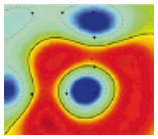
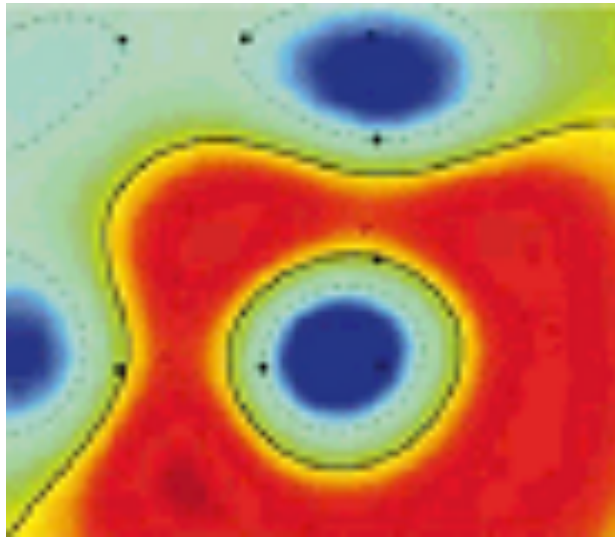


WiSe 2024/25

Deep Learning 1

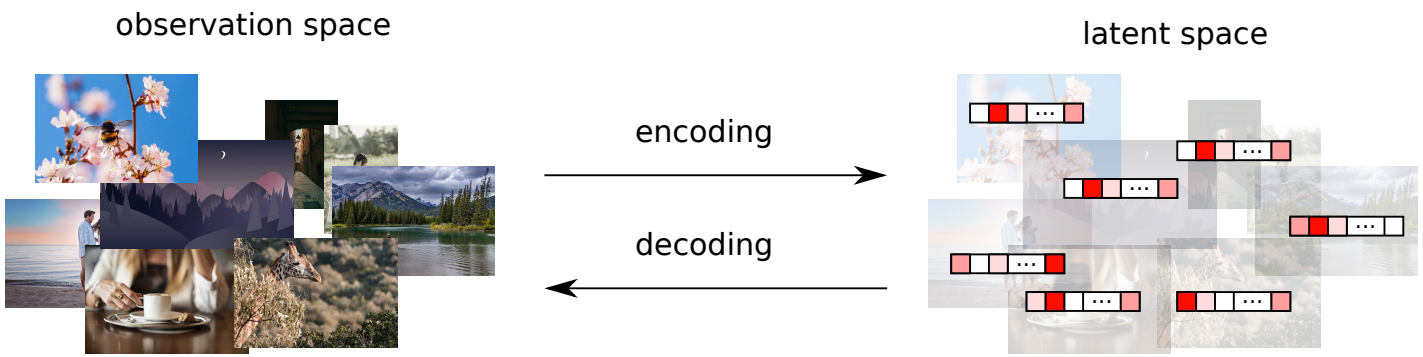
Technische Universität Berlin



Lecture 10

Autoencoders

Motivation



- **Goal:** Representing data in a way that is more efficient for some purpose, for example,
  - Compression, summarization
  - Correlating data to another data modality (e.g. text)
  - Identifying independent components (disentangling the data)
  - Extracting high-level (semantically related) features from the data

目标： 以更高效的方式表示数据，以便实现某些目的，例如：

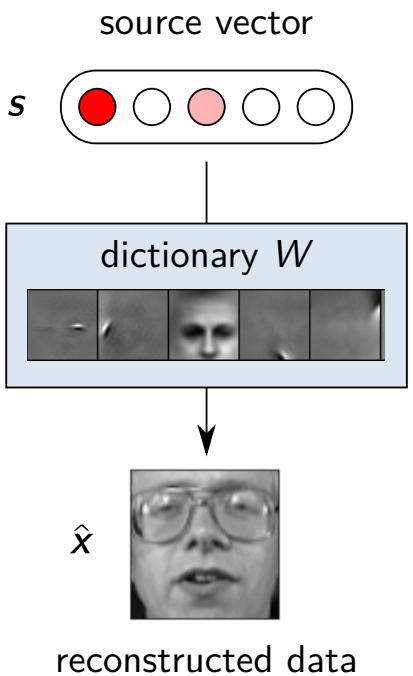
- 压缩、摘要
- 将数据与另一种数据模态（例如文本）关联
- 识别独立成分（解耦数据）
- 从数据中提取高级（语义相关）特征

Sparse Coding

稀疏编码（Sparse Coding）： 旨在将数据（例如图像）表示为具有许多零元素的向量。数据可以通过字典 W 从稀疏编码中重建。

稀疏编码的优点：

- **低存储成本（Low storage cost）：** 具有大量零元素的向量可以更紧凑地表示。例如，稠密形式的向量 $s = (12, 0, 7, 0, 0)$ 可以更紧凑地表示为索引-值对的集合，即 $s = \{0 \to 12; 2 \to 7\}$。
- **可解释性（Interpretable）：** 经验上，稀疏编码学习到的表示通常能够很好地解耦数据中的不同变化因素，例如，源信号可以对应于空间上定义良好的视觉模式。

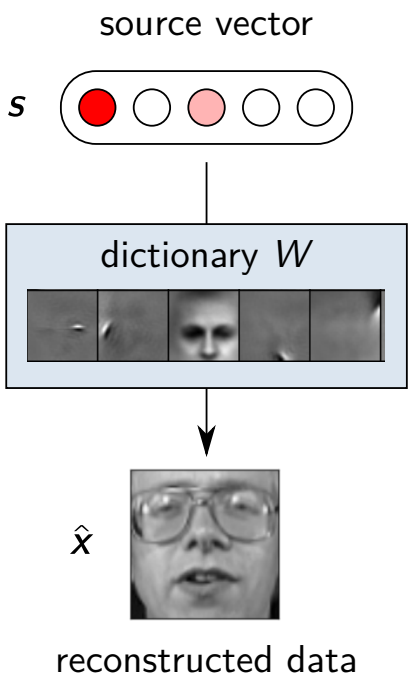


Sparse coding aims to represent data (e.g. an image) as a vector with many elements that are zero. The data can be reconstructed from the sparse code using a dictionary W.

Sparse coding has several advantages:

- **Low storage cost:** A vector with many zeros can be repesented compactly. E.g. the vector given in dense form as $\boldsymbol{s} = (12, 0, 7, 0, 0)$ can be written more compactly as a collection of index-value pairs, i.e. $\boldsymbol{s} = \{0 \to 12; 2 \to 7\}$.
- **Interpretable:** Empirically, the representation learned by sparse coding often disentangles well the different factors of variation in the data, e.g. sources correspond to spatially well-defined visual patterns.

