Center for Brains, Minds & Machines

CBMM Memo No. 040 December 15, 2015

# UNSUPERVISED LEARNING OF VISUAL STRUCTURE USING PREDICTIVE GENERATIVE NETWORKS

by

William Lotter, Gabriel Kreiman and David Cox

Abstract: The ability to predict future states of the environment is a central pillar of intelligence. At its core, effective prediction requires an internal model of the world and an understanding of the rules by which the world changes. Here, we explore the internal models developed by deep neural networks trained using a loss based on predicting future frames in synthetic video sequences, using an Encoder-Recurrent-Decoder framework (Fragkiadaki et al., 2015). We first show that this architecture can achieve excellent performance in visual sequence prediction tasks, including state-of-the-art performance in a standard "bouncing balls" dataset (Sutskever et al., 2009). We then train on clips of out-of-the-plane rotations of computer-generated faces, using both mean-squared error and a generative adversarial loss (Goodfellow et al., 2014), extending the latter to a recurrent, conditional setting. Despite being trained end-to-end to predict only pixel-level information, our Predictive Generative Networks learn a representation of the latent variables of the underlying generative process. Importantly, we find that this representation is naturally tolerant to object transformations, and generalizes well to new tasks, such as classification of static images. Similar models trained solely with a reconstruction loss fail to generalize as effectively. We argue that prediction can serve as a powerful unsupervised loss for learning rich internal representations of high-level object features.

This work was supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216.

# UNSUPERVISED LEARNING OF VISUAL STRUCTURE USING PREDICTIVE GENERATIVE NETWORKS

**William Lotter, Gabriel Kreiman & David Cox**
Harvard University
Cambridge, MA 02138, USA
{lotter,davidcox}@fas.harvard.edu
gabriel.kreiman@tch.harvard.edu



## ABSTRACT

The ability to predict future states of the environment is a central pillar of intelligence. At its core, effective prediction requires an internal model of the world and an understanding of the rules by which the world changes. Here, we explore the internal models developed by deep neural networks trained using a loss based on predicting future frames in synthetic video sequences, using an Encoder-Recurrent-Decoder framework (Fragkiadaki et al., 2015). We first show that this architecture can achieve excellent performance in visual sequence prediction tasks, including state-of-the-art performance in a standard "bouncing balls" dataset (Sutskever et al., 2009). We then train on clips of out-of-the-plane rotations of computer-generated faces, using both mean-squared error and a generative adversarial loss (Goodfellow et al., 2014), extending the latter to a recurrent, conditional setting. Despite being trained end-to-end to predict only pixel-level information, our Predictive Generative Networks learn a representation of the latent variables of the underlying generative process. Importantly, we find that this representation is naturally tolerant to object transformations, and generalizes well to new tasks, such as classification of static images. Similar models trained solely with a reconstruction loss fail to generalize as effectively. We argue that prediction can serve as a powerful unsupervised loss for learning rich internal representations of high-level object features.

## 1 INTRODUCTION

There is a rich literature in neuroscience concerning the notion of "predictive coding" — the idea that neuronal systems predict future states of the world, and primarily encode deviations from those predictions (Rao & Ballard, 1999; Summerfield et al., 2006; Den Ouden et al., 2012; Rao & Sejnowski, 2000; Chalasani & Principe, 2013; Zhao et al., 2014). There are many different scales and domains of prediction, but fundamentally, successful prediction requires an understanding of the latent causes of the world. Such an ability requires a mapping from a high-dimensional, noisy sensory space, to an internal representation where higher level concepts can be inferred. Today's state-of-the-art deep learning models typically rely on millions of labeled training examples to learn such a representation (Krizhevsky et al., 2012), in contrast to biological systems, where learning is largely unsupervised. Here, we explore the idea that prediction is not only a useful end-goal, but may also serve as a powerful unsupervised learning signal.

The problem of prediction is one of estimating a conditional distribution: given recent data, estimate the probability of future states. There has been much recent success in this domain within natural language processing (NLP) (Graves, 2013; Sutskever et al., 2014) and relatively low-dimensional, real-valued problems such as motion capture (Fragkiadaki et al., 2015; Gan et al., 2015). Generating realistic samples for high dimensional images, particularly predicting the next frames in videos, has proven to be much more difficult. Recently, Ranzato et al. (2014) used a close analogy to NLP models by discretizing image patches into a dictionary set, for which prediction is posed as predicting the index into this set at future time points. This approach was chosen because of the innate difficulty of using traditional losses in video prediction. In pixel space, losses such as mean-squared error (MSE) are unstable to slight deformations and fail to capture intuitions of image similarity. These

issues are further illustrated in Srivastava et al. (2015), where their LSTM-based encoder-decoder model produces blurry predictions of natural image patches. Despite this, predictive pre-training improved performance on two action recognition datasets, pointing to the potential of prediction as unsupervised learning.

