Visuomotor Understanding for Representation Learning of Driving Scenes

Seokju Lee[†1]
seokju91@gmail.com

Junsik Kim[1]

Tae-Hyun Oh[2]

Yongseop Jeong[1]

Donggeun Yoo[3]

Stephen Lin[4]

In So Kweon[1]

[1] KAIST
Daejeon, Korea

[2] MIT CSAIL
Cambridge, MA, USA

[3] Lunit Inc.
Seoul, Korea

[4] Microsoft Research Asia
Beijing, China

Abstract

Dashboard cameras capture a tremendous amount of driving scene video each day. These videos are purposefully coupled with vehicle sensing data, such as from the speedometer and inertial sensors, providing an additional sensing modality for free. In this work, we leverage the large-scale unlabeled yet naturally paired data for visual representation learning in the driving scenario. A representation is learned in an end-to-end self-supervised framework for predicting dense optical flow from a single frame with paired sensing data. We postulate that success on this task requires the network to learn semantic and geometric knowledge in the ego-centric view. For example, forecasting a future view to be seen from a moving vehicle requires an understanding of scene depth, scale, and movement of objects. We demonstrate that our learned representation can benefit other tasks that require detailed scene understanding and outperforms competing unsupervised representations on semantic segmentation.

1 Introduction

An essential capability for intelligent vehicles is understanding causal relationships between its motion and the surrounding environment. Knowing how its movement affects what it would see around it can aid the vehicle in selecting safe and proper courses of action.

The ability to synchronize visual information with physical movement is commonly referred to as *visuomotor understanding*. For humans, this understanding is critical for daily functioning, as 80% of human perception depends on vision, and most sensory decision-making is aimed toward movement [12]. This coordination of fine motor skills with visual stimuli is developed from infancy with basic movements such as toddling and eventually improves to perform more complex tasks like buttoning shirts and tying shoelaces [18].

[†]Part of this work was done while S. Lee was at Microsoft Research Asia.

Motivated by the human perception system, we develop an unsupervised framework for developing visuomotor understanding in driving scenes from paired visual and ego-motion sensory information. One of our main goals is to *learn a visual representation* by predicting future frames via dense motion fields from fused visual and ego-motion data. We argue that for effective inference in this task, the model needs to learn semantic and geometric knowledge with respect to the ego-centric viewpoint. Specifically, forecasting future frame appearance driven by motion requires comprehensive understanding of scene depth, object scale, and movements of dynamic objects.

Towards this goal, we propose a novel deep network that takes as input a single frame together with the corresponding motion sensor data, and estimates dense optical flow for predicting the appearance of the next frame. The motion sensor data is concatenated with the encoded visual features after undergoing a learned embedding into a latent space. The predicted flow is used to warp the input frame forward by one-time step, and the training loss is defined based on the difference between the warped image and the actual next frame. A key property of the proposed method is its *time reversal symmetry (T-symmetry)* [45]. Our work takes the physical variables of *velocity* and *angular momentum* which are affected by time reversal that can be used as control inputs in the network and also to introduce additional self-supervision as described in Sec. 3.4. For training, we have collected large-scale pairs of image and motion data by simply driving a vehicle equipped with a camera and a mobile sensor that measures global speed and inertia. After large-scale training with the proposed framework, we finetune our model on a semantic segmentation task with a public dataset to verify its transferability.

Contributions   To sum up, the main contributions of this work are as follows.

1. A generic sensor fusion architecture that predicts dense optical flow for synthesizing future or past frames with the help of motion sensor data and time reversal symmetry. The effectiveness of these components is validated by extensive ablation studies.

2. A visual representation learned by our method is shown to be effective for semantic segmentation in the autonomous driving scenario and useful for other vision applications.

2   Related Works

Visual representation learning   Many previous works [1, 26, 32, 34, 35, 37, 61] for unsupervised visual representation learning have aimed to acquire high-level understanding within a single-image context. Beyond the scope of a single image, several recent works have leveraged an additional dimension of data, such as temporal sequences [22, 53, 58, 54, 63] and multi-modal input [1, 20]. Our work lies in the direction of multi-modal based representation learning, specifically utilizing motor sensor and visual information in a collaborative fashion.

