3D Motion Decomposition for RGBD Future Dynamic Scene Synthesis

Xiaojuan Qi* University of Oxford
Zhengzhe Liu* DJI
Qifeng Chen HKUST
Jiaya Jia CUHK & YouTu Lab

Abstract

A future video is the 2D projection of a 3D scene with predicted camera and object motion. Accurate future video prediction inherently requires understanding of 3D motion and geometry of a scene. In this paper, we propose a RGBD scene forecasting model with 3D motion decomposition. We predict ego-motion and foreground motion that are combined to generate a future 3D dynamic scene, which is then projected into a 2D image plane to synthesize future motion, RGB images and depth maps. Optional semantic maps can be integrated. Experimental results on KITTI and Driving datasets show that our model outperforms other state-of-the-arts in forecasting future RGBD dynamic scenes.

1. Introduction

Future prediction is an exciting direction with limitless potential applications in decision-making, control system design, and navigation for intelligent agents. In this paper, we study RGBD future scene synthesis, which refers to prediction of videos and depth given a number of past frames.

Most approaches on future prediction aim to predict a specific component in the future scene. They are mostly to predict future color video frames [37, 30, 6, 15, 11, 12, 13] or facilitate semantic understanding, including future semantic segmentation [15, 11], instance segmentation prediction [14], and 2D motion trajectories [12, 11].

Depth prediction is still new in this area with early work of [17]. RGBD future prediction makes it possible to model real-world dynamics. Existing approaches are mostly with 2D data and are self-supervised learning based. These approaches take past frames as input. Then a deep neural network is utilized to directly generate future frames [4, 13] with 2D optical flow [12, 11] as an intermediate representation. Approaches of [29, 6, 31] disentangled foreground and background in 2D space. Luo *et al*. [16] proposed an unsupervised solution for forecasting 3D motion in RGBD data. This framework only operates in 2D domain and does not explicitly reason the future 3D scene.

We note when the underlying 3D geometry of a scene is ignored, it becomes difficult to obtain accurate optical flow prediction since optical flow reflects a level of 2D projection of 3D motion in the physical world. Further, without full geometric understanding, it is challenging to estimate future depth where its change is affected by both 3D camera motion and object motion. Our experiments show that simply training a deep neural network in 2D for depth prediction is not feasible.

With this understanding, unlike previous work, we explicitly reason scene dynamics in 3D space, jointly predicting semantic segmentation, RGB pixel color, and depth information. For 3D motion, we separately forecast ego-motion and object motion in the future. Our main contribution is threefold.

First, we raise the RGBD future prediction problem and propose a self-supervised 3D motion decomposition approach for forecasting 3D motion without labeled data. Second, on top of the predicted 3D motion, we present a general framework for holistic future scene prediction for motion, semantic, depth, and RGB images. Finally, our experimental results on KITTI [8], Driving [20] datasets show that our method is effective to solve this new problem.

2. Related Work

Prior work on future prediction can be roughly categorized into two groups. The first focuses on designing deep neural network architectures or loss functions to directly predict future RGB video frames [13, 19, 4, 33, 7, 2] or high-level semantic description [15, 14, 30, 31]. Direct future prediction is challenging because the solution space is enormously large with high uncertainty. Xue *et al*. [37] proposed the Cross Convolutional Network to model future frames in a probabilistic fashion. Similarly, Byeon *et al*. [4] proposed a future prediction framework that aggregates contextual information with LSTM to avoid "blind spot problem". Progressive GAN [2] progressively synthesizes frames from coarse to fine resolution. Luc *et al*. [15] proposed an auto-regressive model for predicting semantic segmentation. The following work [14] contains a feature prediction network for future instance segmentation.

The other group concentrates on exploiting or modeling

* indicates co-first authorship.

