Representations for Object Detection

Instructor - Simon Lucey

16-423 - Designing Computer Vision Apps
Today

- Exhaustive Search & Sampling
- V1 Inspired Descriptors
- Generative Methods
Naive Approach

“Images at various warps (p)”
Naive Approach

“Images at various warps (p)”
Naive Approach

• If you were a person coming straight from machine learning you might suggest,

\[
D = \begin{pmatrix}
255,134,45,\ldots,34,12,124,67 \\
123,244,12,\ldots,134,122,24,02 \\
67,13,245,\ldots,112,51,92,181 \\
65,09,67,\ldots,78,66,76,215
\end{pmatrix}
\]

“Vectors of pixel values at each warp position”
Naive Approach

- If you were a person coming straight from machine learning you might suggest,

\[
\begin{align*}
\{255,134,45,\ldots,34,12,124,67\} \\
\{123,244,12,\ldots,134,122,24,02\} \\
\{67,13,245,\ldots,112,51,92,181\} \\
\{65,09,67,\ldots,78,66,76,215\}
\end{align*}
\]

“Vectors of pixel values at each warp position”

\[D(\ldots,112,51,92,181)\]

“matching function”

We refer to this as Exhaustive Search!!!
Sampling?

- How do we sample every warp parameter value?

\[ p = \{ p_1, p_2 \} \]

"Possible Source Warps"
Sampling?

• How do we sample every warp parameter value?

\[ p = \{ p_1, p_2 \} \]
Sampling

• Shannon sampling theorem:
  
  if you sample densely enough (at the Nyquist rate) you can perfectly reconstruct the original data.

• We will show the desired sampling rate is dependent on the “centre frequency” of the salient edges.
Measures of Similarity

- Sampling strategy dependent on similarity measure,
- Sum of squared differences (SSD)

\[ D(p) = \sum_{i=1}^{M} \left\| I(W(x_i; p)) - T(x_i) \right\|^2 \]
Measures of Similarity

- Sampling strategy dependent on similarity measure,
  - Sum of squared differences (SSD)

\[ D(p) = \| I(p) - T(0) \|^2 \quad \text{"Vector Form"} \]

\( I \)
“Source Image”

\( T \)
“Model”
Measures of Similarity

- Sampling strategy dependent on similarity measure,
- Linear Correlation,

\[ D(p) = -I(p)^T T(0) \]

“Can be done efficiently using 2D convolutions....”
Oriented Edges

- Well known that natural images can be represented as a linear summation of oriented edges,

"Useful as edges capture ONLY relative local change in intensity...."
Sensitivity to Shift

Pixel Intensity vs. Warp(p)

Pixel Intensity vs. Pixel Coordinates

D(p)

T(0) “model” and I(p) “image”

Pixel Intensity vs. Warp(p)

Pixel Intensity vs. Pixel Coordinates

D(p)
Sensitivity to Shift

$D(p)$

Pixel Coordinates

$Warp(p)$

Pixel Intensity

$T(0) \text{ "model"}$  $I(p) \text{ "image"}$
Sensitivity to Shift

\[
\Delta p_l
\]

\[
\Delta p_h
\]
Sensitivity to Shift

• Clear that sampling is linked to center frequency of edge.

\[ p = \{ p_1, p_2 \} \]

“Possible Source Warps”
Sensitivity to Shift

- Clear that sampling is linked to center frequency of edge.

\[ p = \{ p_1, p_2 \} \]
Sensitivity to Shift

• Edges have a center frequency/wavelength,

• Clear that wavelength of the most “salient” edge is proportional to sample size:-

\[ \lambda \propto \Delta \rho \]
Sensitivity to Shift

• Edges have a center frequency/wavelength,

• Clear that wavelength of the most “salient” edge is proportional to sample size:-

\[ \lambda \propto \Delta \rho \]
Sensitivity to Shift

• Edges have a center frequency/wavelength,

• Clear that wavelength of the most “salient” edge is proportional to sample size:-

\[ \lambda \propto \Delta \rho \]
• Tolerance to shift is also important when learning a model.
Sensitivity to Shift

- Tolerance to shift is also important when learning a model.
- Most classification algorithms involve learning a weighted linear sum of the training examples.

