

Stochastic Adversarial Video Prediction

Alex X. Lee, Richard Zhang, Frederik Ebert,
Pieter Abbeel, Chelsea Finn, and Sergey Levine

University of California, Berkeley
`{alexlee_gk,rich.zhang,febert,pabbeel,
cbfinn,svlevine}@cs.berkeley.edu`

Abstract. Being able to predict what may happen in the future requires an in-depth understanding of the physical and causal rules that govern the world. A model that is able to do so has a number of appealing applications, from robotic planning to representation learning. However, learning to predict raw future observations, such as frames in a video, is exceedingly challenging—the ambiguous nature of the problem can cause a naively designed model to average together possible futures into a single, blurry prediction. Recently, this has been addressed by two distinct approaches: (a) latent variational variable models that explicitly model underlying stochasticity and (b) adversarially-trained models that aim to produce naturalistic images. However, a standard latent variable model can struggle to produce realistic results, and a standard adversarially-trained model underutilizes latent variables and fails to produce diverse predictions. We show that these distinct methods are in fact complementary. Combining the two produces predictions that look more realistic to human raters and better cover the range of possible futures. Our method outperforms prior and concurrent work in these aspects.

Keywords: video prediction, GANs, variational autoencoder

1 Introduction

When we interact with objects in our environment, we can easily imagine the consequences of our actions: push on a ball and it will roll; drop a vase and it will break. The ability to imagine future outcomes provides an appealing avenue for learning about the world. Unlabeled video sequences can be gathered autonomously with minimal human intervention, and a machine that learns to accurately predict future events will have gained an in-depth and functional understanding of its physical environment. This leads naturally to the problem of video prediction—given a sequence of context frames, and optionally a proposed action sequence, generate the raw pixels of the future frames. Once trained, such a model could be used to determine which actions can bring about desired outcomes [3,4]. Unfortunately, accurate and naturalistic video prediction remains an open problem, with state-of-the-art methods producing high-quality predictions only a few frames into the future.

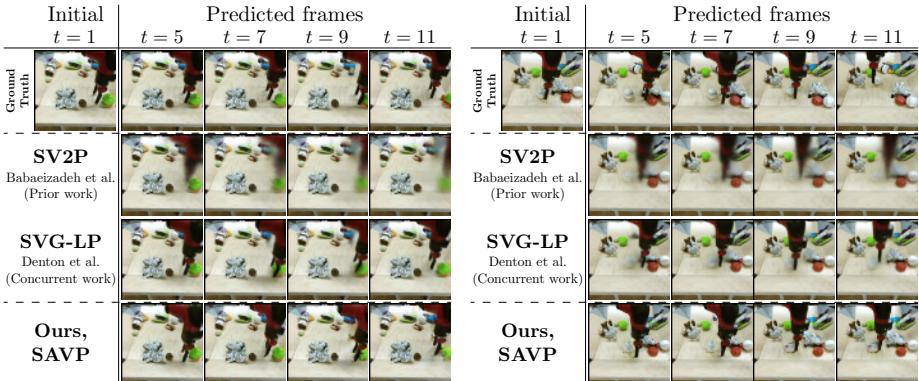


Fig. 1: **Example results.** We show example predictions for two video sequences comparing our method to prior and concurrent work. All methods are conditioned on two initial frames and predict 10 future frames (only some frames are shown). The predictions are stochastic, and we show the closest sample to the ground truth (out of 100 samples). While the prior SV2P method [1] produces blurry, unrealistic images, our method maintains sharpness and realism through time. We also compare to the concurrent SVG-LP method [2], which produces sharper predictions, but still blurs out objects in the background (left) or objects that interact with the robot arm such as the baseball (right).

One major challenge in video prediction is the ambiguous nature of the problem. While frames in the immediate future can be extrapolated with high precision, the space of possibilities diverges beyond a few frames, and the problem becomes multimodal by nature. Methods that use deterministic models and loss functions unequipped to handle this inherent uncertainty, such as mean-squared error (MSE), will average together possible futures, producing blurry predictions. Prior [1] and concurrent [2] work has explored stochastic models for video prediction, using the framework of variational autoencoders (VAEs) [5]. These models predict possible futures by sampling latent variables. During training, they optimize a variational lower bound on the likelihood of the data in a latent variable model. However, the posterior is still a pixel-wise MSE loss, corresponding to the log-likelihood under a fully factorized Gaussian distribution. This makes training tractable, but causes them to still make blurry and unrealistic predictions when the latent variables alone do not adequately capture the uncertainty.

Another relevant branch of recent work has been generative adversarial networks (GANs) [6] for image generation. Here, a generator network is trained to produce images that are indistinguishable from real images, under the guidance of a learned discriminator network trained to classify images as real or generated. The learned discriminator operates on patches or entire images, and is thus capable of modeling the joint distribution of the generated pixels, without making any independence assumptions. Although this overcomes the limitations of pixel-wise losses, GANs are notoriously susceptible to mode collapse, where latent random variables are often ignored by the model, especially in the conditional setting [7,8,9,10]. This makes them difficult to apply to generation of diverse and plausible futures, conditioned on context frames. A handful of re-

cent works have explored video prediction and generation with GANs, typically with models that are either deterministic [7,11,12,13], not conditioned on context images (and therefore applicable only to unconditional generation, rather than prediction) [14,15], or incorporate stochasticity only through input noise [16,17]. As we illustrate in our experiments, GANs without an explicit latent variable interpretation are prone to mode collapse and do not generate diverse futures.

To address these challenges, we propose a model that combines both adversarial losses and latent variables to enable high-quality stochastic video prediction. Our model consists of a video prediction network that can sample multiple plausible futures by sampling time-varying stochastic latent variables and decoding them into multiple frames. At training time, an inference network estimates the distribution of these latent variables, and video discriminator networks classify generated videos from real. The full training objective is the variational lower bound used in VAEs combined with the adversarial loss used in GANs. This enables to capture stochastic posterior distributions of videos while also modeling the spatiotemporal joint distribution of pixels. This formalism, referred to as VAE-GANs, was originally proposed for image generation [18] and recently used in the conditional setting of multimodal image-to-image translation [10]. To our knowledge, our work is the first to extend this approach to stochastic video prediction.

A recurring challenge in video prediction is the choice of evaluation metric. While simple quantitative metrics such as the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) index [19] provide appealing ways to measure the differences between various methods, they are known to correspond poorly to human preferences [20,21]. In fact, SSIM was not designed for situations where spatial ambiguities are a factor [22]. Indeed, in our quantitative comparisons, we show that these metrics rarely provide accurate cues as to the relative ordering of the various methods in terms of prediction quality. We include a detailed ablation analysis of our model and its design decisions on these standard metrics, in comparison with prior work. We also present an evaluation on a “Visual Turing test”, using human judgments via the two-alternative forced choice (2AFC) method, which shows that the best models according to human raters are generally not those that excel on the standard metrics. In addition, we use a perceptual distance metric [23,24,21] to assess the diversity of the output videos, and also as an alternative to the standard full-reference metrics used in video prediction.

The primary contribution of our work is an stochastic video prediction model, based on VAE-GANs. To our knowledge, this is the first stochastic video prediction model in the literature that combines an adversarial loss with a latent variable model trained via the variational lower bound. Our experiments show that the VAE component greatly improves the diversity of the generated images, while the adversarial loss attains prediction results that are substantially more realistic than both prior [1] and concurrent [2] (e.g. see Fig. 1). We further present a systematic comparison of various types of prediction models and losses, including VAE, GAN, and VAE-GAN models, and analyze the impact of these design decisions on image realism, prediction diversity, and accuracy.

2 Related Work

Recent developments in expressive generative models based on deep networks has led to impressive developments in video generation and prediction. Prior methods vary along two axes: the design of the generative model and training objective. Earlier approaches to prediction focused on models that generate pixels directly from the latent state of the model using both feedforward [25,7] and recurrent [26,27] architectures. An alternative to generating pixels is transforming them by applying a constrained geometric distortion to a previous frame [3,28,29,30,11,31,32,16,13,33,34,35]. These methods can potentially focus primarily on motion, without explicitly reconstructing appearance.

Aside from the design of the generator architecture, performance is strongly affected by the training objective. Simply minimizing mean squared error can lead to good predictions for deterministic synthetic videos, such as video game frames [26,36]. However, on real-world videos that contain ambiguity and uncertainty, this objective can result in blurry predictions, as the model averages together multiple possible futures to avoid incurring a large squared error [7].

Incorporating uncertainty is critical for addressing this issue. One approach is to model the full joint distribution using pixel-autoregressive models [37,38], which can produce sharp images, though training and inference are impractically slow, even with parallelism [39]. Another approach is to train a latent variable model, such as in variational autoencoders (VAEs) [5]. Conditional VAEs have been used for prediction of optical flow trajectories [34], single-frame prediction [29], and very recently for stochastic multi-frame video prediction [1]. Concurrent with our work, Denton et al. proposed an improved VAE-based model that can synthesize physically plausible futures for longer horizons [2]. While these models can model distributions over possible futures, the prediction distribution is still fully factorized over pixels, which still tends to produce blurry predictions. Our experiments present a detailed comparison to recent prior [1] and concurrent [2] methods. The basic VAE-based version of our method slightly outperforms Denton et al. [2] in human subjects comparisons, and substantially outperforms all methods, including [2], when augmented with an adversarial loss.

