Additional Applications of Backpropagation

ALVINN (Pomerleau, 1996)

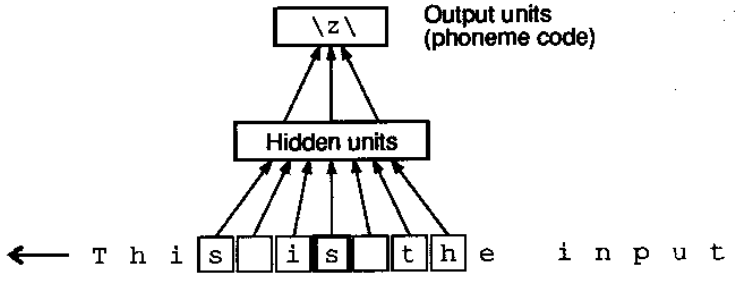
- network controlled steering of a car on a winding road
- network inputs: 30 x 32 pixel image from a video camera, 8 x 32 gray scale image from a range finder
- 29 hidden units
- 45 output units arranged in a line corresponding to steering angle
- achieved speeds of up to 70 mph for 90 minutes on highways outside of Pittsburgh

NETTalk

- Sejnowski, T. J. and Rosenberg, C. R. (1986) NETtalk: a parallel network that learns to read aloud, *Cognitive Science*, 14, 179-211.
- The authors taught a neural network to “read” using backpropagation.
- A stream of words were given to the network, along with the phoneme pronunciation of each in symbolic form.
- A speech generation device was used to convert the phonemes to sound.
- The network learned the phoneme pronunciations, thus was able to “speak” the words from a stream of words.

NETTalk

- 203-80-26 Multi-layer network
- Input is rolling sequence of 7 characters
- 203 = 7 x 29 different characters
- Output is the phoneme (if any) for the middle letter in the sequence



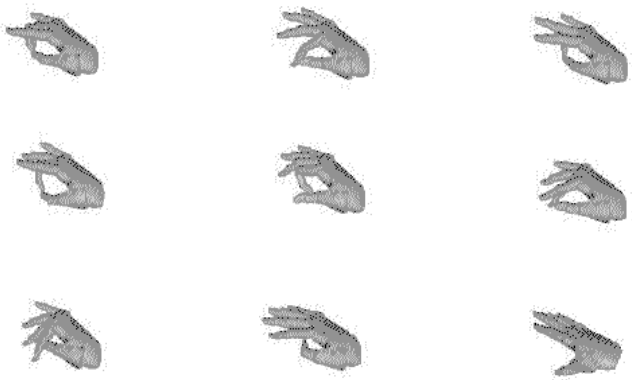
NETTalk

- 80 hidden units
- trained on 1024 words using a side-by-side English/phoneme source
- intelligible speech after 10 training epochs; 95% accuracy on training corpus after 50 epochs
- some hidden units developed meaningful responses (e.g., vowels vs. consonants)
- generalization: 78% accuracy on continuation of training text
- damaging network produced graceful degradation, with rapid recovery on retraining
- DECtalk performs better, but uses hand-coded linguistic rules developed over a decade

GregTalk

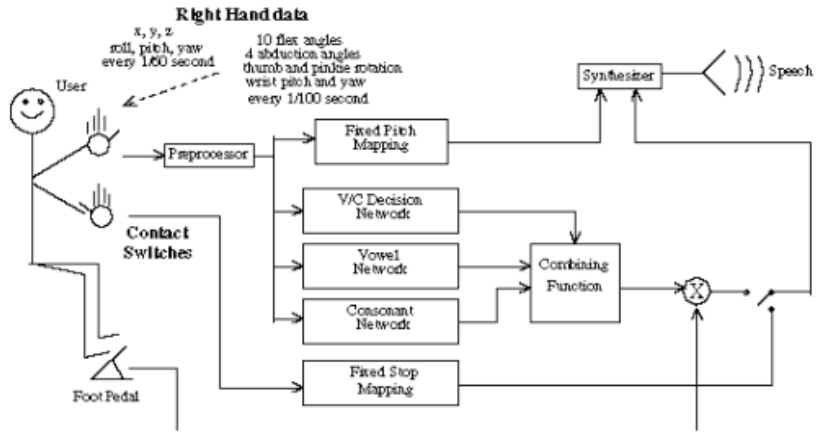
- Similar results were reproduced in a former offering of CS 152 by two Pomona College students named “Greg” (Greg Fishbein and Greg Schueler).

