Learning Mixture of Domain-specific Experts via Disentangled Factors for Autonomous Driving

Inhan Kim,[1] **Joonyeong Lee,** [1] **Daijin Kim** [1]

[1] Department of Computer Science and Engineering, POSTECH, Korea
{kiminhan, joonyeonglee, dkim}@postech.ac.kr

Abstract

Since people know the important factors that affect the control based on the driving situation, they can drive safely even in diverse driving environment. To mimic these behaviors, we propose a two-stage representation learning model that initially splits the latent features as domain-specific features, which contain information in a specific domain, and domain-general features, which are consistent across all domains. Subsequently, the dynamic-object features, which contain information of dynamic objects, are disentangled from latent features using mutual information estimator. In this study, the problem in behavior cloning is divided into several domain-specific subspaces, with experts becoming specialized on each domain-specific policy. The proposed mixture of domain-specific experts (MoDE) model predicts the final control values through the cooperation of experts using a gating function. The domain-specific features are used to calculate the importance weight of the domain-specific experts, and the disentangled domain-general and dynamic-object features are applied in estimating the control values. To validate the proposed MoDE model, we conducted several experiments and achieved a higher success rate on the CARLA benchmarks under several conditions and tasks than state-of-the-art approaches.

Introduction

In the field of autonomous driving, behavior cloning is receiving interest as a promising approach to imitate the demonstrations of human drivers by mapping from camera input to control output. Starting with a basic end-to-end model (Bojarski et al. 2016) that explores the lane-keeping task (Virgo 2017), several successive deep networks (Codevilla et al. 2018; Sauer, Savinov, and Geiger 2018; Chen, Yuan, and Tomizuka 2019) have been developed to solve the task of driving to a destination in complex urban scenarios. Although methods leveraging deep networks can manipulate the ego-vehicle safely under training environments, they predict unexpected control values, particularly under unseen conditions, such as different weather and towns compared to the weather and towns present in the training data. Because the driving task is a sequential decision-making problem, a single unexpected control value can cause an accident or create an unsafe situation that requires a long time to recover to safe driving. In addition, there is still a well-known limitation that threatens the ability of generalization caused by the lack of policy experience (Chen et al. 2020) and the causal confusion (Codevilla et al. 2019). Based on recent efforts in this field, multi-task learning (Li et al. 2018; Yang et al. 2018; Kim et al. 2020), which leverage representation of closely related tasks through a joint optimization strategy is considered a good solution for increasing generalization ability under unseen conditions. Furthermore, representation learning (Xing et al. 2021), which automatically separates the related or unrelated hidden variables through a learning process in multiple source environments, can be another solution by distinguishing true causes in observed training demonstration patterns based on the supervision.

Because the disentanglement with explicit consideration of the given knowledge makes the model focus on distinguishing between essential information and neglecting, our work is concerned with the following observations in an attempt to improve the generalization ability. First, common static components such as lanes and curbs, the types and locations of which are regulated by law, may contain some helpful hints for driving because their existence ensures safety and efficiency by informing drivers about semantic information on the road. Human drivers perceive this information easily; however, the learning-based autonomous driving models need to specify this information explicitly. We adopt a concept from a previous study, latent unified state representation (LUSR) (Xing et al. 2021), to exploit the road components restricted by regulations. It disentangles domain-specific latent features and domain-general latent features from the original latent feature space. The domain-general latent factors contain information in common to all domains and can be used to predict the control values, regardless of the domain-specific variations. Second, dynamic objects appear in various shapes and colors at arbitrary positions and can directly affect the longitudinal control. We disentangle dynamic-object features from latent space using a learning strategy, which maximizes and minimizes mutual information between target latent spaces (Belghazi et al. 2018b; Sanchez, Serrurier, and Ortner 2020). We maximize the mutual information between longitudinal task-related features and observation while minimizing mutual information between the longitudinal task-related representation and the others (Sanchez, Serrurier, and Ortner 2020). Third, we can

estimate the distribution of action value under unseen conditions by combining distribution from the observed driving conditions. Let us assume that we drive for the first time on a rainy evening. Humans can drive based on the experience of clear evenings and rainy days. By applying these conceptual approaches to a mixture of expert (MoE) (Jacobs et al. 1991) algorithms, we train multiple experts to consider the same input data from different points of view.

Our key contribution areas are as follows :

- We propose a novel MoE framework using the latent space that is disentangled into three disjoint parts: domain-specific, domain-general, and dynamic-object feature space. The domain-relevant features are used to estimate the weight of the domain-specific experts, and the domain-irrelevant features are used to predict control values for generalization.
- We introduce the location offset regularization that estimates the current position from the kinetic formulation and penalizes the difference between the calculated location of the vehicle and the predicted location to leverage the useful representation contained in the related subtask.
- We achieve state-of-the-art driving performance on autonomous driving benchmarks, especially in unseen driving environments.

Related Work

End-to-end Autonomous Driving Architecture

In recent years, the end-to-end driving models have been greatly boosted by the development of deep learning techniques. Nvidia is the first to adopt convolutional neural networks (CNNs) to predict the steering angle with a single front-facing RGB camera (Bojarski et al. 2016). Based on the exploration of the CNN architecture, Codevilla et al. (Codevilla et al. 2018) proposed a conditional imitation learning (CIL) model, in which training specialized submodules for each high-level command leads to a goal point. The conditional branch-based driving model is the most popular approach for learning a model with navigation commands. Following this study, enhanced branched networks proposed by Sauel et al., (Sauer, Savinov, and Geiger 2018) and Wang et al. (Wang et al. 2019) used conditional affordance branches and sub-goal angle branches.