Adversarial losses in generative adversarial networks (GANs) for image generation can produce substantially improved realism [6]. However, adversarial loss functions tend to be difficult to tune, and these networks are prone to the mode collapse problem, where the generator fails to adequately cover the space of possible predictions and instead selects one or a few prominent modes. A number of prior works have used adversarial losses for deterministic video prediction [7,11,12,13,40,41]. Liang et al. [35] proposed to combine a stochastic model with an adversarial loss, but the proposed optimization does not optimize a latent variable model, and the latent variables are not sampled from the prior at test-time. In contrast, models that incorporate latent variables via a variational bound (but do not include an adversarial loss) use an encoder at training-time that performs inference from the entire sequence, and then sample the latents from the prior at test-time [1,2]. Several prior works have also sought to produce unconditioned video generations [14,15,17] and conditional generation with

input noise [16,17]. In our comparison, we show that a GAN model with input noise can indeed generate realistic videos, but fails to adequately cover the space of possible futures. In contrast, our approach, which combines VAE-based latent variable models with an adversarial loss, produces predictions that are both visually plausible and highly diverse.

Evaluating the performance of video prediction methods is a major challenge. Standard quantitative metrics, such as PSNR and SSIM [19], provide a convenient score for comparison, but do not equate to the qualitative rankings provided by human raters, as discussed in [20] and illustrated in our evaluation. The success of these methods can also be evaluated indirectly, such as evaluating the learned representations [42,43,44], or using the predictions for planning actions of a robotic system in the real world [45,4,46]. We argue that it is difficult to quantify the quality of video prediction with a single metric, and instead propose to use a combination of quantitative metrics that include subjective human evaluations, and automated measures of prediction diversity and accuracy. Our experiments show that VAE-based latent variable models are critical for prediction diversity, while adversarial losses contribute the most to realism, and we therefore propose to combine these two components to produce predictions that both cover the range of possible futures and provide realistic images.

3 Video Prediction with Stochastic Adversarial Models

Our goal is to learn an stochastic video prediction model that can predict videos that are diverse and perceptually realistic, and where all predictions are plausible futures for the given initial image. In practice, we use a short initial sequence of images (typically two frames), though we will omit this in our derivation for ease of notation. Our model consists of a recurrent generator network G , which is a deterministic video prediction model that maps an initial image \mathbf{x}_0 and a sequence of latent random codes $\mathbf{z}_{0:T-1}$, to the predicted sequence of future images $\hat{\mathbf{x}}_{1:T}$. Intuitively, the latent codes encapsulate any ambiguous or stochastic events that might affect the future. At test time, we sample videos by first sampling the latent codes from a prior distribution $p(\mathbf{z}_t)$, and then passing them to the generator. We use a fixed unit Gaussian prior, $\mathcal{N}(0, 1)$. The training procedure for this includes elements of variational inference and generative adversarial networks. Before describing the training procedure, we formulate the problem in the context of VAEs and GANs.

Variational autoencoders. Our recurrent generator predicts each frame given the previous frame and a random latent code. The previous frames passed to the generator are either the ground truth frames, when available (i.e. the initial sequence of images), or the previous predicted frames. The generator specifies a distribution $p(\mathbf{x}_t | \mathbf{x}_{0:t-1}, \mathbf{z}_{0:t-1})$, and it is parametrized as a fixed-variance Laplacian distribution with the mean given by $\hat{\mathbf{x}}_t = G(\mathbf{x}_0, \mathbf{z}_{0:t-1})$.

The likelihood of the data $p(\mathbf{x}_{1:T} | \mathbf{x}_0)$ cannot be directly maximized, since it involves marginalizing over the latent variables, which is intractable in general.

Thus, we instead maximize the variational lower bound of the log-likelihood. We approximate the posterior with a recognition model $q(\mathbf{z}_t | \mathbf{x}_{t:t+1})$, which is parametrized as a conditionally Gaussian distribution $\mathcal{N}(\mu_{\mathbf{z}_t}, \sigma_{\mathbf{z}_t}^2)$, represented by a deep network $E(\mathbf{x}_{t:t+1})$. The encoder E is conditioned on the frames of the adjacent time steps to allow the latent variable \mathbf{z}_t to encode any ambiguous information of the transition between frames \mathbf{x}_t and \mathbf{x}_{t+1} .

During training, the latent code is sampled from $q(\mathbf{z}_t | \mathbf{x}_{t:t+1})$. The generation of each frame can be thought of as the reconstruction of frame $\hat{\mathbf{x}}_{t+1}$, where the ground truth frame \mathbf{x}_{t+1} (along with \mathbf{x}_t) is encoded into a latent code \mathbf{z}_t , and then it (along with the last frame) is mapped back to $\hat{\mathbf{x}}_{t+1}$. Since the latent code has ground truth information of the frame being reconstructed, the model is encouraged to use it during training. This is a conditional version of the variational autoencoder (VAEs) [5], where the encoder and decoder are conditioned on the previous frame (\mathbf{x}_t or $\hat{\mathbf{x}}_t$). To allow back-propagation through the encoder, the reconstruction term is rewritten using the re-parametrization trick [5],

$$\mathcal{L}_1(G, E) = \mathbb{E}_{\mathbf{x}_{0:T}, \mathbf{z}_t \sim E(\mathbf{x}_{t:t+1})|_{t=0}^{T-1}} \left[\sum_{t=1}^T \|\mathbf{x}_t - G(\mathbf{x}_0, \mathbf{z}_{0:t-1})\|_1 \right]. \quad (1)$$

To enable sampling from the prior at test time, a regularization term encourages the approximate posterior to be close to the prior distribution,

$$\mathcal{L}_{\text{KL}}(E) = \mathbb{E}_{\mathbf{x}_{0:T}} \left[\sum_{t=1}^T \mathcal{D}_{\text{KL}}(E(\mathbf{x}_{t-1:t}) || p(\mathbf{z}_{t-1})) \right]. \quad (2)$$

The VAE optimization involves minimizing the following objective, where the relative weighting of λ_1 and λ_{KL} is determined by the (fixed) variance of the Laplacian distribution,

$$G^*, E^* = \arg \min_{G, E} \lambda_1 \mathcal{L}_1(G, E) + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}(E). \quad (3)$$

Generative adversarial networks. Without overcoming the problem of modeling pixel covariances, it is likely not possible to produce sharp and clean predictions. Indeed, as shown in our experiments, the pure VAE model tends to produce blurry futures. We can force the predictions to stay on the video manifold by matching the distributions of predicted and real videos. Given a classifier D that is capable of distinguishing generated videos $\hat{\mathbf{x}}_{1:T}$ from real videos $\mathbf{x}_{1:T}$, the generator can be trained to match the statistics of the real data distribution using the binary cross-entropy loss,

$$\mathcal{L}_{\text{GAN}}(G, D) = \mathbb{E}_{\mathbf{x}_{0:T}} [\log D(\mathbf{x}_{0:T-1})] + \mathbb{E}_{\mathbf{x}_{1:T}, \mathbf{z}_t \sim p(\mathbf{z}_t)|_{t=0}^{T-1}} [\log (1 - D(G(\mathbf{x}_0, \mathbf{z}_{0:T-1})))]. \quad (4)$$

The classifier, which is not known a priori and is problem-specific, can be realized as a deep discriminator network that can be adversarially learned,

$$G^* = \arg \min_G \max_D \mathcal{L}_{\text{GAN}}(G, D). \quad (5)$$

This is the setting of generative adversarial networks (GANs) [6]. In the conditional case, a per-pixel reconstruction term $\mathcal{L}_1^{\text{GAN}}$ is added to the objective, which is analogous to \mathcal{L}_1 , except that the latent codes are sampled from the prior.

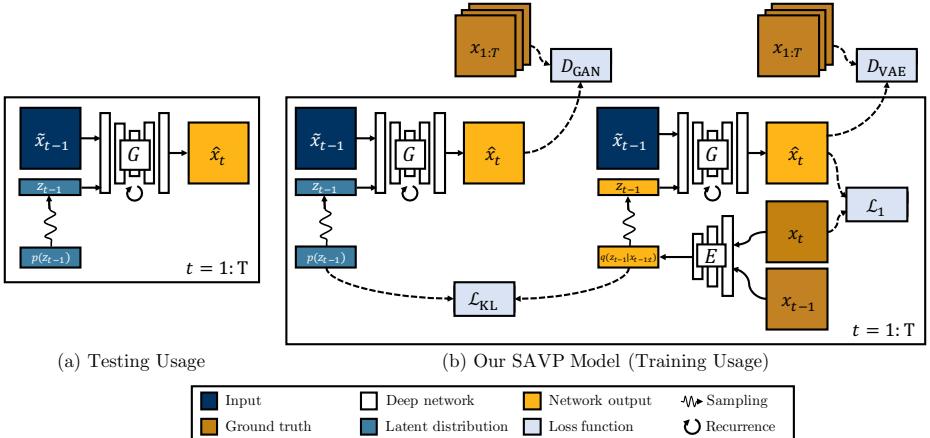


Fig. 2: **Our proposed video prediction model.** (a) During testing, we synthesize new frames by sampling random latent codes \mathbf{z} from a prior distribution $p(\mathbf{z})$ independently at each time step. The generator G takes a previous frame and a latent code to synthesize a new frame. Synthesized frames are fed back into the generator at the next time step. The previous frame is denoted as $\tilde{\mathbf{x}}_{t-1}$ to indicate that it could be a ground truth frame \mathbf{x}_{t-1} (for the initial frames) or the last prediction $\hat{\mathbf{x}}_{t-1}$. Information is remembered within the generator with the use of recurrent LSTM layers [47]. (b) During training, the generator is optimized to predict videos that match the distribution of real videos, using a learned discriminator, as proposed in Generative Adversarial Networks (GANs) [6]. The discriminators operate on the generated videos. We sample latent codes from two distributions: (1) the prior distribution, and (2) a posterior distribution approximated by a learned encoder E , as proposed in variational autoencoders (VAEs) [5]. For the latter, the regression \mathcal{L}_1 loss is used. Separate discriminators D and D_{VAE} are used depending on the distribution used to sample the latent code.

