Unsupervised Learning of Deep Feature Hierarchies from Unlabeled Video Data

Ross Goroshin

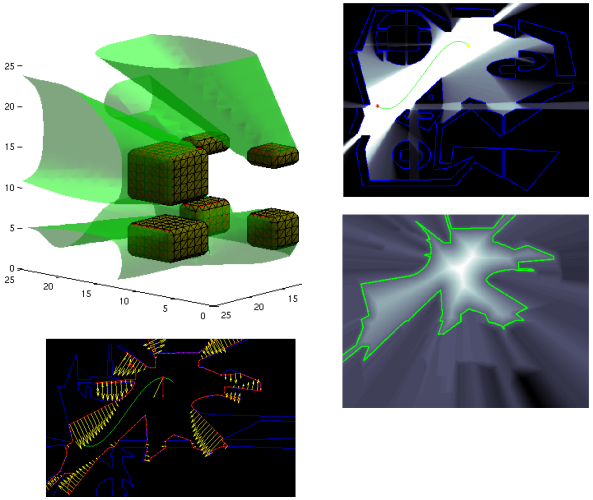
Courant Institute of Mathematical Sciences

September 2015

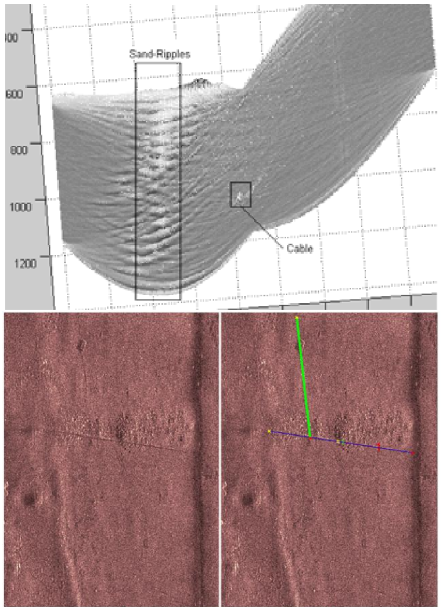
NEW YORK UNIVERSITY

Research Background

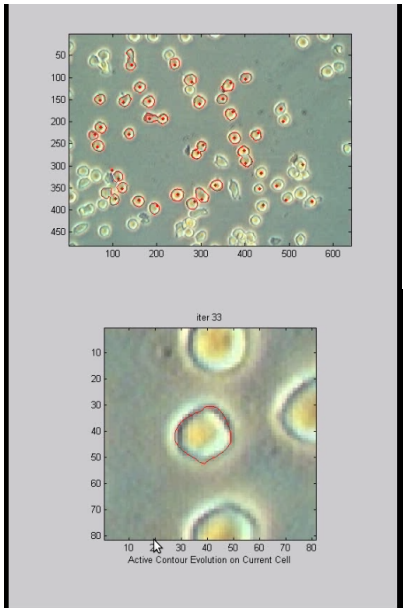
Visibility Path Planning via Variational Optimization



Cable Detection in Sonar Imagery



Segmentation with Active-Contours



Monocular Obstacle Detection

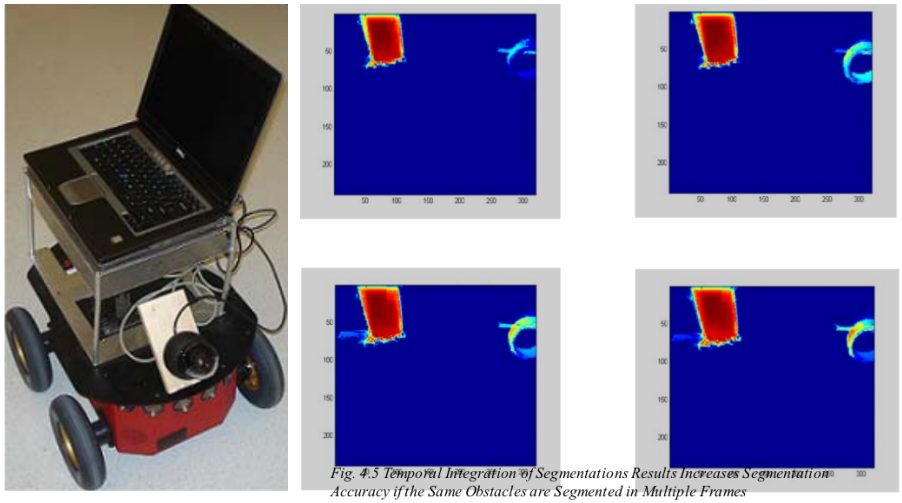
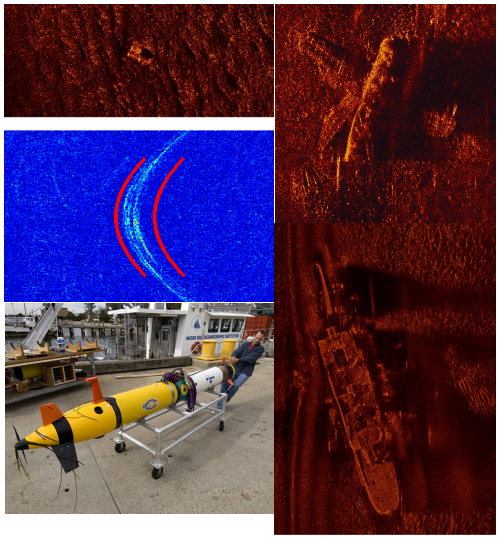


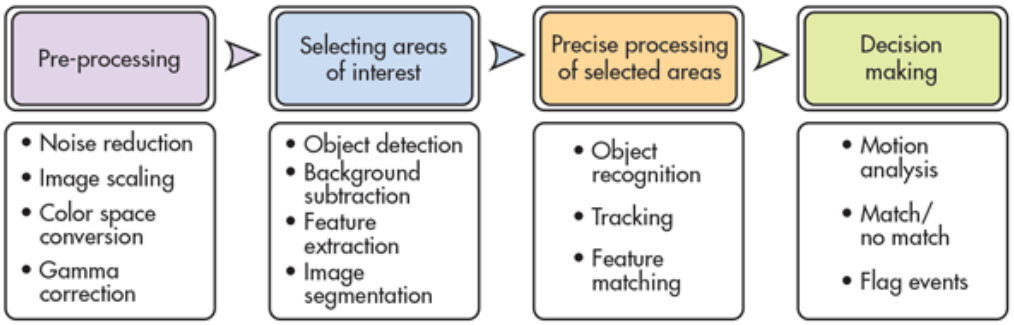
Fig. 4.5 Temporal Integration of Segmentations Results Increases Segmentation Accuracy if the Same Obstacles are Segmented in Multiple Frames

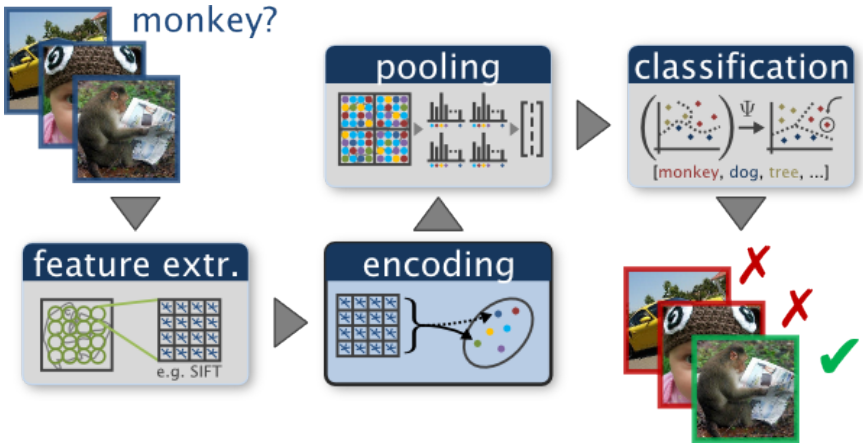
Synthetic Aperture Sonar Imaging



Deep Learning for Computer Vision

Classic Computer Vision Pipeline





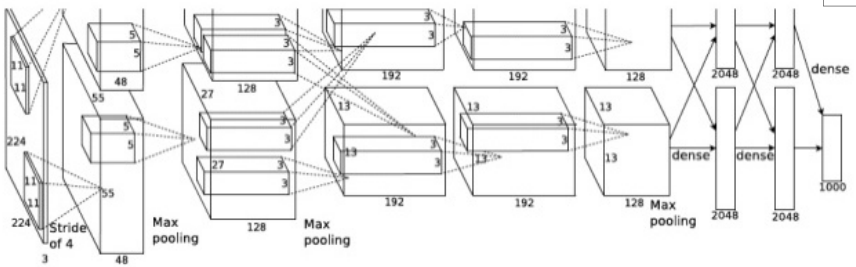
The engineer must manually plug all the "leaks" in the pipeline

http://cs.brown.edu/courses/cs143/

Computer Vision as of 2012

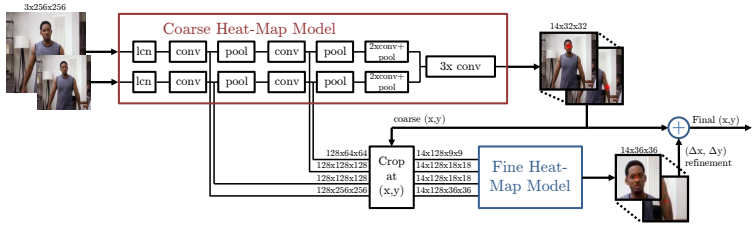
- Leverage machine learning to plug the leaks with data
- From engineering features $\rightarrow$ engineering *feature learning architectures*

