History of Speech Recognition

1965-90: looking for features (spectrum, LPC, cepstrum, cochlear feat.)

1965-75: isolated-word global template matching (nearest neighbors)

1975-85: deformable template matching (nearest neighbors)

1980-90: structural methods / expert systems (no learning, failure)

1985-90: HMMs (lots of learning, generative models, non-convex!)

1990-95: global generative learning (sentence-level HMMs)

1990-95: word-level discriminative learning (HMMs, non-convex!)
   - mixtures of Gaussians, neural nets

1995-00: sentence-level discriminative learning (HMMs, non-convex)

what made it work:
   - lots of data + huge models, training the segmenter, generative + discriminative training, non-convex/non-linear learning
Panel on Shape Representation

- Yann LeCun: recognition architectures and representation learning.
- Martial Hebert: Shape Representation, the historical perspective
- Jean Ponce: Feature Representations, an overview
History of Handwriting Recognition

1965-90: looking for features
- edges, projections, chain code, Zernicke moments, Fourier, Haar, Hadamard, Hough,......

1965-75: classifiers for isolated characters
- nearest neighbors, linear classifiers

1975-85: structural methods (no learning, failure)

1985-95: learning the features (lots of learning, non convex!)
- neural nets, convolutional nets

1990-00: global learning (lots of learning, context, non convex!)
- word-level discriminative learning (d-HMM, graph transformer nets)

since then, people keep re-inventing the same thing

what made it work:
- lots of data, training the segmenter, integrated discriminative training, learning the features (deep learning), non-convex/non-lin.
History of Image Recognition

1965-2008: looking for features
- edges, contours, Hog, Sift, Shape Context,……

1965-08: linear classifiers (Perceptrons!), nearest neighbor classifiers

1975-95: structural methods (no learning, failure)

1993-01: learning the features for face detection (learning, non convex!)
- neural nets, convolutional nets, boosted cascades.

1990-00: structured output models (lots of learning, context, non convex!)
- word-level discriminative learning (d-HMM, graph transformer nets)

what made it work (so far):
- learning, discriminative learning, designing the right features

what's missing:
- learning the features, integrated segmenter, unsupervised/supervised learning
The Future of Image Recognition

- We are still looking for the right features
  - we should try to learn them
  - …but so far, feature learning for object recognition has not worked as well as for handwriting recognition
  - do we have the right learning algorithms (deep learning!)

- We are still stuck with “linear” learning and/or nearest neighbors
  - let's move beyond SVMs and K-NN
  - non-linear/non-convex learning was essential for speech and handwriting: mixtures of Gaussians, convolutional nets…..

- We are just getting started with integrated (global) training
  - training the segmenter was crucial to making speech and handwriting recognition systems work.
  - segmentation/pose are treated as latent variables.
  - This kind of approaches will be crucial for dealing with invariance
  - They will be essential for compound objects with movable parts
    - (see Ramanan/Felzenswalb/McAllester)
Do we have the right architecture?

Speech and Handwriting have settled on an architecture

Image recognitions systems are just about to settle on an architecture

- 04: interest points -> global spatial pooling -> classification
- 05: interest points -> local spatial pooling -> elastic template matching
- 06: local feature detectors -> local spatial pooling -> classification

But these models are “shallow”

- The mammalian visual cortex is deeper
- multiple stages of:
  - local feature detectors (simple cells) -> local pooling (complex cells)
  - Convolutional nets, HMAX……

We will be converging towards the “Multistage Hubel-Wiesel Architecture”

- Hierarchy of increasingly invariant features
- We will have to learn the features
- We can design the first layer, but not the next layers!
Building a complete artificial vision system:
- Stack multiple stages of simple cells / complex cells layers
- Higher stages compute more global, more invariant features
- Stick a classification layer on top
- [Fukushima 1971-1982]
  - neocognitron
- [LeCun 1988-2007]
  - convolutional net
- [Poggio 2002-2006]
  - HMAX
- [Ullman 2002-2006]
  - fragment hierarchy
- [Lowe 2006]
  - HMAX
Supervised Convolutional Nets learn well with lots of data

Supervised Convolutional nets work very well for:

- handwriting recognition (winner on MNIST)
- face detection
- object recognition with few classes and lots of training samples
Learning the Features?

**Decoder:**
- Linear

**Optional encoders of different types:**
- None
- Linear
- Linear-Sigmoid-Scaling
- Linear-Sigmoid-Linear

**Optional sparsity penalty**
- None, L1, Log Student-T

**Feature Vector Z**
- continuous

\[
\bar{Z}_Y = \text{argmin}_Z E(Y, Z, W)
\]

\[
E(Y, W) = \min_Z E(Y, Z, W)
\]
Learning the Right Features?
Learning the Features

96 filters on 9x9 patches trained with PBP
- with Linear-Sigmoid-Gain Encoder

Recognition:
- Normalized_Image -> Learned_Filters -> Rectification -> Local_Normalization -> Spatial_Pooling -> PCA -> Linear_Classifier
- What is the effect of rectification and normalization?

weights: -0.9275 - 0.8688
Caltech-101 Recognition Rate

- [96_Filters->Rectification]->Pooling->PCA->Linear_Classifier
  - [Filters->Sigmoid] 16%
  - [Filters->Absolute_Value] 51%
  - [Local_Norm->Filters->Absolute_Value] 56%
  - [Local_Norm->Filters->Absolute_Value->Local_Norm] 58%

- Multi-Scale Filters->Rectification->Pooling->PCA->Linear_Classifier
  - LN->Gabor_Filters->Rectif->LN (Pinto&diCarlo 08) 59%

- Unsupervised Convolutional Net
  - Filt->Sigm->Pooling->Filt->Sigm->Pooling->Classifier 54%

- Supervised Convolutional Net
  - Filt->Sigm->Pooling->Filt->Sigm->Pooling->Classifier 20%
Martial Hebert

- Context and scene interpretation
- Background knowledge
- Parts
- Geometry
- Shape and relations
surface of the cylinder. It predicts that the length of the ribbon in the image will in fact be:

\[-2.42 \times CYL\_LENGTH \times \cos(-TILT)\]

\[
\frac{\text{CYLINDER\_CAMZ}}{\text{CYLINDER\_CAMZ}}
\]

where 2.42 is the focal ratio of the camera and CYLINDER\_CAMZ is an internal quantifier generated by the prediction module.

