Vision in Human and Machine

Part 3
Principles of Sensory Coding

Heiko Wersing

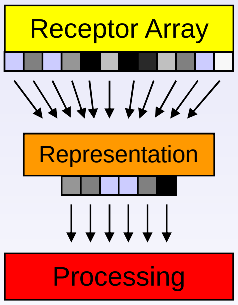
Honda Research Institute Europe GmbH

Sensory Coding

- What is the role of sensory coding ?
  - Transmitting information
  - Building representations
  - Preparing computations

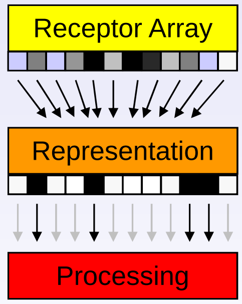
Types of sensory coding

Compact Coding



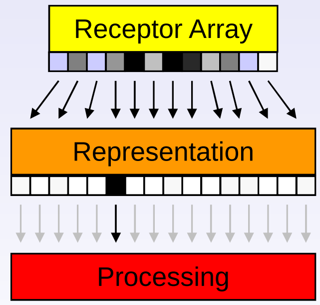
- Dimension reduction
- Few representing units
- Principal component Analysis

Sparse Distributed Coding



- Same or larger dimensionality (Overcomplete representation)
- Few number of active units
- Independent component analysis
- Nonnegative Matrix Factorization
- Sparse Coding Methods

Grandmother Cells



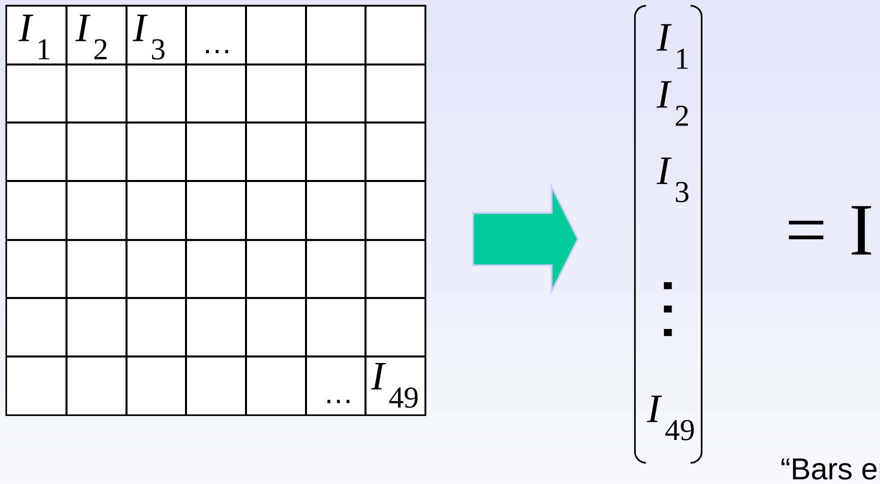
- “Explosion of dimensionality”
- One unit for every sensory concept

Convergence and divergence in the visual pathway

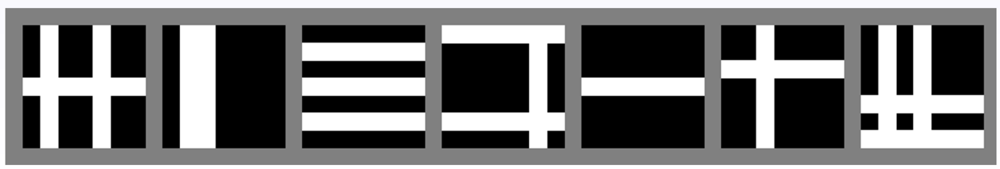
- Critical sampling of the input?
  - – 8 x 8 pixel patch uniquely represented by D=64 (upper bound)
  - Cat V1… 25 output fibers for every input
  - Macaque V1… 50 output fibers for every input
- Overcompleteness in the code
  - Nonlinearities in V1 exist to allow a large population of neurons to represent data with a small number of active units

