[course site]

Day 1 Lecture 3

Convolutional Neural Networks

Elisa Sayrol

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Department of Signal Theory and Communications
Image Processing Group

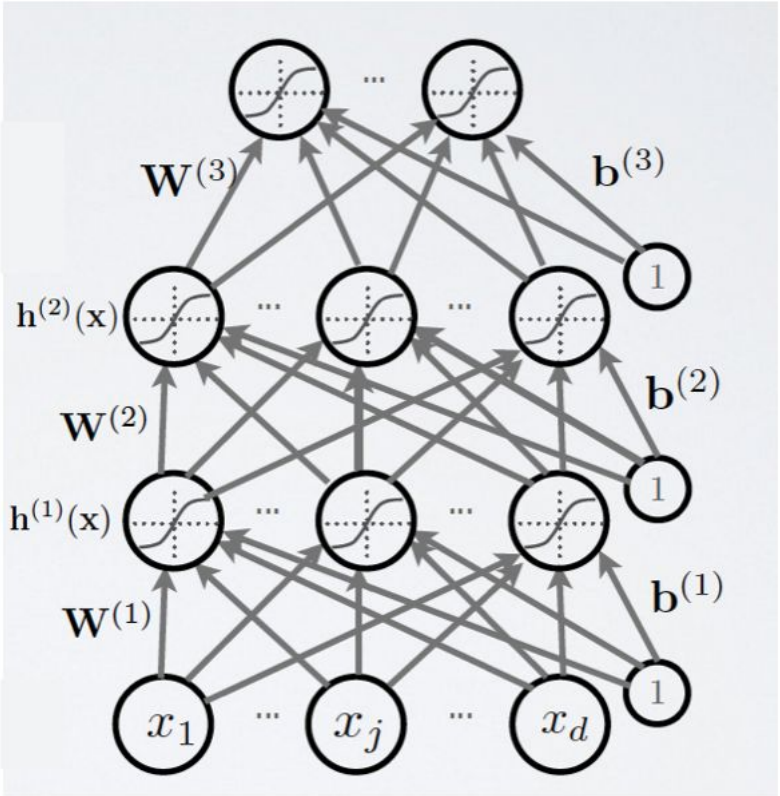
The Deep Neural Network

The i-th layer is defined by a matrix **Wi** and a vector **bi,** and the activation is simply a dot product plus **bi**:

$$h_i = f(W_i \cdot h_{i-1} + b_i)$$

Num parameters to learn at i-th layer:

$$N^i_{params} = N^i_{inputs} \times N^i_{units} + N^i_{units}$$

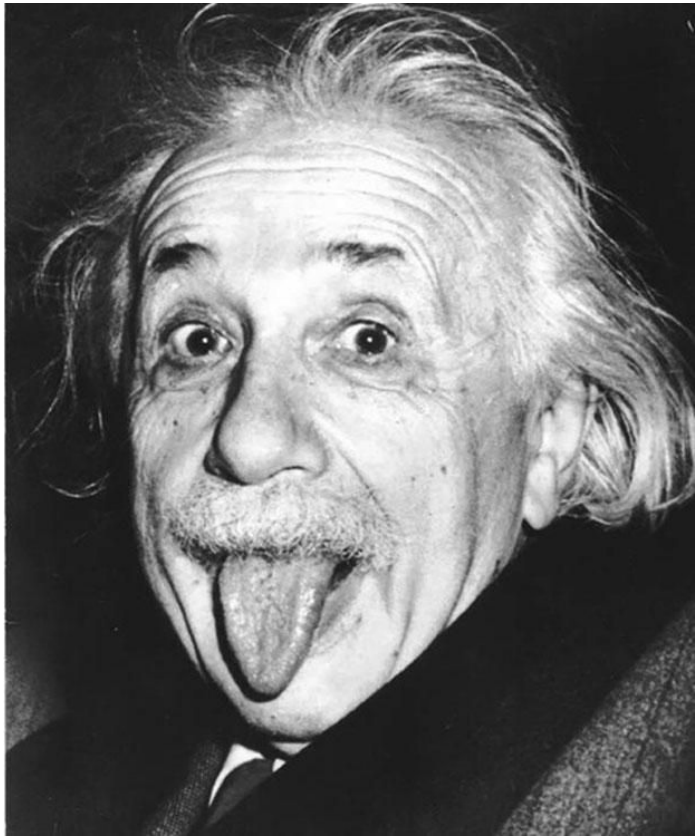


Slide Credit: Hugo Laroche NN course

From Neurons to Convolutional Neural Networks

What if the input is a 2D signal?

(images, spectrogram, but also 1D signals)



From Neurons to Convolutional Neural Networks

For a 200x200 image, we have $4x10^4$ neurons each one with $4x10^4$ inputs, that is $16x10^8$ parameters, only for one layer!!!

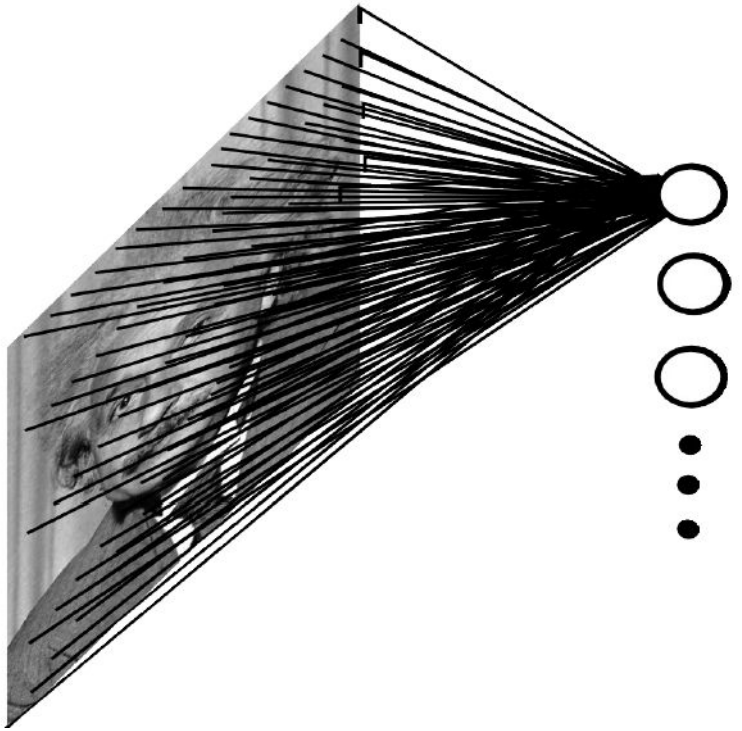


Figure Credit: Ranzatto

From Neurons to Convolutional Neural Networks

For a 200x200 image, we have $4x10^4$ neurons each one with 10x10 "**local connections**" (also called receptive field) inputs, that is $4x10^6$

What else can we do to reduce the number of parameters?

