Shape

Semester 2, 2021

Kris Ehinger

Shape in object recognition

How difficult would it be for a CNN to classify this type of image?

1 (very easy) – 5 (very difficult)

Geirhos, Rubisch, Michaelis, Bethge, Wichmann, & Brendel (2019)

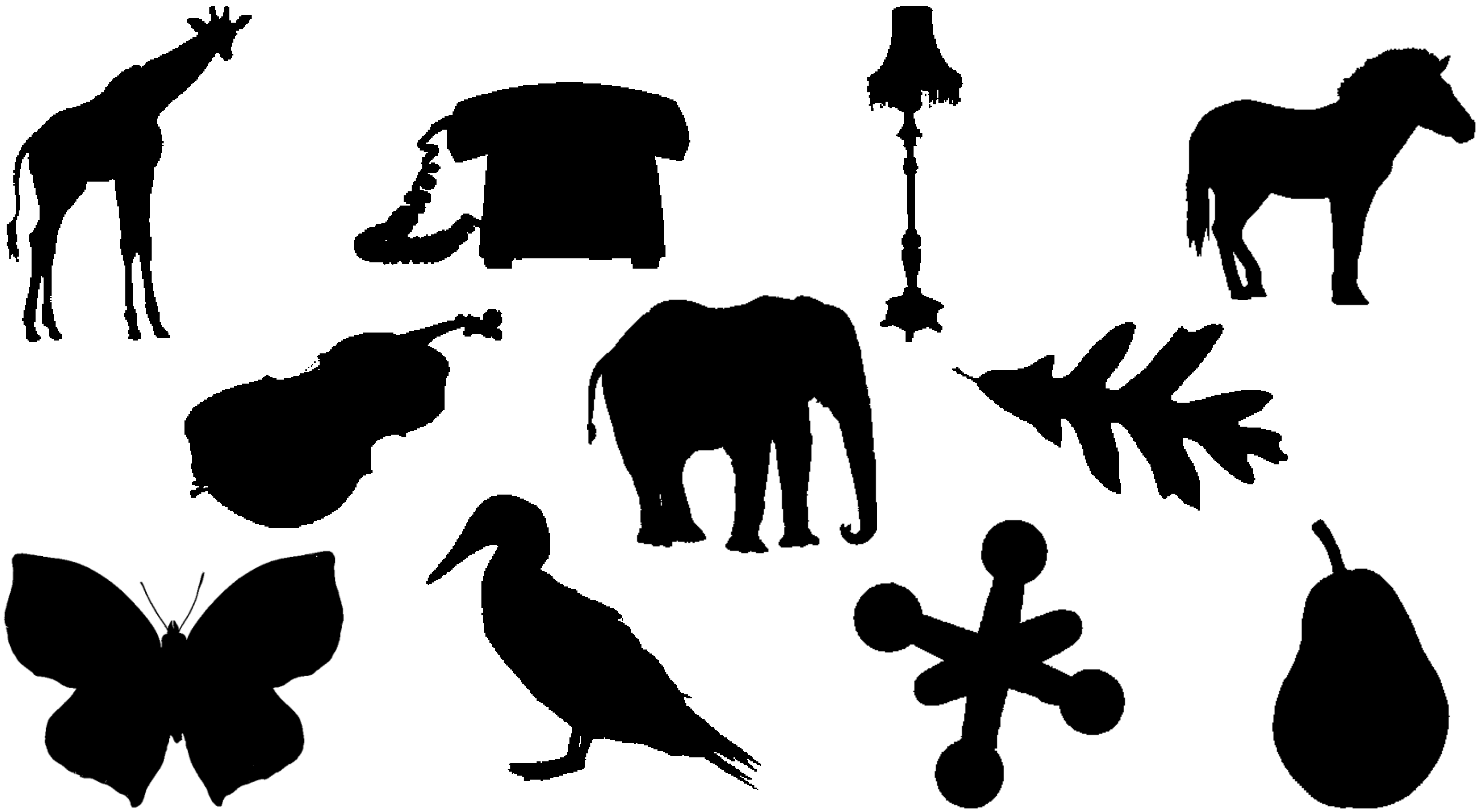
Outline

- Shape skeletons
- Contour representations
- Face models

Learning outcomes

- Explain what a topological skeleton is and how it is computed
- Explain a method for detecting shape contour (active contours/snakes)
- Explain methods for modelling face shape and their applications

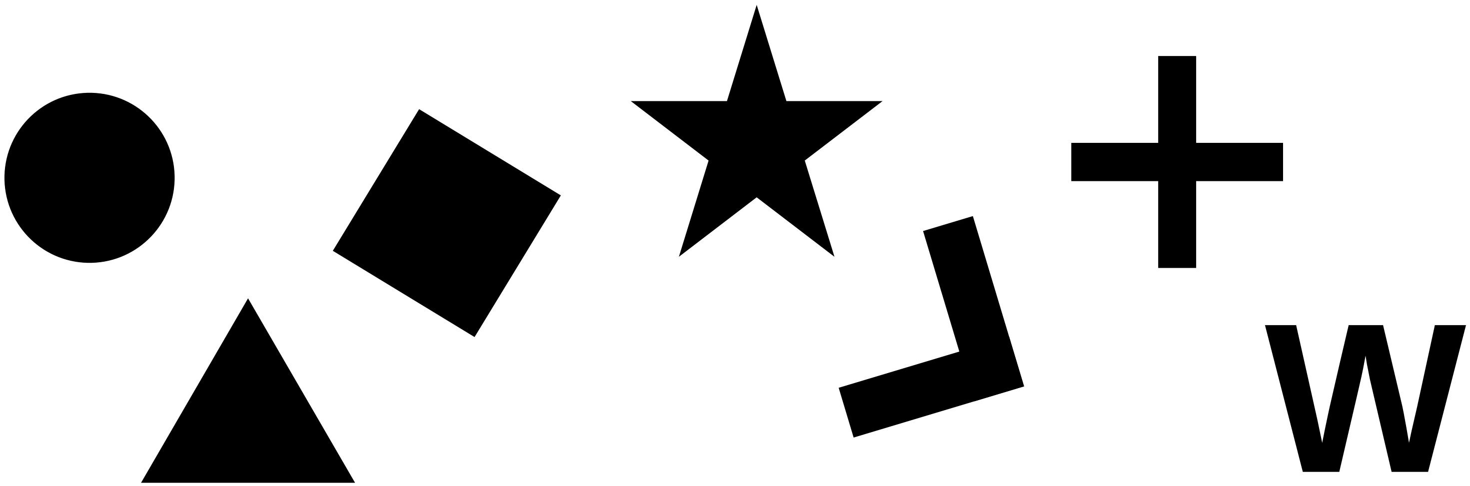
Shape



Images: Hemera

Models of shape

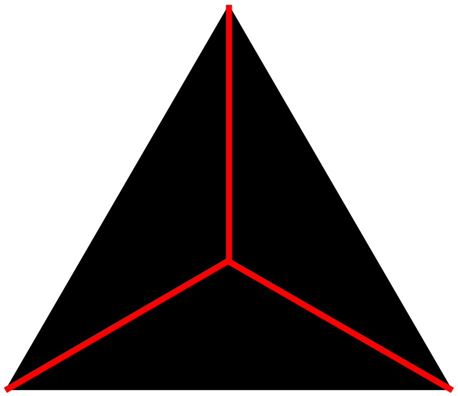
- Models of 2D shape are usually based on either:
  - The bounding contour of the shape (segments, angles)
  - The internal structure of the shape (branches)

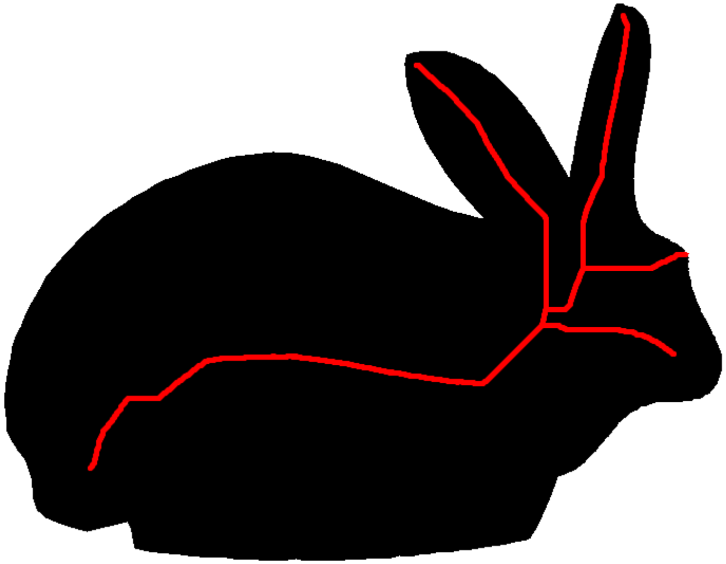


Shape skeletons

Shape skeleton

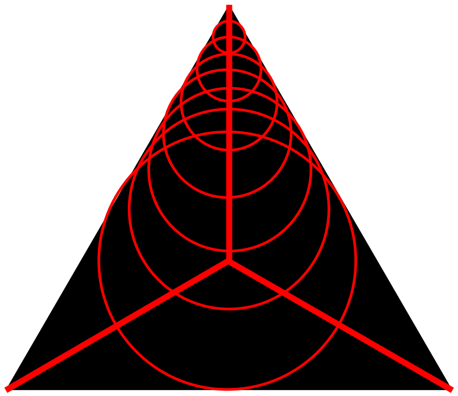
- Topological skeleton = thinnest possible version of a shape
- Formed of lines that are equidistant from the boundaries of the shape