Learning general visual representations from multi-modal data has been addressed in the context of driving scenes [1, 20]. Agrawal *et al*. [1] learn a representation for predicting the camera transformation between a pair of input images, with recorded ego-motion as self-supervision. Given pairs of images and the direction of motion between them, Jayaraman and Grauman [20] acquire an equivariant representation, where the relative positions in the feature space of two images can be predicted by the motion direction between them. Compared to these methods, our work aims to learn a representation with stronger knowledge of scene structure. As Sax *et al*. [47] studied, robotic locomotive tasks, *e.g.*, visual exploration or navigation, require understanding of mid-level visual features [59]. Learning to predict the change in viewed scene appearance with respect to ego-motion requires more detailed

understanding of scene geometry, including occlusions and disocclusions from camera motion, than what is needed to estimate camera pose change between a pair of images [1] or relative feature space displacements [20]. We demonstrate that our learned representation is more effective than these approaches on important driving-related tasks that benefit from structural scene understanding, such as semantic segmentation.

Learning view synthesis   Visuomotor understanding involves the ability to predict changes in frame appearance that accompany camera motion. We use this view synthesis problem as a *proxy task* for learning a visual representation. In other works on view synthesis, Kulkarni *et al.* [25] and Yang *et al.* [59] disentangled latent pose factors of an image, limited to rotations of simple objects such as faces or chairs. View interpolation [9, 21] and extrapolation [64] methods synthesize high-quality novel views, but require more than two input frames. Tatarchenko *et al.* [52] proposed an encoder-decoder network to directly regress the pixels of a new image from a single input image, but tends to produce blurry results. Zhou *et al.* [62] alleviated this problem through a flow-based sampling approach called *appearance flow*, but this often generates artifacts due to warped scene structure. Recently, Liu *et al.* [27] exploited 3D geometry to synthesize a novel view using depth labels, and Park *et al.* [36] and Sun *et al.* [51] jointly trained flow-based pixel generation networks, but these works are geared toward a specific application, rather than learning a visual representation that can be used for various semantic understanding tasks.

Learning optical flow   Estimating optical flow formally requires at least two input images. Although Pintea *et al.* [40] proposed a method for single-image flow prediction, they dealt only with human actions. Obtaining optical flow between two images is a well-studied computer vision problem [4, 43, 49]. Several recent works have proposed CNN-based supervised learning methods [8, 14, 31, 50, 56] with ground truth flow, and unsupervised learning methods [2, 19, 32, 42] with unlabeled pairs of images. However, these approaches for optical flow are not suitable for learning a general semantic representation, because they focus on learning to match local areas between two images, which does not require holistic scene understanding and semantic knowledge.

This difference between our work and existing flow estimation methods can be further explained as follows. Flow estimation with two sequential images, I_t and I_{t+1}, is formulated as $F_{t,t+1} = \mathcal{F}(I_t, I_{t+1})$, where $\mathcal{F}$ is a conventional model for estimating optical flow. By contrast, our newly proposed flow prediction method with motion sensor modality, S_t, can be represented as $F_{t,t+1} = \widetilde{\mathcal{F}}(I_t, S_t)$, where $\widetilde{\mathcal{F}}$ is our model, called SensorFlow. While the function $\mathcal{F}(\cdot)$ is learned from how to match the two images photometrically, our function $\widetilde{\mathcal{F}}(\cdot)$ does not learn such a comparator, as only a single image is given. By learning our function with respect to a static scene image and a physical motion S_t, it is forced to learn a representation based on structural and semantic understanding, rather than a representation targeted at local matching.

