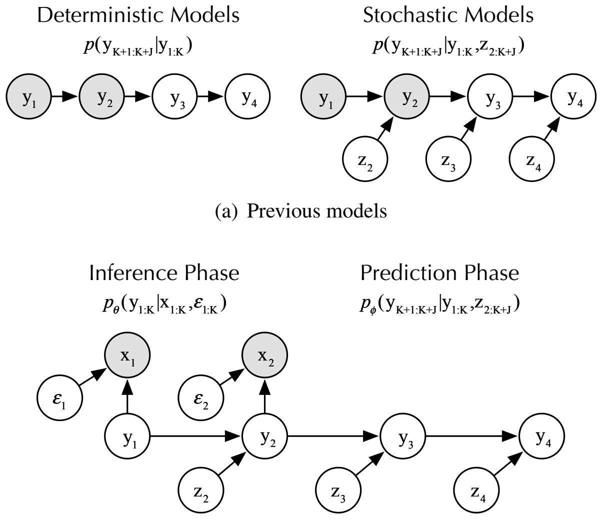
Probabilistic Video Prediction from Noisy Data with a Posterior Confidence

Yunbo Wang[1], Jiajun Wu[2], Mingsheng Long[1], Joshua B. Tenenbaum[3]
[1]Tsinghua University [2]Stanford University [3]Massachusetts Institute of Technology

Abstract

We study a new research problem of probabilistic future frames prediction from a sequence of noisy inputs, which is useful because it is difficult to guarantee the quality of input frames in practical spatiotemporal prediction applications. It is also challenging because it involves two levels of uncertainty: the perceptual uncertainty from noisy observations and the dynamics uncertainty in forward modeling.

*In this paper, we propose to tackle this problem with an end-to-end trainable model named Bayesian Predictive Network (**BP-Net**). Unlike previous work in stochastic video prediction that assumes spatiotemporal coherence and therefore fails to deal with perceptual uncertainty, BP-Net models both levels of uncertainty in an integrated framework. Furthermore, unlike previous work that can only provide unsorted estimations of future frames, BP-Net leverages a differentiable sequential importance sampling (SIS) approach to make future predictions based on the inference of underlying physical states, thereby providing sorted prediction candidates in accordance with the SIS importance weights, i.e., the confidences. Our experiment results demonstrate that BP-Net remarkably outperforms existing approaches on predicting future frames from noisy data.*



(a) Previous models

(b) Our model

Figure 1. A comparison of existing video prediction models and our model. Our model works under the Bayesian filtering framework to jointly consider both perceptual uncertainty ϵ_t and dynamics uncertainty z_t. K is the length of the input sequence. J is the length of the generated sequence. $K = J = 2$ in this much simplified example. Another advantage of our model is that it is able to estimate future frames with posteriori confidence scores.

1. Introduction

Learning to generate future video frames shows remarkable significance in real-world scenarios, such as precipitation forecasting [27, 35], traffic flows prediction [39, 37], and model predictive control in robotics [10, 8]. Existing models assume that training and testing videos are lossless representations of the underlying physical states; in practice, however, the quality of video data is often compromised. Precipitation forecasting depends on radar maps of the past hours, where there are stochastic measurement errors or accidental data noises, *e.g.*, caused by a passing airplane. In live streaming, video frames might be corrupted due to signal instability. Here, predicting future frames in advance could correct the content of upcoming videos.

Predicting future frames from noisy inputs is a new and challenging problem, because it involves uncertainty from two different sources: perceptual uncertainty, *i.e.*, the multi-modal mapping from noisy observations to underlying physical states, and dynamics uncertainty, *i.e.*, the multi-modal mapping from past to future. Solving the problem requires new approaches. Previous pixel-level future prediction models do not consider perceptual uncertainty and have a strong dependency on the temporal consistency and spatial coherence of videos. They thus do not work well for noisy spatiotemporal data videos, because the input-output temporal consistency is significantly broken.

In this paper, we introduce Bayesian Predictive Network (BP-Net) to jointly cope with perceptual uncertainty and dynamics uncertainty in an integrated framework. As shown in Figure 1, we implicitly decouple this problem into a Bayesian inference phase and a prediction phase with the sequential importance sampling (SIS) algorithm. We maintain a set of weighted samples (particles) over time and use them to approximate the belief distribution around the uncorrupted video frames. At each time stamp, we first exploit a prediction module to update the particle states, computing the prior probability of each particle. This module is significant for both inference and prediction phases. We then update the particle weights by measuring the likelihood of the newly received observation conditioned on the predicted particle state. This module is effective for Bayesian inference from noisy observation. It is the key component that differs BP-Net from existing video prediction work.

Our model integrates video denoising and video prediction into an end-to-end trainable paradigm. An alternative is to naively combine a denoising algorithm with standard video prediction algorithms. Compared with these two-step approaches that treat denoising and prediction separately, our integrated pipeline is relieved from the burden of precisely recovering uncorrupted input videos, which, empirically, leads to stronger results (Section 4.3). The second advantage of BP-Net lies on its ability to rank its outputs. Existing stochastic video prediction models generate future candidates without ranking them; it is unclear which of the many samples drawn from the model have better prediction quality. In contrast, BP-Net solves this problem via Bayesian filtering, using the SIS algorithm to approximate an importance weight to each future prediction candidate (particle). Generating future candidates with confidence scores allows the stochastic video prediction models to improve the downstream tasks. We validate the effectiveness of our proposed BP-Net on future prediction from noisy spatiotemporal data using two public video datasets. It remarkably outperforms previous video prediction models. Our experiment results also show that there are strong and positive correlations between particle weights and prediction qualities.