Sparse Coding



Linear sparse coding

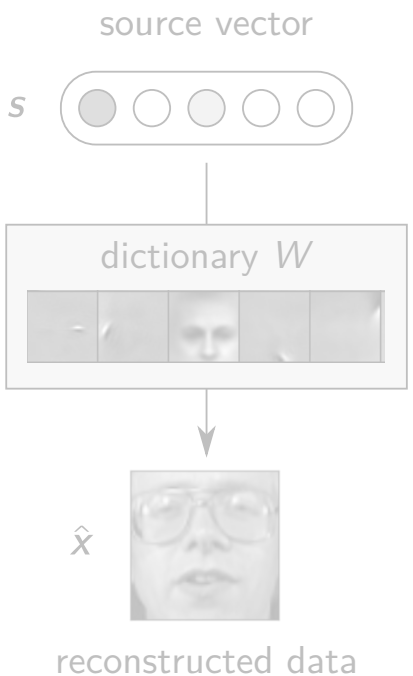
- Assume $\boldsymbol{x} \in \mathbb{R}^d$, and its correspond source code is a sparse vector $\boldsymbol{s} \in \mathbb{R}^h$.
- Using the dictionary W (a matrix of size $d \times h$), linear sparse coding reconstructs approximately the data from the source code as:

$$\hat{\boldsymbol{x}} = W\boldsymbol{s} \qquad \text{(dense formulation)}$$

$$= \sum_{i|s_i \neq 0} W_{:,i}\, s_i \qquad \text{(sparse formulation)}$$

- *Question:* how do we *learn* the dictionary W and the sparse code $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ associated to some dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$.

Sparse Coding Objectives



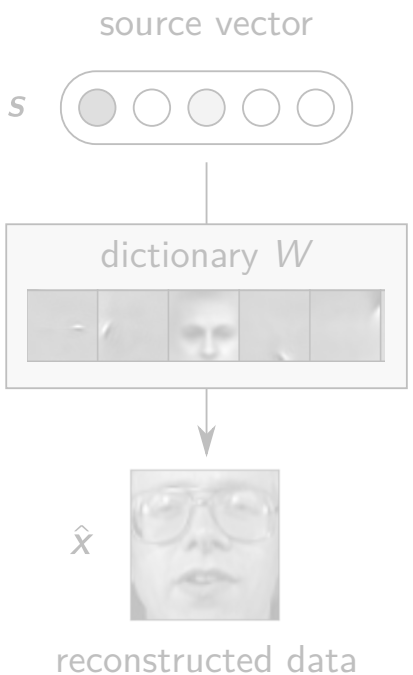
Linear sparse coding (L_0 formulation)

- Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^d$ be a dataset and $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N \in \mathbb{R}^h$ be the representations (sources) of each data point. Let $W \in \mathbb{R}^{d \times h}$ be a dictionary that reconstructs the data from the sources.
- An objective function that implements the sparse coding idea is given by:

$$\min_{W, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_N} \quad \frac{1}{N} \sum_{i=1}^{N} \Big[\underbrace{\left\| \boldsymbol{x}_i - W \boldsymbol{s}_i \right\|^2}_{\text{reconstruction}} + \underbrace{\lambda \| \boldsymbol{s}_i \|_0}_{\text{sparsity}} \Big]$$

where the $|| \cdot ||_0$ is the 0-"norm" (a generalization of the p-norm with $p = 0$). The 0-"norm" counts the non-zero elements in the vector, and therefore penalizes non-sparse solutions. (*Problem:* $\| \boldsymbol{s}_i \|_0$ is nonconvex and is not differentiable.)

Sparse Coding Objectives



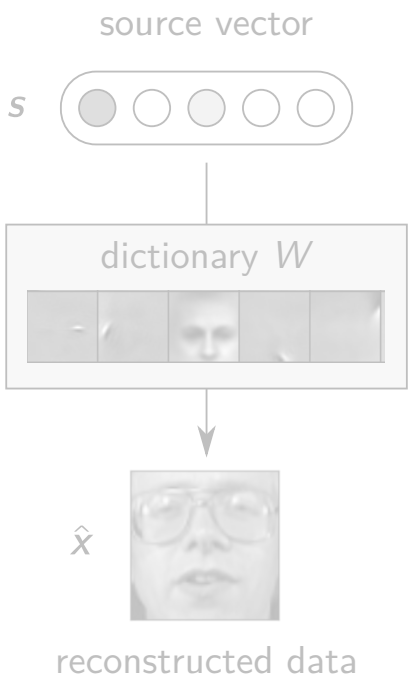
Linear sparse coding (L_1-relaxation)

- *Idea:* Replace $\|\boldsymbol{s}_i\|_0$ by $\|\boldsymbol{s}_i\|_1$:

$$\min_{W, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_N} \frac{1}{N} \sum_{i=1}^{N} \Big[\underbrace{\|\boldsymbol{x}_i - W\boldsymbol{s}_i\|^2}_{\text{reconstruction}} + \underbrace{\lambda \|\boldsymbol{s}_i\|_1}_{\text{sparsity}} \Big]$$

- *Advantage:* Given a dictionary W, finding the sources $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ of all data points is now a convex optimization problem (because $\sum_i \|\boldsymbol{s}_i\|_1$ is convex). The objective is also differentiable almost everywhere.
- *Limitation:* The objective is not the same anymore. $\Rightarrow$ Using $\|\boldsymbol{s}_i\|_1$ will produce a sparse solution but not a maximally sparse solution.
- *Another problem:* The second term can be made arbitarily small by scaling up W and scaling down $\boldsymbol{s}_i$ accordingly, i.e. the sparsity penalty is not effective.

Sparse Coding Objectives



Linear sparse coding (L_1 relaxation)

- *Idea:* Add a regularization term on the weights.

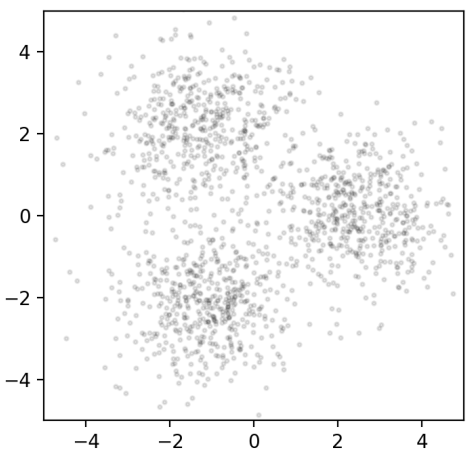
$$\min_{W, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_N} \quad \frac{1}{N} \sum_{i=1}^{N} \Big[\underbrace{\left\| \boldsymbol{x}_i - W \boldsymbol{s}_i \right\|^2}_{\text{reconstruction}} + \underbrace{\lambda \| \boldsymbol{s}_i \|_1}_{\text{sparsity}} \Big] + \eta \| W \|_F^2$$

- This resolves the scaling problem between $\boldsymbol{s}_i$ and W, by penalizing large Ws.

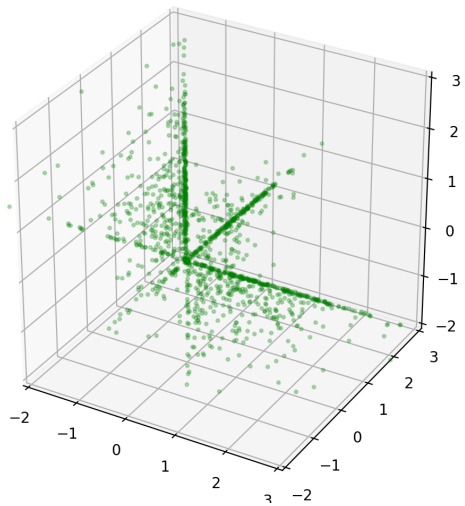
Toy example

Consider some dataset $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^2$ and we perform source coding with sources $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N \in \mathbb{R}^3$. After optimization, most data points are sparse in source space (aligned with the coordinate system), but still reconstruct reasonably well the data.