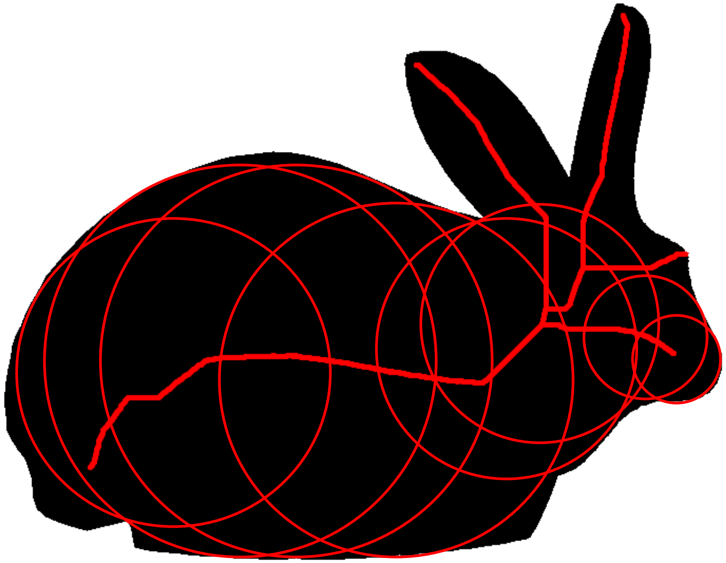




Geometrical description

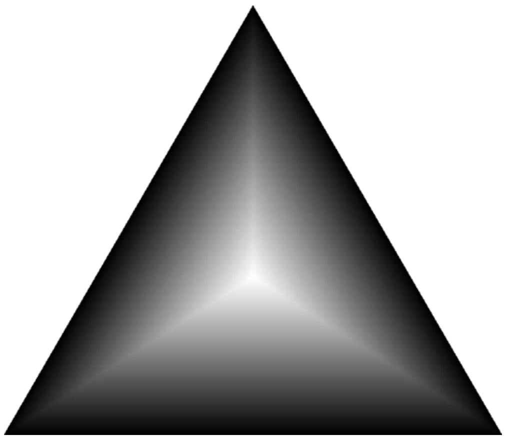
- The skeleton points are the centrepoints of the largest discs that can be fit inside the shape
- If the shape was painted with a circular brush (of variable radius), the skeleton would be the path of the brush

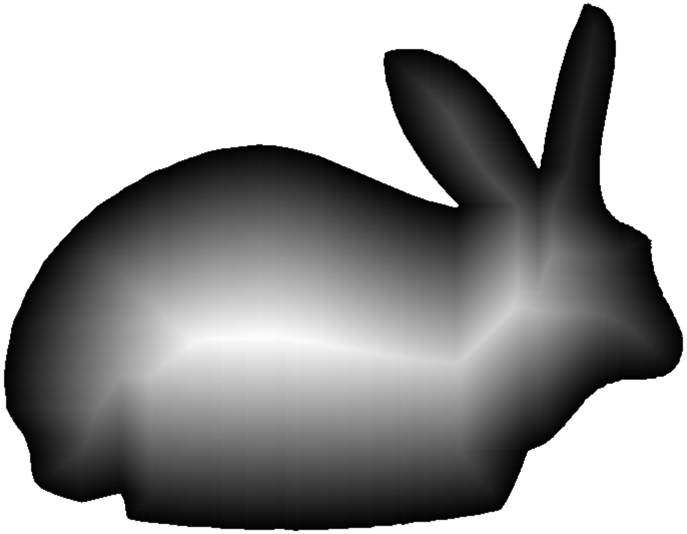




Skeletonisation algorithm

- Grassfire transform – algorithm for shrinking or thinning a shape
- For each pixel within the shape, compute distance to closest boundary; peaks in the distance map are the skeleton

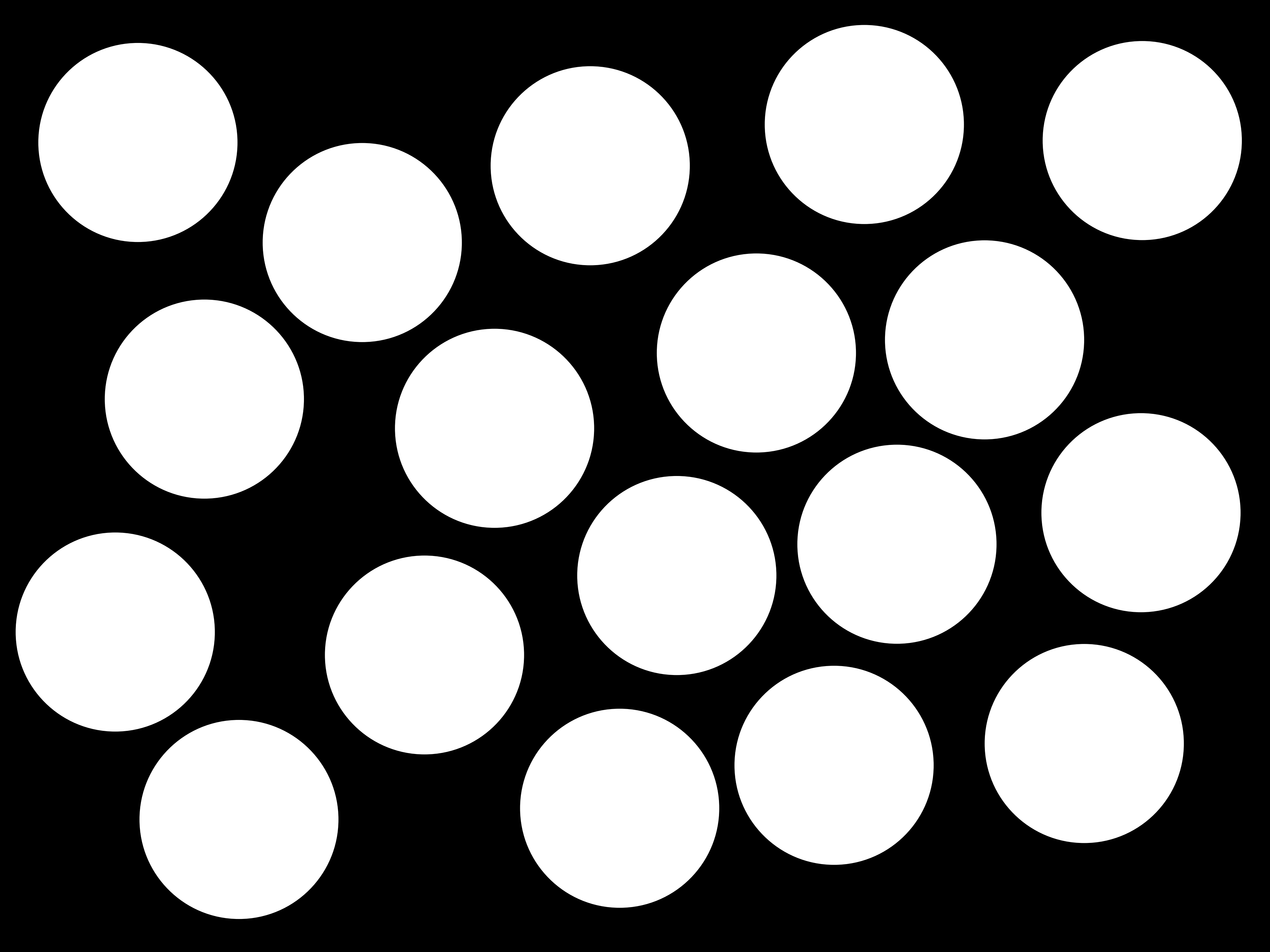




Brightness = Distance to closest boundary (Manhattan)

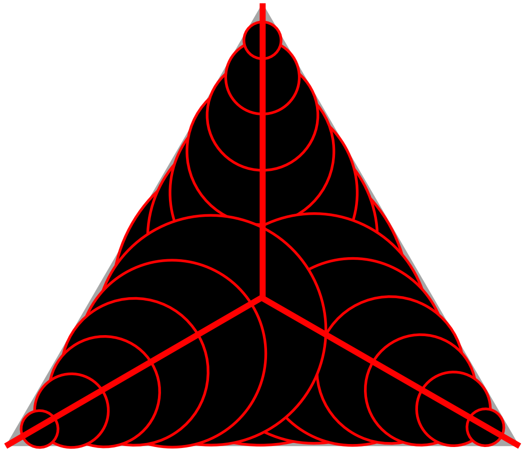
Practice: Shape skeleton

- What is the skeleton?