To sum up, this paper has two major contributions:

- This paper copes with the entangled perceptual and dynamics uncertainty in videos, and provides a pilot study of end-to-end video prediction from noisy data, which is a new problem in both the video modeling research community and real-world scenarios.
- BP-Net combines the merits of Bayesian inference and deep predictive learning. Unlike most SIS methods, BP-Net is suitable for large, complex observation spaces, such as the space of video frames. Further, unlike existing video prediction models, it provides estimations of future frames with posterior confidences that are consistent with the prediction qualities.

2. Related Work

Deterministic Video Prediction. Deep neural networks have been widely used in the deterministic video prediction. Ranzato *et al.* [26] defined a recurrent model predicting frames in a discrete space of patch clusters. Srivastava *et al.* [29] introduced the sequence-to-sequence LSTM network from language modeling to video prediction. But this model can only capture temporal variations. To learn spatial and temporal variations in a unified network structure, Shi *et al.* [27] integrated the convolution operator into recurrent state transition functions and proposed the Convolutional LSTM for a joint modeling of spatial and temporal variations. Some recent literature [21, 28, 9, 31, 24, 32, 34, 35, 16] further extended the convolutional recurrent model and investigated spatiotemporal future prediction in self-driving, weather forecasting, model predictive control, and human motion modeling. Different from these deterministic models, our model makes probabilistic future predictions.

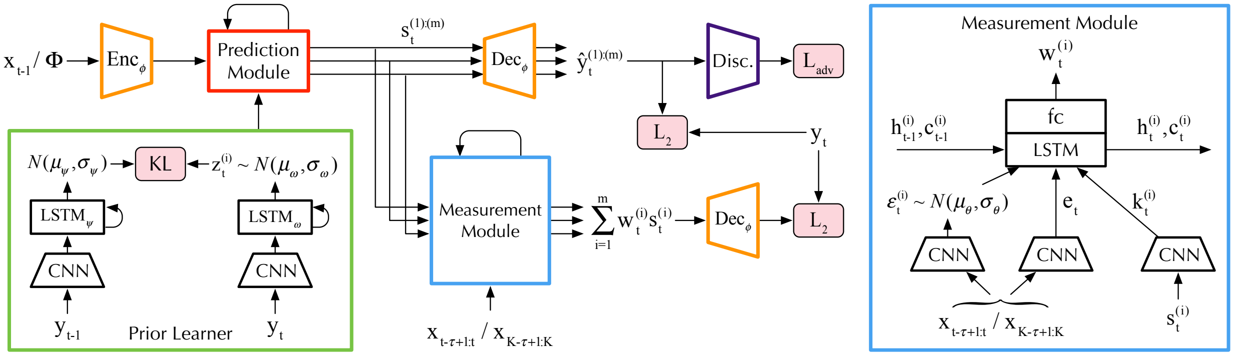
Stochastic Video Prediction. Adversarial learning [11, 6] has been increasingly used in video generation [22, 33, 7, 30, 36], as it aims to solve the multi-modal training difficulty of the future prediction and helps generate less blurry frames. In order to increase the diversity of future frames, variational auto-encoders [19] have also been introduced to stochastic video prediction models [38, 1, 5, 20, 14]. Variational methods also induce disentanglement [13, 3].

Our model differs from the above models in two perspectives. First, it considers both the perceptual and dynamics uncertainty, which brings new challenges to our work. Second, all above models generate future estimations from a prior distribution and cannot provide prediction results with confidence scores. Our model tackles this problem by integrating the differentiable particle filtering method with deep recurrent networks.

Differentiable Sequential Importance Sampling. Our work is also related to the differentiable sequential importance sampling (SIS) approaches. Gu *et al.* [12], Karkus *et al.* [17], and Jonschkowski *et al.* [15] independently discovered methods to make the conventional sequential importance sampling algorithms differentiable in terms of neural networks, showing that end-to-end training improves the performance of state estimations. Our work extends this idea from localization and tracking to video prediction, integrating differentiable SIS with predictive networks. Note that all these models learn simple transition models in a low dimensional state space, or even being trained with a known state transition function. In contrast, our method copes with more complex dynamics uncertainty for a longer future sequence in addition to the perceptual uncertainty.

3. Method

We design a model that combines the stochastic video prediction with a Bayesian inference algorithm to fit the proposed setting, where temporally changing signals have to be estimated online from noisy observations.



(a) The overall architecture of BP-Net at the training time

(b) The measurement module

Figure 2. The Bayesian Predictive Network (BP-Net) for probabilistic prediction of future frames from noisy observations.

3.1. Theoretical Foundations

Let us go back to Figure 1(b). Due to both perceptual noise and dynamics uncertainty, the belief distributions around the true hidden state at each time stamp $\text{Bel}(s_t) = P(s_t|x_{1:t}, z_{2:t}, \epsilon_{1:t})$ can be approximated by

$$\begin{aligned}\text{Bel}(s_t) &= \eta P(x_t|s_t, \epsilon_t) P(s_t|x_{1:t-1}, z_{2:t}, \epsilon_{1:t}) \\ &= \eta P(x_t|s_t, \epsilon_t) \int P(s_t|s_{t-1}, z_t)\text{Bel}(s_{t-1}) ds_{t-1},\end{aligned} \tag{1}$$

where η is a normalization factor. In this paper, we use a sequential importance sampling (SIS) approach to represent $\text{Bel}(s_t)$ with weighted samples, the so-called particles in the SIS context. We have $\text{Bel}(s_t) \approx \sum_{i=1}^{m} w_t^{(i)} s_t^{(i)}$, where m is the number of particles, $w_t^{(i)}$ is the particle weight, and $\sum_i w_t^{(i)} = 1$. The particles are iteratively updated in a Bayesian manner according to Equation (1). The first step is to randomly draw the particle states from a probabilistic prediction model:

$$s_t^{(i)} \sim P_\phi(s_t|s_{t-1}^{(i)}, z_t^{(i)}), \tag{2}$$

where $P_\phi(s_t|s_{t-1}^{(i)}, z_t^{(i)})$ defines the probability of the new particle state given its last state and a random noise, and ϕ denotes the parameters of the prediction model. The approximation of Equation (1) and Equation (2) is based on summing over all s_{t-1} from which our prediction module $P(s_t|s_{t-1}, z_t)$ could have led to s_t. Any implementation of the Bayes filter algorithm for a continuous state space must represent a continuous belief and approximate it, for example, Kalman filters, which represent it by a Gaussian. Particle filters do not require a Gaussian assumption; they represent it by a set of particles. The second step is to approximate the observation likelihood $P(x_t|s_t, \epsilon_t)$ by updating the particle weights, *i.e.*, $w_t^{(i)}$ is set to the probability of the current observation given the predicted particle state, and updated by an observation measurement model based on parameters θ:

$$w_t^{(i)} \sim P_\theta(w_t|s_t^{(i)}, x_t^{(i)}, \epsilon_t^{(i)}), \quad w_t^{(i)} = \eta w_t^{(i)}, \tag{3}$$

where $\eta^{-1} = \sum_i w_t^{(i)}$. The particle-based approaches have proven to be useful for the highly nonlinear filtering problem. On one hand, they can represent any posterior distribution with an accuracy that depends on the number of particles. On the other hand, they are well suited for dynamical priors and can be easily used in a predictive model. Following the particle-based filtering approaches, we propose Bayesian Predictive Network (BP-Net), which jointly learn a predictive model and an inference model.

3.2. Bayesian Predictive Networks (BP-Net)

As shown in Figure 2, BP-Net implements an end-to-end Bayesian inference-prediction framework using six modules.

Frame encoder. We exploit the stacked Residual Multiplicative Blocks (RMB) [16] to construct the frame encoder and decoder. Unlike traditional SIS approaches, we encode the previous observation to $h_{\text{enc}} = \text{RMB}_\phi(x_{t-1})$ and feed it into the particle state prediction module.

Prior learner. The prior learner generates m random variables $z_t^{(1):(m)}$ for uncertainty modeling. We adopt the approach to learn the sampling priors of $z_t^{(i)}$ by minimizing the KL divergence $\mathcal{D}_{\text{KL}}(Q_\omega(z_t|y_{1:t})||P_\psi(z_t|y_{1:t-1}))$ between two conditional Gaussian distributions [5]. During training, $z_t^{(i)}$ is drawn from $\mathcal{N}(\mu_\omega(y_{1:t}), \sigma_\omega(y_{1:t}))$. During testing, it is drawn from $\mathcal{N}(\mu_\psi(\hat{y}_{1:t-1}^{(i)}), \sigma_\psi(\hat{y}_{1:t-1}^{(i)}))$.

Particle Prediction module. Following Equation (2), the prediction module updates the particle states based on their previous states. It also receives the encoded hidden state h_{enc} of the previous observations and the random variables $z_t^{(i)}$ generated by the prior learner. The key component in the prediction module is a convolutional LSTM that updates particle states as follows:

$$[s_t^{(i)}, c_t^{(i)}] = \text{ConvLSTM}(\text{concat}(h_{\text{enc}}, z_t^{(i)}), s_{t-1}^{(i)}, c_{t-1}^{(i)}), \tag{4}$$

where $c_t^{(i)}$ is a memory cell that retains information from a deep history of particle states. Note that all variables above are $\mathbb{R}^{H\times W\times C}$ tensors. Please refer to [27] for the key equations inside the ConvLSTM layer.

Measurement module. According to Equation (3), the measurement module calculates the posterior likelihood of the current observation x_t given each predicted particle state $s_t^{(i)}$. During training, we initialize all particle states with the same weights $w_t^{(1):(m)} = 1/m$. At each time stamp of the inference phase (w.r.t. the input sequence), the measurement module updates $w_t^{(1):(m)}$ conditioning on observations $x_{t-\tau+1:t}$. When a new observation x_t is received, the measurement module decides the importance of each particle state $s_t^{(i)}$. If a particle has positive correlations with the new observation, the measurement module tends to increase its weight. We apply a sliding window with a length of τ to sequential observations, so that during the prediction phase where new observations are not available, the measurement module has a broad view of $x_{K-\tau+1:K}$, where K is the length of the input sequence. As shown in Figure 2(b), we first use stacked convolution layers to encode current observations $x_{t-\tau+1:t}$ and a particle state $s_t^{(i)}$ into vectors:

$$\begin{aligned}[\mu_\theta,\sigma_\theta] &= l_\theta(x_{t-\tau+1:t}), \quad e_t = f_\theta(x_{t-\tau+1:t}),\\ k_t^{(i)} &= g_\theta(s_t^{(i)}),\end{aligned} \tag{5}$$

where θ indicates the overall parameters of the measurement module, and l_θ, f_θ, and g_θ are convolutional networks with different parameters. We then sample observation noise vectors using the re-parametrization trick $\epsilon_t^{(i)} \sim \mathcal{N}(\mu_\theta,\sigma_\theta)$. Finally, we concatenate e_t, $k_t^{(i)}$ and $\epsilon_t^{(i)}$ and update the particle weights using a GRU layer and another feedforward network u_θ:

$$\begin{aligned}h_t^{(i)} &= \text{GRU}(\text{concat}(e_t, k_t^{(i)}, \epsilon_t^{(i)}), h_{t-1}^{(i)}, c_{t-1}^{(i)}),\\ w_t^{(i)} &= u_\theta(h_t^{(i)}),\end{aligned} \tag{6}$$

where $h_t^{(i)}$ is the hidden state in the GRU, which correlate the predictions of particle weights at different time stamps.

Note that we do not use the re-sampling approach as the traditional particle-based filtering algorithm, because in future frames prediction, the recurrent transition states need to be consistent across time. We find that introducing noise vectors $\epsilon_t^{(i)}$ makes BP-Net effectively avoid the so-called *particle degeneracy* problem—one particle dominates most of the particle weight and makes the rest of them useless.