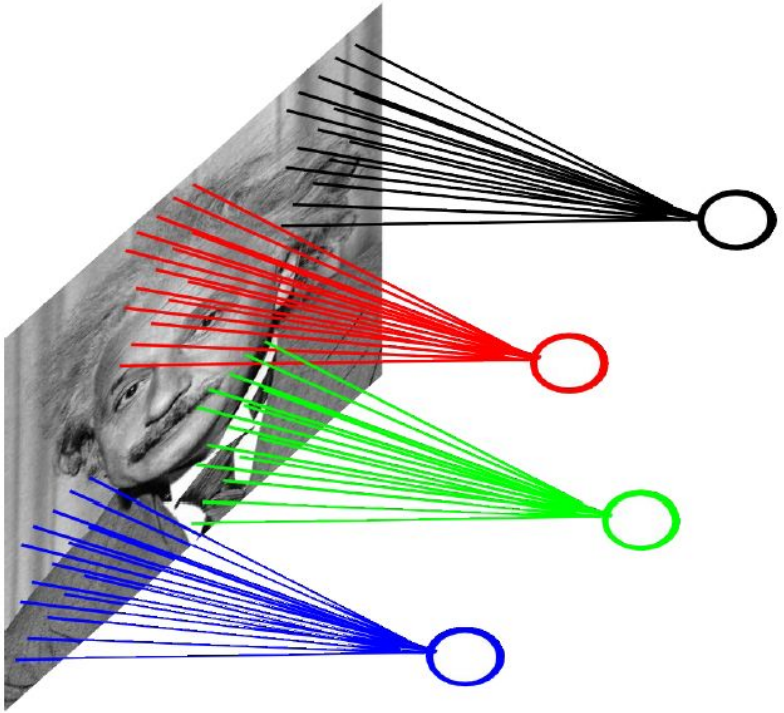


Figure Credit: Ranzatto

From Neurons to Convolutional Neural Networks

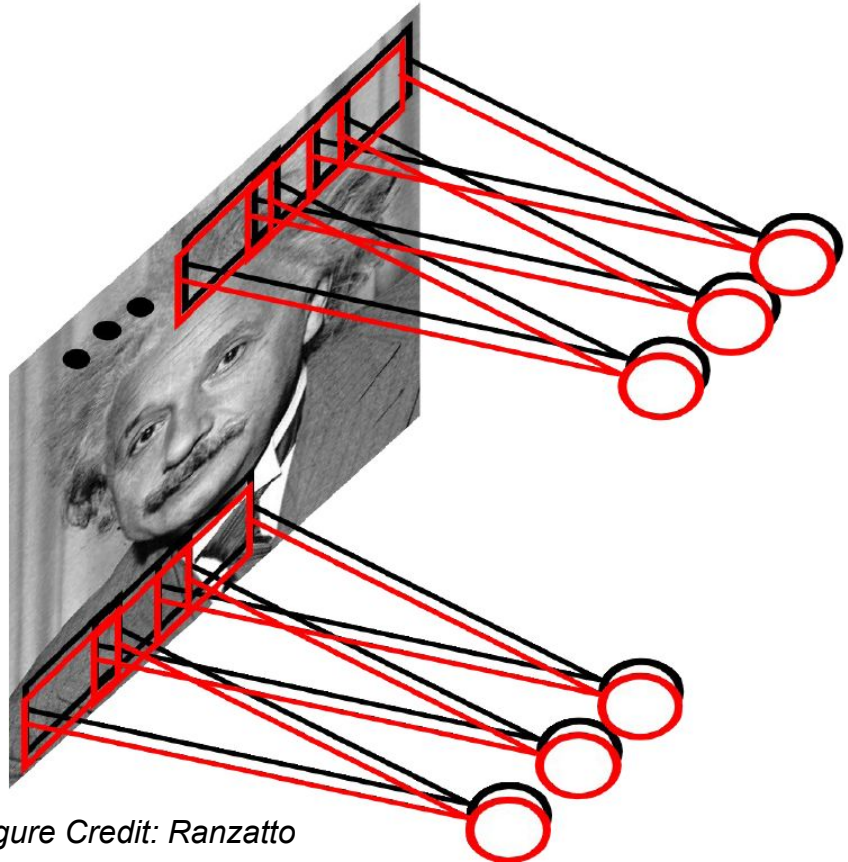


Figure Credit: Ranzatto

Translation invariance: we can use same parameters to capture a specific “feature” in any area of the image. We can try different sets of parameters to capture different features.

These operations are equivalent to perform **convolutions** with different filters.

