Advanced Data Analysis and Machine Learning

Lecture: Deep Learning and Applications

Lasse Lensu

2015-11-22

Outline

LUT
Lappeenranta
University of Technology

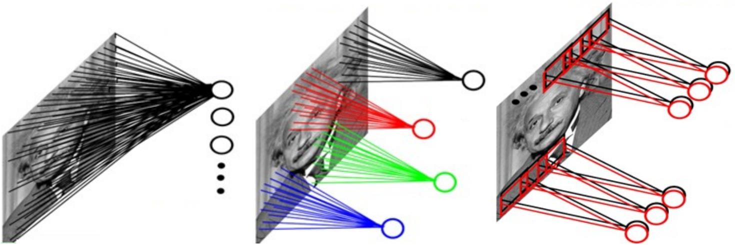
Convolutional neural networks (CNNs)

- Image processing and analysis tasks have been solved by selecting suitable image features to represent the interesting image objects.
- For example, the object detection process can be realised by searching or classifying suitable object candidates based on the image features in a supervised manner.
- Manual or "brute-force" selection of the appropriate image features can be replaced by learning the relevant features from data.

Network architectures

LUT Lappeenranta University of Technology

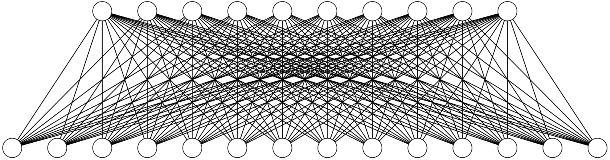
- Fully connected architecture.
- Local receptive field architecture.
- Local detectors with sliding window search.



[1]

RBMs

- Restricted Boltzmann machine (RBM) with two layers and binary stochastic units at both layers.



[1]

EBMs

- The energy of a model

$$E(\boldsymbol{v}, \boldsymbol{h}; \theta) = -\sum_{i=1}^{V} \sum_{j=1}^{H} v_i h_j W_{ij} - \sum_{i=1}^{V} v_i b_i - \sum_{j=1}^{H} h_j a_j \tag{1}$$

  where θ includes the model parameters, $\boldsymbol{W}^{V \times H}$ contains the weights and $\boldsymbol{B}^{V}$ and $\boldsymbol{a}^{H}$ are the biases.

- An energy-based model (EBM) associates a scalar energy value to each configuration. Then, learning is modifying the energy function so that its shape is as desired, for example, desired configurations have low energy.
- The energy function of a RBM is linear in its free parameters.

Gaussian RBM

- To handle real-valued data, Gaussian RBM with binary hidden units and linear visible units with Gaussian noise is used.
- The energy of the model

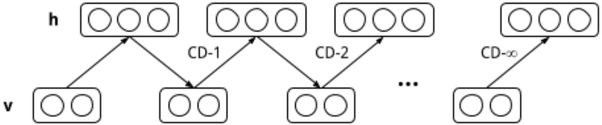
$$E(\boldsymbol{v},\boldsymbol{h};\theta) = -\sum_{i=1}^{V}\sum_{j=1}^{H}\frac{v_i}{\sigma_i}h_j W_{ij} + \sum_{i=1}^{V}\frac{(v_i-b_i)^2}{2\sigma_i^2} - \sum_{j=1}^{H}h_j a_j \quad (2)$$

  where $\boldsymbol{v}$ represents the real-valued activations of the visible units.
- Conditional distributions for the hidden and visible units have to be modified compared to the binary unit case to take into account the real values and standard deviation of the noise.

Contrastive divergence

LUT Lappeenranta University of Technology

- Contrastive divergence (CD) allows efficient learning of an EBM by approximation.
- Gibbs sampling is a Markov chain Monte Carlo (MCMC) technique providing a well-grounded way to approximate the value of an integral/sum.
- The Markov chain is initialised by using a sample from the training data.
- CD does not wait for the chain to converge.
- Alternating Gibbs sampling can be used to learn an RBM in parallel.



[1]

SM

- Score matching (SM) is an alternative to CD.
- EBMs can be estimated by minimizing the expected squared distance between the log-density gradients of the model and observed data.
- A denoising autoencoder (DAE) is a deep neural network (DNN) that enforces the hidden layer to discover more robust features by reconstructing the input.
- Denoising score matching (DSM) allows efficient learning of variances of, for example, natural images.