However, the driving model, which is optimized for training data, can make it difficult to increase the generalization ability across diverse conditions. Recently, many studies have been conducted (Codevilla et al. 2019; Chen et al. 2020; Li et al. 2018) to solve the generalization problem under unseen conditions. Our proposed network adopts the conditional branched network concept for utilizing the navigation command and representation learning to increase generalization ability for the vision-based end-to-end autonomous driving.

Disentangled Representation Learning

Depending on the task at hand, the goal of learning disentangled representations is to train the informative and uninformative factors of data variation. Many generative models (Jeon, Lee, and Kim 2018; Zeno, Kalinovskiy, and Matveev 2019; Chen et al. 2018; Jha et al. 2018) that learn disentangled representations of informative factors show successive results based on generative adversarial networks (Goodfellow et al. 2014) or variational auto-encoders (VAE) (Kingma and Welling 2013). Recently, an additional novel disentanglement approach is proposed by Sanchez et al. (Sanchez, Serrurier, and Ortner 2020) based on the neural mutual information estimator (Belghazi et al. 2018a) by splitting the representation into a shared and exclusive information of paired data. In research related to autonomous driving, Xing et al. (Xing et al. 2021) aimed to learn a latent unified state representation for different domains via cycle-consistent VAE (Jha et al. 2018). The mapping function of LUSR divides the latent space into domain-specific and domain-general spaces. The different domains correspond to different weather conditions, and the domain-general feature is common factors such as lanes on the road. In this case, the domain-general feature can be used as an informative factor to predict the domain-irrelevant control value of an autonomous vehicle.

However, the mapping function of LUSR only considers static components, while, in the driving scene, the objects can appear for a moment and then disappear dynamically. Therefore, we extend the LUSR with a mutual information estimator to disentangle dynamic object features from a given latent space.

Ensemble Learning for Autonomous Driving

Deep ensemble learning combines several individual models or branches to improve the generalization performance. The MoE, which is a well-known ensemble approach, comprises several specialized experts, where each expert tries to learn the target output on a subset of the input space. If we know in advance that a set of training conditions are divided into subsets, the model can be designed as several experts with a gating network, which assigns weights to expert networks according to the specific conditions. Ohn-Bar et al. (Ohn-Bar et al. 2020) proposed a learning situational driving (LSD) strategy, which learns situational policies with a multi-modal agent trained to mimic agents with specific behaviors. The LSD model considers the sub-driving scenarios to be the sub-policies.

However, the conditional module has several action prediction branches, which learn sub-policies that correspond to different commands, act as the scenario-specific experts. Hence, we design a mixture of experts network, where each expert serves as a domain-specific control value estimator, weighted by the importance calculated by the contribution of each domain.

Methods

Task Definition

The goal of a vision-based end-to-end autonomous driving task is to mimic the human driver by learning a model using sequential observations. Each observation ($\mathbf{o} = [I, v] \in O$) consists of tuples, a RGB image from a front-facing camera and measured speed of the ego-vehicle. In addition, a

high-level navigation command ($\mathbf{c} \in C$ = {*follow lane, go straight, turn right, and turn left*}) is provided by a global path planner to guide the vehicle to the final destination. When a vehicle enters an intersection, the command serves as a guide to indicate the direction of the destination. The command is a categorical variable that acts as a switch; this switch controls the selective activation of a prediction branch. Further details on global navigation can be found in the literature (Dosovitskiy et al. 2017; Codevilla et al. 2018).

At every time step, the driving model estimates the low-control action ($\boldsymbol{a} \in A = [-1, 1]^2$), which consists of continuous longitudinal and lateral control values, from the given observations and global paths. In addition, to train domain-specific policy, we utilize the weather conditions ($\mathbf{w} \in W$ = {*weather labels under training condition*}) that are available to set by a data collector. The training dataset is defined $D = \{(\boldsymbol{o}_i, w_i, c_i, \boldsymbol{a}_i)\}_{i=1}^{N}$, where N is the number of data. And D is divided into two types (D_{Obj}, D_{NoObj}). D_{Obj} and D_{NoObj} have the same configurations, with the exception of the inclusion/exclusion of dynamic objects.

A mapping function maps the driving scene I to the latent feature space ($\mathcal{F}^z$). In the space $\mathcal{F}^z$, important factors that directly affect driving are divided into three categories: 1) domain-general factors ($\overline{f^z}$) appearing in common as enacted in the Road Traffic Act, 2) domain-specific factors ($\widehat{f^z}$) according to the driving conditions, such as weather, and 3) dynamic object factors ($\widetilde{f^z}$), such as dynamic vehicles and pedestrians. Depending on the dataset type, $\mathcal{F}^z$ has different factors: $\mathcal{F}^z_{Obj} = [\overline{f^z}, \widehat{f^z}, \widetilde{f^z}]$ and $\mathcal{F}^z_{NoObj} = [\overline{f^z}, \widehat{f^z}]$, where $\mathcal{F}^z_{Obj}$ is the mapped space from $I \in D_{Obj}$ and $\mathcal{F}^z_{NoObj}$ is the mapped space from $I \in D_{NoObj}$.

