Exploring Supervised, Semi-supervised and Self-supervised Learning in Autonomous Driving

Partha Ghosh
University of Tübingen

Abstract

How can we learn generalized autonomous driving models for robust vision-based navigation in complex and dynamic settings? The status quo for solving these visual navigation tasks is to train visual representations and navigation policies with direct supervision. Yet, vast amounts unlabeled, highly diverse ego-centric navigation data is freely available on the internet. Therefore, in this work, we study several different approaches to effectively utilize this huge diverse navigational data to robustly scale across perspectives, platforms, environmental conditions, scenarios, and geographical locations. We study two existing works in this field that employes semi-supervised and self-supervised learning, namely 'SelfD' [53] and 'OVRL' [49]. Moreove, in this work, we introduce SemiD, a framework for learning scalable driving by utilizing large amounts of online monocular images. Our key idea is to leverage deep visual odometry for iterative semi-supervised training when learning imitative agents from unlabeled data. To handle unconstrained viewpoints, scenes, and camera parameters, we train an image-based model that directly learns to plan in the Bird's Eye View (BEV) space. We use deep visual odometry to generate pseduo-labels for the the unlabeled data and augment the decision-making knowledge and robustness of the model via semi-supervised training. We employ a large dataset of publicly available YouTube videos to train SemiD and comprehensively analyze its generalization benefits across challenging navigation scenarios. Without requiring any additional data collection or annotation efforts, SemiD outperforms all the previous approaches and demonstrates consistent improvements from 51% to 95% in route completion and from 7.8% to 13.3% in driving score in challenging CARLA evaluation routes.

Keywords: Imitation Learning, Self-supervised Learning, Semi-supervised Learning

Acknowledgements

I want to thank Bernhard Jaeger and Katrin Renz for supervising the project and providing me with the necessary guidance and valuable support. I want to thank Prof. Andreas Geiger for the helpful discussions. Finally, I want to express my very profound gratitude to my parents for their continuous support and encouragement throughout my whole life.

Contents

Chapter 1

Introduction

How should we teach autonomous systems to drive based on visual input? How can we learn generalized models for robust vision-based navigation in complex and dynamic settings? While humans can effortlessly transfer general navigation knowledge across settings and platforms (e.g. geographical location, use-case, rare-scenarios, camera mounting point), current navigation agents cannot transfer this knowledge well. With this question in mind, we are therefore, interested in exploring learning strategy with which navigation agents can learn to understand, across all settings and platforms, the structure and semantics of their environments and navigate accordingly without providing extensive direct supervision.

The status quo for solving these visual navigation tasks is to train visual representations and navigation policies from scratch with direct supervision. The family of approaches that has demonstrated promising results is imitation learning [14, 18]. The agent is given trajectories generated by an expert driver, along with the expert's sensory input. The goal of learning is to produce a policy that will mimic the expert's actions given corresponding input [1, 5, 12, 13, 22, 24, 29, 30, 32, 27, 34, 35, 36, 52]. Now, every minute, vast amount of highly diverse and freely available ego-centric navigation data containing such scenarios are uploaded to the web. Even though the expert's actions may not be readily available from these demonstration data, these data can be parsed to recover the corresponding expert's action. Another feasible approach could be to learn better representations from these unlabeled data which is a very popular topic in visual recognition. The learned representations are shown to be generalizable across visual tasks ranging from image classification, semantic segmentation, to object detection [15, 21, 25, 9, 8]. However, these methods are primarily built for learning features for recognition tasks rather than navigation. Therefore, in this work we aim towards effectively utilizing such freely available demonstration data to improve the efficiency, safety, and scalability of generalized real-world navigation agents.

We explore two different type of learning in the context of self-driving that facilitates learning from large amounts of unlabeled experience (combined with a small amount of direct super-

vision): (1) Semi-supervised Learning, (2) Self-supervised Representation Learning. Both semi-supervised and self-supervised methods are similar in the sense that they both facilitate learning from large amounts of unlabeled experience, but the way both formulate this, is quite differently. In semi-supervised learning, we devise strategies to generate reasonable pseduo-labels to the unlabeled input driving scenes so that upon training with these pseduo-labels the network can learn better generalized representations of these diverse driving scenes and therefore perform better in the navigation task. On the other hand, in self-supervised learning through different approaches (e.g. contrastive learning, entropy regulation) the network directly learns good representation of the unlabeled inputs. We also try to combine both of these approaches.