Retina → LGN → Visual Cortex V1

An image decomposition toy example



"Bars ensemble"



I^1 I^2 I^3 … I^p … I^{2000}

Formal description of linear encoding

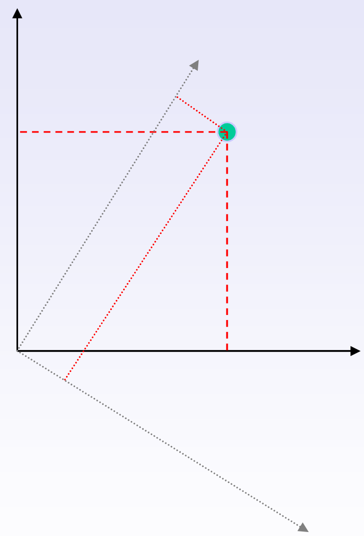
- The simplest case: Linear encoding (remember your linear algebra math course !)

$$E_1 = \underbrace{\sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2}_{\text{Reconstruction error}}$$

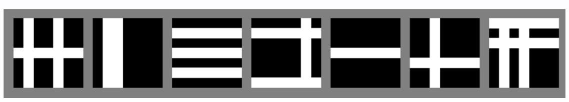
- I^p : Input pattern (vector)
- w_i : Basis (weight) vector representing the data
- s_i^p : Coefficient of the contribution of w

Case 1: Orthonormal basis expansion

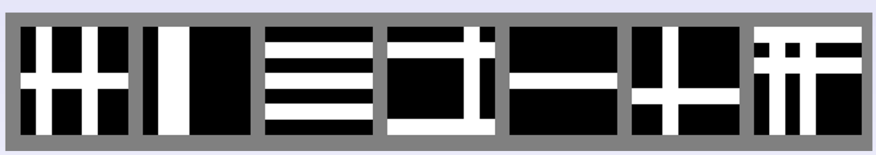
$$E_1 = \sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2$$

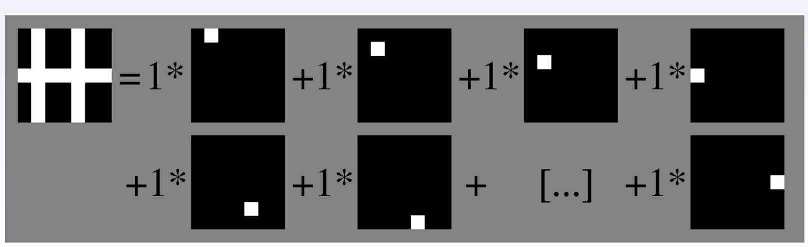


- Standard case: Exactly as many basis vectors as input dimensions
- Any linearly independent basis of weight vectors represents the input perfectly
- Bars ensemble: Any set of 49-dimensional linear independent vectors allow perfect reconstruction



Case 1: Orthonormal basis expansion



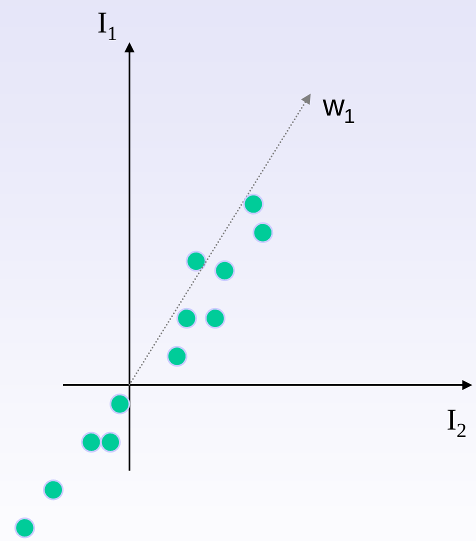


Non-sparse basis representation

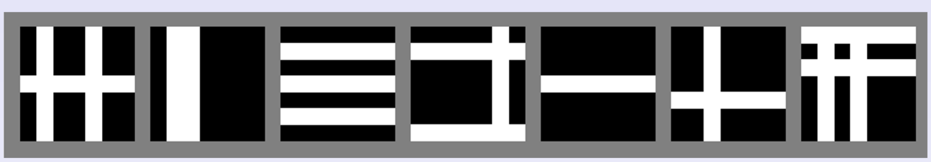
Case 2: Compact coding

$$E_1 = \sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2$$

- Compact coding: Fewer basis vectors as input dimensions
- “Information bottleneck”
- Optimal solution: Search for directions with maximum variance
- Method: Principal Component Analysis, Eigenvalue analysis of the covariance matrix



