An Uncertain Future: Forecasting from Static Images using Variational Autoencoders

Jacob Walker[1]
jcwalker@cs.cmu.edu
Carl Doersch[2]
cdoersch@cs.cmu.edu
Abhinav Gupta[1]
abhinavg@cs.cmu.edu
Martial Hebert[1]
hebert@cs.cmu.edu

[1] The Robotics Institute,
School of Computer Science,
Carnegie Mellon University,
Pittsburgh, Pennsylvania

[2] Machine Learning Department,
School of Computer Science,
Carnegie Mellon University,
Pittsburgh, Pennsylvania

Abstract

In a given scene, humans can easily predict a set of immediate future events that might happen. However, pixel-level anticipation in computer vision is difficult because machine learning struggles with the ambiguity of predicting the future. In this paper, we focus on predicting the dense trajectory of pixels in a scene — what will move in the scene, where it will travel, and how it will deform over the course of one second. We propose a conditional variational autoencoder as a solution to this problem. In this framework, direct inference from the image shapes the distribution of possible trajectories while latent variables encode information that is not available in the image. We show that our method predicts events in a variety of scenes and can produce multiple different predictions for an ambiguous future. We also find that our method learns a representation that is applicable to semantic vision tasks. Our algorithm is trained on thousands of diverse, realistic videos and requires absolutely no human labeling—relying only on labels produced by pixel tracking.

1 Introduction

Visual prediction is one of the most fundamental and difficult tasks in computer vision. For example, consider the woman in the gym in Figure 1. We as humans, given the context of the scene and her sitting pose, know that she is probably performing squat exercises. However, going beyond the action label and predicting the future leads to multiple, richer possibilities. The woman might be on her way up and will continue to go up, or she might be on the way down and continue to descend further. Those motion trajectories might not be exactly vertical, as the woman might lean or move her arms back as she ascends. While there are multiple possibilities, the space of possible futures is significantly smaller than the space of all possible visual motions. For example, we know she is not going to walk forward, she is not going to perform an incoherent action such as a head-bob, and her torso will likely remain in one piece. In this paper, we propose to develop a generative framework which, given a static input image, outputs the space of possible future actions. The key here is that our model characterizes the whole distribution of future states and can be used to sample multiple possible future events.

Even if we acknowledge that our algorithm must produce a distribution over many possible predictions, it remains unclear what is the output space of futures the algorithm should be capable of predicting. An ideal algorithm would predict everything that might be relevant to a human or robot interacting with the scene, but this is far too complicated to be feasible with current methods. A more tractable approach is to predict dense trajectories [12], which are simpler than pixels but still capture most of a video's content. While this representation is intuitive, the output space is high dimensional and hard to parametrize, an issue which forced [12] to use a Nearest Neighbor algorithm and transfer raw trajectories. Unsurprisingly, the algorithm is computationally expensive and fails on testing images which do not have globally similar training images. Others have been successful in predicting the optical flow to the immediate next frame frame [6, 11]. However, this is very short term prediction. Other recent works have proposed predicting pixels [7, 9] or the high dimensional fc7 features [10] themselves. [10] depends on semantics, and direct pixel prediction suffers from a number of drawbacks. Notably, the output space is high dimensional and it is difficult to encode constraints on the output space, e.g., pixels can change colors every frame. There is also an averaging effect of multiple possible predictions which leads to blurry predictions.

In this paper, we propose to address these challenges. We propose to revisit the idea of predicting dense trajectories at each and every pixel using a feed-forward Convolutional Network. Using dense trajectories restricts the output space dramatically which allows our algorithm to learn robust models for visual prediction with the available data. However, the dense trajectories are still high-dimensional, and the output still has multiple modes. In order to tackle these challenges, we propose to use variational autoencoders to learn a low-dimensional latent representation of the output space conditioned on an input image. Specifically, given a single frame as input, our *conditional* variational auto-encoder outputs a mapping from noise variables—sampled from a normal distribution $\mathcal{N}(0,1)$—to output trajectories at every pixel. Thus, we can naively sample values of the latent variables and pass them through the mapping in order to sample different predicted trajectories from the inferred conditional distribution. Unlike other applications of variational autoencoders that generate outputs a priori [4, 5], we focus on generating them *given the image*. Conditioning on the image is a form of inference, restricting the possible motions based on object location and scene context. Sampling latent variables during test time then allows us to explore the space of possible actions in the given scene.