\[ T(0) = \frac{1}{N} \sum_{n=1}^{N} \alpha_n I(p + \Delta p_n) \]
Sensitivity to Shift

\[ I(p + \Delta p_1) \]
\[ I(p + \Delta p_2) \]
\[ I(p + \Delta p_3) \]
\[ I(p + \Delta p_4) \]
\[ I(p + \Delta p_5) \]
\[ I(p + \Delta p_6) \]
\[ I(p + \Delta p_7) \]
\[ \ldots \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p + \Delta p_n) \]
Sensitivity to Shift

\[ I(p + \Delta p_1) \]
\[ I(p + \Delta p_2) \]
\[ I(p + \Delta p_3) \]
\[ I(p + \Delta p_4) \]
\[ I(p + \Delta p_5) \]
\[ I(p + \Delta p_6) \]
\[ I(p + \Delta p_7) \]
\[ \vdots \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p + \Delta p_n) \]
Sensitivity to Shift

\[ I(p + \Delta p_1) \]
\[ I(p + \Delta p_2) \]
\[ I(p + \Delta p_3) \]
\[ I(p + \Delta p_4) \]
\[ I(p + \Delta p_5) \]
\[ I(p + \Delta p_6) \]
\[ I(p + \Delta p_7) \]

\[ \cdots \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p + \Delta p_n) \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p) * \delta(\Delta p_n) \]
Sensitivity to Shift

\[ I(p + \Delta p_1) \]
\[ I(p + \Delta p_2) \]
\[ I(p + \Delta p_3) \]
\[ I(p + \Delta p_4) \]
\[ I(p + \Delta p_5) \]
\[ I(p + \Delta p_6) \]
\[ I(p + \Delta p_7) \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p + \Delta p_n) \]

\[ I(p) \ast \frac{1}{N} \sum_{n=1}^{N} \delta(\Delta p_n) \]
Sensitivity to Shift

\[ I(p + \Delta p_1) \]
\[ I(p + \Delta p_2) \]
\[ I(p + \Delta p_3) \]
\[ I(p + \Delta p_4) \]
\[ I(p + \Delta p_5) \]
\[ I(p + \Delta p_6) \]
\[ I(p + \Delta p_7) \]

\[ \frac{1}{N} \sum_{n=1}^{N} I(p + \Delta p_n) \]

\[ I(p) \ast B \]
Faces, generally have their salient edges in mid-frequency range. Dealing with pixels directly gives good results.

- 640×480 pixels
- scales 24, 36, 48, 60, 72
- sliding every 8 pixels

⇒ 24,000 subwindows to classify

Pixels not Good for Bodies

- Human bodies rely on contrast with background.
- Reliant on high-frequency edges.
- As a result poor sensitivity to shift.

Dalal & Triggs, 2005
What About Blurring?

- As pointed out in seminal work by Berg and Malik (CVPR’01) the effectiveness of SSD will degrade with significant viewpoint change.
- Two options to match patches:
  1. simultaneously estimate the distortion and position of matching patch.
  2. to “blur” the template window performing matching coarse-to-fine.

Berg & Malik, 2001
What About Blurring?

• As pointed out in seminal work by Berg and Malik (CVPR’01) the effectiveness of SSD will degrade with significant viewpoint change.

• Two options to match patches:
  1. simultaneously estimate the distortion and position of matching patch.
  2. to “blur” the template window performing matching coarse-to-fine.

Option 2 is attractive, low computational cost!
What About Blurring?
What About Blurring?
What About Blurring?

“average”
What About Blurring?

“histogram”
“pooling”
“blurring”
What About Blurring?

“histogram”
“pooling”
“blurring”
What About Blurring?

“Edge Filter”

“Blur Kernel”

“Edge \times Blur Filter”
What About Blurring?

“Edge Filter”

“Blur Kernel”

“Edge ∗ Blur Filter”
What About Blurring

Clearly, blurring a high-frequency edge filter simply lowers the centre frequency (not what we want).

\[ \lambda_b > \lambda_e \]

“Blurred Edge Wavelength”

“High Frequency Edge Wavelength”
Today

- Exhaustive Search & Sampling
- V1 Inspired Descriptors
- Generative Methods
Primary Visual Cortex
Spatial Sensitivity

In Figure 6, which is typical, the magnitudes of kurtosis are photometric and geometric noise. Nevertheless, it would be prudent to test whether the statistics of the difference image, whose first-order properties are reflected in the pixel-difference histogram, is the reason for the relatively high geometric transformation thresholds. An anonymous reviewer suggested a way of doing this.