Driving-related Factors Disentanglement

To map our supervision from raw image I to the high-level description in feature space $\mathcal{F}^z$, we first adopt the concept of latent unified state representation (LUSR) (Xing et al. 2021) to disentangle the $\overline{f^z}$ and $\widehat{f^z}$ of variation in the pair image. We sample a pair of images, I_1 and I_2, from D_{NoObj} at the same location in different weather conditions. The objective for cycle-consistent VAE to minimize is

$$L_{cyclic} = L_{forward} + L_{reverse}, \tag{1}$$

where

$$L_{forward} = -\mathbb{E}_{q_\phi(\overline{f^z}, \widehat{f^z}|I)}[log\, p_\theta(I|\overline{f^z}, \widehat{f^{z*}})] + KL(q_\phi(\overline{f^z}|I)||p(\overline{f^z}))$$

$$L_{reverse} = \mathbb{E}_{\overline{f^z} \sim p(\overline{f^z})}[||q_\phi(p_\theta(\overline{f^z}, \widehat{f^z_1})) - q_\phi(p_\theta(\overline{f^z}, \widehat{f^z_2}))||_1]$$

Here, q_ϕ and p_θ are parameterized probability function of the encoder and decoder. $\widehat{f^{z*}}$ is any domain-specific feature embedded by the encoder, whereas $\widehat{f^z_1}$ and $\widehat{f^z_2}$ represent embedded features from different domains.

Similar to the cycle-consistent VAE, the learning process is unsupervised, except for the high-level navigation command classifier and the weather classifier. In the process of embedding, the conditional VAE (Sohn, Lee, and Yan 2015) utilizes a categorical input to control the distribution of latent variables of the encoder and the output of the decoder. Because the main purpose of our method is well-refined feature-based behavior cloning, and not reconstruction of an input image, $\overline{f^z}$ is fed into the command classifier, and $\widehat{f^z}$ is fed into the weather classifier to guide the distribution of the latent space. Intuitively, $\overline{f^z}$ has information of the common factors across all domains, such as lane and curb, and the factors appear similarly in the same command. Furthermore, $\widehat{f^z}$ represents the domain-specific factors, such as weather. With the categorical labels, therefore, we can learn the disentanglement of latent variables and the modeling of multiple modes simultaneously. The objective function is can be defined with categorical cross-entropy (CCE):

$$L_{cce} = CCE(\hat{w}_\phi, w) + CCE(\hat{c}_\phi, c), \tag{2}$$

where $\hat{w}_\phi$ and $\hat{c}_\phi$ are predicted outputs of the weather and command classifier.

Because there is no explicit knowledge about dynamic objects in raw observation I, we assume that $\widetilde{f^z}$ is an exclusive representation of $\overline{f^z}$ and $\widehat{f^z}$ at $I \in D_{Obj}$. To extract the factors, we employ the exclusive representation learning process of the disentangling representation via the mutual information estimator (Sanchez, Serrurier, and Ortner 2020).

Let us assume that $p(x, z)$ is the joint probability density function and that $p(x)$ and $p(z)$ are the marginal probability density function of two random variables $X \in \mathcal{X}$ and $Z \in \mathcal{Z}$. The mutual information can be estimated and maximized based on the Jensen-Shannon divergence with a statistics neural network, $T_\omega : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ with parameter ω, as written in Equation 3.

$$\widehat{MI}^{JSD}_\omega(X, Z) = \mathbb{E}_{p(x,z)}[-\log(1 + e^{-T_\omega(x,z)})] - \mathbb{E}_{p(x)p(z)}[\log(1 + e^{T_\omega(x,z)})] \tag{3}$$

In our case, X and Z can be an input image I and $\mathcal{F}^z = [\overline{f^z}, \widehat{f^z}, \widetilde{f^z}]$, respectively. $\widetilde{f^z}$ is represented by deep neural network $\mathcal{E}_\psi : \mathcal{X} \to \widetilde{f^z}$ of parameters ψ. Equation 4 is the objective function of the mutual information maximization.

$$L^{max}_{MI} = L^{global}_{\omega,\psi}(X, \mathcal{F}^z) + L^{local}_{\gamma,\psi}(X, \mathcal{F}^z) \tag{4}$$

Here, each term in equation 4 can be defined as:

$$L^{global}_{\omega,\psi}(X, \mathcal{F}^z) = \widehat{MI}^{JSD}_\omega(X, \mathcal{F}^z)$$

$$L^{local}_{\gamma,\psi}(X, \mathcal{F}^z) = \sum_i \widehat{MI}^{JSD}_\gamma(C^{(i)}_\psi(X), \mathcal{F}^z),$$

where $C_\psi(X)$ is a feature map encoded from X by $\mathcal{E}_\psi$. The mutual information is computed by the global statistics network T_ω and the local statistics network T_γ of parameters ω and γ. In this process, we enforce the dynamic object information to $\widetilde{f^z}$ by remaining $\overline{f^z}$ and $\widehat{f^z}$ are fixed.

