SyncVP: Joint Diffusion for Synchronous Multi-Modal Video Prediction

Supplementary Material

1. Further implementation details

1.1. Latent autoencoders

The autoencoders are trained separately on each dataset following the pipeline defined by the authors of PVDM [5]. We train them until convergence of FVD, SSIM and PSNR and proceed with a second fine-tuning stage with the adversarial loss for very few iterations. Given a sequence of frames $\mathbf{x}$ of shape $T \times H \times W \times C$ as input, the encoder produces a latent vector of shape $C' \times L$, where L is computed as follows: $\frac{H \cdot W}{P^2} + \frac{T}{P} \cdot (H + W)$. We use the default configuration of PVDM and keep the patch size $P = 4$ for 64×64 and 128×128 resolution, while we use $P = 8$ for higher resolutions. The number of channels for the latent vector is also set to the default value $C' = 4$ for all experiments up to 128×128 resolution, otherwise we use $C' = 16$. For example, for Cityscapes (128×128) the latent vector has the shape of 4×1536. The number of hidden channels for the autoencoder is set to 192, differing from 384 used in the original version for efficiency reasons.

1.2. BAIR depth ground-truth

We generate ground-truth depth images for the BAIR [1] dataset using the off-the-shelf DepthAnything-v2 [4] model, specifically the '`vit-b`' version. Depth estimation is performed frame by frame, which can result in flickering in the depth videos due to the lack of temporal coherence.

1.3. SYNTHIA semantic segmentation maps

The SYNTHIA dataset [2] provides semantic segmentation maps for each RGB frame, consisting of 16 distinct classes. These maps are represented as 3-channel images, where the first channel encodes the class IDs, and the remaining two channels assign instance IDs to individual objects. For our work, we focused solely on the class IDs, transforming the first channel into a grayscale image by dividing the original values by 15 and rescaling them to the range $[0, 255]$. To resize the images to 128×128 pixels, we employed nearest-neighbor down-sampling in order to preserve the integrity of class labels and avoid blending between classes during resizing.

1.4. ERA5-Land data processing

Surface pressure (sp) is expressed in Pascals (Pa), while the two-meter temperature (t2m) is given in Kelvin (K). To adapt these data modalities to our training setup, we normalized them using the minimum and maximum values provided in the dataset's metadata. For evaluation, we rescale the predictions to their original range and compute the L_1 error. The data contains NaN values where the measurements are missing (e.g. seas and ocean). We set these values to 0 for training and mask them out in the prediction before computing evaluation metrics.

1.5. OpenDV-YouTube training

We additionally experimented on higher resolution videos (256×256). Specifically, we re-trained our approach on Cityscapes in a $8 \rightarrow 8$ setting with 256×256 resolution. We then finetune this model on a small subset of the OpenDV-Youtube [3] dataset. The subset essentially includes one video (ID: `JS0gJxhFFJ8`) of 65 minutes at 30 fps. The initial and last frames containing text overlay are dropped. The depth maps for these videos are not available, thus, similarly to BAIR (Sec. 1.2), we used DepthAnything-v2 [4] version '`vit-l`' on the raw images before center cropping and resizing them to 256×256. We report the evaluation metrics in Tab. 1 and show additional qualitative results in Fig. 5.

Models	RGB			Depth	
	FVD↓	SSIM↑	LPIPS↓	SSIM↑	L_2 ↓
SyncVP	247.14	0.611	224.17	0.972	1.1703

Table 1. Results on OpenDV-Youtube (256×256, $8 \rightarrow 24$).

2. Training ablation

We provide additional ablation results (Tab. 2) to show the benefits of using a two-stage training pipeline. In the first stage we learn $p(\mathbf{r}_x \mid \mathbf{r}_c)$ and $p(\mathbf{d}_x \mid \mathbf{d}_c)$, while we exploit in the second stage these pre-trained weights to learn the joint conditional distribution $p(\mathbf{r}_x, \mathbf{d}_x \mid \mathbf{r}_c, \mathbf{d}_c)$. To evaluate this, we compare the results of our SyncVP with a version of the model trained from scratch directly on multi-modal data.

In Fig. 1, we show the loss plot to further validate the effectiveness of our shared noise strategy during training.

Two-stage training	RGB			Depth	
	FVD↓	SSIM↑	LPIPS↓	SSIM↑	L_2 ↓
✗	158.53	**0.674**	176.45	0.827	8.044
✓	**84**	0.649	**159.73**	**0.830**	**7.329**

Table 2. Ablation on Cityscapes about the impact of the proposed two-stage training pipeline.