Take a baseline image $I_B$ and transform it, say by rotation, to image $I_T$. Call the difference between these two images $I_D = I_T - I_B$. Any difference between two images (even a difference caused by an affine transformation) can be described in terms of this difference image. In the third control experiment, we compare the thresholds for detecting the increment versus the decrement of this difference image. That is we compare the thresholds for $I_T$ versus $I_B$ and $I_C$ versus $I_B$.

$$I_T = I_B + I_D$$
$$I_C = I_B - I_D$$

Figure 7 provides an example of the two images that are created by adding and subtracting the difference image. The difference image in the two cases is identical. We can therefore ask whether an increment (which corresponds to Figure 5).
Spatial Sensitivity

adding white noise, the one on the right by stretching the image horizontally. The amount of transformation however is identical in terms of Euclidean distance. While it is easy to see the changes in the left image, the changes in the right image can only be seen with careful scrutiny.

Why are the geometric transformations so much more difficult to detect compared to added noise? One possibility is that the answer lies in the shape of the histogram of pixel differences between the original and transformed images. The pixel-difference histogram captures the first-order (point-wise) statistical differences between two images. Figure 6 shows the pixel-difference histograms for an image transformed by the same Euclidean distance in one of three ways: translation, brightening, and addition of Gaussian noise. The pixel-difference histogram is by definition a Gaussian for the added Gaussian noise condition. For translation it is more kurtotic, and for brightening it is a single-point function (all pixels are incremented by the same value) making it highly kurtotic. The marked difference in the shape of these pixel-difference histograms for the various transformations raises the possibility that pixel histogram shape is a factor determining thresholds. At face value, however, it would seem unlikely that kurtosis is the critical statistic since the relative magnitudes of thresholds are geometric photometric noise, whereas for the image Figure 4.

Figure 3. Example psychometric functions for KWs Gaussian noise (red symbols) and vertical translation (green squares) conditions from Experiment 1. The proportion of correctly detected transformations is plotted against the log Euclidean distance between the transformed and untransformed image. Error bars are binomial standard deviations. Continuous lines are best fitting logistic functions. The horizontal black line shows the 75% correct level, and the vertical dashed lines show the threshold log Euclidean distance for each condition.


Kingdom, Field, & Olmos, 2007
Hierarchical Learning

Hierarchical Learning

View-tuned cells

Complex

Simple
Hierarchical Learning

- Ventral Visual Stream
- IT
  - View-tuned cells
- V1
  - Simple
- V2/V4
  - Complex

Fig. 1.

Hierarchical Model

Riesenhuber and Poggio's model to perform well

- Recognizing known objects, but with limited
  - Viewpoint-invariant performance for
    - Different views

- View-tuned cells typically respond in an invariant manner to
  - Specific object views

- Adding additional input would be relatively easy
  - To achieve viewpoint-invariant recognition

HMAX

- Sun et al. 1997
- Riesenhuber and Poggio 1999
- A computational network
  - For viewpoint-invariant object recognition

- Relies on a non-linear maximum (MAX) operator
  - For combining feature detectors

- Viewpoint-dependent performance for
  - HMAX

- Critical prediction is progressively poor
  - For achieving either type of invariance

- View-based models have proposed similar normalizations
  - For handling viewpoint-specific object representations
  - Because their mechanisms do not completely independently of one another; rather they are 'pooled' to form
  - More complex object representations.

- Patterns of responses for feature detectors in HMAX,
  - Show precisely this sort of response pattern
  - Typically driven by simple stimulus features

- Some neurons that respond most strongly to
  - Viewpoint-invariant objects
  - Are 'view-tuned' - that is, preferentially to a particular viewpoint

- However, have not actually proposed mechanisms
  - To achieve viewpoint-invariant recognition

- Ventral visual cortex
  - Contains view-tuned units that are object-specific
  - More complex object representations can be achieved
  - By pooling simple stimulus features

- In this issue, Riesenhuber and Poggio
  - Demonstrate that this hierarchy continues
  - From a specific viewpoint in which an object's vertices
  - Are dealt almost completely independent of any particular

- As with the pattern of
  - Neurons that respond equally well to any view

- Some neurons appear to respond specifically to certain viewpoints
  - Others respond in an invariant manner to
  - Generalization - in the form of either

- The response of this unit. This method for
  - Strongest signal among features feeding
  - Subsequent stage. In the model, the use of

- Maximum operation ('MAX') for combining
  - More complex feature detectors at a

- HMAX are highly viewpoint-dependent.
  - In contrast to the wide explanatory
  - Ability to remove the impasse.

