Visual Dynamics: Probabilistic Future Frame Synthesis via Cross Convolutional Networks

Tianfan Xue[1]*, Jiajun Wu[1]*, Katherine L. Bouman[1]
{tfxue,jiajunwu,klbouman}@mit.edu
William T. Freeman[12]
billf@mit.edu

[1] Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
[2] Google Research, Cambridge

Abstract

We study the problem of synthesizing a number of likely future frames from a single input image. In contrast to traditional methods, which have tackled this problem in a deterministic or non-parametric way, we propose a novel approach that models future frames in a probabilistic manner. Our probabilistic model makes it possible for us to sample and synthesize many possible future frames from a single input image. Future frame synthesis is challenging, as it involves low- and high-level image and motion understanding. We propose a novel network structure, namely a *Cross Convolutional Network*, to aid in synthesizing future frames; this network encodes image and motion information as feature maps and convolutional kernels, respectively. In experiments, our model performs well on aligned real-world videos. We also show that our model can be applied to tasks such as visual analogy-making.

1 Introduction

From just a single snapshot, humans are often able to imagine how a scene will visually change over time. For instance, due to the pose of the girl in Figure 1, most would predict that her arms are stationary but her leg is moving. However, the exact motion is often unpredictable due to intrinsic ambiguity. Is the girl's leg moving up or down? In this work, we study the problem of *visual dynamics*: modeling the conditional distribution of future frames given an observed image. We propose to tackle this problem using a probabilistic, content-aware motion prediction model that learns this distribution without using annotations. Sampling from this model allows us to visualize the many possible ways that an input image is likely to change over time.

Modeling the conditional distribution of future frames given only a single image as input is a very challenging task for a number of reasons. First, natural images come from a very high dimensional distribution that is difficult to model. Modeling the conditional distribution of future frames further increases the dimensionality of the problem. Not only do the sampled, synthesized images need to look like real images, the motion between the input and synthesized images should also be realistic. Second, in order to properly predict motion distributions, the model must first learn about image parts and the correlation of their respective motions in a unsupervised fashion.

In this work, we propose a neural network structure, based on a variational autoencoder [2] and our newly proposed cross convolutional layer, to tackle this problem. During training, the network observes a set of consecutive image pairs in videos, and automatically infers the relationship between images in each pair without any supervision. Then, during testing, the network predicts the conditional distribution, $P(J|I)$, of future RGB images J (Figure 1b) given an RGB input image I that was not in the training set (Figure 1a). Using this distribution, the network is able to synthesize multiple different image samples corresponding to possible future frames of the input image (Figure 1c). Our network contains a number of key components that contribute to its success:

- We use conditional variational autoencoder to model the complex conditional distribution of future frames [2]. This allows us to approximate a sample, J, from the distribution of future images by using a trainable function $J = f(I,z)$. The argument z is a sample from a simple distribution, *e.g.* Gaussian, which introduces randomness into the sampling of J. This formulation makes the problem of learning the distribution much more tractable than explicitly modeling the distribution.

- Instead of finding an intrinsic representation of the image itself, as most previous work has done [1, 4, 5], or modeling a motion field [6], our network finds an intrinsic representation of intensity changes between two images, also known as the *difference image* This representation is typically sparser and easier to model than content in an original image.

- We model motion using a set of image-dependent convolution kernels operating over an image pyramid. Unlike [1], our proposed cross convolutional network allows us to jointly learn these kernels with features maps from observed frames, and convolve them to synthesize a probable future frame.

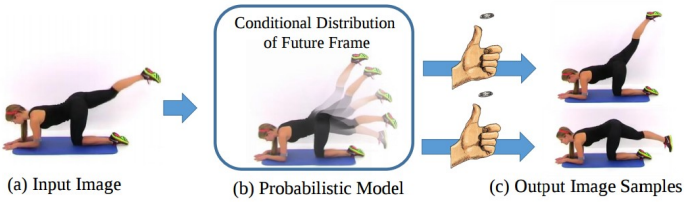


Figure 1: The precise motion corresponding to a snapshot image in time is often ambiguous. For instance, is the girl's leg in (a) moving up or down? We propose a probabilistic, content-aware motion prediction model (b) that learns the conditional distribution of future frames. Using this model we are able to predict and synthesize various future frames (c) that are all consistent with the observed input image (a).

We test the proposed model on a dataset generated from real videos. We show that, given an RGB input image, the algorithm can successfully model a distribution of possible future frames, and generate different samples that cover a variety of realistic motions. We also demonstrate that our model can be easily applied to tasks such as visual analogy-making.

2 Formulation: Conditional Variational Autoencoder

We formulate this problem using a conditional variational autoencoder, following [2, 3]. Consider the following generative process that samples a future frame from a θ parametrized model, conditioned on an observed image I. First the algorithm samples the hidden variable z from a prior distribution $p_z(z)$; in this work we assume $p_z(z)$ is a multivariate Gaussian distribution where each dimension is i.i.d. with zero-mean and unit-variance. Then, given a value of z, the algorithm samples the intensity difference image v from the conditional distribution $p_\theta(v|I,z)$. The final image, $J = I + v$, is then returned as output.

