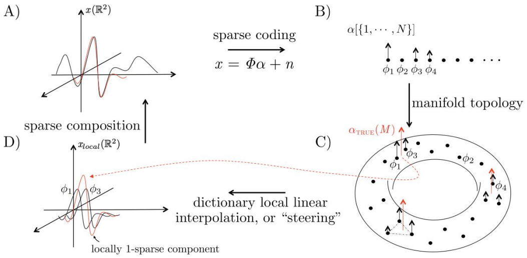
Unsupervised Learning of Image Manifolds with Mutual Information

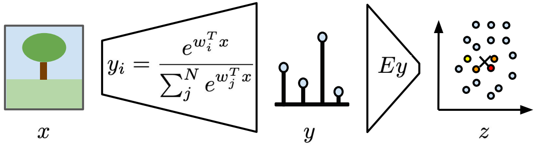
David A. Klindt*[1], Johannes Ballé*[2], Jonathon Shlens[2] & Eero P. Simoncelli[3]   1. University of Tübingen, 2. Google Research, 3. HHMI/NYU   correspondence: klindt.david@gmail.com

Abstract

In the space of natural images, continuous real world transformations such as rotations or deformations of objects give rise to a smooth, nonlinear low-dimensional manifold. It was recently proposed that sparse coding filters represent a discrete sampling of this manifold, and that the filters can be ordered in a low-dimensional embedding space that preserves the topology of the original data manifold [1]. The authors learn this representation by imposing a slowness prior [2], which straightens the trajectories of temporal input sequences [3]. The main motivation for our work is to build a model based on these ideas, but (1) with a feedforward architecture that allows for incorporation into existing CNN models, and (2) a contrastive objective function that doesn't rely on image reconstruction, allows for end-to-end training and operates on images rather than videos.


A)
$x(\mathbb{R}^2)$
sparse coding
$x = \Phi\alpha + n$
B)
$\alpha[\{1, \cdots, N\}]$
$\phi_1\ \phi_2\ \phi_3\ \phi_4$
manifold topology
sparse composition
D)
$x_{local}(\mathbb{R}^2)$
ϕ_1
ϕ_3
C)
$\alpha_{\text{TRUE}}(M)$
ϕ_1
ϕ_2
ϕ_3
ϕ_4
dictionary local linear interpolation, or "steering"
locally 1-sparse component

The Sparse Manifold Transform. (reproduced with permission from [1])

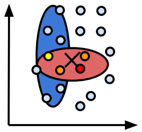


Proposed Model Layer. One layer consists of an overcomplete ($dim(x) < N$), (convolutional) expansion of the input signal, followed by divisive normalization (softmax), and then a projection onto a low-dimensional embedding space. The w_i and E are learned.

Defining Latent Distributions

Since the softmax output is positive and sums to one, it can be interpreted as a probability measure over a finite set, i.e. a categorical distribution with an associated R.V. $p(\hat{y}|x) \sim Cat(y(x))$. Across a batch of K inputs, we can encourage the model to use all channels equally by maximizing the marginal entropy $E[H[\hat{y}|x]] = H[\hat{y}]$.

In the embedding space, we can interpret the y as an activity pattern across the neurons e with center z. We can fit a factorized (computationally cheap) Normal distribution to this spatial pattern to define a R.V.:



$$\hat{z} \sim N(z, \mathbf{\Sigma}(x)\,|\,x)$$

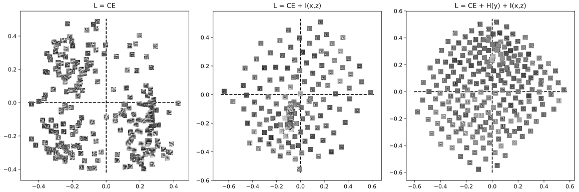
$$\mathbf{\Sigma}(x) = diag(\sigma_1^2(x), ..., \sigma_E^2(x))$$

$$\sigma_j^2(x) = \sum_{i=1}^{K} y_i(e_{ij} - z_j(x))^2$$

$$I[x;\hat{z}] \geq \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K}\log\frac{p(\hat{z}_i|x_i)}{\frac{1}{K}\sum_{j=1}^{K}p(\hat{z}_i|x_j)}\right]$$

$$= \mathbb{E}\left[\underbrace{\frac{-1}{K}\sum_{i=1}^{K}\log\sum_{j=1}^{K}\frac{1}{K}p(\hat{z}_i|x_j)}_{H[\hat{z}]} - \underbrace{\frac{-1}{K}\sum_{i=1}^{K}\log p(\hat{z}_i|x_i)}_{H[\hat{z}|x]}\right]$$

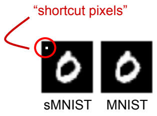
A topological organization in the embedding space is learned by maximizing the lower bound on the mutual information written above [4]. Combined, these terms encourage neurons that are equally utilized, but have locally sparse responses within the embedding space.



Learned Embedding. With *N=256* filters of size *9x9*, trained on the MNIST dataset and a 2 dimensional embedding space. Left) trained with cross-entropy (CE, *H*[*o*,*ô*] between true *o* and predicted labels *ô*). Center) additionally maximizing *I*[*x*,*ẑ*]. Right) additionally maximizing *H*[*ŷ*].

Results: Augmenting Classifiers to prevent Shortcut Learning

We trained a model with 2 layers (and a classification layer on top) on a modified version of the MNIST dataset that contains a "shortcut pixel" whose value indicates the class label [5]. Standard CNNs rely on this shortcut but our unsupervised layers force the model to learn the data distribution.



Model \ Dataset	sMNIST Train	sMNIST Test	MNIST Test
ResNet [5]	100	–	26.2
iCE fi-RevNet [5]	99.9	–	65.2
2-layer CNN	100	100	43.6
2-layer CNN + L2	100	100	79.1
Linear	100	100	72.9
Linear + L2	100	100	89.4
2-layer CNN (max $I[\hat{y}\|x]$)	99.3	97.7	80.3
2-layer CNN (max $H[\hat{y}] + I[\hat{z}\|x]$)	99.1	98	**93.6**

Table 1: **Models trained on shiftMNIST**. Accuracy in (%). 2-layer CNN consists of 2 x [Conv - ReLU - MaxPool], followed by [Linear – ReLU - Linear]. L2 regularization on all trainable parameters is cross-validated.

Conclusions and Outlook

- We propose a stackable model layer that maps the data manifold into a low dimensional embedding space
- Our model layer can be trained using a simple contrastive loss that learns a solution with highly structured filters
- Approximately uniform sampling of the data manifold, with clearly evident continuity of feature attributes
- First layer contains oriented filters laid out topologically, similar to the orientation tuning maps found in primate V1

References

[1] Chen, Y., Paiton, D., & Olshausen, B. (2018). The sparse manifold transform. In *Advances in Neural Information Processing Systems* (pp. 10513-10524).
[2] Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, *14*(4), 715-770.
[3] Hénaff, O. J., & Simoncelli, E. P. (2015). Geodesics of learned representations. *arXiv preprint arXiv:1511.06394*.
[4] Hénaff, O. J., Srinivas, A., De Fauw, J., Razavi, A., Doersch, C., Eslami, S. M., & Oord, A. V. D. (2019). Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*.
[5] Jacobsen, J. H., Behrmann, J., Zemel, R., & Bethge, M. (2018). Excessive invariance causes adversarial vulnerability. *arXiv preprint arXiv:1811.00401*.