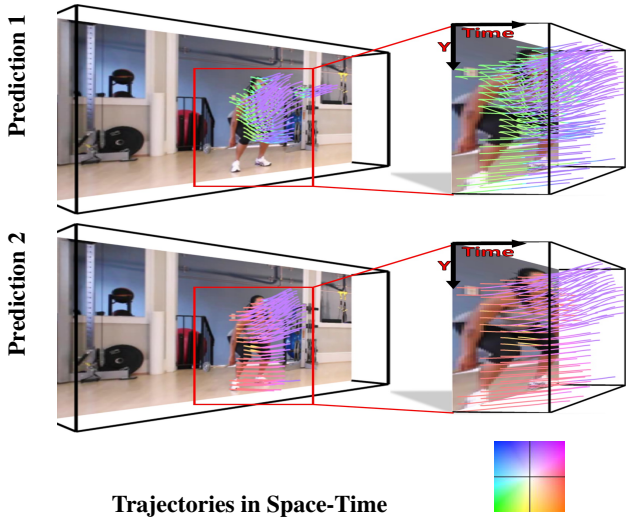
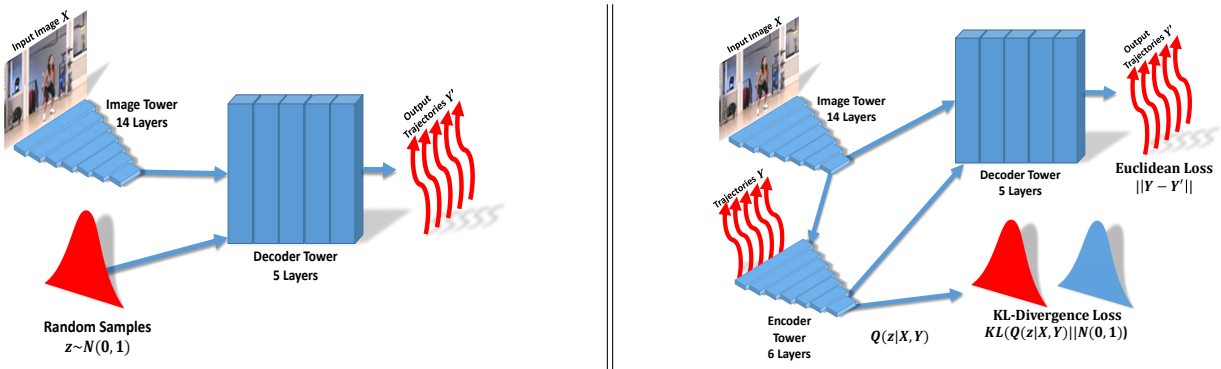


Figure 1: Consider this picture of a woman in the gym — she could move up or down. Our framework is able to predict multiple correct one-second motion trajectories given the scene. The directions of the trajectories at each point in time are color-coded according to the square from [1] on the right. The diagram shows the complexity of the predicted motions in space time.

Our paper makes three contributions. First, we demonstrate that prediction of dense pixel trajectories is a plausible approach to general, non-semantic, self-supervised visual prediction. Second, we propose a conditional variational auto-encoder as a solution to this problem, a model that performs inference on an image by conditioning the distribution of possible movements on a scene. Third, we show that our model is capable of learning representations for various recognition tasks with less data than conventional approaches.

2 Approach

A simple regressor—even a deep network with millions of parameters—will struggle with predicting one-second motion in a single image as there may be many plausible outputs. Our architecture augments the simple regression model by adding another input z to the regressor (shown in Figure 2(a)), which can account for the ambiguity. At test time, z is ran-



(a) Testing Architecture

(b) Training Architecture

Figure 2: Overview of the architecture. During training, the inputs to the network include both the image and the ground truth trajectories. A variational autoencoder encodes the joint image and trajectory space, while the decoder produces trajectories depending both on the image information as well as output from the encoder. During test time, the only inputs to the decoder are the image and latent variables sampled from a normal distribution.

dom Gaussian noise: passing an image as input and sampling from the noise variable allows us to sample from the model's posterior given the image. That is, if there are multiple possible futures given an image, then for each possible future, there will be a different set of z values which map to that future. Furthermore, the likelihood of sampling each possible future will be proportional to the likelihood of sampling a z value that maps to it. Note that we assume that the regressor—in our case, a deep neural network—is capable of encoding dependencies between the output trajectories. In practice, this means that if two pixels need to move together even if the direction of motion is uncertain, then they can simply be influenced by the same dimension of the z vector.

