Introduction

• Model-based approach towards image interpretation named deformable models has proven very successful.

• This is especially true in the case of images containing objects with large variability.

Introduction (Cont.)

• Among the earliest and most well known deformable models is the Active Contour Model – Known as Snakes proposed by Kass et. al.

• Snakes represent objects as a set of outline landmarks upon which a correlation structure is forced to constrain local shape changes.

Introduction (Cont.)

• Cootes et al. proposed the active shape models (ASM) where shape variability is learned through observation.

• In practice, this is accomplished by a training set of annotated examples followed by a Procrustes analysis combined with a principal component analysis.

Introduction (Cont.)

• A direct extension of the ASM approach has lead to the Active Appearance Models.

• Besides shape information, the textural information, i.e., the pixel intensities across the object, is included into the model.

View Correspondence in Identification

• Instead of trying to establish correspondence between individual (facial) feature points across views, one can learn a form of holistic shape based on a set of feature points.

• This gives more consistent and robust constraints for correspondence.

• In other words, if one is able to model the nonlinear shape variations of faces between different views, one implicitly solves the problem of establishing correspondences for a set of facial landmark points across these views.
View Correspondence in Identification (Cont.)

- At a fixed or restricted view, the 2D shape of a non-rigid object such as a human face can in general be modeled using a linear active shape model (ASM) based on a set of facial landmark points and their local gray-levels.
- However, the local gray-levels around the landmarks vary widely in situations when fitting a 2D shape model to images of faces rotating in depth from the left to the right profile view.

Active Shape Models (ASM)

- Active shape models (ASM) are flexible models that have been used for the modeling and representation of a range of objects.
- An ASM consists of two parts:
  - A Point Distribution Model (PDM): It is used to model the shape of an object and its variants using a set of landmark points.
  - A set of Local Gray-Level (LGL) models: They are used to capture the local gray-level variations observed at each landmark point of the shape PDM.

PDM -- Shape and Landmarks

- Shape is all the geometrical information that remains when location, scale, and rotational effects are filtered out from an object.
- Landmark is a point of correspondence on each object that matches between and within populations. That is, it is a point which identifies a salient feature on an object and which is present on every example of the class.

Automatic Landmark Selection -- Baumberg and Hogg Method

- Baumberg and Hogg [94] describe a system which generates landmarks automatically for outlines of walking people.
- The outlines are represented as pixellated boundaries extracted automatically from a sequence of images using motion analysis.
- Landmarks are generated on an individual basis for each boundary by computing the principal axis of the boundary, identifying a reference pixel on the boundary at which the principal axis intersects the boundary, and generating a number of equally spaced points from the reference point with respect to the path length of the boundary.

Automatic Landmark Selection -- Baumberg and Hogg method

- While this process is satisfactory for silhouettes of pedestrians, it is unlikely that it will be generally successful (consider the hand example).
- Baumberg and Hogg [95] describe how the position of the landmarks can be iteratively updated in order to generate improved shape models generated from the landmarks.
Automatic Landmark Selection -- Hill et al. Method

• Assumption:
  – The object we wish to model can be represented by a closed boundary within an image.
  – The training set consists of a set of images, each containing one or more examples of the objects, which can appear with varying orientation, position, scale, and shape.
• In order to produce a set of landmarks on the boundary of each example, we must first be able to identify corresponding points on different examples of the object.

Automatic Landmark Selection -- Hill et al. Method (Cont.)

• Critical Point Detection (CPD) Algorithm can be used to choose a set of points which best represent the object shape as the vertices of a sparse polygon.
• The CPD algorithm assigns a critical value to each point on the boundary which is simply the area of the triangle constructed from the given point and its two immediate neighbors.
• An iterative decimation process is used which removes the point with the smallest critical value, recomputes the critical value of the immediate neighbors of the point which has just been deleted, and reidentifies the point with the smallest critical value.
• The process terminates when the remaining smallest critical value is above some threshold set by the user.

Landmark Examples

Point Distribution Model -- Shape Formulation (Cont.)

• A mathematical representation of an n-point shape in k dimensions could be concatenating each dimension in a k x n-vector.
• The vector representation for planar shapes would then be:
  \[ x = (x_1, y_1, x_2, y_2, \ldots, x_n, y_n)^T \]
• Notice that the above representation does not contain any explicit information about the point connectivity.
A classical statistical method for dealing with redundancy in multivariate data – such as shapes – is the linear orthogonal transformation: principal component analysis (PCA).

