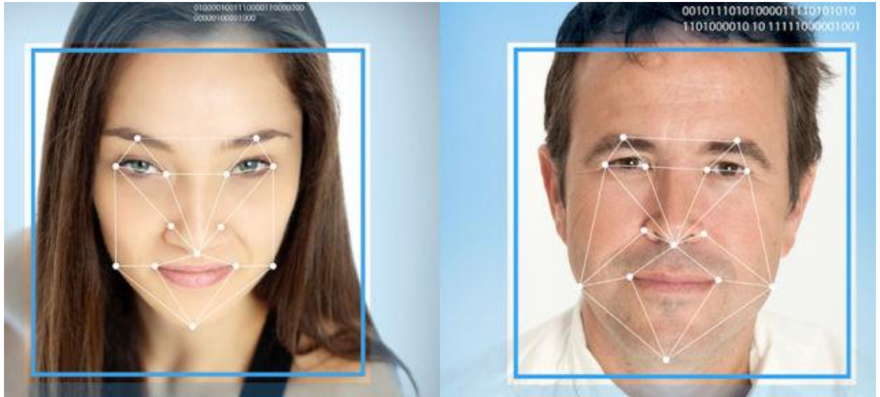
Learned versus Hand-Designed Feature Representations for 3d Agglomeration

John A. Bogovic, Gary B. Huang & Viren Jain
As presented by Mary Yen

Terms

- **Machine learning** is a set of algorithms that classify data input.
  - eg: We want to know whether a set of facial images are male or female.
- Algorithms train **classifiers** on a **training set** to identify data input. Classifiers need **quantifiable** data to work. Can test classifier's effectiveness on **testing set.**
  - eg: Assign numerical values to facial shape by edge curvature. Classifiers can ident facial image is male or female depending on whether the image's edge values are c female edge values.