Stochastic Adversarial Video Prediction. The VAE and GAN models provide complementary strengths. GANs use a learned loss function through the discriminator, which learns the statistics of natural videos. However, GANs can suffer from the problem of mode collapse, especially in the conditional setting [9,8,10]. VAEs explicitly encourage the latent code to be more expressive and meaningful, since the learned encoder produces codes that are useful for making accurate predictions at training time. However, during training, VAEs only observe latent codes that are encodings of ground truth images, and never train on completely randomly drawn latent codes, leading to a potential train and test mismatch. GANs, however, are trained with randomly drawn codes.

Our stochastic adversarial video prediction (SAVP) model combines both approaches, shown in Fig. 2. Another term \mathcal{L}_{GAN}^{VAE} is introduced, which is analogous to \mathcal{L}_{GAN} except that it uses latent codes sampled from $q(\mathbf{z}_t|\mathbf{x}_{t:t+1})$ and a separate video discriminator D_{VAE} . The objective of our SAVP model is

$$G^*, E^* = \arg \min_{G, E} \max_{D, D_{VAE}} \lambda_1 \mathcal{L}_1(G, E) + \lambda_{KL} \mathcal{L}_{KL}(E) + \mathcal{L}_{GAN}(G, D) + \mathcal{L}_{GAN}^{VAE}(G, E, D_{VAE}). \quad (6)$$

Our model is inspired by VAE-GANs [18] and BicycleGAN [10], which have shown promising results in the conditional image generation domain, but have not been applied to video prediction. The video prediction setting presents two important challenges. First, conditional image generation can handle large appearance changes between the input and output, but suffer when attempting to produce large spatial changes. The video prediction setting is precisely the opposite—the appearance remains largely the same from frame to frame, but the most important changes are spatial. Secondly, video prediction involves sequential prediction. To help solve this, modifications can be made to the architecture, such as adding recurrent layers within the generator and using a video-based discriminator to model dynamics. However, errors in the generated images can easily accumulate, as they are repeatedly fed back into the generator in an autoregressive approach. Our approach is the first to use VAE-GANs in a recurrent setting for stochastic video prediction.

Networks. The generator is a convolutional LSTM [27,47] that predicts pixel-space transformations between the current and next frame, with additional skip connections with the first frame as done in SNA [4]. At every time step, the network is conditioned on the current frame and latent code. After the initial frames, the network is conditioned on its own predictions. The conditioning on the latent codes is realized by concatenating them along the channel dimension to the inputs of all the convolutional layers of the convolutional LSTM. The encoder is a feed-forward convolutional network that encodes pairs of images at every time step. The video discriminator is a feed-forward convolutional network with 3D filters, based on SNGAN [48] but with the filters “inflated” from 2D to 3D. See Appendix A for additional details on the network architectures and training procedure.

4 Experiments

Our experimental evaluation studies the realism, diversity, and accuracy of the videos generated by our approach and prior methods, evaluate the importance of various design decisions, including the form of the reconstruction loss and the presence of the variational and adversarial objectives. Evaluating the performance of stochastic video prediction models is exceedingly challenging: not only should the samples from the model be physically realistic and visually plausible given the context frames, but the model should also be able to produce diverse samples that match the conditional distribution in the data. These properties are difficult to evaluate precisely: realism is not accurately reflected with simple metrics of reconstruction accuracy, and the true conditional distribution in the data is unknown, since real-world datasets only have a single future for each initial sequence. Below, we discuss the metrics that we use to evaluate realism, diversity, and accuracy. No single metric alone provides a clear answer as to which model is better, but considering multiple metrics can provide us with a more complete understanding of the performance and trade-offs of each approach.

4.1 Evaluation Metrics

Realism: comparisons to real videos using human judges. The realism of the predicted videos is evaluated based on a *real vs. fake* two-alternative forced choice (2AFC) test. Human judges on Amazon Mechanical Turk (AMT) are presented with a pair of videos—one generated and one real—and asked to identify the generated, or “fake” video. We use the implementation from [49], modified for videos. Each video is 10 frames long and shown over 2.5 seconds. For each method, we gather 1000 judgments from 25 human judges. Each human evaluator is provided with 10 training trials followed by 40 test trials. A method that produces perfectly realistic videos would achieve a fooling rate of 50%.

Diversity: distance between samples. Realism is not the only factor in determining the performance of a video prediction model: aside from generating predictions that look physically plausible and realistic, a successful model must also be able to adequately cover the range of possible futures in an uncertain environment. We compute diversity as the average distance between randomly sampled video predictions, similar to [10]. Distance is measured in the VGG feature space (pretrained on ImageNet classification), averaged across five layers, which has been shown to correlate well with human perception [23,24,21].

Accuracy: similarity of the best sample. One weakness of the above metric is that the samples may be diverse but still not cover the feasible output space. Though we do not have the true output distribution, we can still leverage the single ground truth instance. This can be done by sampling the model a finite number of times, and evaluating the similarity between the best sample and the ground truth. This has been explored in prior work on stochastic video prediction [1,2], using PSNR or SSIM as the evaluation metric. In addition to these, we use cosine similarity in the pretrained VGG feature space.

4.2 Datasets

We evaluate on two real-world datasets: the BAIR action-free robot pushing dataset [4] and the KTH human actions dataset [50]. See Appendix B.3 for additional results on the action-conditioned version of the robot pushing dataset.

BAIR action-free. This dataset consists of a randomly moving robotic arm that pushes objects on a table [4]. This dataset is particularly challenging since (a) it contains large amounts of stochasticity due to random arm motion, and (b) it is a real-world application, with a diverse set of objects and large cluttered scene (rather than a single frame-centered object with a neutral background). The videos have a spatial resolution of 64×64 . We condition on the first 2 frames and train to predict the next 10 frames. We predict 10 future frames for the 2AFC experiments and 28 future frames for the other experiments.

KTH. This dataset consists of a human subject doing one of six activities: walking, jogging, running, boxing, hand waving, and hand clapping [50]. For the first three activities, the human enters and leaves the frame multiple times, leaving the frame empty with a mostly static background for multiple frames

at a time. The sequences are particularly stochastic when the initial frames are all empty since the human can enter the frame at any point in the future. As a preprocessing step, we center-crop each frame to a 120×120 square and then resize to a spatial resolution of 64×64 . We condition on the first 10 frames and train to predict the next 10 frames. We predict 10 future frames for the 2AFC experiments and 30 future frames for the other experiments. For each sequence, subclips of the desired length are randomly sampled at training and test time.

4.3 Methods: Ablations and Comparisons

We compare the following variants of our method, in our effort to evaluate the effect of each loss term. Code and models are available at our website¹.

Ours, SAVP. Our full stochastic adversarial video prediction model, with the VAE and GAN objectives shown in Fig. 2 and in Eq. (6).

Ours, GAN-only. An ablation of our model with only a conditional GAN, without the variational autoencoder. This model still takes a noise sample as input, but the noise is sampled from the prior during training. This model is broadly representative of prior stochastic GAN-based methods [14,15,16,17].

Ours, VAE-only. An ablation of our model with only a conditional VAE, with the reconstruction \mathcal{L}_1 loss but without the adversarial loss. This model is broadly representative of prior stochastic VAE-based methods [1,2].

We also compare to the following prior and concurrent stochastic VAE-based methods, both of which use the reconstruction \mathcal{L}_2 loss and no adversarial loss.

Stochastic Variational Video Prediction (SV2P). A VAE-based method proposed by Babaeizadeh et al. [1]. To our knowledge, this is the most recent prior VAE-based method, except for the concurrent method [2] discussed below.

Stochastic Video Generation with a Learned Prior (SVG-LP). This is a concurrent VAE-based method proposed by Denton et al. [2], with learned priors for the latent codes, that shows results that outperform Babaeizadeh et al. [1].

See Appendix B.2 for qualitative results on representative prior work of an unconditioned GAN-based method, MoCoGAN [17].

4.4 Experimental Results

We show qualitative results on the BAIR and KTH datasets in Fig. 3 and Fig. 4, respectively. For the quantitative results, we evaluate the realism, diversity, and accuracy of the predicted videos.