In this thesis, we aim to build and compare three different learning approaches — two of which can be catogorized as semi-supervised learning and the other one as self-supervised learning. We incorporate and explore an existing semi-supervised learning approach "SelfD" proposed by Zhang et. al. [53] in our autonomous driving framework. We explore in our self-driving framework, "OVRL", a self-supervised learning approach proposed by Yadav et. al. [49] which has been studied in the domain of embodied navigation. Moreover, we propose SemiD, a new semi-supervised learning method that outperforms the prior works in the challenging NEAT evaluation routes [17]. Also, we propose a data cleansing and sampling technique for effective training. We show that combining the aforementioned technique with mild image augmentation improve the driving performance by a huge margin.

In summary, the main contributions of this work are:

- We present SemiD, a novel semi-supervised learning approach based on deep visual odometry for autonomous driving that outperforms prior works in the challenging NEAT evaluation routes.
- We showed that combining SemiD and OVRL brings further improvements in route completion.
- We propose a data cleaning pipeline and stratified sampling in training to improve the driving perfomance.
- We find that image augmentations is quite important for achieving good performance.
- We show that the inertia problem can be solved by employing the above techniques.

We organize the structure of the thesis as follows. We first provide an overview of the related works in this field in Section 2. Then, in Section 3, we introduce our autonomous driving framework where we will deploy all our learning approaches. In Section **??**, we describe in details various semi-supervised and self-supervised learning approaches. Next, we discuss the experiment results in Section **??**, followed by the conclusion in Section **??**.

Chapter 2

Related Work

2.1 End-to-End Autonomous Driving

End-to-End driving describes approaches in which the entire driving task is done by a single neural network that directly maps the raw sensor data to the driving commands. The neural network can be trained using different algorithms, the two most important ones being imitation learning and reinforcement learning. Even though the models are hard to interpret, the advantages of this approach is that these models can be optimized directly for driving. Furthermore, data annotations are cheap, since a camera can be attached to a car and sensors to the steering mechanisms to collect data automatically. This approach was used early on by researchers like Pomerleau et al. and their ALVINN-vehicle [35] and still active research is going on in this field [26, 40]. Imitation Learning for driving has advanced significantly [5, 18, 32] and is currently employed in several state-of-the-art approaches, some of which predict waypoints [10, 14, 20], whereas others directly predict vehicular control [3, 6, 19, 33, 48, 37]. While other learning-based driving methods such as affordances [39, 47] and reinforcement learning [13, 43, 45] could also benefit from a semi-supervised or self-supervised learning, in this work, we try to improve imitation learning based autonomous driving through semi/self-supervised learning.

2.2 Imitation Learning

Imitation Learning is the most promising approach for self-driving. Our main idea is to leverage the scale and diversity of readily accessiblem online ego-centric navigation data to learn a robust conditional imitation learning policy [14, 18]. While learning from labeled demonstrations can significantly simplify the challenging vision-based policy learning task [1, 5, 12, 13, 22, 24, 29, 30, 32, 27, 34, 35, 36, 52, 55, 56], observed images in our settings are not labeled with corresponding actions of a demonstrator. Therefore we aim to generalize

current conditional imitation learning (CIL) approaches [14, 18, 19] to learn, from unlabeled image observations, an agent that can navigate in complex urban scenarios. To address this challenging observational learning task, prior work has recently explored introducing various restrictive assumptions, including access to a hand-designed reward function [11], an interactive environment for on-policy data collection [41], or demonstrator optimality [41, 42]. We instead facilitate scalable training from diverse data sources, by employing semi-supervised or self-supervised learning approaches. Our resulting model can also be used to bootstrap other methods for policy training, e.g., model-based or model-free reinforcement learning approaches [13, 30, 33, 43].

2.3 Semi-Supervised Learning for Navigation

2.3.1 Semi-supervised Learning

Semi-supervised learning is an approach to machine learning that combines a small amount of labeled data with a large amount of unlabeled data during training. It falls between unsupervised learning (with no labeled training data) and supervised learning (with only labeled training data). It is a special instance of weak supervision. Semi-supervised learning more closely imitates the way humans learn. In semi-supervised learning, the neural network learns in two steps:

- *Transductive Learning:* First, the task is solved based on pseudo-labels (i.e. by labelling the given unlabelled data) which help to initialize the network weights.
- *Inductive Learning:* The pre-trained network is then fine-tuned with the small ground truth data.

