2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
September 27 - October 1, 2021. Prague, Czech Republic

Latent Attention Augmentation for Robust Autonomous Driving Policies

Ran Cheng*[1], Christopher Agia*[2], Florian Shkurti[2], David Meger[1], Gregory Dudek[1]

***Abstract*— Model-free reinforcement learning has become a viable approach for vision-based robot control. However, sample complexity and adaptability to domain shifts remain persistent challenges when operating in high-dimensional observation spaces (images, LiDAR), such as those that are involved in autonomous driving. In this paper, we propose a flexible framework by which a policy's observations are augmented with robust attention representations in the latent space to guide the agent's attention during training. Our method encodes local and global descriptors of the augmented state representations into a compact latent vector, and scene dynamics are approximated by a recurrent network that processes the latent vectors in sequence. We outline two approaches for constructing attention maps; a supervised pipeline leveraging semantic segmentation networks, and an unsupervised pipeline relying only on classical image processing techniques. We conduct our experiments in simulation and test the learned policy against varying seasonal effects and weather conditions. Our design decisions are supported in a series of ablation studies. The results demonstrate that our state augmentation method both improves learning efficiency and encourages robust domain adaptation when compared to common end-to-end frameworks and methods that learn directly from intermediate representations.**

I. INTRODUCTION

Can reinforcement learning (RL) one day become practical for reliable visual navigation and autonomous driving in city streets? Entertaining the possibility of an affirmative answer would require, among many other desiderata: data efficiency, robustness, as well as safe exploration and adaptation. In this paper we focus on the first two requirements, and we argue that in order to satisfy them we cannot rely on learning *tabula rasa*. We need to augment the driving policy's input with informative state representations for both low-level control and high-level decision-making. While this remark applies to vision-based robot control in general, herein we focus on the case of autonomous driving and 2D visual navigation.

Attempts to improve data efficiency and robustness in RL models include the following strategies: a) reducing the variance and complexity of observations through transforming the data into representations better suited for policy learning [1], [2], [3]; b) exploration strategies such as space covering exploration or committed exploration [4], [5], [6]; c) improved optimization methods [7] as well as residual learning [8]; d) attempting to learning the dynamics [9], [10] and reward model [11], [12] to do model-based RL. Incorporating an expert-in-the-loop to hand-pick visual state representations that are invariant to domain shifts and relevant to the task, and using them to train robust RL policies is the *de facto* approach to a). However, a straightforward process that informs the optimal choice of representation with respect to a task, target domain, and robotic system has yet to be developed. Moreover, training exclusively with mid-level representations removes the policy's access to salient image-space features that may be informative to the task.

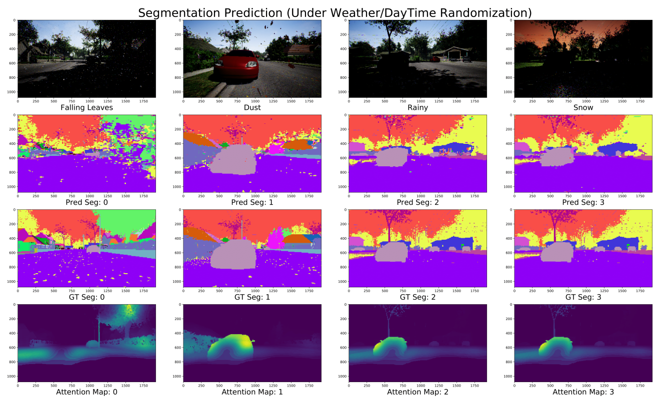


Fig. 1: Segmentation module is robust to changes in brightness, shadows and seasonal effects. Last row shows the attention map extracted along the segment boundaries, which will be used to augment the policy's input.

In this work, we aim to accelerate policy learning and promote robust adaptation by training with augmented visual state representations. By partially decoupling the perception and policy tasks during training, we enable the policy to learn representations that maximize the task objective while simultaneously accepting *advice* in the form of attention representations provided via lightweight perception modules. Composing the policy and expressing it as a function of an intermediate, augmented representation achieves the desirable characteristics for our self-driving agent. As noted in Rasmussen *et al.* [13], Frankel *et al.* [14]: high-level decision making warrants highly abstracted state spaces.

The main contributions of this paper are: (a) we demonstrate that our attention-based state augmentation technique outperforms methods that learn in an end-to-end setup or train directly on intermediate representations; (b) we propose an attention model fusing semantic and spatial information that improves learning efficiency and robustness.