Does our method produce *realistic* results? In Fig. 5, variants of our method are compared to prior work. On the BAIR action-free dataset, our GAN variant achieves the highest fooling rate, whereas our proposed SAVP model, a VAE-GAN-based method, achieves a fooling rate that is roughly halfway between the GAN and VAE models alone. The SV2P method [1] does not achieve realistic results. The concurrent VAE-based SVG-LP method [2] achieves high realism, similar to our VAE variant, but substantially below our GAN-based

¹ https://alexlee-gk.github.io/video_prediction

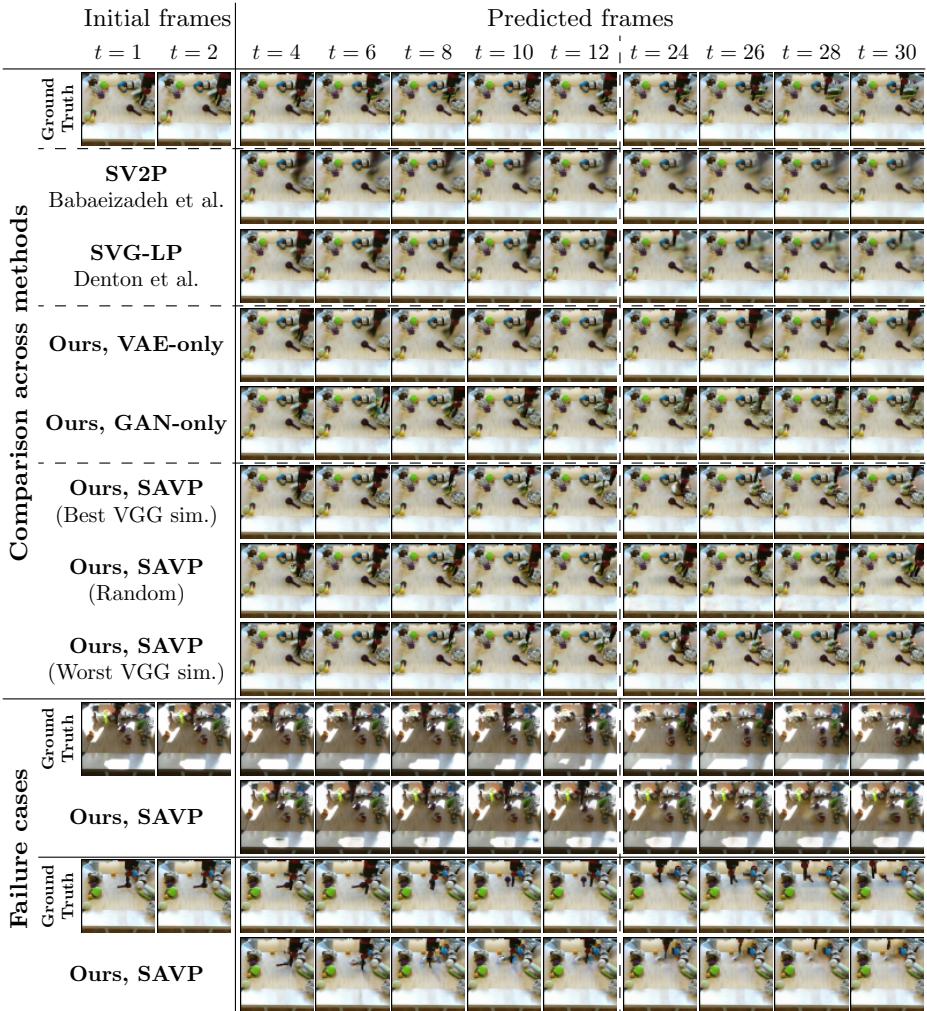


Fig. 3: **Qualitative Results (BAIR action-free dataset).** (top) We show qualitative comparisons between our variants and previous and concurrent work. The models are conditioned on 2 frames. Unless otherwise labeled, we show the closest generated sample to ground truth using VGG cosine similarity. The prior SV2P method [1] immediately produces blurry results. Our GAN and VAE-based variants, as well as the concurrent SVG-LP method [2] produce sharper results. Note, however, that SVG-LP still blurs out the jar on the right side of the image when it is touched by the robot, while our GAN-based models keep the jar sharp. We show three results for our SAVP model—using the closest, furthest, and random samples. There is large variation between the three samples in the arm motion, and even the furthest sample from the ground truth looks realistic. (bottom) We show failure cases—the arm disappears on the top example, and in the bottom, the object smears away in the later time steps. Additional videos of the test set can be found at: https://alexlee-gk.github.io/video_prediction.

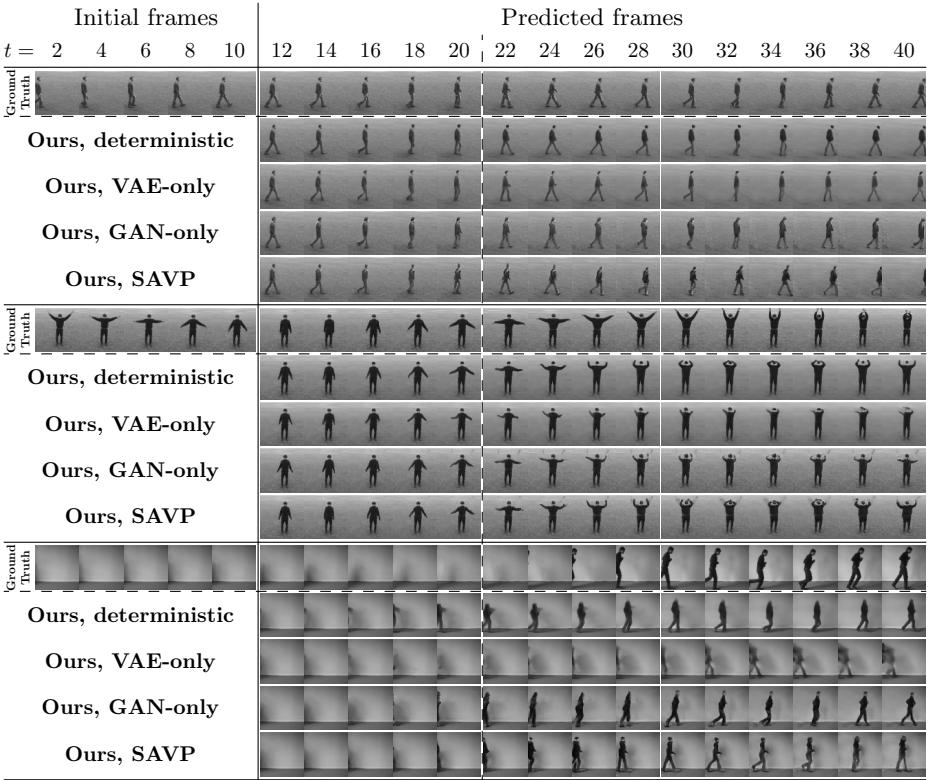


Fig. 4: **Qualitative Results (KTH dataset).** We show qualitative comparisons between our model and ablations of our model. The models are conditioned on 10 frames and trained to predict 10 future frames (indicated by the vertical dashed line), but are being tested to predict 30 frames. For the non-deterministic models, we show the closest generated sample to ground truth using VGG cosine similarity. We found that the samples of all the methods are less diverse in this dataset, so we only show one sample per sequence. We hypothesize that this dataset has much less stochasticity, and even our deterministic model is able to generate reasonable predictions. The three sequences shown correspond to the human actions of walking, hand waving, and running. In the first one, both the deterministic and the VAE models generate images that are slightly blurry, but they do not degrade over time. Our GAN-based methods produce sharper predictions. In the middle sequence, the VAE model generates images where the small limbs disappear longer into the future, whereas our SAVP method preserves them. In the last sequence, all the conditioning frames are empty except for a shadow on the left. All our variants are able to use this cue to predict that a person is coming from the left, although our SAVP model generates the most realistic sequence. Additional videos of the test set can be found at: https://alexlee-gk.github.io/video_prediction.

variants. On the KTH dataset, our SAVP model achieves the highest realism score, substantially above our GAN variant. Among the VAE-based methods without adversarial losses, our VAE-only model outperforms prior SV2P [1] and concurrent SVG-FP [2] methods in terms of realism.

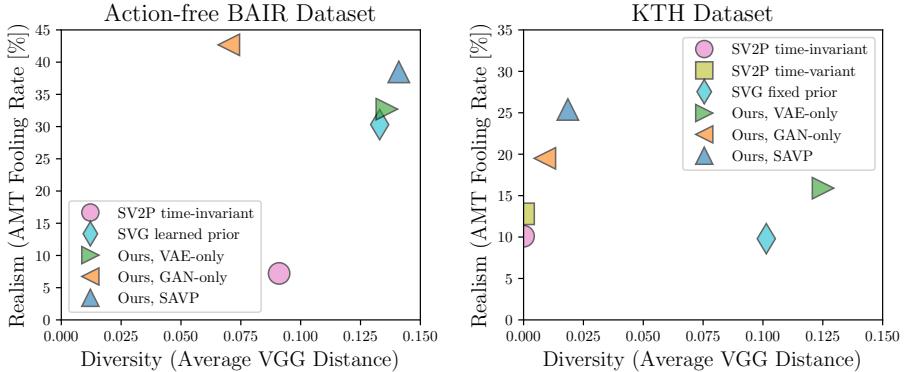


Fig. 5: **Realism vs Diversity.** We measure realism using a real vs fake Amazon Mechanical Turk (AMT) test, and diversity using average VGG cosine distance. Higher is better on both metrics. **Our VAE** variant achieves higher realism and diversity than prior **SV2P** [1] and concurrent **SVG** [2] methods, both of them based on VAEs. **Our GAN** variant achieves higher realism than the pure VAE methods, at the expense of significantly lower diversity. **Our SAVP** model, based on VAE-GANs, improves along the realism axis compared to a pure VAE method, and improves along the diversity axis compared to a pure GAN method. Although the SV2P methods mode-collapse on the KTH dataset, we note that they did not evaluate on this dataset, and their method could benefit from hyperparameters that are better suited for this dataset.