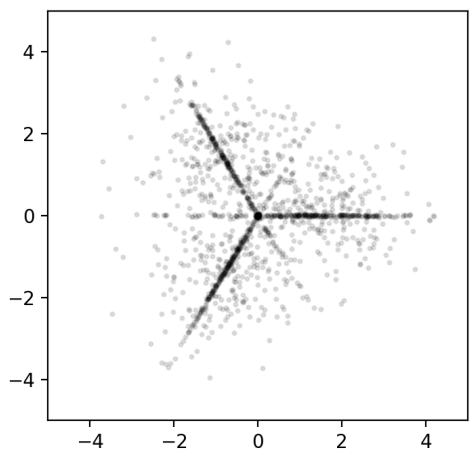
input data



optimized sources



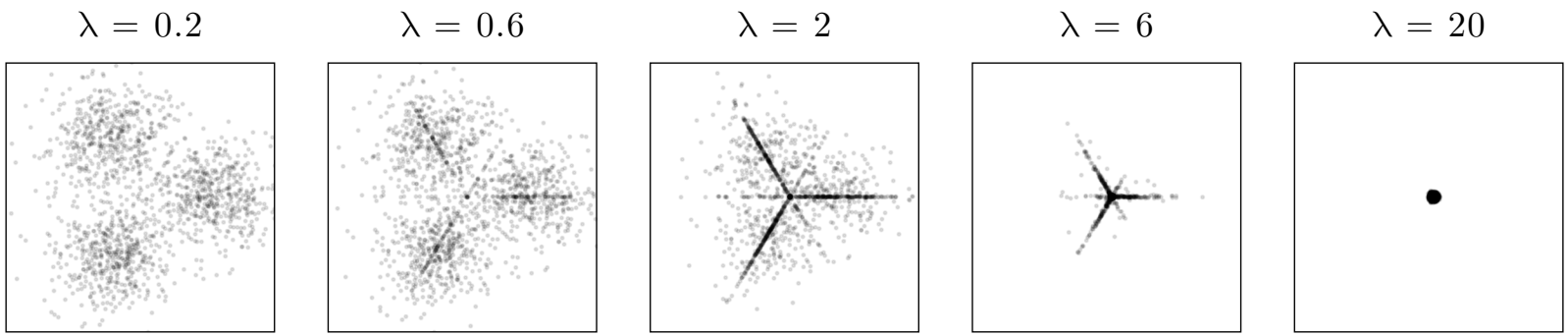
reconstructed data



Toy example

稀疏表示（Sparse Representation） 是指使用较少的非零元素来表示数据的一种方式。换句话说，它通过只使用少量的基本成分（或者称为“字典基”）来表示复杂的信号或数据，从而达到数据压缩、降维或特征提取的目的。

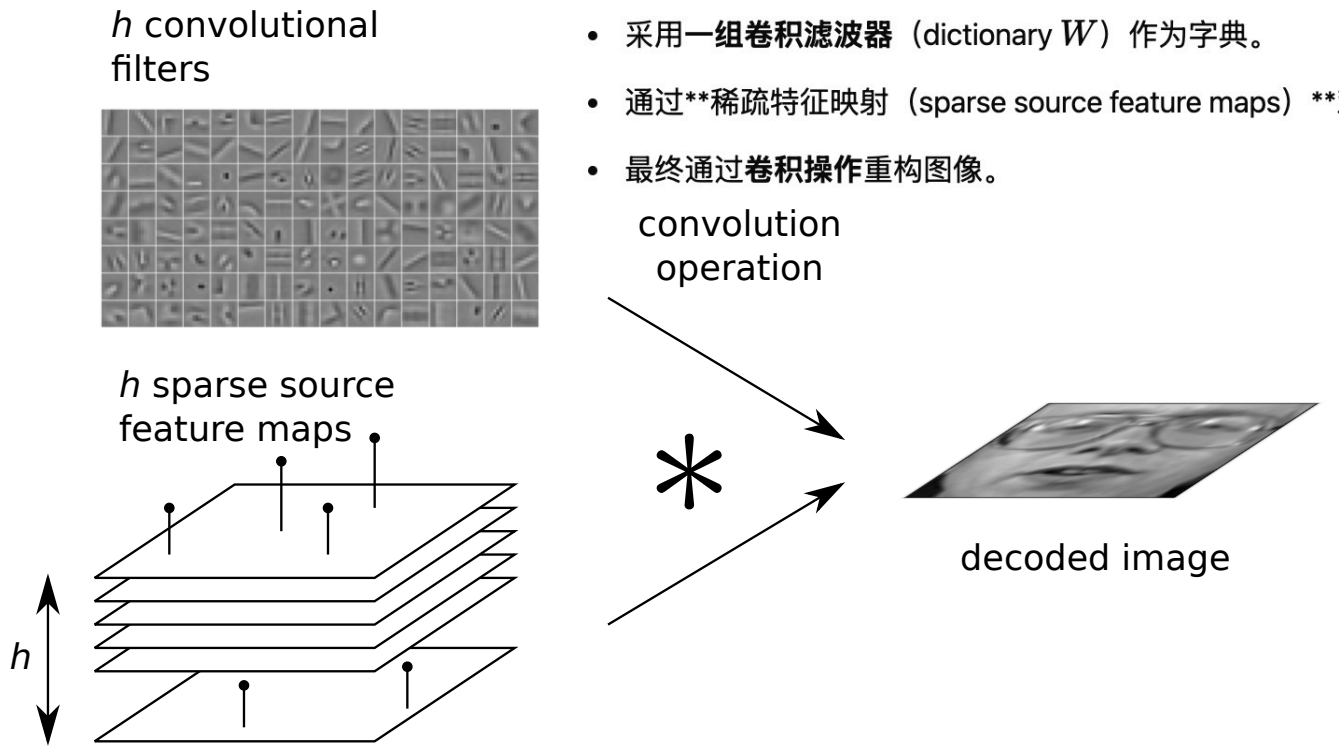
Effect of the sparsity parameter λ on the reconstructed data:



- A low parameter λ perfectly reconstructs the data but does not produce a sparse representation.
- A high parameter λ produces very sparse representation but does not reconstruct the data.
- A intermediate value of λ produces the desired tradeoff.

Convolutional Sparse Coding

Idea: Code images as the convolution of multiple sparse feature maps. Dictionary W is now a set of convolutional filters.



The same convolution filter can be used for any location in pixel space ($\rightarrow$ statistically efficient).