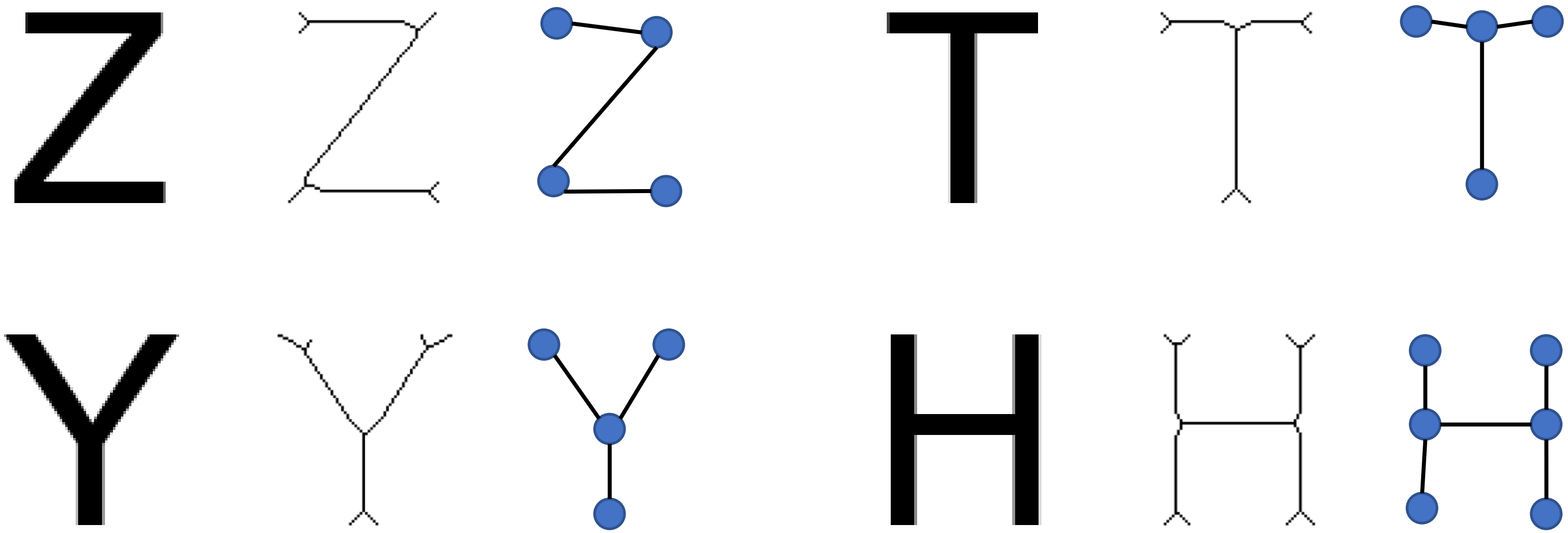
Skeleton representation

- Skeleton + distance to boundary at each skeleton pixel is a compact, invertible representation of shape
- To “inflate” skeleton, place a disc at each skeleton pixel (radius = distance to boundary at that pixel)



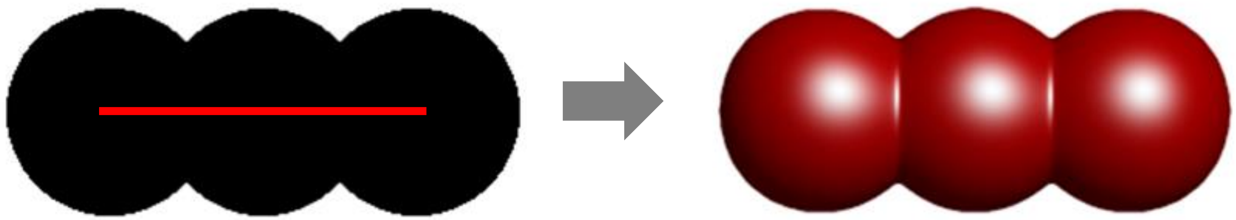
Application: Shape recognition

- Shape skeletons are easily converted to graphs
- Graph representation can be used for shape matching, pose recognition



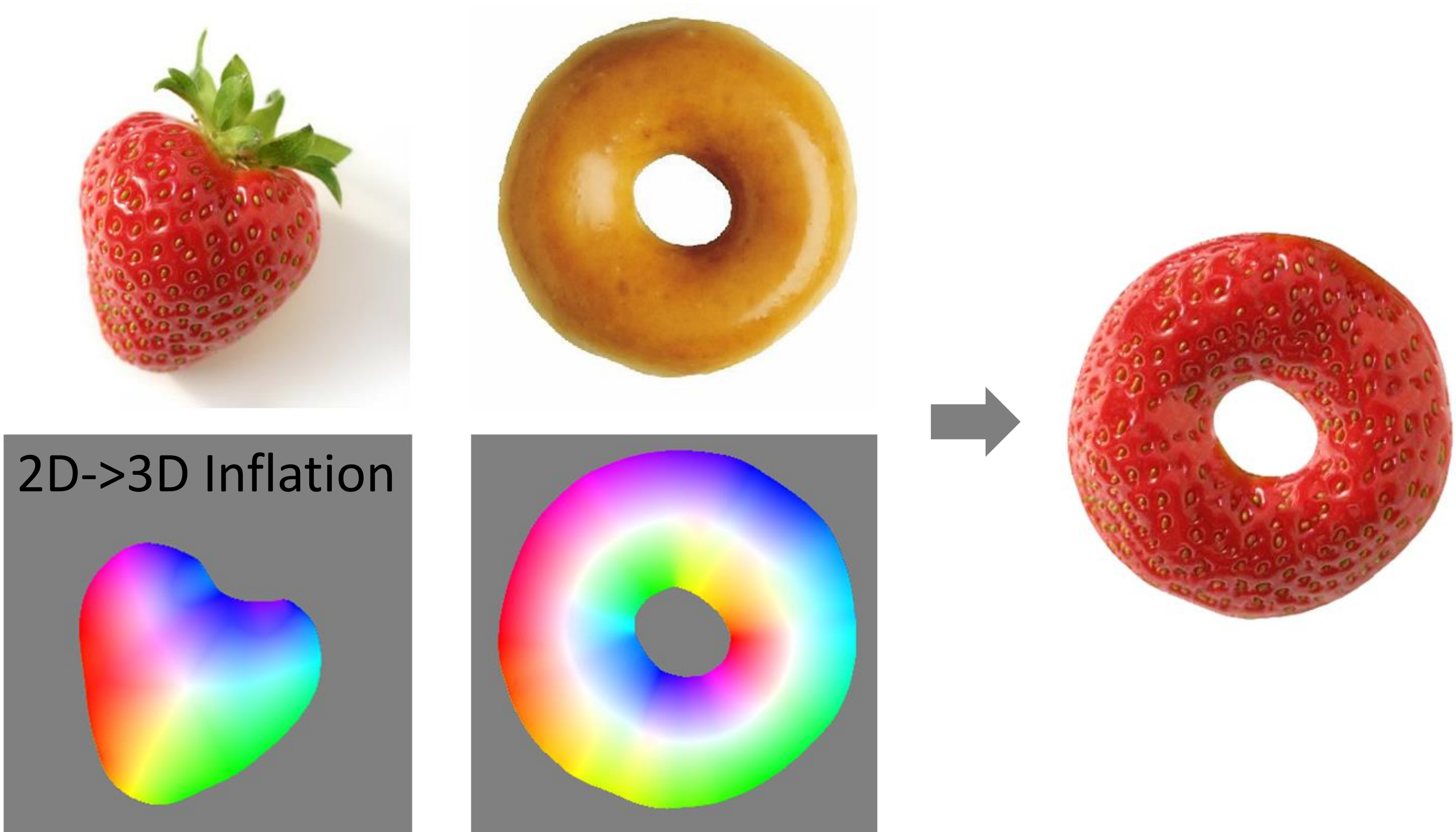
Application: 2D->3D

- Shape skeletons can also be used as the basis for a simple 3D model – just “inflate” with spheres instead of disks