- Some, I have dubbed it 'HMAX'
  - Term for the model — 'Hierarchical Model'

- The body of extant results has been
  - Complementary to view-based models
  - For achieving either type of invariance.

- Viewpoint-specific object representations
  - Have not actually proposed mechanisms
  - To achieve viewpoint-invariant recognition

- View-based models have proposed similar normalizations
  - For handling viewpoint-specific object representations
  - Because their mechanisms do not completely independently of one another; rather they are 'pooled' to form
  - More complex object representations.
How much do we know about V1?

~85% of V1 function not understood

Figure 5: 85% of V1 function remains to be understood.

3 New theories

Given the above observations, it becomes clear that there is so much unexplored territory that it is very difficult to rule out theories at this point, although there are some obvious bounds dictated by neural architecture—e.g., the spatial extent of axonal and dendritic arbors, etc. In the sections below, we discuss some of the theories that are plausible given our current data. However, the goal here is not to provide a detailed review of the theories currently in the literature, rather the goal is to provide a few examples of the range of theories that are consistent with the experimental data. It must be emphasized that considering that there may exist a large family of neurons with unknown properties, and given the low level of prediction for the neurons studied, there is still considerable room for theories dramatically different than those theories presented here.

3.1 Dynamical systems and the limits of prediction.

Imagine tracking a single molecule within a hot gas as it interacts with the surrounding molecules. The particular trajectory of one molecule will be erratic and fundamentally unpredictable without knowledge of all other molecules with potential influence. Even if we presumed the trajectory of the particular molecule was completely deterministic and following simple laws, in a gas with large numbers of interacting molecules one could never provide a prediction of the path of a single molecule except over very short distances.

Olshausen & Fields, 2005
When the position in 3D of the face and its facial features has
respectively. 2D appearance is a weighting matrix. It is often advantageous to
is not symmetrical, we...
Sparseness and Positiveness

- Blurring only works if the signals being matched are sparse and positive.
- Unfortunately natural images are neither.
- Combination of oriented filter banks and rectification can remedy this problem with little loss in performance.
Sparseness and Positiveness

- Blurring only works if the signals being matched are sparse and positive.
- Unfortunately natural images are neither.
- Combination of oriented filter banks and rectification can remedy this problem with little loss in performance.
Sparseness and Positiveness

- Comes at additional computational cost, as new representation is F times larger (where F is the number of filters employed).

\[ \phi\{I(p)\} = \text{image descriptor function} \]
Sparseness and Positiveness

- Comes at additional computational cost, as new representation is F times larger (where F is the number of filters employed).
- Downsampling strategies can be employed to lessen this problem.
Sensitivity to Shift

No Blurring
Sensitivity to Shift

No Blurring
Sensitivity to Shift

Rectified Edge

$D(p)$

Warp $p$

Pixel Coordinates

Gaussian Blur
Sensitivity to Shift

Gaussian Blur

Rectified Edge

$\text{Warp}(p)$

$D(p)$

Pixel Coordinates
Sensitivity to Shift

Histogram Blur
Sensitivity to Shift

Histogram Blur
The Cohn Kanade dataset was used to explore the margin with relationships between frequency and orientation bands. The sight becomes clearer in Equation scriptors can be reinterpreted as a margin manipulation on 6. Discussion classification accuracy across all expressions with increasfl neutral expression in the first frame and the peak formation presentations have better robustness to registration error. The central magnified panel shows that with perfect registrationfi the rank

We register each face to a canonical pose then measure...
Photometric Normalization

- Nearly all detection methods employ some sort of photometric normalization.
- Based on old idea of the “Reflectance Perception Model”.

\[ I(x, y) \frac{1}{L(x, y)} = R(x, y) \]
Photometric Normalization

- Nearly all detection methods employ some sort of photometric normalization.
- Based on old idea of the “Reflectance Perception Model”.

\[
\frac{I(x, y)}{L(x, y)} = R(x, y)
\]