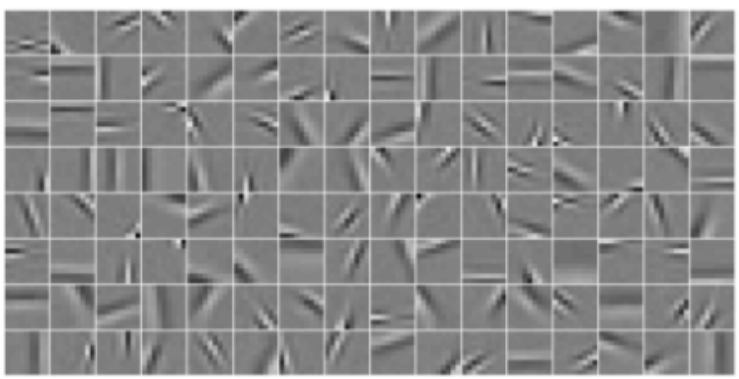
Convolutional Sparse Coding

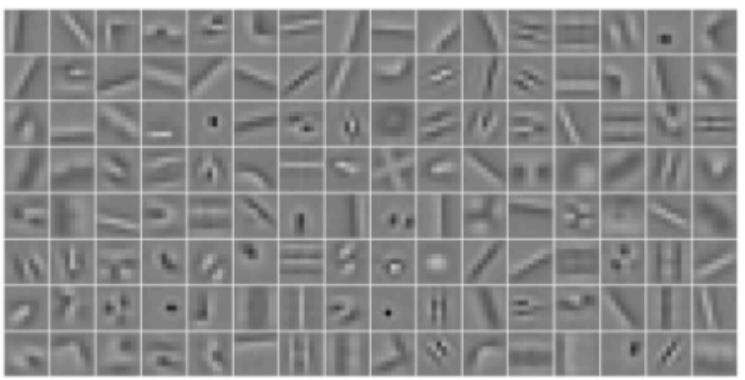
Standard Sparse Coding:

$\boldsymbol{s}_i$ is a vector, W is a matrix, and $\boldsymbol{x}_i = W \cdot \boldsymbol{s}_i$.

Convolutional Sparse Coding:

$\boldsymbol{s}_i$ is a collection of images, W is a collection of convolution filters, and $\boldsymbol{x}_i = \sum_{j=1}^{h} W_j * \boldsymbol{s}_{ij}$





(Kavukcuoglu'10, Learning Convolutional Feature Hierarchies for Visual Recognition)

Observation: Standard sparse coding learns many replicas of the same filter at different locations. Convolutional coding doesn't have to do this, and instead learns a richer set of features.

- 标准稀疏编码通常会在不同位置学习多个相似的滤波器（即同一个特征会被重复学习）。
- 而卷积稀疏编码（CSC）能够共享卷积滤波器，使其能学习到更丰富的特征，并提高效率。

Topological Sparse Coding

Additional constraint: Sources dimensions $\boldsymbol{s}_{:,1}, \ldots, \boldsymbol{s}_{:,h}$ must correlate on some predefined 2D grid (topology).

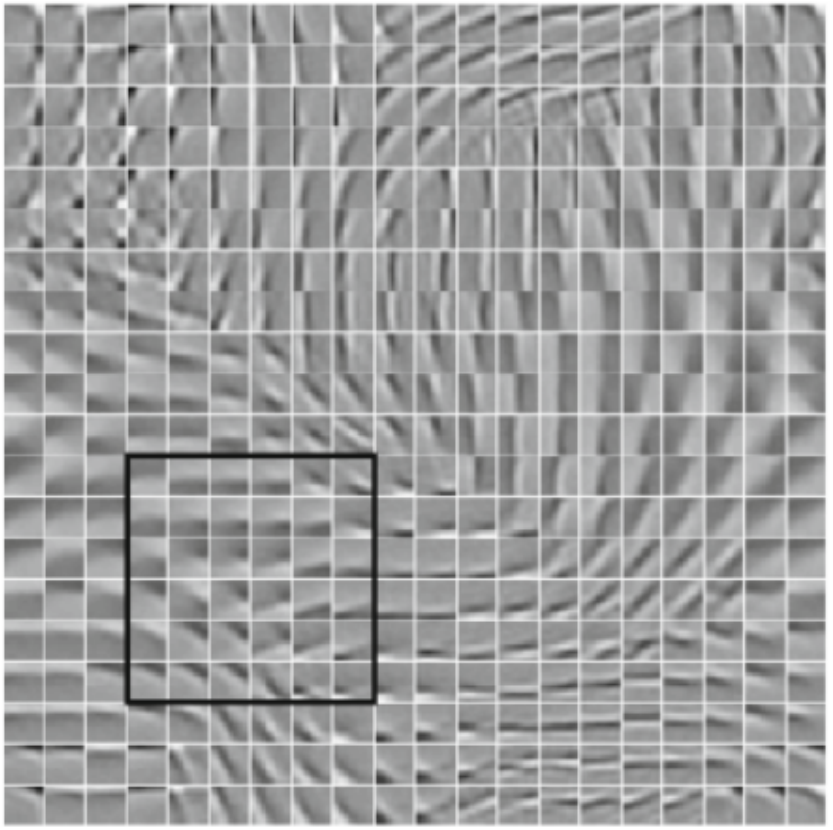


Image source: Yann LeCun, 2012, Learning Invariant Feature Hierarchies

Implementing Sparse Coding

Observation: The total number of variables to optimize is $N \times h$ for the sources and $d \times h$ for the dictionary. For large N, this may not fit in memory or lead to slow optimization. Therefore, two different approaches can be considered depending on whether N is small or large.

Batch algorithm
(works well for small N).

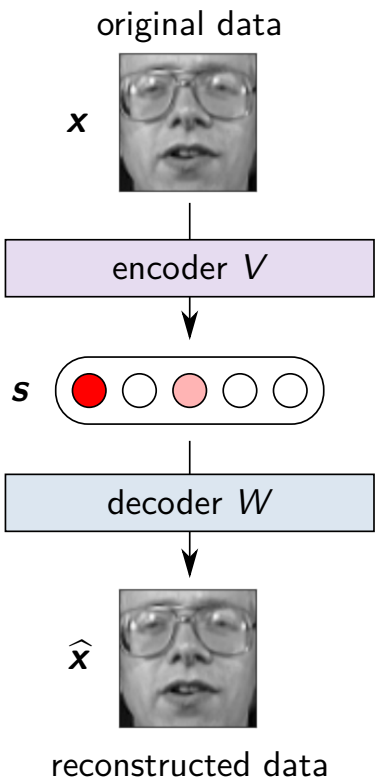
```
Initialize W, s_1, ..., s_N
while not converged do
  update s_1, ..., s_N
  update W
end while
```

Online algorithm
(works well for N large).

```
Initialize W
while W not converged do
  Choose i randomly
  infer s_i
  update W approximately from s_i
end while
```

Many possible implementations of sparse coding, including efficient ones (cf. Lee et al. 2006, Efficient sparse coding algorithms).

Auto-Encoders



Idea: Facilitate training with an encoding function

$$\boldsymbol{s}_i = V\boldsymbol{x}_i$$

and learn this encoding function from the data.