The observation measurement module is a key component of BP-Net that distinguishes it from previous video prediction models. One of its advantages is that it incorporates a particle-based Bayesian filtering algorithm into the video prediction problem, so that we can handle more complicated situations of noisy input frames. An additional benefit is that it approximates the likelihood of the current noisy observations given each particle states. Thus, we can use the particle weights as the reference for selecting prediction candidates at the test time.

Frame decoder and discriminator. The frame decoder map particle states back to the target space of uncorrupted frames and generates the pixel-level frame predictions. We use 6 RMBs with 2–4 transposed convolution layers for upsampling. It runs $m+1$ times at each time stamp during training by taking individual particles $s_t^{(1):(m)}$ as well as their weighted sum $\sum_i w_t^{(i)} s_t^{(i)}$ as its inputs. The inference model at test time can be seen in the supplementary material, it generates frames only based on $s_t^{(1):(m)}$. We also use a discriminator to train our predictive model adversarially. It adopts the DCGAN discriminator architecture [25] and is trained to differentiate the generated frames and the ground truth, uncorrupted frames. The generative model (rest parts of BP-Net) is optimized to fool the discriminator into believing the generated frames are real.

3.3. Objective Function

BP-Net is an end-to-end trainable approach. The prediction module and measurement module are jointly trained with a unified objective function $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{sis}} + \lambda_{\text{vae}}\mathcal{L}_{\text{vae}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}}$, where $\mathcal{L}_{\text{sis}}$ follows the sequential importance sampling algorithm, $\mathcal{L}_{\text{vae}}$ is for optimizing the prior learner and the prediction results of individual particles based on the variational lower bound, $\mathcal{L}_{\text{adv}}$ is for optimizing both the discriminator and the rest parts of BP-Net in an adversarial manner, and λ_{vae} and λ_{adv} are hyper-parameters that are respectively set to 0.0001 and 100 throughout training.

We now discuss these terms in detail. During training, the belief distribution around the ground truth frames is approximated by the normalized weighted sum of all particles $\sum w_t^{(i)} s_t^{(i)}$ followed by a CNN frame decoder. Based on the SIS algorithm, we penalize the $\mathcal{L}_2$ distance between the ground truth frames and the generated frames:

$$\mathcal{L}_{\text{sis}} = \sum_{t=1}^{K+J} \mathcal{L}_2(\text{Dec}_\phi(\sum_{i=1}^{m} w_t^{(i)} s_t^{(i)}, y_t). \tag{7}$$

The second term in the final loss optimizes the conditional VAE. We penalize the $\mathcal{L}_2$ distance between the ground truth future frames and the predicted frames. We also close the KL divergence between two Gaussian distributions of z_t:

$$\begin{aligned}\mathcal{L}_{\text{vae}} = &\frac{1}{m}\sum_{t=2}^{K+J}\sum_{i=1}^{m}\Big[\mathcal{L}_2(\text{Dec}_\phi(s_t^{(i)}), y_t)\\ &+ \mathcal{D}_{\text{KL}}(Q_\omega(z_t^{(i)}|y_{1:t})||P_\psi(z_t^{(i)}|y_{1:t-1}))\Big].\end{aligned} \tag{8}$$

The third term is the adversarial loss provided by the discriminator. It attempts to close the pixel intensity distributions of generated frames and ground truth frames. Unlike the $\mathcal{L}_2$ loss that would tolerate fuzzy predictions, the adversarial loss can approximate multi-modal distributions [22].

During testing, we do not calculate the weighted sum of the particle states. Instead, we select the top-k particles according to their particle weights, which reveal the likelihoods of observations conditioned on each particle state.

We would like to emphasize again that our main contribution is not on applying PFs, but solving the prediction problem in the context of deep learning, with an end-to-end differentiable model to cope with perceptual and prediction uncertainty. In other words, this paper provides a pilot study of integrating particle-based method with deep recurrent networks. Another contribution of this paper is that we introduce the approximate posterior confidence to the predicted future space-time data, which is novel in the area of video prediction and spatiotemporal modeling.

4. Experiments

We train and evaluate our proposed model on two public video datasets that have been widely used in the field of video prediction. To fit them into our setting, we add man-made corruptions to the input frames. The BP-Net remarkably outperforms the compared models on both datasets.

4.1. Compared Models

Deterministic approaches. We compare with the deterministic video prediction models [4, 23, 34]. Deterministic models make point estimations of the future frames given input frames, hence, they tend to generate blur images in a multi-modal prediction setting.

GAN-based approaches. We compare with models [33, 31] that also exploit the adversarial training paradigm. Note that we also leverage adversarial training in BP-Net.

VAE-based approaches. We compare with a state-of-the-art variational model, SVG-LP [5], which is also based on the conditional VAE. As BP-Net is also a variational model, in the following experiments, we mainly compare it with the SVG-LP model. Note that the SVG-LP model focuses on future uncertainty, while BP-Net simultaneous copes with perceptual uncertainty and dynamics uncertainty.

A baseline model with two separate stages. Specifically, we also compare our model with a two-stage baseline. The first stage is a stochastic denoising network. It has the same architecture as BP-Net, and the only difference is that it is trained to reconstruct the noise-free input sequence instead of predicting future frames. Unlike most existing denoising methods, it requires no prior knowledge on the noises, and therefore better fits our problem setup. The second stage is a deterministic prediction network based on the output sequence of the first stage. It consists of an encoder, a ConvLSTM, and a decoder.