Ex: With100 different filters (or feature extractors) of size 10x10, the number of parameters is 10^4

From Neurons to Convolutional Neural Networks

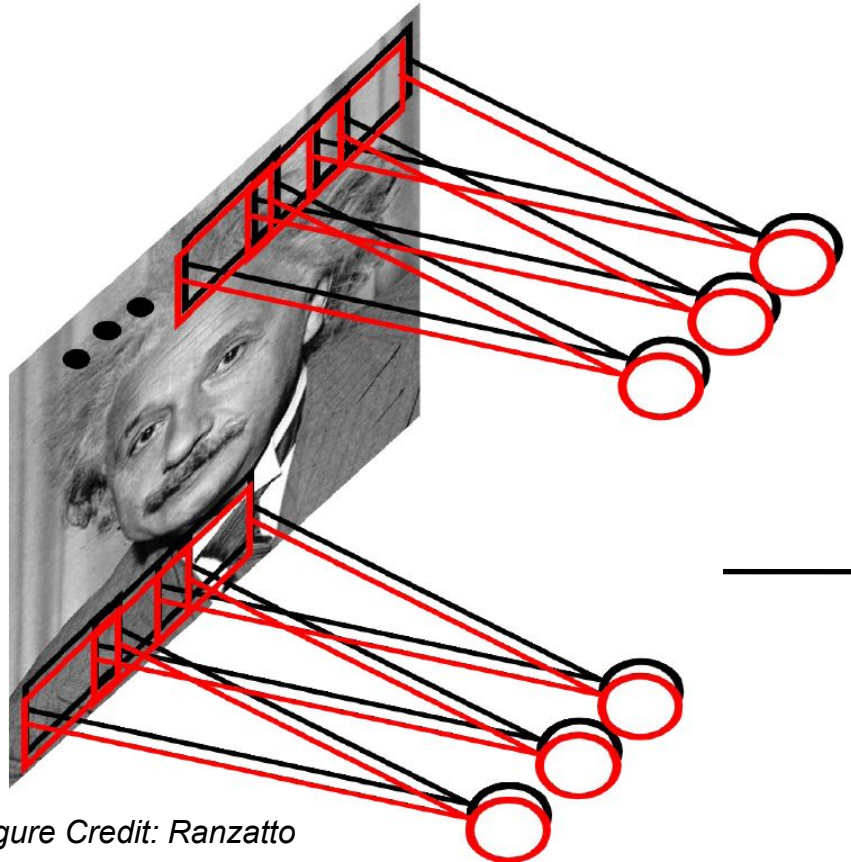
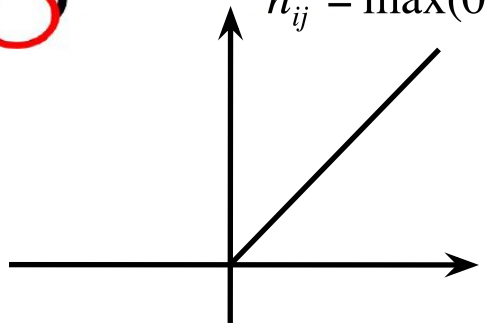


Figure Credit: Ranzatto

… and don't forget the activation function!