GloveTalk II

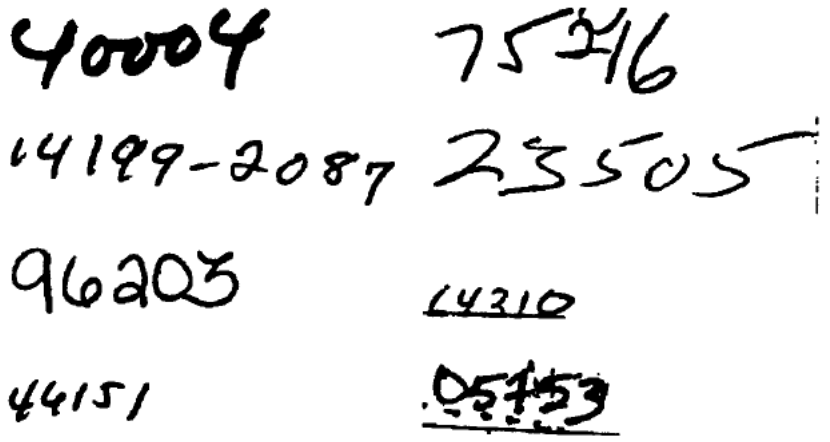


CHI '95 Proceedings
www.acm.org/sigchi/chi95/Electronic/documnts/papers/ssf_bdy.htm

GloveTalk

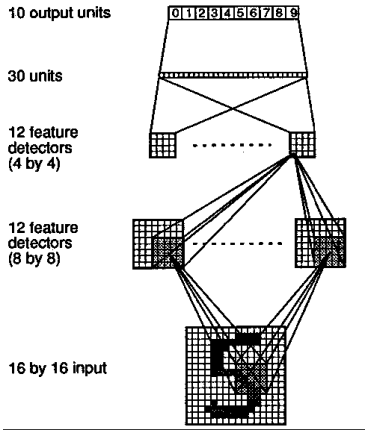


Zipcode Recognition (Yann LeCun, 1990)



Normalize Digits First

Network Structure



Feature Detectors

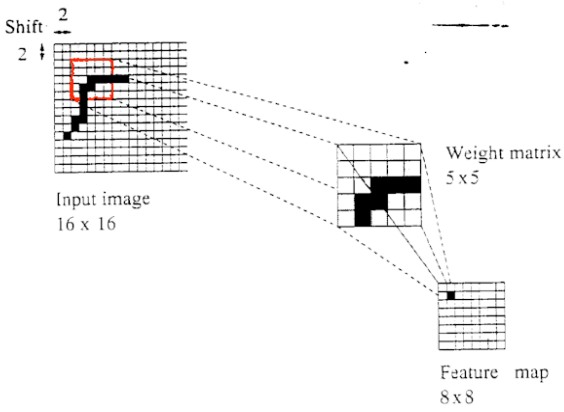


Figure 5.3: A feature map.

