MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING
DEPARTMENT OF BRAIN AND COGNITIVE SCIENCES

A.I. Memo No. 1696 September, 2000
C.B.C.L Paper No. 191

Learning-Based Approach to Estimation of Morphable Model Parameters

Vinay P. Kumar and Tomaso Poggio

This paper describes a method for estimating the parameters of a linear morphable model (LMM) that models mouth images. The method uses a learning-based approach to estimate the LMM parameters directly from the images of the object class (in this case mouths). Thus this method can be used to bypass current computationally intensive methods that use analysis by synthesis, for matching objects to morphable models. We have used the invariance properties of Haar wavelets for representing mouth images. We apply the robust technique of Support Vector Machines (SVM) for learning a regression function from a sparse subset of Haar coefficients to the LMM parameters. The estimation of LMM parameters could possibly have application to other problems in vision. We investigate one such application, namely viseme recognition.

This report describes research done within the Center for Biological and Computational Learning in the Department of Brain and Cognitive Sciences and at the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. This research is sponsored by a grant from the National Science Foundation under contract No. IIS-9800032. Additional support is provided by Eastman Kodak Company, Daimler-Benz, Siemens Corporate Research, Inc. and AT&T.

1 Introduction-Motivation

Amongst the many model-based approaches to modeling object classes, the Linear Morphable Model is an important one (Vetter and Poggio [10], Jones and Poggio [4]). It has been been used successfully to model faces, cars and digits. In these applications, the task of matching a novel image to the LMM is achieved through a computationally intensive analysis by synthesis approach. In Jones and Poggio [4], the matching parameters are computed by minimizing the squared error between the novel image and the model image using a stochastic gradient descent algorithm. This technique may take several minutes for matching even a single image. A technique that could compute the matching parameters with considerably less computations and using only view-based representations would make these models useful in real-time applications. The motivation for this work comes from the use of a learning-based approach in real-time analysis of mouths (Kumar and Poggio [7]), in which it was shown that a regression function can be learnt from a Haar wavelet based input representation of mouths to hand labeled parameters denoting openness and smile. Therefore, it points to the possibility that learning may be a way for directly estimating the matching parameters of an LMM from the image.

Previously, morphable models of mouths have been constructed for the purpose of synthesis of visual speech (Ezzat and Poggio [5]). We will explore the morphable model as a tool in the analysis of mouth images. There has also been an attempt in the work by Cootes et al. [2] to speed up the process of analysis by synthesis for computing the matching parameters of morphable models (which they call active appearance model). The speed-up is achieved by learning several multivariate linear regressions from the error image (difference between the novel and the model images) and the appropriate perturbation of the model parameters (the known displacements of the model parameters), thus avoiding the computation of gradients. This method is akin to learning the tangent plane to the manifold in pixel space formed by the morphable model.

In this work, we propose to construct an LMM to model various mouth shapes and expressions. Following Jones and Poggio [4] the LMM is constructed from examples of mouths. However, we reduce the parameter set by performing PCA on the example textures and flows. We then use Support Vector Machine (SVM) regression (Vapnik [9]) to learn a non-linear regression function from a sparse subset of Haar wavelet coefficients to the matching parameters of this LMM directly. The training set (the outputs) for this learning problem is generated by estimating the true matching parameters using the stochastic gradient descent algorithm described in Jones and Poggio [4].

An obvious application of estimating LMM parameters is in image synthesis (or graphics). However, recently it has been suggested that LMM parameters could also be used for higher level image analysis (or vision) such as face identification (Blanz [6] and Edwards et al. [3]). In this paper, since we are working with mouth shapes, we explore a different application, namely, viseme recognition. Visemes are the visual analogues of phonemes (Ezzat and Poggio [5]). Recognizing visemes have potential applications in enhancing the performance of speech recognition systems or driving photorealistic avatars. Our approach is based on classifying single images into viseme classes by training classifiers on the matching LMM parameters. This approach can be compared with one where we use the Haar coefficients directly instead of going through the LMM as an intermediate representation. Our experiments, attempted on six viseme classes, raise some important questions about the efficacy of LMM-based representations vis-a-vis pixel-based representations in higher level vision tasks but need further experimentation before final conclusions

