Object Modeling with Layers

Charudatta Phatak
cphatak@andrew.cmu.edu
Department of Materials Science and Engineering

***Abstract*—Appearance models provide a framework for modeling the shape and texture of objects from their images. The models use different forms of dimensionality reduction to represent the high dimensional data describing the objects. Objects like human faces are typically modeled using PCA based linear methods. But for certain class of objects like cars where there are instances of missing features like license plate, a simple global PCA based model fails, giving rise to undesired artifacts. To model such objects, instead of using a global model, we separate the images into layers and model each layer separately. This allows us to handle occlusions, missing features easily and also extends the model so that we can add or remove certain features. We will build such models using PCA based method and LLE based methods and compare the results.**

***Index Terms*—Layered Appearance Model, PCA model, LLE model**

I. Introduction

APPEARANCE based models of variable objects have been studied extensively in the recent years. Much of the focus has been primarily on human face modeling [1], [2], [3]. In the traditional model framework, the model is constructed from a training set of images and correspondence points identifying the landmarks or common features within the images. These points are typically hand-labelled. These landmarks relate the common features in the objects, e.g, nose,eyes of a person's face. In order to construct the model, a mean shape is computed and all the images are warped to the mean shape. These "shape-free" images are then processed using either principal component analysis (PCA) or Local Linear Embedding (LLE). The model is constructed for both the shape and the texture. The model can then be used for face recognition or synthesis of new faces.This framework gives good results for objects like faces. The warping involved is local and usually requires very small changes. The important thing though is that all the faces have the features/landmarks like eyes, nose etc. Some additional features may be present like glasses which may occlude the original feature points but it is assumed that they are always present.

However for objects like cars, there are problems with tracking the landmarks as all the images of the cars need not necessarily have the landmark points like fog lights, license plates. They are completely missing. When we try to model such objects using the previously mentioned appearance model, it results in the presence of significant artifacts. Moreover, there is no data for certain features which is creates problems in modeling with PCA based methods. Hence warping to mean shape is very difficult and does not necessarily represent a mean shape.

To overcome such problems, Jones et. al suggested using a layered model approach [4]. In this model, we separate the object into different layers according to the landmarks and apply weights for each layer depending on its presence/absence in the object. We then construct models for each layer using PCA/LLE methods.

In this paper, we will build such a layered appearance model for cars using PCA based method. We will also compare the result of PCA model with a LLE based model. Section 2 describes the construction of layers from images and then model construction for each layer. Section 3 describes the experimental results obtained by the application of the model to a training set of images. We demonstrate the additional capabilities of feature adding and feature matching which can be done using the layered model.

II. Model Description

We begin with a training set of images I, labeled with feature points S on each object. Now we divide the set of landmarks into separate layers $\phi = [1 \ldots G]$ where each layer corresponds to a particular feature. Then we construct arrays of S_ϕ describing the shape of each feature and T_ϕ describing the texture for each feature and W_ϕ describing the weight for the texture of each feature. Currently we define the weights naively with 1's when the feature is present and 0's when the feature is absent. Figure 1 shows the original car and the layers for the same car. We then compute the mean for each layer individually instead of a common mean shape. We then warp the texture and corresponding weight to the mean shape of that layer. For constructing model, we use the mean warped texture and corresponding warped weights. For shape, we subtract the mean shape and then construct the model.

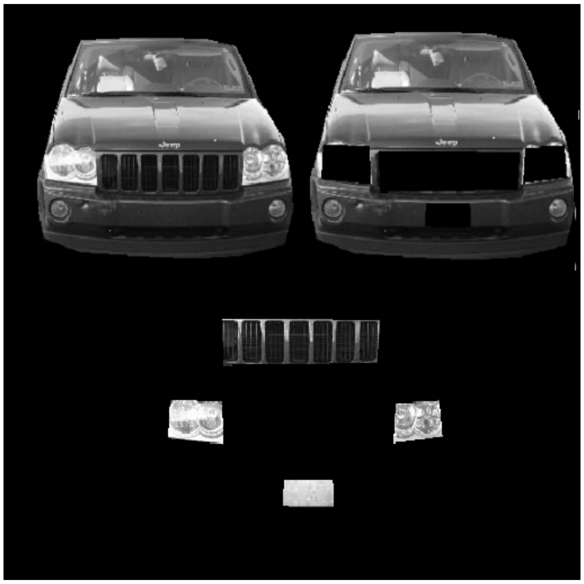

Fig. 1. The layers of a car object.