Krizhevsky, *NIPS 2012*

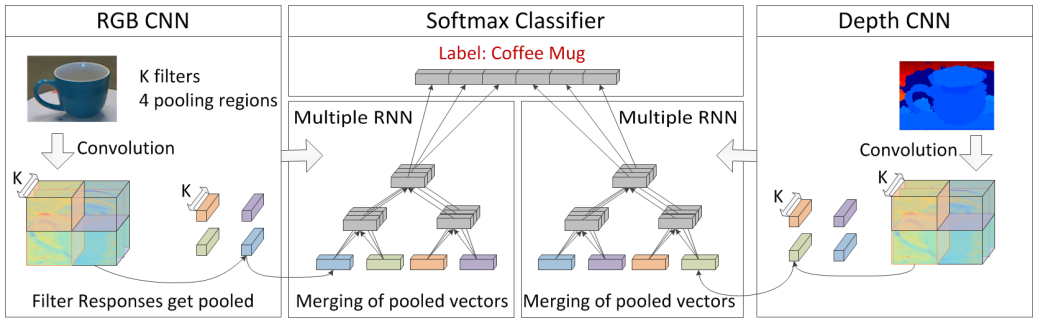


A. Krizhevsky, I. Sutskever, and G. Hinton,
ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012

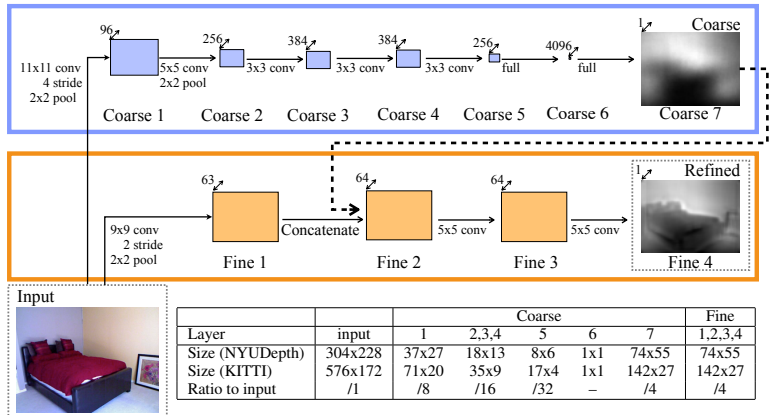
Tompson, Goroshin, Jain, LeCun, Bregler. *CVPR 2015*



Socher, Huval, Bhat, Manning, Ng. *NIPS 2012*



Eigen, Puhrsch, Fergus. *NIPS 2014*



Layer	input	Coarse					Fine
		1	2,3,4	5	6	7	1,2,3,4
Size (NYUDepth)	304x228	37x27	18x13	8x6	1x1	74x55	74x55
Size (KITTI)	576x172	71x20	35x9	17x4	1x1	142x27	142x27
Ratio to input	/1	/8	/16	/32	–	/4	/4

...and many more

Example: Human Pose Estimation

Tompson, Goroshin, Jain, LeCun, Bregler. ***CVPR 2015***

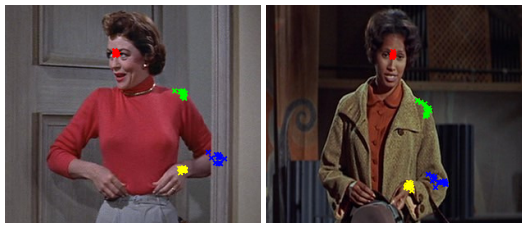
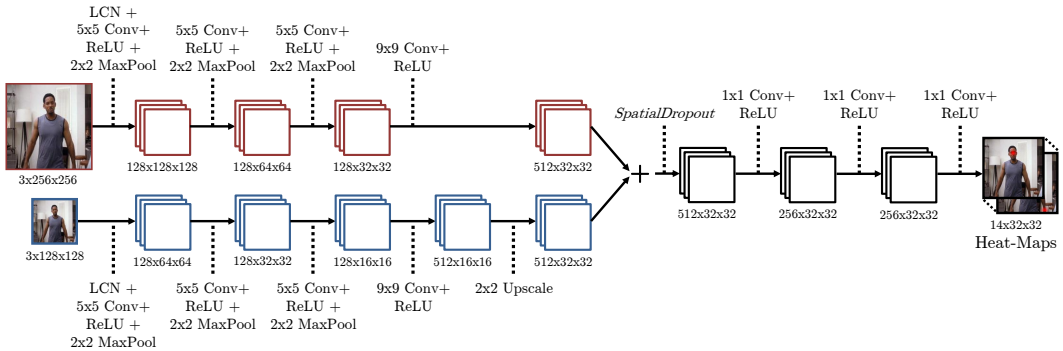
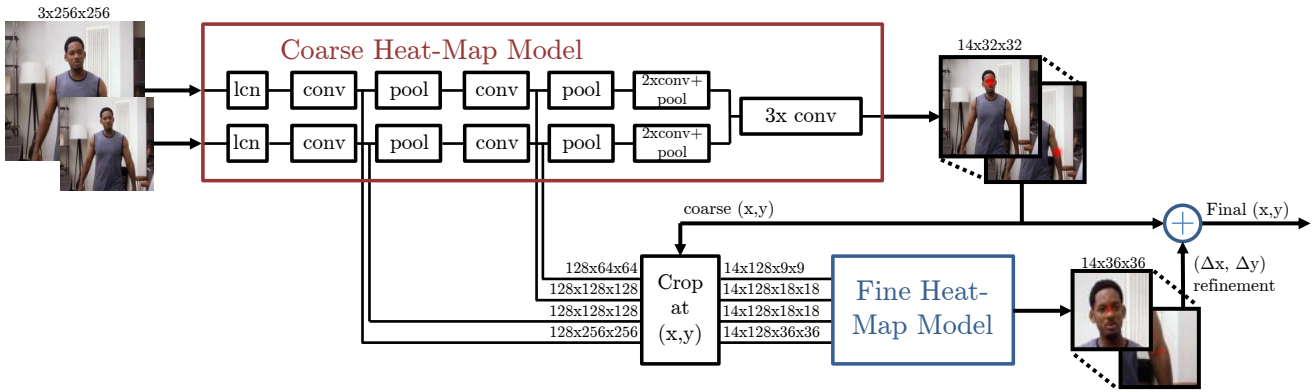


Figure 10: User generated joint annotations

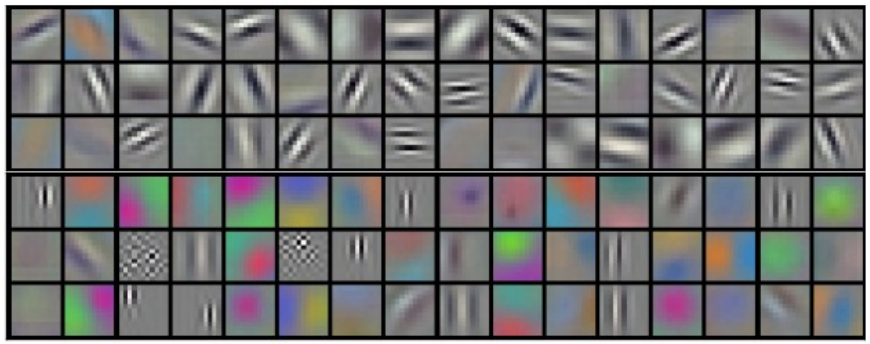




Unsupervised Feature Learning

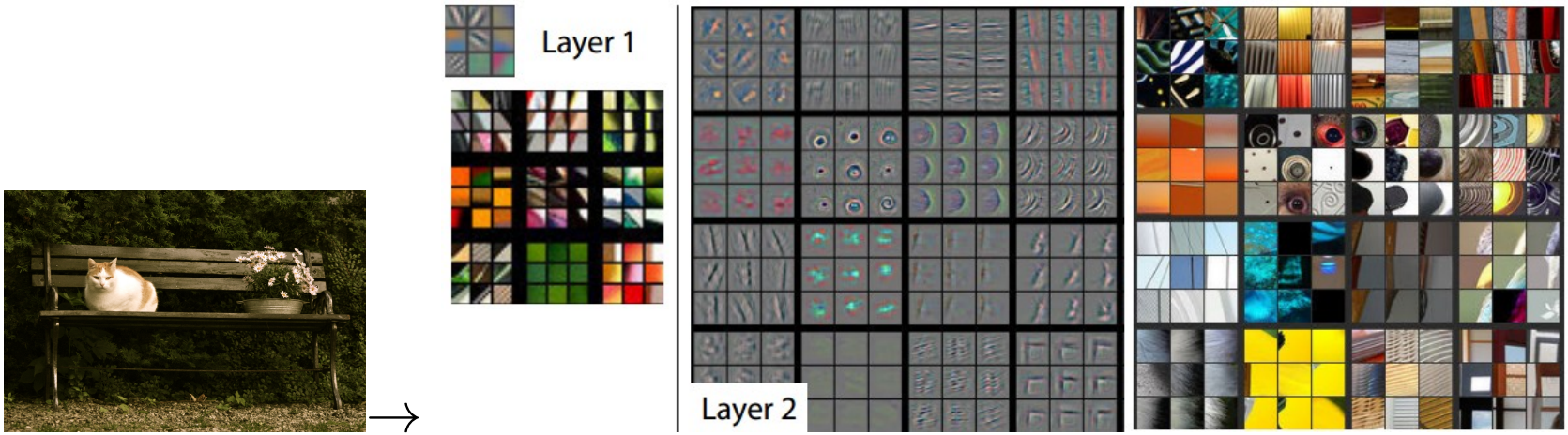
Generically useful Feature Representations

- Experiments show that feature hierarchies are transferable
- Natural learning machines don't need 10^6 labeled examples
- Is there an objective that allows us to learn high-level task-agnostic representations?
- Leverage virtually infinite amount of unlabeled data $\rightarrow$ **reduce training time, obtain better generalization, and enable on-line learning/adaptation**



Krizhevsky et al. NIPS 2012

Representation Learning



↓

[Third Layer Features]

↓

...

↓

[objects, orientations, positions, lighting, etc.]