Objective Function In the training stage, the algorithm attempts to maximize the log-likehood of the conditional marginal distribution $\sum_i \log p(v^{(i)}|I^{(i)})$. Assuming I and z are independent, the marginal distribution is expanded as $\sum_i \log \int_z p(v^{(i)}|I^{(i)},z)p_z(z)dz$. Directly maximizing this marginal distribution is hard, thus we instead maximize its variational upper-bound, as proposed by [2]. Each term in the marginal distribution is upper-bounded by $\mathcal{L}(\theta,\phi,v^{(i)}|I^{(i)})$ defined as

$$-D_{\mathrm{KL}}(q_\phi(z|v^{(i)},I^{(i)})||p_z(z)) + E_{q_\phi(z|v^{(i)},I^{(i)})}\left[\log p_\theta(v^{(i)}|z,I^{(i)})\right], \quad (1)$$

where D_{KL} is the KL-divergence, and $q_\phi(z|v^{(i)},I^{(i)})$ is the variational distribution that approximates the posterior $p(z|v^{(i)},I^{(i)})$. For simplicity, we refer to the conditional data distribution, $p_\theta(v^{(i)}|z,I^{(i)})$, as the *generative model*, and the variational distribution, $q_\phi(z|v^{(i)},I^{(i)})$, as the *recognition model*.

The first KL-divergence term in Eq. 1 has an analytical form. To make the second term tractable, we approximate the variational distribution, $q_\phi(z|x^{(i)},I^{(i)})$, by its empirical distribution as follows

$$-D_{\mathrm{KL}}(q_\phi(z|v^{(i)},I^{(i)})||p_z(z)) + \frac{1}{L}\sum_{l=1}^{L}\left[\log p_\theta(v^{(i)}|z^{(i,l)},I^{(i)})\right], \quad (2)$$

* indicates equal contributions

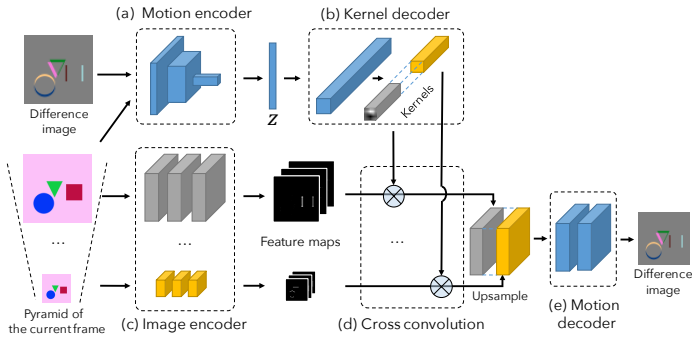
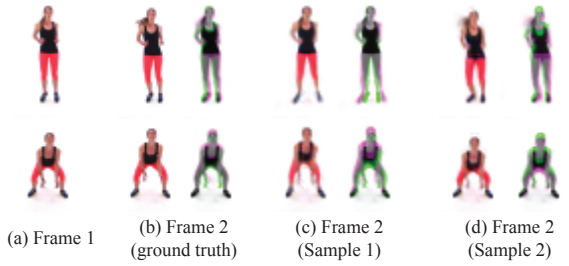


Figure 2: Our network consists of five components: (a) a motion encoder, (b) a kernel decoder, (c) an image encoder, (d) a cross convolution layer, and (e) a motion decoder. Our image encoder takes images at four scales as input, while for simplicity we only show two in the figure.



(a) Frame 1 (b) Frame 2 (ground truth) (c) Frame 2 (Sample 1) (d) Frame 2 (Sample 2)

Figure 3: Sampled novel future frames. For each 'Frame 2' we show the RGB image along with an overlay of green and magenta versions of the 2 consecutive frames to illustrate motion.

where $z^{(i,l)}$ are samples from the variational distribution.

Distribution Reparametrization Now we need to define distributions for the generative model, $p_\theta(v^{(i)}|z^{(i,l)},I^{(i)})$, and for the recognition model, $q_\phi(z^{(i,l)}|v^{(i)},I^{(i)})$. Using the reparameterization trick [2], we approximate both distributions as Gaussian, where the mean and the variance of the distributions are functions specified by a generative network and a recognition network, respectively. Specifically, let us define*:

$$p_\theta(v^{(i)}|z^{(i,l)},I^{(i)}) = \mathcal{N}(v^{(i)};f_{\text{mean}}(z^{(i,l)},I^{(i)}),\sigma^2\mathbf{I}), \quad (3)$$

$$q_\phi(z^{(i,l)}|v^{(i)},I^{(i)}) = \mathcal{N}(z^{(i,l)};g_{\text{mean}}(v^{(i)},I^{(i)}),g_{\text{var}}(v^{(i)},I^{(i)})), \quad (4)$$

where $\mathcal{N}(\cdot;a,b)$ is a conditional data distribution with mean a and variance b. f_{mean} is a function that predicts the mean of the variational distribution, defined by the generative network. g_{mean} and g_{var} are functions that predict the mean and variance of the variational distribution, respectively, defined by the recognition network. In the next section, we will describe the details of the network structure.