Furthermore, to minimize the mutual information information between $[\overline{f^z}, \widehat{f^z}]$ and $\widetilde{f^z}$, we use an adversarial objective as written in Equation 5.

$$L^{min}_{MI} = \mathbb{E}_{p([\overline{f^z}, \widehat{f^z}])p(\widetilde{f^z})}[\log D([\overline{f^z}, \widehat{f^z}], \widetilde{f^z})] + \mathbb{E}_{p([\overline{f^z}, \widehat{f^z}], \widetilde{f^z})}[\log(1 - D([\overline{f^z}, \widehat{f^z}], \widetilde{f^z}))], \tag{5}$$

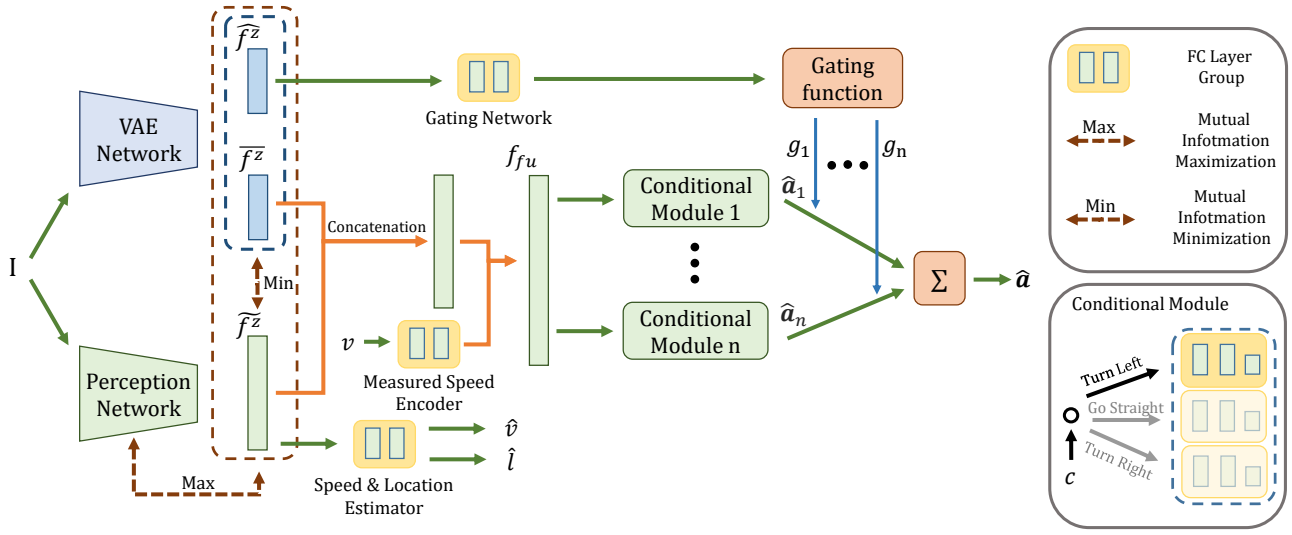

Figure 1: Overall architecture of the proposed network. The input I is given to the VAE and perception network. $\widehat{fz}$ and $\overline{fz}$ are represented by the VAE Network. The features $\widetilde{fz}$, independent of $\widehat{fz}$ and $\overline{fz}$ and required to determine the control values, are encoded by the perception network. $\overline{fz}$, $\widetilde{fz}$, and measured speed feature are fused by FC layers and fed into the conditional modules called the expert. The gating network assigns a weight g_i to each of the expert's output $\hat{\boldsymbol{a}}_i$.

where D is a discriminator defined by neural network to classify representations drawn from $p([\overline{fz}, \widehat{fz}])p(\widetilde{fz})$ as real and representations drawn from $p([\overline{fz}, \widehat{fz}], \widetilde{fz})$ as fake sample. The adversarial objective is to make the dimensions of the concatenated representation $[\overline{fz}, \widehat{fz}]$ independent of the dimensions of the representation $\widetilde{fz}$.

Action Prediction with Mixture of Domain-specific Experts

We adopt a mixture of experts model to train sub-policies on each specialized expert instead of using a single expert for mapping visual observations to control values. Hence, we divide the policy space into several weather conditions, with experts becoming specialized on each domain-specific policy.

The disentangled features are fed into the next neural network layers to be used for each purpose as displayed in Figure 1. Weather-specific weights for the experts are calculated by gating function with a gating network and $\widehat{fz}$. The gating network consists of two fully connected (FC) layers, and the gating function is a softmax nonlinear operator to allocate the weather-specific weight of each conditional module for predicting the action values according to a certain driving condition. Because $\overline{fz}$ and $\widetilde{fz}$ are the domain-irrelevant features, they have consistent representation on any domain. However, the control values in the action space (A) have different distributions according to the driving condition. For example, on rainy days a higher brake value is required due to tire slip. The disentangled domain-irrelevant features have well-refined information about the road components and they can be better utilized to train the domain-specific action space. With this approach, the proposed mixture of domain-specific experts (MoDE) framework can estimate an accurate distribution of action vector through the cooperation of experts. For this reason, $\overline{fz}$ and $\widetilde{fz}$ are fed into the experts to estimate vehicle action values, such as longitudinal and lateral control values. The objective of proposed MoDE framework can be written as:

$$L_{MoE} = \sum_i g_i ||\hat{\boldsymbol{a}}_{i,\psi} - \boldsymbol{a}||^2, \tag{6}$$

where $\hat{\boldsymbol{a}}_\psi$ are predicted action values with the perception network of parameter ψ and $\boldsymbol{g}$ is probability of picking expert for each condition.