A promising alternative to MSE is an adversarial loss, as in the Generative Adversarial Network (GAN) (Goodfellow et al., 2014). This framework involves training a generator and discriminator in a minimax fashion. Successful extensions, including a conditional GAN (Gauthier, 2014; Mirza & Osindero, 2014) and a Laplacian pyramid of GANs (Denton et al., 2015), point to its promise as a useful model for generating images.

Here we build upon recent advances in generative models, as well as classical ideas of predictive coding and unsupervised temporal learning, to investigate deep neural networks trained with a predictive loss. We use an Encoder-Recurrent-Decoder (ERD) architecture (Fragkiadaki et al., 2015), which is trained end-to-end to combine feature representation learning with the learning of temporal dynamics. In addition to MSE, we implement an adversarial loss, which we extend to a recurrent setting with limited, but some, success.

We demonstrate the effectiveness of our model on a standard "bouncing balls" experiment (Sutskever et al., 2009) before applying the same architecture to a dataset of computer-generated faces undergoing rotations. This dataset is an appropriate intermediate step between toy examples and full-scale natural images, where we can more fully study the representational learning process. We find that, over the course of training, the model becomes better at representing the latent variables of the underlying generative model. We test the generality of this representation in a face identification task requiring transformation tolerance. In the classification of *static* images, the model, trained with a predictive loss on dynamic stimuli, strongly outperforms comparable models trained with a reconstruction loss on static images. Thus, we illustrate the promise of prediction as unsupervised model learning from video.

## 2 Related Work

Along with prediction, our work has strong roots in learning from temporal continuity. Early efforts demonstrated how invariances to particular transformations can be learned through temporal exposure (Földiák, 1991). Related algorithms, such as Slow Feature Analysis (SFA) (Wiskott & Sejnowski, 2002), take advantage of the persistence of latent causes in the world to learn representations that are robust to noisy, quickly-varying sensory input. More recent work has explicitly implemented temporal coherence in the cost function of deep learning architectures, enforcing the networks to develop a representation where feature vectors of consecutive video frames are closer together than those between non-consecutive frames (Mohabi et al., 2009; Goroshin et al., 2015; Wang & Gupta, 2015).

Also related to our approach, especially in the context of rotating objects, is the field of relational feature learning (Memisevic & Hinton, 2007; Taylor & Hinton, 2009). This posits modeling time-series data as learning representations of the *transformations* that take one frame to the next. Recently, Michalski et al. (2014) proposed a predictive training scheme where a transformation is first inferred between two frames and then is applied again to obtain a prediction of a third frame. They reported evidence of a benefit of using predictive training versus traditional reconstruction training.

Finally, using variations of autoencoders for unsupervised learning and pre-training, is certainly not new (Erhan et al., 2010; Bengio et al., 2006). In fact, Palm (2012) coined the term "Predictive Encoder" to refer to an autoencoder that is trained to predict future input instead of reconstructing current stimuli. In preliminary experiments, it was shown that such a model could learn Gabor-like filters in training scenarios where traditional autoencoders failed to learn useful representations.

## 3 Predictive Generative Networks

A schematic of our framework is shown in Figure 1. Our generative model first embeds a sequence of frames, successively, into a lower-dimensional feature space using an encoder. For both of our tested datasets, we used an encoder in the form of a convolutional neural network (CNN) consisting of two layers of alternating convolution, rectification, and max-pooling. The encoder feeds into a

recurrent neural network (RNN). We used an RNN consisting of Long Short-Term Memory (LSTM) units (Hochreiter & Schmidhuber, 1997). Briefly, LSTM units are a particular type of hidden unit that improve upon the vanishing gradient problem that is common when training RNNs (Bengio et al., 1994). An LSTM unit contains a cell, $c_t$, which can be thought of as a memory state. Access to the cell is controlled through an input gate, $i_t$, and a forget gate, $f_t$. The final output of the LSTM unit, $h_t$, is a function of the cell state, $c_t$, and an output gate, $g_t$. We used a version of the LSTM with the following update equations:

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\
c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\
h_t &= o_t \tanh(c_t)
\end{aligned}
$$

where $x_t$ is the input to the LSTM network, $W_{\bullet\bullet}$ are the weight matrices, and $\sigma$ is the elementwise logistic sigmoid function. We used 1568 LSTM units for the bouncing balls dataset and 1024 for the rotating faces. All models were implemented using the software package Keras (Chollet, 2015).