Zeiler and Fergus. ECCV 2014

Desirable Properties of Features

Unsupervised Learning Problem: *Implicitly* learn features that facilitate solving *many* problems simultaneously

Approach: guess useful properties of features, then experimentally validate their usefulness on multiple problems

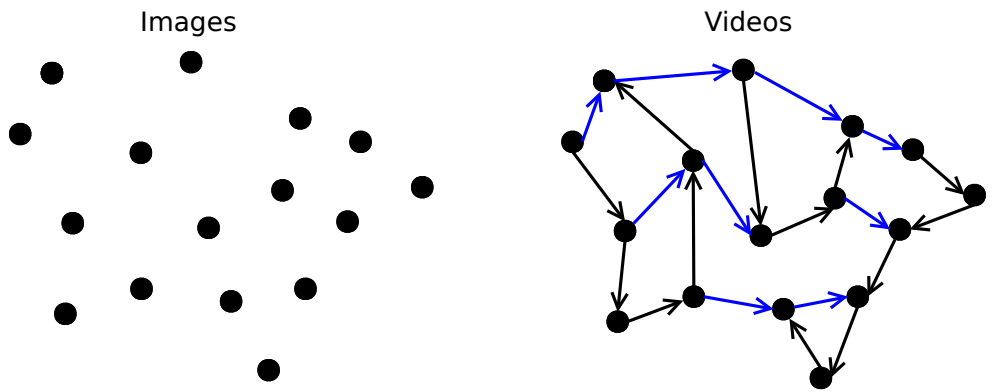
- Informativeness $\rightarrow$ reconstruction, max-likelihood
- Independence $\rightarrow$ Independent Component Analysis, Sparsity
- Invariance $\rightarrow$ metric learning, Slow Feature Analysis, classifiers
- Equivariance $\rightarrow$ linearization of transformations

Unsupervised Learning from Video

1-Unsupervised Learning of Spatiotemporally Coherent Metrics
Goroshin et al. http://arxiv.org/abs/1412.6056
ICCV2015-Accepted

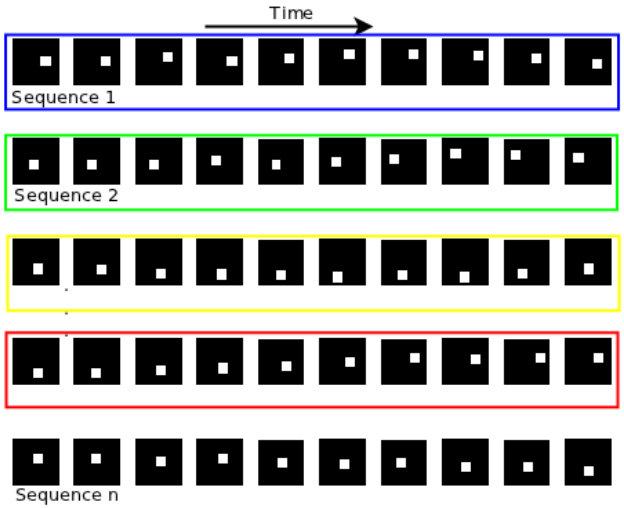
2-Learning to Linearize Under Uncertainty
Goroshin, Mathieu, LeCun. http://arxiv.org/abs/1506.03011
NIPS2015-Under Review

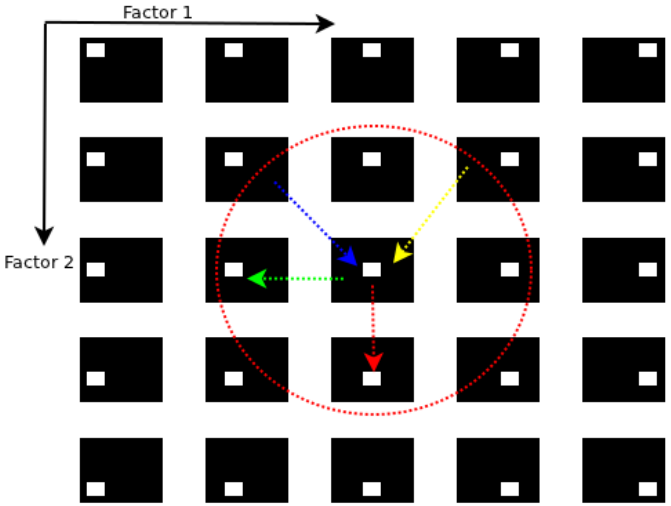
The Role of Time



- Natural videos are one-dimensional trajectories on the natural image manifold
- *Time reveals who your semantic nearest neighbors are*
- Temporal coherence can be exploited as weak supervision

Toy Example





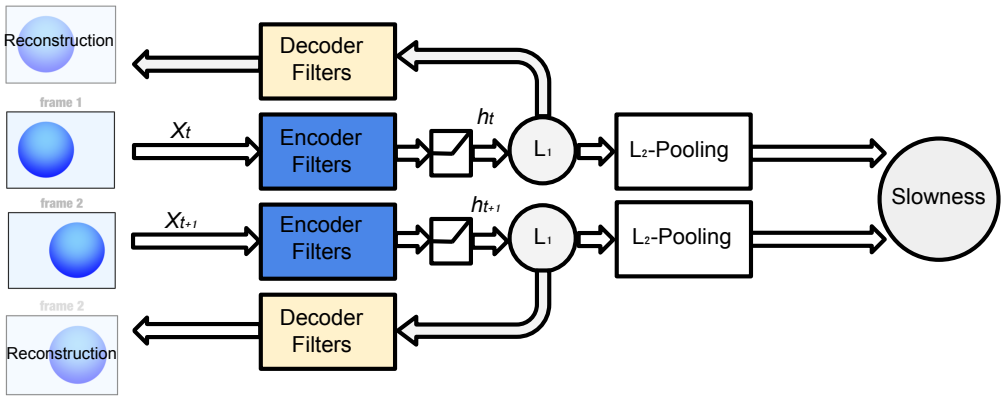
- Consider video sequences of a single pixel moving with *no overlap between successive frames*
- Extrinsic measures (e.g. L^2) of similarity are useless

Unsupervised Learning of Spatiotemporally Coherent Metrics

Goroshin et al. http://arxiv.org/abs/1412.6056
ICCV2015

The Model

Description: Sparse auto-encoder whose activations are L^2-pooled in local groups, on which slowness regularization is applied.



$$L(x_t, x_{t'}, W_e, W_d) = \sum_{\tau=\{t,t'\}} \left(\underbrace{\| W_d ReLU(W_e x_\tau) - x_\tau \|}_{\text{Reconstruction}} + \alpha \underbrace{ReLU(W_e x_\tau)}_{\text{Sparsity}} \right) + \underbrace{\beta \sum_{i=1}^{K} \left| \| ReLU(W_e x_t) \|^{P_i} - \| ReLU(W_e x_{t'}) \|^{P_i} \right|}_{\text{Slowness after local } L^2 \text{ pooling}}$$

Goroshin et al. http://arxiv.org/abs/1412.6056

Fourier Interpretation

$\rightarrow$ Fully connected features learned by training on natural videos patches mainly learn (local) *translation invariance*

$\rightarrow$ Train convolutional features to learn a richer class of invariants

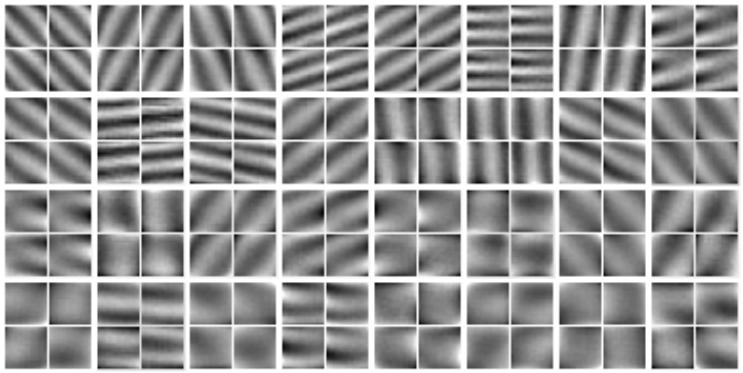


Figure: Without sparsity, $\alpha = 0$

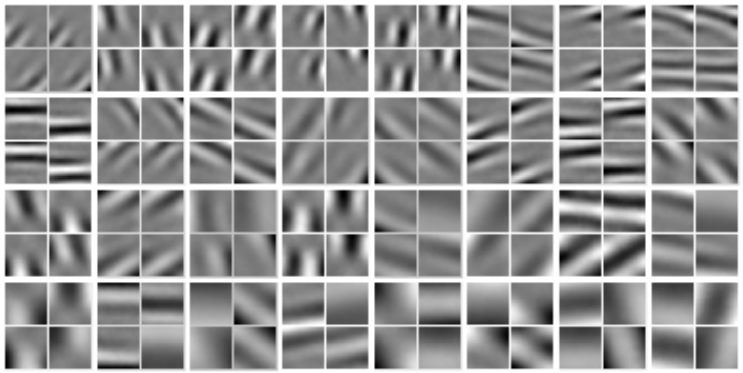


Figure: With sparsity, $\alpha > 0$

Dataset and Hyper-Parameter Selection

Problem: what is the correct trade-off between informativeness and invariance for natural data? In other words, how do we set the hyper-parameters α and β without supervision?