4.2. Implementation details

The network details are shown in Table 1. The encoder and decoder are not pre-trained. The entire model is trained from scratch in an end-to-end manner with an Xavier initializer. We apply the scheduled sampling strategy [2] to all of the compared models. This technique can stitch the discrepancy between training and testing. We scale the pixel values of each input frame to $[0, 1]$ and predict 10 future frames from 10 noisy input frames. Unless otherwise stated, we use 30 particles for training and 100 particles for testing. We select the best-performing λ_{vae} (10^{-4}) and λ_{adv} (10^{2}) from $\{10^{-6}, 10^{-4}, \cdots, 10^{2}, 10^{4}\}$. Similar to the previous work [5], we find the performance not very sensitive to λ_{vae}. We use the Adam optimizer [18] with a 10^{-3} learning rate to train BP-Net, and set the batch size of an iteration as 8.

Module	Layers	Output
Encoder	2 Convs, 2 RMBs [16]	$16 \times 16 \times 64$
Prior learner	4 Convs, 2 GRUs	$4 \times \mathbb{R}^{256}$
Prediction	1 ConvLSTM	$16 \times 16 \times 64$
Measurement	4 Convs, 1 GRU, 2 FCs	$\mathbb{R}^1$
Decoder	3 RMBs [16], 2 Deconvs	$64 \times 64 \times 1$
Discriminator	from DCGAN [25]	$\mathbb{R}^1$

Table 1. The architecture details of the BP-Net.

4.3. Moving MNIST Dataset

Dataset construction. The standard Moving MNIST dataset consists of $10{,}000$ training sequences, $3{,}000$ testing sequences and $2{,}000$ validation sequences. Each sequence contains of 20 frames of 64×64 pixels with two flying digits. Based on this dataset, we construct two benchmarks:

- For *perceptual* uncertainty, we make each input frame have a 24×24 randomly localized missing part. The past-to-future mapping is deterministic in this case.
- For *dynamics* uncertainty, we add a time-independent Gaussian noise to the time-invariant speed of the digits (thus the future frames are still predictable). We keep the *perceptual* uncertainty, and so these two kinds of uncertainty are entangled in space-time.

Quantitative results. We show the quantitative results of our proposed BP-Net and the compared models in Table 2 and Table 3. We use the Mean Square Error (MSE) and the Structural Similarity Image Measurement (SSIM) as metrics. A higher SSIM or a lower MSE denotes the better quality of the generated images. On both tasks with perceptual and dynamics uncertainty, the BP-Net performs the best. For each entry of the test set, we first select the sequences with the highest sequential level SSIM or the lowest MSE among 100 random particles. We find that these results are better than all compared models, including SVG-LP. We also find

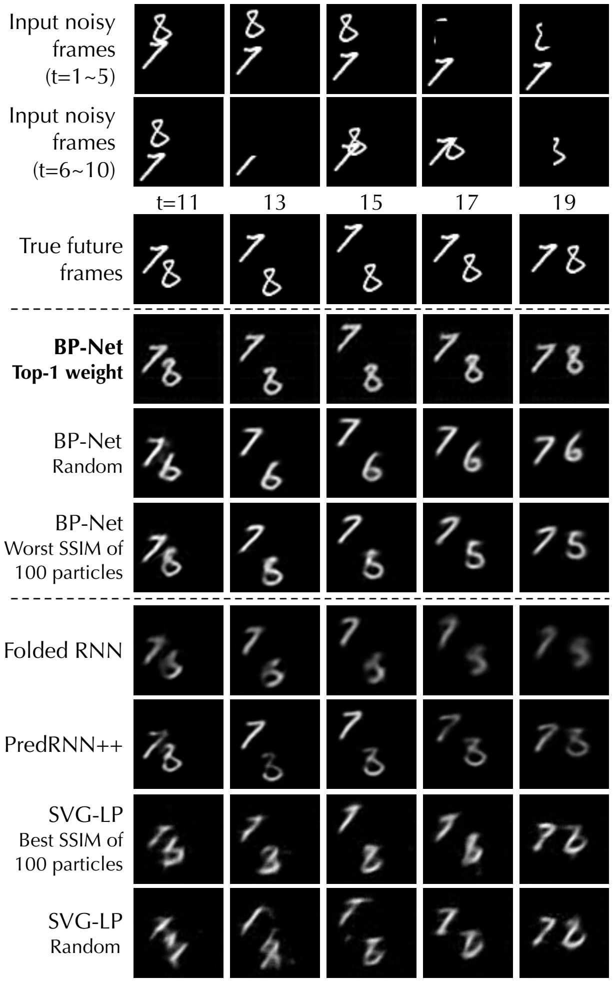

Input noisy frames (t=1~5)
Input noisy frames (t=6~10)
t=11
13
15
17
19
True future frames
BP-Net Top-1 weight
BP-Net Random
BP-Net Worst SSIM of 100 particles
Folded RNN
PredRNN++
SVG-LP Best SSIM of 100 particles
SVG-LP Random

Figure 3. A showcase on the Moving MNIST dataset with noisy inputs. 10 future frames are generated from 10 observations. Future frames are shown in intervals of 2 time stamps. We deploy 100 samples for SVG-LP and BP-Net at test time. Note that BP-Net models the future uncertainty well and makes diverse predictions. Also note that the prediction candidate with the Top-1 importance weight by BP-Net matches well with high prediction quality.

Model	SSIM	MSE
DFN [4]	0.732	93.7
Folded-RNN [23]	0.750	73.6
PredRNN++ [34]	0.779	67.0
VideoGAN [33]	0.706	82.5
MCnet [31]	0.763	79.4
SVG-LP [5] (best of 100 samples)	0.789	56.7
SVG-LP [5] (worst of 100 samples)	0.744	72.1
BP-Net (best of 100 particles)	**0.810**	**51.8**
BP-Net (with highest particle weight)	0.807	53.2
BP-Net (worst of 100 particles)	0.768	63.1

Table 2. Results on the Moving MNIST dataset with perceptual uncertainty, averaged over the 10 predicted frames. Higher SSIM or lower MSE denotes the better quality of the generated images.