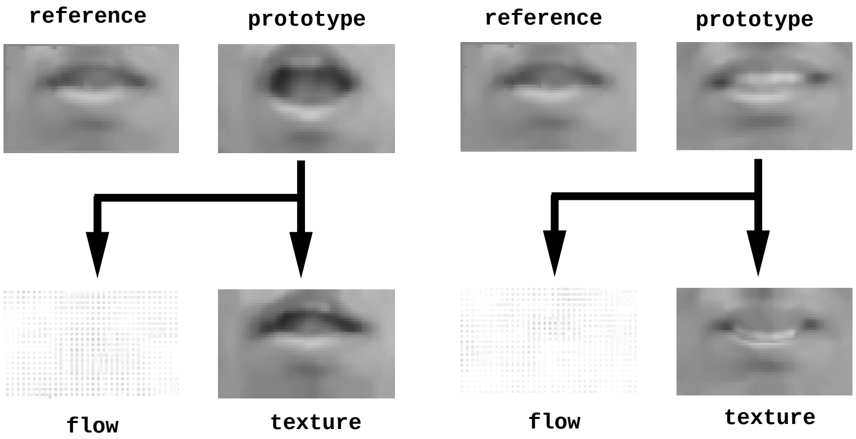

Figure 1: *Illustrating the vectorized representation of an image. Pixelwise correspondences are computed between the prototype image and a standard reference image. The flow vector consists of the displacements of each pixel in the reference image relative to its position in the prototype. The texture vector consists of the prototype* backward warped *to the reference image.*

can be drawn.

2 Linear morphable model for modeling mouths

In this section we provide a brief overview of LMMs and their application to modeling mouths.

2.1 Overview of LMMs

A linear morphable model is based on linear combinations of a specific representation of example images. The representation involves establishing a correspondence between each example image and a reference image. Thus it associates with every image a shape vector and a texture vector. Figure 1 illustrates this vectorized representation, which can be computed by the linear combination example images as shown in Figure 2 (See Jones and Poggio [4] for more details).

2.2 Constructing an LMM for modeling mouths

We collected 2066 images of mouths from one person. 93 of these images were manually chosen as example images to construct the LMM. The reference image can be chosen such that it corresponds to the average (in the vector space defined by the LMM) of the example images. However, the LMM can be defined only by choosing a reference. Therefore we take recourse to an iterative method where the reference image is initially chosen arbitrarily. Using this reference and the LMM that it defines, the average of the example images is computed. This average image then forms the reference image for the next step of the iteration. This method converges in a few iterations to a stable average image.

Once the reference image is found, we get a 93 dimensional LMM. The dimensionality of pixel space being 2688, the LMM constitutes a lower dimensional representation of the space (or manifold) of mouth images. However since many of the example images are alike there is likely to be a great deal of redundancy even in this representation. In order to remove this redundancy,