Speed and Location Offset Prediction Regularization

Except for the explicit mapping of on-policy demonstrations to the model, a proven approach to increase the generalization ability is to jointly train with a related subtask. Firstly, following the exploration (Codevilla et al. 2019), we utilize a speed prediction branch; the loss term is defined as follows:

$$L_v = ||\hat{v}_\psi - v||^2. \tag{7}$$

In addition to the previous study, we introduce another subtask called location offset prediction. Based on the kinematic dynamics, the location offset from the ego-vehicle in current global coordinates (x_t, y_t) at the next time-step $t+1$ are defined as follows:

$$x_{t+1} = v_t cos(\delta_t)\Delta t \tag{8}$$

$$y_{t+1} = v_t sin(\delta_t)\Delta t, \tag{9}$$

where v_t is the current speed from the sensorimotor controller and δ_t is the wheel steering angle. Δt is a constant

time-gap defined by the collecting frequency, and in this study, we set it to 0.1. The loss function of the location offset can be rewritten:

$$L_l = ||\hat{l}_\psi - l||^2. \tag{10}$$

Here, $\hat{v}_\psi$ and $\hat{l}_\psi$ are the predicted outputs from perception network and l is that calculated offset by the equation 8 and 9. Through the joint optimization process, additional diversity is embedded in the perception network for learning generalized representation.

Experiments

Benchmarks and Datasets

The CARLA simulator supports dynamic objects, such as vehicles and pedestrians, for a large variety of driving situations. To empirically justify our approach, we conducted several experiments on the *NoCrash* CARLA benchmark (Codevilla et al. 2019) and the *AnyWeather* benchmark (Ohn-Bar et al. 2020) of CARLA simulator version 0.8.4. All benchmarks evaluate the driving performance under various situations, such as "Training Conditions", "New Weather", "New Town", and "New Town & Weather", in terms of success rate (SR), which is the percentage of episodes that successfully reached the goal within a specified time budget without colliding with any object. For a fair comparison with other methods, we conducted experiments according to the benchmark policies (Dosovitskiy et al. 2017; Codevilla et al. 2019). Since the previous models (Ohn-Bar et al. 2020; Kim et al. 2020) achieve almost perfect SRs on the original CARLA benchmark (Dosovitskiy et al. 2017), it is difficult to compare the driving performance of the model using the benchmark. In addition, *AnyWeather* benchmark includes "New Town & Weather" condition, which is the most difficult environment of the original CARLA benchmark. For these reasons, we do not compare the experimental results on the original CARLA benchmark.

With the autopilot system (felipecode 2018) for the CARLA simulator, we collect the training data D under two conditions: the driving dataset with objects (D_{Obj}) and without objects (D_{NoObj}). The D_{Obj} is constructed by referring to the configuration of the CIL (Codevilla et al. 2018) approach: three frontal RGB cameras, an arbitrary number of objects, and random noise injection. We randomly collected the sample of the number of objects from the intervals [100, 150] and [200, 300] for vehicles and pedestrians, respectively. The D_{NoObj} consists of paired driving images captured by three frontal RGB cameras, with the same angle configuration to D_{Obj}, under four training weather conditions at the same position.

Data preprocessing is an essential step in overcoming the limitations derived from the extremely unbalanced data (Codevilla et al. 2019). Because most of the data were collected in straight driving scenarios, the experts are trained to only focus on going straight. Referring to the idea of Wang et al. (Wang et al. 2019), we divide the control values into bins of size 0.1 and sample them randomly. The total number of D is about 3.5 million, and data augmentation (gaussian blur, additive gaussian noise, multiplicative brightness variation, contrast variation, and saturation variation) is performed for generalization.

Implementation in Detail

Concerning the model architecture, we adopt a similar model to the LUSR for cycle-consistent VAE, except for the number of CNN layers. To increase the representation ability of the encoder network, we use twice as many CNN layers at each convolutional block in LUSR. In addition, we utilized ResNet50 (He et al. 2016) pre-trained on ImageNet as a backbone network for the perception network.

The number of hidden units of the $\overline{fz}$, $\widehat{fz}$, and $\widetilde{fz}$ is 256, and the size of the f^{fu} is 512. Furthermore, the classifiers and gating network are three FC layers with sizes of 256, 128, and 4. In the second training stage, the weights of the gating network are initialized using the weights of the weather classifier learned in the first training stage. The size of the other FC layer groups and the conditional modules (Figure 1) are the same as those in the conditional imitation learning extension with a ResNet backbone and speed regularization (CILRS) (Codevilla et al. 2019) framework. In our experiment, we employ a 256×256 input image resolution based on the previous study (Ohn-Bar et al. 2020). Every model is trained using the Adam solver with mini-batches of 200 samples. The learning rate is set to 0.0001 at the beginning and then it decreases by a factor 0.1 at 25 %, 50 %, and 75 % of the total number of training epochs.

Total loss function of the first training stage is defined as:

$$L_{first} = \lambda_{vae} L_{cyclic} + \lambda_{cce} L_{cce}, \tag{11}$$

where λ_{vae} and λ_{cce} are 0.8 and 0.2, respectively. Additionally, total loss function for the second stage is defined as follows:

$$\begin{aligned} L_{second} = \lambda_{MI}(-L_{MI}^{max} + L_{MI}^{min}) + \\ \lambda_{MoE} L_{MoE} + \lambda_r (L_v + L_l), \end{aligned} \tag{12}$$

where λ_{MI}, λ_{MoE}, and λ_r are 0.2, 0.7, and 0.1, respectively.

