

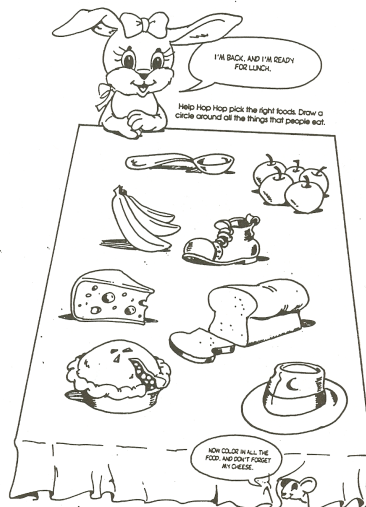
# High-Level Vision

## Object Recognition



**CS332 Visual Processing**  
Department of Computer Science  
Wellesley College

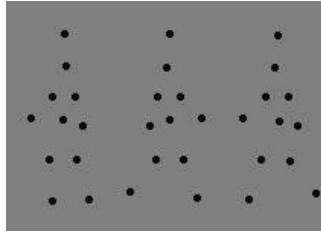
## Recognition from geometric shape



Objects can be recognized easily from the shape of their image contours

Very young children can easily recognize a wide variety of common objects

## Other recognition cues



characteristic motion



color



context

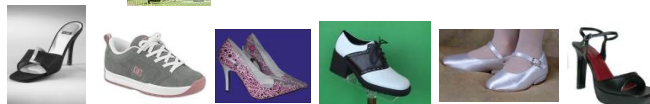
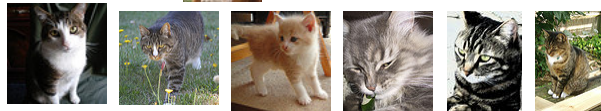


texture



1-3

## Why is recognition difficult?



1-4

## What is a chair?

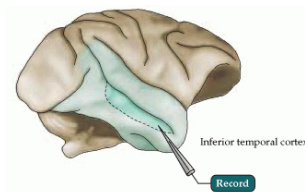


1-5

## Face recognition



AR Face Database



1-6

## Approaches to recognition...

... differ in how *regularities* are used to constrain the interpretation of the viewed object

Three main approaches:

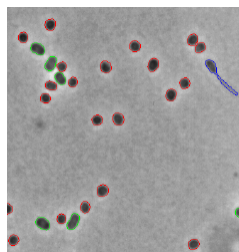
- invariant properties
- parts decomposition
- alignment

1-7

## Invariant properties

Every instance of each object class exhibits certain properties

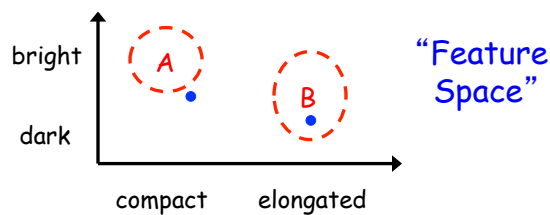
- (1) measure properties of viewed object
- (2) apply decision procedure



classify cells in culture

Properties:

- (1) ratio: perimeter/area
- (2) brightness



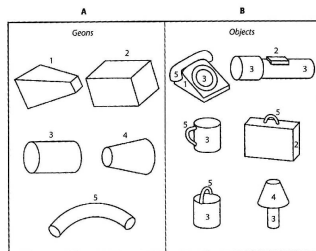
1-8

## Parts decomposition

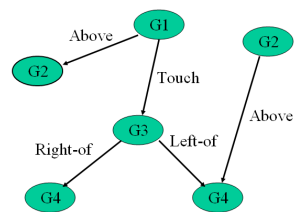
Every instance of each object class shares certain parts, arranged in a certain way

(1) find object parts

(2) recognize objects by presence of parts with proper relationship



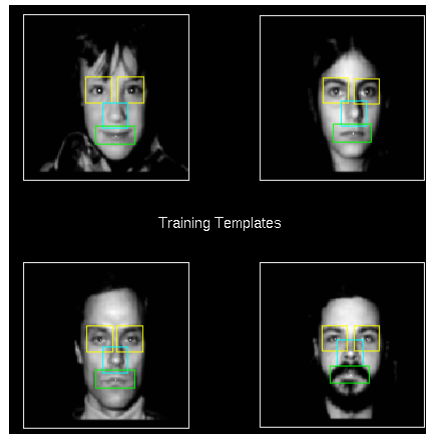
Biederman's Geons



"Structural Description"