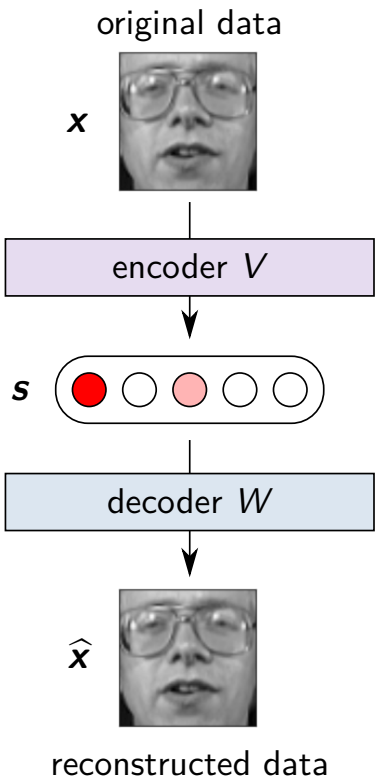
Objective becomes:

$$\min_{W,V} \quad \frac{1}{N}\sum_{i=1}^{N}\Big[\underbrace{\|\boldsymbol{x}_i - WV\boldsymbol{x}_i\|^2}_{\text{reconstruction}} + \underbrace{\lambda\|V\boldsymbol{x}_i\|_1}_{\text{sparsity}}\Big] + \eta\|W\|_F^2$$

It only depends on the matrices W and V. These matrices can be learned via (stochastic) gradient descent.

Problem: The linear encoding function may not have sufficient representation power to produce sparse representations.

Auto-Encoders with Sparsifying NonLinearity



Idea: Use a *nonlinear* encoding function

$$\boldsymbol{s}_i = \max(0, V\boldsymbol{x}_i)$$

where $\max(0, \cdot)$ is applied element-wise. This function produce values exactly zero for every negative input ($\rightarrow$ helps sparsity!).

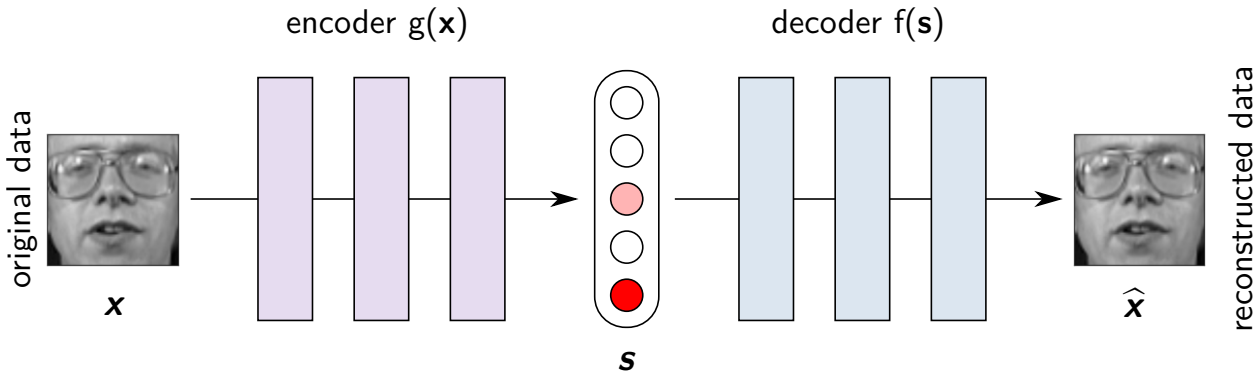
The objective becomes:

$$\min_{W,V} \frac{1}{N} \sum_{i=1}^{N} \Big[\underbrace{\big\|\boldsymbol{x}_i - W \max(0, V\boldsymbol{x}_i)\big\|^2}_{\text{reconstruction}} + \underbrace{\lambda \|\boldsymbol{s}_i\|_1}_{\text{sparsity}} \Big] + \eta \|W\|_F^2$$

To avoid creating dead units, we can add an 'entropy term' that forces each unit to activate at least a few times.

Auto-Encoders with Multiple Layers

For complex dataset, components can be better represented using more complex encoding and decoding functions, e.g. each of them consisting of multiple layers.



$$\min_{f,g} \frac{1}{N} \sum_{i=1}^{N} \left\| \boldsymbol{x}_i - f \circ g(\boldsymbol{x}_i) \right\|^2$$

- The linear projections are replaced by general functions f and g (typically multilayer network).
- This allows to learn better codes compared to shallow autoencoders.
- A deep autoencoder can be trained using error backpropagation.