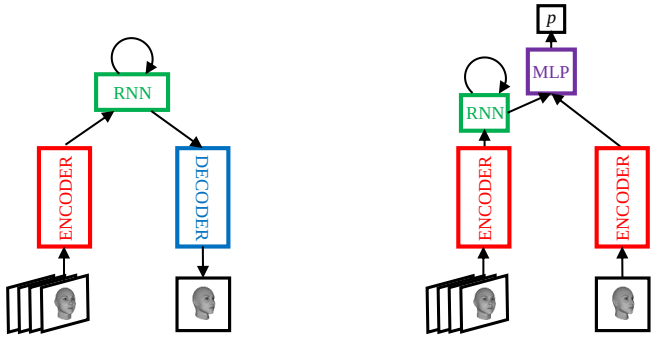


Figure 1: Predictive Generative Networks

Upon successively processing the output of the encoder, the LSTM hidden state is outputted to a decoder network, which then outputs a predicted image. The parameters of the network were chosen such that the predicted image is the same size of the input. For the rotating faces dataset, the first layer of the decoder is fully-connected (FC), followed by two layers of nearest-neighbor upsampling, convolution, and rectification. The last layer also contains a saturating non-linearity set at the maximum pixel value. The model trained on bouncing balls lacked the FC layer.

For adversarial loss, the predicted frame from the generator is passed to a discriminator network. The discriminator is also conditioned on the original input sequence. The input frames, as well as the proposed next frame, are first processed by an encoder. We enforced weight sharing between the encoders of the generator and discriminator, with the belief that features that would help predict the next frame would also be helpful in determining if a proposed frame was "real" or not, and vice versa. The feature vectors of the input frames are next processed by the discriminator LSTM network, which was not shared with the generator. Upon processing the last input vector, the discriminator LSTM hidden state is concatated with the feature vector of the proposed image and passed to a multi-layer perceptron (MLP) read-out. The MLP consists of three FC layers, of which the first two have a rectifying non-linearity and the last contains a softmax. When training the discriminator, we used batches consisting of predictions from the generator as well as the corresponding ground truth sequences.

We used the same form of the adversarial loss function as in the original paper (Goodfellow et al., 2014). The discriminator outputs a probability that a proposed frame came from the ground truth data. It is trained to maximize this probability when the frame came from the true distribution and minimize it when it is produced by the generator. The generator is trained to maximally confuse the

Table 1: Average prediction error for the bouncing balls dataset. †(Gan et al., 2015) ◇(Mittelman et al., 2014)

| Model | Error |
|---|---|
| **PGN** | $\mathbf{0.65 \pm 0.11}$ |
| DTSBN † | $2.79 \pm 0.39$ |
| SRTRBM ◇ | $3.31 \pm 0.33$ |
| RTRBM ◇ | $3.88 \pm 0.33$ |

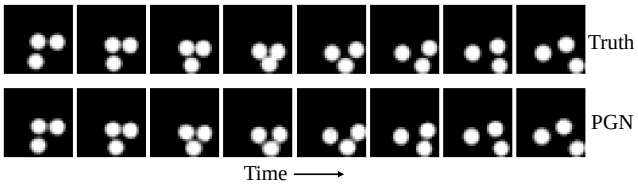


Figure 2: Example prediction sequence the for bouncing balls dataset. Predictions are repeatedly generated one step ahead using the prior ten frames as input.

generator by having it output $0.5$ for every sample. Let $x^i_{1:t}$ be an input sequence of $t$ frames and $x^i_{t+1}$ be the true next frame. Let the proposed frame from the generator be $G(x^i_{1:t})$ and $D(\bullet, x^i_{1:t})$ be the discriminator's output. Given a batch size of $n$ sequences, the loss of the discriminator, $J_D$, and of the generator, $J_G$, have the form:

$$J_D = -\frac{1}{2n}\sum_{i=1}^{n} \log D(x^i_{t+1}, x^i_{1:t}) + \log(1 - D(G(x^i_{1:t}), x^i_{1:t}))$$

$$J_G = \frac{1}{n}\sum_{i=1}^{n} \log(1 - D(G(x^i_{1:t}), x^i_{1:t}))$$

## 4 Prediction Performance

We have evaluated the Predictive Generative Networks on two datasets of synthetic video sequences. As a baseline to compare against other architectures, we first report performance on a standard bouncing balls paradigm (Sutskever et al., 2009). We then proceed to a dataset containing out-of-the-plane rotations of computer-generated faces, where we thoroughly analyze the representations learned under our framework.