Model	SSIM	MSE
DFN [4]	0.658	122.4
Folded-RNN [23]	0.718	81.7
PredRNN++ [34]	0.735	75.8
VideoGAN [33]	0.688	97.1
MCnet [31]	0.703	83.5
SVG-LP [5] (best of 100 samples)	0.757	66.0
SVG-LP [5] (worst of 100 samples)	0.689	80.4
BP-Net (best of 100 particles)	**0.788**	**58.5**
BP-Net (with highest particle weight)	0.783	59.1
BP-Net (worst of 100 particles)	0.730	74.2

Table 3. Results on the Moving MNIST dataset with the entangled perceptual and dynamics uncertainty.

that even the worst samples can easily outperform the deterministic models and the GAN-based models. We give this credit to the integrated filtering and predictive framework.

Another finding is that the generated sequences with the highest particle weights (the average value over all time stamps) are almost the best sequences among all candidates. It indicates that the particle weights can roughly estimate the correlations between the observations and the predicted samples. We can pick results before calculating SSIM or MSE. It will be useful for some online applications, in which the ground truth frames are not available.

Qualitative results. Figure 3 gives an example of the future frames generated by our model and some compared models. We have the following findings. First, the deterministic models make very blurry predictions because they can only learn unimodal past-to-future mappings. Second, BP-Net consistently generates more recognizable frames compared with SVG-LP. Third, the Top-1 prediction candidate by BP-Net with the highest particle weight achieves a more accurate estimation of the next 10 frames. Last but not least, the frame content by our model is diverse, showing digits "8" (BP-Net Top-1 weight), "6" (BP-Net random), "5" (BP-Net worst SSIM of 100 particles) in the estimated future sequences. This result indicates that BP-Net is not likely to suffer from the particle degeneracy problem.

Ablation study. Table 4 includes results of ablation studies. In baseline-I, we use vanilla LSTMs to take the place of the ConvLSTMs. It verifies the effectiveness of using the ConvLSTMs in the prediction model. In baseline-II, we remove the random vectors ϵ_t in the measurement module. Note that in this circumstance, the BP-Net is easily suffered from the particle degeneracy problem as the highest particle weight approaching 1. Baseline-III is the *two-stage baseline model* that is previously described in Section 4.1. Our end-to-end inference-prediction framework significantly outperforms the combination of a stochastic denoising method and a deterministic prediction method which have similar network architectures to BP-Net.

Model	SSIM	MSE	Highest Particle Weight
Baseline I	0.765	63.4	0.57
Baseline II	0.782	60.3	0.99
Baseline III	0.756	68.0	n/a
BP-Net	**0.788**	**58.5**	0.23

Table 4. An ablation study on the Moving MNIST with entangled perceptual and dynamics uncertainty. We report the best results among 100 particles. See text for details of the baseline models.

Model	SSIM	MSE
DFN [4]	0.758	136.7
Folded-RNN [23]	0.765	124.2
PredRNN++ [34]	0.772	113.8
VideoGAN [33]	0.766	120.6
MCnet [31]	0.781	105.0
SVG-LP [5] (best of 100 samples)	0.775	96.8
SVG-LP [5] (worst of 100 samples)	0.757	113.4
Denoising + PredRNN++	0.781	101.7
Denoising + SVG-LP (best)	0.783	97.0
BP-Net (best of 100 particles)	**0.792**	**88.1**
BP-Net (with highest particle weight)	0.791	88.5
BP-Net (worst of 100 particles)	0.774	104.7

Table 5. Results on the KTH action dataset with noisy inputs, including those of the two-stage baselines that combine the denoising part of BP-Net with other video prediction models.

4.4. KTH Dataset

Dataset construction. The original KTH dataset consists of 600 videos of 15–20 seconds with 25 persons performing 6 actions. We resize the frames into 128 × 128 pixels. In our task, each input frame has a 64 × 64 randomly localized area covered with the mosaic. We use person 1–16 for training and person 17–25 for testing. Note that the dynamics uncertainty always exists in natural videos.

Results. Table 5 shows quantitative results. The BP-Net performs the best on the KTH dataset. We also notice that, despite having more parameters, the two-stage methods perform worse than the final BP-Net, because *the video denoising part can only capture the perceptual uncertainty rather than the dynamics uncertainty.* Figure 4 shows an example of the predicted sequence. We may find that our model generates reasonable diverse content (see the different poses at the last time stamp). We also find that the sequence with the highest particle weights is very close to the best sample. Figure 5 show the model sensitivity to different numbers of training and test particles. We find that using 30 training particles and 100 test particles strikes a balance between prediction quality and efficiency. As shown in Figure 6, these visualizations reflect the diversity of the generated frames of BP-Net, showing that BP-Net does not suffer from the *particle degeneracy* problem. Further, the output sequence of the top-1 particle aligns well with the ground truth future frames, suggesting the accuracy of BP-Net predictions.