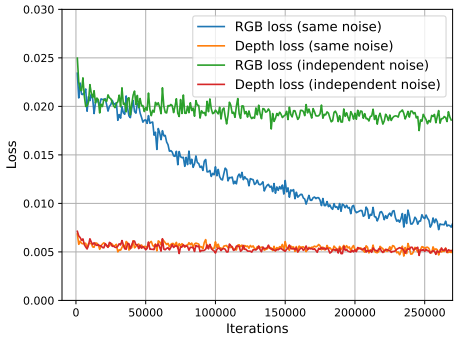


Figure 1. Training loss comparison between using the same noise for both modalities or independent noise for each modality.

3. Additional qualitative results

We provide some more qualitative results, particularly for cases where one modality is missing. Specifically, Fig. 2 demonstrates how our model is able to predict future frames using only the non-RGB modality as observation. Such task is way more complex than standard joint conditional generation or the case in which the low detail modality (depth or semantic) is missing. Nevertheless, the results still exhibit strong cross-modal alignment between the predicted frames. Fig. 3 shows further examples where the model is conditioned solely on past RGB frames. In these cases, our approach is still able to predict aligned depth or semantic segmentation images.

3.1. Agriculture data

We additionally apply our model on agricultural data collected from a sweet pepper greenhouse. The dataset contains images of 415 plants captured on four different dates, which can be considered as 415 short video sequences. Given the dataset's limited size and the challenge of forecasting the plant growth stages, we adopt a leave-one-out strategy for training and testing. All images are resized to a resolution of 512×288, and depth is estimated using DepthAnything-v2 [4]. An example of a prediction is shown in Fig. 6.

4. Robustness to noise

Beside showing the ability of SyncVP to deal with missing modality input, we further test its resilience to noisy input. Namely, we inject random Gaussian noise with increasing σ on the depth input. As shown in Tab. 3, the noisy input does not affect much the predictions.

noise σ	RGB FVD↓	RGB SSIM↑	RGB LPIPS↓
5	87.78	0.641	161.61
2.5	85.98	0.644	160.75
0	**84**	**0.649**	**159.73**

Table 3. Impact of Gaussian noise in observed disparity (Cityscapes).

References

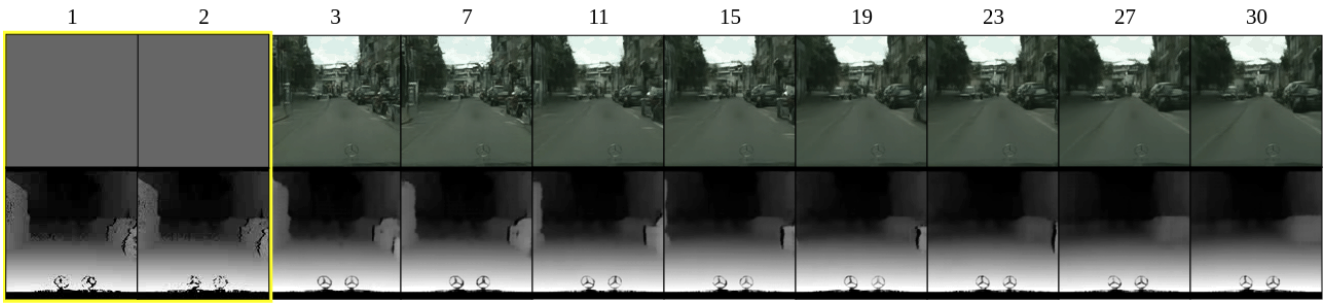
[1] Frederik Ebert, Chelsea Finn, Alex Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. In *Conference on Robot Learning (CoRL)*, 2017. 1