Case 2 : Compact Coding



Bars ensemble

First 20 principal components of the bars ensemble

- Principal components capture abstract directions of maximum variance
- Representation does not capture parts-based structure, but holistic variation
- First 14 principal components allow near-perfect reconstruction of the 49-dimensional bars stimuli

Sparse coding

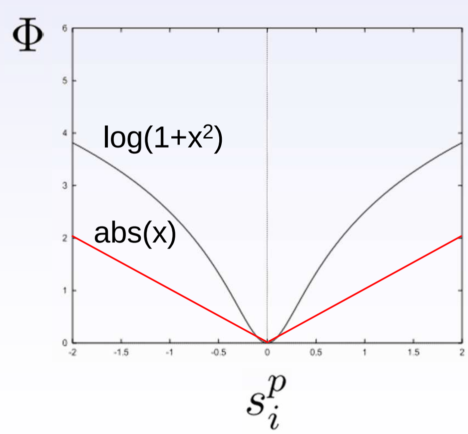
$$E_1 = \underbrace{\sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2}_{\text{Reconstruction error}} + \underbrace{\sum_p \sum_i \Phi(s_i^p)}_{\text{Sparsity-enforcing cost}}$$

- Sparse coding: More basis vectors than dimensionality
- "Overcomplete representations"
- Several different approaches exist
- Learning consists of solving a difficult complex optimization problem with several local minima

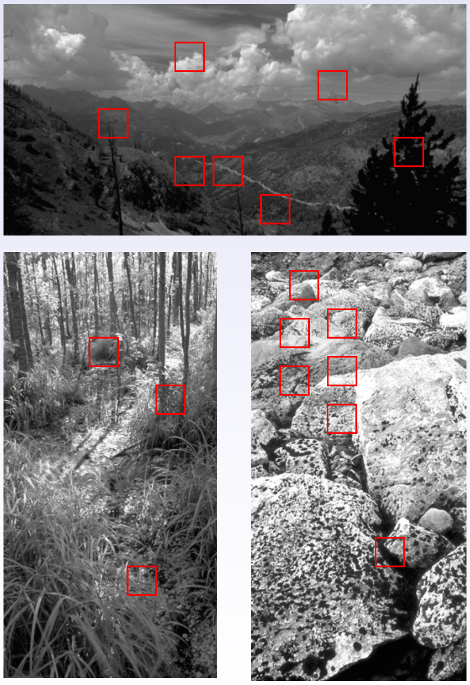
Sparse coding: 1. Olshausen & Field

$$E_1 = \underbrace{\sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2}_{\text{Reconstruction error}} + \underbrace{\sum_p \sum_i \Phi(s_i^p)}_{\text{Sparsity-enforcing cost}}$$

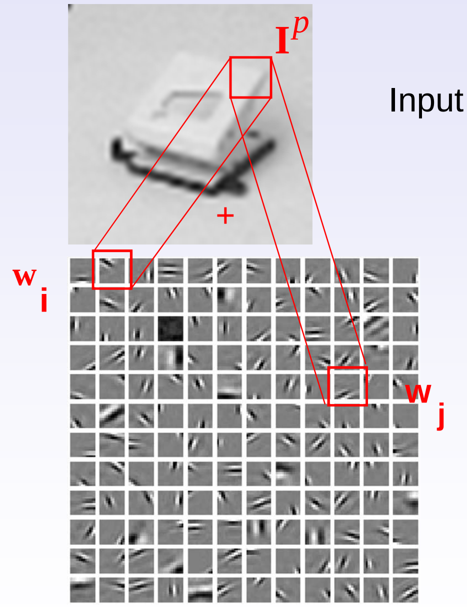
- Generating an overcomplete basis
- Take much more basis vectors than input dimensionality
- Choose Phi as log $(1+x^2)$
- Apply to small patches taken from natural images