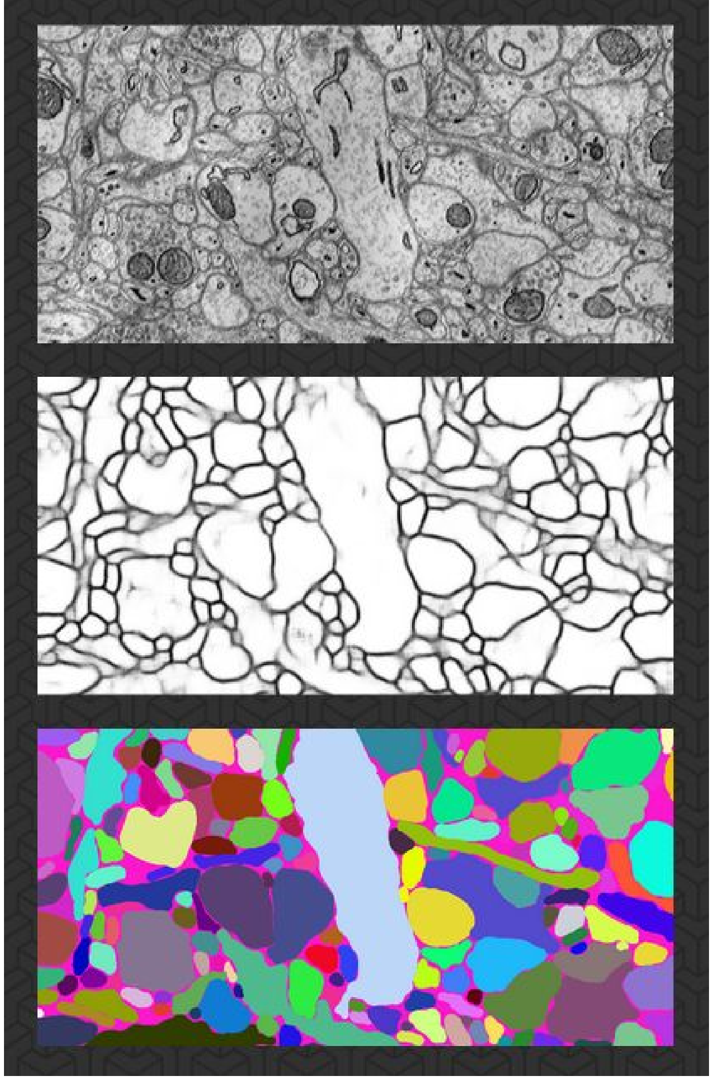


Opportunity

- Automatic machine learning methods may be more efficient than manually specified methods.
    - Doesn’t require domain expertise
    - potentially yields a much larger set of features for a classifier.
    - May find algorithms or features that are more finely tuned for the particular proble and thus lead to improved accuracy.
    - can be easily adapted to new types of data
- Is purely automatic or hand designed better? Is there a balance between the two?

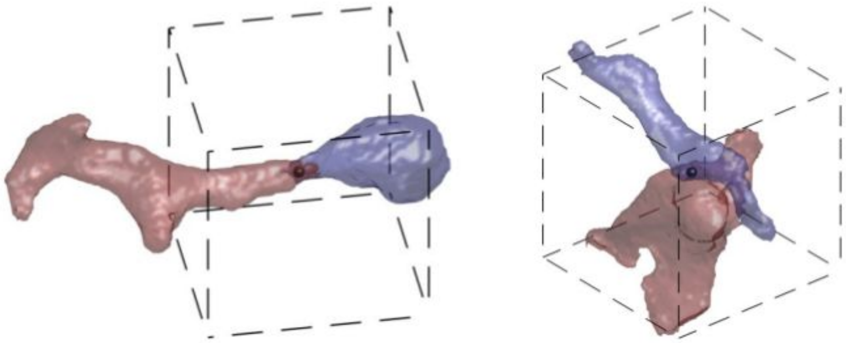
Challenge

- Specifying 3D features seems easier than using hand-designed representations for more low-level data (such as raw image patches).
- describing a neuron fragment in terms of quantities such as curvature, volume, and orientation seems natural.
- Can easily use this data in learning algorithms whereas hand-designed data takes longer to collect & needs to be quantified.



Challenge

- Is this intuitive appeal is a good justification for feature representation in specific tasks such as neuron fragment agglomeration?



- Intuitively makes sense, but is it effective in practice?

Action

Compared the performance of the following:

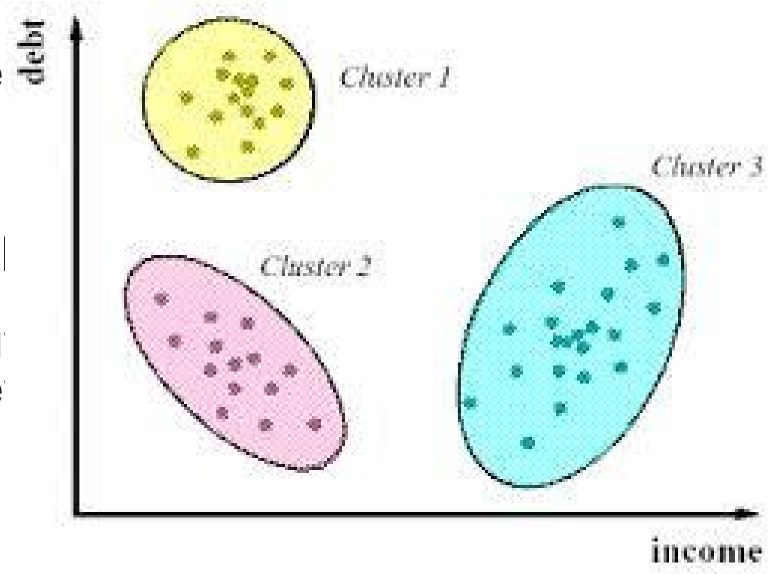
1. A large set of diverse hand-designed 3d shape descriptors.