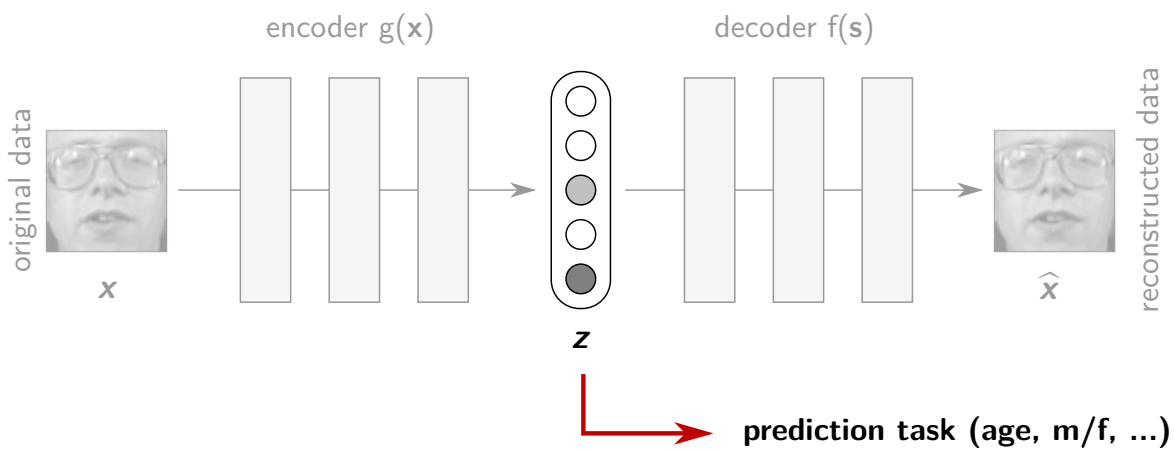
Learning Invariant Representations

学习不变表示（Learning Invariant Representations）

▶ 在实际应用中，稀疏编码并准确地重建/生成数据可能并不是最终目标。

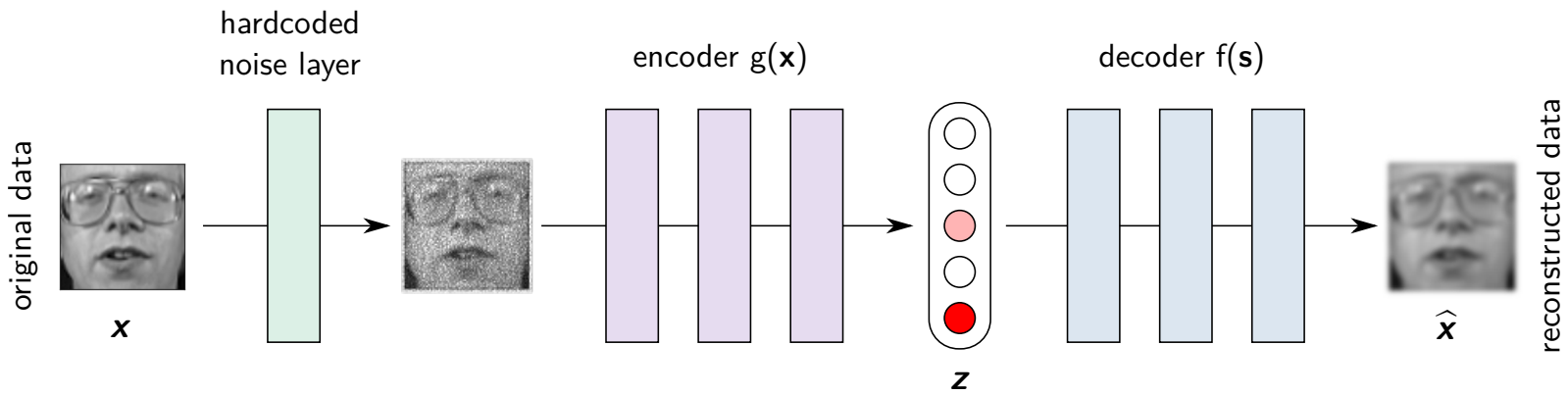
▶ 我们可能希望学习一种满足额外属性的表示，例如：**对任务无关的变化因素保持不变**（如小范围的平移或旋转）。

- In practice, sparsely encoding and accurately reconstructing/generating the data may not be the end goal.
- We may instead want to learn a representation that satisfies additional properties, e.g. being invariant to factors of variations that are irrelevant for the task (e.g. small translations or rotations).



Denoising Autoencoder

The denoising autoencoder can be seen as a standard deep autoencoder to which we have added a first layer that adds noise to the input.



$$\min_{f,g} \frac{1}{N} \sum_{i=1}^{N} \left[\left\| \boldsymbol{x}_i - f \circ g \circ \text{noise}(\boldsymbol{x}_i) \right\|^2 \right]$$

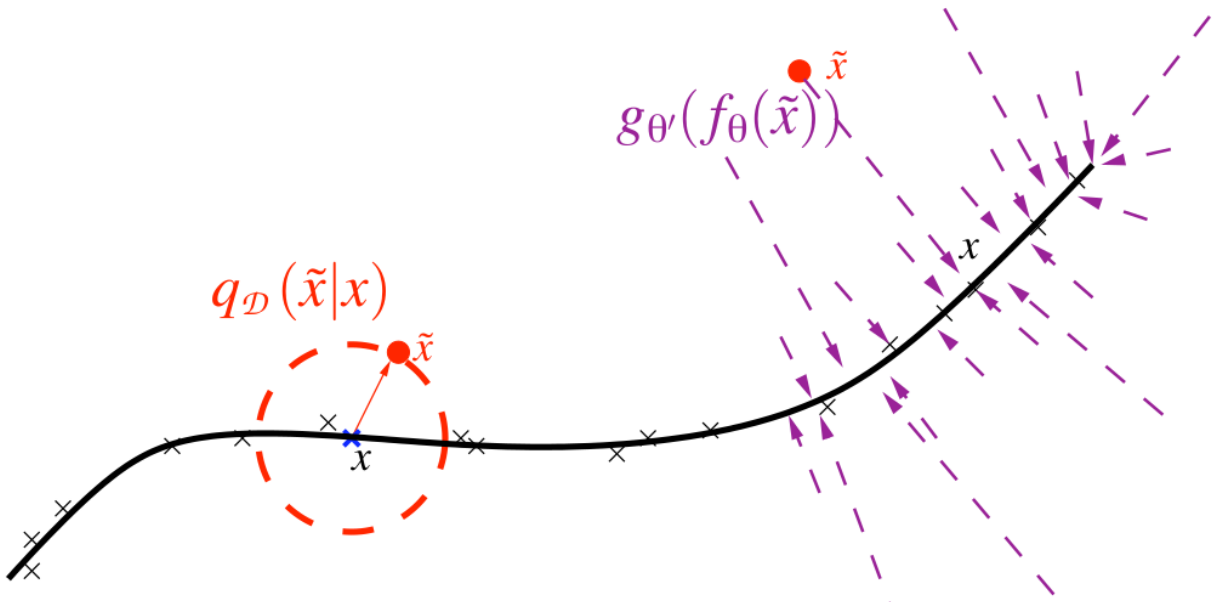
Reference publication: Vincent et al. 2010, Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion, JMLR

Local Invariance with Denoising Autoencoders

利用去噪自编码器实现局部不变性（Local Invariance with Denoising Autoencoders）

说明： 训练**去噪自编码器（denoising autoencoder）**能够学习到对局部微小变化不敏感的表示 $f(x)$，从而在预测任务中更加稳健（通过增大分类边界与数据流形之间的间隔来提升分类性能）。

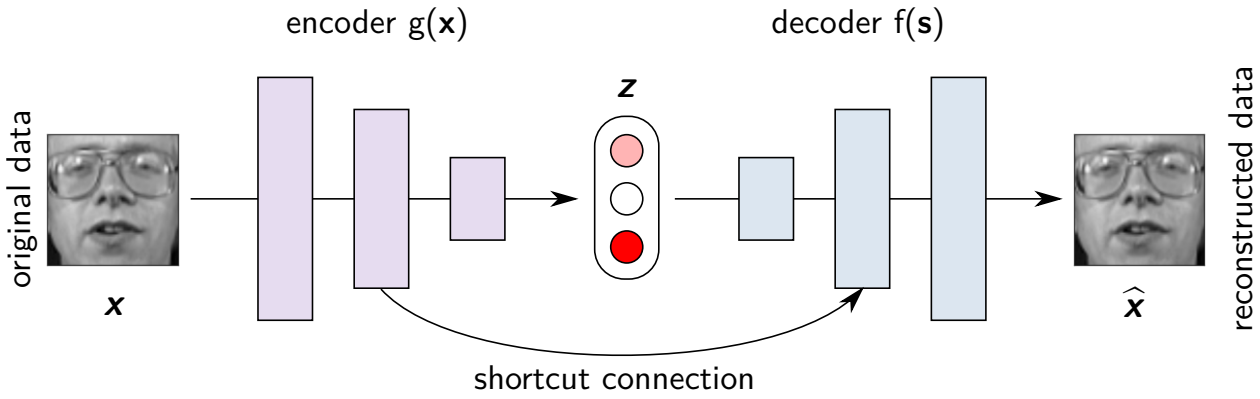
紫色箭头：去噪自编码器学习到的梯度方向，使得数据点更接近流形。



流形学习（Manifold Learning）：自编码器将扰动的数据点 $\tilde{x}$ 映射回数据流形，从而学习到对局部扰动不敏感的特征表示。

Training a denoising autoencoder produces a representation $f(\boldsymbol{x})$ that is insensitive to small local variations and therefore more useful for prediction tasks (promotes large margins between classification boundaries and data manifolds).