Twarog, Tappen, & Adelson (2012)

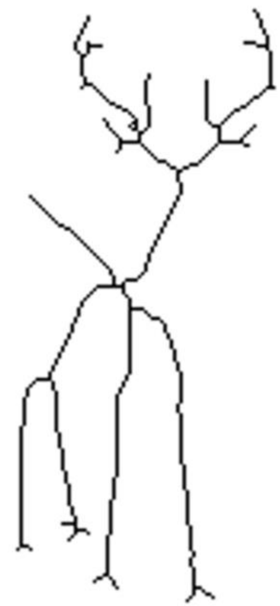
Application: 2D->3D



Twarog, Tappen, & Adelson (2012)

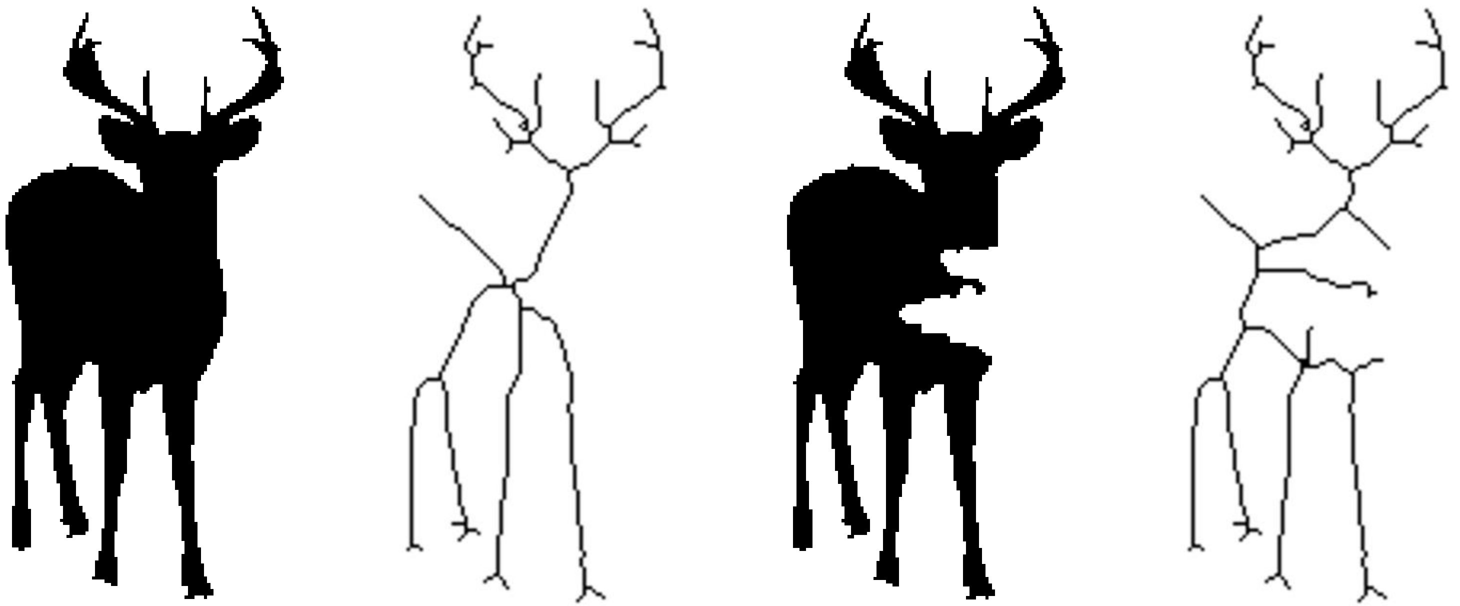
Drawbacks to skeletons

- Shape must be segmented from background



Drawbacks to skeletons

- Shape must be segmented from background
- Small changes in shape boundary produce large changes in skeleton



Summary

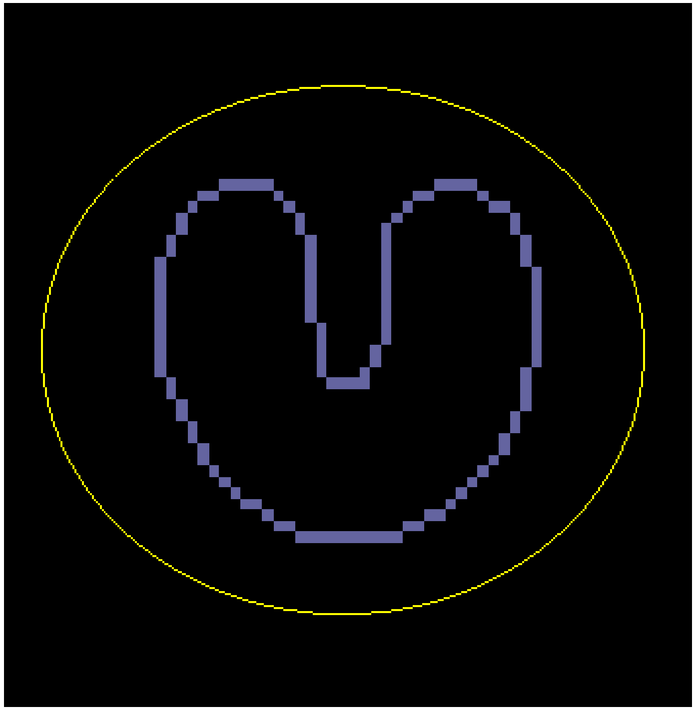
- Shape skeletons represent the internal structure of shapes
- Skeleton representations work well to model shapes that have a skeleton-like structure
  - Human/animal figures
  - Written characters
  - Paths/networks (e.g., city roads, blood vessels)

Contour representations

Automatic contour detection?

Active contours (snakes)

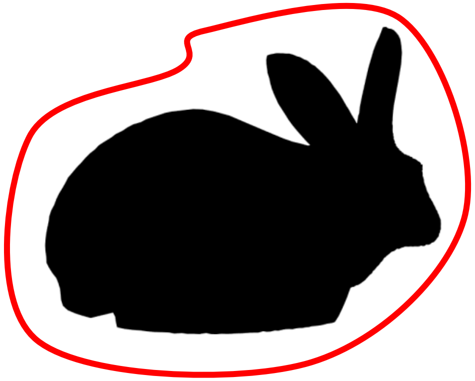
- Parametric model that fits itself to object boundary
- “Shrink wraps” around object to capture shape



Animation: C. Xu & J. Prince

Active contour algorithm

- Initialise contour outside object boundary
- On each step, allow each point on the contour to shift 1 pixel in any direction:
  - Shift to minimize a loss (or energy) function:
    $$E_{total} = \alpha E_{elasticity} + \beta E_{stiffness} + E_{edge}$$
- Repeat until loss does not change



Active contour algorithm

$$E_{total} = \alpha E_{elasticity} + \beta E_{stiffness} + E_{edge}$$

- $E_{elasticity}$ is based on contour length, penalises longer contours
- $E_{stiffness}$ is based on contour curvature, lowest for straight contour segments
- E_{edge} is based on image gradient at contour locations, lowest where image gradient is highest
- α, β are free parameters

Application: Segmentation

- Active contours are used for segmentation and tracking, particularly in medical image analysis