By taking advantage of learning in simulation (utilizing Airsim [15]), we generate a large dataset augmented with permutations of weather conditions and seasonal effects to supervise the training of the segmentation network in the attention pipeline. Fig. 1 shows that the segmentation predic-

* Authors contributed equally.
[1] Mobile Robotics Lab (MRL), McGill University, Montreal, Quebec H3A 2A7, Canada, e-mail: {rancheng, dmeger, dudek}@cim.mcgill.ca.
[2] Robot Vision and Learning Lab (RVL), University of Toronto, Toronto, Ontario L5L 1C6, Canada, e-mail: christopher.agia@mail.utoronto.ca, florian@cs.toronto.edu.

2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-1714-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/IROS51168.2021.9636449

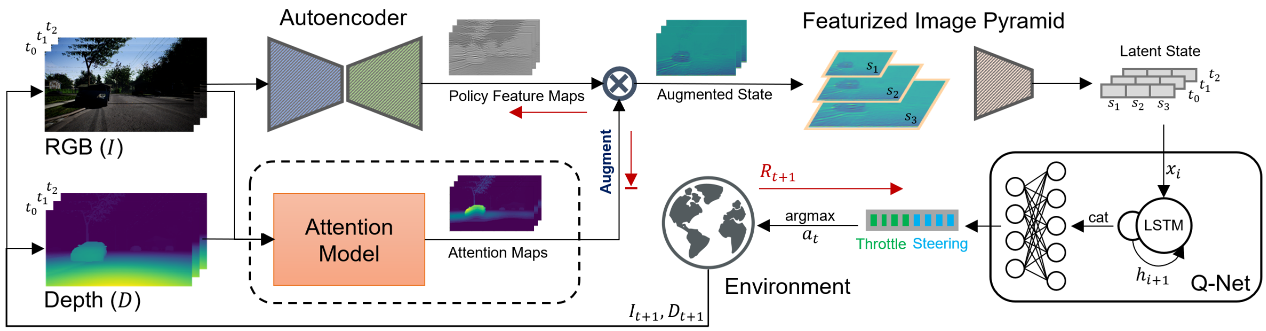


Fig. 2: Model overview. The principal component of the RL pipeline is the fusion of attention maps with the agent's feature maps (i.e. attention augmentation) promoting sample efficiency and robust domain adaptation.

tions remain consistent across varying sources of observation noise. Integrating depth into the pipeline allows for extremely close or far pixel regions (e.g. immediate road surface or sky) that distract the policy to be suppressed, drawing focus to the static and dynamic components of the scene that pose the main challenge for safe navigation. As an alternative to the segmentation-based attention pipeline, we present an unsupervised variant that offers comparable benefits to policy learning as its supervised counterpart. The unsupervised attention pipeline does not require any of the labels provided by the simulator, and thus opens the possibility for prompt deployment in real environments.

II. RELATED WORK

Data efficiency and domain adaptation. Extending RL with deep neural networks has had remarkable success in low-dimensional primitive control policy learning and high-dimensional control tasks [16]. Yet, three key challenges remain outstanding: objective specification, data hunger and domain adaptation. Methods like policy search and policy gradient appeal to objective specification, but concerns of sample complexity makes them unsuitable to train on robots directly [16]. Maximum entropy inverse reinforcement learning [17] is efficient for real robots as it decomposes task demonstrations into hierarchies, but the assumptions on dynamic priors makes it difficult to adapt and transfer. While model-agnostic meta-learning improves domain adaptation, it requires prohibitively large quantities of diverse training data. In the visual adaptation domain, Yu *et al.* [18] achieved one-shot learning with large meta-learning prior libraries learned from diverse videos and transferred into real-robots. Zhu *et al.* [16] bridge the reality gap through domain randomization.

Reinforcement learning in simulation. Learning in simulators and scaling to real-world has become the standard procedure for robot learning in both indoor [19] and outdoor [20] scenarios (supported by large synthetic datasets). Muller *et al.* [3] employ segmentation to train a policy network for PID control, and Zhang *et al.* [21] perform segmentation adaptation by aligning label distributions from the simulators' ground truth to refine predictions in the target domain. Sadeghi *et al.* [19] apply domain randomization throughout policy learning and perform validation on indoor corridors with structural similarities to the training dataset. Trying to learn complex tasks from limited datasets requires data efficiency and a well-defined model. Rather than benefiting speed from off-policy correction [22] in an end-to-end training scheme, one could leverage ground-truth semantic annotations and depth maps (from the simulator) to train an isolated perception module and use it to augment the agent's observations during the policy learning phase. We find that decoupling visual effects from the learning agent's encoder achieves high data efficiency and induces domain adaptation.