### 4.1 Bouncing Balls

The bouncing balls dataset is a common test set for models that generate high dimensional sequences. It consists of simulations of three balls bouncing in a box. We followed standard procedure to create $4000$ training videos and $200$ testing videos (Sutskever et al., 2009) and used an additional $200$ videos for validation. Our networks were trained to take a variable number of frames as input, selected randomly each epoch from a range of $5$ to $15$, and output a prediction for the next timestep. We used MSE loss optimized with RMSprop (Tieleman & Hinton, 2012) with a learning rate of $0.001$. In Table 1, we report the average squared one-step-ahead prediction error per frame. Our model compares favorably to the recently introduced Deep Temporal Sigmoid Belief Network (Gan et al., 2015) and restricted Boltzmann machine (RBM) variants, the recurrent temporal RBM (RTRBM) and the structured RTRBM (SRTRBM) (Mittelman et al., 2014). An example prediction sequence is shown in Figure 2, where each prediction is made one step ahead by using the ten previous frames as input.

### 4.2 Rotating Faces

For the rotating faces dataset, each video consisted of a unique randomly generated face rotating about the vertical axis with a random speed and initial angle. The speed was sampled uniformly from $[0, \pi/6]$ rad/frame and initial angle from $[-\pi/2, \pi/2]$, with $0$ corresponding to facing forward. We used $4000$ clips for training and $200$ for validation and testing with a frame size of $150$x$150$ pixels. Five frames were used as input with the networks trained to predict the sixth. We explored using both MSE and adversarial loss, as well as a combination of both. For all losses, the generator was optimized using RMSprop (Tieleman & Hinton, 2012) with a learning rate of $0.001$. Vanilla stochastic gradient descent (SGD), with a learning rate of $0.01$ and momentum of $0.5$ was more effective for the discriminator.

We did not have much success training with adversarial loss alone, as the generator tended to easily find solutions that fooled the discriminator, but did not look anything like the correct samples. However, a combined MSE and adversarial loss (AL), with the MSE acting to restrict the search space of the generator, produced reasonable solutions. Example predictions for our MSE and approximately equally weighted AL/MSE models are shown in Figure 3. Predictions are for faces not seen during training. The models are able to extrapolate the rotations to estimate the angle and basic shape of the face very well. The MSE model is fairly faithful to the identities of the faces, but produces blurred, low-passed versions, as expected. The combined AL/MSE model tends to underfit the identity towards a more average face, which is especially illustrated in the first and fourth rows in Figure 3, but it does occasionally produce more realistic, low-level features. For instance, the AL/MSE model seems to have learned an ear template. While not always placing the ear in the precisely correct location, if offers a significant improvement upon MSE, which largely omits the ears. It is possible that a more thorough hyperparameter search could lead to solutions that are simultaneously true to the underlying face, while still having detailed facial features.