3 SensorFlow Architecture

Our objective is to train a non-linear mapping to predict optical flow given an RGB image and synchronized sensor data. In this section, we introduce the SensorFlow architecture to achieve this goal. Given an RGB image, the network estimates optical flow, of which the direction is controlled by the input sensor data. Further, we describe how sensor values are used as control parameters and fused with the visual representation, and explain the loss functions used to train the network.
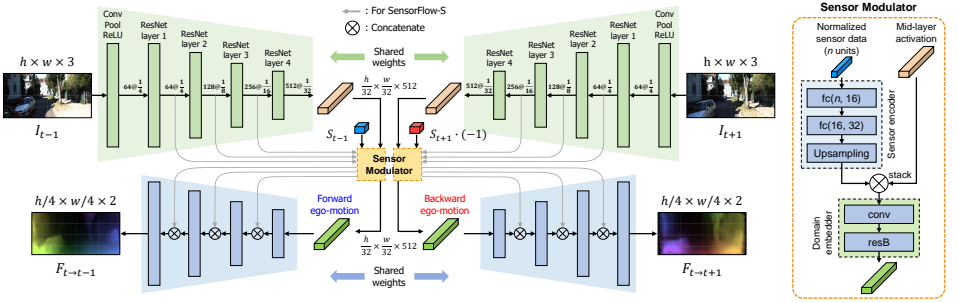
Figure 1: Illustration of SensorFlow architecture and its sensor modulator. The base encoder here is a ResNet. The network is trained on image data (I) and sensor data (S) collected from a vehicle while driving. The sensor modulator controls the direction of the flow by encoding the sensor data into the visual domain (fc: fully-connected layer, conv: convolutional layer, resB: residual block). A natural causal relationship exists between this vehicle data and flow fields (F). Leveraging this relationship, our network learns to predict the current frame ($\hat{I}_t$) from a past frame (I_{t-1}) or a future frame (I_{t+1}) given sensor data that is embedded in the latent space. By increasing its visuomotor understanding in this manner, our network learns a visual representation built on semantic and geometric knowledge of driving scenes.

3.1 Basic Architecture

We designed a simple and novel network that utilizes motion sensor information to learn the relationship between ego-motion and changes in scene appearance while learning a versatile visual representation. The basic architecture of SensorFlow is illustrated in Figure 1.

Let us first focus on the left side of the architecture in Figure 1. SensorFlow consists of an encoding part to extract visual features from an image and a decoding part to decode the features into optical flow. Our ultimate goal of training SensorFlow is to obtain an encoder network which can be reused for various recognition tasks such as semantic segmentation in driving scenes. To this end, we design the encoder to be compatible with any general-purpose network architecture such as AlexNet [24], VGG [43] or ResNet [57].

For the decoding part, we stack multiple deconvolution layers to upsample the feature map as done in [8]. We also employ skip connections as in [44] where the intermediate feature maps of the encoder are passed to the decoder to enhance fine detail in the output. To accommodate various backbone architectures, the skip connections are applied in a layer-symmetric manner. This version of SensorFlow containing these skip connections is denoted as SensorFlow-S.

Let us take a look at both the left and right sides of the architecture in Figure 1. To implement the idea of *T-symmetry*, we design the network as two streams so that it can learn to predict bidirectional flows simultaneously for the forward and backward motions. Specifically, given three temporally consecutive images I_1, I_2 and I_3, the left side of the network generates a flow map $F_{2\rightarrow1}$ from I_1, and the right side of the network generates a flow map $F_{2\rightarrow3}$ from I_3. Details on this are given in the following sections. For further details on the architecture, readers can refer to the supplementary material.

3.2 Learning Sensor Representations

Predicting optical flow or a neighboring frame from a single image is an ill-posed problem. However, with information about camera motion under an assumption that the surrounding environment is static, we can predict the global flow of the scene structure. Here, our goal is

to estimate a fine flow map from only a single image frame and paired sensor values, with performance comparable to two-view flow methods.

Given a single image $I_t : X \to \mathbb{R}^3$ at time t, we define forward and backward sensor data S_t^+ and S_t^- as follows:

$$S_t^+ = [s_1 \; s_2 \; s_3 \; \cdots \; s_n]^\top, \;\; S_t^- = -[s_1 \; s_2 \; s_3 \; \cdots \; s_n]^\top, \tag{1}$$

where n denotes the number of sensor measurements. Given the inputs I_t and S_t, the predicted flow maps, F, and generated image frames, $\hat{I}$, of both forward and backward motions are represented as follows:

$$F_{t+1 \to t} = f_{\text{flow}}(I_t, S_t^+), \;\; F_{t-1 \to t} = f_{\text{flow}}(I_t, S_t^-), \tag{2}$$

$$\hat{I}_{t+1}^f = f_{\text{warp}}(I_t, F_{t+1 \to t}), \;\; \hat{I}_{t-1}^b = f_{\text{warp}}(I_t, F_{t-1 \to t}), \tag{3}$$

where f_{flow} is the function for flow prediction and f_{warp} is the function for image warping using a differentiable grid sampling layer proposed by Jaderberg *et al.* [17]. Note that the grid sampling layer is used to transform an image in the reverse direction of the flow.