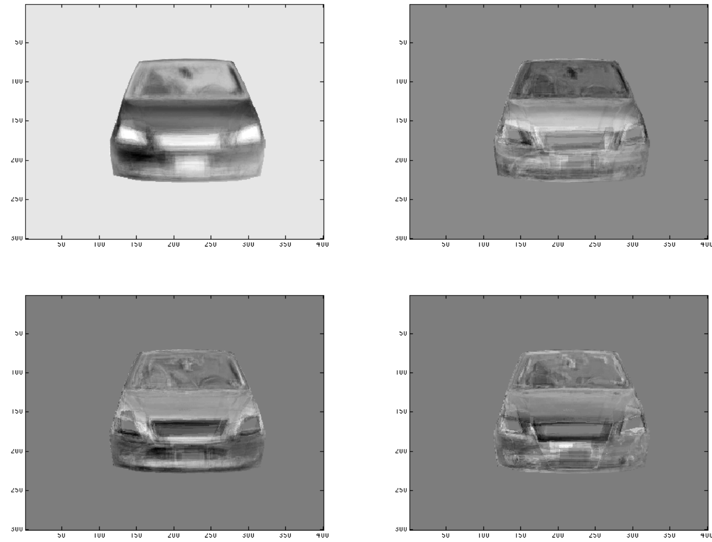

Fig. 2. The first 4 eigenvectors for the body layer of the car are shown. They have been warped to the mean shape.

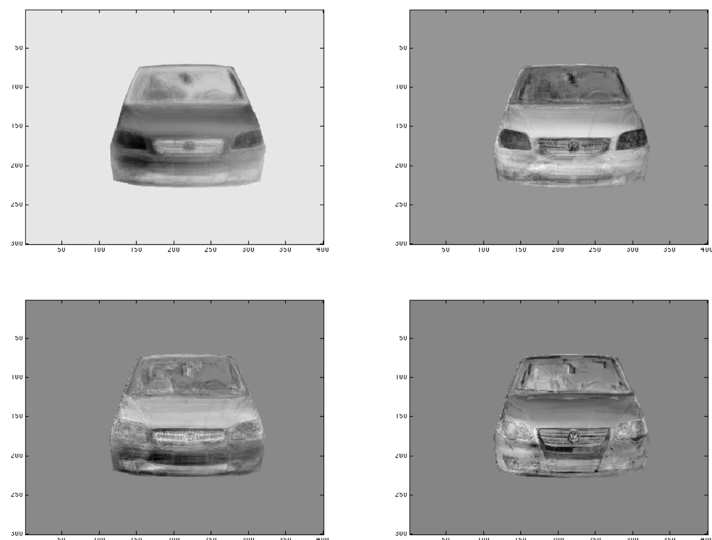

Fig. 3. The first 4 eigenvectors for the regular appearance model constructed using the same input images

A. PCA based model

We generate the model from the "mean free" shape vectors by applying the principal component analysis using the singular value decomposition method. Thus, any example of a shape vector can be represented by:

$$\mathrm{s}_\phi = \bar{\mathrm{s}}_\phi + [\mathbf{U_s}\mathrm{S_s}\mathrm{V}'_\mathrm{s}]_\phi \qquad (1)$$

where $\bar{\mathrm{s}}$ is the mean shape, $\mathbf{U_s}$ are the orthogonal eigenspace vectors, $\mathrm{S_s}$ are the corresponding weights of the eigenvectors and $\mathrm{V_s}$ is the representation of the shape in eigenspace.

In the same way, a model for the textures is constructed. In this paper, we used gray scale images of the cars. The images are then normalized to reduce the changes due to lighting effects. The texture of the car can then be represented as:

$$\mathrm{t}_\phi = \bar{\mathrm{t}}_\phi + [\mathbf{U_t}\mathrm{S_t}\mathrm{V}'_\mathrm{t}]_\phi \qquad (2)$$

where $\bar{\mathrm{t}}$ is the mean texture, $\mathbf{U_s}$ are the orthogonal eigenspace vectors, $\mathrm{S_s}$ are the corresponding weights of the eigenvectors and $\mathrm{V_s}$ is the representation of the texture in eigenspace.