In the shape case, the data acquisition is straightforward because the landmarks in the shape vector constitute the data itself.

Hence, a PDM can be used to represent the shape of an object as a set of N labeled landmark points in a vector $x = (x_1, y_1, \ldots, x_n, y_n)$.

In order to generate a flexible model that captures the intrinsic variations of an object, N aligned (translated, rotated, and scaled) example object images are acquired as the training data set.

Align the training data is to align all the training shapes in an approximate sense. This is done by selecting for each example a suitable translation, scaling, and rotation to ensure that they all correspond as closely as possible. That is, the transformations are chosen to reduce (in a least-squares sense) the difference between the aligned shape and a ‘mean’ shape derived from the whole set.

The best such transformation may be found by minimizing the expression

$$E = \left[ X^1 - RX^2 \right]^T \left[ X^1 - RX^2 \right]$$

Equation 1

This minimization is a routine application of a least-square approach. Partial derivatives of $E$ are calculated with respect to the unknowns $(\theta, s, t_x, t_y)$ and set to 0, leaving simultaneous linear equations to solve.

In a pairwise fashion, rotate, scale, and align each $x^i$ with $x^1$ for $i = 2, 3, \ldots, M$ to give the set \{x$^1$, $x^{2(\text{align})}$, $x^{3(\text{align})}$, $x^{M(\text{align})}$\}.

Calculate the mean of the transformed shapes

Rotate, scale, and align the mean shape to align to $x^1$.

Rotate, scale, and align $x^{2(\text{align})}$, $x^{3(\text{align})}$, $x^{M(\text{align})}$ to match to the adjusted mean.

If the mean has not converged, go to step 2.
PDM
-- Approximate alignment of similar training shapes (Cont.)

- Step 3 is necessary because otherwise it is ill-conditioned (under-constrained).
- Without doing this, convergence will not occur.
- Final convergence may be tested by examining the differences involved in realigning the shapes to the mean.

**Equation 1**

\[ E = [X^i - RX^2 - (\ell_1, \ell_2, \ldots)] W [X^i - RX^2 - (\ell_1, \ell_2, \ldots)]^T \]

**Equation 2**

The elements of \( W \) indicate the relative 'stability' of each of the landmarks, by which a high number indicates high stability (so counts for more in the error computation), and a low number the opposite.

PDM
-- Shape Formulation (Cont.)

- It is assumed that the set of \( N \) shapes constitutes some ellipsoid structure of which the centroid – the mean shape – can be estimated as:

\[ \bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i \]

- The maximum likelihood estimate of the covariance matrix can thus be given as:

\[ \Sigma = \frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \]

PDM
-- Shape Formulation (Cont.)

- The modes of variation of the object shape are captured by applying PCA on the derivations of the \( N \) example objects from the mean object shape.

- That is: the principal axis of the \( 2n \)th dimensional shape ellipsoid are now given as the eigenvectors of the covariance matrix:

\[ \sum \phi_i = \phi_i \lambda_i \]

PDM
-- Shape Formulation (Cont.)

- Using PCA, valid shapes can be represented by a small set of principal components as:

\[ b = \Phi^T (x - \bar{x}) \]

Where the \( k \)th column of \( \Phi \) is the \( k \)th eigenvector of the covariance matrix whose eigenvalue is \( \lambda_i \).

- The compactness of the model is achieved by retaining only the modes of variations (principal components) which account for most of the variance (largest eigenvalues).

- That is: \( \Phi = [\Phi_1, \Phi_2, \ldots, \Phi_t] \) is eigenvectors corresponding to the \( t \) largest eigenvalue \( \lambda_i \).

PDM
-- Shape Formulation (Cont.)

- Any shape projected to the PCA space can be approximated and projected back into the input space using

\[ x = \bar{x} + \Phi \phi \]

- That is, a new shape instance can then be generated by deforming the mean shape by a linear combination of eigenvectors, weighted by \( b \), also called the modal deformation parameters.
Local Gray Level (LGL)

- PCA is also used to model the Local Gray-Level (LGL) at the location of each landmark point of the PDM.
- This is normally achieved by sampling gray level values along the profile normal to the shape boundary at each landmark point.
- An LGL model is learned for each landmark and together with the PDM, these are used to construct an Active Shape Model (ASM).