Comparison with the state of the art

We compare our model with the recent state-of-the-art approaches: the CILRS (Codevilla et al. 2019), learning situational driving (LSD) (Ohn-Bar et al. 2020) model, and future action and states network (FASNet) (Kim et al. 2020). Every model utilizes the ResNet backbone as the perception network and uses an RGB image only for training.

Table 1 reports the quantitative comparison on the *NoCrash* benchmark with state-of-the-art networks. This benchmark measures the driving ability of the model to react to dynamic objects under various conditions for three driving tasks: "Empty", "Regular Traffic", and "Dense Traffic". As shown in Table 1, our proposed MoDE achieves state-of-the-art SRs, except for some conditions. Although we have a little lower performance compared to other models under the "New Town" condition for the "Empty" task, we achieve higher SRs in other tasks. Especially, the most important result is that our model has the highest SRs under every "Regular Traffic" task. The "Empty" task focuses on measuring the capacity of lateral control, and there exists randomness

Table 1: Comparison with the state-of-the-art networks on the *NoCrash* CARLA benchmark in terms of SR in each condition. The results are percentage (%) of SR and higher values are better.

	Training Conditions				New Weather			
Task	CILRS	LSD	FASNet	MoDE	CILRS	LSD	FASNet	MoDE
Empty	97±2	-	96±0	**98±0**	96±1	-	**98±0**	**98±0**
Regular	83±0	-	90±1	**93±0**	77±1	-	80±1	**84±2**
Dense	42±2	-	44±2	**45±3**	**47±5**	-	38±4	46±2
	New Town				New Town & Weather			
Task	CILRS	LSD	FASNet	MoDE	CILRS	LSD	FASNet	MoDE
Empty	66±2	94±1	**95±1**	93±0	90±2	**95±1**	92±2	94±2
Regular	49±5	68±2	77±2	**80±2**	56±2	65±4	66±4	**68±2**
Dense	23±1	30±4	**37±2**	**37±2**	24±8	32±3	32±4	**34±4**

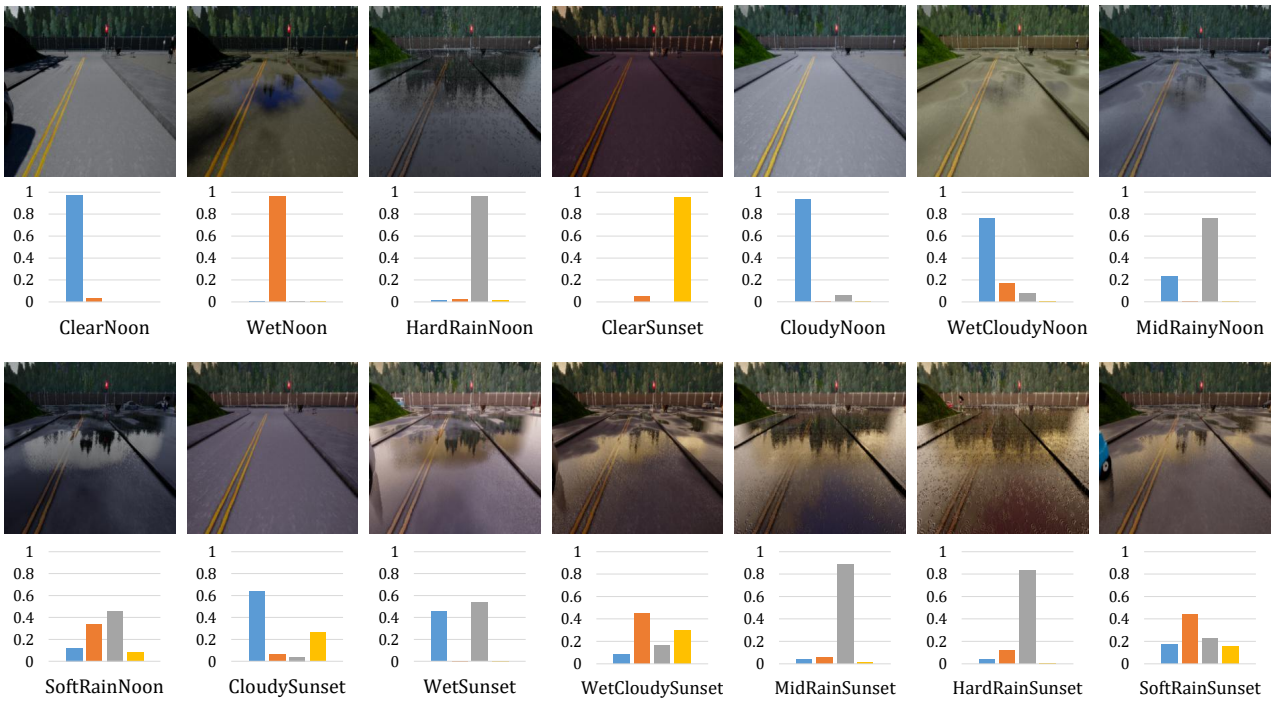

Figure 2: Average outputs of the gating function for every weather in the CARLA simulator. Every odd row contains images of the driving scene; even rows contain graphs of the calculated importance weights for the experts under certain weather. The first four weather are training conditions: "ClearNoon", "WetNoon", "HardRainNoon", and "ClearSunset".