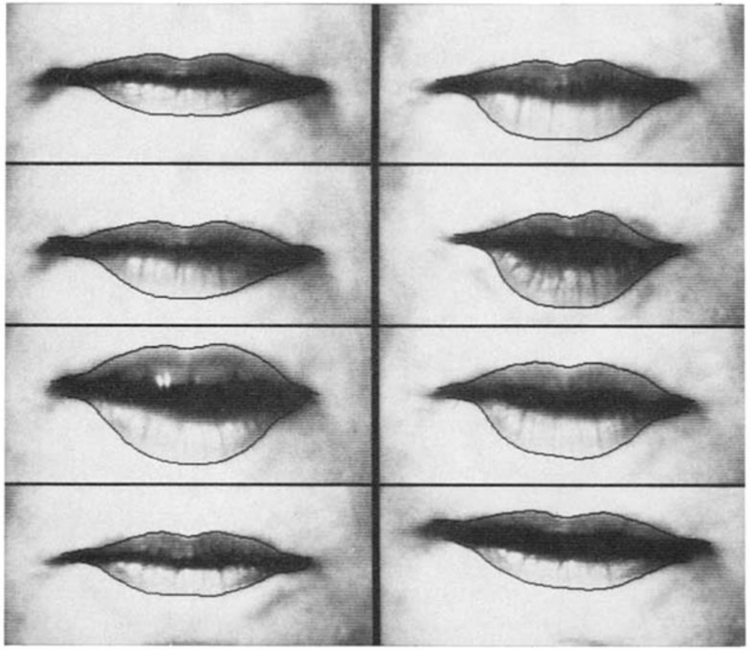
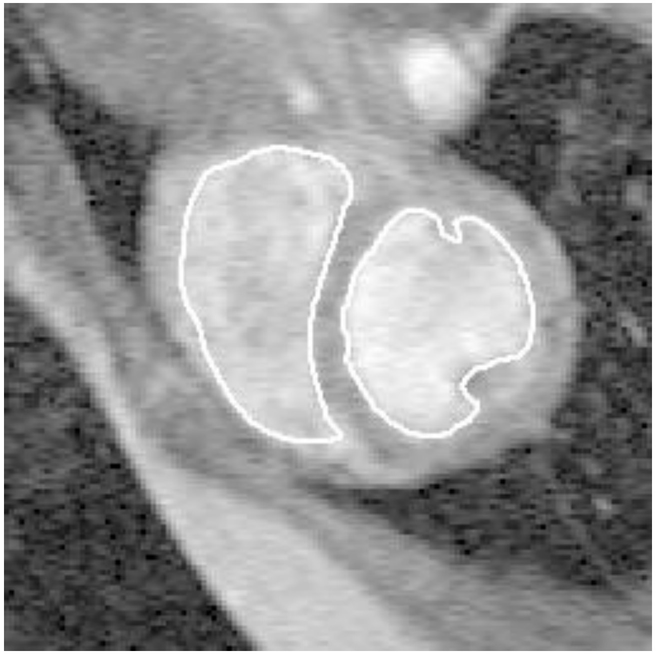


Image: Kass, Witkin, & Terzopoulos (1988)

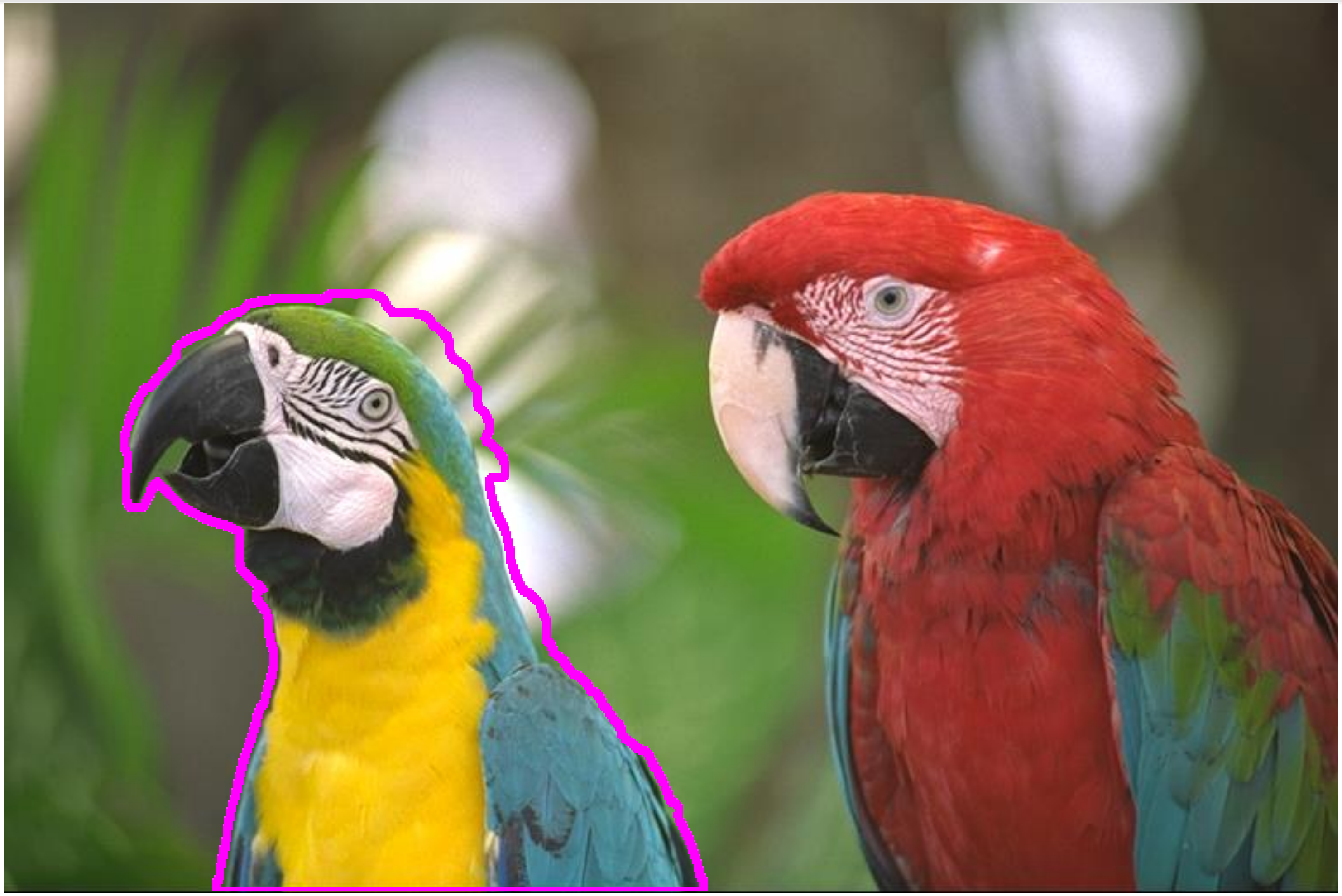


Animation: Y. Boykov

Drawbacks to active contours

- Requires initialisation (often from a human annotator)
- May not fit shape correctly
  - Trade-off between elasticity/smoothness and edge-matching – may fail to fit concavities in complex shapes
  - Difficult to detect shapes in clutter

Drawbacks to active contours



Summary

- Active contours fit a shape boundary
- Tries to find an optimal shape which is both well-fit to the edges and fairly simple (smooth, compact)
- Works well to segment objects with uniform appearance, moving objects

Face models

Face models

- It’s difficult to develop a general-purpose model of shape that can represent all possible shapes well
- However, it is possible to develop parametric models for particular classes of shape
- One very widely-studied class of shapes is the human face

Eigenfaces

- If faces are aligned, pixel luminance values are sufficient to capture face shape
- Simple pixel-based model: eigenfaces

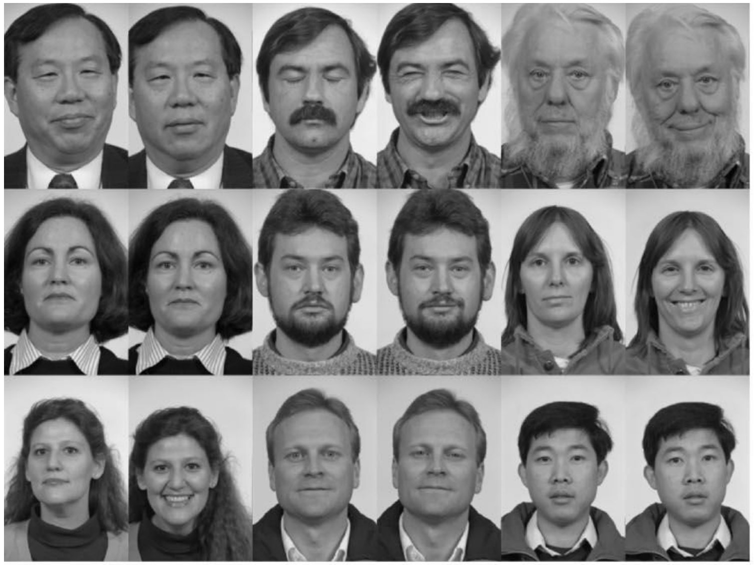
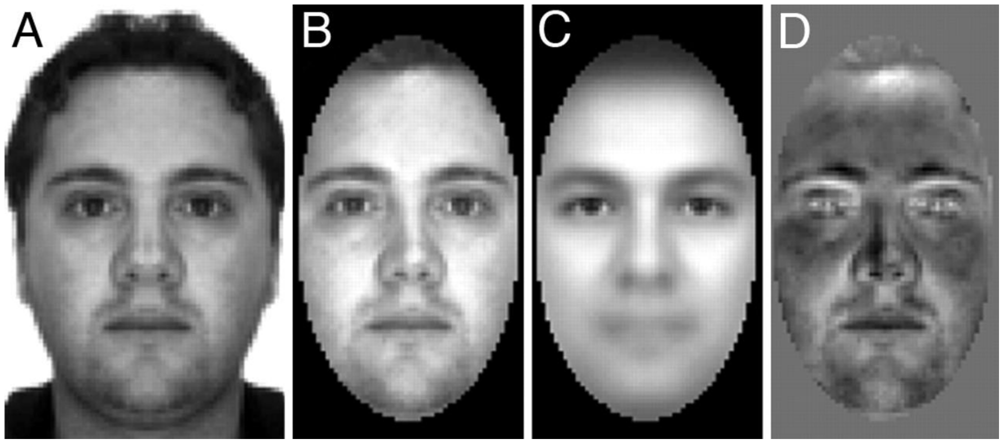