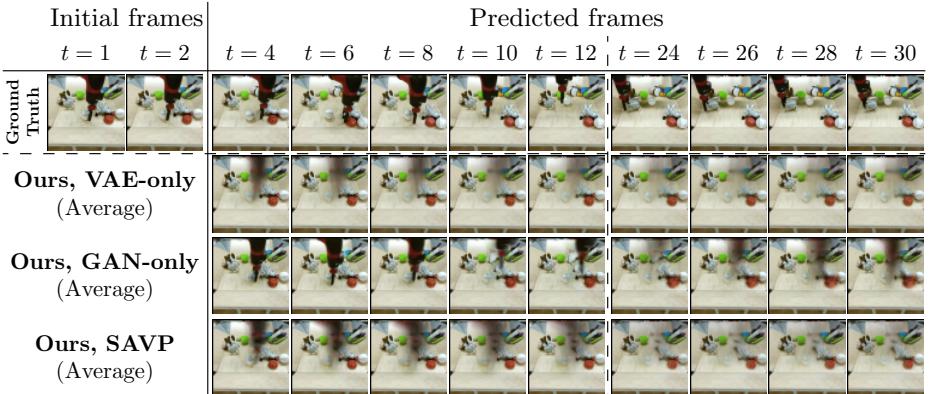


Fig. 6: **Qualitative visualization of diversity.** We show predictions of our models, averaged over 100 samples. A model that produces diverse outputs should predict that the robot arm moves in random directions at each time step, and thus the arm should “disappear” over time in these averaged predictions. Consistent with our quantitative evaluation of diversity, we see that both our SAVP model and our VAE variant produce diverse samples, whereas the GAN-only method is prone to mode-collapse.

Does our method generate *diverse* results? We measure diversity by taking the distance between random samples. Diversity results are also shown in Fig. 5 and a qualitative visualization of diversity is shown in Fig. 6. While the

GAN-only approach achieves realistic results, it shows lower diversity than the VAE-based methods. This is an example of the commonly known phenomenon of mode-collapse, where multiple latent codes produce the same or similar images on the output [51]. Intuitively, the VAE-based methods explicitly encourage the latent code to be more expressive by using an encoder from the output space into the latent space during training. This is verified in our experiments, as the VAE-based variants, including our SAVP model, achieve higher diversity than our GAN-only models on both datasets. On the KTH dataset, our VAE variant and concurrent VAE-based SVG-FP method [2] both achieve significantly higher diversity than all the other methods. Although the VAE-based SV2P methods [1] mode-collapse on the KTH dataset, we note that they did not evaluate on this dataset, and as such, their method could benefit from different hyperparameters that are better suited for this dataset.

Does our method generate accurate results? Following recent work on VAE-based video prediction [1,2], we evaluate on full-reference metrics by sampling multiple predictions from the model. We draw 100 samples for each video, find the “best” sample by computing similarity to the ground truth video, and show the average similarity across the test set as a function of time. The results on the BAIR and KTH datasets are shown in Fig. 7 and Fig. 8, respectively. We test generalization ability by running the model for more time steps than it was trained for. Even though the model is only trained to predict 10 future frames, we observe graceful degradation over time.

While PSNR and SSIM [19] are commonly used for video prediction, these metrics are not necessarily indicative of prediction quality. In video prediction, structural ambiguities and geometric deformations are a dominant factor, and SSIM is not an appropriate metric in such situations [22,21]. This is particularly noticeable with the SV2P method, which achieves high PSNR and SSIM scores, but produces blurry and unrealistic images. Furthermore, we additionally trained our VAE and deterministic variants using the standard MSE loss \mathcal{L}_2 to understand the relationship between the form of the reconstruction loss and the metrics. The general trend is that models trained with \mathcal{L}_2 , which favors blurry predictions, are better on PSNR and SSIM, but models trained with \mathcal{L}_1 are better on VGG cosine similarity. See Appendix B.1 for quantitative results comparing models trained with \mathcal{L}_1 and \mathcal{L}_2 . In addition, we expect for our GAN-based variants to underperform on PSNR and SSIM since GANs prioritize matching joint distributions of pixels over per-pixel reconstruction accuracy.

To partially overcome the limitations of these metrics, we also evaluate using distances in a deep feature space [23,24], which have been shown to correspond better with human perceptual judgments [21]. We use cosine similarity between VGG features averaged across five layers. Notice that this VGG similarity is a held-out metric, meaning that it was not used as a loss function during training. Otherwise, a model trained for it would unfairly and artificially achieve better similarities by exploiting potential flaws on that metric. Our VAE variant, along with concurrent VAE-based SVG method [2], performs best on this metric. Our SAVP model achieves slightly lower VGG similarity, striking a desirable balance

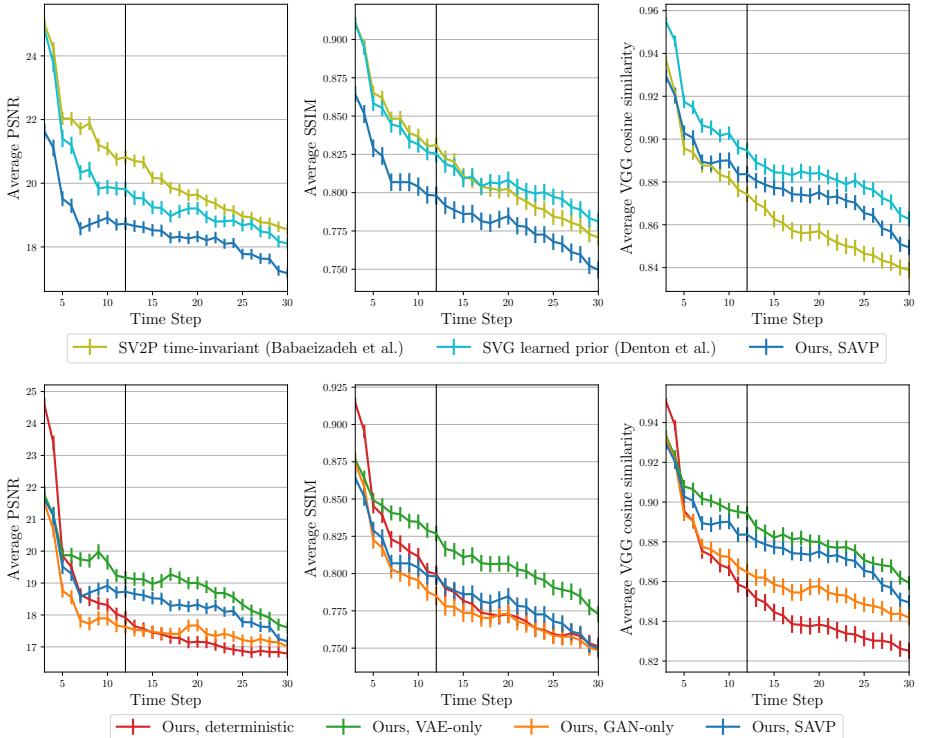


Fig. 7: Similarity of the best sample (BAIR action-free dataset). We show the similarity (higher is better) between the best sample as a function of prediction time step across different methods and evaluation metrics. Each test video is sampled 100 times for each method, and the similarity between the ground truth and predicted video is computed. We use three metrics—PSNR (left), SSIM [19] (middle), and VGG cosine similarity (right). While PSNR and SSIM are commonly used for video prediction, they correspond poorly to human judgments [20,21]. In fact, SSIM was not designed for problems where spatial ambiguities are a factor [22]. We include these metrics for completeness but note that they are not necessarily indicative of prediction quality. VGG similarity has been shown to better match human perceptual judgments [21]. (top) We compare to prior SV2P [1] and concurrent SVG-LP [2] methods. Although SV2P produces blurry and unrealistic images, it achieves the highest PSNR. Both our SAVP and concurrent SVG-LP methods outperform SV2P on VGG similarity. We expect our GAN-based variants to underperform on PSNR and SSIM since GANs prioritize matching joint distributions of pixels over per-pixel reconstruction accuracy. (bottom) We compare to ablated versions of our model. Our VAE variant achieves higher scores than our SAVP model, which in turn achieves significantly higher VGG similarities compared to our GAN-only model. Note that the models were only trained to predict 10 future frames (indicated by the vertical line), but is being tested on generalization to longer sequences.

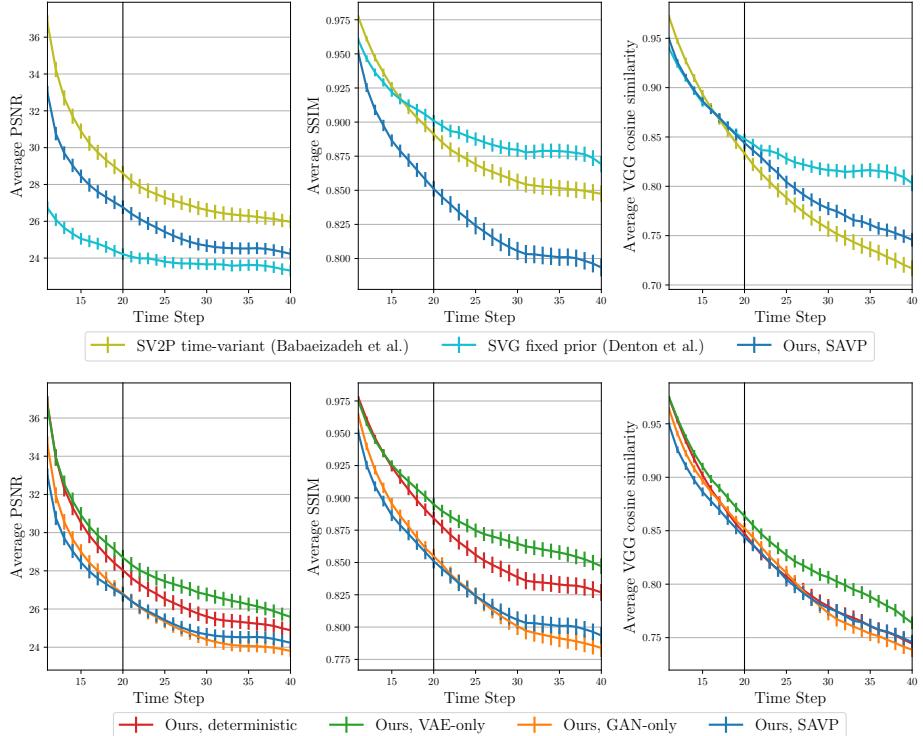


Fig. 8: Similarity of the best sample (KTH dataset). We show the similarity between the best predicted sample (out of a 100 samples) and the ground truth video. We evaluate on the standard video prediction metrics of PSNR and SSIM [19]. To partially mitigate the limitations of these metrics, we also evaluate on VGG cosine similarity, which is a full-reference metric that have been shown to correspond better to human perceptual judgments [21]. (top) We compare to prior SV2P [1] and concurrent SVG-FP [2] methods. As in the case of the robot dataset, SV2P achieves high PSNR values, even though it produces blurry and unrealistic images. Although all three methods achieve comparable VGG similarities for the first 10 future frames (which is what the models were trained for, and indicated by the vertical line), our SAVP model predicts videos that are substantially more realistic, as shown in our subjective human evaluation, thus achieving a desirable balance between realism and accuracy. (bottom) We compare to ablated versions of our model. Our VAE-only method outperforms all our other variants on the three metrics. In addition, our deterministic model is not that far behind in terms of similarity, leading us to believe that the KTH dataset is not as stochastic when conditioning on the past 10 frames.

in predicting videos that are realistic, diverse, and accurate. Among the VAE-based methods, prior SV2P method [1] achieves the lowest VGG similarity.