LGL: Computing Normal to the Boundary

\[ (t_x, t_y) = \frac{d_x}{\sqrt{d_x^2 + d_y^2}}, \quad \frac{d_y}{\sqrt{d_x^2 + d_y^2}} \]

\[ (X_{i-1}, Y_{i-1}) \quad (X_{i+1}, Y_{i+1}) \]

\[ d_x = X_{i+1} - X_{i-1} \]

\[ d_y = Y_{i+1} - Y_{i-1} \]

(Unit vector)

LGL (Cont.)

- Sample k pixels either side of model point in the ith training image.
- Normalize the first derivatives of these profiles.

\[ g_i = \frac{g_i}{\sum_{j=1}^{N} |g_j|} \quad 1 \leq i \leq N \]

The 1st derivative of the ith training image at its landmark points

The 1st derivative of the ith training image at local regions of its landmark points

LGL: First Derivative Example

- E.g. Sample k pixels either side of model point in the ith training image.
• The LGL mean profile and the covariance matrix are computed as:
\[
\bar{g} = \frac{1}{N} \sum_{i=1}^{N} g_i
\]
\[
S_g = \frac{1}{N} \sum_{i=1}^{N} (g_i - \bar{g})(g_i - \bar{g})^T
\]
• They are also incorporated into the ASM.

Fitness of the LGL
• The fitness of the LGL of the new object (i.e., f(g)) with regards to the LGL mean profile and covariance matrix of the training images can be measured by:
  - Mahalanobis distance
\[
f(g) = (g - \bar{g})^T S_g^{-1} (g - \bar{g})
\]
  - or the average squared Euclidean distance
\[
f(g) = \frac{1}{N} [(g - \bar{g})^T (g - \bar{g})]
\]

Match an Unknown Object -- Descriptive Explanation
• An iterative search process is used to match an ASM to a novel object image.
• Each iteration consists of two steps:
  - **Step 1:** Given a starting position for the 2D shape of an object, one aims to move all the individual landmark points independently towards their local “ideal” positions, known as the local target points. This is based on the LGL models learned from the training examples.
  - **Step 2:** Once plausible movements have been estimated for each of the landmarks, the positions of the other landmarks are taken into account using the PDM. This ensures that the overall shape remains valid within the modes of variation learned by the PDM from the training examples.

Match an Unknown Object -- Descriptive Explanation (Cont.)
• In general, the local target points only give an initial estimate for the shape of an object. This initial shape may not be a valid one according to the PDM. Therefore, the shape is projected into the PCA shape of the PDM, the shape space, for comparison using
\[
b = \phi'(x - \bar{x})
\]
• In this shape space, there exists a continuous region known as the **Valid Shape Region (VSR).** This VSR defines the following limits for all valid shape variations in the directions of the principal components:
\[
x = \bar{x} + \phi b, \quad -3\sqrt{\lambda_1} \leq b_1 \leq 3\sqrt{\lambda_1}
\]
• If the projected initial shape given by the local target points is outside the VSR of the PDM, the nearest valid shape is projected back to the landmark feature space using
\[
x = \bar{x} + \phi b
\]
• This gives a new starting position for the shape of the face in the image and **step 1** is repeated.
• This iterative process is continued until the changes in the shape become negligible.
Fitting an ASM -- Pseudo code

1. Initialize an approximate fit to the training data. This yields local (model) co-ordinates $\mathbf{x}$ for a shape description.

2. At each landmark point, inspect the boundary normal close to the boundary, and locate the pixel of highest intensity gradient; mark this as the best target position to which to move this landmark point.

3. Adjust the pose parameters $(\theta, s, t, \phi)$ to provide the best fit to the target points of the current landmarks. (Refer to the "Align the training data" section).

4. Determine the displacement vector $\mathbf{dx}$ that adjusts the model in the new pose to the target point (i.e., After adjusting the pose variables, there remain residual adjustments which can only be satisfied by deforming the shape of the model).

5. Determine the model adjustment $\mathbf{db}$, that best approximates $\mathbf{dx}$. That is:

$$
\mathbf{dx} \approx \Phi \mathbf{db} \Rightarrow \mathbf{db} \approx \Phi^T \mathbf{dx}
$$

Note that we need to prevent components of the vector $\mathbf{b}$ from growing in magnitude beyond any limits.