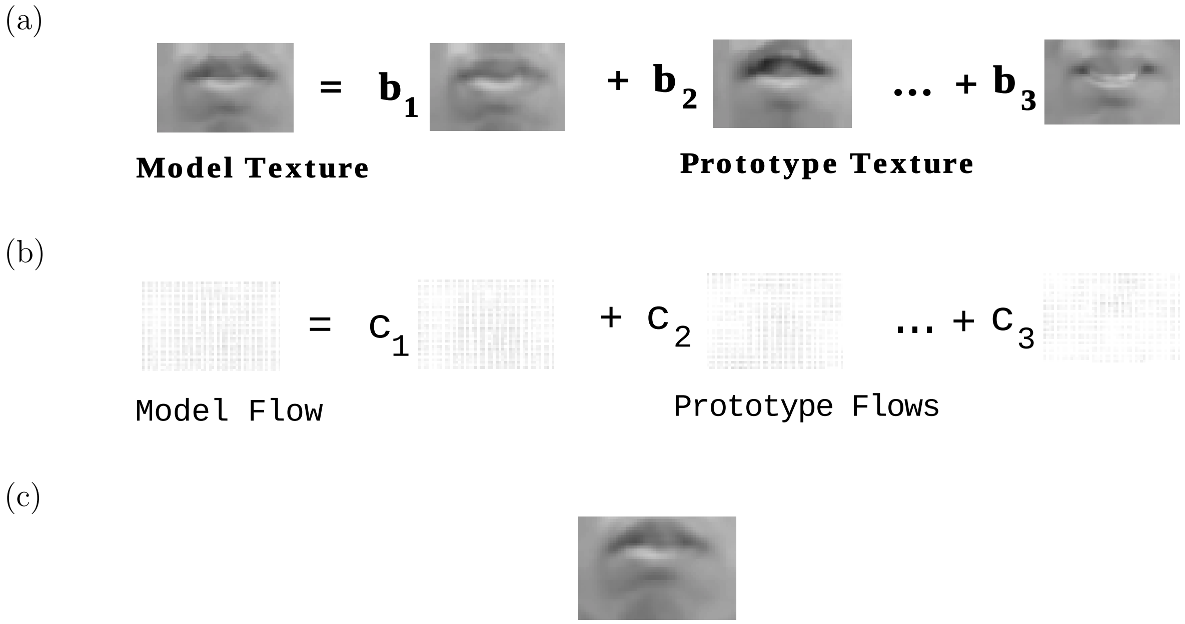

Figure 2: *A linear combination of images in the LMM framework involves (a) linearly combining the prototype textures using the coefficients* **b** *to yield a model texture and (b) the prototype flows using the coefficients* **c** *to yield a model flow. (c) The model image is obtained by warping the model texture along the model flow*

we perform PCA on the example texture and shape vectors and retain only those principal components with the highest eigenvalues. As a result we obtain an LMM where a novel texture is a linear combination of the principal component textures (which do not resemble any of the example textures) and similarly a novel flow is a linear combination of the principal component flows.

3 Learning to estimate the LMM parameters directly from images

The problem of estimating the matching LMM parameters directly from the image is modeled as learning a regression function from an appropriate input representation of the image to the set of LMM parameters. The input representation is chosen to be a sparse set of Haar wavelet coefficient while we use support vector regression as the learning algorithm.

3.1 Generating the Training Set

The training set was generated as follows.

- Each of the 2066 images is matched to the LMM that retains the top three principal component textures and flows respectively, and using the stochastic gradient descent algorithm from Jones and Poggio [4]. Thus each image is represented as a six dimensional vector, which form the outputs for the learning problem.

- Each of the 2066 images is subject to the Haar wavelet transform and feature selection involving selection of those Haar coefficients with the highest variance (See Kumar and Poggio [7]). We select 12 coefficients with the highest variance which form the inputs for the learning problem.

3.2 Training the SVM-based regression

In this section, we sketch the ideas behind using SVM for learning regression functions (a more detailed description can be found in Golowich, et al. [8] and Vapnik [9]). Let $G = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, be the training set obtained by sampling, with noise, some unknown function $g(\mathbf{x})$. We are asked to determine a function f that approximates $g(\mathbf{x})$, based on the knowledge of G. The SVM considers approximating functions of the form:

$$f(\mathbf{x}, \mathbf{c}) = \sum_{i=1}^{D} c_i \phi_i(\mathbf{x}) + b \tag{1}$$

where the functions $\{\phi_i(\mathbf{x})\}_{i=1}^{D}$ are called *features*, and b and $\{c_i\}_{i=1}^{D}$ are coefficients that have to be estimated from the data. This form of approximation can be considered as an hyperplane in the D-dimensional feature space defined by the functions $\phi_i(\mathbf{x})$. The dimensionality of the feature space is not necessarily finite. The SVM distinguishes itself by minimizing the following functional to estimate its parameters.

$$R(\mathbf{c}) = \frac{1}{N} \sum_{i=1}^{N} \mid y_i - f(\mathbf{x}_i, \mathbf{c}) \mid_{\epsilon} + \lambda \|\mathbf{c}\|^2 \tag{2}$$

where λ is a constant and the following *robust* error function has been defined

$$\mid y_i - f(\mathbf{x}_i, \mathbf{c}) \mid_{\epsilon} = \max(\mid y_i - f(\mathbf{x}_i, \mathbf{c}) \mid - \epsilon, 0) \tag{3}$$