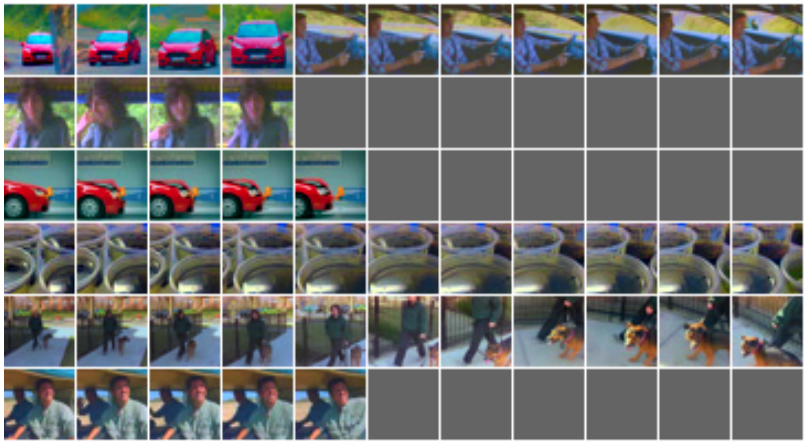
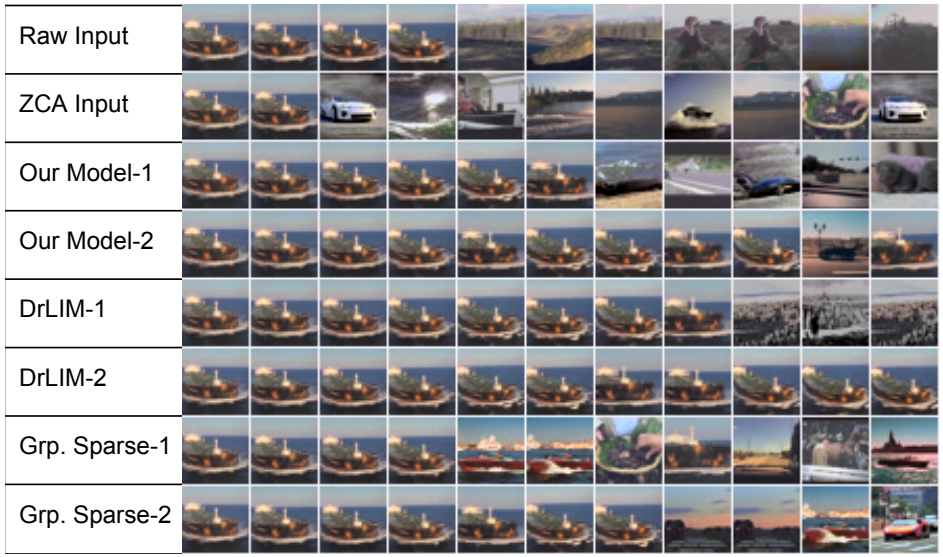
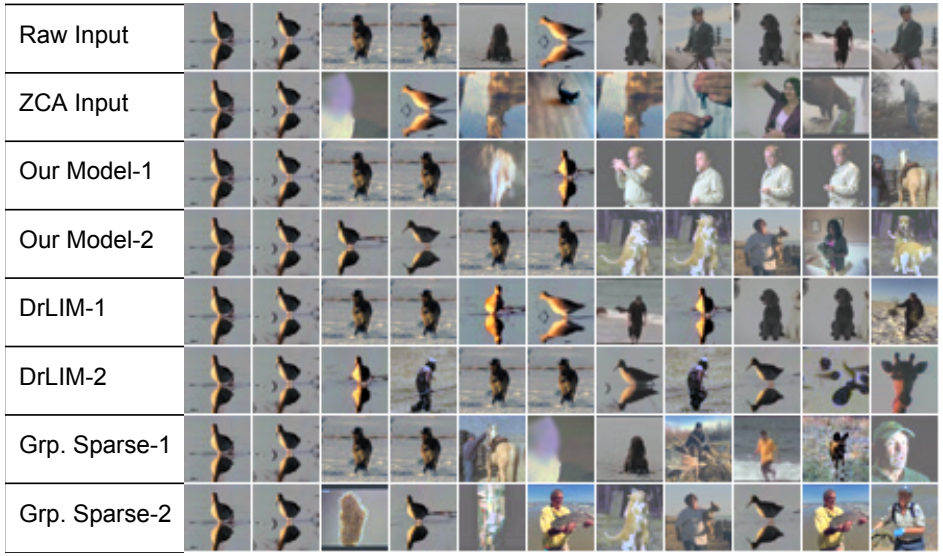


Figure: Each row shows a segmented scene from our YouTube dataset

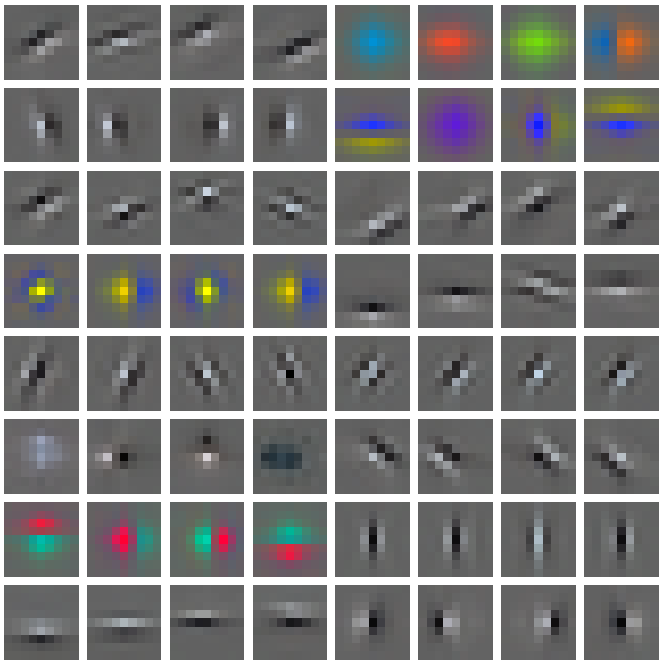
Temporally Coherent Feature Space: a feature space which induces the nearest neighbors to be the frames from the same scene

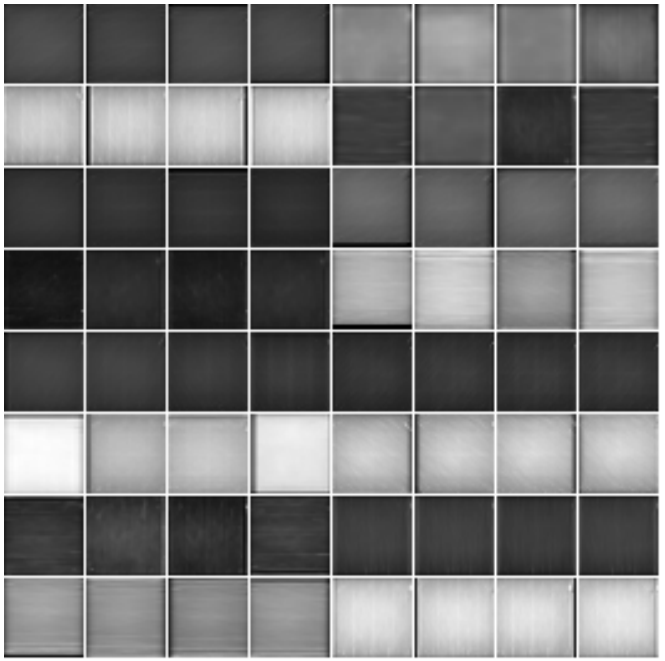
Visualizing Temporal Coherence - YouTube





Convolutional 1^{st} Layer Features

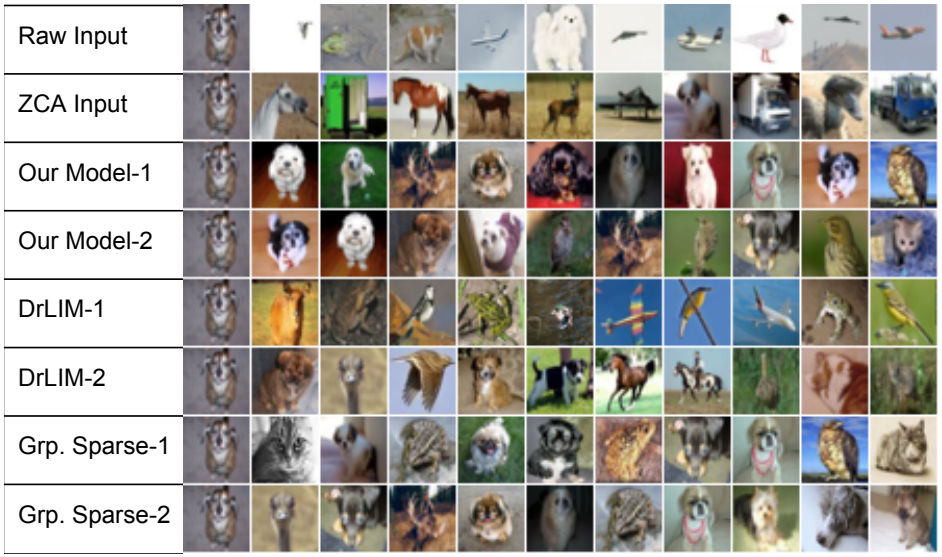


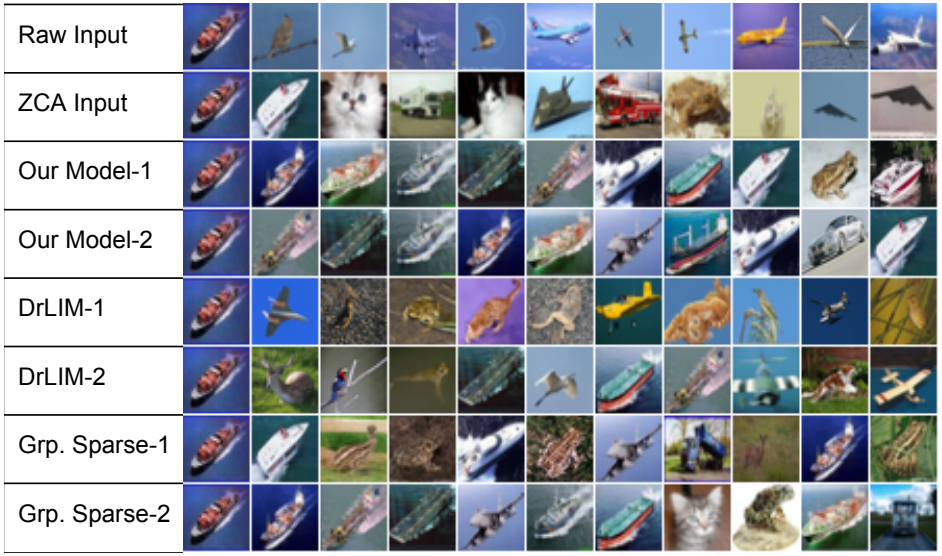


Left: Activations are pooled *across feature maps* in non-overlapping groups of four. **Features are grouped together to maximize temporal invariance**