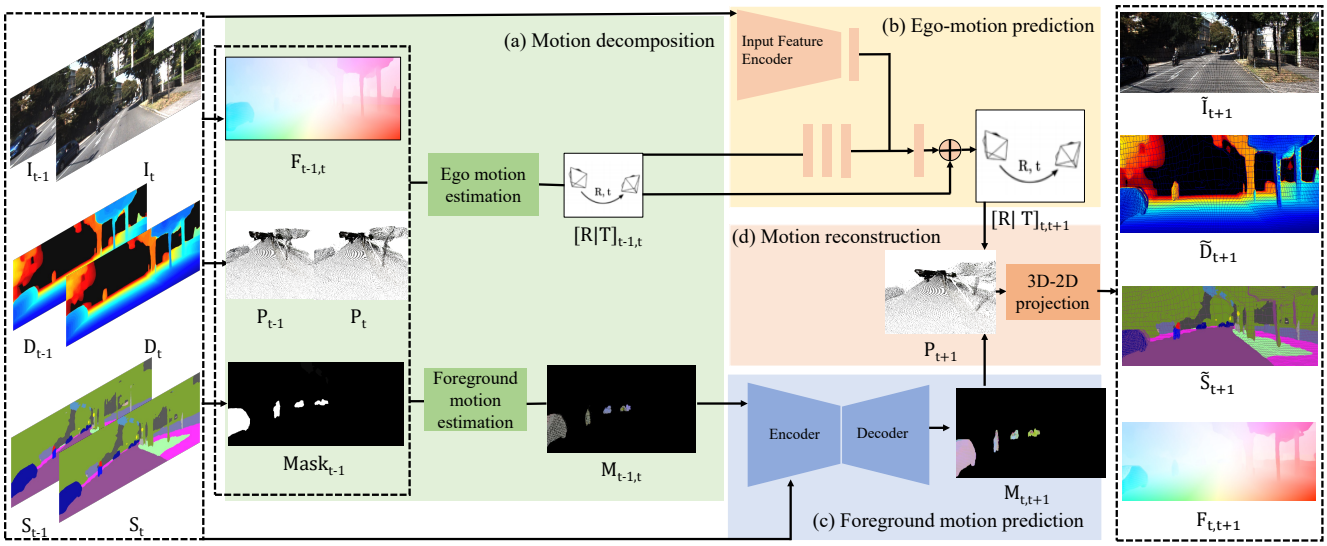

Figure 1: Motion forecasting with decomposition and composition. The input includes images (I_{t-1}, I_t), depth maps (D_{t-1}, D_t), and semantic maps (S_{t-1}, S_t). (a) Motion decomposition module decomposes motion into ego motion $[R|T]_{t-1,t}$ and moving object motion $M_{t-1,t}$. (b) The ego-motion prediction network and (c) the foreground motion prediction network generate future ego-motion $[R|T]_{t,t+1}$ and foreground motion $M_{t,t+1}$ respectively. (d) The motion composition module composes a predicted motion field and a new 3D point cloud P_{t+1}. P_{t+1} is then projected to a 2D image plane. $M_{t-1,t}$ and $M_{t,t+1}$ are color coded where R, G, B channels represent movement along x, y, z directions.

motion for understanding future dynamics [12, 11, 29, 26, 23, 32]. Liang *et al*. [12] jointly reasoned the duality relationship of optical flow and RGB videos with an adversarial objective. Terwilliger *et al*. [26] suggested a recurrent flow prediction framework for semantic prediction. Reda *et al*. [23] learned a motion vector and a kernel for each pixel to synthesize future frames. Jin *et al*. [11] designed a model that jointly predicts complementary optical flow and semantic information. Walker *et al*. [32] utilized variational methods to capture future uncertainty for motion trajectory prediction from a static image. Luo *et al*. [16] directly predicted future 3D trajectories via LSTM for RGBD videos.

Mahjourian *et al*. [18] and Villegas *et al*. [29] are most related to ours. Mahjourian *et al*. [18] synthesized frames with the estimated depth maps and given future ego-motion. Motion of objects and camera trajectories are not modeled explicitly. Thus, this approach is limited to static scenes where independently moving objects do not exist. In contrast, we explicitly model scene dynamics in 3D space by separately predicting camera and object motion to produce future frames. Villegas *et al*. [29] decomposed motion and content to generate dynamics in videos. An encoder-decoder architecture is utilized to synthesize frames directly, which may result in distortion of rigid objects. On the contrary, our approach follows the geometry constraints and can preserve rigid objects better.

Our work also shares similar spirit with unsupervised motion estimation [38, 39, 28], where motion is decomposed into ego motion and camera motion, and depth estimation [36]. These methods estimate motion and depth in current frame, while we predict future dynamics.

3. Overview

The proposed holistic RGBD future scene synthesis task is to predict future motion and frames. The input includes two most recent RGBD frames (and possibly semantic maps). The goal is to jointly predict future motion, RGB frames, depth maps, and semantic maps. The variation without semantic segmentation as input will be discussed in Section 6.2. Our holistic prediction framework predicts future frames by first forecasting 3D motion (Figure 1) and then synthesizing frames (Figure 2).

To predict motion of future frame $t+1$, we first decompose motion into ego-motion $[R|T]_{t-1,t}$ and foreground object motion $M_{t-1,t}$ (Figure 1(a)). Then an ego-motion prediction network and a foreground motion prediction network are used to synthesize future camera motion $[R|T]_{t,t+1}$ (Figure 1(b)) and 3D foreground motion $M_{t,t+1}$ (Figure 1(c)) separately. 3D points P_t are then locally transformed by $M_{t,t+1}$ and globally transformed by $[R|T]_{t,t+1}$ to generate 3D point cloud P_{t+1} in next frame (Figure 1(d)). It, along with RGBD and semantics, is projected to the image plane in frame $t+1$, resulting in intermediate RGB image $\tilde{I}_{t+1}$, depth map $\tilde{D}_{t+1}$, and semantic map $\tilde{S}_{t+1}$.

These intermediate results are updated by a three-branch

refinement network (Figure 2). It outputs the refined color image I_{t+1}, depth map D_{t+1} and semantic segmentation S_{t+1}, as illustrated in Figure 2. Further, it fills in missing pixels, removes noise and harmonizes structure. A sequence of future video frames can be synthesized by applying this model recurrently in future frames.

4. Motion Forecasting

We introduce the motion decomposition module to estimate ego-motion $[R|T]_{t,t-1}$ (equivalent to finding camera pose) and 3D foreground object motion $M_{t-1,t}$. We present an optical flow based method to separate motion in 3D.

Ego-motion estimation. Ego-motion is estimated based on matching of background pixels. We find corresponding background points and estimate camera trajectory in existing frames. First, we compute point clouds P_{t-1} and P_t from the depth maps as shown in Figure 1(a). Let (u_i, v_i) be the 2D coordinates of pixel i and z_i^t be corresponding depth in frame t. The 3D coordinates $P_t(u_i, v_i) = (x_i^t, y_i^t, z_i^t)$ in the camera coordinate system are derived as

$$\begin{aligned} x_i^t &= (u_i - c_x) * z_i^t / f_x, \\ y_i^t &= (v_i - c_y) * z_i^t / f_y, \end{aligned} \tag{1}$$

where (c_x, c_y) are the coordinates of the camera principal point. f_x and f_y are camera focal lengths. We apply FlowNet 2.0 [10] to obtain 2D correspondence (u_i, v_i) in frame $t-1$ and $(u_i + \Delta u_i^{t-1,t}, v_i + \Delta v_i^{t-1,t})$ in frame t. Also, the 3D location of correspondent points is derived according to Equation (1). Then $[R|T]_{t-1,t}$ is estimated with these points in the background (*e.g.* road, building). A background segmentation mask visualization is given in Figure 1 (black pixels in $Mask_{t-1}$). With point pairs $P_{t-1}(u_i, v_i)$ and $P_t(u_i + \Delta u_i^{t-1,t}, v_i + \Delta v_i^{t-1,t})$, the SVD based algorithm [24] is adopted to estimate ego-motion $[R|T]_{t-1,t}$.