2. An end-to-end supervised learning approach for deriving 3d feature descriptors.

3. An unsupervised learning approach for deriving 3d feature descriptors.

Supervised vs. Unsupervised Learning

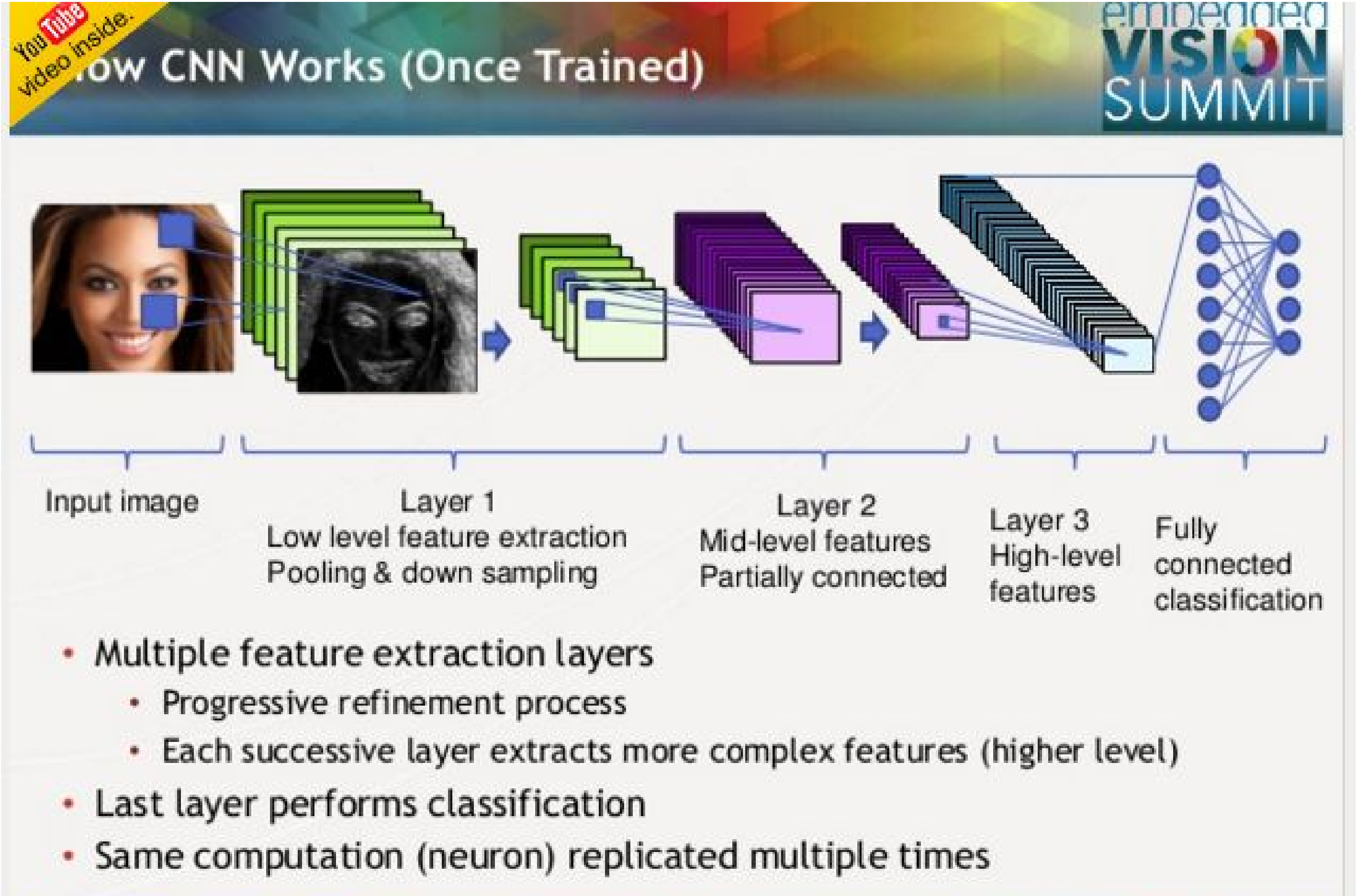
- **supervised**: examples must be *labeled*
  - You tell the algorithm which faces are female or male
- **unsupervised**: examples are *unlabeled*
  - You hand algorithm faces but don't tel which faces are male or female
  - Algorithm will plot features about inpu and look at what features cluster toge to classify data