2.3.2 Learning by Cheating

The prior works [7, 50, 28, 38], emphasizing semi-supervised learning through image and object-level recognition tasks, has limited utility for complex decision-making tasks. On the other hand, CIL involves learning to make complex actions, from known actions of human/privileged experts [18, 19, 36]. This issue has been addressed in the recent work 'Learning by Cheating' (LBC) by Chen et al. [14]. LBC utilizes a multi-stage training step, where a privileged (i.e., 'teacher') CIL agent is employed to provide supervision to a non-privileged (i.e., 'student') visuomotor CIL agent. The privileged CIL agent uses a semantic segmentation bird's eye view image as input that is processed by a ResNet and outputs waypoints that are processed by a PID controller to produce the driving controls. The non-privileged agent uses a similar network design but takes as input a frontal camera image, the car's velocity and the conditional command by the navigational planner. As the privileged agent is given access to extensive ground truth information about the world in training and testing, it pro-

duces highly plausible and clean trajectories, and thus helps the non-privileged CIL agent to learn how to drive. Learning by Cheating was the first approach to solve the original CARLA benchmark.

In contrast, our framework for learning is very different. In semi-supervised driving with DeepVO, we generate pseduo-labels for diverse out-of-distribution driving scenes and to enable transductive learning of the sensorimotor agent. SelfD, on the other hand, leverages the same visuomotor architecture as teacher and student. We also train in inherently noisy settings, as teacher inference is performed on diverse out-of-distribution image data and not on the original training dataset.

2.4 Self-Supervised Visual Representation Learning

2.4.1 Self-supervised learning

Self-supervised learning (SSL) is a machine learning approach where the supervisory signal is automatically generated. More precisely, SSL refers to learning data representations by solving a so-called pretext (or auxiliary) task, in a self-supervised fashion, i.e. you automatically generate the supervised signal from the unlabelled data.

2.4.2 Contrastive Learning

The core idea of contrastive learning is to attract the positive sample pairs and repulse the negative sample pairs. This methodology has been recently popularized for self-supervised representation learning [46]. Simple and effective instantiations of contrastive learning have been developed using Siamese networks [25, 15, 51]. In practice, contrastive learning methods benefit from a large number of negative samples. These samples can be maintained in a memory bank [46]. In a Siamese network, MoCo [25] maintains a queue of negative samples and turns one branch into a momentum encoder to improve consistency of the queue. SimCLR [15] directly uses negative samples coexisting in the current batch, and it requires a large batch size to work well.

2.4.3 Non-Contrastive Learning

Recent works have shown that we can learn unsupervised features without discriminating between images. Grill et al. [21] propose a metric-learning formulation called BYOL, where features are trained by matching them to representations obtained with a momentum encoder. Methods inspired from BYOL [16, 8], have shown that this method works even without a momentum encoder.

2.4.4 Action-COnditioned Policy Pretraining (ACO)

Action-Conditioned policy pretraining paradigm, uses contrastive learning to capture important features in the neural representation relevant to the decision making and benefits downstream tasks. The methods, proposed by Zhang et. al. [54] works by first collecting a large corpus of driving videos with a wide range of weather conditions, from wet to sunny, from all across the world without labeling and then generating action pseudo labels for each frame using a pretrained inverse dynamics model. Then, instead of contrasting images based on different augmented views, this method considers a new contrastive pair conditioned on action similarity and by learning with those action-conditioned contrastive pairs, the representation captures policy-related elements that are highly correlated to the actions. The experimental results show that ACO successfully learns generalizable features for the downstream task such as policy learning through Imitation Learning (IL) and Reinforcement Learning (RL) in end-to-end autonomous driving, and Lane Detection (LD).

In contrast, we focus on non-contrastive learning, particularly, the method DINO proposed by Caron et. al. [8], to learn visual representations from the unlabeled data. These generic representations can then be transferred to the policy learning task.

