# Neural networks

## Computer vision - convolutional RBM

# CONVOLUTIONAL RBM

**Topics:** convolutional RBM

Lee et al. 2009

- How about designing convolutional unsupervised networks
  - let's consider the case of the RBM
  - could use same convolutional connectivity between input ($\mathbf{v}$) and hidden layer ($\mathbf{h}$)

$$P(\mathbf{v},\mathbf{h}) = \frac{1}{Z}\exp(-E(\mathbf{v},\mathbf{h}))$$

$$E(\mathbf{v},\mathbf{h}) = -\sum_{k=1}^{K}\sum_{i,j=1}^{N_H}\sum_{r,s=1}^{N_W} h_{ij}^k W_{rs}^k v_{i+r-1,j+s-1} - \sum_{k=1}^{K} b_k \sum_{i,j=1}^{N_H} h_{ij}^k - c\sum_{i,j=1}^{N_V} v_{ij}.$$

$$= -\sum_{k=1}^{K} h^k \bullet (\tilde{W}^k * v) - \sum_{k=1}^{K} b_k \sum_{i,j} h_{i,j}^k - c\sum_{i,j} v_{ij}$$

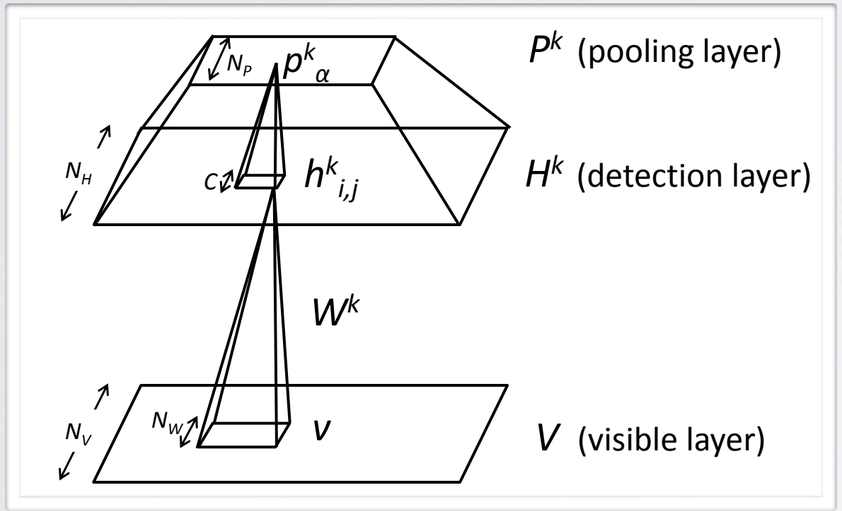
- $h^k{}_{ij}$ are the hidden units of the k[th] feature map
- $W^k{}_{rs}$ are the weights to the k[th] feature map
- $\tilde{W}^k$ are the weights with flipped rows and columns (convolution kernel)

# CONVOLUTIONAL RBM

**Topics:** convolutional RBM

Lee et al. 2009

- We can introduce a notion of probabilistic pooling
  - pooling unit $p_\alpha^k$ above is 1 only if at least one hidden unit $h_{i,j}^k$ in neighborhood is 1
  - within a pooling neighborhood, allow at most only a single unit $h_{i,j}^k$ equal to 1




$$I(h_{ij}^k) \triangleq b_k + (\tilde{W}^k * v)_{ij}$$

implies $p_\alpha^k$ is 1

$$P(h_{i,j}^k = 1|\mathbf{v}) = \frac{\exp(I(h_{i,j}^k))}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^k))}$$

$$P(p_\alpha^k = 0|\mathbf{v}) = \frac{1}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^k))}$$

implies all $h_{i,j}^k$ are 0

# CONVOLUTIONAL RBM

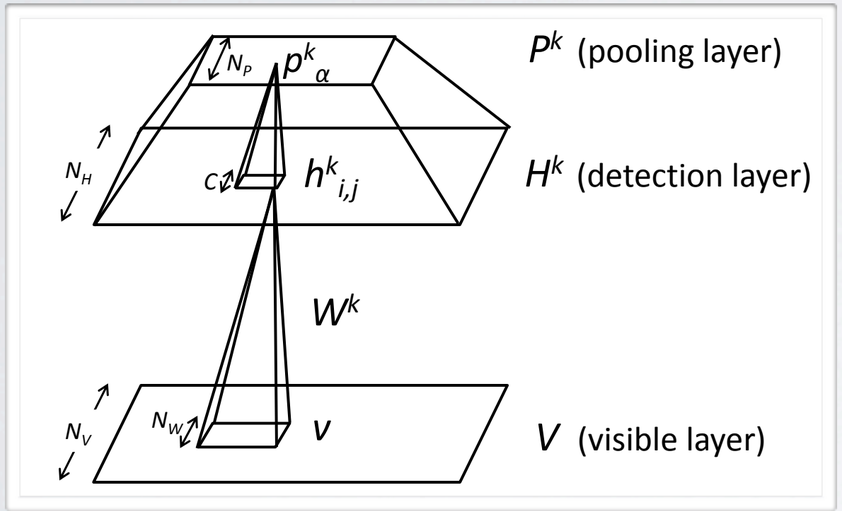
**Topics:** convolutional RBM

Lee et al. 2009

- Given the $h^k_{i,j}$ units, we sample each input independently:

$$P(v_{ij} = 1|\mathbf{h}) = \sigma((\sum_k W^k * h^k)_{ij} + c)$$

sigmoid




$$I(h^k_{ij}) \triangleq b_k + (\tilde{W}^k * v)_{ij}$$

implies $p^k_\alpha$ is 1

$$P(h^k_{i,j} = 1|\mathbf{v}) = \frac{\exp(I(h^k_{i,j}))}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h^k_{i',j'}))}$$

$$P(p^k_\alpha = 0|\mathbf{v}) = \frac{1}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h^k_{i',j'}))}$$

implies all $h^k_{i,j}$ are 0

# CONVOLUTIONAL RBM

**Topics:** convolutional RBM

Lee et al. 2009

- Using these adapted conditionals, we can perform contrastive divergence
  - energy gradients involve convolutions, similar to the backprop gradients in convolutional network
- Can stack convolutional RBMs
  - provides a pretraining procedure which doesn't require the extraction of patches
- See Lee et al. 2009 for more details