Right: Activation frequency for corresponding feature over entire dataset

Visualizing Semantic Coherence - CIFAR10





Precision-Recall

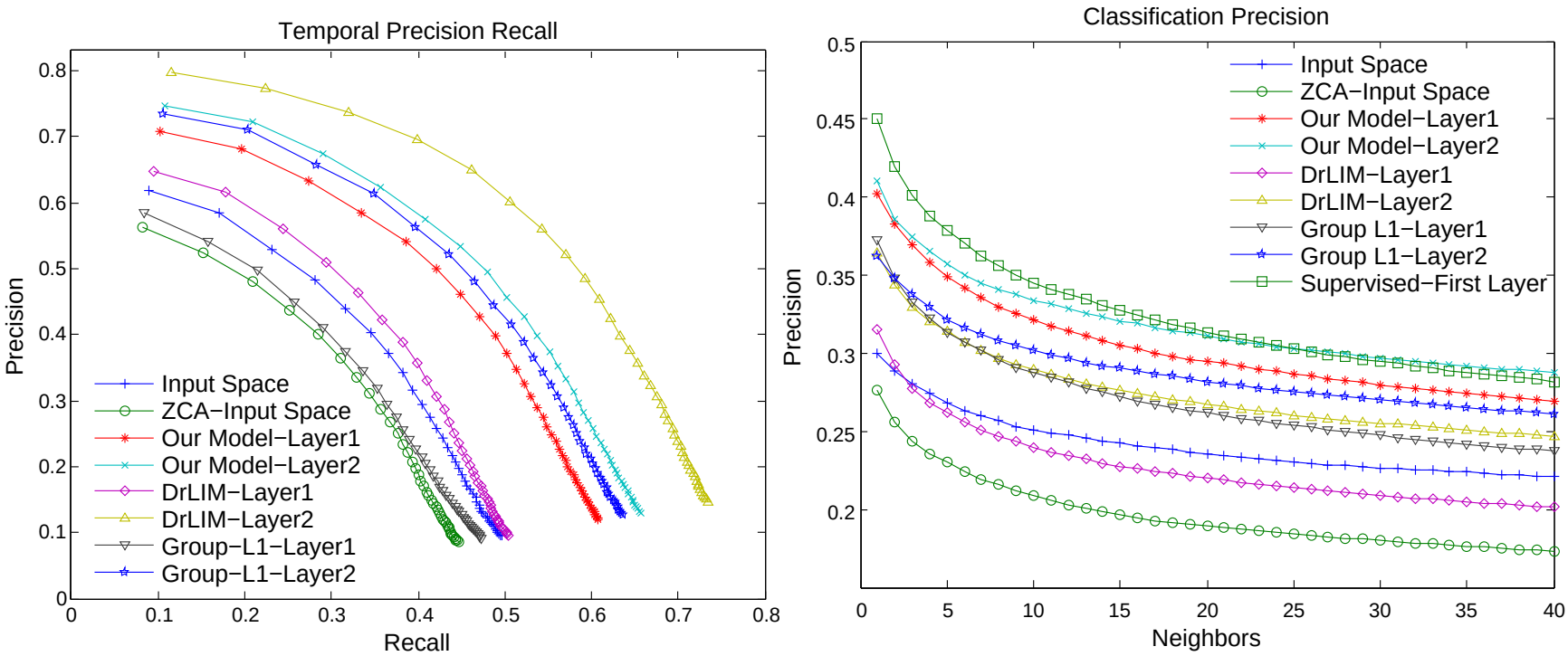
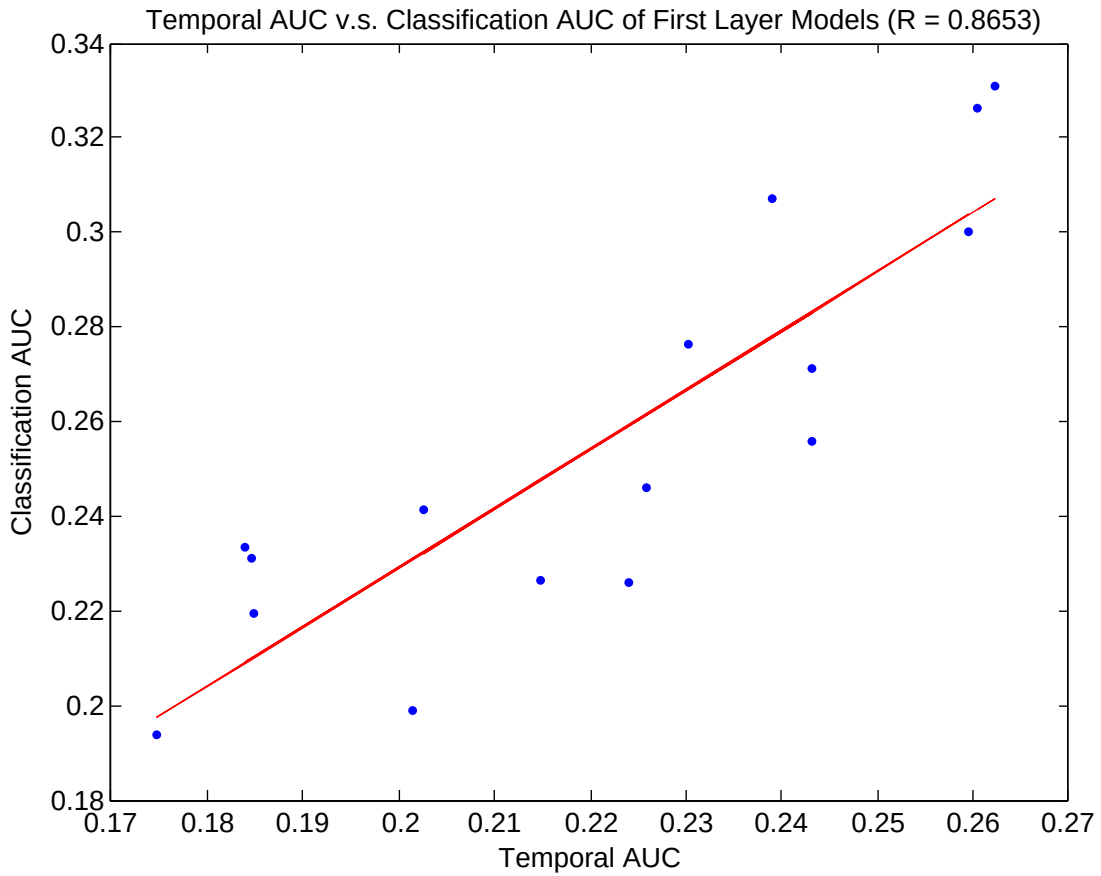


Figure: Temporal PR: Percent of nearest neighbors from the same ***scene***
Class PR: Percent of nearest neighbors from the same ***class***

Correlation between Temporal and Class AUCs



This demonstrates that a *semantically* coherent metric can be learned implicitly by maximizing *temporal coherence*.

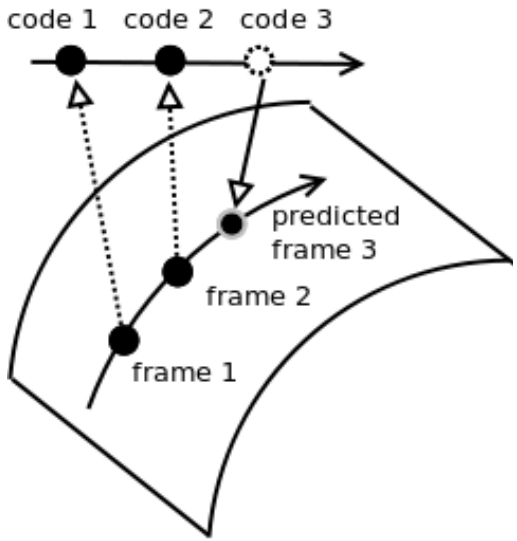
Shortcomings

- Layer-wise greedy training
- Requires heuristic measure for hyper-parameter selection
- Too many hyper-parameters require costly optimization
- Partial reconstruction means that the final features may have lost information

Learning to Linearize under Uncertainty

Goroshin, Mathieu, LeCun. http://arxiv.org/abs/1506.03011

Features that Linearize Temporal Trajectories



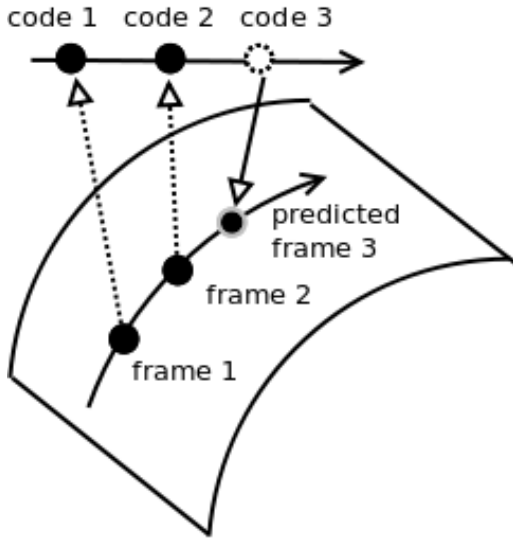
Let $X = \{..., x^{t-1}, x^t, x^{t+1}, ...\}$ be a sequence of frames and denote the code for frame x^t by $z^t = F_W(x^t)$, where $F_W()$ and $G_W()$ denote the encoder and decoder, respectively.

$$L = \frac{1}{2} \| G_W([2 \quad -1] \begin{bmatrix} z^t & z^{t-1} \end{bmatrix}^T) - x^{t+1} \|_2^2$$