Foreground motion estimation. To compute foreground motion, ego-motion $[R|T]_{t-1,t}$ is utilized to transform P_t to the camera coordinate system in frame $t-1$. The transformed location is denoted as $\bar{P}_{t-1} = [R|T]_{t-1,t}^{-1} P_t$. Then, the 3D motion field $M_{t-1,t}$ (shown in Figure 1) at location (u_i, v_i) is computed as

$$\begin{aligned} &M_{t-1,t}(u_i, v_i) = Mask_{t-1} \odot \\ &[\bar{P}_{t-1}(u_i + \Delta u_i^{t-1,t}, v_i + \Delta v_i^{t-1,t}) - P_{t-1}(u_i, v_i)]. \end{aligned} \tag{2}$$

The motion vector for pixel i is represented as $M_{t-1,t}(u_i, v_i) = (\Delta x_i^{t-1,t}, \Delta y_i^{t-1,t}, \Delta z_i^{t-1,t})$ where $(\Delta x_i^{t-1,t}, \Delta y_i^{t-1,t}, \Delta z_i^{t-1,t})$ represents motion along x, y, z regarding camera coordinates of frame $t-1$.

4.1. Ego-motion Prediction

The ego-motion prediction network shown in Figure 1(b) predicts the next-frame ego-motion. We design a network to estimate the difference between $[R|T]_{t-1,t}$ and $[R|T]_{t,t+1}$. $[R|T]$ can be represented as a 6D vector $(\theta_p, \theta_r, \theta_y, T_x, T_y, T_z)$, where $(\theta_p, \theta_r, \theta_y)$ encodes rotation and (T_x, T_y, T_z) denotes translation.

We first design input feature encoder for the input of color image I_{t-1}, I_t, depth map D_{t-1}, D_t, and semantic map S_{t-1}, S_t. Structure of the input feature encoder has convolutional layers followed by a fully connected layer to generate encoded feature. Meanwhile, a geometric network with three fully connected layers maps previously estimated ego-motion *i.e.* $(\theta_p^{t-1,t}, \theta_r^{t-1,t}, \theta_y^{t-1,t}, T_x^{t-1,t}, T_y^{t-1,t}, T_z^{t-1,t})$ to intermediate feature. The output features of the two networks are then processed by a fully connected layer to produce the difference of ego motion between frames t and $t+1$.

4.2. Foreground Motion Prediction

Our foreground motion prediction network predicts a 3D motion field on foreground pixels. Since background object motion can be determined by the ego-motion combined with depth, we focus on estimating foreground motion. We use a binary mask $Mask_t$ to indicate (potentially) moving objects in frame t. The mask is determined based on the semantic class of each object. For example, a car is in foreground while buildings go to background. The foreground motion prediction network is an encoder-decoder that outputs a three-channel prediction map $M_{t,t+1}$ representing the 3D motion of frame t. The architecture of this network is provided in the supplementary material.

4.3. Motion Reconstruction

The motion reconstruction module reconstructs 3D motion combining the ego-motion $[R|T]_{t,t+1}$ and foreground motion $M_{t,t+1}$. In this process, a 3D point cloud P_t in frame t corresponds to P_{t+1} in frame $t+1$ with relation of

$$P_{t+1} = [R|T]_{t,t+1}[P_t + M_{t,t+1} \odot Mask_t]. \tag{3}$$

Then the 3D point P_{t+1} is projected onto the image plane in frame $t+1$ as

$$\begin{aligned} u_i^{t+1} &= f_x x_i^{t+1} / z_i^{t+1} + c_x, \\ v_i^{t+1} &= f_y y_i^{t+1} / z_i^{t+1} + c_y, \end{aligned} \tag{4}$$

where (u_i^{t+1}, v_i^{t+1}) represents the corresponding location at frame $t+1$ for pixel i in frame t. With this formulation, the future optical flow $F_{t,t+1}$ can be derived as

$$F_{t,t+1}(u_i, v_i) = (u_i^{t+1} - u_i, v_i^{t+1} - v_i), \tag{5}$$

and $\tilde{I}_{t+1}, \tilde{D}_{t+1}, \tilde{S}_{t+1}$ are represented as

$$\begin{aligned} \tilde{D}_{t+1}(u_i^{t+1}, v_i^{t+1}) &= z_i^{t+1}, \\ \tilde{I}_{t+1}(u_i^{t+1}, v_i^{t+1}) &= I_t(u_i, v_i), \\ \tilde{S}_{t+1}(u_i^{t+1}, v_i^{t+1}) &= S_t(u_i, v_i). \end{aligned} \tag{6}$$