[2] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3234–3243, 2016. 1

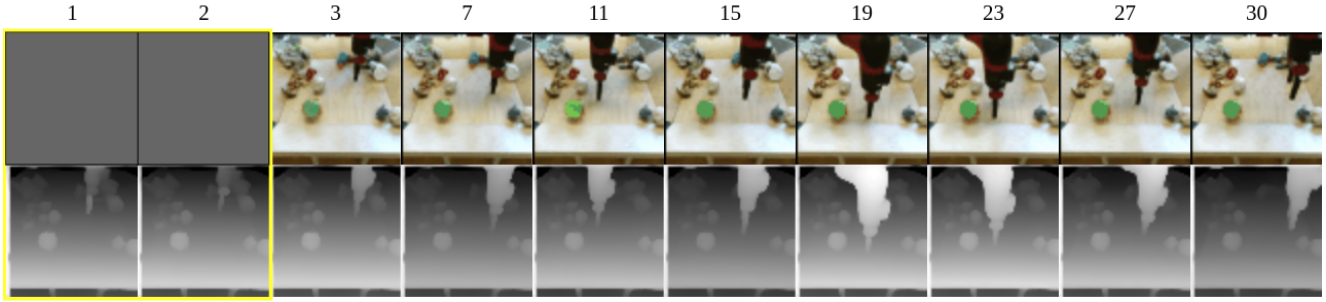
[3] Jiazhi Yang, Shenyuan Gao, Yihang Qiu, Li Chen, Tianyu Li, Bo Dai, Kashyap Chitta, Penghao Wu, Jia Zeng, Ping Luo, Jun Zhang, Andreas Geiger, Yu Qiao, and Hongyang Li. Generalized predictive model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1, 6

[4] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37: 21875–21911, 2024. 1, 2

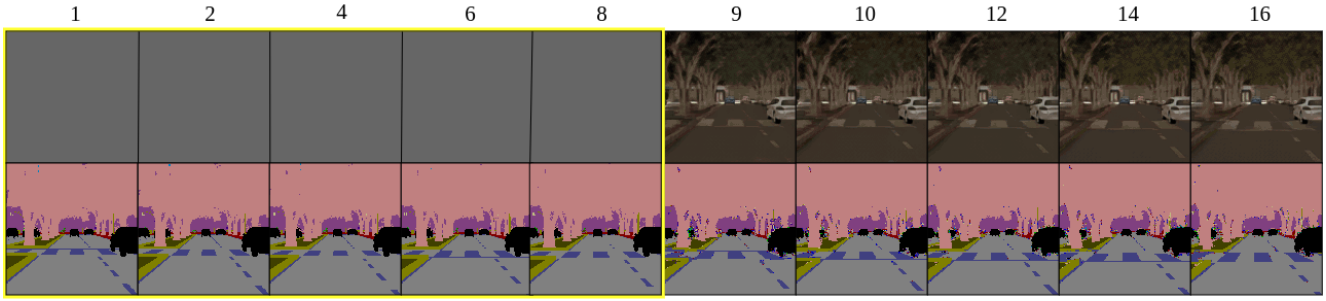
[5] Sihyun Yu, Kihyuk Sohn, Subin Kim, and Jinwoo Shin. Video probabilistic diffusion models in projected latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1



(a) Prediction on Cityscapes using only depth conditioning.

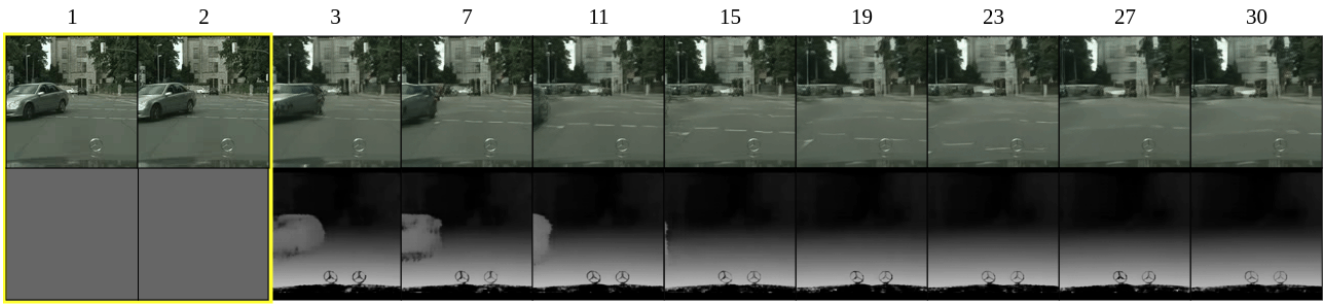


(b) Prediction on BAIR using only depth conditioning.