3.3 Sensor Modulator

A key element of SensorFlow is a proposed sensor modulator that can encode a vector of sensor signals into the visual domain. The sensor modulator receives two inputs, normalized sensor data and a mid-layer activation. For sensor data preprocessing, we perform normalization by obtaining the mean and standard deviation over the entire training set for each of the n sensor units. At each time step, both forward and backward data are processed concurrently in training, and the average sensor value for each unit between the two directions is zero, even after normalization. As discussed later for T-symmetry, this property will be utilized for regularization. Another input, a mid-layer activation, is the neural output from an encoding layer. For the basic SensorFlow model without the skip-connection structure, this is the final output, which is a latent variable of the encoder.

The sensor modulator is divided into two parts: a sensor encoder and a domain embedder. Figure 1 shows the structure of our sensor modulator. First, the sensor encoder transforms the sensor values into the visual domain. This is done via two fully-connected layers that extend the channel size, and an upsampling layer that expands the spatial size to the same resolution as the mid-layer activations. This expansion is achieved by repeating the same 1×1 vector to a size of $h \times w$. In SensorFlow-S, the weights of the sensor encoder are shared for all mid-layer activations. Second, the domain embedder stacks the sensor feature plane with the mid-layer activation plane and converts them into a common domain via one convolutional layer and one residual block [13]. As a design note, the sensor modulator does not include any normalization layer (*e.g.* batch normalization [16], local response normalization), as the neurons must preserve the scale of the motions. Each convolutional layer is followed by a ReLU. The generated encoding contains visual information as well as information on the direction and scale of the motion.

3.4 Self-Supervised Loss

Similar to the loss in [19], we use an unsupervised loss that measures the photometric inconsistency between I and $\hat{I}$. Since the photometric loss does not reflect the movement of dynamic objects or non-rigid motions, we apply the structural similarity index SSIM [55] to

mitigate the effects of this movement. Our basic image warping cost with forward motion is written as

$$\mathcal{L}_w(\hat{I}^f, \mathbf{M}^f) = \sum_{x \in X} \left\{ \lambda_1 \rho \left(\mathbf{M}^f(x) \cdot \|I(x) - \hat{I}^f(x)\|_1 \right) + \lambda_2 \left(1 - SSIM(I(x), \hat{I}^f(x)) \right) \right\}, \quad (4)$$

where x indicates each pixel location and $\rho(x) = (x^2 + \varepsilon^2)^\alpha$ is the robust generalized Char-bonnier penalty function [49] with $\alpha = 0.4$. This function is equal to the original Charbonnier penalty when $\alpha = 0.5$, which is a differentiable variant of the absolute function. λ_1 and λ_2 are set to 0.3 and 0.7 respectively. In order to exclude invalid gradients from occluded or exiting regions, we follow [52] by setting the forward valid mask $\mathbf{M}^f(x)$ to be 1 if the condition

$$\left| F^f(x) + F^b\left(x + F^f(x)\right) \right|^2 > \gamma_1 \left(\left| F^f(x) \right|^2 + \left| F^b\left(x + F^f(x)\right) \right|^2 \right) + \gamma_2, \quad (5)$$

is satisfied, and 0 otherwise. We set $\gamma_1 = 0.01$, $\gamma_2 = 0.5$. For the backward valid mask $\mathbf{M}^b(x)$, we exchange F^f and F^b in the above condition. Each forward and backward flow, F^f and F^b, is processed on two consecutive frames, i.e., $\mathbf{M}^f(x)$ by $\{I_{t-1}, I_t\}$ and $\mathbf{M}^b(x)$ by $\{I_{t+1}, I_t\}$.

    To regularize the bidirectional training, we design a forward and backward flow consistency check in our learning scheme. This consistency check is based on the observation that within a short time interval, the flow of rigid objects generated by camera ego-motion can be linearly modeled [15], such that incremental flows in the forward and backward directions should sum to zero. Previous works [11, 60] utilized a related idea in their depth prediction frameworks with a geometric consistency loss. We exclude both forward and backward occluded regions from the consistency check. Specifically, our bidirectional flow consistency cost is imposed as

$$\mathcal{L}_c(F^f_{t \to t-1}, F^b_{t \to t+1}, \mathbf{M}^f, \mathbf{M}^b) = \sum_{x \in X} \mathbf{M}^f(x) \cdot \mathbf{M}^b(x) \cdot \left(F^f_{t \to t-1}(x) + F^b_{t \to t+1}(x) \right), \quad (6)$$

where each non-occluded pixel x is enforced to have consistent flow magnitudes between its bidirectional motions.