Representation Learning in RL. Self-supervised state representation learning aims to improve data efficiency and generalization by encapsulating states into informative latent encodings. Approaches come in two forms: task-agnostic representation learning (reconstruction and contrastive regimes), and task-specific representation learning through bisimulation metrics—prior works have been primarily applied to MuJoCo and Atari control tasks [23], [24]. Yarats *et al.* [25] provided a model-free off-policy method to learn latent image encodings with a β-VAE and optimizes it under the reconstruction and Q-learning objectives. Zhang *et al.* [26] proposed Deep Bisimulation for Control (DBC) which models the behavioural similarity between observations according to the downstream control task, instead of optimizing for task-agnostic representations. They apply their method on a self-driving highway control task and show improved results over learning-from-pixels and representation learning counterparts. However, these models have not yet been demonstrated for self-driving control under domain shifts in complex urban environments. Alternatively, Yang *et al.* [27] models the state space as a graph embedding based on local relationships *a priori*, eliminating the need to model dynamics and requiring no explicit interaction with the environment. While methods for modelling attention have been explored [28], [29], their applications are constrained to vision and primitive control tasks.

RL-based autonomous navigation. Incorporating CNN and RNN structures in RL frameworks has advanced development in high-dimensional, continuous-control robotic learning domains. Notable attempts for efficient learning of self-driving control policies from raw sensory data or semantic representations have required pre-training data from simulation [30], [31] or demonstrations [32], [33], which might be of unreliable quality. Prior to that, Riedmiller *et* *al.* [34] showed that Neural Fitted Q-Iteration could be used to learn to drive a real car in 20mins, however, not on image inputs. Chen *et al.* [35] applied RL to run mobile robots in crowded spaces, and Wang *et al.* [36] trained an RL-based autonomous driving policy in TORCS [37] by customizing the action space and reward function to suit the simulator. Shalev *et al.* [38] modeled the autonomous driving problem within non-MDP settings and reduced the variance of gradient estimation by an option graph (temporal abstraction). Other applications include: predicting control policies with RNNs, using reinforcement learning to solve cross sections of autonomous driving rather than the whole pipeline, and the use state-action representations. Another prominent literature by Huval *et al.* [39] presented an extensive analysis of deep learning techniques for autonomous highway driving. In the context of non-automotive navigation, Manderson *et al.* [40] use a combination of model-free and model-based methods for efficient navigation in complex domains.

III. METHODS

We aim to learn a robust driving policy from sequences of RGB images in diverse and dynamic randomized environments. The proposed reinforcement learning system leverages an attention model which processes each new frame and augments the agent's observations during training, as illustrated in Fig. 2. We propose two attention models: a supervised model that exploits real-time semantic segmentation and depth priors as its primary building blocks, and a fully unsupervised pipeline that replaces the segmentation network with classical image processing techniques. Recurrent networks are used to capture long-horizon dynamics from latent vectors of the agent's augmented state. Details of each component are provided in the following sections.

A. Attention Model

1) Supervised Pipeline: The design of our attention mechanism is based on results of Burr *et al.* [41] suggesting that under the guidance of top-down interests (i.e. high-level decision making), perception lies along the segment boundaries of objects. For a single image, we construct an attention map by inferring its class-wise segmentation, extracting the segmentation boundaries, and scaling them by the inverse proportion of their depths.

The attention model employs a U-Net architecture [42] for semantic segmentation. We assume access to 3D meshes of each object in the simulated environment and convert them to ground truth annotations. By altering the time of day and weather conditions, we generate a diverse dataset with approximately 100k (2TB) training samples and validate over random mini-batches of normal weather and daytime combinations. The U-Net is trained by minimizing the negative log-likelihood of the pixel-wise predictions over all classes [42]. While expressing the state in semantic form encodes topological landmarks and reduces variability, adopting semantic segmentation as the sole perceptual input for RL-based vehicle control is restrictive, as segmentation, along with other classical computer vision objectives,

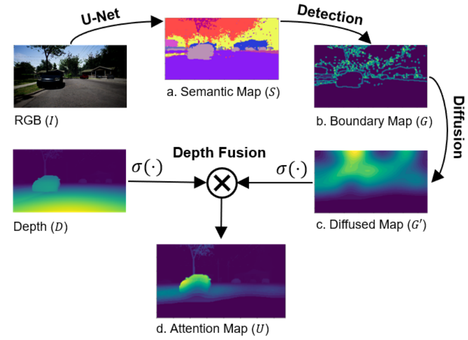


Fig. 3: Attention pipeline. Constructs an attention map from RGB image and depth map inputs drawing the agent's focus to foreground obstacles rather than background context.