Learning Abstract Representations



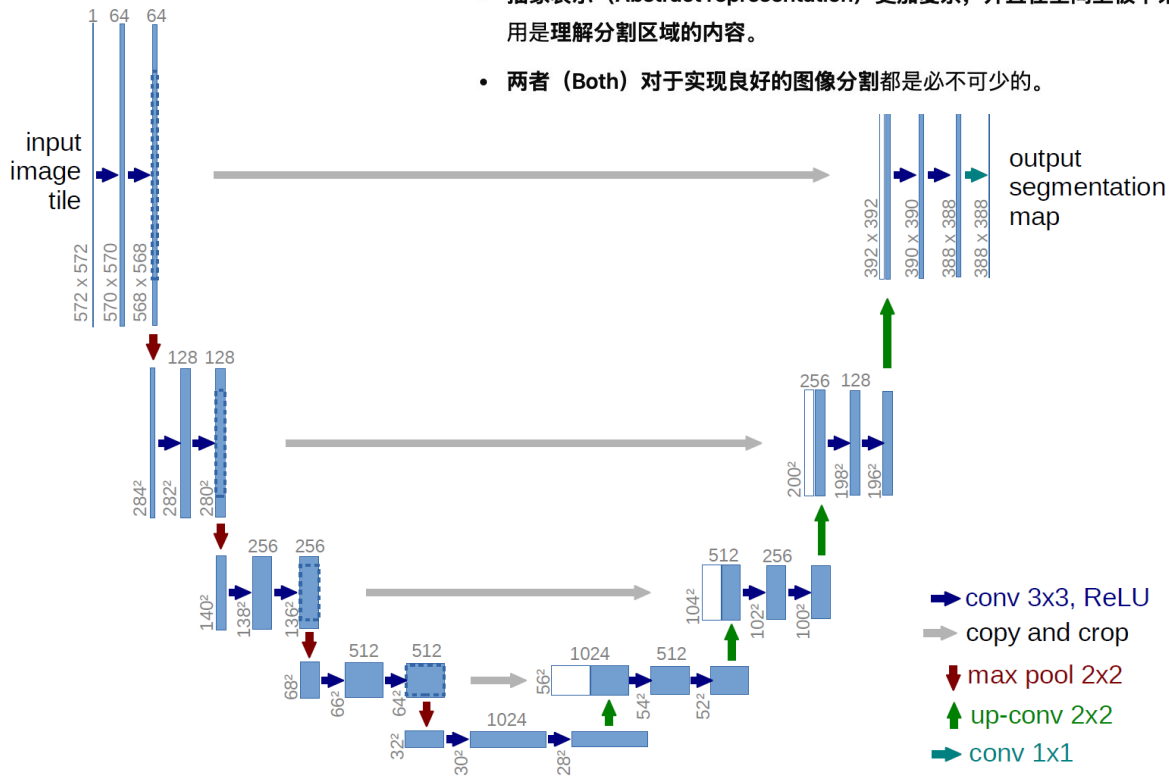
- Low-level information (that doesn't require many layers of abstraction) can flow through a shortcut connection.
- Only information that requires more layers (often abstract concepts) is retained in the representation.
- Models that implement this (or similar) idea include the U-Net (Ronneberger 2015), the Deep Boltzmann Machine (Salakhutdinov 2009), and the Ladder Network (Rasmus 2015).

- 低级信息（不需要多层抽象的特征）可以通过快捷连接（shortcut connection）直接传播。
- 只有需要更深层次抽象的信息（通常是高级概念）才会保留在表示中。

Shortcut Connections for Image Segmentation

U-Net Model: *Shortcut connections* finely encode the boundaries of the segments. *Abstract representation* is more complex and spatially downsampled, and serves to understand the segment's content. *Both* are necessary to achieve good segmentation.

- 快捷连接（Shortcut connections）能够精细地编码分割区域的边界。
- 抽象表示（Abstract representation）更加复杂，并且在空间上被下采样（downsampled），其作用是理解分割区域的内容。
- 两者（Both）对于实现良好的图像分割都是必不可少的。



Reference publication: Ronneberger et al. (2015), U-Net: Convolutional Networks for Biomedical Image Segmentation, MICCAI [2]

Shortcut Connections for Image Segmentation

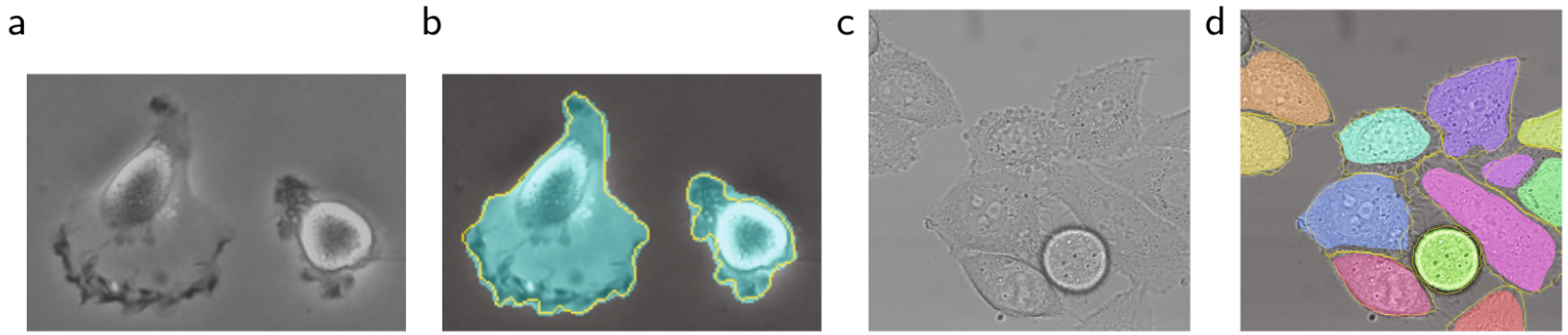
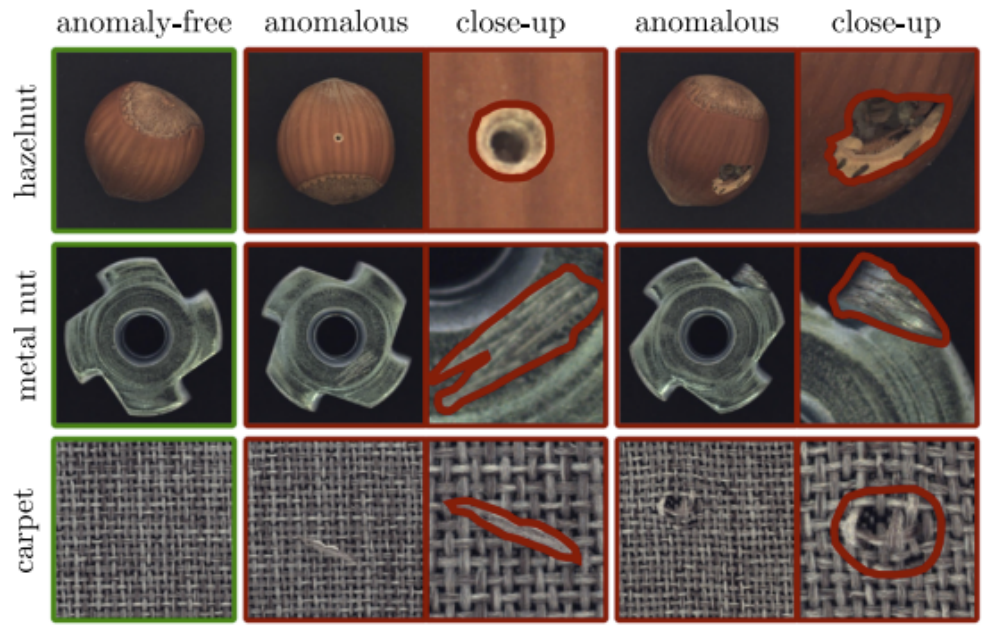


Fig. 4. Result on the ISBI cell tracking challenge. (**a**) part of an input image of the "PhC-U373" data set. (**b**) Segmentation result (cyan mask) with manual ground truth (yellow border) (**c**) input image of the "DIC-HeLa" data set. (**d**) Segmentation result (random colored masks) with manual ground truth (yellow border).