Chapter 3

Autonomous Driving Framework

We consider the task of point-to-point navigation in an urban setting where the goal is to complete a given route while safely reacting to other dynamic agents and following traffic rules. To achieve this, we consider the imitation learning approach of learning policy, as in self-driving it is easier for an expert to demonstrate the desired behaviour rather than to specify a reward function.

3.1 Imitation Learning

The goal of Imitation Learning (IL) is, for an agent to learn a policy $\pi_{\boldsymbol{\theta}}$, that imitates the behavior of an expert π^*. The agent learns to map an input to a navigational decision. In general, the decision may either be a low-level vehicle control action [19] (e.g. steering, throttle and break) or a desired future trajectory relative to the ego-vehicle, i.e., a set of K waypoints [14, 32] in the BEV (birds-eye-view) space. In the latter case, future waypoints may be paired with a hand-specified or learned motion controller to produce the low-level action [14, 32]. In this work, we focus on the later representation due to its interpretability and genralizability. To find the mapping, we consider the Behavior Cloning (BC) approach of IL. To explore different learning approaches in Section **??**, we would need access to small amount of ground truth data. For that, an expert policy is first rolled out to collect at each time-step, high-dimensional observations of the environment including front camera image, ego-vehicle position and orientation, high-level navigational command and high-level goal location provided as GPS coordinates etc. From these high-dimensional observations, we derive our dataset $\mathcal{D} = \{(\mathbf{X}_i, \mathbf{W}_i)\}_{i=1}^{N} \in (\mathcal{X}, \mathcal{W})$ of size N, where the input $\mathbf{X}$ consists of the front camera image and the goal location and the corresponding expert trajectory $\mathbf{W}$, defined by a set of 2D waypoints relative to the coordinate frame of the ego-vehicle in BEV space, i.e., $\mathbf{W} = \{\mathbf{w}_t = (x_t, y_t)\}_{t=1}^{T}$, are calculated from the ego-vehicle positions and orientations from the subsequent frames. Our goal is to find a decision making policy i.e. a waypoint

prediction function $\pi_{\boldsymbol{\theta}} : \mathcal{X} \rightarrow \mathcal{W}$ with learnable parameters $\boldsymbol{\theta} \in \mathbb{R}^d$. In BC, the policy $\pi_{\boldsymbol{\theta}}$ is learned by training a neural network in a supervised manner using the dataset, $\mathcal{D}$, with a loss function, $\mathcal{L}$ i.e.

$$\underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{(\mathbf{X},\mathbf{W})\sim\mathcal{D}}[\mathcal{L}(\mathbf{W}, \pi_\theta(\mathbf{X}))].$$

We use the L_1 distance between the predicted trajectory, $\pi_{\boldsymbol{\theta}}(\mathbf{X})$, and the corresponding expert trajectory, $\mathbf{W}$, as the loss function. We assume access to an inverse dynamic model [4], implemented as a PID controller $\mathbb{I}$, which performs the low-level control, i.e., steer, throttle and brake, provided the future trajectory $\mathbf{W}$. The action are dertermined as $\mathbf{A} = \mathbb{I}(\mathbf{W})$.

3.2 Input and Output Representations

3.2.1 Input Representation

3.2.1.1 Driving Scenes

We note that, even though our collected expert demostrations from CARLA and the YouTube driving videos are sequential, we do not use temporal data for training. Contrary to our intuition of getting better generalization in decision-making from sequential observations, the prior works on IL for autonomous driving have shown that using observation histories may not lead to performance gain [23, 31, 2, 44]. Thus, we use a single time-step input. We consider the front camera with a FOV of $120°$. We extract the front image at a resolution of 960×480 pixels which we resize and crop to 256×256 to remove radial distortion at the edges. Figure 3.2 shows some example driving scenes from CARLA and the YouTube driving videos.

3.2.1.2 Global Planner

We follow the standard protocol of CARLA 0.9.10 and assume that high-level goal locations c are provided as GPS coordinates by an A^* navigational planner. Agents are supposed to follow routes directed by these GPS coordinates. Note that, these goal locations are sparse and can be hundreds of meters apart, as opposed to the local waypoints predicted by the policy $\pi_{\boldsymbol{\theta}}$.

3.2.2 Output Representation

We predict the future trajectory $\mathbf{W}$ of the ego-vehicle in BEV space, centered at the current coordinate frame of the ego-vehicle. The trajectory is represented by a sequence of 2D waypoints, $\mathbf{W} = \{\mathbf{w}_t = (x_t, y_t)\}_{t=1}^{T}$. We use $T = 4$, which is the default number of waypoints required by our inverse dynamics model.