Vapnik showed in [9] that the function that minimizes the functional in equation (2) depends on a finite number of parameters, and has the following form:

$$f(\mathbf{x}, \alpha, \alpha^*) = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(\mathbf{x}, \mathbf{x}_i) + b, \tag{4}$$

where $\alpha_i^* \alpha_i = 0, \;\; \alpha_i, \alpha_i^* \geq 0 \;\; i = 1, \ldots, N$, and $K(\mathbf{x}, \mathbf{y})$ is the so called *kernel* function, and describes the inner product in the D-dimensional feature space

$$K(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{D} \phi_i(\mathbf{x}) \phi_i(\mathbf{y})$$

Only a small subset of the $(\alpha_i^* - \alpha_i)$'s are different from zero, leading to a small number of support vectors. In our case, we obtained the best results for the case when the Kernel was a Gaussian, i.e. $K(\mathbf{x}, \mathbf{y}) = e^{\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{\sigma^2}}$, where σ is a variance which acts as a normalization factor. The main advantage accrued by using a SVM is that since it uses the robust error function given by equation (3), we obtain an estimate which is less sensitive to outliers.

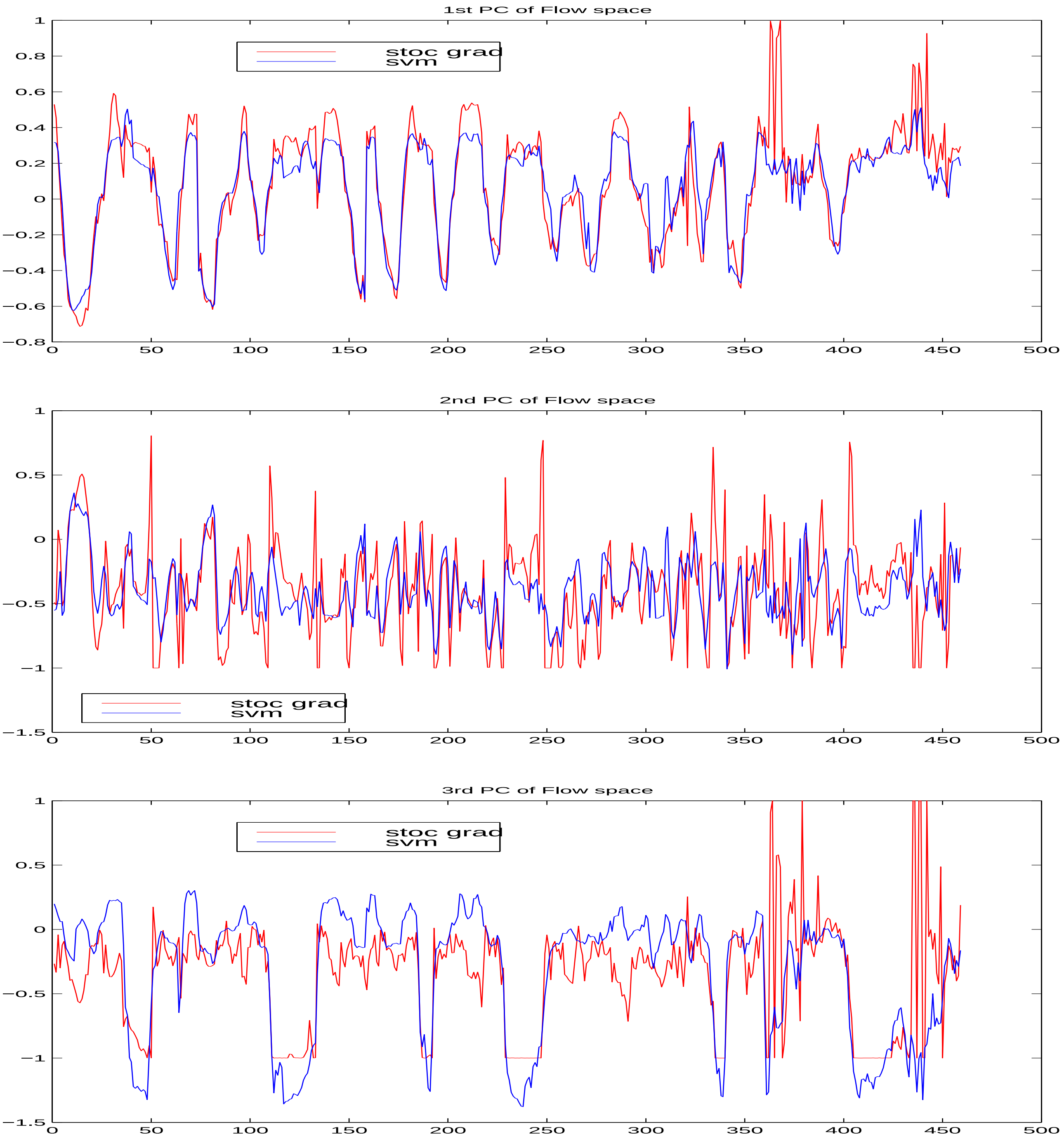

Figure 3: *Estimates of LMM flow parameters using the stochastic gradient descent and support vector regression on a test sequence of 459 images.*