    As done in previous methods [8, 19], we adopt a smoothness cost, $\mathcal{L}_s$. The smoothness term is used to suppress spatial fluctuations. We have empirically found that a relatively small loss weight for the smoothness term improves flow prediction.

    To sum up, our final self-supervised loss is defined as

$$\mathcal{L}_{tot} = \lambda_w \left(\mathcal{L}_w(\hat{I}^f, \mathbf{M}^f) + \mathcal{L}_w(\hat{I}^b, \mathbf{M}^b) \right) + \lambda_s \left(\mathcal{L}_s(F^f_{t \to t-1}) + \mathcal{L}_s(F^b_{t \to t+1}) \right)$$
$$+ \lambda_c \mathcal{L}_c(F^f_{t \to t-1}, F^b_{t \to t+1}, \mathbf{M}^f, \mathbf{M}^b), \quad (7)$$

where λ denotes loss weights. We set $\lambda_w = \lambda_c = 1$ and $\lambda_s = 0.1$. The total loss is measured in a bidirectional manner with warped forward and backward images.

4   Experiments

4.1   Training

Our dataset   For the representation learning of driving scenes, we collected a large-scale set of paired image and motion data from driving a vehicle equipped with a camera and a mobile sensor that measures global speed and various inertial quantities. Nearly 350,000

Table 1: SensorFlow ablations on KITTI 2012 optical flow dataset. Photometric error is averaged over forward and backward view syntheses, and EPE is averaged endpoint error.

	Options	Trials									SensorFlow			SensorFlow-S		
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	1st	2nd	3rd	1st	2nd	3rd
Training	Sensor modality		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Bidirectional motion	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Flow consistency			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Skip-connection					✓								✓	✓	✓
	Horizontal flip						✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Time variation							✓	✓	✓	✓	✓	✓	✓	✓	✓
Modulator	stack		✓	✓	✓	✓	✓	✓								
	stack+conv								✓							
	stack+conv+resB									✓						
	fc(2)+stack+conv+resB										✓	✓	✓	✓	✓	✓
Units	v_x,v_y,v_z,w_x,w_y,w_z		✓	✓	✓	✓	✓	✓	✓	✓	✓			✓		
	v_x,w_x,w_y,w_z											✓			✓	
	v_x,w_z												✓			✓
	Photometric error	0.340	0.269	0.207	0.194	0.192	0.193	0.190	0.189	0.186	0.184	0.186	0.201	0.183	0.185	0.199
	EPE	24.22	16.70	15.39	14.91	14.18	14.76	14.05	14.02	13.80	13.77	13.79	15.11	13.35	13.68	14.93

frames were obtained at 10 Hz from 12 cities and 11 countryside routes by driving 757 *km* under various climate conditions. Detailed comparisons with existing driving datasets [5, 6, 10, 29, 41, 46, 53, 58] and necessity of ours are presented in the supplementary material.

Proxy task For experiments involving the proxy task, including the ablation study and view synthesis experiments, training is done using the KITTI dataset, as it provides ground truth optical flow for quantitative evaluation. The network is trained by the ADAM optimizer [23] for 350K iterations with a batch size of 20 on an Nvidia Titan X GPU and an Intel i7@3.4GHz CPU. The initial learning rate is set to 0.0002, and it is decreased by half every 100K iterations. While training, we take three consecutive frames as input to our two-stream network. The observed sensor set of each frame is $\{v_x,v_y,v_z,\omega_x,\omega_y,\omega_z\}$, where v_x and ω_x are the linear velocity and angular velocity along the x axis. Note that the sampling time, Δt, is different for each dataset (*e.g.* Ours and KITTI: $\Delta t = 100$ *ms*, Cityscapes: $\Delta t \simeq 60$ *ms*). Also, we average the sensor readings of three consecutive frames to reduce noise in the training data.