6. Repeat from step 2 until changes become negligible.

Problems in Face Identification

- Unfortunately, linear ASMs of faces can only cope with very limited pose variations.

- An implicit but important assumption is that correspondences between landmark points of different views can be established solely based on the gray-level information.

ASM Example

a) Initial position b) After 10 iterations c) At convergence of ASM search
Nonlinear Shape Models

A multi-view, nonlinear, active face shape model can overcome many of the limitations of a single-view, linear ASM.

Such a multi-view model is learned from a set of labeled face images and their corresponding 2D shapes across the view sphere.

In particular, the model is required to perform a nonlinear transformation between significantly different views in order to extract shape-free textures for learning statistical knowledge of facial identities.

This nonlinear model transformation is achieved using a type of nonlinear PCA, the Kernel PCA.

It is important to point out that in order to constrain model search in KPCA space so as to both speed up and avoid local minima in model transformation across views, one aims to utilize any form of contextual knowledge.

Kernel PCA

Cover’s theory states that a complex pattern classification problem cast in a high-dimensional space nonlinearly is more likely to linearly separable than in a low-dimensional space.

Kernel Functions:

- Polynomial kernel: \( k(x, y) = (x \cdot y)^d \)
- Gaussian kernel: \( k(x, y) = e^{-\frac{||x - y||^2}{2\sigma^2}} \)
- Sigmoid kernel: \( k(x, y) = \tanh(k(x \cdot y + \nu)) \)

The use of labeled pose information on all the training images and their corresponding shapes is one obvious strategy to adopt.

This is realized by explicitly encoding 3D pose information in 2D active face shape model representation.

In other words, both the shape PDM and the corresponding LGL models for the landmarks are trained by concatenations of feature vectors and their corresponding known pose angles.

Given an initial position of a novel face in an image, one can apply this pose-indexed and KPCA-based, nonlinear, active face shape model to match faces at different poses.

The process of matching the nonlinear face ASM to a novel image finds both the 2D shape and the corresponding 3D pose of the face.

Such a multi-view, active face model can be used to establish nonlinear shape correspondence with both known and unknown (novel) faces across views from profile to profile.
Modes Determination

- Essentially, the point or nodal representation of shape has now been transformed into a modal representation where modes are ordered according to their deformation energy – i.e., the percentage of variation that they explain.
- What remains is to determine how many modes to retain. This leads to a tradeoff between the accuracy and the compactness of the model. However, it is safe to consider small-scale variation as noise.

Active Appearance Model (AAM) -- Texture Formulation

- In computer graphics, the term texture relates directly to the pixels mapped upon virtual 2D and 3D surfaces. Thus, texture can be defined as:
  - Texture is the pixel intensity across the object in question (if necessary after a suitable normalization).

AAM -- Texture Formation (Cont.)

- A vector is chosen, as the mathematical representation of texture, where \( m \) denotes the number of pixel samples over the object surface:
  \[
g = (g_1, g_2, \ldots, g_m)^T
  \]
- In the texture case, one needs a consistent method for collecting the texture information between the landmarks, i.e., an image warping function needs to be established.

AAM -- Texture Formulation (Cont.)

- Image warping will warp each example image so that its control points match the mean shape.
- Image warping can be done in several ways:
  - A piece-wise affine warp based on the Delaunay triangulation of the mean shape is used.
  - Thin-plate splines.
AAM
-- Texture Formulation (Cont.)
• Following the warp sampling of pixels, a photometric normalization of the g-vectors of the training set is done to avoid the influence from global linear changes in pixel intensities.
• Hereafter, the analysis is identical to that of the shapes.

AAM
-- Texture Formulation (Cont.)
• Hence, a compact PCA representation is derived to deform the texture in a manner similar to what is observed in the training set:
  \[ g = \bar{g} + \phi_g b_g \]
• Where \( \bar{g} \) is the mean texture
• \( \phi_g \) represents the eigenvector of the covariance matrix
• \( b_g \) are the modal texture deformation parameters.

AAM
-- Combined Model Formulation
• Concatenate vector \( b_s \) and \( b_g \) due to the linear nature of the model:
  \[ b = \begin{pmatrix} w_i b_i \\ b_g \end{pmatrix} = \begin{pmatrix} w_i \phi_i^T (x - \bar{x}) \\ \phi_g^T (g - \bar{g}) \end{pmatrix} \]
• Where a suitable weighting between pixel distances and pixel intensities is obtained through the diagonal matrix \( W_s \).