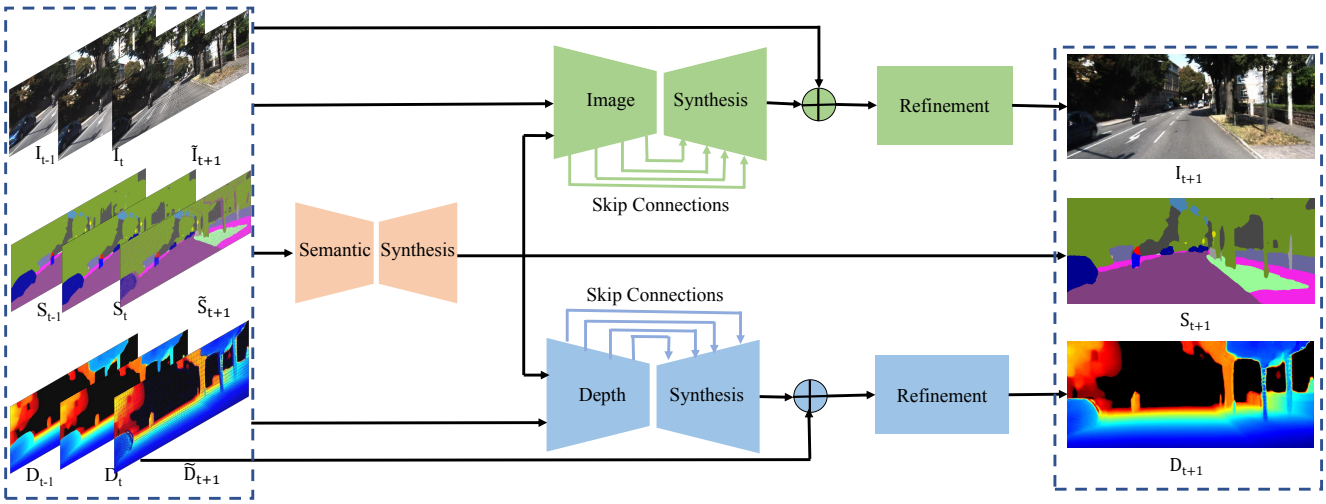

Figure 2: Refinement network. Taking as input the color images $(I_{t-1}, I_t, \tilde{I}_{t+1})$, depth maps $(D_{t-1}, D_t, \tilde{D}_{t+1})$, and semantic maps $(S_{t-1}, S_t, \tilde{S}_{t+1})$, the refinement network synthesizes image I_{t+1}, depth map D_{t+1} and semantic map S_{t+1} by refining the projected image $\tilde{I}_{t+1}$, depth $\tilde{D}_{t+1}$ and $\tilde{S}_{t+1}$.

The color and semantic information is directly copied from the previous frame. Depth is determined by the 3D point P_{t+1}. Further, depth $\{z_i^{t+1}\}$ associated with each pixel is used to determine the order of projection to handle occlusion. When two points project into the same 2D location, the point with larger depth is discarded.

4.4. Training

All modules in the motion prediction framework with ego-motion prediction, motion reconstruction, and foreground motion prediction are differentiable. Thus the whole framework can be trained in an end-to-end manner.

Note that it is hard to obtain labeled data to supervise foreground motion. To self-supervise 3D motion prediction during training, we utilize the estimated optical flow $\hat{F}_{t,t+1}$ [10] and the ground-truth depth $\hat{D}_{t+1}$ to penalize incorrect prediction on $\tilde{D}_{t+1}$ and $F_{t,t+1}$. The predicted depth map $\tilde{D}_{t+1}$ in Figure 1 is incomplete. We thus use V_D^{t+1}, a binary mask, to represent pixels with depth. The loss function L_M for training this framework is

$$L_M = \tilde{L}_F + \tilde{L}_D, \tag{7}$$

where $\tilde{L}_F$ and $\tilde{L}_D$ are the loss functions for optical flow and depth respectively. They are expressed as

$$\begin{aligned} \tilde{L}_F &= \sum_i ||F_{t,t+1}(u_i, v_i) - \hat{F}_{t,t+1}(u_i, v_i)||_1, \\ \tilde{L}_D &= \sum_i ||\tilde{D}_{t+1}(u_i, v_i) - \hat{D}_{t+1}(u_i, v_i)||_1 V_D^{t+1}(u_i, v_i). \end{aligned} \tag{8}$$

By combing $\tilde{L}_F$ and $\tilde{L}_D$, training of the 3D motion prediction network is well constrained. It can learn valid physical movement of the camera and objects in 3D.

5. Refinement Network

The refinement network is visualized in Figure 2. The semantic map is updated first, which is then utilized as guidance to facilitate updating of depth map D_{t+1} and RGB image I_{t+1}. The predicted semantic map provides category specific information beneficial to color image and depth map prediction. This framework utilizes the auxiliary information from multiple tasks for future video prediction.

The refinement network consists of three encoder-decoders as sub-networks for predicting semantics, color, and depth respectively. The encoder-decoders for image and depth synthesis are trained to learn the difference between $\tilde{I}_{t+1}, \tilde{D}_{t+1}$ and the ground truth. We add a refinement module of three convolution layers with ReLU and layer normalization to produce the final results.

5.1. Training

The refinement network is trained in an end-to-end manner supervised by task specific targets. The overall loss function L_C for this network is defined as

$$\begin{aligned} L_C &= L_I + L_S + L_D, \\ L_I &= \sum_{i=1}^{HXW} ||I_{t+1}(u_i, v_i) - \hat{I}_{t+1}(u_i, v_i)||_1, \\ L_D &= \sum_{i=1}^{H\times W} ||D_{t+1}(u_i, v_i) - \hat{D}_{t+1}(u_i, v_i)||_1, \\ L_S &= \sum_{i=1}^{H\times W} \sum_{k=1}^{K} -\hat{S}_{t+1}(u_i, v_i, k) \log S_{t+1}^p(u_i, v_i, k), \end{aligned} \tag{9}$$

where L_I, L_S, and L_D are task-specific loss functions for color images, semantics and depth maps. H and W are im-

age spatial sizes. K is the number of categories for semantic segmentation. $\hat{I}_{t+1}$, $\hat{S}_{t+1}$, and $\hat{D}_{t+1}$ are ground-truth for image, semantic and depth respectively.

6. Experiments

Dataset. We conduct experiments on the KITTI dataset [9] and the scene flow driving dataset [20]. The KITTI dataset contains 375x1242-resolution stereo image sequences for driving scenes captured at 10FPS. The dense depth maps are generated with the stereo matching approach CRL [22]. The optical flow fields are derived with FlowNet 2.0 [10]. We obtain semantic segmentation by fine-tuning the method of [35] on KITTI semantic segmentation dataset.