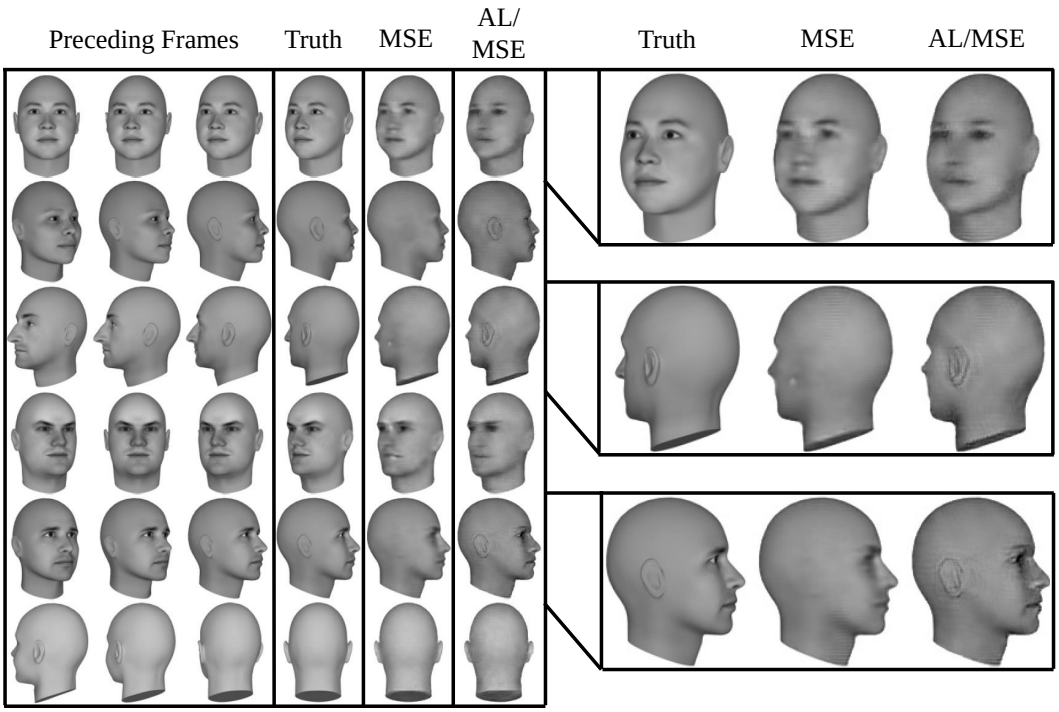


Figure 3: Example predictions for the rotating faces dataset. Predictions for models trained with MSE and a combination of MSE and adversarial loss (AL) are shown.

## 5 Exploring Latent Representation Learning

Beyond generating realistic predictions, we are interested in understanding the representations learned by the models, especially in relation to the underlying generative model. The faces are created according to a principle component analysis (PCA) in "face space", which was estimated from real-world faces. In addition to the principle components (PCs), the videos had latent variables of angular speed and initial angle. We began by conducting a decoding analysis of these variables as a function of training epoch. We focus on the MSE model because of its more faithful predictions and less stochastic training.

An L2-regularized regression was used to estimate the latent variables from the LSTM representation. We decoded from both the hidden unit and memory cell unit responses after the processing of five input frames. The regression was fit, validated, and tested using a different dataset than the one used to train the model. Decoding performance as a function of training epoch is shown in Figure 4. Epoch 0 corresponds to the random initial weights, from which the latent variables can already be

decoded fairly well. This to be expected given the empirical evidence and theoretical justifications for the success of random weights in neural networks (Kevin Jarrett & LeCun, 2009; Pinto et al., 2009; Saxe et al., 2010). Nonetheless, it is clear that the ability to estimate all latent variables increases over training. The model quickly peaks at its ability to linearly encode for speed and initial angle. The PC components are learned more slowly, with decoding accuracy for some PC's actually first decreasing while speed and angle are rapidly learned. The sequence in which the model learns is reminiscent of theoretical work supporting the notion that modes in the dataset are learned in a coarse-to-fine fashion (Saxe et al., 2013).

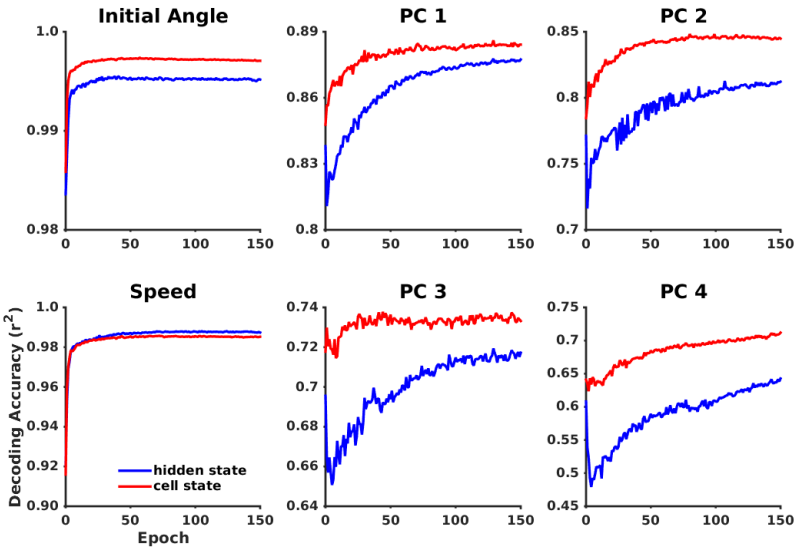


Figure 4: Decoding of latent variables from internal representation of PGN trained with MSE.

In Figures 5 and 6, we provide visualizations of the hidden unit feature space over training. Figure 5 contains a multidimensional-scaling (MDS) plot for the initial random weights and the weights after Epoch 150. Points are colored by PC1 value and rotation speed. Although a regression on this feature space at Epoch 0 produces an $r^2$ of ~0.83, it is clear that the structure of this space changes with training. To have a more clear understanding of these changes, we linearized the feature space in two dimensions with axes pointing in the direction of the regression coefficients for decoding PC1 and rotation speed. Points from a held-out set were projected on these axes and plotted in Figure 6 and we show the evolution of the projection space, with regression coefficients calculated separately for each epoch. Over training, the points become more spread out over this manifold. This is not due to an overall increase in feature vector length, as this does not increase over training. Thus, with training, the variance in the feature vectors become more aligned with the latent variables of the generative model.