1-9

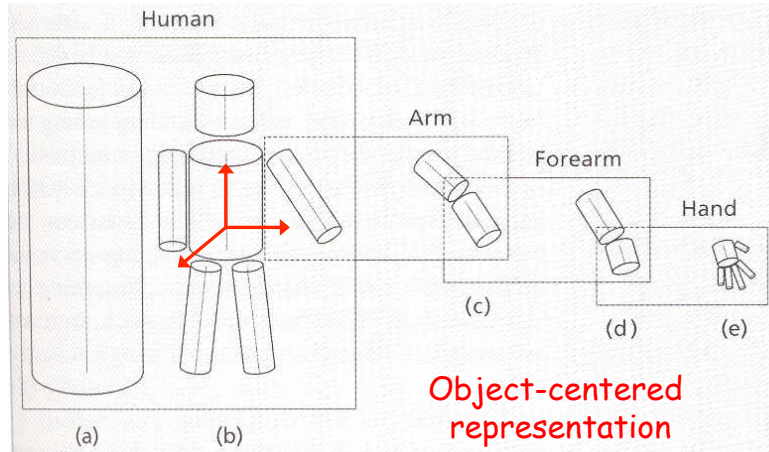
## Face recognition by parts decomposition



MIT Media Lab Vision & Modeling Group

1-10

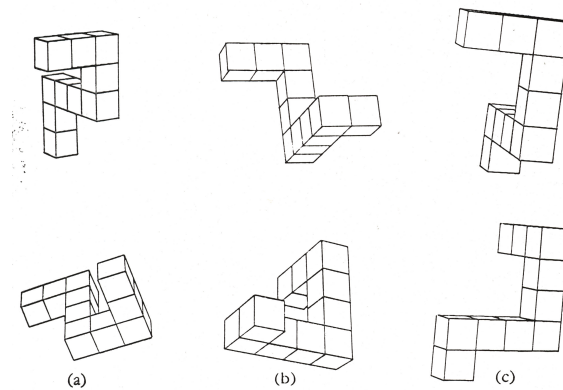
## Feature hierarchies



Marr & Nishihara

1-11

## Mental rotation

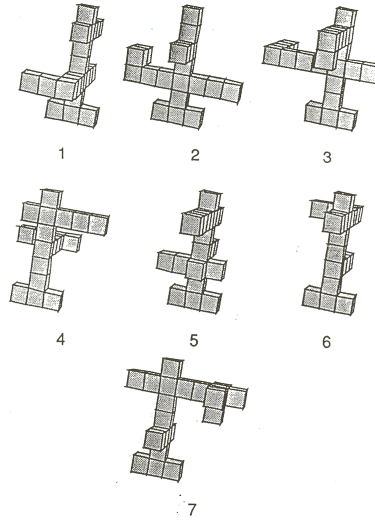


Time needed to determine whether pair of objects are the same is proportional to angle of rotation between pair

1-12

## Viewer-centered object representation?

Tarr, '95: After learning to recognize a set of 3-D objects from a small set of specific 2-D views of these objects, the time needed to recognize a novel view is proportional to the 3-D angle between the new view and closest learned view



1-13

## The debate continues...

Viewpoint invariant  
object representations



Viewpoint dependent  
object representations



Biederman

Object-centered!

Viewer-centered!



Bulthoff

1-14

## Alignment methods

Find an object model and geometric transformation that *best match* the viewed image

$V$  viewed object (image)

$M_i$  object models

$T_{ij}$  allowable transformations between viewed object and models

$F$  measure of fit between  $V$  and the expected appearance of model  $M_i$  under the transformation  $T_{ij}$

**GOAL:** Find a combination of  $M_i$  and  $T_{ij}$  that maximizes the fit  $F$

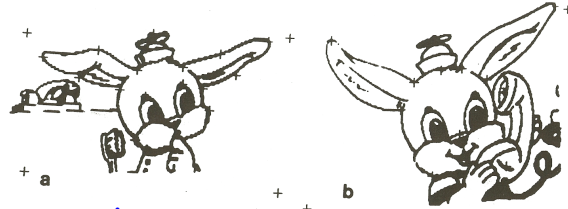
1-15

## Alignment method: recognition process

- (1) Find best transformation  $T_{ij}$  for each model  $M_i$  (optimizing over possible views)
- (2) Find  $M_i$  whose best  $T_{ij}$  gives the best match to image  $V$

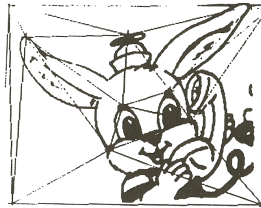
1-16

## Aligning pictorial models



image

model



triangulated model



transformed model  
superimposed on image

1-17

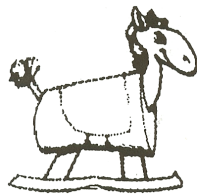
## When the model doesn't fit...