AAM
-- Combined Model Formulation
• A 3rd PCA is performed on the shape and texture PCA scores of the training set, \( b \), to obtain the combined model parameter, \( c \):
  \[ b = Qc \]
  where \( Q \) is the eigenvector and \( c \) is a vector of appearance parameters controlling both shape and gray-levels of the models.
• Regarding the compression of the model parameters, the rank of \( Q \) will never exceed the number of examples in the training set.
• This 3rd PCA can remove correlation between shape and texture model parameters and make the model representation more compact.

AAM
-- Combined Model Formulation (Cont.)
• Use simple linear algebra, a complete model instance including shape, \( x \), and texture, \( g \), is generated using the c-model parameters:
  \[ x = \bar{x} + \phi \bar{w}_s Q_s c \]
  \[ g = \bar{g} + \phi \bar{w}_g Q_g c \]
  where
  \[ Q = \begin{pmatrix} Q_s \\ Q_g \end{pmatrix} \]

AAM
-- Combined Model Formulation (Cont.)
• AAM Example

AAM Example

Labelled image Points Shape-free patch
AAM Example

AAM Variant (1)
- The first alternative approach is to perform the two initial PCAs based on the correlation matrix as opposed to the covariance matrix.

\[
rr = \frac{1}{n-1} \sum \left( \frac{x - \bar{x}}{S_x} \right) \left( \frac{y - \bar{y}}{S_y} \right)
\]

AAM Variant (2)
- The second feasible method to obtain the combined model by concatenating both shape points and texture information into one observation vector from the start and then performing PCA on the correlation matrix of these observations.

Search Optimization
- In AAMs, the search is treated as an optimization problem in which the difference between the synthesized object delivered by the AAM and an actual image is to be minimized.
- In this way, by adjusting the AAM-parameters (c and pose), the model can deform to fit the image in the best possible way.

Search Optimization
- Though we have seen that the parameterization of the object class in question can be compacted markedly by the principal component analysis, it is far from an easy task to optimize the system.
- This is not only computationally cumbersome but also theoretically challenging – optimization theory-wise – since it is not guaranteed that the search-hyperspace is smooth and convex.

Offline Parameter Optimization
- It is proposed that the spatial pattern in \( \delta I \) can predict the needed adjustments in the model and pose parameters to minimize the difference between the synthesized object delivered by the AAM and an actual image.

\[
\delta I = I_{\text{image}} - I_{\text{model}}
\]
- The simplest model we can arrive at constitutes a linear relationship

\[
\delta c = R \delta I
\]
Offline Parameter Optimization (Cont.)

• To determine a suitable R, a set of experiments are conducted, the results of which are fed into a multi-variate linear regression using principal component regression due to the dimensionality of the texture vectors.
• Each experiment displaces the parameters in question by a known amount and measuring the difference between the model and the image-part covered by the model.

Offline Parameter Optimization (Cont.)

• The optimization is performed as a set of iterations, where the linear model, in each iteration, predicts a set of changes in the pose and model parameters leading to a better model to image fit.
• Convergence is declared when an error measure is below a suitable threshold.

Offline Parameter Optimization (Cont.)

• Several error measures may be used:
  – Squared L2 norm
  – Mahalanobis distance
  – Lorentzian error norm
• Fitness functions allowing for global non-linear transformations such as the mutual information measure might also be considered.

Initialization

• The optimization scheme described above is inherently sensitive to a good initialization.
• The search-based scheme explained in the paper [Stegmann et al.] can make the use of AAMs fully automated.

Initialization (Cont.)

• The fact that the AAMs are self-contained or generative is exploited in the initialization – i.e., they can fully synthesize (near) photo-realistic objects of the class that they represent with regard to shape and texture appearance.
• Hence, the model, without any additional data, is used to perform the initialization.

References

-- Snake (Active Contour Models)

References
-- AAM


References
-- Image Warping


References
-- Automatic Landmark Extraction


References
-- Online Information

• http://www.isbe.man.ac.uk/~bim/
• http://www.isbe.man.ac.uk/val/asmtk/
• http://www.cs.sfu.ca/~hamarneh/software/asm/