Image: Moghaddam, Jabara & Pentland (2000)

Eigenfaces algorithm

- Each face is represented as a vector to the mean face image
- Parameters of face shape are obtained from PCA of face vectors



Individual face

Mean of 685 faces

D = B-C
Vector from mean face to face B

Image: Sirovich & Meytlis (2009)

Eigenfaces algorithm

- Principal components of face vectors:

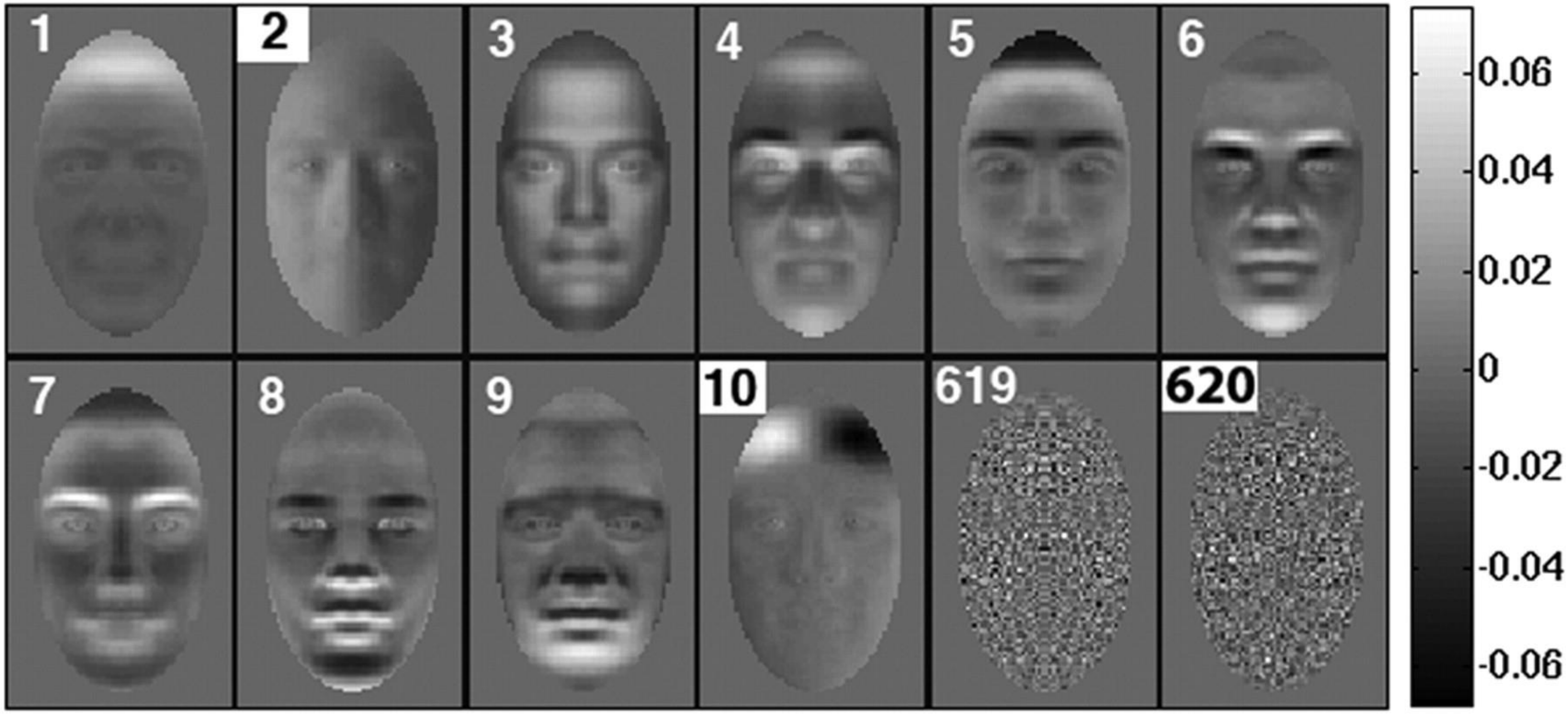


Image: Sirovich & Meytlis (2009)

Eigenfaces

- Problem: Usually we can’t assume faces appear in consistent alignment (or consistent lighting)!
- To model faces under real-world conditions, we need models that can consider shape/pose

Active appearance models

- Label corresponding landmark points in each image
- Warp images onto the mean shape to get shape-free texture

Image with labeled landmarks

Shape-free textures

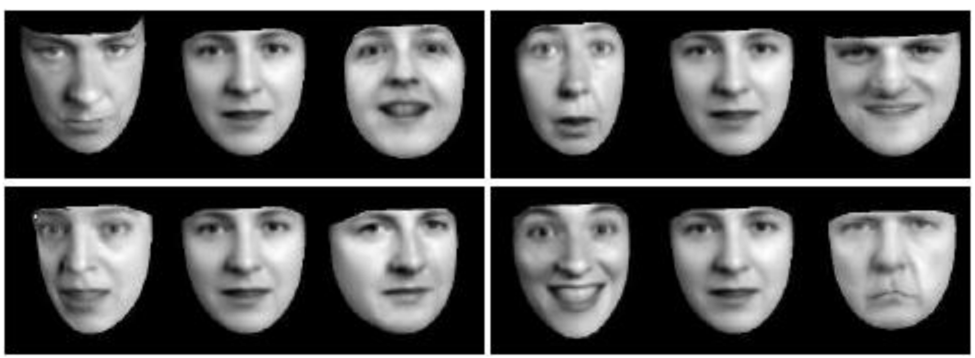
Cootes, Edwards, & Taylor (1998)

Active appearance models

- Obtain “shape,” “texture,” and “appearance” (shape+texture) parameters through PCA

Principal components 1-2 of shape (mean ±3 sd)

Principal components 1-2 of texture (mean ±3 sd)

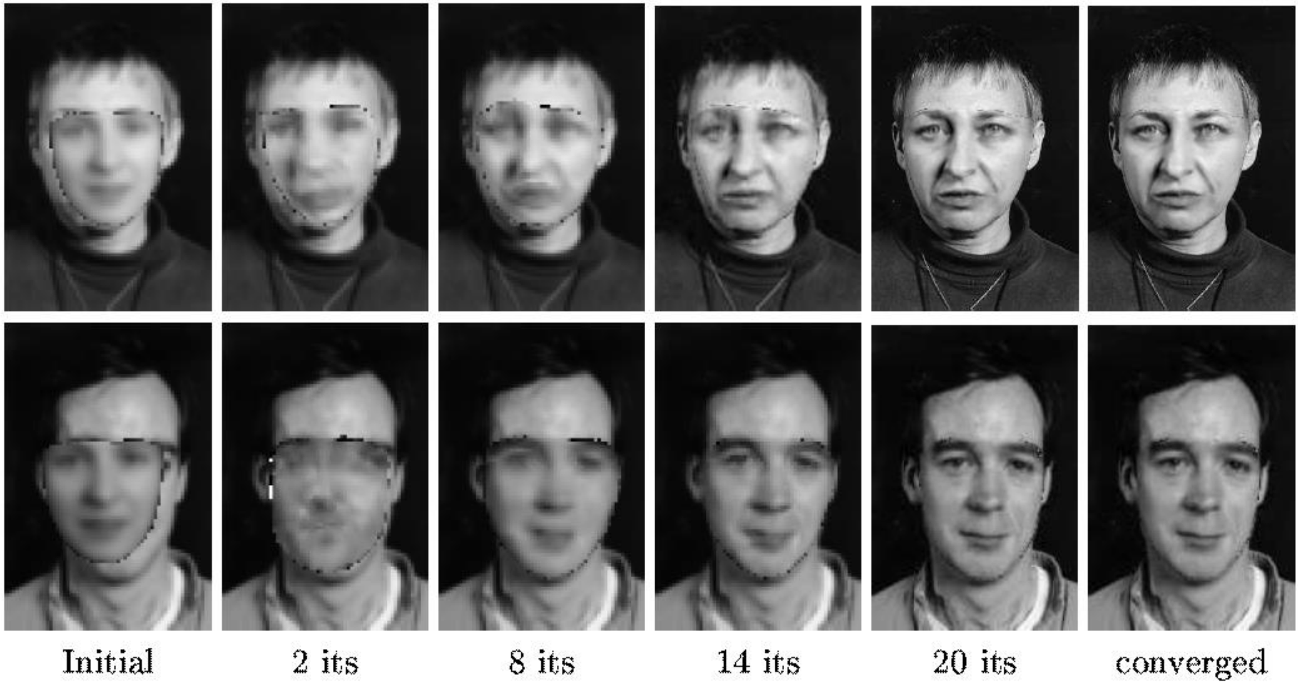


Principal components 1-4 of appearance (mean ±3 sd)