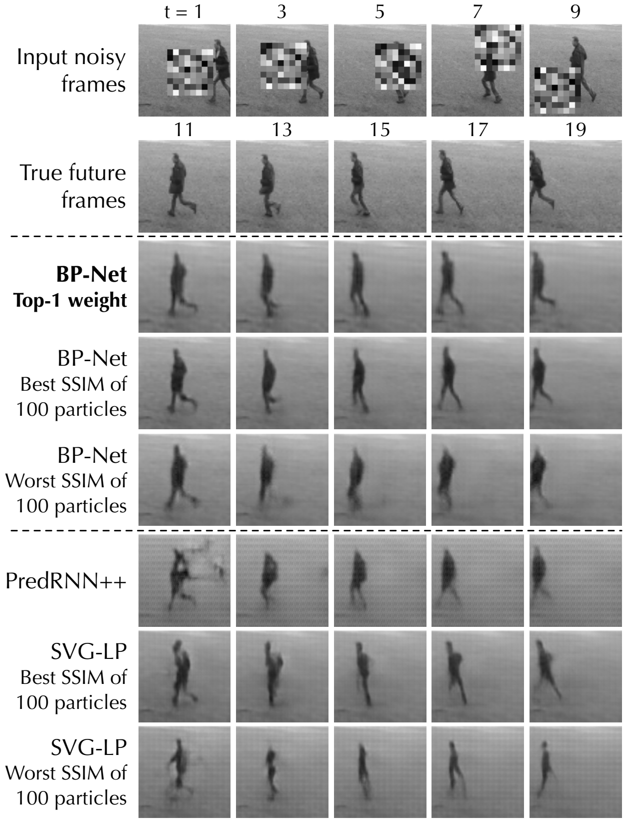
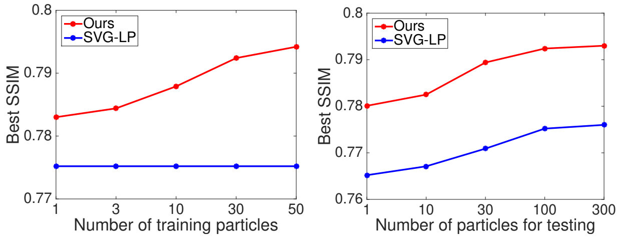

t = 1
3
5
7
9
Input noisy frames
11
13
15
17
19
True future frames
BP-Net Top-1 weight
BP-Net Best SSIM of 100 particles
BP-Net Worst SSIM of 100 particles
PredRNN++
SVG-LP Best SSIM of 100 particles
SVG-LP Worst SSIM of 100 particles

Figure 4. A showcase of predicting 10 future frames with noisy inputs. Frames are shown in intervals of 2 time stamps.



(a) Evaluated with 100 particles

(b) Trained with 30 particles

Figure 5. The influence of number of particles.

How does the confidence align with prediction quality? Figure 7 shows how the particle weights evolving over time. This is the same video sequence as what in Figure 4. All particle weights are initialized as 0.01 as there are 100 particles during testing. We can see that the particle weight of the worst predicted sequence (by SSIM) remains low. Actually, it is even lower than 10^{-5} at the last time stamp. On the contrary, the particle weight of the best particle state increases over time. Therefore, we can see that ranking the particle states according to their particle weights is not only theoretically reasonable but also empirically effective.

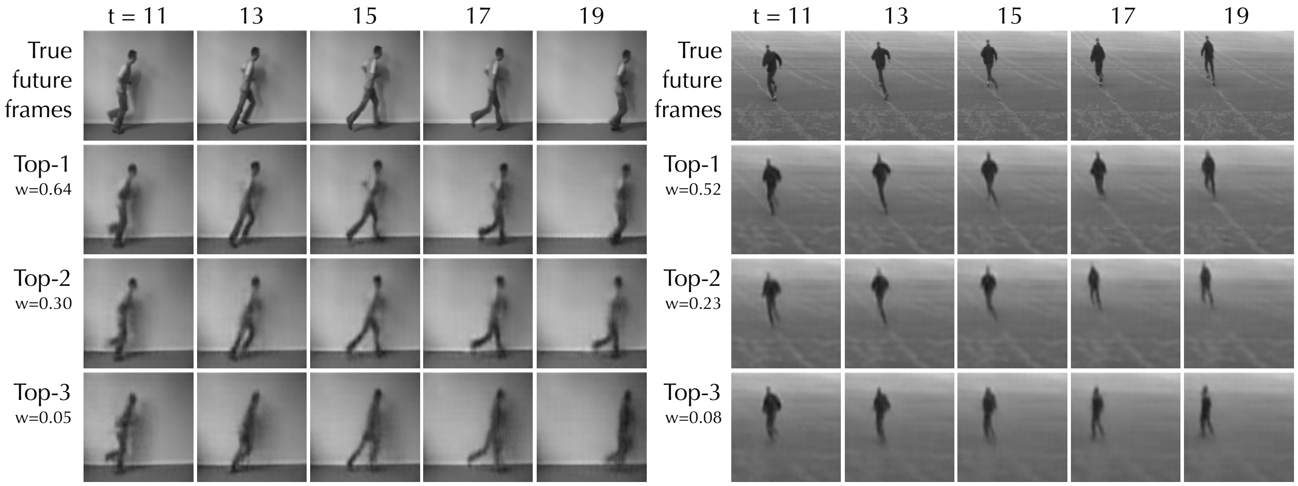


Figure 6. The generated future frames of the top-3 particles based on the same input sequence. We show them in intervals of 2 time stamps.

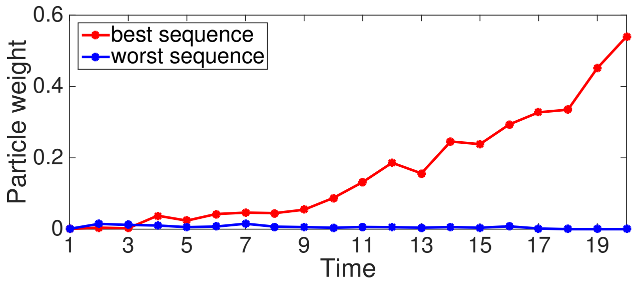
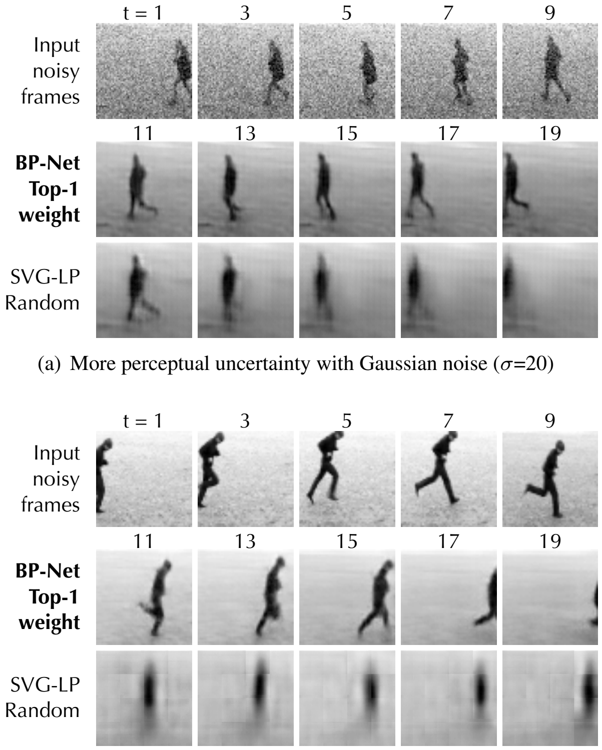