\(I\): Image, \(L\): Illuminance, \(R\): Reflectance

Gross & Brajovic, 2003, Land & McCann, 1971
Photometric Normalization

- When dealing with “smaller” image patches, can get similar types of performance through power-normalization.
SIFT descriptor [Lowe’99]

Approach
- 8 orientations of the gradient (**sparseness**)
- 4x4 spatial grid (**blur**)
- soft-assignment to spatial bins of gradient magnitude (**recitifcation**)
- normalization of the descriptor to norm one (**illumination**)
- comparison with Euclidean distance (**SSD**)

![Image of SIFT descriptor approach](image)

Lowe, 1999
VLFeat Library

- VLFeat is an open source library that implements many common feature extraction methods.
- Has MATLAB port and a C API.
- Common features include V1 inspired features like HOG, SIFT and Dense SIFT and many more!!!
V1 not always the answer

Figure 9

The three line-strokes at left are interpreted as different objects depending on the arrangement of occluders. Thus, pattern completion depends on resolving figure-ground relationships. At what level of processing is this form of completion taking place? Since it would seem to demand access to high-resolution detail in the image, it cannot simply be relegated to high-level areas.

Knowing about Vpk combined with the importance of rD surface representations for guiding behavior, it is a plausible hypothesis to consider. In addition, problems such as occlusion demand resolving figure-ground relationships in a relatively high-level representation where topography is preserved. (Lee et al.) There is now beginning to emerge physiological evidence supporting this idea. Neurons in Vp have been shown to produce a differential response to the figure vs. background in a scene of texture elements. (Lamme et al.)—Zipser et al. and a substantial fraction of neurons in Vp are selective to border ownership. (Zhou et al.) In addition, Lee et al. have demonstrated evidence for a medial axis representation of surfaces in which Vp neurons become most active along the skeleton of an object. It seems quite possible such effects are just the tip of the iceberg and there could be even more effects lurking.

3.5 Top-down feedback and disambiguation

Although our perception of the visual world is usually quite clear and unambiguous, the raw image data that we start out with is not. Looking back at Figure 10, one can see that even the presence of a simple contour can be ambiguous in a natural scene. The problem is that information at the local level is insufficient to determine whether a change in luminance is due to an object boundary, simply part of a texture, or a change in reflectance. Although boundary junctions are also quite crucial to the interpretation of a scene, a number of studies have now shown that human observers are poor judges of what constitutes a boundary or junction when these features are shown in isolation. (Elder et al.)—McDermott and thus, the calculation of what forms a boundary is dependent on the context, which provides information about the assignment of figure and ground, surface layout, and so forth. Arriving at the correct interpretation of an image, then, constitutes something of a chicken-and-egg problem between lower and higher levels of image analysis.
Today

• Exhaustive Search & Sampling
• V1 Inspired Descriptors
• Generative Methods
• SSD and correlation measures are good if the models stems from a **single** source image.
• Not so good if there is noise stemming from appearance variation, e.g.:
Eigen-Objects

• Is there a way to learn an object model that can handle these appearance variations?
• Turk and Pentland (1991) proposed a method they referred to as “Eigenfaces” that could handle appearance variation.
• The technique employed principal component analysis (PCA) to model how a registered object could vary in appearance.
Training Examples

\[ I_1(p_1) = T_1(0) \]

\[ I_2(p_2) = T_2(0) \]

\[ I_N(p_N) = T_N(0) \]
• First, we concatenate all the training objects into a large vector.

\[
C = [T_1(0), T_2(0), \ldots, T_N(0)]
\]

= [first three eigenvectors]

• Then we remove the mean,

\[
C \rightarrow \text{Ensemble mean } A_0(0)
\]

• Then we apply PCA.

\[
C^T C = \Lambda \Lambda^T = [a_1, \ldots, a_M] \text{diag } [\sigma_1, \ldots, \sigma_M] [a_1, \ldots, a_M]^T
\]

First three eigenvectors
Eigen-Objects

• We can use conventional SSD to then gain the match at a particular warp by minimizing,

\[
D(p, \lambda) = \|I(p) - A_0(0) - \sum_{m=1}^{M} \lambda_m A_m(0)\|^2
\]

• Interestingly, we can “project out” the appearance change so that we are just minimizing,

\[
D(p) = \|I(p) - A_0(0)\|^2_{\text{null}(A)}
\]

• where,

\[
\|u\|^2_{\text{null}(A)} = u^T u - u^T A A^T u
\]

“Sometimes referred to as the distance from feature space (DFFS)”
DFFS Interpretation

- PCA is useful for assuming the object stems from a low-dimensional manifold.
- Choose M < N eigenvectors for better “generalization”.
- Assume N-M other eigenvectors stem from sample noise.