Cootes, Edwards, & Taylor (1998)

Active appearance models

- To fit the model to a new face, use gradient descent to minimize difference between model and image
- Applications: face synthesis, face segmentation



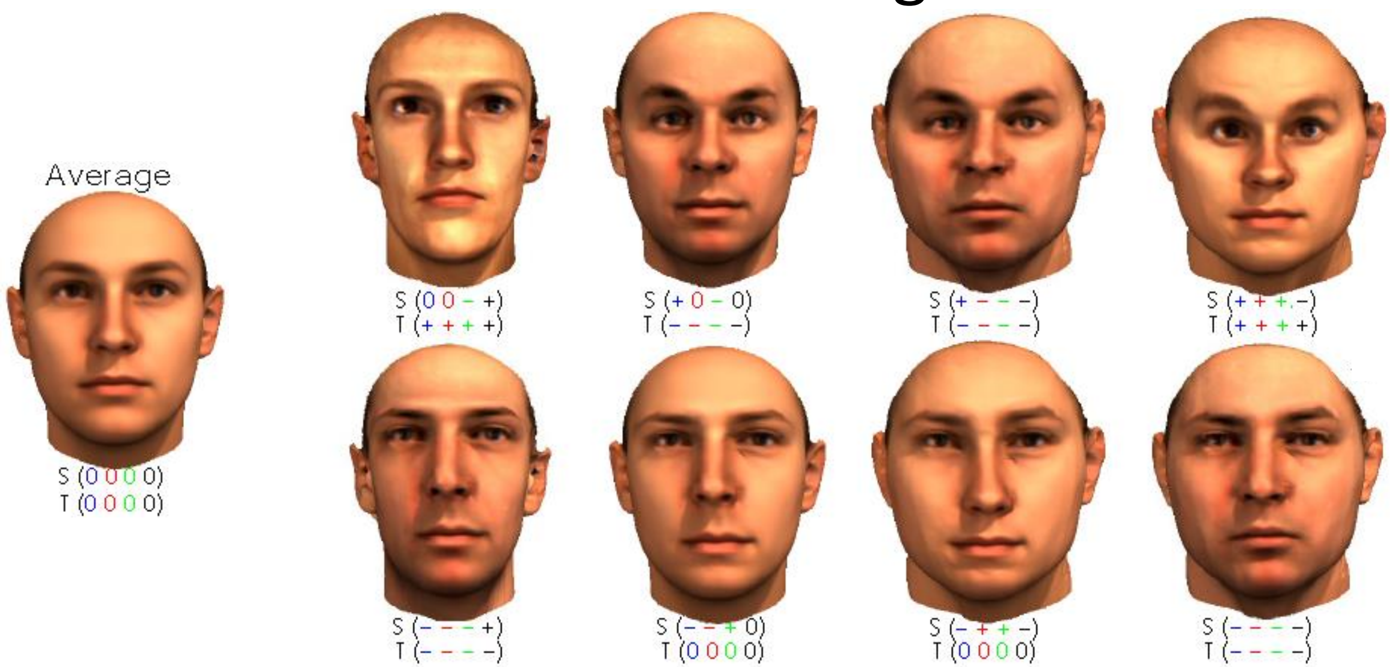
Cootes, Edwards, & Taylor (1998)

Active appearance models

- Active appearance models separate shape and texture
  - Allows alignment of facial features, even when images are not aligned
- Problem: Shape is represented using 2D contours
  - Can’t separate face shape vs. pose
  - Can’t separate surface colour vs. lighting

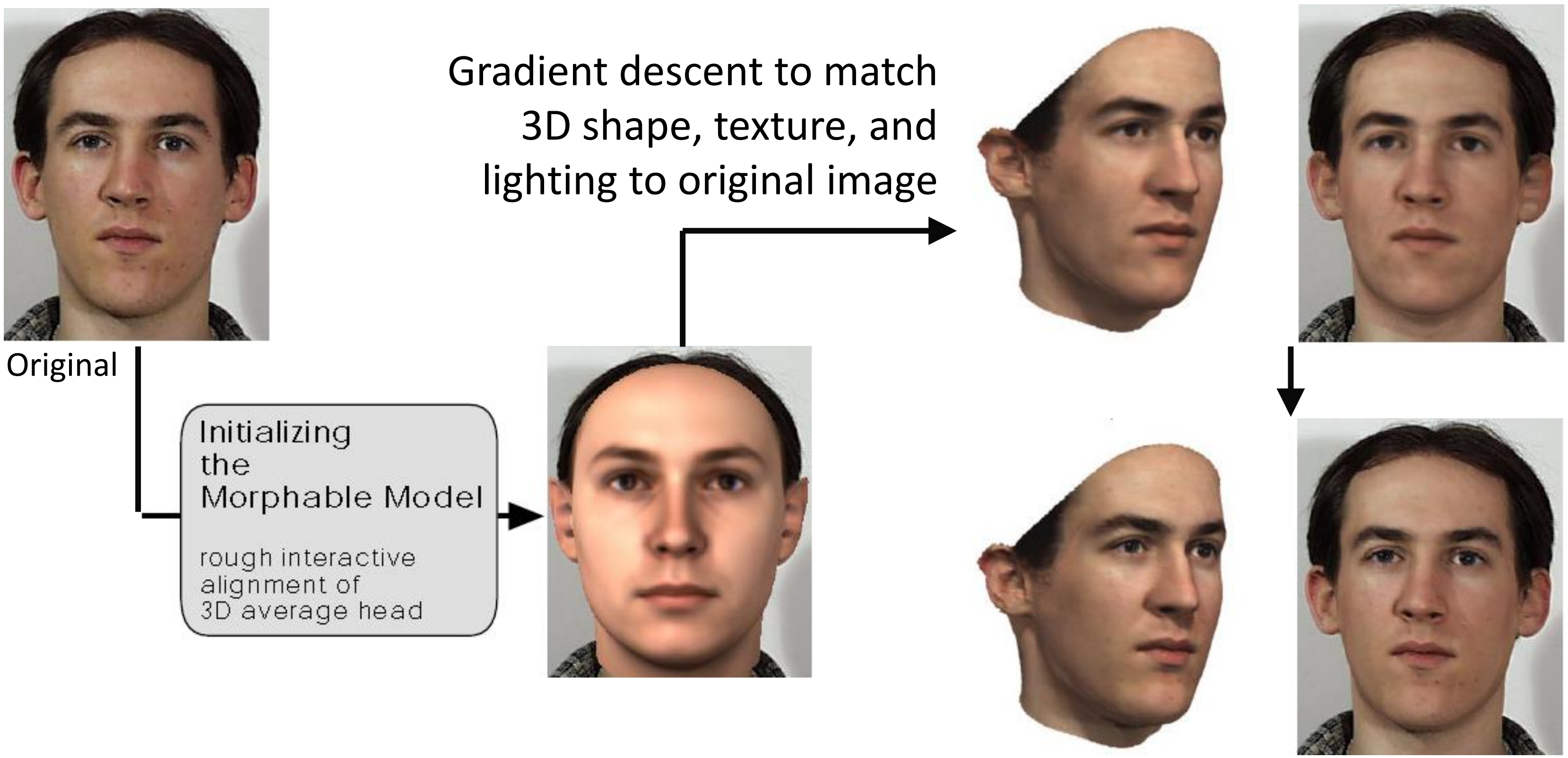
3D face models

- 3D version of active appearance model: morphable 3D mesh + texture map
- Parameters based on PCA of a large 3D dataset



Blanz & Vetter (1999)