Representation learning task For experiments on representation learning, we pretrain our models using our large-scale dataset, and finetuned on the CamVid and CityScapes datasets for various architectures, namely the original AlexNet, VGG16, ResNet18, and ResNet34, using the same training techniques as in their respective works [13, 24, 48]. We start the finetuning with a learning rate of 0.0001.

The models are evaluated on the Cityscapes [6] and CamVid [4] datasets. Specifically, the evaluation uses the Cityscapes training set (3,000 images) and validation set (500 images), as well as the CamVid training set (367 images) and test set (233 images). The Cityscapes dataset contains high resolution images which requires large GPU memory when training deep networks, so we downsize these images by half for training and evaluation. It is reported that downscaled images have consistently negative effects on both training and test [6]. The gap between accuracy values found in our experiments and those previously reported in other works is mainly due to the image size difference.

4.2 Ablation Study

Design process The ultimate goal of this work is to learn a visual representation for the driving scenario through the estimation of neighboring frames. In this section, we conduct an ablation study to verify that this is accurately estimated by SensorFlow. This study comprises

three parts as shown in Table 1. The first part considers training options. The second part is on how to embed the sensor readings. Finally, we compare the performance for different sensor combinations in a driving environment. All ablation experiments are conducted by training ResNet18-based SensorFlow models on the KITTI raw dataset. The performances are compared using the average photometric error of forward and backward warping and the average endpoint error (EPE) on the KITTI 2012 optical flow dataset.

Regularization   To verify the effect of bidirectional training based on *T-symmetry*, models trained with only a forward motion, and with both forward and backward motions are compared (2nd and 3rd columns of the trials in Table 1). It was found that the model without bidirectional motion is easily biased to always predict flow with forward motion, regardless of the sensor readings. Another advantage of bidirectional training comes from the flow consistency loss as proposed in Equation 6. Ablations without and with this loss (3rd and 4th columns of the trials in Table 1) show that our bidirectional flow consistency term improves performance considerably via constraints on the opposite flow directions.

We utilize two forms of data augmentation for regularization. One is the common technique of image flipping, which yields improvements from comparison of the 4th and 6th columns of the trials in Table 1. The other is to vary the time intervals of optical flows, *e.g.*, by also generating I_3 from I_1 and I_1 from I_3 with $2 \cdot S^+$ and $2 \cdot S^-$, respectively. This leads to a significant improvement from the 6th and 7th columns of the trials in Table 1.

More descriptions on other design choices, *e.g.*, sensor embedding and controllability, are presented in our supplementary material.

4.3   View Synthesis

To demonstrate that the proxy task is effectively learned, we conduct experiments on view synthesis. We control to the sensory input to synthesize a new view from a different viewpoint. Table 2 shows that our proposed method performs favorably against the competing appearance flow techniques while accounting for the number of parameters of each model. Detailed experimental settings are given in the supplementary material due to limited space. The results indicate the effectiveness of embedding the control variables from the sensor into the continuous latent space. Note that the purpose of view synthesis is to validate whether our representations are plausibly learned to understand scene changes according to sensor inputs, rather than to generate visually pleasing results.

Furthermore, we qualitatively test our network by generating a new view and applying a stereo matching algorithm between an input and its new synthesized view, *i.e.*, single view depth estimation. This allows us to see whether our network learns plausible depth perception capability. As shown in the supplementary material, the results indicate that our model is potentially extensible to single-view depth estimation.

4.4   Applying Learned Representation to Semantic Segmentation

We examine the transferability of our learned representation to other driving tasks, by applying it to semantic segmentation in a driving environment. For this essential application in autonomous driving systems, we finetune the FCN [28] architecture and evaluate it on the CamVid and Cityscapes datasets. Four base encoders – AlexNet, VGG16, ResNet18 and ResNet34 – are used for FCN. For AlexNet, we use FCN-32s, defined in the original paper. For the VGG16, ResNet18 and ResNet34 encoders, FCN-8s is used. Table 3 shows the results in

Table 2: Photometric errors of view synthesis on KITTI with different time steps.

Method	Parameters	± One time step	± Two time step
MV3D [■]	69.3M	0.241	0.316
Appearance Flow [■]	5.5M	0.223	0.285
SensorFlow (AlexNet)	4.6M	0.191	0.239
SensorFlow (ResNet34)	29.2M	0.178	0.212
SensorFlow-S (ResNet34)	31.3M	0.173	0.204

Table 3: Mean IoU comparisons for semantic segmentation.