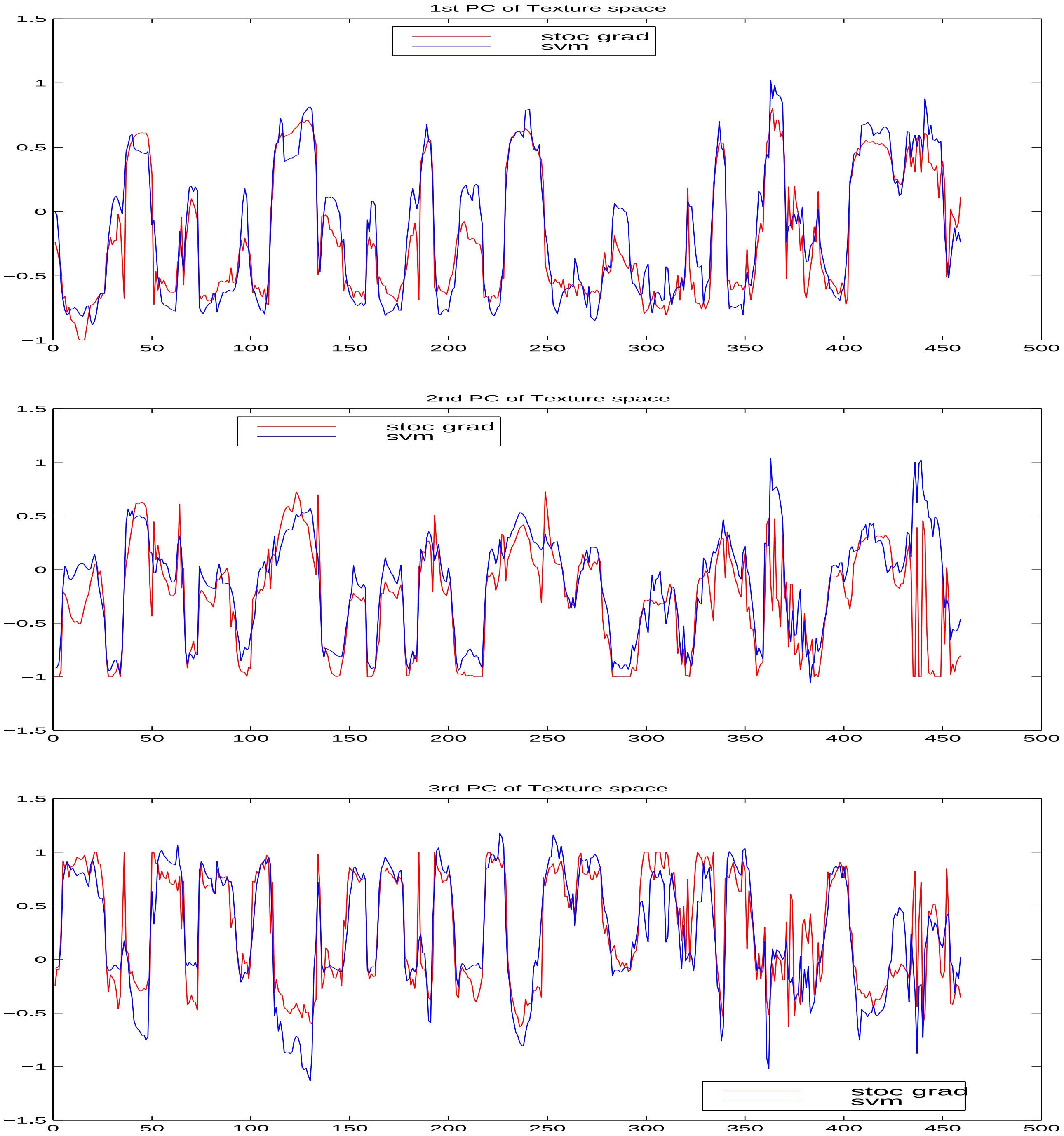

Figure 4: *Estimates of LMM texture parameters using the stochastic gradient descent and support vector regression on a test sequence of 459 images.*

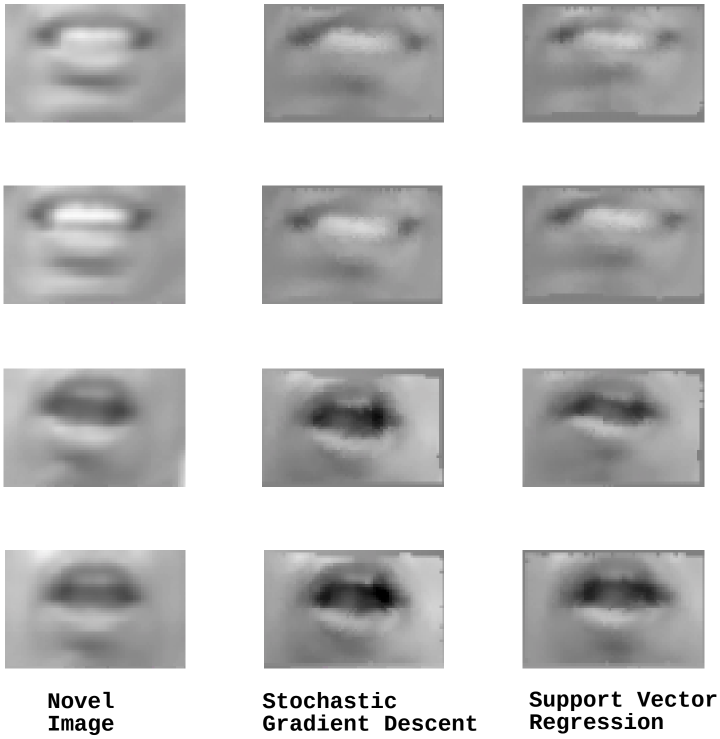

Figure 5: *Example matches of novel mouths using stochastic gradient descent and support vector regression from a test sequence*

4 Results

In our experiments, we attempted to estimate six LMM coefficients corresponding to the top three principal components of the texture space and the flow space respectively. For each LMM coefficient a separate regression function had to be learnt.

Preliminary experiments indicated that Gaussian kernels were distinctly superior to estimating the LMM parameters in terms of number of support vectors and training error compared to polynomial kernels. As a result we next confined ourselves to experimenting with Gaussian kernels. The free parameters in this problem, namely, the insensitivity factor ϵ, a weight on the cost for breaking the error bound C and the normalization factor σ were estimated independently for each of the LMM parameters using cross-validation. The regression function was used to estimate the LMM parameters of a test sequence of 459 images. Figures 3 and 4 display the results for flow and texture parameters respectively where the performance of support vector regression is compared to that obtained using stochastic gradient descent. Examples of matches using the two methods is shown in Figure 5.