$\rightarrow$ **Predict future frame via linear extrapolation in code space**

Goroshin, Mathieu, LeCun. http://arxiv.org/abs/1506.03011

Features that Linearize Temporal Trajectories

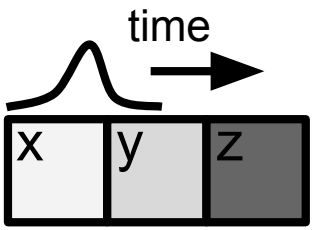


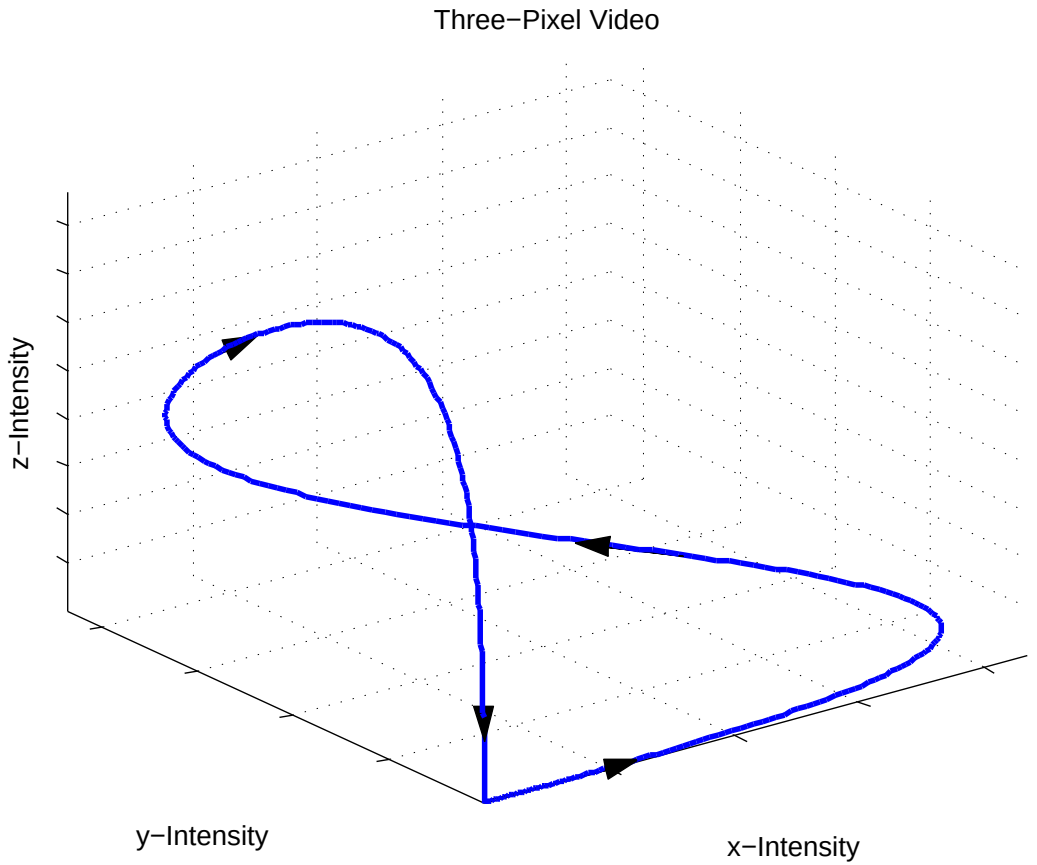
Let $X = \{..., x^{t-1}, x^t, x^{t+1}, ...\}$ be a sequence of frames and denote the code for frame x^t by $z^t = F_W(x^t)$, where $F_W()$ and $G_W()$ denote the encoder and decoder, respectively.

$$L = \frac{1}{2}\| G_W([2 \ -1] \begin{bmatrix} z^t & z^{t-1} \end{bmatrix}^T) - x^{t+1}\|_2^2 - \lambda \frac{(z^t - z^{t-1})^T (z^{t+1} - z^t)}{\|z^t - z^{t-1}\| \|z^{t+1} - z^t\|}$$

Goroshin, Mathieu, LeCun. http://arxiv.org/abs/1506.03011

Toy Example: 3-Pixel Movie





What is the linearized feature of this data set?

Implicitly learn to track salient image features

Phase-Pooling Operator

- Define a soft version of the *max* ("what") and *argmax* ("where") operators applied to each pool group

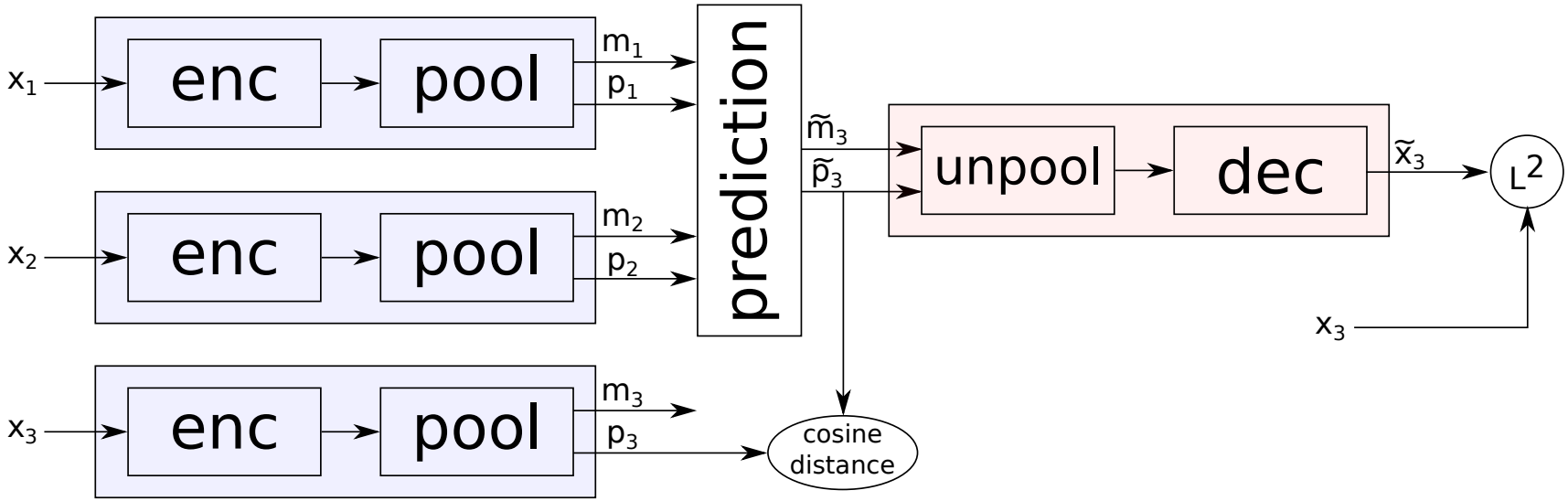
$$m_k = \sum_{N_k} z(f,x,y) \frac{e^{\beta z(f,x,y)}}{\sum_{N_k} e^{\beta z(f\prime,x\prime,y\prime)}} \approx \max_{N_k} z(f,x,y)$$

- Assuming that the activation pattern within each neighborhood is approximately unimodal
- The vector $\mathbf{p}_k$ approximates the local coordinates in the feature topology at which the max activation value occurred

$$\mathbf{p}_k = \sum_{N_k} \begin{bmatrix} f \\ x \\ y \end{bmatrix} \frac{e^{\beta z(f,x,y)}}{\sum_{N_k} e^{\beta z(f\prime,x\prime,y\prime)}} \approx \arg\max_{N_k} z(f,x,y)$$

- We can now back-propagate through the location of the activation, i.e. the "switch locations"

Complete Architecture

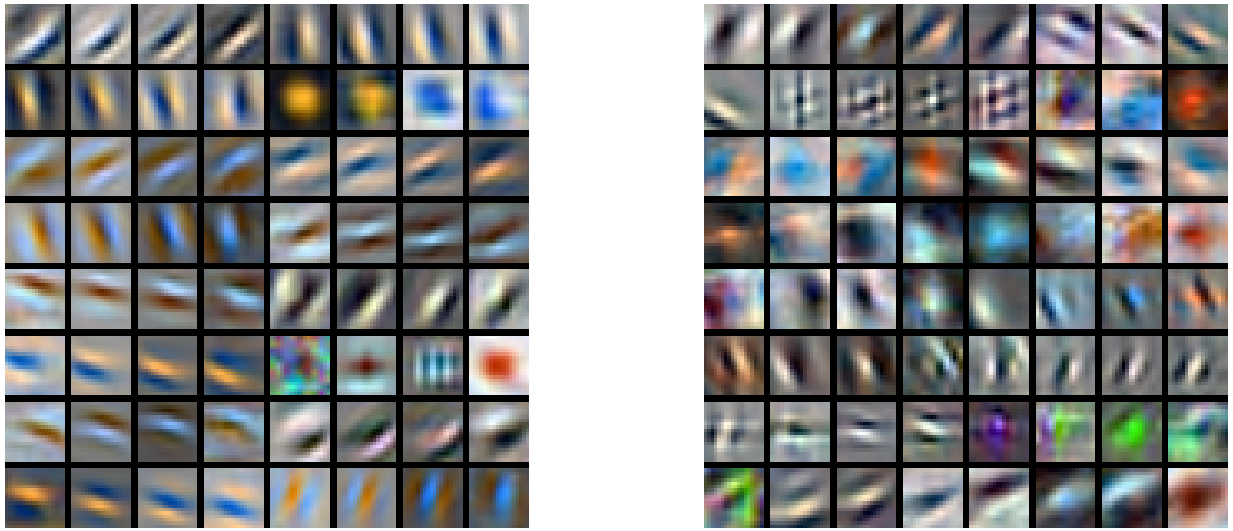


The predicted magnitude and phase are defined as follows:

$$m^{t+1} = \frac{m^t + m^{t-1}}{2}$$
$$\mathbf{p}^{t+1} = 2\mathbf{p}^t - \mathbf{p}^{t-1}$$

Define an "*un-pooling*" operation of the decoder that produces reconstructed activation maps by placing the magnitudes m at appropriate locations given by the phases $\mathbf{p}$.