These models are built for each of the layers. Figure 2 shows the first 4 eigenvectors for the body of the car. For a comparison, the first 4 eigenvectors of the regular appearance model for the same set of training images is shown in Figure 3.

B. LLE model

Local Linear Embedding (LLE) a form of dimensionality reduction that computes low-dimensional, neighborhood preserving embeddings of high dimensional data. The method is based on a unsupervised learning algorithm. This method has been used by Roweis and Saul for images of faces and documents of text [5]. Figure 4 shows the basic algorithm for implementing LLE. We define out dataset with an array X of D×N, where N is the number of images and D is the dimensionality typically the total no. of pixels in the image. We then select the nearest neighbors to a point X_i and compute weights that best reconstruct the point X_i based on those neighbors. This is done by minimizing the equation $\epsilon(W) = \sum_i |X_i - \sum_j W_{ij}X_j|^2$. After calculating

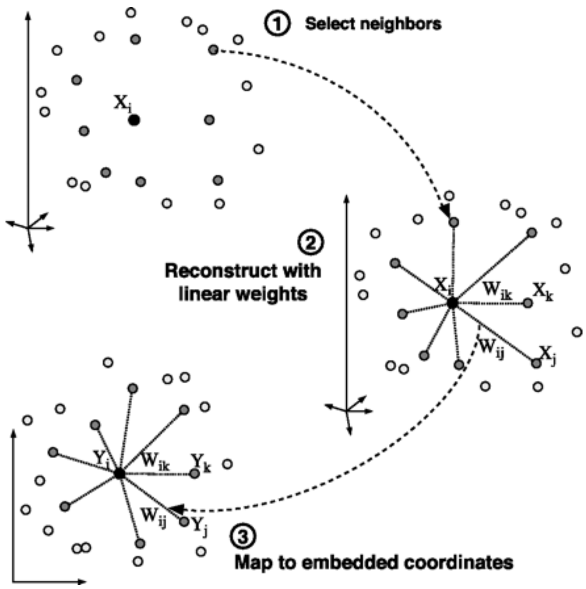

Fig. 4. Algorithm for implementing LLE

the weights, we compute the low-dimensional embedding vectors Y_i in a similar way by minimizing the equation $\phi(Y) = \sum_i |Y_i - \sum_j W_{ij} Y_j|^2$. The difference between this and previous equation is that in the second one we fix W and minimize with respect to Y. This gives us the low dimensional embedding vectors for our input data.

III. Experimental Results

We generated the layered appearance model for a collection of 20 frontal images of cars similar to the one showed in Figure 1. The images were converted to gray scale before construction of the model.

Once the model is generated, we can study various aspects of the model. Figure 5 shows the effect of using different number of eigenvectors for the reconstruction of a car from the training dataset along with the original car. The images show the car which is warped into the mean shape. We can randomly sample through our model and generate novel cars not present in the initial dataset. Figure 6 shows the images of new car images generated using the layered model. We can compare this with a similar random sampling done on the regular model shown in Figure 7. In the regular model image, we can clearly see ghosting effects and weird geometry artifacts near the headlights and grill area of the car. Since we have modeled each layer individually, we have additional functionality of adding or replacing features. The model estimates the shape and texture for the new features. Figure 8 shows the image of the original car and modified version of the car. We can also perform feature search within our model. Here we have used a grill of a car not in the dataset for matching. The grill image was converted into

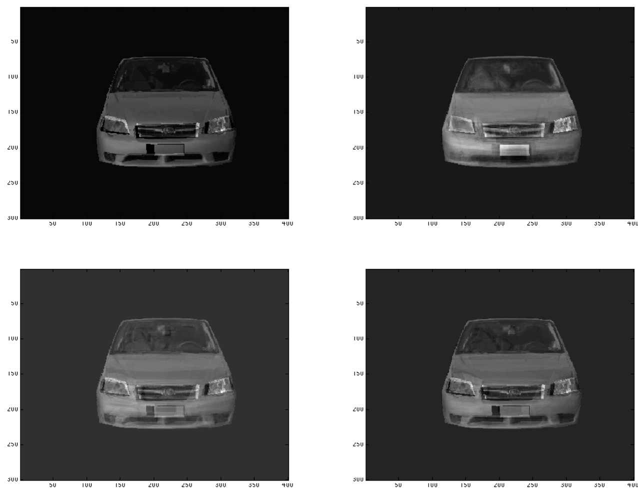

Fig. 5. The top-left image shows the original image of the car warped to mean shape. The anomalies are because the individual layers were warped to their respective mean shape. The next 3 images are the reconstruction using 5,10,15 eigenvectors respectively