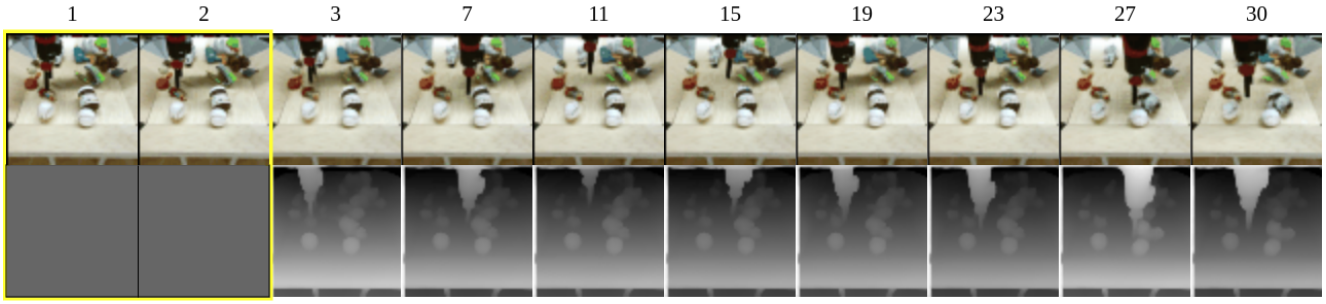


(c) Prediction on SYNTHIA using only semantic segmentation maps conditioning.

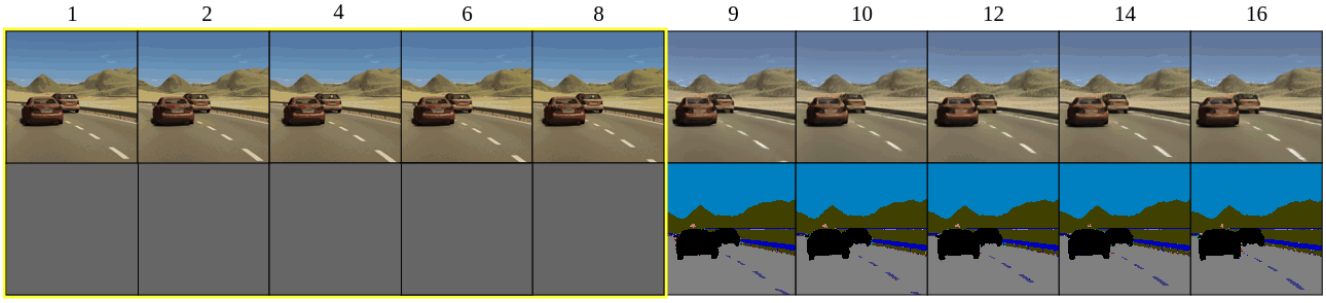
Figure 2. SyncVP video prediction without conditioning on past RGB frames .



(a) Prediction on Cityscapes.



(b) Prediction on BAIR.



(c) Prediction on SYNTHIA.

Figure 3. SyncVP video prediction with conditioning only on RGB frames.

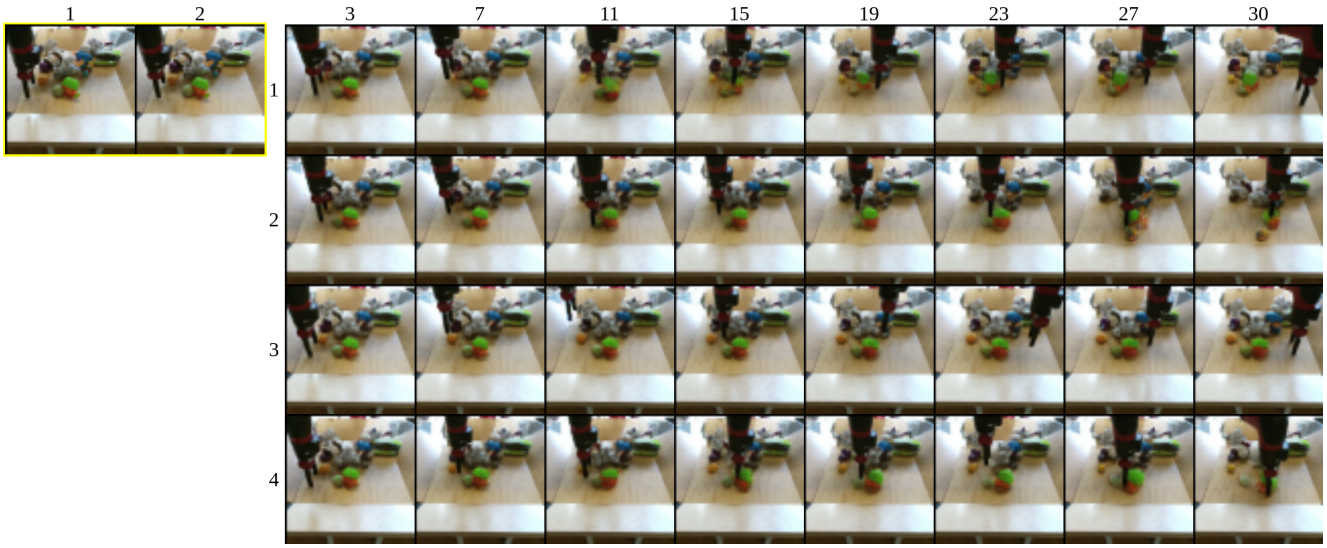


Figure 4. Multiple predicted trajectories for the same observation on BAIR.