The previous analyses suggests that the model learns a low dimensional, linear representation of the face space. As an illustration of this, in Figure 7, we start at a given representation in the feature space and move linearly in the direction of a principle component axis. The new feature vector is then passed to the pre-trained decoder to produce an image. We compare this with actually changing the PC values in the generative model. We are reasonably able to produce extrapolations that are realistic and correlate with the underlying model. The PC dimensions do not precisely have semantic meanings, but differences can especially be noticed in the cheeks and jaw lines. The linear extrapolations in feature space generally match changes in these features.

While the generation of frame-by-frame future predictions is potentially useful *per se*, we were especially interested in the extent to which prediction could be used as an unsupervised loss for learning representations that are suited to other tasks. We tested this hypothesis through a task completely orthogonal to the original loss function, namely classification of static images. As a comparison, we trained control models using reconstruction loss and either dynamic or static stimuli. The first control had the same architecture and training set of the PGN, but was trained to reconstruct

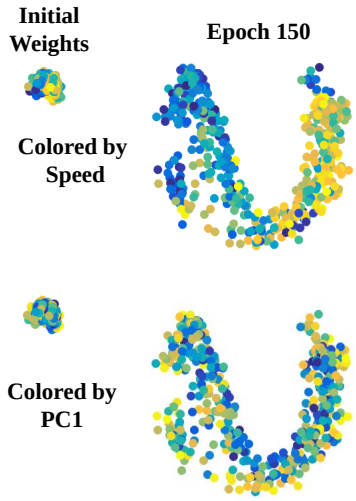


Figure 5: MDS of LSTM Space

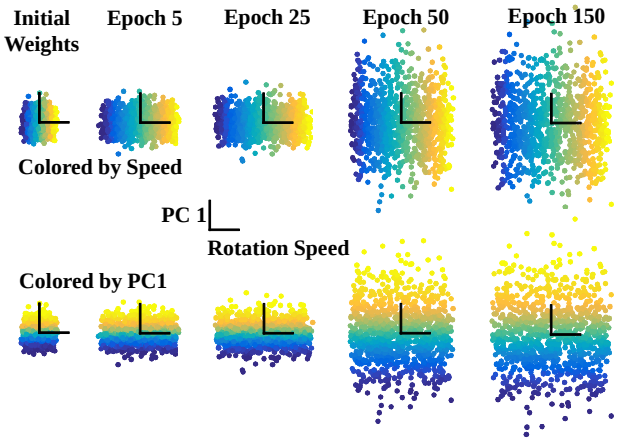


Figure 6: Projection of LSTM feature space on latent variables axes. Axes are in the direction of regression coefficients. A different regression was fit for each epoch.

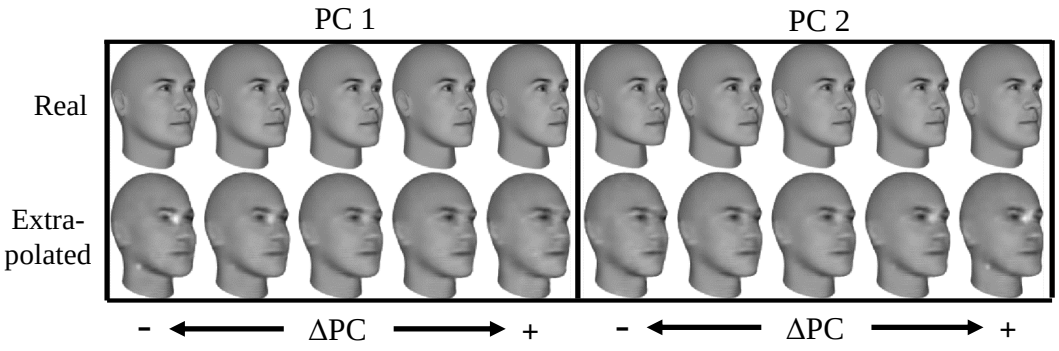


Figure 7: Linearly moving through LSTM feature space along principle component axes.

the last input frame instead of predict (denoted as **AE LSTM (dynamic)** in Fig. 8). The next model was similar but was trained on static videos [**AE LSTM (static)**]. A video was created for each unique frame in the original training set. For the last two models, the LSTM was replaced by a fully-connected (FC) layer, one with an equal number of weights [**AE FC (= # weights)**] and the other with an equal number of units [**AE FC (= # units)**] as the LSTM. These were trained in a more typical autoencoder fashion to simply reconstruct single frames, using every frame in the original video set. All comparisons were made with MSE-trained models.