abstracts-out salient image features which could enable more performant control. Therefore, we further simplify the state representation into a single-channel attention map, which is of lower complexity and facilitates a simpler state augmentation process (discussed in Sec. III-B). The initial step is extracting object boundaries from the segmented image, S:

$$G_{xy} = \eta\tau\left(|\nabla_x S(x,y)| + |\nabla_y S(x,y)|\right), \ \forall x, y \in \phi \quad (1)$$

$$\tau(x) = \begin{cases} 0, & \text{if } x = 0 \\ 1, & \text{otherwise} \end{cases} \quad (2)$$

Here, $\nabla_x S$ and $\nabla_y S$ denote first-order image gradients applied to the segmentation map in the x and y directions, respectively. This is implemented with an efficient boundary search algorithm, which applies a convolution of two orthogonal $n \times n$ kernels (empirically set to 6×6 pixels) to the map. We observe noisier predictions in local regions that contain multiple classes, and hence, we incorporate the η normalization term to down-weight the extracted boundaries in such kernel regions. The resultant is a high-precision boundary map, G, shown in Fig. 3b.

The semantic map associates edges to class categories, allowing edges to be excluded from the boundary map should they correspond to background classes. However, the boundary map's sparsity makes it unsuitable for drawing attention to homogeneous regions of the semantic map, like those occupied by approaching vehicles or large obstacles. The diffusion process induces a spread of attention outwards from the extracted boundaries, which reduces the effect of sparsity and more accurately encodes the objects' complete geometry. We adopt a diffusion process based on the ADI-method [43] and sweep for 50 iterations to acquire a diffused boundary map denoted by G' (illustrated in Fig. 3c).

Spatial priors reflecting the proximity of obstacles is key to filtering foreground from background and differentiating dynamic from static objects across time, both of which are necessary for safety-critical navigation. Thus, we integrate depth priors into the attention model by fusing a ground truth disparity map (D) with the diffused boundary map after normalizing their quantities with a sigmoid activation.

$$U_{x,y} = \sigma(G'_{x,y}) \cdot \sigma(D_{x,y}),\ \sigma(x) = \frac{1}{1+e^{-\sqrt{x}}} \tag{3}$$

The output is a low-dimensional attention map (U, corresponding to Fig. 3d) which will be used to boost sample efficiency and policy robustness through state augmentation.

2) Unsupervised Pipeline: We also propose an unsupervised variant of the attention pipeline that replaces the semantic segmentation network and boundary extraction layers with a traditional edge detection algorithm. This is accomplished by sweeping a Laplacian of Gaussian kernel (3x3) over a given image in a single convolutional layer, diffusing the extracted boundaries and scaling them by the inverse depth map. Since the classical edge detector is sensitive to environmental artifacts (i.e. falling leaves, snow), the attention maps produced through this procedure contain slightly more noise; although, the diffusion process offsets much of the error. The results provided in Fig. 5a indicate that the minor effects of noise in the attention maps are almost negligible in comparison to the full attention model. Hence, the unsupervised attention pipeline is a viable approach when segmentation is undesired or supervision cannot be acquired.

B. Attention State Augmentation

The attention models emphasize navigation obstacles in the vehicles field of view and prioritizes them based on proximity. High-variance and the presence of distractors that deter domain adaptation and slow convergence when learning *tabula rasa* are addressed through the invariance of the attention maps across a wide range of environmental conditions. Furthermore, the inference time of the full attention pipeline is fast enough to apply on real robots (~10fps on Nvidia Jetson TX2). Although the system utilizes RGB-D sensor data, the dependency on depth sensors could be removed by use of off-the-shelf depth prediction networks [44], which have shown remarkable progress in terms of accurate and scale consistent inference in challenging domains. Note that the training and configuration of modules in the attention pipelines—the U-Net segmentation network, boundary extraction and diffusion, and inverse depth scaling—are decoupled (i.e. frozen) from all other modules that learn from sparse rewards within the reinforcement learning framework.

A used technique dubbed *attention augmentation* improves upon the robustness and sample efficiency of autonomous driving policies that learn from intermediate representations (e.g. segmentation maps, optical flow, surface normals) [1], [2]. Attention augmentation enables us to exploit the generated attention maps as a task-centric prior while allowing the policy to learn representations through sparse rewards in parallel. As depicted in Fig. 2, this technique is expressed as an element-wise scaling of the attention map with the feature map reconstructed by an unsupervised autoencoder, and corresponds to augmenting the driving policy's observation in the latent image space. We utilize a featurized image pyramid [45] composed of lightweight VGG16 networks to encode both coarse and fine-grained descriptors of the augmented state into a latent vector representative of a single frame. Sequences of these latent states are then processed by an LSTM to model the temporal properties of the scene.

