comparable (although Var is a bit better than others).
- LOO cross-validation can be used to estimate the optimal amount of regularization for the adaptation set provided.

8 Conclusion

We have studied several forms of regularization for adapting a tongue shape model using local feature transformations. While all our regularizers helped to estimate a good accuracy model with very few adaptation contours, we found it most effective to penalize the transformation matrices' variance. This smoothly blends a global adaptation using a single transformation into a local adaptation using a different transformation at each contour point. A near-optimal amount of regularization for a specific adaptation dataset can be obtained automatically with leave-one-out cross-validation. We have also shown that centering and rescaling the data accelerates the optimization.

Acknowledgements: Work funded by NSF award IIS–0711186.