Fig. 6. Novel image of car generated by randomly sampling through the model. Since we have separate models for each layer and the corresponding shape vectors, the results do not show any ghosting or tearing effects.

eigenspace and then a simple SSD match was performed in eigenspace. The matching result was obtained and the complete image of the car with matching grill was reconstructed from the model. Figure 9 shows the original car and the car with matching grill.

Similar to PCA based model, we also compute the LLE model based on the description given in the previous section. Figure 10 shows the data set distributed using the first two coordinates of LLE for each layer of the car. Figure 11 shows the corresponding images of the car represented by the points in the embedded space.A comparison of LLE and PCA was done by Roweis and Saul [5] which is shown

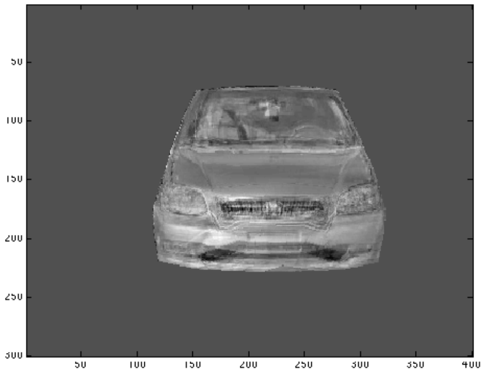

Fig. 7. Image is generated by random sampling though the regular model. The images show significant amount of ghosting and weird artifacts around the features like headlights etc.

Fig. 8. Images on left shows the original car. We can add the license plate to the car on top. Similarly we can replace the grill of the car on the bottom. Images on the left show the modified cars.

in Figure 13 for a set of images of a face translated along x-y directions. We also made a similar comparison between the LLE and PCA for a image of a car translated along x-y directions, shown in Figure 12.

IV. Conclusions

In this paper, we constructed a layer based appearance model for a input data set of cars. This allowed us to incorporate the occlusion or missing feature problem and also expanded the capability of the model further to include feature addition and feature matching. The LLE model was also constructed but due to less amount of images in data set, significant understanding or comparison of layered vs unlayered LLE model could not be studied. Further work would involve increasing the number of images so that the training set is representative of the entire class of cars. By doing this, trends in embedded space can be studied easily. The PCA based model fared will with low number of images in training set. It would be interesting to extend these models to include variation of pose or automatic feature detection.

Fig. 9. The image on the left shows the car not in dataset used for feature matching. In this case, we tried to find a matching grill from our input data set. The image on the left shows a car with matched grill.

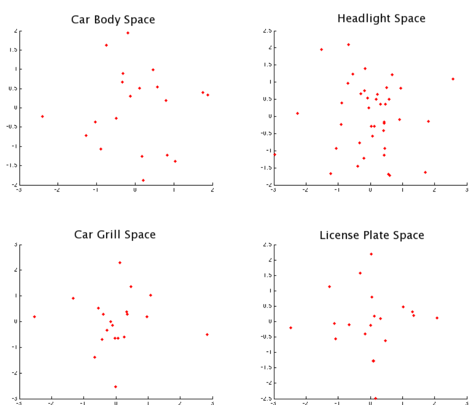

Fig. 10. Distribution of points in embedded space of different layers. The no. of points in headlights are more than 20 because the left and right headlight were modeled separately.