Visualization of 1^{st}-Layer Features

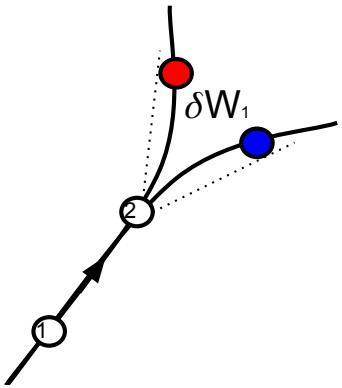


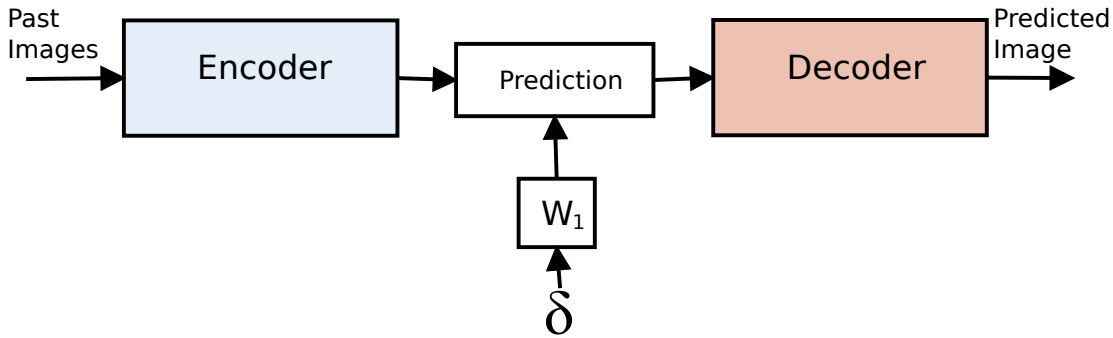
Left: Phase-pooling in non-overlapping groups (synthetic video)
Right: Phase-pooling in overlapping groups (natural video)

Features are grouped together to minimize prediction error by implicitly tracking salient image features

Addressing Uncertainty

- Inherent uncertainty in temporal data renders L^2 error meaningless especially for long-term predictions
- If multiple outcomes are present in the training set then minimizing the L^2 distance to these multiple outcomes induces the network to predict the *average outcome*
- Introduce latent slack variable δ that act to modulate the predicted code in order to switch between outcomes





Addressing Uncertainty

The δ-corrected code is defined as:

$$\hat{z}_{\delta}^{t+1} = z^t + (W_1\delta) \odot (z^t - z^{t-1})$$

The δ-corrected loss is:

$$L = \min_{\delta} \| G_W(\hat{z}_{\delta}^{t+1}) - x^{t+1} \|_2^2 - \lambda \frac{(z^t - z^{t-1})^T (z^{t+1} - z^t)}{\|z^t - z^{t-1}\| \|z^{t+1} - z^t\|}$$

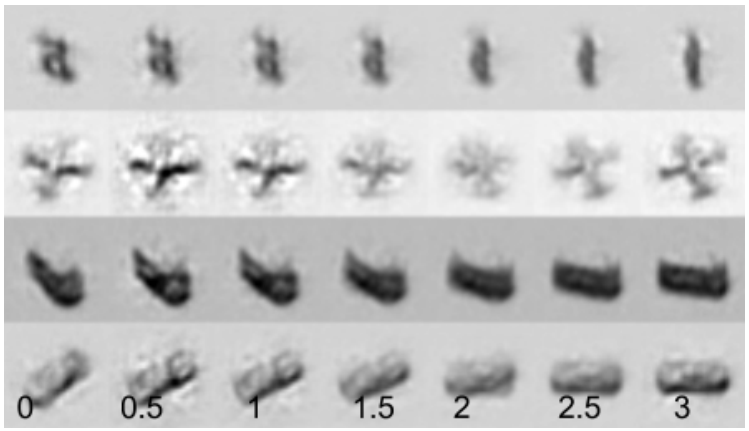
Before backproping through the encoder, the predicted is optimized using δ

NORB Experiment

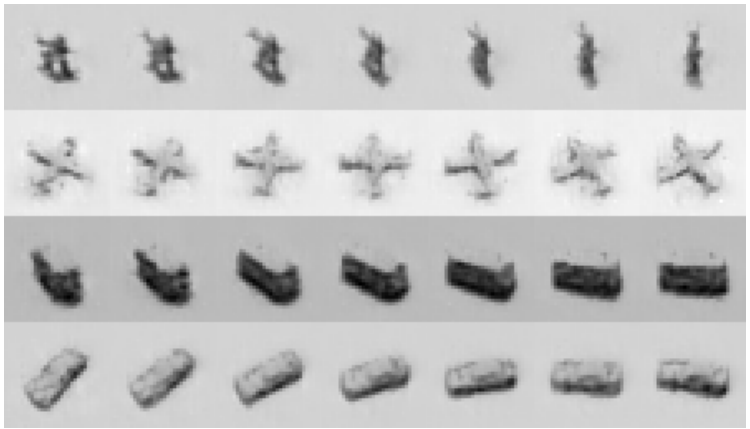
- Simulate video via random trajectories in the latent space
- Network is trained to predict the third frame from the two previous frames
- To evaluate the **linearity and informativeness** of the representation: 1-interpolate/extrapolate new codes, 2-generate the corresponding sample using the decoder

NORB Interpolated Test Frames

Baseline Siamese Network

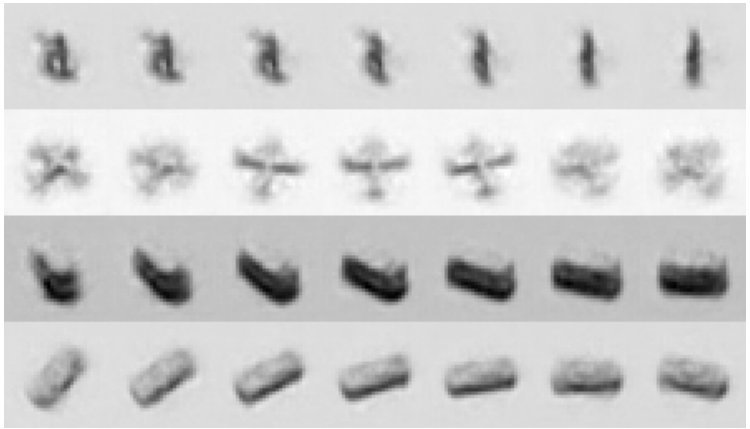


Proposed Architecture

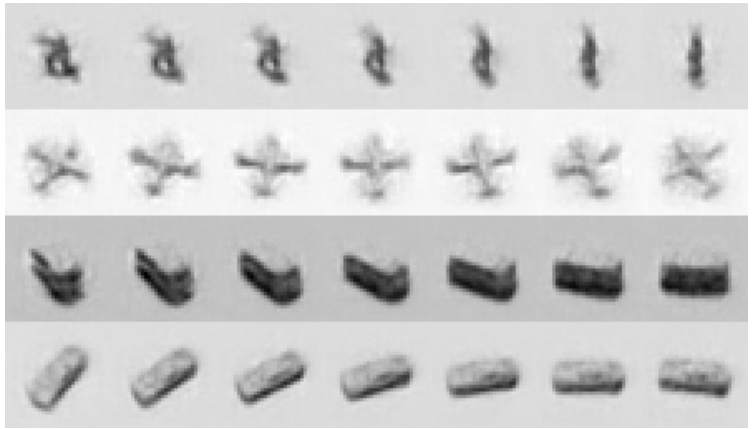


Simulated Uncertainty

Without δ Correction



With δ Correction



Simulate uncertainty by training on sequences with randomly skipped frames, i.e. train on $\{x^{t-1}, x^t, x^{t+1}\}$ or $\{x^{t-1}, x^t, x^{t+2}\}$ with equal probability.

Summary

Unsupervised Feature Learning

- Supervised $\rightarrow$ explicit solutions
- Unsupervised $\rightarrow$ implicit solutions
- Explicit solutions are usually going to outperform implicit solutions, thus we need new evaluation criteria (e.g. transfer learning tasks)
- Sparsity, independence, slowness, and linearity are just a few examples...There are probably many more generic priors out there!

Summary

Unsupervised Learning of Spatiotemporally Coherent Metrics

- Time is a source of weak supervision
- Connection between slow features and metric learning
- Not *trully* unsupervised until a recipe is provided for automatically tuning hyper-parameters
- Semantic coherence learned via temporal coherence

Summary

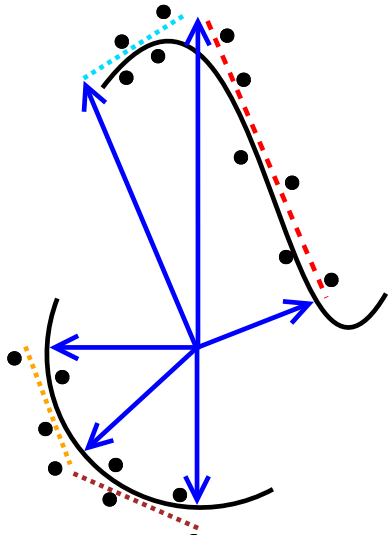
Learning to Linearize under Uncertainty

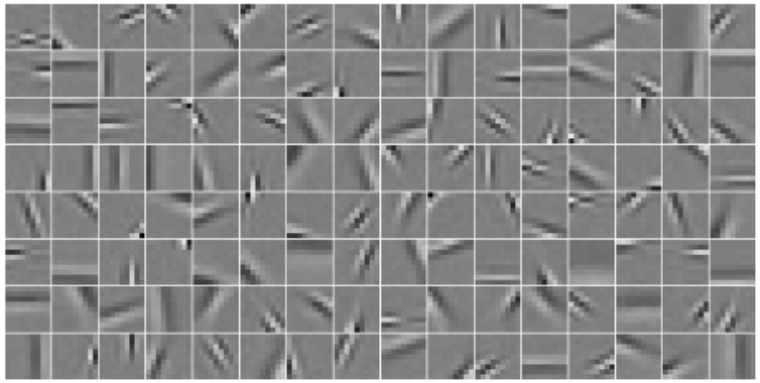
- Introduced linearizing features by formulating a prediction problem
- Phase-Pooling
- Addressed uncertainty without resorting to probabilistic methods (require sampling)

Thank You!