Dataset	CamVid		Cityscapes			
Base architecture	AlexNet	ResNet34	AlexNet	VGG16	ResNet18	ResNet34
SCRATCH	25.42	42.72	26.37	29.78	39.98	40.82
IMAGENET	33.44	50.47	36.27	49.01	54.04	56.91
MOVING [■]	25.57	–	26.64	–	–	–
EGO-MOTION [■]	21.89	–	26.03	–	–	–
COLORIZATION [■]	26.97	–	28.25	–	–	–
CONTEXT [■]	25.82	–	26.41	–	–	–
FLOW [■]	–	46.09	–	–	47.95	50.39
DEPTH [■]	–	45.11	–	–	48.76	50.76
DEPTH [■]+POSE	–	46.32	–	–	49.58	52.37
SensorFlow	30.48	49.46	29.35	36.52	52.97	54.24

*Only IMAGENET uses labeled data for pretraining.

terms of mean IoU for FCN with different base networks and different initialization methods, including random initialization from SCRATCH, ImageNet-pretrained model (IMAGENET), our approach (SensorFlow), and several other unsupervised representation learning methods. Our approach shows clear performance improvements over random initialization for AlexNet-, VGG-, and ResNet-based FCNs on both datasets, and comes close to that of supervised ImageNet in some cases, demonstrating the effectiveness of our pretrained models.

One might raise a question of whether motion information really plays an important role for representation learning. Is it insufficient to learn a representation from multiple frames using a photometric loss? FLOW [19] is an unsupervised optical flow learning method using a photometric loss. Since originally it takes two concatenated frames as input, we finetuned its base network with a random initialization for the first layer, which is replaced to handle the single-image input of semantic segmentation. The results show that learning flow through only the visual domain does not capture scene semantics while our proposed method does. We conjecture that learning pixel displacements between images depends on local pattern matching, rather than semantic scene understanding. In comparison, learning with motion data paired with visual domain data provides a better way for acquiring a semantic representation.

Furthermore, we compare with existing self-supervised representation learning methods that exploit ego-motion data [■, 20], and that utilize appearance information such as a color or context [37, 61]. Since few previous works conduct semantic segmentation as a test for representation learning, we have retrained each model, with the same experimental setup as ours. It is shown in Table 3 that our method yields significant improvements on the target task over both the motion- and appearance-driven methods on AlexNet.

Depth information has recently been shown to be useful for semantic tasks [22]. For the fair comparisons with depth-motivated representations, we validate ours with DEPTH learned on unsupervised single-image depth estimation [63], and the DEPTH trained with pose obtained from motion sensors, termed DEPTH+POSE. From the results, we confirm that inaccuracies in pose estimation lead to uncertainty at object boundaries. We note that while pose estimation from images is susceptible to low image quality, e.g., from adverse weather and saturated exposure, sensor data is insensitive to these factors and serves as a stable complementary modality. Still, with given pose values, ours achieves better performance than DEPTH+POSE. This may be explained by two reasons. First, we conjecture that constraints by geometric priors, e.g., epipolar constraint, hinder learning a generic transferable representation. Second, reconstruction losses based on depth re-projection are known to be quite noisy, as discussed in Sec. 3.3 of Mahjourian et al. [50]. They mention that this problem could be avoided by directly learning to predict the adjacent frames. Supported by the aforementioned results, our network is more stable to train and yields more favorable performance in comparison to existing learned representations for driving scenes.

5 Conclusion

In this work, we proposed a novel sensor fusion architecture that predicts a dense flow map from physical sensor readings fused with the input frame, while exploiting time symmetry for regularization. Though our system is trained to synthesize nearby frames, the visual representation it learns can be effectively transferred to other scene understanding tasks in the driving scenario. In particular, the transfer of our model to semantic segmentation yields leading results in comparison to existing representations acquired by unsupervised learning.

References

[1] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *ICCV*, 2015.

[2] Aria Ahmadi and Ioannis Patras. Unsupervised convolutional neural networks for motion estimation. In *ICIP*, 2016.

[3] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2): 88–97, 2009.

[4] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(3):500–513, 2011.