Acknowledgement

The author woule like to thank Prof. Alexei A. Efros for providing valuable insights and help regarding project

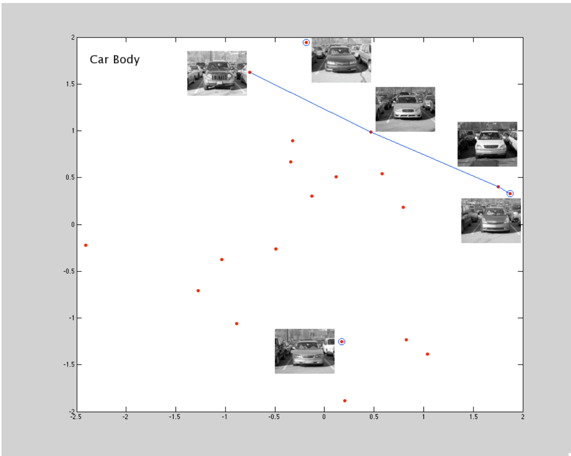

Fig. 11. Images representing the points in the embedded space for car body layer. Some trends are observed in the point distribution. The points far away from center are examples of car with very different body texture. The shape has not been incorporated in this model.

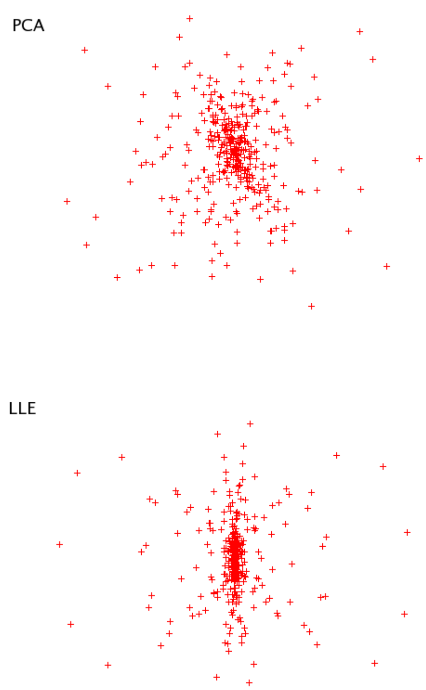

Fig. 12. Comparison of distribution of points using LLE and PCA. In both the cases, the first two coordinates were used.

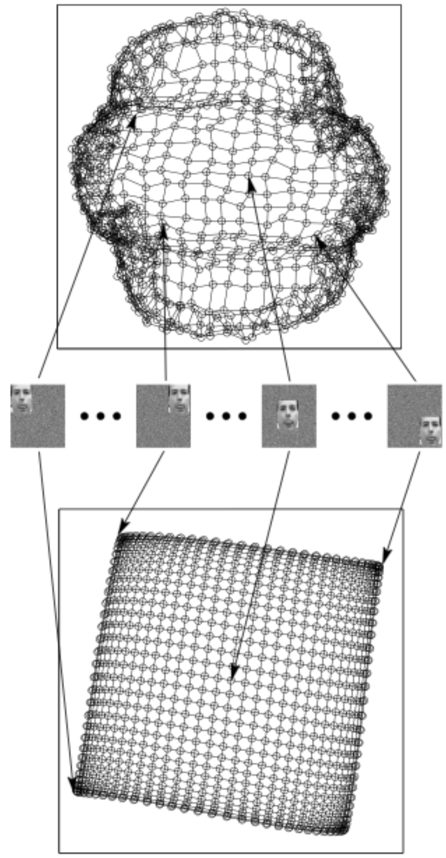

Fig. 13. Comparison of distribution of points using LLE and PCA for images of faces [6]

experiments.

REFERENCES

[1] T.F. Cootes, G.J.Edwards, and C.J.Taylor, *Active Appearance Models*, European Conference on Computer Vision, Vol.2, pp.484-498, 1998.

[2] M.A. Turk, and A.P. Pentland, *Face Recognition using Eigenfaces*, Proceedings, CVPR, 1991.

[3] D.A. Rowland, and D.I Perrett, *Manipulating Facial Appearance through Shape and Color*, IEEE Computer Graphics and Applications, Vol.15, pp.70-76, 1995.

[4] E. Jones, and S. Soatto, *Layered Active Apperance Models*, IEEE, ICCV, 2005.

[5] S.T. Roweis, and L.K. Saul, *Nonlinear Dimensionality Reduction by Locally Linear Embedding*, Science, Vol.290, pp.2323-2326, 2000.

[6] S.T. Roweis, and L.K. Saul, *Introduction to Locally Linear Embedding.* http://www.cs.toronto.edu/ roweis/lle/publications.html