Both of the above approaches are used to generate back constraints to ensure coverage of all the relevant quantifiers. They are:

\[m_h \geq -2.096 \times CYL\_LENGTH \times (1/CYLINDER\_CAMZ)\]

\[m_i \leq -2.338 \times CYL\_LENGTH \times (1/CYLINDER\_CAMZ)\]

\[-TILT \leq -\arccos(\sup(-0.413 \times m_h \times CYLINDER\_CAMZ \times (1/CYL\_LENGTH)))\]

\[-TILT \geq -\arccos(\inf(-0.413 \times m_i \times CYLINDER\_CAMZ \times (1/CYL\_LENGTH)))\]
Thomas O. Binford
Stanford University Artificial Intelligence Project

We describe a formal representation for a class of primitive
three-dimensional shapes. These primitive representations are
combined into compound articulated representations of familiar
objects. With regard to primitive representations, we discuss
only the formalism, not the inference of such descriptions from
visual data for complex scenes. The primary design criteria for a
representation are the ease with which we can recognize an object as
essentially similar to another we have seen before, or the ease with
which we can identify that objects with distinct differences have
important similarities (a child and an adult, or a man and a woman).
This is one basis for generalization. A representation is intended
to express low-level knowledge about shape, i.e., class knowledge
about familiar shapes, and to serve as a basis for approximation of
shape, and conjecture about missing information, for example, the
hidden half of objects. The primary criterion is not the simplicity
(a) Bottom-up process

(b) Top-down process

(c) Result
### Final Region Interpretations

<table>
<thead>
<tr>
<th>Interpretations</th>
<th>Regions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sky</td>
<td>1, 2, 3, 4</td>
</tr>
<tr>
<td>Mountain</td>
<td>5, 6, 7, 8</td>
</tr>
<tr>
<td>Sea</td>
<td>9, 10, 11, 12</td>
</tr>
<tr>
<td>Ground</td>
<td>13</td>
</tr>
<tr>
<td>Rock</td>
<td>14, 15</td>
</tr>
<tr>
<td>Tree (Crown)</td>
<td>16</td>
</tr>
<tr>
<td>Tree (Bark)</td>
<td>17, 18, 19, 20</td>
</tr>
</tbody>
</table>

**Fig. 4** Final semantic partitioning of landscape scene

### Final Region Interpretations

<table>
<thead>
<tr>
<th>Interpretations</th>
<th>Regions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Door</td>
<td>1</td>
</tr>
<tr>
<td>Wall</td>
<td>2</td>
</tr>
<tr>
<td>Floor</td>
<td>3</td>
</tr>
<tr>
<td>Picture</td>
<td>4</td>
</tr>
<tr>
<td>Tabletop</td>
<td>5</td>
</tr>
<tr>
<td>Chairseat</td>
<td>6</td>
</tr>
<tr>
<td>Chairback</td>
<td>7</td>
</tr>
<tr>
<td>Waste Basket</td>
<td>8</td>
</tr>
</tbody>
</table>

**Fig. 5** Final semantic partitioning of SRI office scene
(A-5) Output of region grower based on semantics. (Melting weakest boundary first where boundary strength is computed using the semantic world model)

(A-7) Final grouping of regions based on the interpretation assigned to them by the world model. Regions whose meaning was assigned with confidence less than 10 are not mergable. They occur usually on the real boundary between two regions.

(B-2) Output of the non-semantic weakest boundary melted first region grower.

(B-3) Output of the semantic based region grower.

(B-1) Original picture.

(B-4) Result of grouping regions by their assigned meaning. Taking only regions which were assigned meaning with confidence over 10 to be mergable.
Figure 3 A set of intrinsic images derived from a single monochrome intensity image. The images are depicted as line drawings, but, in fact, would contain values at every point. The solid lines in the intrinsic images represent discontinuities in the scene characteristic; the dashed lines represent discontinuities in its derivative.
Feature Representations

Jean Ponce
Color histograms
(Swain & Ballard’91)
Local jets (Florack'93)
Spin images (J&H'99)
Sift (Lowe'99)
Shape contexts (B&M'95)
Local jets (Florack’93)
Spin images (J&H’99)
Sift (Lowe’99)
Shape contexts (B&M’95)
Texton histograms (?)
Gist (O&T’05)
Spatial pyramids (LSP’06)
Hog (D&T’06)
Phog (B&Z’07)
Convolutional nets (LC’70)
Locally orderless structure of images (K&vD'99)
Felzwenszalb, McAllester, Ramanan, 2007
Kushal, Schmid, Ponce, CVPR’07
Boiman, Schechtman, Irani, CVPR'08
(73% classification rate on Caltech 101)
Boiman, Schechtman, Irani, CVPR’08
(73% classification rate on Caltech 101)
Boiman, Schechtman, Irani, CVPR’08
(73% classification rate on Caltech 101)
Essentially the modified Hausdorff distance for object matching of Dubuisson & Jain’95 (see also Farach-Colton & Indyk’99 for ANNs).