C. Reinforcement Learning Model

In simplifying the state space via attention representations the navigation system can effectively learn from sparse rewards. Furthermore, evading the taxing memory costs of learning from high-dimensional RGB images provides a strong boost in convergence speeds, in addition to the benefits offered by recurrent memories (LSTM) which back-propagates rewards to batched samples in a sliding window.

Within the DQN [46] training sessions, we cast our model into the Partially Observable Markov Decision Process framework, which is a tuple of $(\mathcal{S}, \mathcal{A}, \mathcal{T}_{sa}, R, \Omega, \mathcal{O}, \gamma)$: state space $\mathcal{S}$, action space $\mathcal{A}$, transition probabilities $\mathcal{T}_{sa}$, reward function R, observation space Ω, conditional observation probabilities $\mathcal{O}$, and discount factor γ. The action space is a discrete set of throttle-steering combinations. The objective function and policy π are expressed as:

$$J(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R_t | s_0\right] \tag{4}$$

$$\pi^* = \text{argmax}_{\pi} J(\pi) \tag{5}$$

The reward function below is comprised of three components: R_{gdis} provides a reward in proportion to the distance travelled between initial state s_0 and terminal state s_T, R_{dis} rewards precise lane following as a function of the vehicle's distance relative to center of the road, and R_{speed} simply encourages the agent to drive at higher speeds. In the last term, δ_a penalizes the agent for frequently changing actions, and scales the penalty by normalization constant η. $\lambda \in [0, 1]$ is empirically set to balance navigation quality against speed.

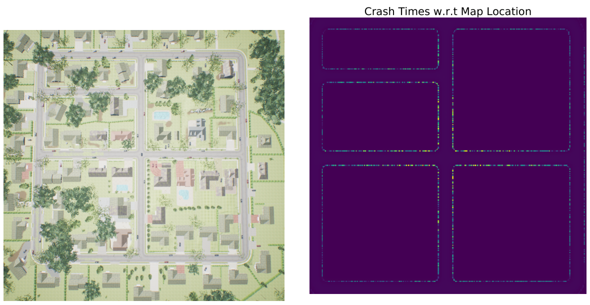
$$R_t = \lambda(R_{gdis,t} + R_{dis,t}) + (1-\lambda)R_{speed,t} - \eta\delta_a \tag{6}$$

For a given time step t, we define an augmented attention state as u_t, all other state parameters as k_t, and the action taken as a_t. The goal of the model is to estimate the Q-function $Q^{\pi}_{w}(s_t, a_t)$, where $s_t = [u_t, ..., u_{t-m}, k_t]$. We do so by minimizing the following loss function:

$$L_t(w) = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{D}}[(q_{t+1} - Q^{\pi}_{w}(s_t, a_t))^2] \tag{7}$$
$$q_{t+1} = R_t + \gamma \max_{a_{t+1}} Q^{\pi}_{w'}(s_{t+1}, a_{t+1})$$

In Eq. (7), q_{t+1} is the target value generated by the fixed target network weights $w^{'}$ from the previous iteration. With the learned Q-function, the agent may then select the action based on $Q(s_t, a_t)$ to maximize expected rewards. As for the policy function $\pi_\theta(s)$ in DDPG [47] experiments, we update the policy network weights θ with the partial derivative from accumulated reward expectations:

$$\frac{\partial J}{\partial \theta} = \frac{\partial Q^{\pi}(s,a)}{\partial a}\bigg|_{a=\pi_\theta(s)} \frac{\partial \pi_\theta(s)}{\partial \theta} \tag{8}$$



(a) AirSim Neighborhood Training environment. This diverse map offers us the essential elements for normal daily driving.

(b) Heat map of crash locations averaged over 7 runs. Note that we always initialize the vehicle at the map center with randomized orientations and goal locations.

Fig. 4: Experiment environment overview.

D. Recurrent Architecture

Our initial experiments indicated that DQN and DDPG were able to learn effective steering and speed control policies for a single route, but diverged through episodic randomization. To remedy this, we represent our Q-function as an LSTM followed by several fully-connected layers to process the concatenated outputs of the LSTM cells. Thus, Q-values are predicted as $Q(s_t = [u_t, ..., u_{t-m}, k_t], a_t)$, where u_{t-i} for $i \in \{0, .., m\}$ are the latent encodings of augmented attention maps from the current and previous time steps. This formulation enables the policy to hone-in on the dynamical and geometrical properties of a scene across sequential frames, rather than memorizing the state-action mapping of single frame states.