The KITTI dataset for our training and evaluation includes totally 29 video sequences (with 5k frames). We randomly select 4 sequences (1.7k frames) for evaluation. Hyper-parameters in experiments are tuned on the training set. We note that the depth and semantic maps are not perfect as they are generated by existing algorithms. We also evaluate the method on the Driving dataset [20] with synthetic videos of perfect depth maps and optical flow fields without segmentation information. We train our model on the first 600 frames and test on the remaining 200 frames. The frame resolution in the Driving dataset is 540x960.

Implementation details. Our whole model is implemented with Tensorflow 1.2.1 [1]. For all networks, the batch size is set to 1 with 50 epochs for training. Our learning rate is $1e-4$ in the first 10 epochs and $1e-5$ for others. In all experiments, our model takes two frames as input and outputs one or multiple future frames.

Evaluation metrics. We evaluate our model, baselines, and prior work using several metrics measuring the accuracy of motion fields, video frames, depth maps, and semantic segmentation in the future. Predicted motion fields are measured using the average endpoint error (EPE) [3]. We also evaluate predicted camera poses by comparison against the ground-truth odometry. The translation components in camera poses are measured with the root mean square error (RMSE) [25]; the rotation components are evaluated with the relative angle error (RAE) [25]. Semantic segmentation is evaluated with mean intersection-over-union (IoU) [5]. Depth maps are evaluated in terms of the mean absolute error (MAE) [27] and the mean absolute error of the inverse depth (iMAE) [27]. Future video frames are evaluated using Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) index [34].

Baselines. To evaluate our 3D motion decomposition framework for future prediction, we compare our model with the following baselines where the first seven are variants of our model.

- "Copy the previous frame" (Copy): The next-frame optical flow $F_{t,t+1}$ is copied from previous motion field $F_{t-1,t}$. The image, depth map, and semantic segmentation in frame $t+1$ are directly copied from frame t. It is a simple baseline assuming static future.

- "Warp the previous frame" (Warp): We replace our 3D motion decomposition module with optical flow $F_{t-1,t}$. We obtain a warped optical flow $F_{t,t+1}$ by warping $F_{t-1,t}$. $F_{t,t+1}$ is then used to generate intermediate image $\tilde{I}_{t+1}$, depth map $\tilde{D}_{t+1}$, and semantic segmentation $\tilde{S}_{t+1}$, which are further processed with our refinement network to generate final results. This baseline verifies that recurrently warping the optical flow is not sufficient to model future motion.

- "2D optical flow prediction" (Pred2D): We replace our 3D motion synthesis network with 2D optical flow prediction network. The network takes as input images (I_{t-1}, I_t), semantic segmentation maps (S_{t-1}, S_t) and depth maps (D_{t-1}, D_t) to predict the next-frame optical flow. This baseline models motion only in 2D.

- "Copy 3D motion" (Copy3D): We remove the ego-motion and foreground motion prediction modules from Figure 1(b)&(c). Also, we directly copy the ego-motion $[R|T]_{t-1,t}$ and $M_{t-1,t}$ to the next frame. To produce 3D motion $M_{t,t+1}$, $M_{t-1,t}$ is warped according to motion field $F_{t-1,t}$. This baseline aims to evaluate the necessity of camera ego-motion and foreground motion prediction in our model.

- "Directly predict 3D motion" (Pred3D): We design a network to directly predict the whole 3D motion field of the scene without motion decomposition. This baseline is to evaluate importance of our 3D motion decomposition module in Figure 1(a).

- "Without refinement" (WR): We evaluate the performance without the refinement network to evaluate the efficiency of refinement module.

- "Without joint refinement" (WJR): We optimize the refinement module fixing all other parts of the network to validate the efficiency of joint refinement strategy.

- S2S [15]: S2S is a state-of-the-art method for future semantic prediction. We finetune the released model on our dataset with the publicly available code. Four consecutive frames are used as input for S2S, in contrast to the two-frame input in our method.

- PredNet [13]: This is a previous approach to next-frame prediction. We directly adopt released code and model trained on KITTI. For multiple-frame prediction, we apply the PredNet recurrently by taking the

	Flow	Depth		Image		Seg
	EPE ↓	MAE ↓	iMAE ↓	PSNR ↑	SSIM ↑	IoU↑
S2S [15]	-	-	-	-	-	60.42
PredNet [13]	-	1.23	2.10	13.54	0.44	-
MCNet [29]	-	-	-	17.25	0.52	-
Copy	11.73	1.38	2.29	15.50	0.48	53.30
Warp	10.39	1.30	2.31	15.65	0.48	54.57
Pred2D	7.56	1.24	2.68	16.44	0.53	62.13
Pred3D	8.74	1.15	1.99	16.23	0.56	58.85
Copy3D	5.43	1.07	1.62	17.52	0.55	67.14
WR	-	-	-	14.61	0.38	57.74
WJR	-	0.87	1.41	19.78	0.65	67.38
Ours	**3.65**	**0.83**	**1.32**	**19.83**	**0.66**	**69.07**

Table 1: Next-frame prediction on the KITTI dataset. ↑ means the higher the better and ↓ is contrary. "-" means invalid field.

	Flow	Depth		Image		Seg
	EPE ↓	MAE ↓	iMAE ↓	PSNR ↑	SSIM ↑	IoU ↑
S2S [15]	-	-	-	-	-	37.31
PredNet [13]	-	3.71	5.72	12.37	0.35	-
Copy	11.88	3.25	5.38	12.36	0.36	31.85
Warp	11.51	3.32	5.67	12.48	0.35	32.67
Pred2D	8.63	3.92	7.77	12.41	0.37	37.33
Pred3D	10.56	3.09	5.38	11.99	0.38	31.87
Ours	**5.57**	**2.63**	**4.17**	**13.05**	**0.41**	**41.70**

Table 2: Qualitative results of predicting five future frames. ↑ means the higher the better and ↓ means contrary. "-" means invalid field.

prediction results in the current frame to generate the next frame prediction. We train PredNet to predict both video frames and depth maps.