Supervised Learning

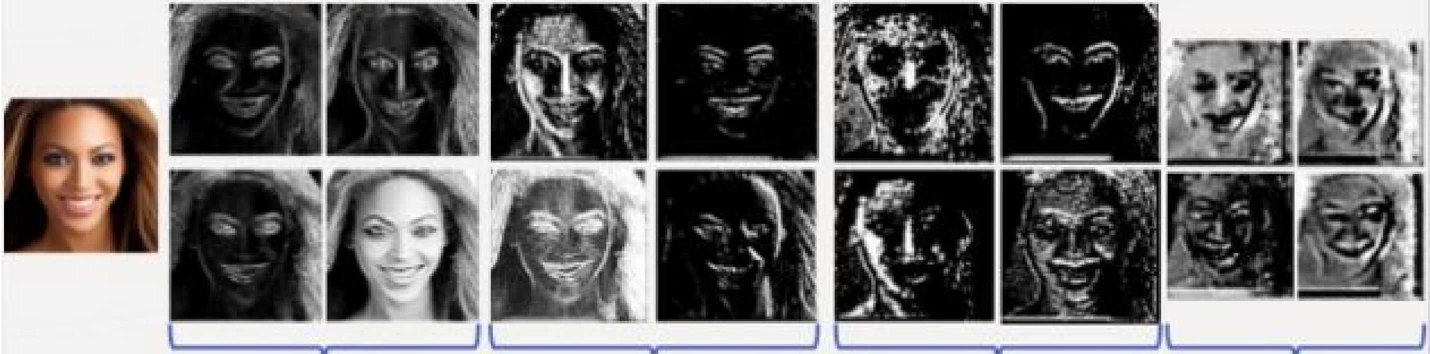
- End-to-End Learning
- Algorithm derives its own features from the labeled data
- Requires no hand-designed features
- provide the raw input signal values to the classifier
- Used convolutional neural network