Compression

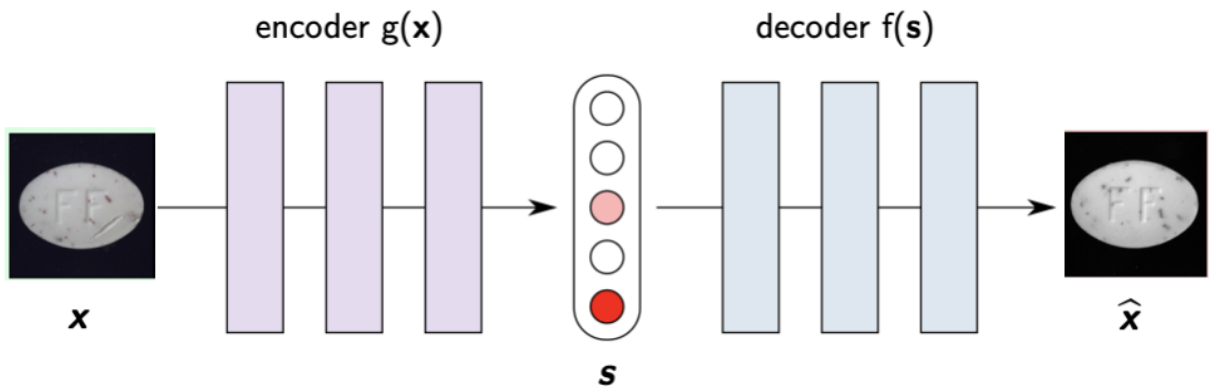
- Based on the size of the bottleneck dimension, we can control the tradeoff between storage size and loss factor during compression.
- This results in very efficient lossy compression for e.g videos.

Anomaly Detection



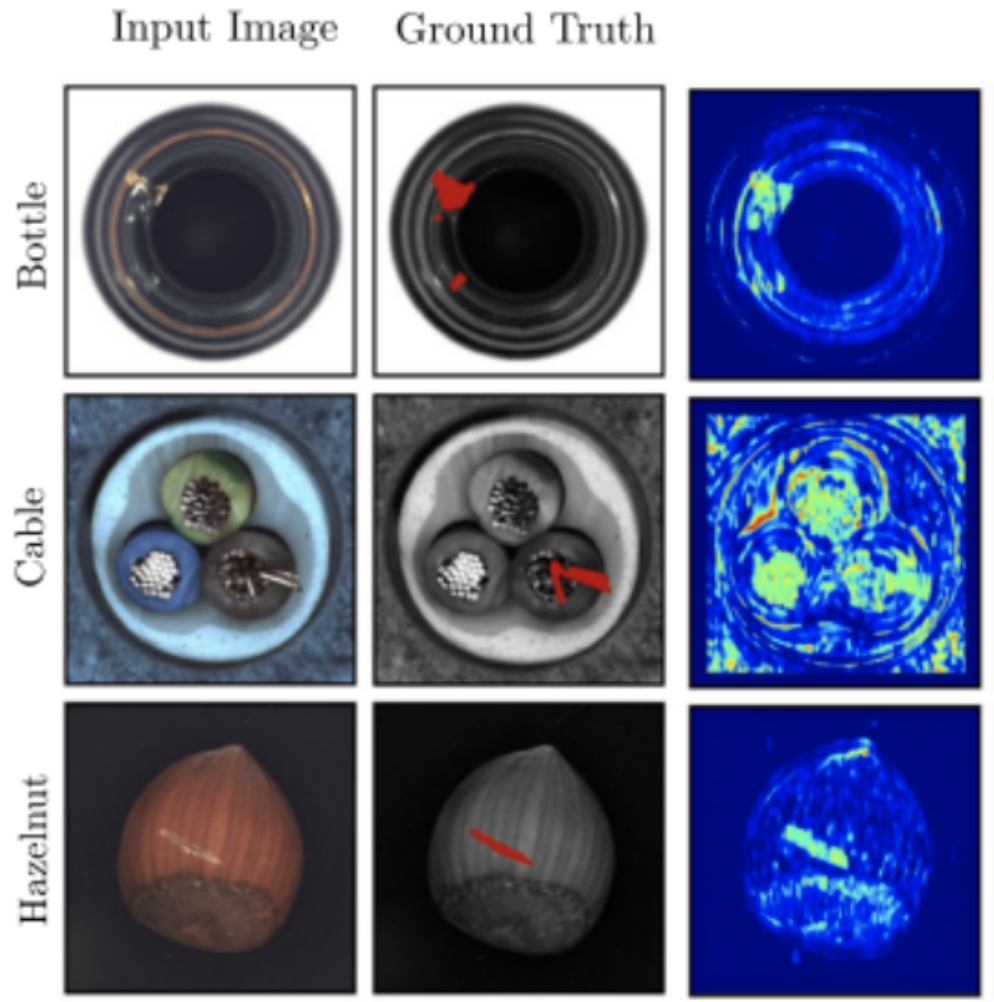
- Anomaly detection wants to find samples which are abnormal from a set of normal samples (e.g industry defects [1]).
- We want to assign a high anomaly score to anomalous samples.
- Ideally, we also want to highlight why the sample is considered abnormal by the model.

Anomaly Detection



- Both reconstruction and anomaly detection can be carried out simultaneously by an autoencoder.
- The autoencoder is trained on the normal data. Thereby finding the common variations.
- We can use the reconstruction error $L(x, \tilde{x}) = \left\| x - f \circ g(\tilde{x}) \right\|^2$ as an anomaly score and the reconstruction error per pixel as a heatmap.

Anomaly Detection



- By visualizing the reconstruction error, we get an indication about the location of the anomaly.

Summary

- Autoencoders learn representations from *unsupervised* data.
- Autoencoders can be trained quickly and can be implemented as neural networks. Thus, they can make use of all neural network layers available such as convolution, pooling, etc.
- Autoencoders can serve different purposes, e.g. produce compact (sparse) representations, denoise the input data, semantic segmentation, restore missing pieces of an image and anomaly detection.

References

P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger.

The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection.

International Journal of Computer Vision, 129(4):1038–1059, 2021.

O. Ronneberger, P. Fischer, and T. Brox.

U-net: Convolutional networks for biomedical image segmentation.

In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.