- MCNet [29] : This is a state-of-the-art approach to next-frame prediction. Our method shares a similar idea with MCNet to decompose the scene into motion and content. We train and evaluate their method with the released code on our dataset.

6.1. Evaluation on KITTI Dataset

We conduct both quantitative and qualitative experiments on the KITTI dataset concerning the capability of predicting future motion, images, depth maps, and semantic segmentation. We also experiment with both next- and multiple-frame prediction.

Next-frame prediction. Quantitative comparison between our approach and the baselines are shown in Table 1. In terms of all the metrics, our method consistently outperforms the baselines. Compared with Pred2D, our method achieves a much lower EPE, *i.e.* 3.65 vs 7.56. This demonstrates that our 3D motion prediction framework can predict more accurate optical flow compared to 2D-based solutions. Our approach outperforms Pred3D with more accurate future 3D motion.

	RMSE ↓	RAE ↓
Copy	0.483	0.024
Ours	**0.380**	**0.013**

Table 3: Next frame pose evaluation on KITTI dataset. RAE means relative angle error for the rotation component. RMSE represents root mean square error for the translation component. ↓ means the lower the better.

	Flow	Depth		Image	
	EPE↓	MAE↓	iMAE↓	PSNR↑	SSIM↑
Copy	20.16	6.39	3.21	17.58	0.62
Warp	9.56	6.06	3.75	17.45	0.63
Pred2D	5.47	14.70	5.55	17.22	0.63
Pred3D	6.43	3.14	3.11	18.48	0.67
Ours	**1.87**	**1.88**	**1.27**	**22.08**	**0.77**

Table 4: Qualitative results on Driving dataset for next frame prediction. ↑ means the higher the better, and ↓ means the lower the better.

Our method also works better than PredNet in terms of image and depth prediction, and better than S2S regarding semantic prediction. Further, our approach performs better than MCNet in synthesizing future frames in terms of both PSNR and SSIM. More importantly, our method achieves more accurate future depth prediction than 2D-based baselines such as Pred2D, manifesting that a 3D-based model can potentially capture more complete geometry of the scene for future depth. In addition, we also evaluate our model regarding the refinement module (WR) and the joint refinement strategy (WJR). The refinement module improves results by filling holes and harmonizing overall appearance. Joint refinement is also helpful.

Visual comparisons are shown in Figure 3. Compared with MCNet and Pred2D, our method preserves higher quality structure of objects. Our results also do not contain the blur visual artifacts that are however noticed in others. More qualitative comparisons are contained in the supplementary material.

Compared with the segmentation prediction results of S2S [15] and Pred2D, our results retain small and thin objects in segmentation prediction. For example, the pole is left out in the segmentation results of S2S and Pred2D. Without motion decomposition, Pred3D does not distinguish between camera and moving-object motion. It makes