“Eigenspectrum”
“Visualization”

Moghaddam & Pentland, 97
We have compared the detection performance of three different detectors on approximately 7,000 test images from this database: a sum-of-square-differences (SSD) detector based on the average facial feature (in this case the left eye), an eigentemplate or DFFS detector and a ML detector based on $S(i,j)$ as defined in Section 2.2. Fig. 5a shows the receiver operating characteristic (ROC) curves for these detectors, obtained by varying the detection threshold independently for each detector. The DFFS and ML detectors were computed based on a five-dimensional principal subspace. Since the projection coefficients were unimodal, a Gaussian distribution was used in modeling the true distribution for the ML detector as in Section 2.2. Note that the ML detector exhibits the best detection vs. false-alarm tradeoff and yields the highest detection rate (95 percent). Indeed, at the same detection rate, the ML detector has a false-alarm rate which is nearly two orders of magnitude lower than the SSD.

Fig. 5b provides the geometric intuition regarding the operation of these detectors. The SSD detector's threshold is based on the radial distance between the average template (the origin of this space) and the input pattern. This leads to hyperspherical detection regions about the origin. In contrast, the DFFS detector measures the orthogonal distance to $F$, thus forming planar acceptance regions about $F$. Consequently, to accept valid object patterns in $W$, which are very different from the mean, the SSD detector must operate with high thresholds which result in many false alarms. However, the DFFS detector cannot discriminate between the object class $W$ and non-$W$ patterns in $F$. The solution is provided by the ML detector which incorporates both the $F$-space component (DFFS) and the $F$-space likelihood (DIFS). The probabilistic interpretation of Fig. 5b is as follows: SSD assumes a single prototype (the mean) in additive white Gaussian noise, whereas the DFFS assumes a uniform density in $F$. The ML detector, on the other hand, uses the complete probability density for detection.

We have incorporated and tested the multiscale version of the ML detection technique in a face detection task. This multiscale head finder was tested on the FERET database where 97 percent of 2,000 faces were correctly detected. Fig. 6 shows examples of the ML estimate of the position and scale on these images. The multiscale saliency maps $S(i,j,k;W)$ were computed based on the likelihood estimate $P_x|W_{di}$ in a 10-dimensional principal subspace using a Gaussian model (Section 2.2). Note that this detector is able to localize the position and scale of the head despite variations in hair style and hair color, as well as presence of sunglasses. Illumination invariance was obtained by normalizing the input subimage $x$ to a zero-mean unit-norm vector. We have also used the multiscale version of the ML detector as the attentional component of an automatic system for recognition and model-based coding of faces. The block diagram of this system is shown in Fig. 7, which consists of a two-stage object detection and alignment stage, a contrast normalization stage, and a feature extraction stage whose output is used for both recognition and coding. Fig. 8 illustrates the operation of the detection and alignment stage on a natural test image containing a human face. The function of the face finder is to locate regions in the image which have a high likelihood of containing a face.
Eigen-Objects = Generative

• Approach is inherently a generative approach:
  • Learn how to synthesize a registered object’s appearance variation using a linear model. (i.e. PCA)
  • Invert model to gain a measure of similarity. (i.e., DFFS).

• Unfortunately, like many generative models this approach has poor generalization properties.
  • What happens if my model cannot synthesize the registered object in a given source image?

Generative Model

“Source Image”
Eigen-Objects = Generative

• Approach is inherently a generative approach:
  • Learn how to synthesize a registered object’s appearance variation using a linear model. (i.e. PCA)
  • Invert model to gain a measure of similarity. (i.e., DFFS).
• Unfortunately, like many generative models this approach has poor generalization properties.
  • What happens if my model cannot synthesize the registered object in a given source image?
Discriminative Approaches

• Better generalization performance can often be realized by learning the difference between two classes.
• We no longer get caught up with the problem of attempting to synthesize all variations of an object.

“How do we get negative examples?”

More next lecture!