Momentum

- Momentum-based methods such as Nestrov's accelerated gradient (NAG) are common for training deep networks with stochastic gradient descent (SGD).
- The methods accelerate in the directions of low-curvature and explore new areas of the parameter space that are local minima of higher quality.
- NAG update rule is as follows:

$$v_{t+1} = \mu v_t - \alpha \nabla_{\theta_t} f(\boldsymbol{x}; \theta_t + \mu v_t) \tag{3}$$

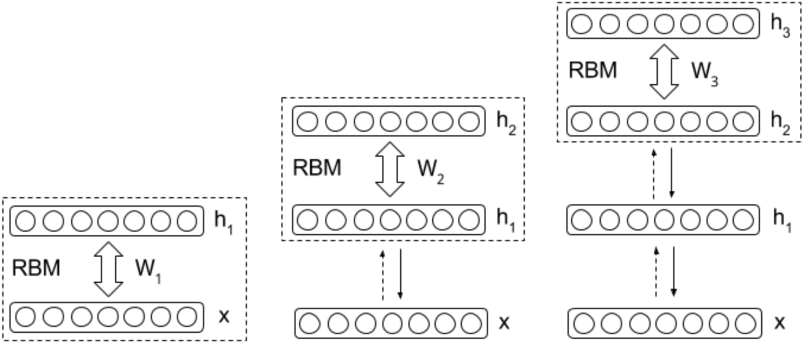
$$\theta_{t+1} = \theta_t + v_{t+1} \tag{4}$$

where $\mu \in [0, 1]$ is the momentum coefficient and v_t is the velocity.

DBN

LUT Lappeenranta University of Technology

- Deep belief networks (DBNs) combine a set of simpler models that are learned sequentially.
- The approach greedily trains one layer at a time by carrying out unsupervised learning for each layer.



[1]

Applications of deep learning

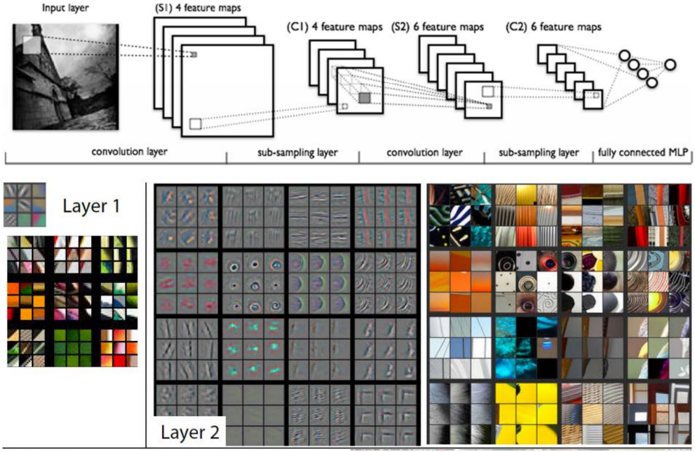
LUT Lappeenranta University of Technology

- Learning the image features and convolving.
- Example applications:
  - Categorising natural images
  - Prediction of facial keypoint locations
  - Pretraining for classifying hand-written numbers
  - Pretraining for categorising natural images

Categorising natural images

LUT Lappeenranta University of Technology

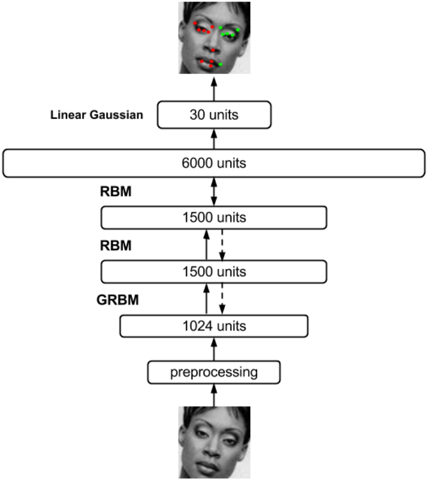
- Learning the image features for convolution.



[Matthew Zeiler & Rob Fergus]

Facial keypoint locations

Designing the architecture of a convolutional neural network:

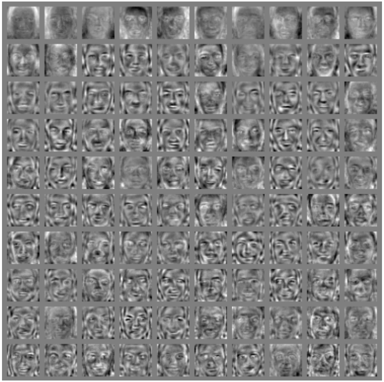


[2]

Facial keypoint locations

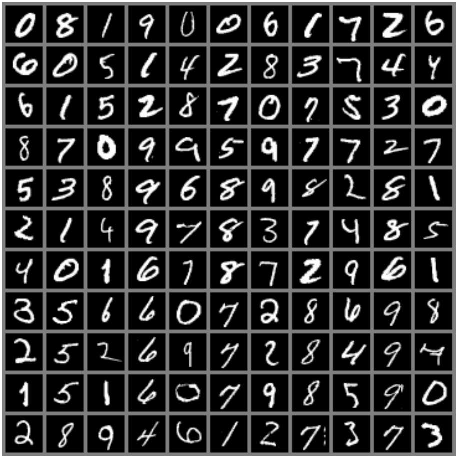
LUT
Lappeenranta
University of Technology