When training the recurrent model, we utilize a batched sliding window strategy that corresponds to the number of LSTM units deployed. However, the strong correlation of input samples within the sliding window scope introduces gradient bias, thus we construct a replay buffer with five windows truncated by time and backpropagate rewards to the corresponding window after each batch in processed. As this method results in over-fitting in the early stages of training, we apply both concrete-dropout and L2 regularization.

The best results were obtained when using a single LSTM layer with 128 memory cells. Despite the initial decrease in learning efficiency due to model complexity, the advantage of time-series prediction begins to gradually take effect during training and ultimately contributes to the final convergence.

IV. EXPERIMENTAL RESULTS

We conduct our experiments on the AirSim Simulator's Neighborhood Environment shown in Fig. 4a and average the results over 7 independent training iterations. Each episode terminates when the vehicle drives safely to arrive at a randomly generated destination a set distance away or when a collision occurs; Fig. 4b depicts the averaged crash locations. The average attention inference frame rate on a PC with a GTX1080Ti graphics card is $20 \pm 3fps$ and is reduced to $9 \pm 4fps$ on an Nvidia Jetson TX2. Additional optimization efforts would be required for real-time operation on robots with limited computational resources. Hyperparameters for our proposed DRL model include: 32 batch size, $1e^{-4}$ learning rate, 128 LSTM memory, $50k$ replay buffer, 1.0 epsilon start, 0.01 epsilon end, $100k$ epsilon decay steps, 0.1 epsilon decay rate, and $1k$ target network update interval.

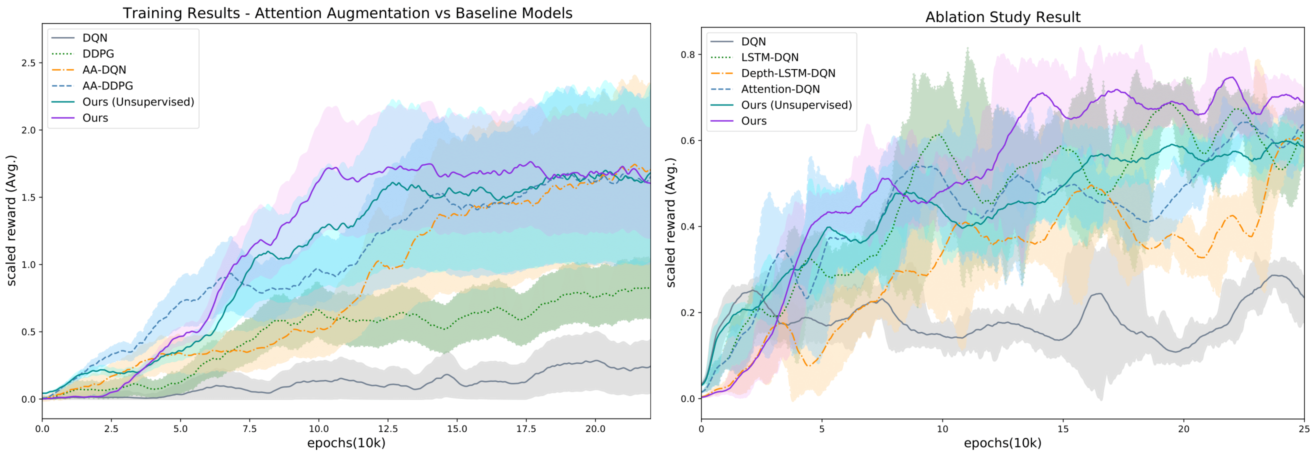
A photo-realistic environment combined with a "physics engine" (simulator) helps bridge the transfer to real-robots. We thus focus on synthesizing a dataset with key canonical classes of environmental perturbation, notable with images diverse in times of day, weather condition, camera angles, and noise (salt-and-pepper). Training the RL agent under these conditions promotes robust behaviour when encountering objects of irregular shapes and textures in the environment. As policy learning in randomized simulated environments for safety-critical navigation has been previously shown to scale up to real-world images [19], we anticipate that our model, and more specifically, the segmentation network will have strong transfer characteristics and only require re-training in environments with extreme visual differences or geometrical layouts.