Table 2: Experimental results on the *AnyWeather* benchmark in terms of SR.

	New Town & Weather			
Task	CILRS	LSD	FASNet	MoDE
Straight	83.2	85.6	93.2	**93.6**
One Turn	78.4	81.6	87.0	**89.2**
Navigation	76.4	79.6	82.8	**83.6**
Nav. Dynamic	75.6	78.4	81.2	**82.4**

in the results of the "Dense Traffic" task due to numerous objects appearing and moving in random positions. Some episodes fail because a pedestrian suddenly crashes itself into the ego-vehicle or the intersection is already blocked by an accident of other vehicles. For these reasons, comparing the results of the "Regular Traffic" is most reasonable for evaluating the longitudinal and lateral control abilities of the models.

The *AnyWeather* benchmark is a new benchmark to measure the generalization capability under drastically diverse visual conditions, which are the "New Town" under all ten kinds of weather unseen in the training process. The results are presented in Table 2 and the MoDE model exhibits state-of-the-art SRs under all tasks. It is observable that the proposed model has a higher generalization ability to unseen driving environments. Analyzing the result, the most difficult weather is "MidRainSunset" and "HardRainSunset" because the lane is invisible in heavy rain and dark illumination.

(a) (b) (c) (d)

Figure 3: Results of reconstructed images. The first row is input images, and the second row is reconstructed images that take $\overline{fz}$, $\widehat{fz}$, and $\widetilde{fz}$ from the first row. The last row is reconstructed images by swapping $\widetilde{fz}$ from (a) and (c) with (b) and (d), respectively.

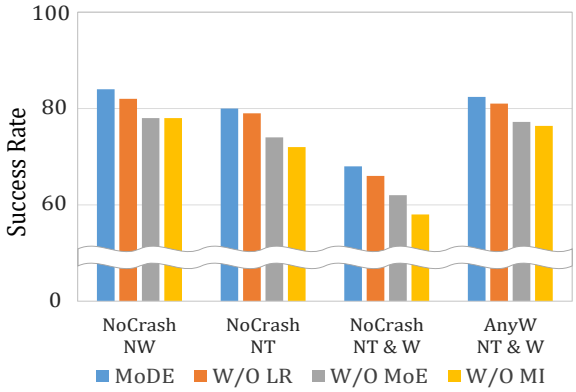

Figure 4: Results of the ablation studies on the *NoCrash* benchmark under "Regular Traffic" task and *AnyWeather* benchmark (AnyW) under "Nav. Dynamic" task. The NW and NT are "New Weather" and "New Town", respectively.

Ablation Study

We performed an ablative analysis to assess whether our approaches improve the driving performance for the "Regular Traffic" of *NoCrash* benchmark and the "Nav. Dynamic" of *AnyWeather* benchmark using without the location offset regularization (W/O LR), the mixture of experts (W/O MoE), and mutual information estimator (W/O MI). The elimination of each module does not cause the SRs to decrease much under the "Training Conditions". This aspect means that commonly used neural network architecture, such as the perception network, VAE network, and single conditional module, have sufficient ability to encode the features needed to predict the control value for training data. As shown in Figure 4, the SRs decrease consistently with the elimination of each module. It can be interpreted that each module improves the generalization ability for the unseen driving environments. In the case of W/O LR, the SRs slightly dropped under every conditions. This suggests that the joint optimization with location offset task implements the perception network to learn general representation for the diverse situation. Moreover, we have experimentally demonstrated that combination of experts specialized in each domain is a good strategy to achieve reliable prediction of action values under unseen environment. Finally, when we train the model without the mutual information estimator (W/O MI), the SRs dropped the most. It seems that the generalization ability is significantly degraded because the $\widetilde{fz}$ cannot be guaranteed to be a domain-irrelevant feature.

Weights of Experts under Diverse Conditions

Figure 2 shows the mixing proportion assigned to each expert. Since we set the number of experts to four, the bars in the histogram represent the importance of experts under the weather condition. The first four results are the training weather condition and the rest are the results of the test condition. For the training conditions, most outputs are predicted by a particular conditional module. Furthermore, for the test environment, the assigned combination of experts for certain weather is quite reasonable. For example, to predict the control values of "CloudyNoon" the gating network assigns the experts of "ClearNoon" and "HardRainNoon".

Dynamic Object Factor Disentanglement

To show that our proposed two-stage representation learning can disentangle the dynamic object factor, we train a simple decoder using fixed $\overline{fz}$, $\widehat{fz}$, and $\widetilde{fz}$ for image reconstruction. After finishing the training, we first select images I_1 from D_{Obj} and I_2 from D_{NoObj} and then extract their latent factors. Second, we reconstruct the images $(\hat{I}_1, \hat{I}_2)$ using swapped factors $\widetilde{fz}_1$ from I_1 with $\widetilde{fz}_2$ from I_2. As a result, the vehicle moves from $\hat{I}_1$ to the $\hat{I}_2$ displayed in the last row of Figure 3.