Visualising the learning and facial keypoint prediction:



[2]

MNIST data

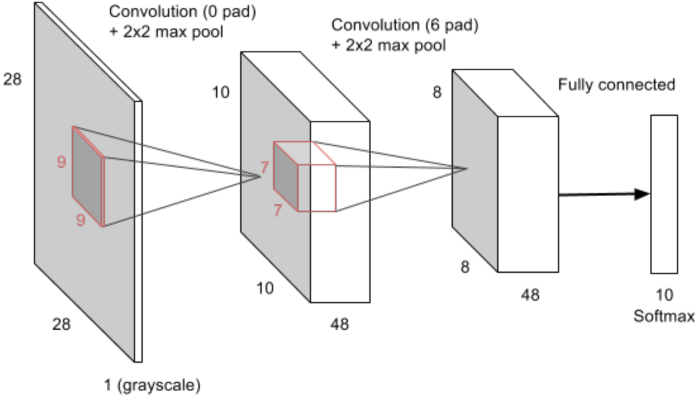


[1]

Classifying hand-written numbers

LUT Lappeenranta University of Technology

Designing the architecture of a convolutional neural network:

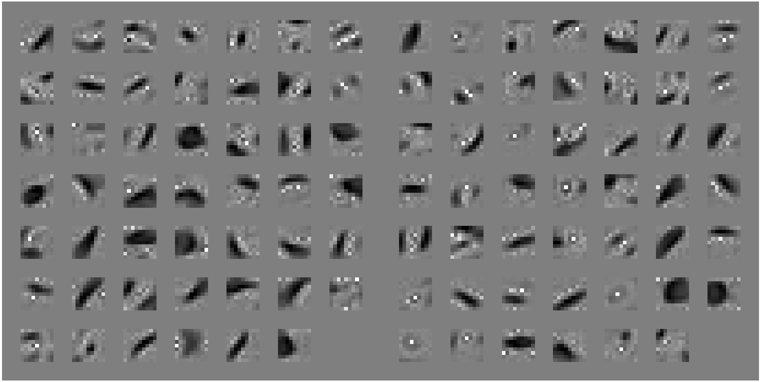


[1]

Classifying hand-written numbers

LUT
Lappeenranta
University of Technology

CNN and convolutional deep belief network (CDBN) weights



[1]

Classifying hand-written numbers

Table : Training with MNIST: average and STD of test error (do denotes dropout). [1]

Labeled training samples	1,000	5,000	60,000
ReLU-CNN	4.92% (0.35)	1.98% (0.08)	0.61%
ReLU-CDBN	4.53% (0.30)	1.85% (0.14)	0.51%
ReLU-CDBN (do)	4.94% (0.44)	1.88% (0.12)	0.48%

Classifying hand-written numbers

LUT Lappeenranta University of Technology

Table : Regularization and momentum optimization with MNIST: average and STD of test error (do denotes dropout). [1]

Labeled training samples	1,000	5,000	60,000
ReLU-CNN	2.87% (0.32)	1.31% (0.07)	0.56%
ReLU-CDBN	3.21% (0.37)	1.35% (0.04)	0.65%
ReLU-CDBN (do)	3.44% (0.60)	1.39% (0.09)	0.63%

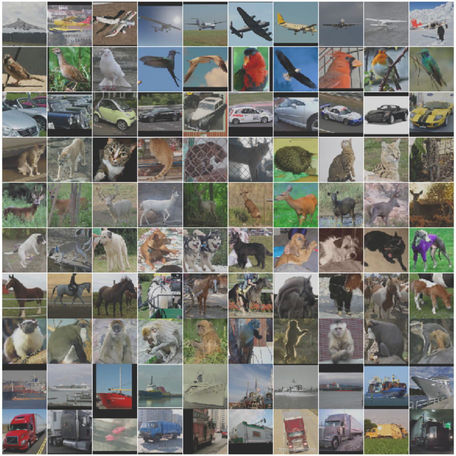
Classifying hand-written numbers

Table : State of the art with MNIST (do denotes dropout). [1]

Model	Error
ReLU-CNN	0.56%
ReLU-CDBN	0.51%
ReLU-CDBN (do)	0.48%
DNN + Dropconnect	0.21%
Multi-column DNN	0.23%
Convolutional Kernel Networks	0.39%
Maxout Networks	0.45%
Sigmoid-CDBN + SVM	0.82%