2.1 Training by "Autoencoding"

It is straightforward to sample from the posterior at test time, but it is much less straightforward to train a model like this. The problem is that given some ground-truth trajectory Y, we cannot directly measure the probability of the trajectory given an image X under a given model; this prevents us from performing gradient descent on this likelihood. It is in theory possible to estimate this conditional likelihood by sampling a large number of z values and constructing a Parzen window estimate using the resulting trajectories, but this approach by itself is too costly to be useful.

Variational Autoencoders [4, 5] make this approach tractable. The key insight is that the vast majority of samples z contribute almost nothing to the overall likelihood of Y. Hence, we should instead focus only on those values of z that are likely to produce values close to Y. We do this by adding another pathway Q, as shown in Figure 2(b), which is trained to map the output Y to the values of z which are likely to produce them. That is, Q is trained to "encode" Y into the latent z space such that the values can be "decoded" back to the trajectories. The entire pipeline can be trained end-to-end using reconstruction error. An immediate objection one might raise is that this is essentially "cheating" at training time: the model sees the values that it is trying to predict, and may just copy them to the output. To prevent the model from simply copying, we force the encoding to be lossy. The Q pathway does not produce a single z, but instead, produces a distribution over z values, which we sample from before decoding the trajectories. We then directly penalize the information content in this distribution, by penalizing the $\mathcal{KL}$-divergence between the distribution produced by Q and the trajectory-agnostic $\mathcal{N}(0,1)$ distribution. The model is thereby encouraged to extract as much information as possible from the input image before relying on encoding the trajectories themselves. Surprisingly, this formulation is a very close approximation to maximizing the posterior likelihood $P(Y|X)$ that we are interested in. In fact, if our encoder pathway Q can estimate the exact distribution of z's that are likely to generate Y, then the approximation is exact.

3 Experimental results

We utilized the UCF101 dataset [8] to train our model. Testing data for quantitative evaluation came from the testing portion of the THUMOS 2015 challenge dataset [3]. The UCF101 dataset is the training dataset for the THUMOS challenge, and thus THUMOS is a relevant choice for the testing set. We use two baselines for trajectory prediction. The first is a direct regressor (i.e., no autoencoder) for trajectories using the same layer architecture from the image data tower. The second baseline is the optical flow prediction network from [11], which was trained on the same dataset. We simply extrapolate the predicted motions of the network over one second.

On two different metrics—log likelihood via Parzen window estimation as well as minimum Euclidean distance—our method outperforms both baselines. This is reasonable since the regressor is inherently unimodal: it is unable to predict distributions where there may be many reasonable futures, Interestingly, extrapolating the predicted optical flow from [11] does not seem to be effective, as motion may change direction considerably even over the course of one second.

We also evaluate the representation learned by our network on the task of object detection. We take layers from the image tower and fine-tune them on the PASCAL 2012 training dataset. We find that from a relatively small amount of data, our method outperforms other methods that were trained on datasets with far larger diversity in scenes and types of objects. While the improvement is small over all objects, we do have the highest performance on humans over all unsupervised methods, even [2]. This is expected as most of the moving objects in our training data comes from humans.

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31, 2011.

[2] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.

[3] A. Gorban, H. Idrees, Y.-G. Jiang, A. Roshan Zamir, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. `http://www.thumos.info/`, 2015.

[4] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *ICLR*, 2014.

[5] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *NIPS*, 2014.

[6] Silvia L Pintea, Jan C van Gemert, and Arnold WM Smeulders. Déjà vu: Motion prediction in static images. In *ECCV*, 2014.

[7] MarcAurelio Ranzato, Arthur Szlam, Joan Bruna, Michael Mathieu, Ronan Collobert, and Sumit Chopra. Video (language) modeling: a baseline for generative models of natural videos. *arXiv preprint arXiv:1412.6604*, 2014.

[8] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[9] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using LSTMs. In *ICML*, 2015.

[10] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating the future by watching unlabeled video. In *CVPR*, 2016.

[11] Jacob Walker, Abhinav Gupta, and Martial Hebert. Dense optical flow prediction from a static image. In *ICCV*, 2015.

[12] Jenny Yuen and Antonio Torralba. A data-driven approach for event prediction. In *ECCV*, 2010.