On stochastic environments, such as in the BAIR action-free dataset, there is some correlation between diversity and accuracy of the best sample: a model with diverse predictions is more likely to sample a video that is close to the

ground truth. This relation can be seen in Fig. 5 and Fig. 7 for the robot dataset, e.g. our SAVP model is both more diverse and achieves higher similarity than our GAN-only variant. This is not true on less stochastic environments. We hypothesize that the KTH dataset is not as stochastic when conditioning on 10 frames, as evidenced by the fact that the similarities of the deterministic and stochastic models are not that far apart from each other. This would explain why our GAN variant and SV2P achieve modest similarities despite achieving low diversity on the KTH dataset.

Does combining the VAE and GAN produce better predictions? The GAN alone achieves high realism but low diversity. The VAE alone achieves lower realism but provides increased diversity. Adding the GAN to the VAE model increases the realism without sacrificing diversity, at only a small or no cost in realism on stochastic datasets. This is consistent with [10], which showed that combining GAN and VAE-based models can provide benefits in the case of image generation. To our knowledge, our method is the first to extend this class of models to the stochastic video prediction setting, and the first to illustrate that this leads to improved realism with a degree of diversity comparable to the best VAE models in stochastic environments. The results show that this combination of losses is the best choice for realistic coverage of diverse stochastic futures.

5 Conclusion

In this work, we proposed a video prediction model that combines latent variables trained via a variational lower bound with an adversarial loss to produce a high degree of visual and physical realism. The VAE-style training with latent variables enables our method to make diverse stochastic predictions, and our experiments show that the adversarial loss is effective at producing predictions that are more visually realistic according to human raters. Evaluation of video prediction models is a major challenge, and we evaluate our method, as well as ablated variants that consist of only the VAE or only the GAN loss, in terms of a variety of quantitative and qualitative measures, including human ratings, diversity, and accuracy of the predicted samples. Our results demonstrate that our approach produces more realistic predictions than prior methods, while preserving the sample diversity of VAE-based methods.

Acknowledgments. We thank Emily Denton for providing pre-trained models and extensive and timely assistance with reproducing the SVG results, and Mohammad Babaeizadeh for providing data for comparisons with SV2P. This research was supported in part by the Army Research Office through the MAST program, the National Science Foundation through IIS-1651843 and IIS-1614653, and hardware donations from NVIDIA. Alex Lee and Chelsea Finn were also supported by the NSF GRFP. Richard Zhang was partially supported by the Adobe Research Fellowship.

References

1. Babaeizadeh, M., Finn, C., Erhan, D., Campbell, R.H., Levine, S.: Stochastic variational video prediction. In: International Conference on Learning Representations (ICLR). (2018)
2. Denton, E., Fergus, R.: Stochastic video generation with a learned prior. arXiv preprint arXiv:1802.07687 (2018)
3. Finn, C., Goodfellow, I., Levine, S.: Unsupervised learning for physical interaction through video prediction. In: Neural Information Processing Systems (NIPS). (2016)
4. Ebert, F., Finn, C., Lee, A., Levine, S.: Self-supervised visual planning with temporal skip connections. In: Conference on Robot Learning (CoRL). (2017)
5. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (ICLR). (2014)
6. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Neural Information Processing Systems (NIPS). (2014)
7. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. In: International Conference on Learning Representations (ICLR). (2016)
8. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Conference on Vision and Pattern Recognition (CVPR). (2017)
9. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.: Context encoders: Feature learning by inpainting. In: Conference on Vision and Pattern Recognition (CVPR). (2016)
10. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: Neural Information Processing Systems (NIPS). (2017)
11. Vondrick, C., Torralba, A.: Generating the future with adversarial transformers. In: Conference on Vision and Pattern Recognition (CVPR). (2017)
12. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. In: International Conference on Learning Representations (ICLR). (2017)
13. Lu, C., Hirsch, M., Schölkopf, B.: Flexible spatio-temporal networks for video prediction. In: Computer Vision and Pattern Recognition (CVPR). (2017)
14. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: Neural Information Processing Systems (NIPS). (2016)
15. Saito, M., Matsumoto, E., Saito, S.: Temporal generative adversarial nets with singular value clipping. In: International Conference on Computer Vision (ICCV). (2017)
16. Chen, B., Wang, W., Wang, J., Chen, X., Li, W.: Video imagination from a single image with transformation generation. arXiv preprint arXiv:1706.04124 (2017)
17. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: MoCoGAN: Decomposing motion and content for video generation. In: Computer Vision and Pattern Recognition (CVPR). (2018)
18. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. In: International Conference on Machine Learning (ICML). (2016)

19. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *Transactions on Image Processing (TIP)* (2004)
20. Ponomarenko, N., Jin, L., Ieremeiev, O., Lukin, V., Egiazarian, K., Astola, J., Vozel, B., Chehdi, K., Carli, M., Battisti, F., et al.: Image database TID2013: Peculiarities, results and perspectives. *Signal Processing: Image Communication* (2015)
21. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep networks as a perceptual metric. In: *Computer Vision and Pattern Recognition (CVPR)*. (2018)
22. Sampat, M.P., Wang, Z., Gupta, S., Bovik, A.C., Markey, M.K.: Complex wavelet structural similarity: A new image similarity index. *Transactions on Image Processing (TIP)* (2009)
23. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: *Neural Information Processing Systems (NIPS)*. (2016)
24. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: *European Conference on Computer Vision (ECCV)*. (2016)
25. Ranzato, M., Szlam, A., Bruna, J., Mathieu, M., Collobert, R., Chopra, S.: Video (language) modeling: a baseline for generative models of natural videos. *arXiv preprint arXiv:1412.6604* (2014)
26. Oh, J., Guo, X., Lee, H., Lewis, R.L., Singh, S.: Action-conditional video prediction using deep networks in atari games. In: *Neural Information Processing Systems (NIPS)*. (2015)
27. Xingjian, S., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.c.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: *Neural Information Processing Systems (NIPS)*. (2015)
28. De Brabandere, B., Jia, X., Tuytelaars, T., Van Gool, L.: Dynamic filter networks. In: *Neural Information Processing Systems (NIPS)*. (2016)
29. Xue, T., Wu, J., Bouman, K., Freeman, B.: Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In: *Neural Information Processing Systems (NIPS)*. (2016)
30. Byravan, A., Fox, D.: SE3-nets: Learning rigid body motion using deep neural networks. *arXiv preprint arXiv:1606.02378* (2016)
31. van Amersfoort, J., Kannan, A., Ranzato, M., Szlam, A., Tran, D., Chintala, S.: Transformation-based models of video sequences. *arXiv preprint arXiv:1701.08435* (2017)
32. Liu, Z., Yeh, R., Tang, X., Liu, Y., Agarwala, A.: Video frame synthesis using deep voxel flow. In: *International Conference on Computer Vision (ICCV)*. (2017)
33. Walker, J., Gupta, A., Hebert, M.: Dense optical flow prediction from a static image. In: *International Conference on Computer Vision (ICCV)*. (2015)
34. Walker, J., Doersch, C., Gupta, A., Hebert, M.: An uncertain future: Forecasting from static images using variational autoencoders. In: *European Conference on Computer Vision (ECCV)*. (2016)
35. Liang, X., Lee, L., Dai, W., Xing, E.P.: Dual motion GAN for future-flow embedded video prediction. In: *International Conference on Computer Vision (ICCV)*. (2017)
36. Chiappa, S., Racanière, S., Wierstra, D., Mohamed, S.: Recurrent environment simulators. In: *International Conference on Learning Representations (ICLR)*. (2017)
37. van den Oord, A., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. In: *International Conference on Machine Learning (ICML)*. (2016)