[5] Yiping Chen, Jingkang Wang, Jonathan Li, Cewu Lu, Zhipeng Luo, Han Xue, and Cheng Wang. Lidar-video driving dataset: Learning driving policies effectively. In *CVPR*, 2018.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[7] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.

[8] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015.

[9] John Flynn, Ivan Neulander, James Philbin, and Noah Snavely. Deepstereo: Learning to predict new views from the world's imagery. In *CVPR*, 2016.

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[11] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017.

[12] E Bruce Goldstein and James Brockmole. *Sensation and perception*. Cengage Learning, 2016.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[14] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *CVPR*, 2017.

[15] Sunghoon Im, Hyowon Ha, Gyeongmin Choe, Hae-Gon Jeon, Kyungdon Joo, and In So Kweon. High quality structure from small motion for rolling shutter cameras. In *ICCV*, 2015.

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[17] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015.

[18] Lorna S Jakobson, Virginia Frisk, Rachel M Knight, Andrea LS Downie, and Hilary Whyte. The relationship between periventricular brain injury and deficits in visual processing among extremely-low-birthweight (< 1000 g) children. *Journal of Pediatric Psychology*, 26(8):503–512, 2001.

[19] J Yu Jason, Adam W Harley, and Konstantinos G Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *ECCV*, 2016.

[20] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *ICCV*, 2015.

[21] Dinghuang Ji, Junghyun Kwon, Max McFarland, and Silvio Savarese. Deep view morphing. In *CVPR*, 2017.

[22] Huaizu Jiang, Erik Learned-Miller, Gustav Larsson, Michael Maire, and Greg Shakhnarovich. Self-supervised relative depth learning for urban scene understanding. In *ECCV*, 2018.

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

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[25] Tejas D Kulkarni, William F. Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *NIPS*, 2015.

[26] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *CVPR*, 2017.

[27] Miaomiao Liu, Xuming He, and Mathieu Salzmann. Geometry-aware deep network for single-image novel view synthesis. In *CVPR*, 2018.

[28] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[29] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1): 3–15, 2017.

[30] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *CVPR*, 2018.

[31] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

[32] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, 2018.

[33] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.

[34] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[35] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, 2017.

[36] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *CVPR*, 2017.

[37] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[38] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, 2017.

[39] Jonathan W Peirce. Understanding mid-level representations in visual processing. *Journal of Vision*, 2015.

[40] Silvia L Pintea, Jan C van Gemert, and Arnold WM Smeulders. Déja vu. In *ECCV*, 2014.

[41] Vasili Ramanishka, Yi-Ting Chen, Teruhisa Misu, and Kate Saenko. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *CVPR*, 2018.

[42] Zhe Ren, Junchi Yan, Bingbing Ni, Bin Liu, Xiaokang Yang, and Hongyuan Zha. Unsupervised deep learning for optical flow estimation. In *AAAI*, 2017.

[43] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015.

[44] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[45] Robert G Sachs. *The physics of time reversal*. University of Chicago Press, 1987.

[46] Eder Santana and George Hotz. Learning a driving simulator. *arXiv:1608.01230*, 2016.

[47] Alexander Sax, Bradley Emi, Amir R Zamir, Leonidas Guibas, Silvio Savarese, and Jitendra Malik. Mid-level visual representations improve generalization and sample efficiency for learning active tasks. *arXiv preprint arXiv:1812.11971*, 2018.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[49] Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014.

[50] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *CVPR*, 2018.

[51] Shao-Hua Sun, Minyoung Huh, Yuan-Hong Liao, Ning Zhang, and Joseph J Lim. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *ECCV*, 2018.

[52] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Multi-view 3d models from single images with a convolutional network. In *ECCV*, 2016.

[53] Udacity. Public driving dataset. 2017.

[54] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, 2015.

[55] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[56] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *ICCV*, 2013.

[57] Ian H Witten, Eibe Frank, Mark A Hall, and Christopher J Pal. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[58] Huazhe Xu, Yang Gao, Fisher Yu, and Trevor Darrell. End-to-end learning of driving models from large-scale video datasets. In *CVPR*, 2017.

[59] Jimei Yang, Scott E Reed, Ming-Hsuan Yang, and Honglak Lee. Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In *NIPS*, 2015.

[60] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*, 2018.

[61] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[62] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *ECCV*, 2016.

[63] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017.

[64] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018.