3D face matching

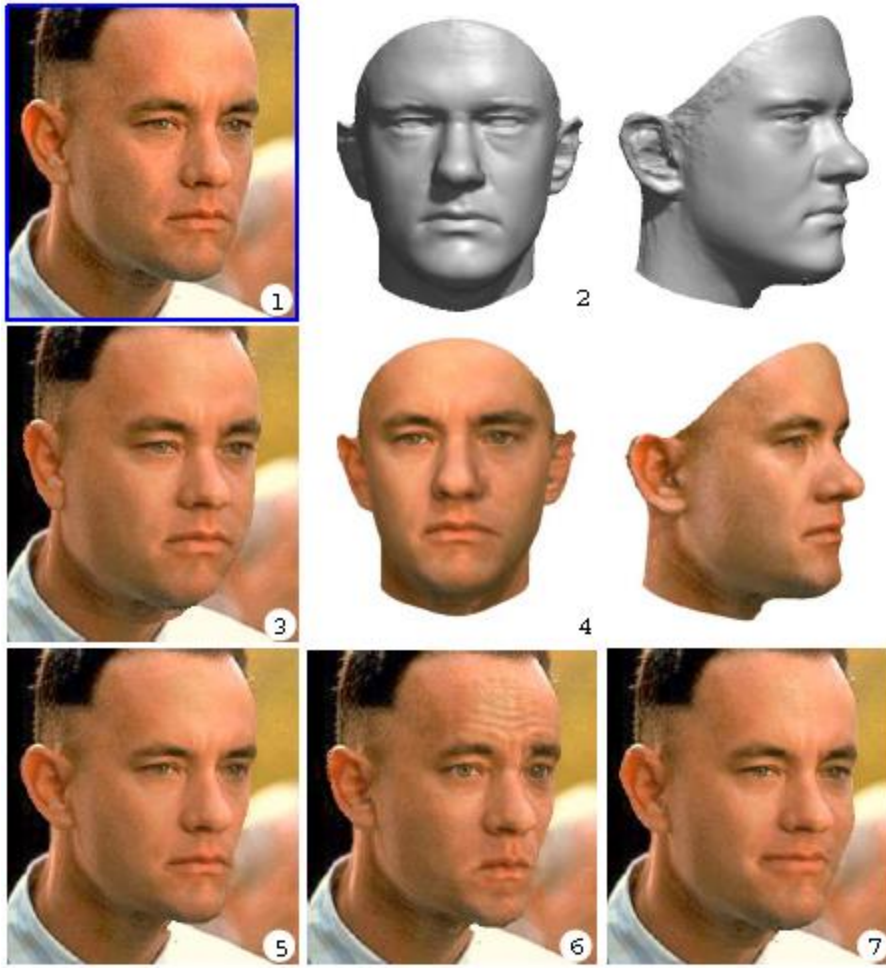


Extract texture from original, using
3D model to correct for lighting

Blanz & Vetter (1999)

3D face model results

Original



Shape reconstruction

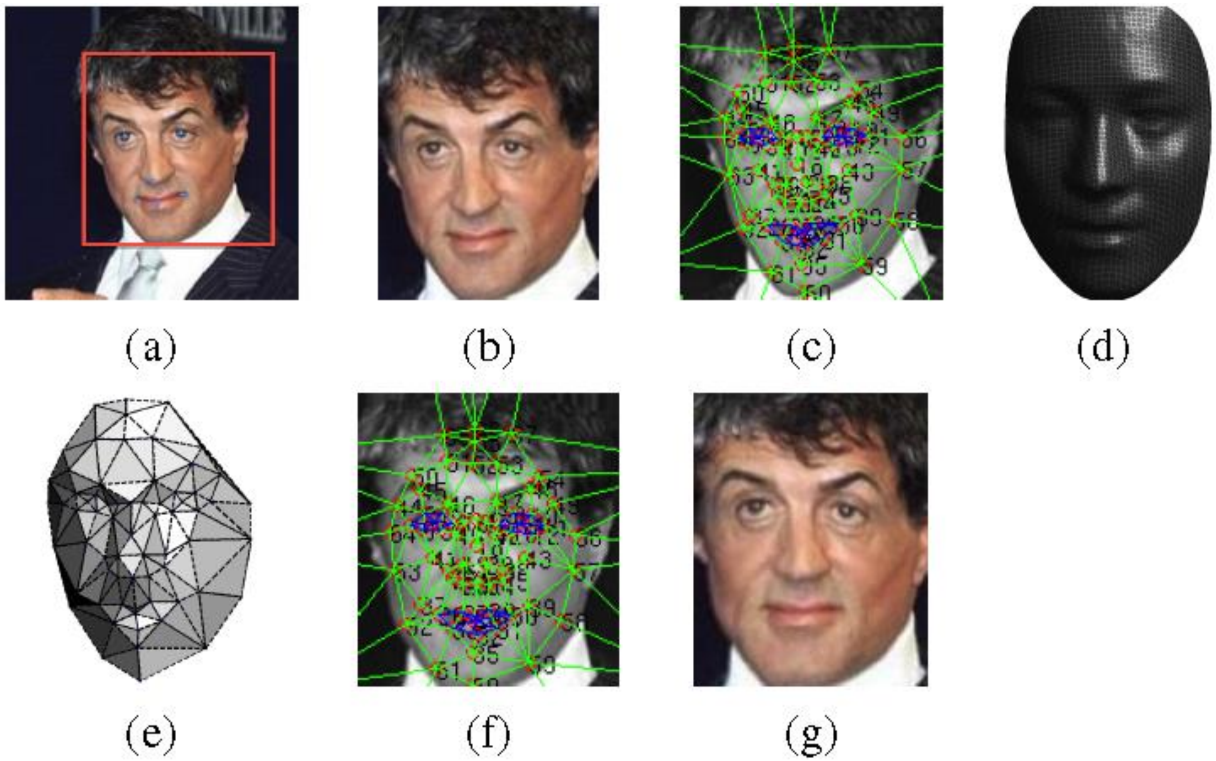
Texture reconstruction

Modified 3D model

Blanz & Vetter (1999)

Application: Facial recognition

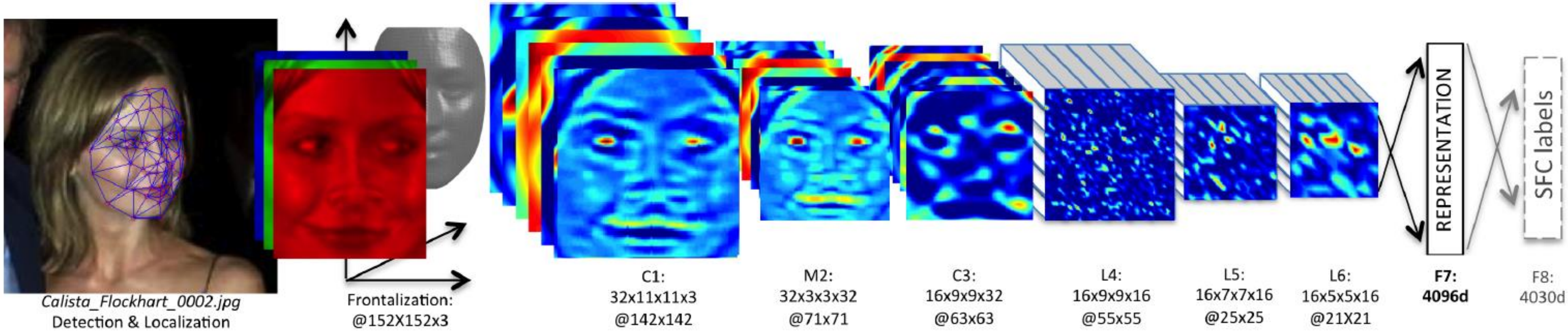
- Most recognition algorithms use a shape model to align faces as a first step



Taigman, Yang, Ranzato, & Wolf (2014)

Application: Facial recognition

- Once faces are aligned, a standard CNN pipeline can be trained for face recognition
- Why is alignment critical for CNNs?



Taigman, Yang, Ranzato, & Wolf (2014)

Summary

- Face models are one of the main applications of shape representation in computer vision
- Current state-of-the-art algorithms are based on 3D face models
- Applications:
    - Facial recognition
    - Computer graphics (movie CGI, video games)
    - Zoom filters :)

Summary

- Although shape is not required for category-level object recognition, shape is important for fine-grained recognition and separating out effects of lighting and pose
- 2D shapes are typically represented in terms of skeleton structure or bounding contours
- 3D shape models have been developed for specific recognition problems (mainly faces and body pose)