STL-10 data

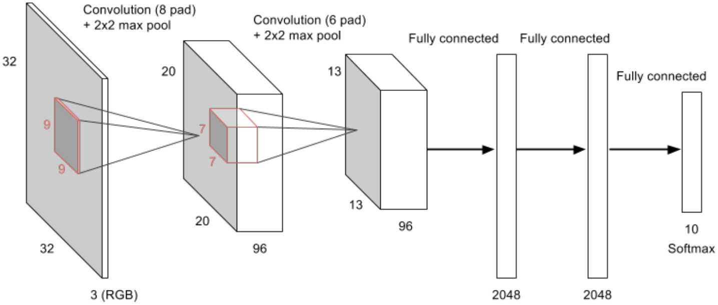
LUT
Lappeenranta
University of Technology



[1]

Categorising natural images

Designing the architecture of a convolutional neural network:

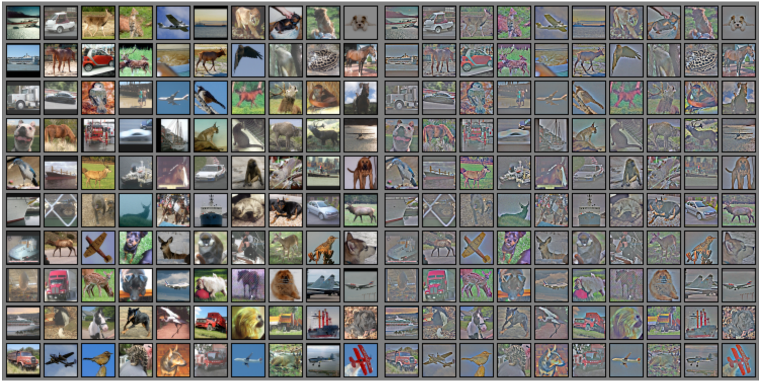


[1]

Categorising natural images

LUT Lappeenranta University of Technology

Downsampled example images and zero-phase component analysis (ZCA) whitened versions

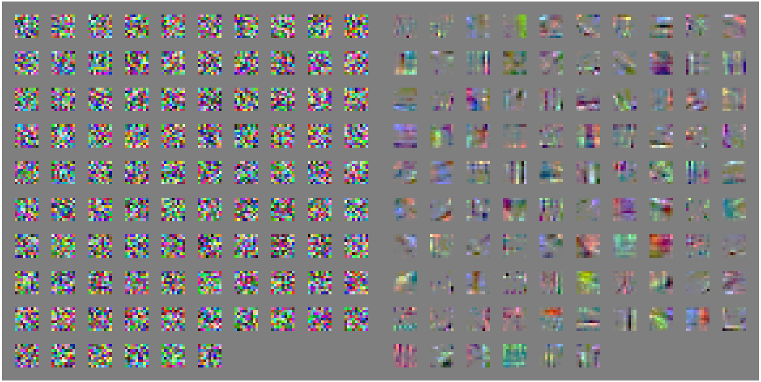


[1]

Categorising natural images

LUT Lappeenranta University of Technology

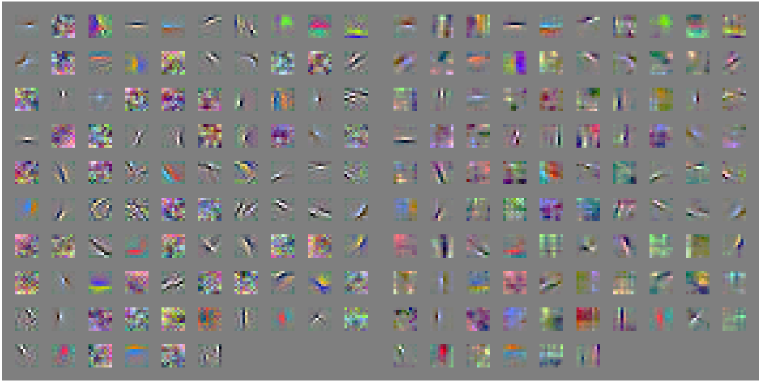
Randomly initialised weights and learned CNN



[1]

Categorising natural images

Pretrained weights and learned CDBN



[1]

Categorising natural images

LUT Lappeenranta University of Technology

Table : Pretraining with STL-10: average and STD of test error. [1]

Model	Accuracy
ReLU-CNN	51.38% (0.41)
ReLU-CDBN	53.02% (0.39)
Committees of CNNs	68% (0.55)

Summary

- The traditional way of manually selecting appropriate image features can be replaced with CNNs which learn the relevant features.
- EBMs can be used to characterise the network for searching a better configuration (learn network parameters).
- A DBN combines a set of simpler models and can be learned greedily layer by layer.

References

LUT
Lappeenranta
University of Technology

Mikko Haavisto.
Pretraining convolutional neural networks for visual recognition tasks.
Master's thesis manuscript, 2015.

Mikko Haavisto, Arto Kaarna, and Lasse Lensu.
Deep learning for facial keypoints detection.
In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications, VISAPP 2015*, volume 2, pages 289–296, Berlin, Germany, March 11–14, 2015.