Quantitative Evaluation. Fig. 5a compares our proposed system with four baselines; two traditional RL methods, each configured with an end-to-end architecture and with the segmentation-based attention pipeline. The reported scaled reward is simply the accumulated return over a training episode. Notice that our method converges to the highest reward lower-bound at substantially faster rates, while the DQN and DDPG baselines which process high-dimension RGB inputs struggle to converge and develop random exploratory behaviours. Modifying the DQN and DDPG models with our attention augmentation pipeline results in a significant increase in lower-bound convergence, suggesting that learning-from-pixels is limited in its capacity to extract informative and compact state representations for autonomous driving. Moreover, computing gradient updates from randomly sampled single-state trajectories (τ) as so: $\nabla_w J(w) \simeq \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\nabla_w \log\pi_w(\tau)r(\tau)$ leads to undesired variance and learning instability. Thus, leveraging the temporal relationships of sequential latent state encodings with our recurrent Q-network (LSTM) leads to faster convergence with reduced variance across all models.

Powered by our attention pipeline, the reinforcement learning agent exhibits an intriguing behaviour. Particularly, its decisions follow the path of least action, reacting to environment changes only when objects come within a proximity concerning safe navigation. With the agent focusing its computational efforts on only attention-worthy objects, we observe the desired smooth yet safe driving policy.

TABLE I: Scaled total reward in different test environments. Note that we average the rewards across 20 runs in different times-of-day for each environmental category.

			Scaled Average Total Reward				
Model	**Description**	**Configuration**	**Leaves**	**Fog**	**Rain**	**Dust**	**Snow**
DQN [46]	Value Func.	End-to-end	0.41	0.08	0.61	0.38	0.24
		Attention Aug.	0.77	0.93	0.89	0.73	0.54
DDPG [37]	Policy Grad.	End-to-end	0.65	0.22	1.33	0.89	0.56
		Attention Aug.	1.57	1.32	0.90	**1.02**	1.39
Unsupervised	Hybrid	Edge+LSTM-DQN	1.04	1.55	1.25	**1.02**	1.13
Ours	Hybrid	AA+LSTM-DQN	**1.59**	**1.79**	**1.52**	0.93	**1.46**



(a) Learning efficiency (smooth scale: 0.92) of our method against baselines. The plots are averaged over 7 runs with different random seeds (error bars indicate 1-σ) and identical hyperparameters. Here, AA refers to Attention Augmentation.

(b) Ablation study: Contribution of each individual component in our model averaged over 7 runs. Notice the oscillations without attention maps, as learning from segmentation as an intermediate representation is subject to higher variance.

Fig. 5: Experimental results overview.

We further compare the robustness of our model to the baselines by training all models in bright day-time conditions, then evaluating their performance in challenging and unseen conditions. The results in Table. I show the performance improvement of our proposed method across most weather conditions and the notable increase in robustness when applying attention augmentation to the baselines. We also notice a slight drop in performance when traditional edge detection is employed instead of semantic segmentation boundaries in scenes rich of distractors. In more standard conditions, these two methods perform equivalently.

TABLE II: Ablation study of our architecture. Return reflects the mean reward obtained on 10 episodes with a horizon of 10,000 steps. Results are averaged over 7 runs.

Model	DQN	Depth	Seg.	AM	AAM	FIM	LSTM	Return
Baseline	✓							0.238
DQN + Depth	✓	✓						0.219
DQN + Seg.	✓		✓					0.343
DQN + AM	✓			✓				0.439
DQN + AAM	✓				✓			0.627
DQN + AAM + FIM	✓				✓	✓		0.611
DQN + AAM + FIM + LSTM	✓				✓	✓	✓	**0.655**

AM: Attention Map
AAM: Augmented Attention Map
FIM: Featurized Image Pyramid, pretrained VGG 16 ConvNet

Ablation Study. The ablation study presented in Table. II emphasizes the contribution of each individual component in our system by excluding all but one component and evaluating its effect on self-driving performance. The results indicate that state augmentation with robust attention maps is approximately twice as effective in comparison to learning-from-pixels or mid-level representations.

V. FUTURE WORK

The performance of the proposed model is attributed to attention augmentation; accuracy of the attention representations depend primarily on the deep segmentation network which lacks interpretability (black-box in nature). Such dependencies reduce the robustness of the overall system, and thus, future work includes diversifying the input signals, using segmentation as one of several auxiliary methods to more robustly produce high-level representations. In addition, reliance on ground-truth depth maps can be circumvented with depth prediction networks. We also expect an increase in learning efficiency by adopting modern imitation learning or inverse RL techniques. We aim to explore and potentially integrate them, as they open up promising opportunities to solve complex robotic tasks in an effective manner.