The classification dataset consisted of 50 randomly generated faces at 12 equally-spaced angles between $[-\frac{\pi}{2}, \frac{\pi}{2}]$. A support vector machine (SVM) was fit on the feature representations of each model. For the models containing the LSTM layer, the feature representation at the fifth time step was chosen. To test for tranformation tolerance, training and testing were done with separate sets of angles.

The classification performance curves are shown in Figure 8. While all models show improvement compared to random initial weights, the PGN strongly outperforms the controls, having the highest score for each size of the training data. Its advantage is the most extreme at intermediate training sizes, where the it adds ~7-10% performance upon the next best model.

## 6 Conclusion

In extending ideas of predictive coding and learning from temporal continuity to modern, deep learning architectures, we have shown that an unsupervised, predictive loss can result in a rich internal

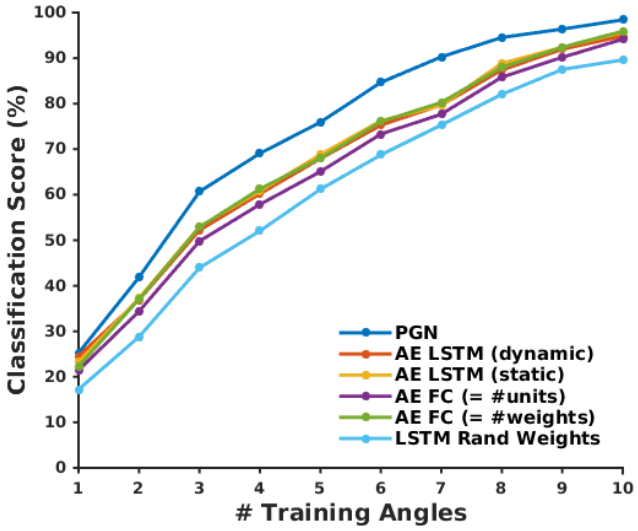


Figure 8: Face identity classification accuracy. AE: Autoencoder.

representation of visual objects. Models trained with such a loss successfully learn to predict future image frames in several contexts, ranging from the physics of simulated bouncing balls to the out-of-plane rotations of previously unseen computer-generated faces. However, importantly, models trained with a predictive unsupervised loss are also well-suited tasks beyond the domain of video sequences. For instance, representations trained with a predictive loss outperform other models of comparable complexity in a supervised classification problem with static images. This effect is particularly pronounced in the regime where a classifier must operate from just a few example views of a new object (in this case, face). Taken together, these results support the idea that prediction can serve as a powerful framework for developing transformation-tolerant object representations of the sort needed to support one- or few-shot learning.

The experiments presented here are all done in the context of highly-simplified artificial worlds, where the underlying generative model of the stimuli is known, and where the number of degrees of freedom in the data set are few. While extending these experiments to real world imagery is an obvious future priority, we nonetheless argue that experiments with highly controlled stimuli hold the potential to yield powerful guiding insights. Understanding how to scale predictive generative models of this form to encompass all of the transformation degrees of freedom found in real-world objects is an area of great interest for future research.

### ACKNOWLEDGMENTS

We would like to thank Chuan-Yung Tsai for fruitful discussions. This work was supported in part by a grant from the National Science Foundation (NSF IIS 1409097).

## REFERENCES

Bengio, Y., Simard, P., and Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *Neural Networks, IEEE Transactions on*, 1994.

Bengio, Yoshua, Lamblin, Pascal, Popovici, Dan, and Larochelle, Hugo. Greedy layer-wise training of deep networks. In *NIPS*. 2006.

Chalasani, Rakesh and Principe, Jose C. Deep predictive coding networks. *CoRR*, abs/1301.3541, 2013.

Chollet, François. Keras, 2015. URL `http://keras.io/`.

Den Ouden, Hanneke EM, Kok, Peter, and De Lange, Floris P. How prediction errors shape perception, attention and motivation. *Frontiers in Psychology*, 2012.

Denton, Emily L., Chintala, Soumith, Szlam, Arthur, and Fergus, Robert. Deep generative image models using a laplacian pyramid of adversarial networks. *CoRR*, abs/1506.05751, 2015.

Erhan, Dumitru, Bengio, Yoshua, Courville, Aaron, Manzagol, Pierre-Antoine, Vincent, Pascal, and Bengio, Samy. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 11, 2010.

Földiák, Peter. Learning invariance from transformation sequences. *Neural Computation*, 1991.

Fragkiadaki, Katerina, Levine, Sergey, and Malik, Jitendra. Recurrent network models for kinematic tracking. *CoRR*, abs/1508.00271, 2015.