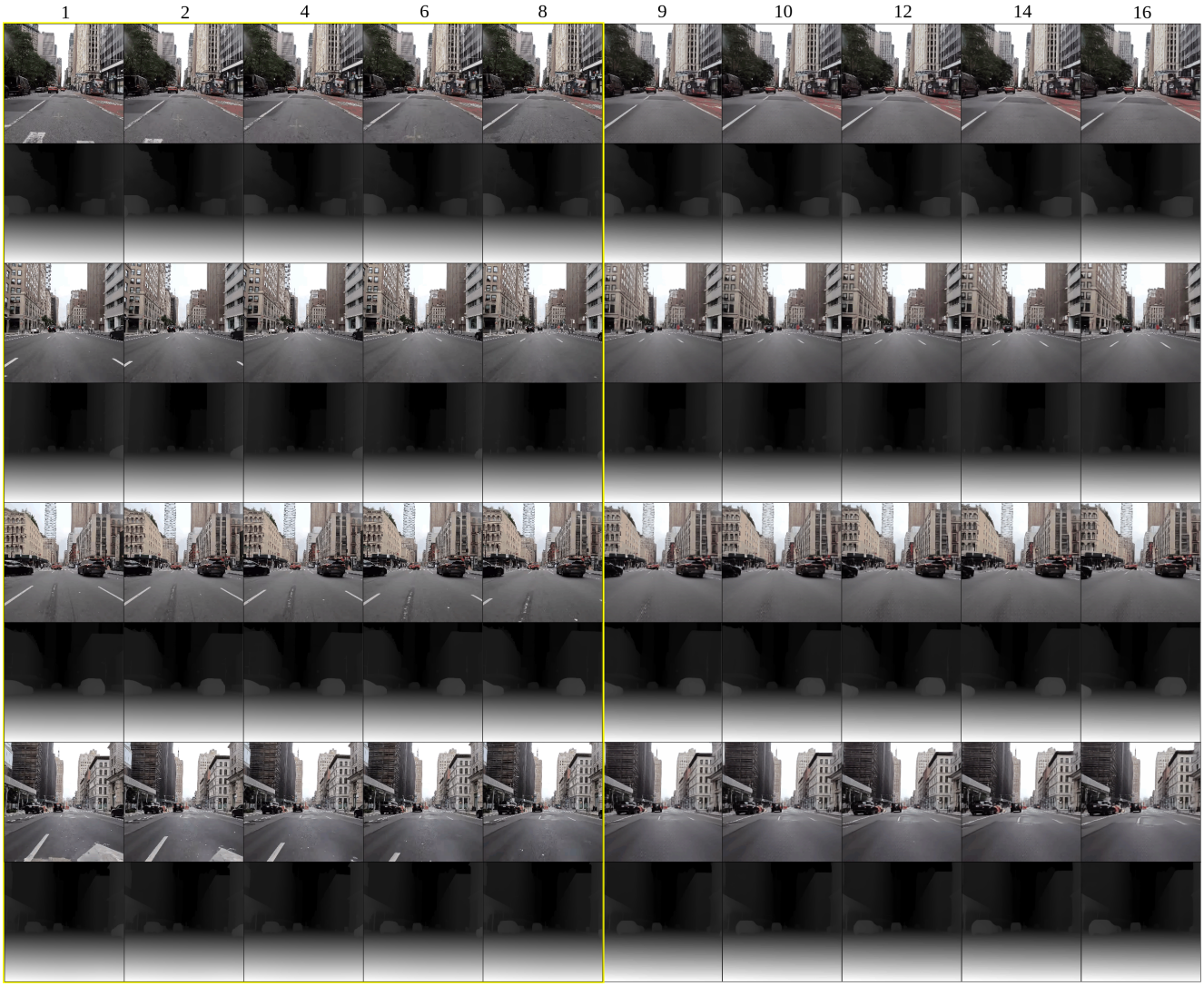


Figure 5. SyncVP predictions on OpenDV-Youtube [3] (256×256).

Figure 6. SyncVP prediction ($3 \rightarrow 1$) on agriculture timeseries data.