1 Convolutional Networks

This is a post written for the AI Helsinki study group *Image and Video Statistics*. It is based on the Chapter 9 of the book *Deep Learning*.

1.1 Definition

1.2 Use of Convolutional Networks

1.3 Benefits

1.4 Examples

1.5 Sparse Connectivity

1.6 Growing Receptive Fields

1.7 Parameter Sharing

1.8 Convolutional Network Components

1.9 Max Pooling

Pictures: Without Shift, Shifted

1.10 Example of Learned Invariances

1.11 Pooling with Down Sampling

1.12 Examples of Architectures

1.13 Convolution with Strides

1.14 Zero Pading Enables Deeper Networks

1.15 Comparison of Local Connections, Convolution, and Full Connections

Pictures: Local, Convolution, FC

1.16 Partial Connectivity Between Channels

1.17 Tiled Convolution

Pictures: Local Connection, Tiled Convolution, Traditional Convolution

1.18 Recurent Convolutional Network

1.19 Gabor Functions (optional)

1.20 Gabor-like Learned Kernels (optional)

2 Appendix: Tensors

We define tensors as multidimensional arrays, since this definition is the most suitable one for our practical purposes.

Namely, the zero-dimensional tensor are just real numbers, also known as *scalars* and for $d > 0$ the d-dimensional tensors T are d-dimensional arrays with real valued components $T_{i_1,\ldots,i_d}$, where the indicies $i = (i_1, \ldots, i_d)$ belong to some rectangular grid

$$I = [1, \ldots, n_1] \times \ldots \times [1, \ldots, n_d]$$

and the component index upper bounds n_k are fixed positive numbers for the given tensor. Often the component indicies are written with separate letters and the commas are dropped, as in T_{ij}.

Examples: The one-dimensional tensors can be seen as *vectors*:

$$T = \begin{pmatrix} T_1 \\ \vdots \\ T_n \end{pmatrix}, \quad T_i \in \mathbb{R} \quad \text{for all} \quad i \in \{1, \ldots, n\}.$$

The two-dimensional tensors can be seen as *matricies*:

$$T = \begin{pmatrix} T_{11} & \cdots & T_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}, \quad T_{ij} \in \mathbb{R} \quad \text{for all} \quad (i, j) \in [1, \ldots, m] \times [1, \ldots, n].$$

One may notice that a d-dimensional tensor can be wieved as an array of $(d-1)$-dimensional tensors of the same type. That is, a vector is an array of scalars, a matrix is an array of vectors of the same length, a three-dimensional tensor is an array of matricies with the same $m \times n$ shape, and so on.

There are also more abstract ways of defining tensors, which take into account the coordinate transformation rules for different types of tensors, but since in all our cases the coordinate system is fixed and the tensors are used merely for grouping of the calculations, the definition stated above is sufficient.

3 References/Links