Olshausen & Field: Overcomplete code for natural image patches



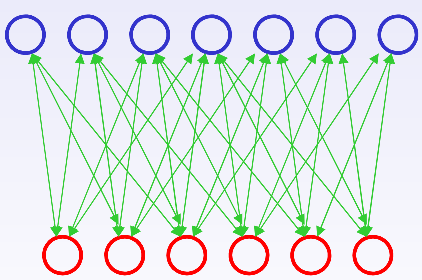
Input data: patches

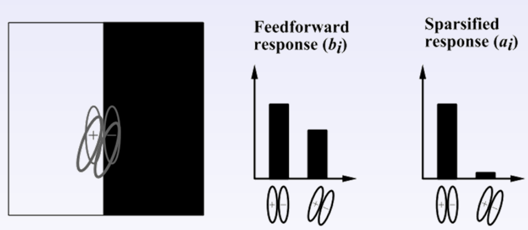


Learned Features

Interpretation of sparse coding

$$(\mathbf{I} - W\mathbf{s})^2 + \lambda|\mathbf{s}| \rightarrow \min$$





- Sparse coding introduces competition between feature detectors

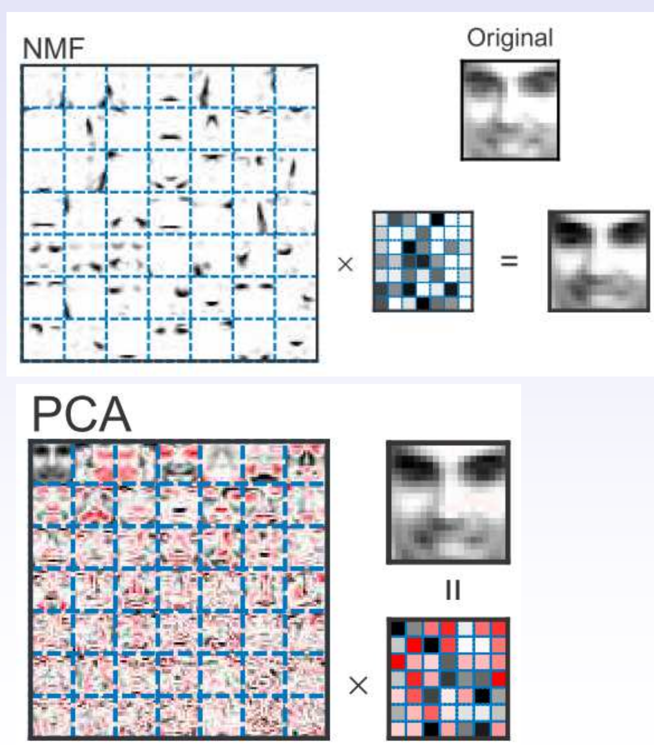
Nonnegative sparse coding

$$E_1 = \underbrace{\sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2}_{\text{Reconstruction error}} + \underbrace{\sum_p \sum_i \Phi(s_i^p)}_{\text{Sparsity-enforcing cost}}$$

- Constrain inputs, weights and coefficients to be positive or zero (≥ 0)
- Choose Φ as f(x)=λx
- λ controls the sparsity cost
- For the right number of basis vectors, this can solve the bars test
- See Practical exercise !