$$a_{ij} = \sum_{k,l} w_{kl} x_{k-i,l-j} + b$$

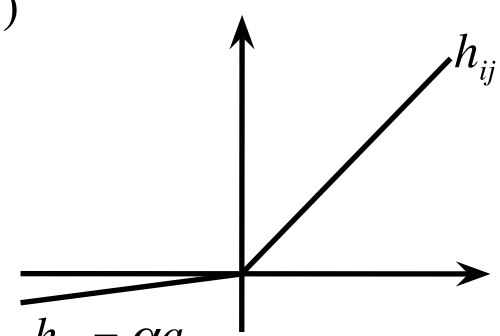
$$h_{ij} = \max(0, a_{ij})$$



ReLu

$$h_{ij} = a_{ij}$$

$$h_{ij} = \alpha a_{ij}$$



PReLu

$$g(a) = sigm(a) = \frac{1}{1+\exp(-a)}$$

$$g(a) = \tanh(a)$$

From Neurons to Convolutional Neural Networks

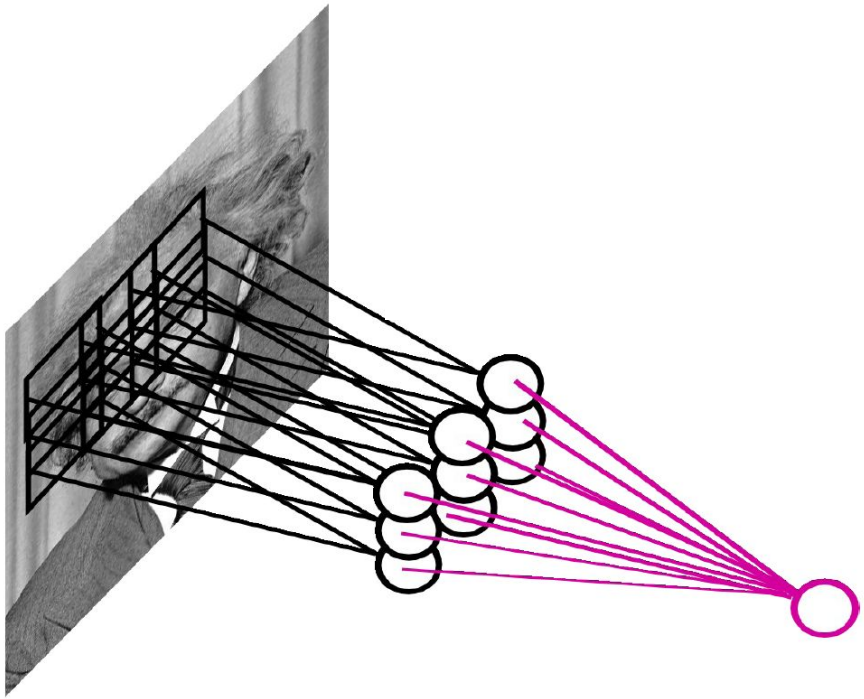


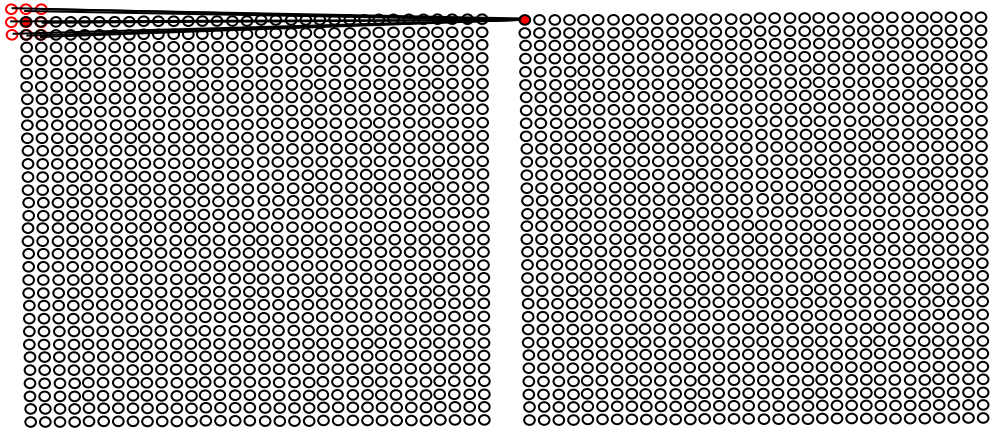
Figure Credit: Ranzatto

Most ConvNets use **Pooling** (or subsampling) to reduce dimensionality and provide invariance to small local changes.

Pooling options:

- **Max**
- Average
- Stochastic pooling

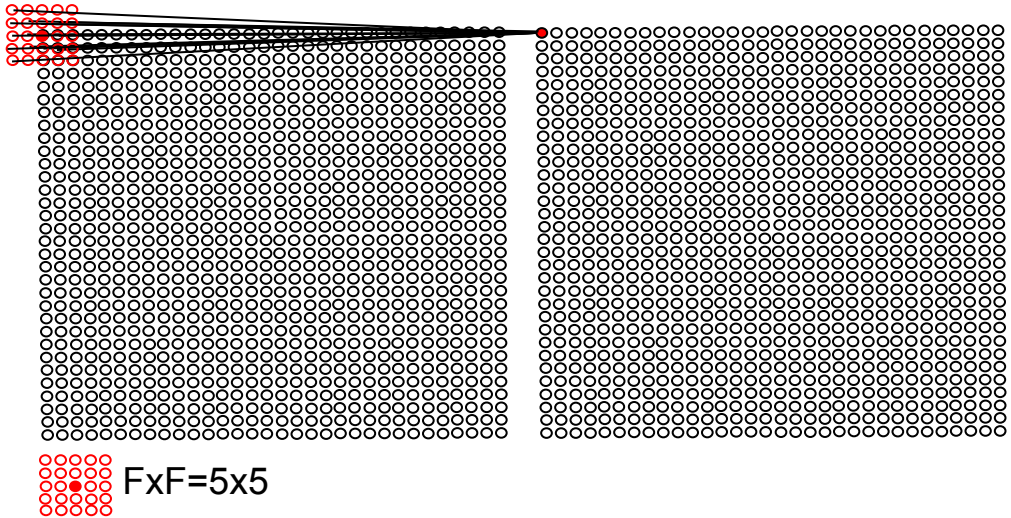
From Neurons to Convolutional Neural Networks



FxF=3x3

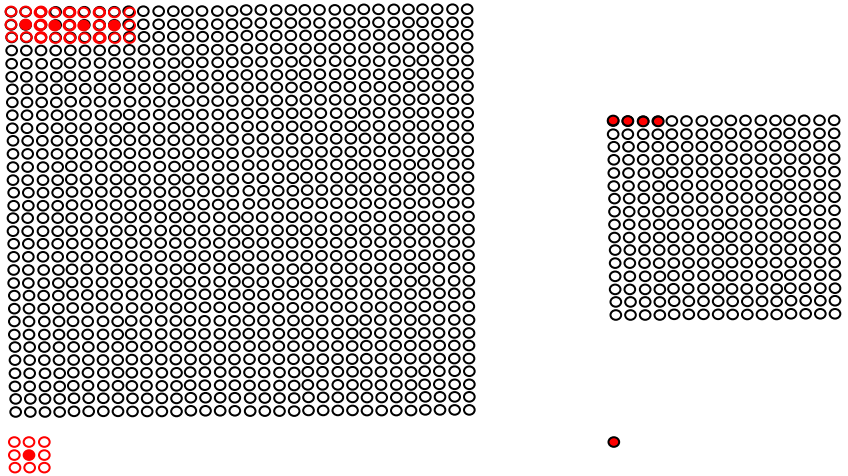
Padding (P): When doing the convolution in the borders, you may add values to compute the convolution.
When the values are zero, that is quite common, the technique is called zero-padding.
When padding is not used the output size is reduced.

From Neurons to Convolutional Neural Networks



Padding (P): When doing the convolution in the borders, you may add values to compute the convolution.
When the values are zero, that is quite common, the technique is called zero-padding.
When padding is not used the output size is reduced.

From Neurons to Convolutional Neural Networks



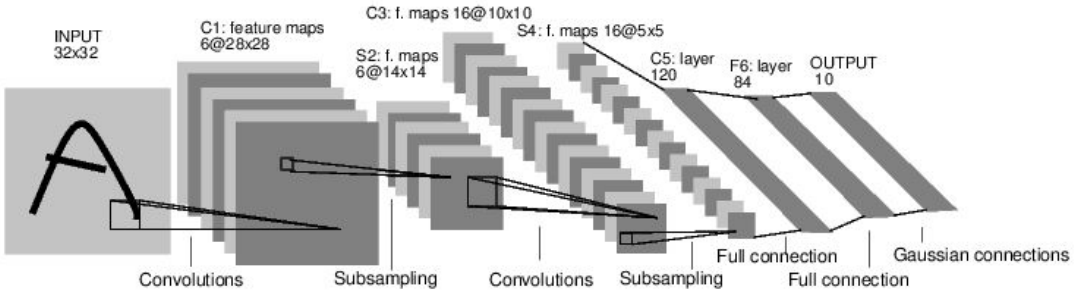
Stride (S): When doing the convolution or another operation, like pooling, we may decide to slide not pixel by pixel but every 2 or more pixels. The number of pixels that we skip is the value of the stride.
It might be used to reduce the dimensionality of the output