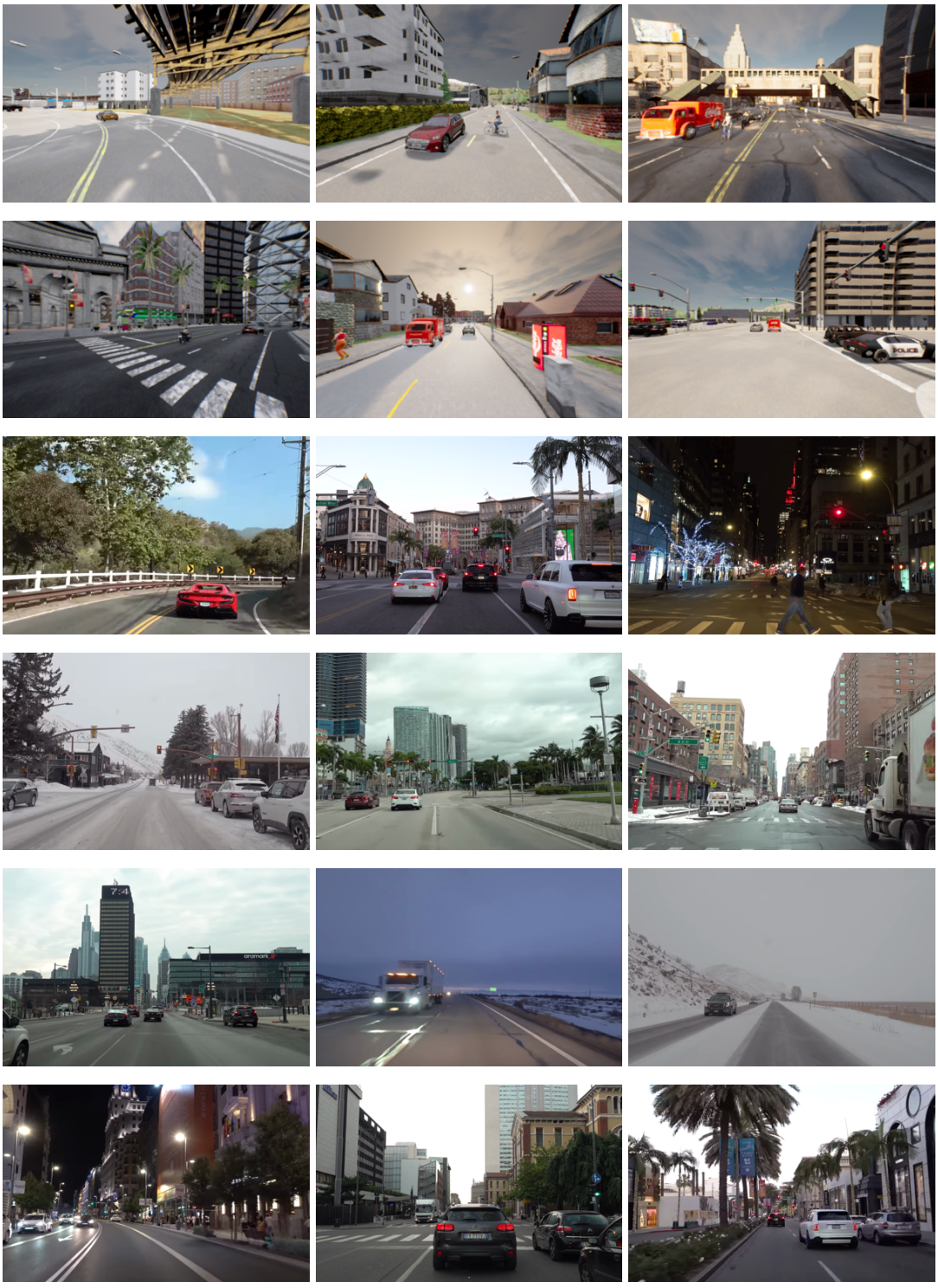

Figure 3.2: First two rows shows examples of driving scenes from the CARLA simulator and rest of the images are extracted from the YouTube driving videos.