Conclusion

In this study, we introduced a two-stage representation learning approach to disentangle the latent feature space into three disjoint features: domain-specific, domain-general, and dynamic object features. To solve the problem that no explicit knowledge about the dynamic objects is present, we adopted the statistics neural network, that splits the latent features using a mutual information estimator. We guided each factor in the representation learning process to ensure that each feature has a distribution suitable for its purpose by simultaneously optimizing related tasks in a multi-task learning manner. These well-refined features are utilized to train the MoDE framework in realizing a stable autonomous driving model. We have empirically shown that the proposed strategy can improve the robustness of the model under unseen environments through various driving experiments.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No.B0101-15-0266, Development of High Performance Visual BigData Discovery Platform for Large-Scale Realtime Data Analysis), (No.2017-0-00897, Development of Object Detection and Recognition for Intelligent Vehicles) and (No.2018-0-01290, Development of an Open Dataset and Cognitive Processing Technology for the Recognition of Features Derived From Unstructured Human Motions Used in Self-driving Cars)

References

Belghazi, M. I.; Baratin, A.; Rajeshwar, S.; Ozair, S.; Bengio, Y.; Courville, A.; and Hjelm, D. 2018a. Mutual information neural estimation. In *International Conference on Machine Learning*, 531–540. PMLR.

Belghazi, M. I.; Baratin, A.; Rajeswar, S.; Ozair, S.; Bengio, Y.; Courville, A.; and Hjelm, R. D. 2018b. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*.

Bojarski, M.; Del Testa, D.; Dworakowski, D.; Firner, B.; Flepp, B.; Goyal, P.; Jackel, L. D.; Monfort, M.; Muller, U.; Zhang, J.; et al. 2016. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.

Chen, D.; Zhou, B.; Koltun, V.; and Krähenbühl, P. 2020. Learning by cheating. In *Conference on Robot Learning*, 66–75. PMLR.

Chen, J.; Yuan, B.; and Tomizuka, M. 2019. Deep imitation learning for autonomous driving in generic urban scenarios with enhanced safety. *arXiv preprint arXiv:1903.00640*.

Chen, R. T.; Li, X.; Grosse, R.; and Duvenaud, D. 2018. Isolating sources of disentanglement in variational autoencoders. *arXiv preprint arXiv:1802.04942*.

Codevilla, F.; Miiller, M.; López, A.; Koltun, V.; and Dosovitskiy, A. 2018. End-to-end driving via conditional imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1–9. IEEE.

Codevilla, F.; Santana, E.; López, A. M.; and Gaidon, A. 2019. Exploring the limitations of behavior cloning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9329–9338.

Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; and Koltun, V. 2017. CARLA: An open urban driving simulator. *arXiv preprint arXiv:1711.03938*.

felipecode. 2018. CARLA 0.8.4 Data Collector. https://github.com/carla-simulator/data-collector.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Jacobs, R. A.; Jordan, M. I.; Nowlan, S. J.; and Hinton, G. E. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1): 79–87.

Jeon, I.; Lee, W.; and Kim, G. 2018. IB-GAN: Disentangled representation learning with information bottleneck GAN.

Jha, A. H.; Anand, S.; Singh, M.; and Veeravasarapu, V. 2018. Disentangling factors of variation with cycle-consistent variational auto-encoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 805–820.

Kim, I.; Lee, H.; Lee, J.; Lee, E.; and Kim, D. 2020. Multi-task Learning with Future States for Vision-based Autonomous Driving. In *Proceedings of the Asian Conference on Computer Vision*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Li, Z.; Motoyoshi, T.; Sasaki, K.; Ogata, T.; and Sugano, S. 2018. Rethinking self-driving: Multi-task knowledge for better generalization and accident explanation ability. *arXiv preprint arXiv:1809.11100*.

Ohn-Bar, E.; Prakash, A.; Behl, A.; Chitta, K.; and Geiger, A. 2020. Learning Situational Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11296–11305.

Sanchez, E. H.; Serrurier, M.; and Ortner, M. 2020. Learning disentangled representations via mutual information estimation. In *European Conference on Computer Vision*, 205–221. Springer.

Sauer, A.; Savinov, N.; and Geiger, A. 2018. Conditional affordance learning for driving in urban environments. *arXiv preprint arXiv:1806.06498*.

Sohn, K.; Lee, H.; and Yan, X. 2015. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28: 3483–3491.

Virgo, M. 2017. Udacity Public Driving Dataset. https://github.com/udacity/self-driving-car.

Wang, Q.; Chen, L.; Tian, B.; Tian, W.; Li, L.; and Cao, D. 2019. End-to-end autonomous driving: An angle branched network approach. *IEEE Transactions on Vehicular Technology*.

Xing, J.; Nagata, T.; Chen, K.; Zou, X.; Neftci, E.; and Krichmar, J. L. 2021. Domain Adaptation In Reinforcement Learning Via Latent Unified State Representation. *arXiv preprint arXiv:2102.05714*.

Yang, Z.; Zhang, Y.; Yu, J.; Cai, J.; and Luo, J. 2018. End-to-end multi-modal multi-task vehicle control for self-driving cars with visual perceptions. In *2018 24th International Conference on Pattern Recognition (ICPR)*, 2289–2294. IEEE.

Zeno, B.; Kalinovskiy, I.; and Matveev, Y. 2019. IP-GAN: Learning Identity and Pose Disentanglement in Generative Adversarial Networks. In *International Conference on Artificial Neural Networks*, 535–547. Springer.