38. Kalchbrenner, N., van den Oord, A., Simonyan, K., Danihelka, I., Vinyals, O., Graves, A., Kavukcuoglu, K.: Video pixel networks. In: International Conference on Machine Learning (ICML). (2017)
39. Reed, S.E., van den Oord, A., Kalchbrenner, N., Colmenarejo, S.G., Wang, Z., Chen, Y., Belov, D., de Freitas, N.: Parallel multiscale autoregressive density estimation. In: International Conference on Machine Learning (ICML). (2017)
40. Zhou, Y., Berg, T.L.: Learning temporal transformations from time-lapse videos. In: European Conference on Computer Vision (ECCV). (2016)
41. Bhattacharjee, P., Das, S.: Temporal coherency based criteria for predicting video frames using deep multi-stage generative adversarial networks. In: Neural Information Processing Systems (NIPS). (2017)
42. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using LSTMs. In: International Conference on Machine Learning (ICML). (2015)
43. Lotter, W., Kreiman, G., Cox, D.: Deep predictive coding networks for video prediction and unsupervised learning. In: International Conference on Learning Representations (ICLR). (2017)
44. Denton, E., Birodkar, V.: Unsupervised learning of disentangled representations from video. In: Neural Information Processing Systems (NIPS). (2017)
45. Finn, C., Levine, S.: Deep visual foresight for planning robot motion. In: International Conference on Robotics and Automation (ICRA). (2017)
46. Byravan, A., Leeb, F., Meier, F., Fox, D.: SE3-pose-nets: Structured deep dynamics models for visuomotor planning and control. In: International Conference on Robotics and Automation (ICRA). (2018)
47. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation (1997)
48. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. In: International Conference on Learning Representations (ICLR). (2018)
49. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European Conference on Computer Vision (ECCV). (2016)
50. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: International Conference on Pattern Recognition (ICPR). (2004)
51. Goodfellow, I.: NIPS 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160 (2016)
52. Ulyanov, D., Vedaldi, A., Lempitsky, V.S.: Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022 (2016)
53. Odena, A., Dumoulin, V., Olah, C.: Deconvolution and checkerboard artifacts. Distill (2016)
54. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Computer Vision and Pattern Recognition (CVPR). (2017)
55. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. In: International Conference on Computer Vision (ICCV). (2017)
56. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR). (2015)
57. Bowman, S., Vilnis, L., Vinyals, O., Dai, A.M., Jozefowicz, R., Bengio, S.: Generating sentences from a continuous space. In: Conference on Computational Natural Language Learning (CoNLL). (2016)

A Networks and Training Details

A.1 Network Details

Generator. Our generator network, shown in Fig. 9, is inspired by the convolutional dynamic neural advection (CDNA) model proposed by [3]. The video prediction setting is a sequential prediction problem, so we use a convolutional LSTM [47,27] to predict future frames. We initialize the prediction on the initial sequence of ground truth frames (2 or 10 frames for the BAIR and KTH datasets, respectively), and predict 10 future frames. The model predicts a sequence of future frames by repeatedly making next-frame predictions and feeding those predictions back to itself. For each one-step prediction, the network has the freedom to choose to copy pixels from the previous frame, used transformed versions of the previous frame, or to synthesize pixels from scratch. The transformed versions of the frame are produced by convolving in the input image with predicted convolutional kernels, allowing for different shifted versions of the input. In more recent work, the first frame of the sequence is also given to the compositing layer [4].

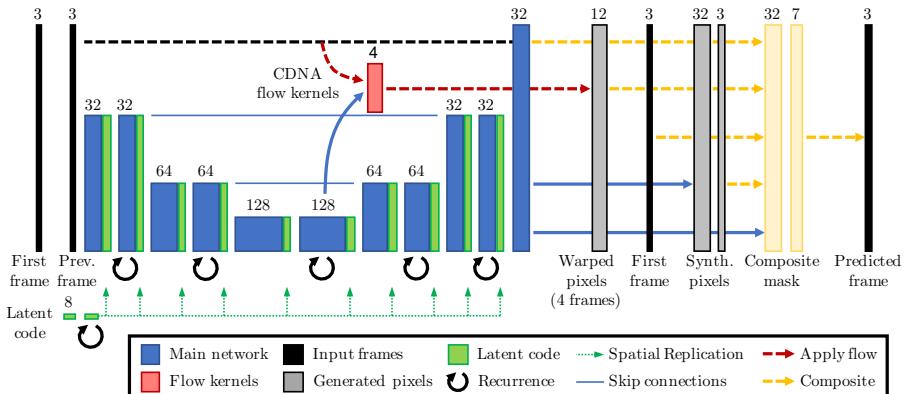


Fig. 9: **Architecture of our generator network.** Our network uses a convolutional LSTM [47,27] with skip-connection between internal layers. As proposed by [3], the network predicts (1) a set of convolution kernels to produce a set of transformed input images (2) synthesized pixels at the input resolution and (3) a compositing mask. Using the mask, the network can choose how to composite together the set of warped pixels, the first frame, previous frame, and synthesized pixels. One of the internal feature maps is given to a fully-connected layer to compute the kernels that specify pixel flow. The output of the main network is passed to two separate heads, each with two convolutional layers, to predict the synthesized frame and the composite mask. These two outputs use sigmoid and softmax non-linearities, respectively, to ensure proper normalization. We enable stochastic sampling of the model by conditioning the generator network on latent codes. These are first passed through a fully-connected LSTM, and then given to all the convolutional layers of the the convolutional LSTM.

To enable stochastic sampling, the generator is also conditioned on time-varying latent codes, which are sampled at training and test time. Each latent code \mathbf{z}_t is an 8-dimensional vector. At each prediction step, the latent code is passed through a fully-connected LSTM to facilitate correlations in time of the latent variables. The encoded latent code is then passed to all the convolutional layers of the main network, by concatenating it along the channel dimension to the inputs of these layers. Since they are vectors with no spatial dimensions, they are replicated spatially to match the spatial dimensions of the inputs.

We made a variety of architectural improvements to the original CDNA [3] and SNA [4] models, which overall produced better results on the per-pixel loss and similarity metrics. See Fig. 13 for a quantitative comparison of our deterministic variant (without VAE or GAN losses) to the SNA model on the BAIR action-conditioned dataset. Each convolutional layer is followed by instance normalization [52] and ReLU activations. We also use instance normalization on the LSTM pre-activations (i.e., the input, forget, and output gates, as well as the transformed and next cell of the LSTM). In addition, we modify the spatial downsampling and upsampling mechanisms. Standard subsampling and upsampling between convolutions is known to produce artifacts for dense image generation tasks [53,54,55]. In the encoding layers, we reduce the spatial resolution of the feature maps by average pooling, and in the decoding layers, we increase the resolution by using bilinear interpolation. All convolutions in the generator use a stride of 1. In the case of the action-conditioned dataset, actions are concatenated to the inputs of all the convolutional layers of the main network, as opposed to only the bottleneck.

Encoder. The encoder is a standard convolutional network that, at every time step, encodes a pair of images \mathbf{x}_t and \mathbf{x}_{t+1} into $\mu_{\mathbf{z}_t}$ and $\log \sigma_{\mathbf{z}_t}$. The same encoder network with shared weights is used at every time step. The encoder architecture consists of three convolutional layers, followed by average pooling of all the spatial dimensions. Two separate fully-connected layers are then used to estimate $\mu_{\mathbf{z}_t}$ and $\log \sigma_{\mathbf{z}_t}$, respectively. The convolutional layers use instance normalization, leaky ReLU non-linearities, and stride 2. This encoder architecture is the same one used in BicycleGAN [10] except that the inputs are pair of images, concatenated along the channel dimension.

Discriminator. The discriminator is a 3D convolutional neural network that takes in all the images of the video at once. We use spectral normalization and the SNGAN discriminator architecture [48], except that we “inflate” the convolution filters from 2D to 3D. The two video discriminators, D and D_{VAE} , share the same architecture, but not the weights, as done in BicycleGAN [10].

A.2 Training Details

Our generator network uses scheduled sampling during training as in Finn et al. [3], such that at the beginning the model is trained for one-step predictions, while by the end of training the model is fully autoregressive. We trained all

models with Adam [56] for 300000 iterations, linearly decaying the learning rate to 0 for the last 100000 iterations. The same training schedule was used for all the models, except for SVG, which was trained by its author. Our GAN-based variants used an optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.999$, learning rate of 0.0002, and a batch size of 16. Our deterministic and VAE models (including SNA and SV2P from prior work) used an optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, learning rate of 0.001, and a batch size of 32.

We used $\lambda_1 = 100$ for our GAN-based variants, and $\lambda_1 = 1$ for all the other models. For our VAE-based variants, we linearly anneal the weight on the KL divergence term from 0 to the final value λ_{KL} during training, as proposed by Bowman et al. [57], from iterations 50000 to 100000. We used a relative weighting of $\lambda_{\text{KL}}/\lambda_1 = 0.001$ for the BAIR robot pushing datasets, and $\lambda_{\text{KL}}/\lambda_1 = 0.00001$ for the KTH dataset. This hyperparameter was empirically chosen by computing similarity metrics on the validation set.

B Additional Experiments

B.1 Comparison of Pixel-Wise \mathcal{L}_1 and \mathcal{L}_2 Losses

We train our deterministic and VAE variants with the \mathcal{L}_1 and \mathcal{L}_2 losses to compare the effects of these reconstruction losses on the full-reference metrics used in this work. The pixel-wise \mathcal{L}_1 loss assumes that pixels are generated according to a fully factorized Laplacian distribution, whereas the \mathcal{L}_2 loss corresponds to a fully factorized Gaussian distribution. See Fig. 10 and Fig. 13 for quantitative results on the action-free and action-conditioned BAIR datasets, respectively.