Supplementary Material

Sparse Features - Informativeness, Independence

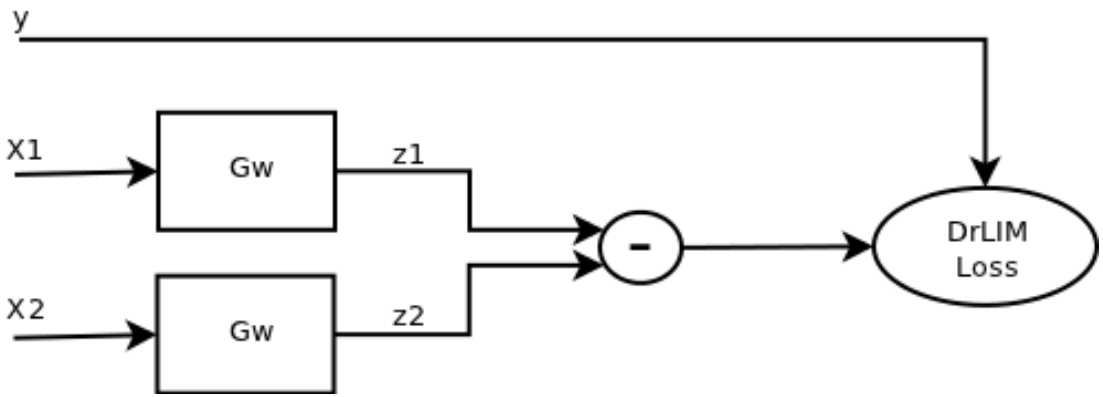




$$L(x, W_e, W_d) = \sum_i \left(\underbrace{\|W_d ReLU(W_e x_i) - x_i\|^2}_{Reconstruction} + \alpha \underbrace{ReLU(W_e x_i)}_{Sparsity} \right)$$

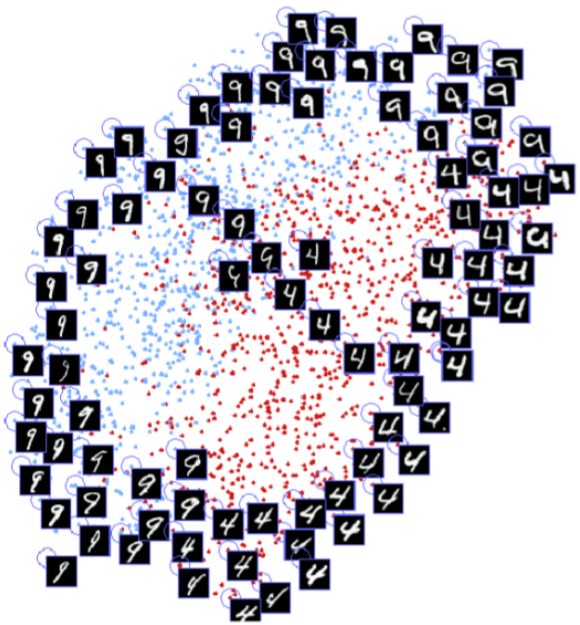
- Learns an over-complete basis which reconstructs the data by linearly combining a small subset of the available elements
- Represents a local-linear model of the data manifold
- Features are roughly independent
- Highly unstable representation

Similarity Metric Learning - Invariance, Equivariance



$$L(x_i, x_j, W) = \begin{cases} \|G_W(x_i) - G_W(x_j)\|_p, & \text{if } i \text{ similar to } j \\ \max(0, m - \|G_W(x_i) - G_W(x_j)\|_p) & \text{if } i \text{ dissimilar to } j \end{cases}$$

- Global structure can be learned via local (stochastic) comparisons
- Repulsive term does not guarantee informative features (low-dimensional visualization)
- $G_w()$ is the learned representation

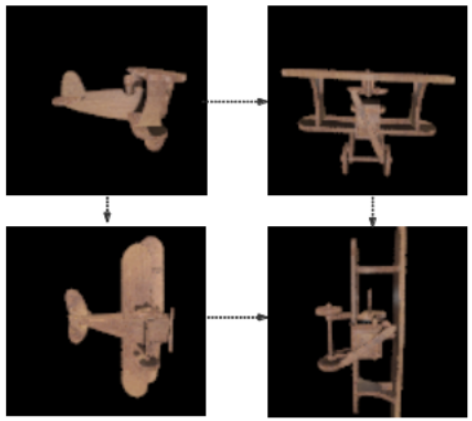


Hadsell et al. CVPR 2006

Temporal DrLIM Example

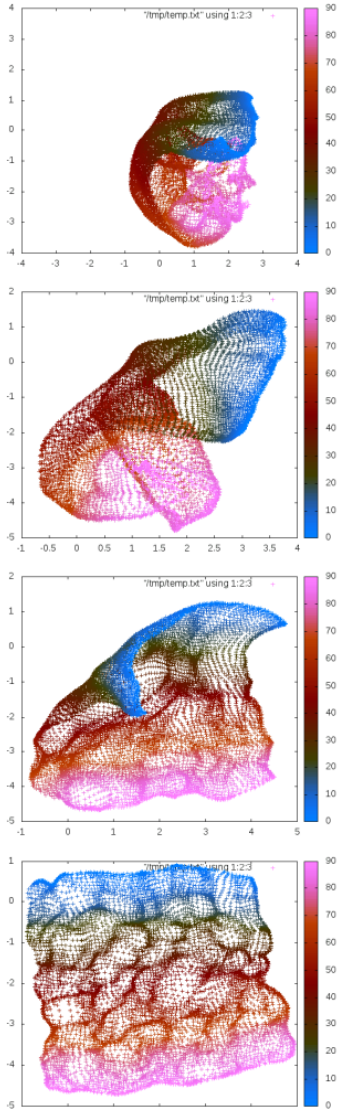
DrLIM learns intrinsic factors of variation when trained on video

- 2-dimensional manifold living in a $\approx 10,000$-dimensional space (96x96 images)
- Similarity relationships can be naturally assigned via adjacent frames in a video

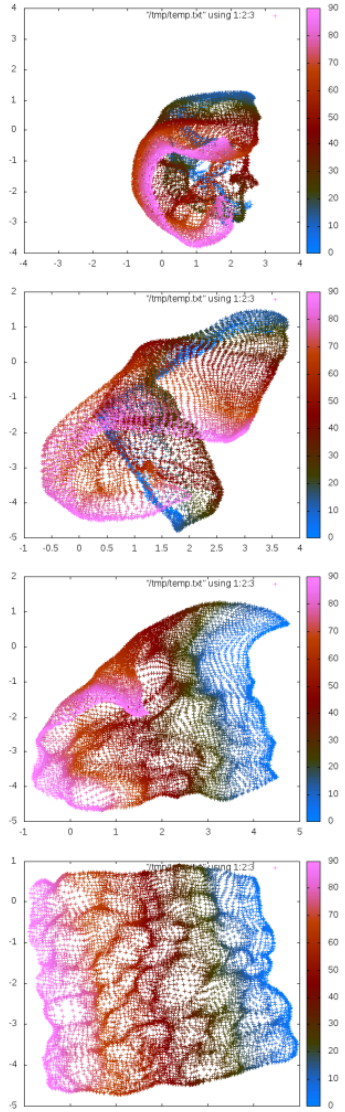


- Learning features that decrease the variability between temporally adjacent samples are known as "slow features"
- Without additional constraints slow features collapse to constants

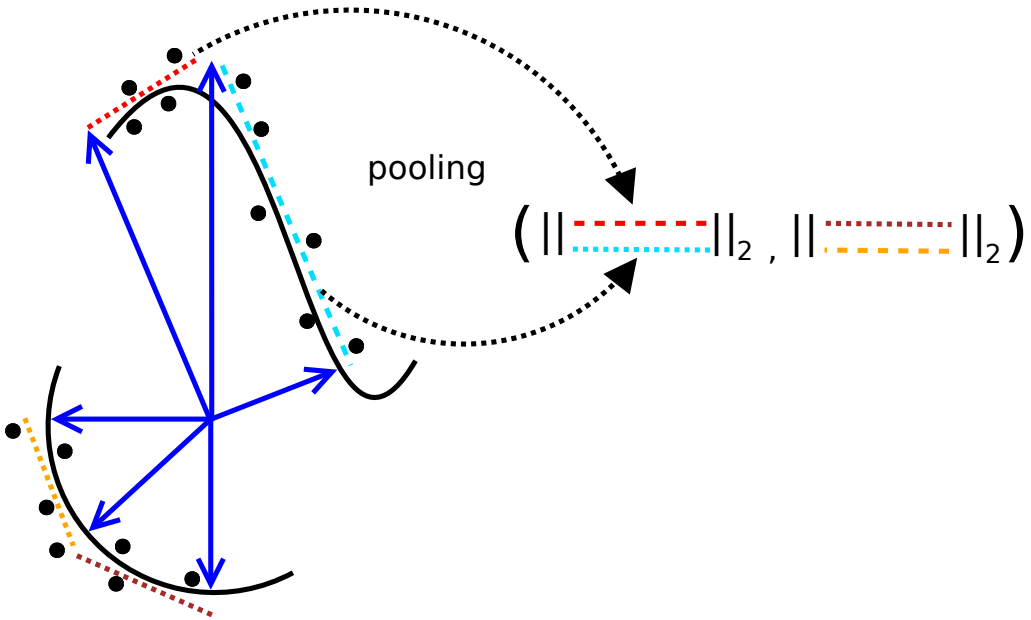
Color-map Yaw



Color-map Roll



Intuitive Interpretation of Slow Features



- Reconstruction $\rightarrow$ promotes informative features
- Sparsity $\rightarrow$ promotes independent features
- Slowness $\rightarrow$ promotes invariant features

Prediction Architectures

	Encoder	Prediction	Decoder
Shallow Architecture 1	Conv+ReLU $64 \times 9 \times 9$ Phase Pool 4	Average Mag. Linear Extrap. Phase	Conv $64 \times 9 \times 9$
Shallow Architecture 2	Conv+ReLU $64 \times 9 \times 9$ Phase Pool 4 stride 2	Average Mag. Linear Extrap. Phase	Conv $64 \times 9 \times 9$
Deep Architecture 1	Conv+ReLU $16 \times 9 \times 9$ Conv+ReLU $32 \times 9 \times 9$ FC+ReLU 8192×4096	None	FC+ReLU $\mathbf{8192} \times 8192$ Reshape $32 \times 16 \times 16$ SpatialPadding 8×8 Conv+ReLU $16 \times 9 \times 9$ SpatialPadding 8×8 Conv $1 \times 9 \times 9$
Deep Architecture 2	Conv+ReLU $16 \times 9 \times 9$ Conv+ReLU $32 \times 9 \times 9$ FC+ReLU 8192×4096	Linear Extrapolation	FC+ReLU 4096×8192 Reshape $32 \times 16 \times 16$ SpatialPadding 8×8 Conv+ReLU $16 \times 9 \times 9$ SpatialPadding 8×8 Conv $1 \times 9 \times 9$
Deep Architecture 3	Conv+ReLU $16 \times 9 \times 9$ Conv+ReLU $32 \times 9 \times 9$ FC+ReLU 8192×4096 Reshape $64 \times 8 \times 8$ Phase Pool 8×8	Average Mag. Linear Extrap. Phase	Unpool 8×8 FC+ReLU 4096×8192 Reshape $32 \times 16 \times 16$ SpatialPadding 8×8 Conv+ReLU $16 \times 9 \times 9$ SpatialPadding 8×8 Conv $1 \times 9 \times 9$

Table: Summary of architectures