convolutional neural network



isualising a CNN

embedded VISION SUMMIT

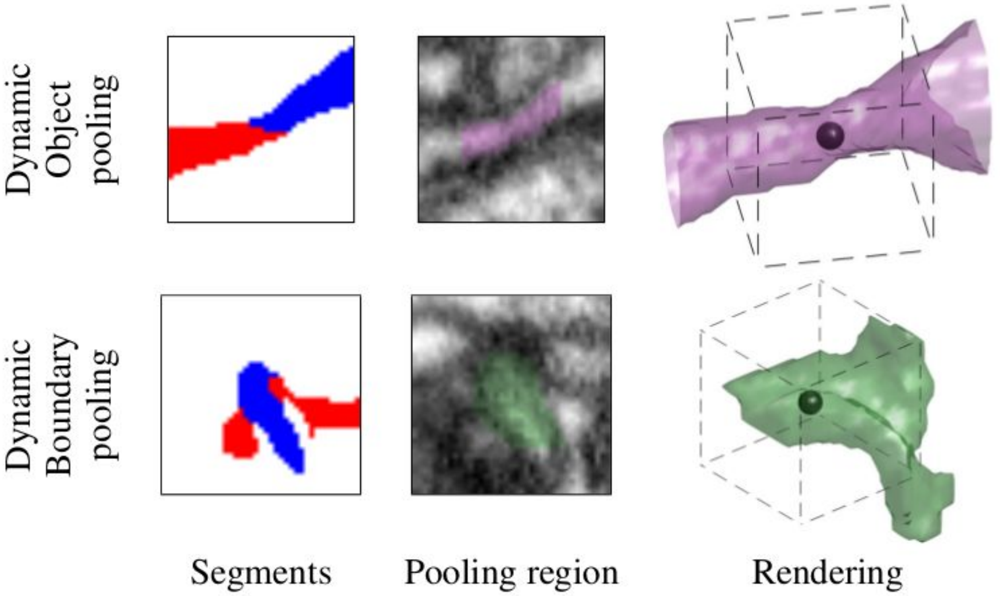


Layer 1 output sample | Layer 2 output sample | Layer 3 output sample | Layer 4 output sample

- Each layer of convolutions extract progressively higher level features
  - Subsampling / max pooling to “zoom out” and detect bigger objects with smaller convolutions
  - Non-linear function on each neuron to activate it

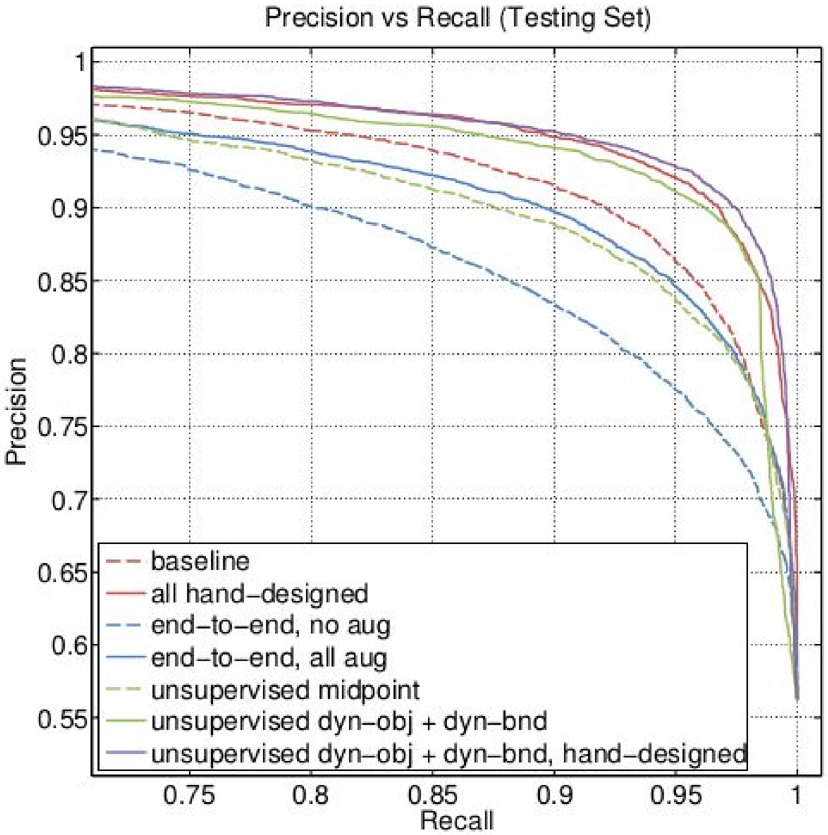
Unsupervised Learning

- Similar CNN technique as supervised learning
- **dynamic pooling:** the region to pool over is dependent on the segments themselves.
  - we can restrict the average pooling to be over only features corresponding to locations in either of the two segments
  - improves accuracy by eliminating unneeded data



Resolution

- unsupervised learning, when combined with a novel dynamic pooling scheme, yields performance comparable to an ensemble set of all hand-designed features.
- To our knowledge, this is the first time purely learned features have been shown to provide competitive performance on a task involving analysis or classification of 3d shapes
- Substantial improvement in performance results as the feature set increases from a simple set of 6 features derived from boundary map values to the combined set of all hand-designed features



Future

- Found methods that can act as starting points for future feature learning methods
- a more sophisticated end-to-end strategy
- dynamic pooling
    - optimize pool sampling