5 Viseme Recognition

In this section, we ask the question: of what use is the direct estimation of LMM parameters? As noted earlier, in addition to its obvious application to image synthesis (or graphics), recently Blanz [6] and Edwards et al. [3] have worked on using matching LMM parameters for higher level image analysis (or vision) applications such as face identification and shown encouraging results. This prompts us to ask if these parameters might find use for other crucial vision tasks. One such application is viseme recognition.

Viseme class	Associated phonemes	Training examples	Testing examples
pbm	p, b, m	76	30
tdsz	t, d, s, z, th, dh	113	57
aao	aa, o	47	23
ii	ii, i	38	27
ea	e, a	40	14
kgnl	k, g, n, l, ng, y	87	37

Table 1: The viseme classes and their associated phonemes and the size of the training and testing sets.

	Linear SVM	k nearest neighbors			
	Top-Down	k = 1	k = 2	k = 3	k = 4
LMM Representation	0.69	0.63	0.63	0.64	0.66
Wavelet Representation	0.72	0.65	0.66	0.67	0.68

Table 2: Overall accuracy of viseme recognition.

Visemes are the visual analogues of phonemes (Ezzat and Poggio [5]). However, the mapping from phonemes to visemes is a many to one mapping. Different phonemes can lead to a single mouth shape and thus to a single viseme. Visemes like phonemes have temporal extent. However, in this work we investigate viseme recognition assuming visemes to be static images.

We used the visual speech corpus described in Ezzat and Poggio [5] for the viseme recognition problem. In this corpus, 39 phoneme classes which maps to 15 viseme classes have been identified. However, there was sufficient data to train for only six visemes classes (3 consonant classes and 3 vowel classes). Those six classes are 'pbm', 'tdsz', 'kgnl', 'ii', 'ea', 'aao'. Details about these classes and the training and testing sets are given in Table 1.

Two different representations were investigated as input for classification, namely, wavelets and LMM parameters. In the former case, coarse Haar wavelet coefficients were selected using the method described in section 3.1. In the latter case, 91 images from this corpus were used to construct an LMM and the model was matched to the remaining images. The top 3 coefficients corresponding to the principal component texture and flow vectors, were used as a feature set to represent each image. The classifiers needed for viseme recognition were linear in the space of the inputs and trained using the SVM method (Vapnik [9]).

We have experimented with the top-down decision graph (Nakajima, et al. [1]) as a multi-class strategy for viseme recognition. This strategy involves the training of a classifier to distinguish between any two visemes, each of which is a linear SVM. We have compared the performance of this technique with the k-nearest neighbors technique. The results comparing different representations and different multi-class strategies are presented in Tables 2 and 3.

6 Conclusion - Future Work

The results of directly estimating the matching parameters of an LMM are encouraging. In some cases where the LMM is not tolerant to translation and scale changes, the method of direct estimation proves to be more robust and can give us a recognizable matching image. It also raises several questions and opens new possibilities. The questions pertain to the generality of the method. So far, we have worked with LMMs designed to model the mouth shapes and expressions of only one person. Can this method be extended to multiple persons? Where does the potential problem with this extension lie - at the stage of building the LMM or the learning of the regression functions? This work has potential applications in several areas, namely, real-time recognition of expressions/visual speech, a new method for temporal representations of expressions/visual speech.

The one application considered here, namely, viseme recognition is in its preliminary stages but the results open several interesting questions. At the outset it seems that using a wavelet-based representation gives a slightly better performance than a LMM-based representation. This goes contrary to the encouraging results shown by Blanz [6] and Edwards et al. [3] and therefore raises questions about the applicability of LMM-based representations to higher level image analysis. Are LMM-based representations suited for only certain vision tasks (such as face identification) and not for others (such as viseme recognition)? If so, what could be the reasons for these differences?

A closer look at the results, however, shows that the better performance of wavelet-based representation is not quite uniform. While the performance improves considerably for some viseme classes, it also deteriorates for others. Thus, it is clear that more experiments are needed before any conclusions can be drawn about this phenomenon. One path that might provide some answers is using this method for representing visemes as time sequences, and their application to improving the performance of speech recognition systems.