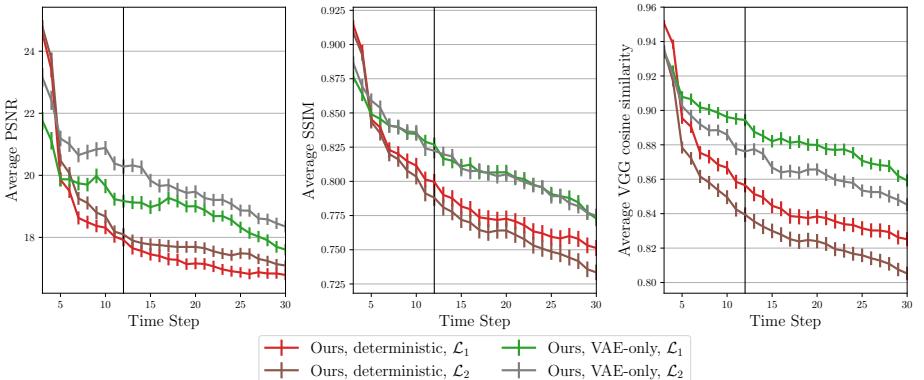


Fig. 10: **Similarity of the best sample (BAIR action-conditioned dataset).** We show the similarity between the predicted video and the ground truth, using the same evaluation as in Fig. 7. We compare our deterministic and VAE variants when trained with \mathcal{L}_1 and \mathcal{L}_2 losses, and observe that they have a significant impact on the quality of our predictions. The models trained with \mathcal{L}_1 produce videos that are qualitatively better and achieve higher VGG similarity than the equivalent models trained with \mathcal{L}_2 .

The general trend is that models trained with the \mathcal{L}_2 loss tend to generate blurry predictions but achieve higher PSNR scores than equivalent models trained with the \mathcal{L}_1 loss. This is because PSNR and \mathcal{L}_2 are closely related, the former being a logarithmic function of the latter. The opposite is true for the VGG cosine similarity metric, which has been shown to correspond better with human perceptual judgments [21]. Models trained with \mathcal{L}_1 significantly outperforms equivalent models trained with \mathcal{L}_2 . On the SSIM metric, models trained with \mathcal{L}_1 achieve roughly the same or better similarities than models trained with \mathcal{L}_2 . Although both losses are pixel-wise losses, the choice between \mathcal{L}_1 and \mathcal{L}_2 have a significant impact on the quality of our predicted videos.

B.2 Example Generations of an Unconditional GAN

We consider the motion and content decomposed GAN (MoCoGAN) model [17] as a representative unconditional GAN method from prior work, and use it to generate videos from the BAIR robot pushing dataset. We use their publicly available code and show qualitative results in Fig. 11. The results show the variant that uses patch-based discriminators, since this one achieved higher realism than the variant that uses image-based discriminators. Since this prior work demonstrates competitive results in comparison to other prior unconditional GAN methods, we chose it as the most representative recent example of purely GAN-based video generation for this comparison. MoCoGAN produces impressive results on various applications related to human action, which are focused on a single actor in the middle of the frame. However, it struggles on videos in the robot pushing domain where multiple entities are moving at a time, i.e. the robot arm and the objects it interacts with.

	Generated frames									
	$t = 1$	$t = 3$	$t = 5$	$t = 7$	$t = 9$	$t = 11$	$t = 13$	$t = 15$	$t = 17$	$t = 19$
MoCoGAN (Patch) Tulyakov et al.										

Fig. 11: **Example generations of MoCoGAN.** We use the unconditional version of MoCoGAN [17] to generate videos from the BAIR robot pushing dataset. We chose this model as a representative recent example of purely GAN-based unconditioned video generation. MoCoGAN produces impressive results on various applications related to human action, which are focused on a actor in the middle of the frame. However, this model struggles in the robot dataset where multiple entities are moving at a time. Note that since the patch-based discriminator has a limited receptive field of the image, the model can produce videos with two robot arms (last row) even though this is not in the dataset. We did not observe this behavior with their image-based discriminator.

B.3 Results on Action-Conditioned BAIR robot pushing dataset

We use the same dataset as the one in the main paper, except that we use the robot actions. Each action is a 4-dimensional vector corresponding to Cartesian translations and a value indicating if the gripper has been closed or opened. As in the action-free dataset, we condition on the first 2 frames of the sequence and train to predict the next 10 frames. In this dataset, the video prediction model is now also conditioned on a sequence of actions $\mathbf{a}_{0:T-1}$, in addition to the initial frames. The generator network is modified to take an action \mathbf{a}_t at each time step, by concatenating the action to the inputs of all the convolutional layers of the main network, similar to how the latent code \mathbf{z}_t is passed in (but without the additional fully-connected LSTM).

We show the realism and diversity results in Fig. 12 and the accuracy results in Fig. 13. In addition to the methods compared in the action-free dataset, we also compare to SNA [4], an action-conditioned deterministic video prediction model. The results indicate that our VAE model significantly outperforms prior methods on the full-reference metrics, and that our models significantly outperforms the model by Babaeizadeh et al. [1] both in terms of diversity of predictions and realism.

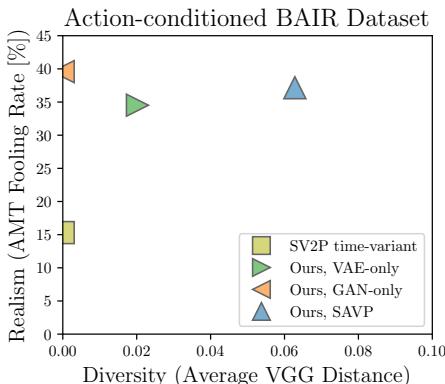


Fig. 12: **Realism vs Diversity (BAIR action-conditioned dataset).** The SV2P method [1] from prior work produces images with low realism, whereas our GAN, VAE, and SAVP models fool the human judges at a rate of around 35-40%. Our VAE-based models also produce videos with higher diversity, though lower diversity than other datasets, as this task involves much less stochasticity. The trend is the same as in the other datasets. Our SAVP model improves the realism of the predictions compared to our VAE-only model, and improves the diversity compared to our GAN-only model.

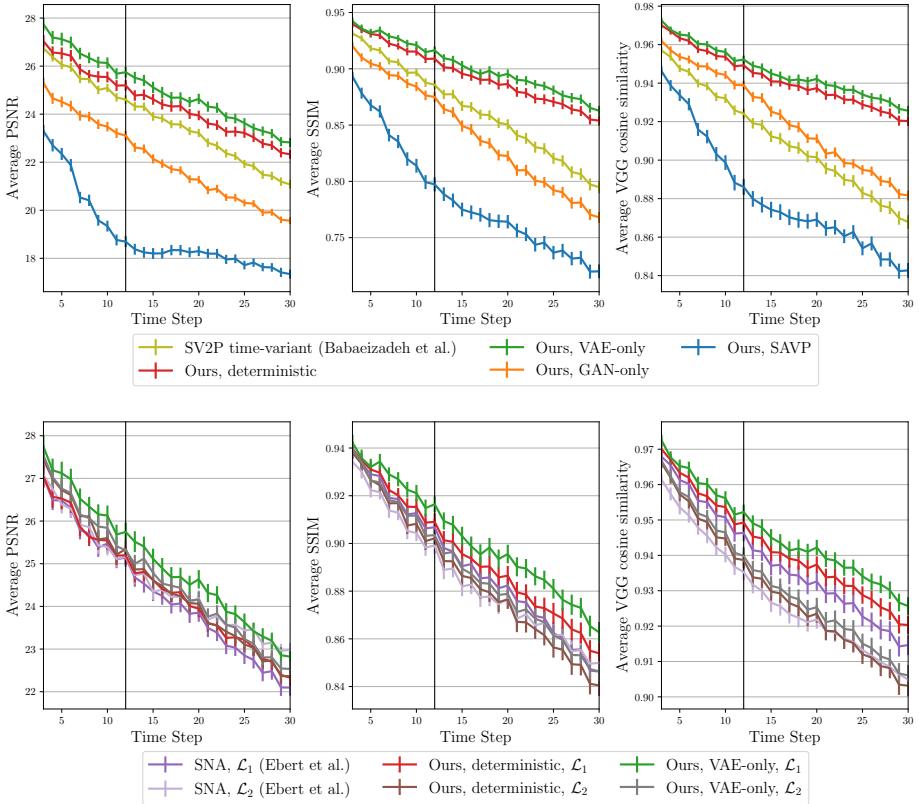


Fig. 13: Similarity of the best sample (BAIR action-conditioned dataset). We show the similarity between the predicted video and the ground truth, using the same evaluation as in Fig. 7, except that we condition on robot actions. (top) We compare to prior SV2P [1] and ours ablations. Our VAE and deterministic models both outperform SV2P, even though it is VAE-based. However, notice that the gap in performance between our VAE and deterministic models is small, as the dataset is less stochastic when conditioning on actions. Our SAVP model achieves much lower scores on all three metrics. We hypothesize that our SAVP model, as well as SV2P, is underutilizing the provided actions and thus achieving more stochasticity at the expense of accuracy. (bottom) We compare deterministic models—SNA [4] and ours—and our VAE model when trained with \mathcal{L}_1 and \mathcal{L}_2 losses. As in the action-free case, we observe that the choice of the pixel-wise reconstruction loss significantly affects prediction accuracy. Models trained with \mathcal{L}_1 are substantially better in SSIM and VGG cosine similarity compared to equivalent models trained with \mathcal{L}_2 . Surprisingly, the VAE model trained with \mathcal{L}_1 outperforms the other models even on the PSNR metric. We hypothesize that VAE models trained with \mathcal{L}_1 are better equipped to separate multiple modes of futures, whereas the ones trained with \mathcal{L}_2 might still average some of the modes. In fact, we evidenced this in preliminary experiments on the toy shapes dataset used by Babaeizadeh et al. [1]. Among the deterministic models, ours improves upon SNA [4], which is currently the best deterministic action-conditioned model on this dataset.