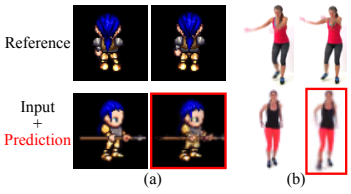
3 Layered Motion Representations and Cross Convolutional Networks

Motion can often be decomposed in a layer-wise manner [7]. Intuitively, different semantic segments in an image should have different distributions over all possible motions; for example, a building is often static, but a river flows.

To model the layered motion, we propose a novel cross convolutional network (Figure 2). The network first decomposes an input image pyramid into multiple feature maps through an image encoder (Figure 2(c)). It then convolves these maps using different convolutional kernels (Figure 2(d)), and uses the outputs to synthesize a difference image (Figure 2(e)). This network structure naturally fits the layered motion representation, as each feature map characterizes an image *layer* (note this is different from a network *layer*) and the corresponding kernel characterizes the motion of that layer. In other words, we model motions as convolutional kernels, which are applied to image segments (feature maps) at multiple scales.

Unlike a traditional convolutional network, these kernels used in our network should not be identical for all inputs, as different images typically have different motions (kernels). We therefore propose a novel cross convolutional layer to tackle this problem. The cross convolutional layer does not learn the weights of the kernels itself. Instead, it takes both kernel weights and feature maps as input and computes convolution during a forward pass; for back propagation, it also computes the gradients of both convolutional kernels and feature maps.

*Here the bold $\mathbf{I}$ denotes an identity matrix, whereas the normal-font I denotes the observed image.



Model	MSE
Add [5]	56.0
Dis [5]	56.5
Dis + Cls [5]	23.0
Our Model	**15.9**

Figure 4: Left: results on visual analogy-making. Right: comparison with [5]. Mean squared pixel error (MSE) on test analogies is used as the metric.

At last, in order to find the intrinsic representation z of motion to be sampled in the testing time, we include a motion encoder and a kernel decoder. The motion encoder (Figure 2(a)) is a variational autoencoder that learns the compact representation z of possible motions. The kernel decoder (Figure 2(b)) is a network that decodes the compact motion representation z into motion kernels.

In sum, the motion encoder forms the recognition functions g_{mean} and g_{var}, whereas the image encoder, the kernel decoder, the cross convolutional layer, and the motion decoder form the generative function f_{mean}. During training, the image encoder takes a single frame $I^{(i)}$ as input, and the motion encoder takes both $I^{(i)}$ and the difference image $v^{(i)} = J^{(i)} - I^{(i)}$ as input, where $J^{(i)}$ is the next frame. The network aims to regress the difference image using an l-2 loss. During testing, the image encoder still sees a single image I; however, instead of using a motion encoder, we directly sample motion vectors $z^{(j)}$ from the prior distribution $p_z(z)$.

4 Experiments

We first collect 20 workout videos from YouTube, each about 30 to 60 minutes long. We extract 56,838 pairs of frames for training and 6,243 pairs for testing. The training and testing pairs come from different video sequences. Figure 3 shows that our framework works well in predicting the movement of the legs and torso.

We further conduct behavior experiments on Amazon Mechanical Turk to quantitatively evaluate the algorithm. We randomly select 200 images, sample possible next frames using our algorithm, and show them to multiple human subjects as an animation side by side with the ground truth animation. We then ask the subject to choose which animation is real (not synthesized). An ideal algorithm should achieve a success rate of 50% and our algorithm achieves 31.3%, demonstrating the effectiveness of the proposed network.

Inspired by some recent work on visual analogy-making [5], we also demonstrate that our framework can be easily applied to the same task, even without supervision on analogies during training. Specifically, [5] studied the problem of inferring the relationship between a pair of images and synthesizing a new image by applying the inferred relationship to a new input image. Our motion encoder, which aims to extract motion information from two consecutive frames, can also be used to extract and synthesize relationships between pairs of images, as shown in Figure 4. Although our method requires no analogy supervision, it still performs better than those introduced in [5], which uses visual analogy labels during training.

[1] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *arXiv preprint arXiv:1605.07157*, 2016.

[2] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

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

[4] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016.

[5] Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. In *NIPS*, 2015.

[6] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *ECCV*, 2016.

[7] John YA Wang and Edward H Adelson. Layered representation for motion analysis. In *CVPR*, 1993.