Sub-sampling Map

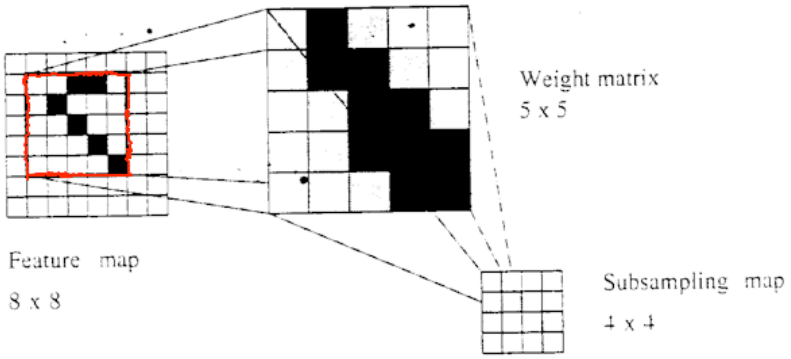


Figure 5.4: A sub-sampling map.

Architecture

- **Input layer**: 256 = 16x16 neurons with input values in range [-1, 1].
- **Hidden layer H1**: consists of 12 feature maps H1.1, … , H1.12.
- Feature map:
  - 8x8 neurons.
  - Each neuron in the feature map has the *same* incoming weights, but is connected to a square at a unique position in the input image. This square is called a *template*.

Architecture

- Hidden layer H2: consists of 12 sub-sampling maps H2.1, … , H2.12.
- Sub-sampling map:
  - Consists of 4x4 neurons.
  - Each neuron of the sub-sampling map is connected to a 5x5 square of H1.j, for each j in 8 of the 12 feature maps.
  - All neurons of the sub-sampling map share the same 25 weights.

Architecture

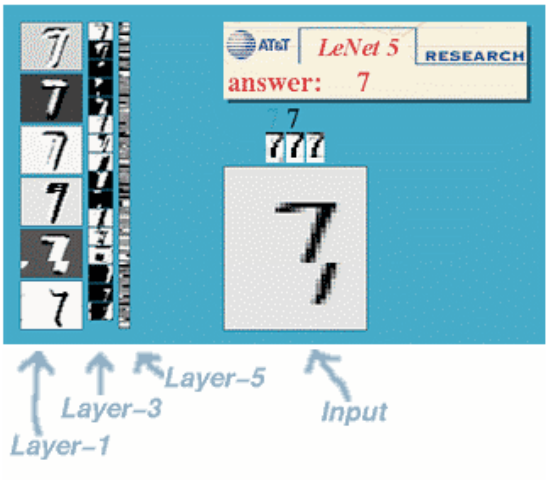
- Hidden layer H3:
  - Consists of 30 neurons.
  - H3 is completely connected to the sub-sampling layer (H2).
- Output layer: consists of 10 neurons, numbered 0, … , 9 and the neuron with the highest activation value is chosen. The digit recognized is equal to the cell number.

Atypical Data Recognized

Further Details and Results

- ~10,000 digits from the U.S. mail were used to train and test system
- ZIP codes on envelopes were initially located and segmented by a separate system (difficult task in itself)
- *weight sharing* used to constrain number of free parameters
- 1256 units + 30060 links + 1000 biases, but only 9760 free parameters
- used an accelerated version of backprop (pseudo-Newton rule)
- trained on 7300 digits, tested on 2000
- error rate of ~1% on training set, ~5% on test set
- if marginal cases were rejected (two or more outputs approximately the same), error reduced to ~1% with 12% rejected
- used "optimal brain damage" technique to prune unnecessary weights
- after removing weights and retraining, only ~1/4 as many free parameters as before, but better performance
- 99% classification accuracy with 9% rejection rate
- achieved state of the art in digit recognition
- much problem-specific knowledge was designed into the network architecture
- preprocessing of input data was crucial to success

http://yann.lecun.com/exdb/lenet/



Sonar target recognition (Gorman and Sejnowski)

- trained 2-layer backprop network to distinguish between reflected sonar signals of rocks and metal cylinders at bottom of Chesapeake Bay
- 60 input units, 2 output units
- input patterns based on Fourier transform of raw time signal
- tried varying numbers of hidden units (0, 3, 12, 24)
- best performance with 12 hidden units (close to 100% accuracy)
- 85-90% classification accuracy for signals not in training set