Figure 7. The weight curves of the particles that result in the best/worst SSIM values, indicating the consistency between the prediction quality and the value of particle weights. We use the video sequence in Figure 4 as the test sample.

Results with more perceptual and dynamics uncertainty. We increase the Gaussian noise for more perceptual uncertainty and add time-independent image jittering for dynamics uncertainty, which will result in more multi-modal inputs-outputs relationships. Results are shown in Figure 8. The generated video sequences with the highest particle weights by BP-Net have better quality than those randomly sampled by SVG-LP.

5. Conclusions

In this paper, we studied a new problem of predicting future frames from noisy videos, which is meaningful for practical online video applications. To tackle this problem, we proposed a probabilistic model, Bayesian Predictive Network (BP-Net), based on the sequential importance sampling (SIS) algorithm, also known as the particle filtering algorithm. Different from all existing video prediction models, BP-Net makes future predictions based on the inference of underlying physical states. BP-Net outperformed all compared models on two public video datasets of noisy videos. We achieved an additional benefit by integrating the particle-based filtering algorithm into our proposed model. BP-Net approximates the likelihood of the current observation given each possible particle states. Experiments suggest that higher posterior confidence reflects better prediction qualities.



(a) More perceptual uncertainty with Gaussian noise (σ=20)

(b) More dynamics uncertainty with frame jittering (0-10 pixels)

Figure 8. Qualitative results on KTH under more uncertainty. 10 future frames are predicted from 10 previous noisy inputs with increased noise and jittering (shown in 2 time stamps intervals).

Acknowledgements

This work is in part supported by ONR MURI N00014-16-1-2007 and NSFC 61772299.

References

[1] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. In *ICLR*, 2018. 2

[2] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *NeurIPS*, 2015. 5

[3] Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *NeurIPS*, 2018. 2

[4] Bert De Brabandere, Xu Jia, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In *NeurIPS*, 2016. 5, 6, 7

[5] Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *ICML*, 2018. 2, 3, 5, 6, 7

[6] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *NeurIPS*, 2015. 2

[7] Emily L Denton et al. Unsupervised learning of disentangled representations from video. In *NeurIPS*, 2017. 2

[8] Frederik Ebert, Chelsea Finn, Alex X Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. In *CoRL*, 2017. 1

[9] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *NeurIPS*, 2016. 2

[10] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *ICRA*, 2017. 1

[11] Ian J. Goodfellow, Jean Pougetabadie, Mehdi Mirza, Bing Xu, David Wardefarley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *NeurIPS*, 2014. 2

[12] Shixiang Shane Gu, Zoubin Ghahramani, and Richard E Turner. Neural adaptive sequential monte carlo. In *NeurIPS*, 2015. 2

[13] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017. 2

[14] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li F Fei-Fei, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *NeurIPS*, 2018. 2

[15] Rico Jonschkowski, Divyam Rastogi, and Oliver Brock. Differentiable particle filters: End-to-end learning with algorithmic priors. In *RSS*, 2018. 2

[16] Nal Kalchbrenner, Aaron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *ICML*, 2017. 2, 3, 5

[17] Peter Karkus, David Hsu, and Wee Sun Lee. Particle filter networks: End-to-end probabilistic localization from visual observations. *arXiv preprint arXiv:1805.08975*, 2018. 2

[18] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

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

[20] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Flow-grounded spatial-temporal video prediction from still images. In *ECCV*, 2018. 2

[21] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. In *ICLR*, 2017. 2

[22] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 2, 5

[23] Marc Oliu, Javier Selva, and Sergio Escalera. Folded recurrent neural networks for future video prediction. In *ECCV*, 2018. 5, 6, 7

[24] Viorica Patraucean, Ankur Handa, and Roberto Cipolla. Spatio-temporal video autoencoder with differentiable memory. In *ICLR Workshop*, 2016. 2

[25] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016. 4, 5

[26] 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. 2

[27] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NeurIPS*, 2015. 1, 2, 4

[28] Xingjian Shi, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Deep learning for precipitation nowcasting: A benchmark and a new model. In *NeurIPS*, 2017. 2

[29] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. In *ICML*, 2015. 2

[30] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *CVPR*, 2018. 2

[31] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. In *ICLR*, 2017. 2, 5, 6, 7

[32] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. Learning to generate long-term future via hierarchical prediction. In *ICML*, 2017. 2

[33] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016. 2, 5, 6, 7

[34] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S Yu. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *ICML*, 2018. 2, 5, 6, 7

[35] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and S Yu Philip. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *NeurIPS*, 2017. 1, 2

[36] Nevan Wichers, Ruben Villegas, Dumitru Erhan, and Honglak Lee. Hierarchical long-term video prediction without supervision. In *ICML*, 2018. 2

[37] Ziru Xu, Yunbo Wang, Mingsheng Long, and Jianmin Wang. Predcnn: Predictive learning with cascade convolutions. In *IJCAI*, 2018. 1

[38] Tianfan Xue, Jiajun Wu, Katherine Bouman, and Bill Freeman. Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In *NeurIPS*, 2016. 2

[39] Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *AAAI*, 2017. 1