image



model



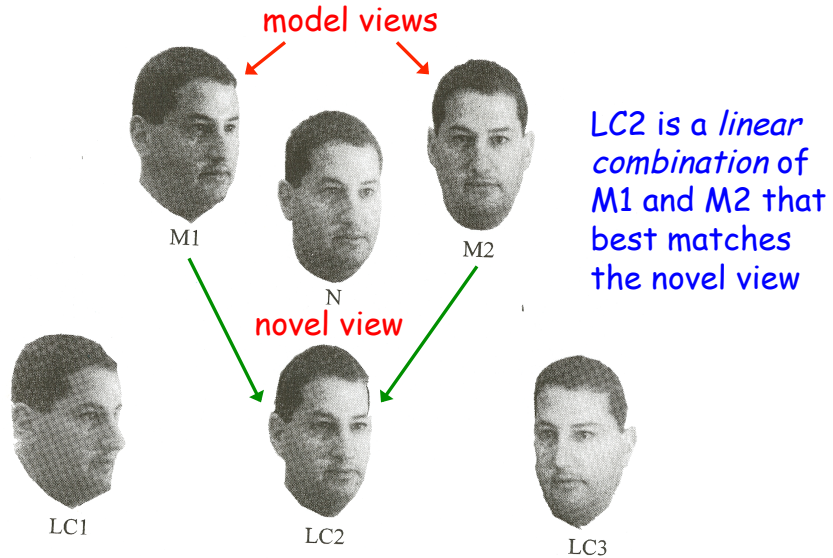
transformed  
model



transformed model  
superimposed on image

1-18

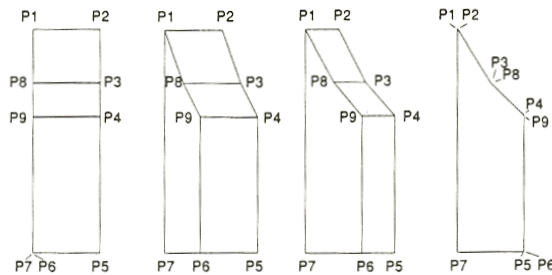
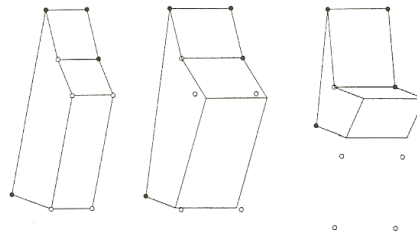
## Recognition by linear combination of views



1-19

## Obelisk, jukebox or seat?

Each object model consists of multiple 2-D views

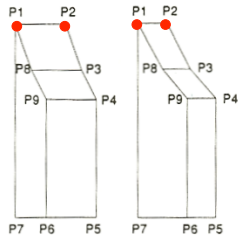


obelisk model

Goal: recognize novel views of these objects

1-20

## Predicting object appearance



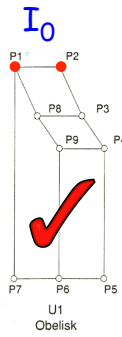
$I_1$

$I_2$

two known views  
of obelisk

$$X_{P_1 I_0} = \alpha X_{P_1 I_1} + \beta X_{P_1 I_2}$$

$$X_{P_2 I_0} = \alpha X_{P_2 I_1} + \beta X_{P_2 I_2}$$



Recognition process:

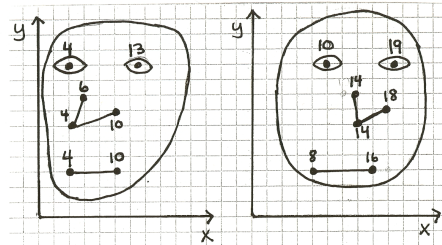
(1) compute  $\alpha, \beta$  that predict P1 and P2

(2) use  $\alpha, \beta$  to predict other points

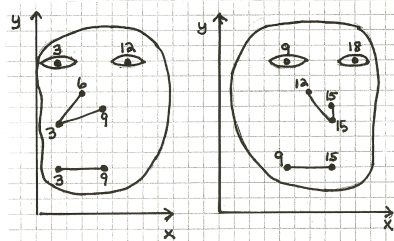
(3) evaluate fit of model to image

1-21

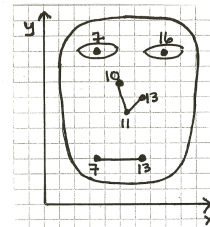
## Face recognition by linear combination of views



Harry model



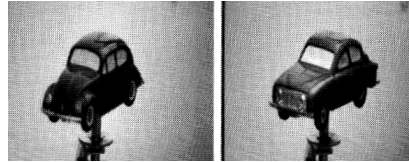
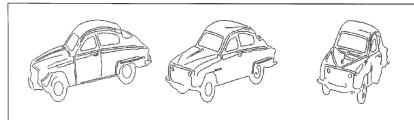
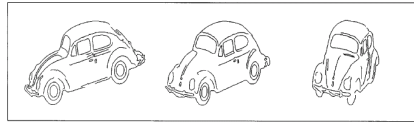
Henry model



Who am I?

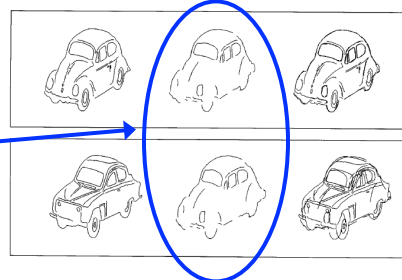
1-22

## Ullman & Basri



Object models: edge maps from multiple 2D views

Template: linear combination of locations of edge points from model views that "best fits" edge map from image of unknown object



1-23