Gan, Zhe, Li, Chunyuan, Henao, Ricardo, Carlson, David, and Carin, Lawrence. Deep temporal sigmoid belief networks for sequence modeling. *CoRR*, abs/1509.07087, 2015.

Gauthier, Jon. Conditional generative adversarial nets for convolutional face generation. *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition*, 2014.

Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative adversarial nets. In *NIPS*. 2014.

Goroshin, Ross, Bruna, Joan, Tompson, Jonathan, Eigen, David, and LeCun, Yann. Unsupervised learning of spatiotemporally coherent metrics. *CoRR*, abs/1412.6056, 2015.

Graves, Alex. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013.

Hochreiter, Sepp and Schmidhuber, Jurgen. Long short-term memory. *Neural Computation*, 1997.

Kevin Jarrett, Koray Kavukcuoglu, MarcAurelio Ranzato and LeCun, Yann. What is the best multi-stage architecture for object recognition? In *ICCV*. 2009.

Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In *NIPS*. 2012.

Memisevic, Roland and Hinton, Geoffrey. Unsupervised learning of image transformations. In *CVPR*. 2007.

Michalski, Vincent, Memisevic, Roland, and Konda, Kishore. Modeling deep temporal dependencies with recurrent "grammar cells". In *NIPS*. 2014.

Mirza, Mehdi and Osindero, Simon. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

Mittelman, Roni, Kuipers, Benjamin, Savarese, Silvio, and Lee, Honglak. Structured recurrent temporal restricted boltzmann machines. In *ICML*, pp. 1647–1655. 2014.

Mohabi, Hossein, Collobert, Ronan, and Weston, Jason. Deep learning from temporal coherence in video. In *ICML*, pp. 737–744. 2009.

Palm, Rasmus Berg. Prediction as a candidate for learning deep hierarchical models of data. *Master's thesis, Technical University of Denmark*, 2012.

Pinto, Nicolas, Doukhan, David, DiCarlo, James J., and Cox, David D. A high-throughput screening approach to discovering good forms of biologically inspired visual representation. *PLoS Comput Biol*, 2009.

Ranzato, Marc'Aurelio, Szlam, Arthur, Bruna, Joan, Mathieu, Michaël, Collobert, Ronan, and Chopra, Sumit. Video (language) modeling: a baseline for generative models of natural videos. *CoRR*, abs/1412.6604, 2014.

Rao, Rajesh P. N. and Ballard, Dana H. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 1999.

Rao, Rajesh P. N. and Sejnowski, Terrence J. Predictive sequence learning in recurrent neocortical circuits. In *NIPS*. 2000.

Saxe, A., Bhand, M., Chen, Z., Koh, P. W., Suresh, B., and Ng, A. Y. On random weights and unsupervised feature learning. In *Workshop: Deep Learning and Unsupervised Feature Learning (NIPS)*. 2010.

Saxe, Andrew M., Mcclelland, James L., and Ganguli, Surya. Learning hierarchical category structure in deep neural networks. *Proc. of the Cognitive Science Society*, 2013.

Srivastava, Nitish, Mansimov, Elman, and Salakhutdinov, Ruslan. Unsupervised learning of video representations using lstms. *CoRR*, abs/1502.04681, 2015.

Summerfield, Christopher, Egner, Tobias, Greene, Matthew, Koechlin, Etienne, Mangels, Jennifer, and Hirsch, Joy. Predictive codes for forthcoming perception in the frontal cortex. *Science*, 314, 2006.

Sutskever, Ilya, Hinton, Geoffrey E., and Taylor, Graham W. The recurrent temporal restricted boltzmann machine. In *NIPS*. 2009.

Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc V. V. Sequence to sequence learning with neural networks. In *NIPS*. 2014.

Taylor, Graham W. and Hinton, Geoffrey. Factored conditional restricted boltzmann machines for modeling motion style. In *Proceedings of The 26th International Conference on Machine Learning*, pp. 1–8. 2009.

Tieleman, Tijmen and Hinton, Geoffrey. Lecture 6.5 - rmsprop, coursera. *Neural networks for machine learning*, 2012.

Wang, Xiaolong and Gupta, Abhinav. Unsupervised learning of visual representations using videos. *CoRR*, abs/1505.00687, 2015.

Wiskott, Laurenz and Sejnowski, Terrence J. Learning invariance from transformation sequences. *Neural Computation*, 2002.

Zhao, Mingmin, Zhuang, Chengxu, Wang, Yizhou, and Lee, Tai Sing. Predictive encoding of contextual relationships for perceptual inference, interpolation and prediction. *CoRR*, abs/1411.3815, 2014.