Nonnegative Matrix Factorization

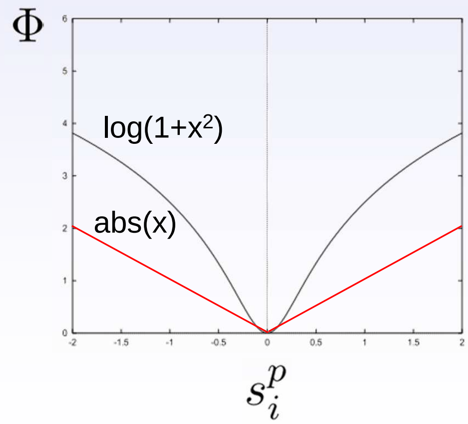
- Lee & Seung (1999)
- Positivity constraint for basis vectors and coefficients
- Multiplicative update rules
- No explicit sparsity cost



Sparse coding: Independent Component Analysis

$$E_1 = \underbrace{\sum_p ||\mathbf{I}^p - \sum_i s_i^p \mathbf{w}_i||^2}_{\text{Reconstruction error}} + \underbrace{\sum_p \sum_i \Phi(s_i^p)}_{\text{Sparsity-enforcing cost}}$$

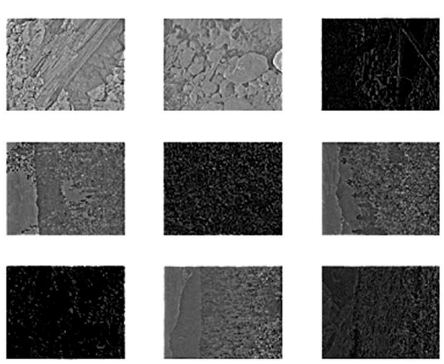
- ICA: Independent Component Analysis
  (Olshausen & Field 1996, Bell & Sejnowski 1998)
- Take same dimensionality as number of basis vectors
- Choose Phi appropriately
- Then this is equivalent to independent component analysis: find directions in the data, where the data is maximally independent

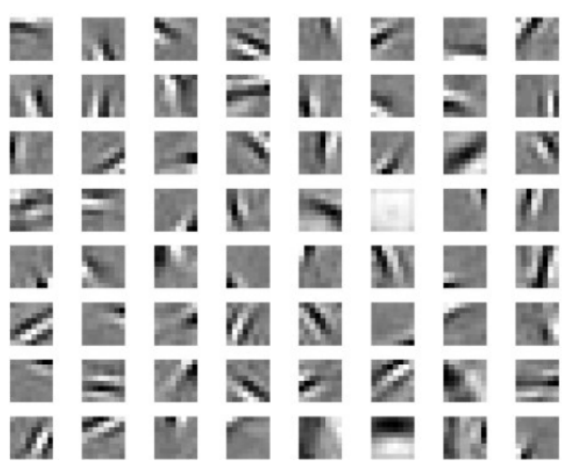


ICA

Training with Natural Images

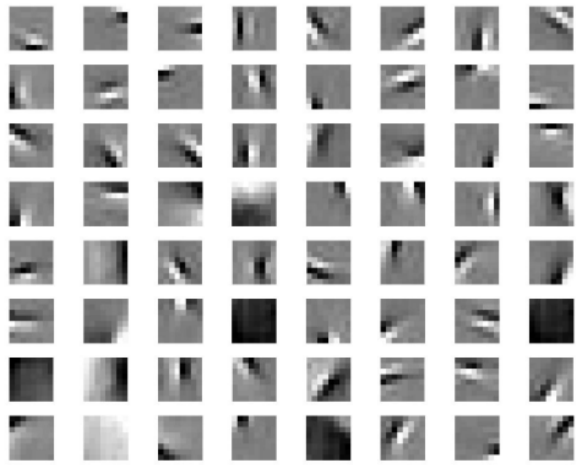
- ❑ Training: 10 images (512x512)
- ❑ 10,000 presentations
- ❑ Batch size: 100
- ❑ Basis Function: 16x16





Training with Face Database

- Training: 100 images (100x100)
- 10,000 presentations
- Batch size: 100
- Basis Function: 16x16



Training with Car Database

- Training: 200 images (128x128)
- 10,000 presentations
- Batch size: 100
- Basis Function: 16x16