From Neurons to Convolutional Neural Networks

Example: Most convnets contain several convolutional layers, interspersed with pooling layers, and followed by a small number of fully connected layers

A layer is characterized by its width, height and depth (that is, the number of filters used to generate the feature maps)

An architecture is characterized by the number of layers



LeNet-5 From Lecun ´98

From Neurons to Convolutional Neural Networks

Example 1: CNN for SL

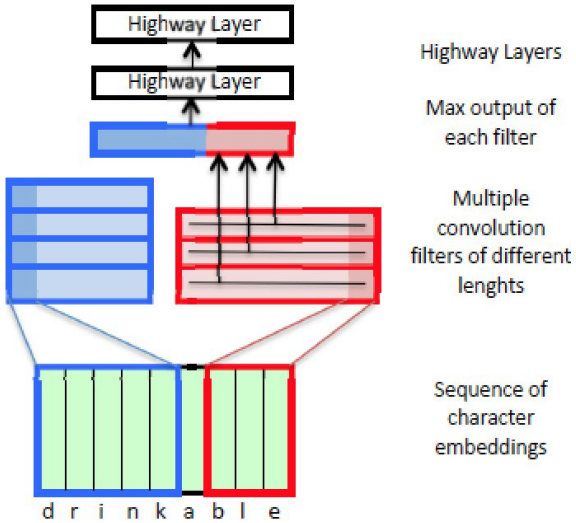


Figure 1: Character-based word embedding

"Character-based Neural Machine Translation"
Marta R. Costa-Jussà and José A. R. Fonollosa

From Neurons to Convolutional Neural Networks

Example 2: CNN for SL

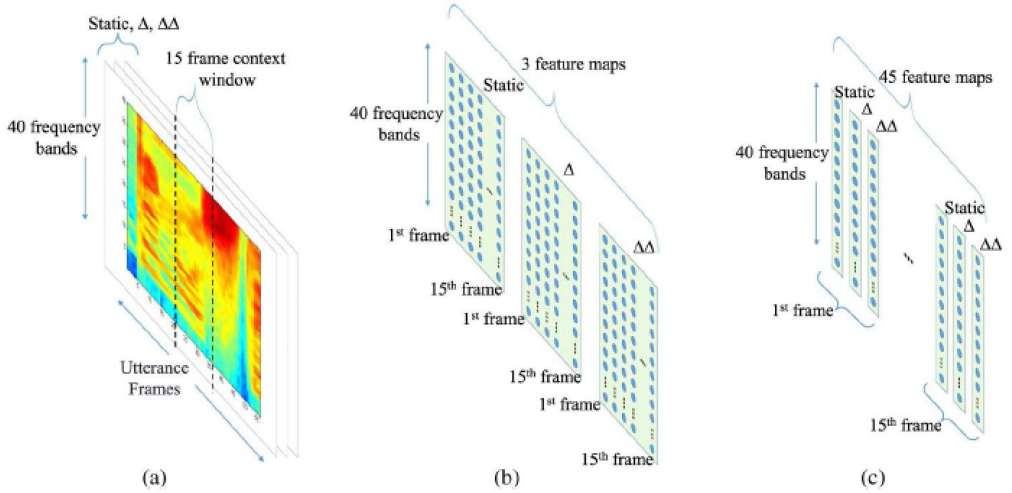


Fig. 1. Two different ways can be used to organize speech input features to a CNN. The above example assumes 40 MFSC features plus first and second derivatives with a context window of 15 frames for each speech frame.

"Convolutional Neural Network for Speech Recognition"
Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu
IEEE/ACM Transactions on Audio, Speech, and Language Processing, Vol.22, No. 10, October 2014