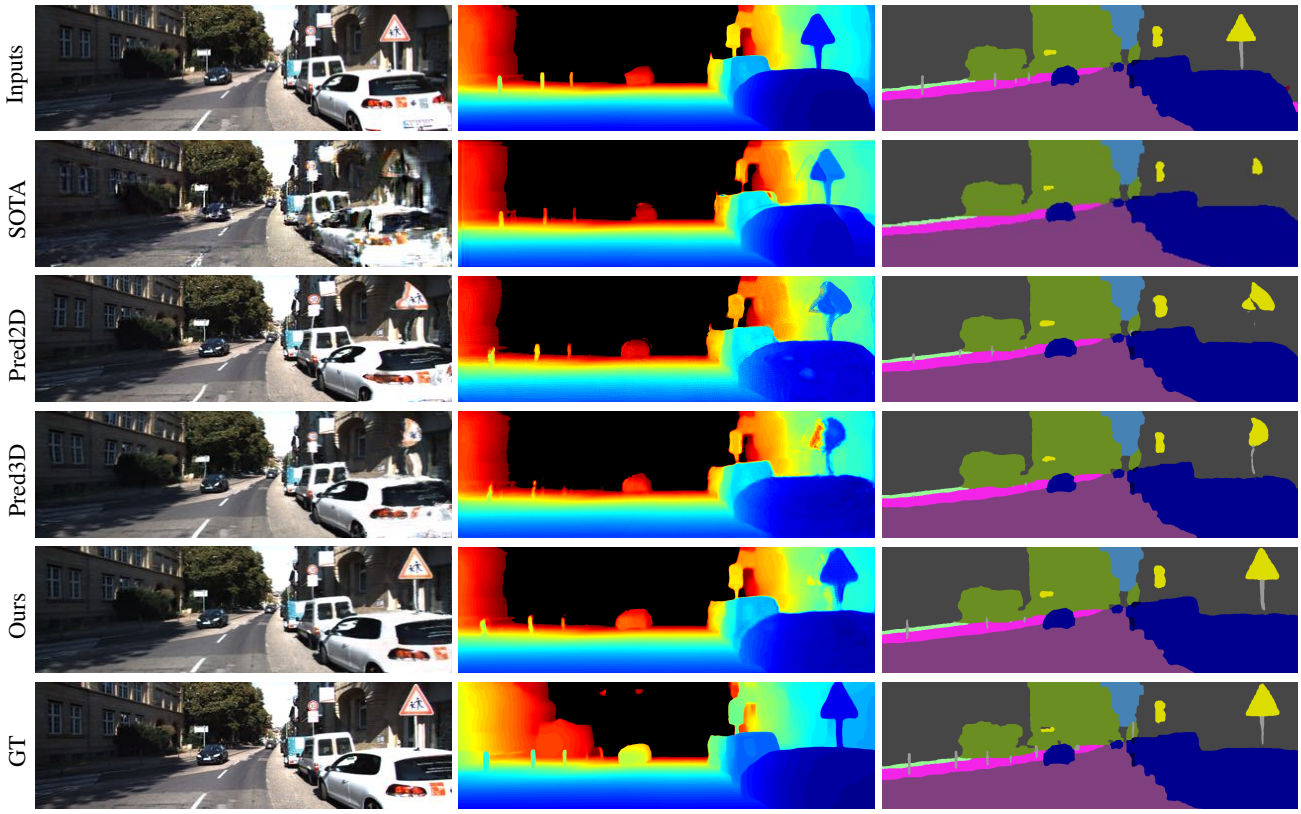

Figure 3: Visualization of different methods on next-frame prediction on the KITTI dataset. Input images are at time t. In the second row, the image is produced by MCNet [29] and depth map is produced by PredNet [13] while the segmentation map is from S2S [15].

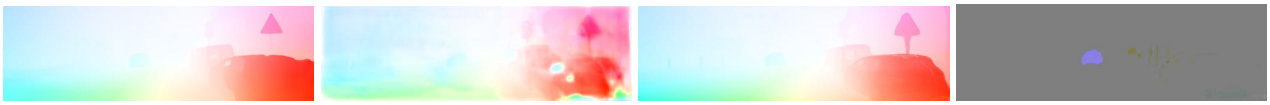

Figure 4: Future motion prediction results. The 3D motion is color coded where R-G-B corresponds to movement in x-y-z directions respectively. In this case, the car is moving closer, corresponding to the example shown in Figure 3.

static objects not well regularized and possibly generate undesired effect (*e.g.* the static traffic sign is distorted). In comparison, our results are closer to the next-frame ground truth (the fifth row in Figure 3) while the baselines fail on large motion regions (*e.g.* the nearest white car).

Evaluation of our predicted camera poses is listed in Table 3. Compared with directly copying from $[R|T]_{t-1,t}$, our ego-motion prediction module reduces the mean angle error (RAE) by approximately 50%. Our approach also improves the translation metric RMSE by over 20%, which demonstrates that our self-supervised framework for ego-motion prediction can predict accurate future camera poses without ground-truth for supervision.

We show the predicted next-frame motion produced by our approach in Figure 4. Compared with the motion field produced by Pred2D, our results are more natural regarding *e.g.* the shape of cars. Visualization of moving-object motion is shown in Figure 4 where the car moves forward. Our method generates 3D movement without 3D supervision.

Multiple-frame prediction. We compare our approach with baselines on generating multiple future frames (5 frames on KITTI). Note that the frame rate of the KITTI dataset is 10FPS and it contains large motion between frames, which makes KITTI challenging to predict multiple steps ahead. For all the approaches evaluated, we repeat them to produce multiple future frames. We show qualitative comparisons in Table 2. Our method outperforms all the baselines regarding all the metrics. Our 3D motion decomposition model facilitates long-term future prediction.

Qualitative comparison of generating 5 future frames is

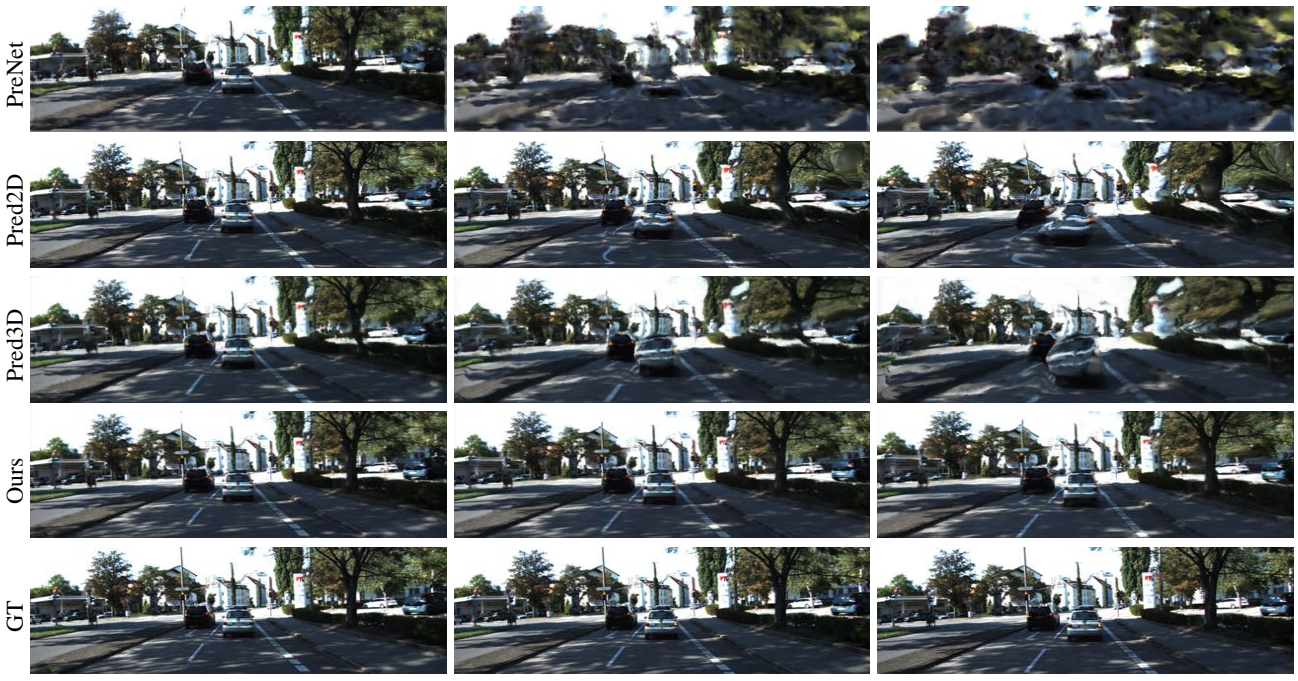

Figure 5: Results of predicting multiple frames. Depth and segmentation are provided in the supplement.