7 Acknowledgment

We would like to thank Thomas Vetter and Mike Jones for helpful discussions and providing C code for stochastic gradient descent. We would also like to thank the members of CBCL for discussions and help at various stages of this research. We also request the scientific community to not use this research for military or any other unethical purposes.

References

[1] C. Nakajima, M. Pontil, B. Heisele and T. Poggio. People Recognition in Image Sequences by Supervised Learning. *MIT AI Memo No. 1688/CBCL Memo No. 188*, 2000.

[2] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proceedings of the European Conference on Computer Vision*, Freiburg, Germany, 1998.

[3] C.J. Taylor G.J. Edwards and T.F. Cootes. Learning to identify and track faces in image sequences. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, pages 260–265, 1998.

[4] M. Jones and T. Poggio. Multidimensional morphable models: A framework for representing and matching object classes. In *Proceedings of the International Conference on Computer Vision*, Bombay, India, 1998.

[5] T. Ezzat and T. Poggio. Visual Speech Synthesis by Morphing Visemes. *MIT AI Memo No. 1658/CBCL Memo No. 173*, 1999.

[6] V. Blanz. *Automated Reconstruction of Three-Dimensional Shape of Faces from a Single Image.* Ph.D. Thesis (in German), University of Tuebingen, 2000.

[7] V. Kumar and T. Poggio. Learning-Based Approach to Real Time Tracking and Analysis of Faces. In *Proceedings of the Fourth International Conference on Automatic Face and Gesture Recognition*, pages 96–101, Grenoble, France, 2000.

[8] S.E. Golowich V. Vapnik and A. Smola. Support vector method for function approximation, regression estimation, and signal processing. In *Advances in Neural Information Processing Systems*, volume 9, pages 281–287, San Mateo, CA, 1997. Morgan Kaufmann Publishers.

[9] V. N. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1998.

[10] T. Vetter and T. Poggio. Linear object classes and image synthesis from a single example image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:7:733–742, 1997.

	pbm	tdsz	aao	ii	ea	kgnl
pbm	0.94	0.03	0.0	0.01	0.0	0.02
tdsz	0.01	0.82	0.0	0.04	0.0	0.14
aao	0.01	0.04	0.77	0.04	0.11	0.03
ii	0.04	0.15	0.03	0.58	0.03	0.15
ea	0.02	0.13	0.26	0.10	0.36	0.14
kgnl	0.03	0.41	0.05	0.11	0.07	0.33

(a)

	pbm	tdsz	aao	ii	ea	kgnl
pbm	0.97	0.03	0.0	0.0	0.0	0.0
tdsz	0.04	0.82	0.0	0.0	0.0	0.14
aao	0.0	0.04	0.78	0.0	0.17	0.00
ii	0.07	0.07	0.04	0.41	0.0	0.41
ea	0.0	0.14	0.07	0.21	0.50	0.07
kgnl	0.0	0.38	0.0	0.08	0.05	0.47

(b)

	pbm	tdsz	aao	ii	ea	kgnl
pbm	0.91	0.05	0.0	0.0	0.01	0.02
tdsz	0.01	0.90	0.0	0.02	0.0	0.08
aao	0.0	0.04	0.77	0.02	0.16	0.05
ii	0.01	0.12	0.02	0.62	0.05	0.11
ea	0.03	0.07	0.31	0.16	0.26	0.16
kgnl	0.02	0.40	0.03	0.11	0.08	0.36

(c)

	pbm	tdsz	aao	ii	ea	kgnl
pbm	0.84	0.03	0.0	0.11	0.0	0.03
tdsz	0.02	0.83	0.0	0.04	0.0	0.12
aao	0.0	0.0	0.70	0.0	0.26	0.04
ii	0.07	0.14	0.0	0.71	0.0	0.07
ea	0.04	0.09	0.17	0.04	0.52	0.13
kgnl	0.0	0.28	0.0	0.03	0.13	0.58

(d)

Table 3: Confusion matrices for (a) LMM-representation, k nearest neighbor, k = 4, (b) LMM-representation, linear SVM, top-down multi-class, (c) wavelet-representation, k nearest neighbor, k = 4 and (d) wavelet-representation, linear SVM, top-down multi-class.