Figure 6: Visualization of our results on the Driving dataset for next frame prediction. "GT" stands for ground truth.

shown in Figure 5. In the video produced by PredNet [13], the video frames are blurry. Similarly, in the results of Pred3D, objects are distorted. In contrast, our results preserve the global structure of the scene and details of the objects. More results on multi-frame prediction are shown in the supplement.

6.2. Evaluation on Driving Dataset

Driving dataset does not have segmentation annotation. Therefore we train a deep neural network to produce moving-object masks. We obtain the ground-truth masks by the unsupervised motion segmentation method [21]. We replace the semantic segmentation by estimated moving-object masks in our model. The refinement network is modified to update the color images and depth maps.

Quantitative results are listed in Table 4. Our method outperforms all the baselines on all the metrics. We demonstrate that our method achieves competitive performance even without semantic segmentation as input. Our approach is applicable to RGBD videos without the need of semantic segmentation. Visual illustrations are shown in Figure 6.

7. Conclusion

We have presented a 3D motion decomposition model for future RGBD dynamic scene synthesis. Our method predicts future scenes by first modeling scene dynamics into camera motion and moving-object motion. We forecast future ego-motion and object motion separately to avoid influence between them. We then integrate the two motion fields for future scene synthesis. In our extensive experiments, we have demonstrated that 3D motion decomposition is effective for future prediction. We believe our work shows a new and promising direction for future scene prediction.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, 2016. 5

[2] S. Aigner and M. Körner. Futuregan: Anticipating the future frames of video sequences using spatio-temporal 3d convolutions in progressively growing autoencoder gans. *arXiv*, 2018. 1

[3] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *IJCV*, 2011. 5

[4] W. Byeon, Q. Wang, R. K. Srivastava, P. Koumoutsakos, P. Vlachas, Z. Wan, T. Sapsis, F. Raue, S. Palacio, T. Breuel, et al. Contextvp: Fully context-aware video prediction. In *ECCV*, 2018. 1

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 5

[6] E. L. Denton et al. Unsupervised learning of disentangled representations from video. In *NIPS*, 2017. 1

[7] C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In *NIPS*, 2016. 1

[8] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 2013. 1

[9] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 5

[10] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017. 3, 4, 5, 7

[11] X. Jin, H. Xiao, X. Shen, J. Yang, Z. Lin, Y. Chen, Z. Jie, J. Feng, and S. Yan. Predicting scene parsing and motion dynamics in the future. In *NIPS*, 2017. 1, 2

[12] X. Liang, L. Lee, W. Dai, and E. P. Xing. Dual motion gan for future-flow embedded video prediction. In *ICCV*, 2017. 1, 2

[13] W. Lotter, G. Kreiman, and D. Cox. Deep predictive coding networks for video prediction and unsupervised learning. In *ICLR*, 2017. 1, 5, 6, 7, 8

[14] P. Luc, C. Couprie, Y. Lecun, and J. Verbeek. Predicting future instance segmentation by forecasting convolutional features. In *ECCV*, 2018. 1

[15] P. Luc, N. Neverova, C. Couprie, J. Verbeek, and Y. LeCun. Predicting prednetinto the future of semantic segmentation. In *ICCV*, 2017. 1, 5, 6, 7

[16] Z. Luo, B. Peng, D.-A. Huang, A. Alahi, and L. Fei-Fei. Unsupervised learning of long-term motion dynamics for videos. In *CVPR*, 2017. 1, 2

[17] R. Mahjourian, M. Wicke, and A. Angelova. Geometry-based next frame prediction from monocular video. In *IVS*, 2017. 1

[18] R. Mahjourian, M. Wicke, and A. Angelova. Geometry-based next frame prediction from monocular video. In *IV*, 2017. 2

[19] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv*, 2015. 1

[20] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016. 1, 5

[21] P. Ochs, J. Malik, and T. Brox. Segmentation of moving objects by long term video analysis. *TPAMI*, 2014. 8

[22] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. In *ICCV*, 2017. 5

[23] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro. Sdc-net: Video prediction using spatially-displaced convolution. In *ECCV*, 2018. 2

[24] O. Sorkine-Hornung and M. Rabinovich. Least-squares rigid motion using svd. *Technical notes*, 2017. 3

[25] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IROS*, 2012. 5

[26] A. M. Terwilliger, G. Brazil, and X. Liu. Recurrent flow-guided semantic forecasting. *arXiv*, 2018. 2

[27] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger. Sparsity invariant cnns. In *3DV*, 2017. 5

[28] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv*, 2017. 2

[29] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee. Decomposing motion and content for natural video sequence prediction. *arXiv*, 2017. 1, 2, 6, 7

[30] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee. Learning to generate long-term future via hierarchical prediction. In *ICML*, 2017. 1

[31] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. In *NIPS*, 2016. 1

[32] J. Walker, C. Doersch, A. Gupta, and M. Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *ECCV*, 2016. 2

[33] Y. Wang, M. Long, J. Wang, Z. Gao, and S. Y. Philip. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In *NIPS*, 2017. 1

[34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004. 5

[35] Z. Wu, C. Shen, and A. Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *PR*, 90:119–133, 2019. 5

[36] L. Xu and J. Jia. Stereo matching: An outlier confidence approach. In *ECCV*, 2008. 2

[37] T. Xue, J. Wu, K. Bouman, and B. Freeman. Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In *NIPS*, 2016. 1

[38] Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018. 2

[39] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, pages 1851–1858, 2017. 2