VI. CONCLUSION

In this paper, we investigated the effect of augmented attention representations on robustness and efficiency in deep reinforcement learning for the task of autonomous robot navigation; more specifically, self-driving in simulated urban environments. We proposed an efficient pipeline that abstracts RGB image features into an attention map, which is used to augment the policy's observations during training. By encoding the augmented feature maps into multi-scale latent vectors and aggregating them across several frames with an LSTM, the agent is induced to act upon the underlying dynamics of the environment without disregarding global context. Our results illustrate a notable improvement over common end-to-end architectures that do not explicitly model state representations, operating on high-dimensional images and consuming potentially redundant or unrelated representations which burdens efficiency and produces learning ambiguity. We showed that these attention maps can be constructed either with or without supervision, opening the possibility for deployment in the field.

REFERENCES

[1] B. Zhou, P. Krähenbühl, and V. Koltun, "Does computer vision matter for action?" *arXiv preprint arXiv:1905.12887*, 2019.

[2] B. Chen, A. Sax, G. Lewis, I. Armeni, S. Savarese, A. Zamir, J. Malik, and L. Pinto, "Robust policies via mid-level visual representations: An experimental study in manipulation and navigation," *arXiv preprint arXiv:2011.06698*, 2020.

[3] M. Müller, A. Dosovitskiy, B. Ghanem, and V. Koltun, "Driving policy transfer via modularity and abstraction," *arXiv preprint arXiv:1804.09364*, 2018.

[4] A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson, "Maven: Multi-agent variational exploration," in *Advances in Neural Information Processing Systems*, 2019, pp. 7613–7624.
[5] T. Wang, H. Dong, V. Lesser, and C. Zhang, "Roma: Multi-agent reinforcement learning with emergent roles," in *Proceedings of the 37th International Conference on Machine Learning*, 2020.
[6] M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castañeda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, *et al.*, "Human-level performance in 3d multiplayer games with population-based reinforcement learning," *Science*, vol. 364, no. 6443, pp. 859–865, 2019.
[7] P. Henderson, R. Islam, P. Bachman, J. 22 Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[8] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine, "Residual reinforcement learning for robot control," 2018.
[9] K. Kang, S. Belkhale, G. Kahn, P. Abbeel, and S. Levine, "Generalization through simulation: Integrating simulated and real data into deep reinforcement learning for vision-based autonomous flight," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6008–6014.
[10] Y. Du and K. Narasimhan, "Task-agnostic dynamics priors for deep reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 1696–1705.
[11] Y. Yang, K. Caluwaerts, A. Iscen, T. Zhang, J. Tan, and V. Sindhwani, "Data efficient reinforcement learning for legged robots," in *Conference on Robot Learning*. PMLR, 2020, pp. 1–10.
[12] R. Kaushik, K. Chatzilygeroudis, and J.-B. Mouret, "Multi-objective model-based policy search for data-efficient learning with sparse rewards," in *Conference on Robot Learning*. PMLR, 2018, pp. 839–855.
[13] J. Rasmussen, "The role of hierarchical knowledge representation in decisionmaking and system management," *IEEE Transactions on systems, man, and cybernetics*, no. 2, pp. 234–243, 1985.
[14] C. B. Frankel and M. D. Bedworth, "Control, estimation and abstraction in fusion architectures: Lessons from human information processing," in *Proceedings of the Third International Conference on Information Fusion*, vol. 1. IEEE, 2000, pp. MOC5–3.
[15] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, 2017. [Online]. Available: https://arxiv.org/abs/1705.05065
[16] Y. Zhu, Z. Wang, J. Merel, A. Rusu, T. Erez, S. Cabi, S. Tunyasuvunakool, J. Kramár, R. Hadsell, N. de Freitas, *et al.*, "Reinforcement and imitation learning for diverse visuomotor skills," *arXiv preprint arXiv:1802.09564*, 2018.
[17] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, *et al.*, "Maximum entropy inverse reinforcement learning." in *Aaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.
[18] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine, "One-shot imitation from observing humans via domain-adaptive meta-learning," *arXiv preprint arXiv:1802.01557*, 2018.
[19] F. Sadeghi and S. Levine, "Cad2rl: Real single-image flight without a single real image," *arXiv preprint arXiv:1611.04201*, 2016.
[20] T. Manderson, R. Cheng, D. Meger, and G. Dudek, "Navigation in the service of enhanced pose estimation," in *International Symposium on Experimental Robotics (ISER)*, 2018.
[21] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2020–2030.
[22] O. Nachum, S. S. Gu, H. Lee, and S. Levine, "Data-efficient hierarchical reinforcement learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 3303–3313.
[23] C. Gelada, S. Kumar, J. Buckman, O. Nachum, and M. G. Bellemare, "Deepmdp: Learning continuous latent space models for representation learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2170–2179.
[24] A. X. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," *arXiv preprint arXiv:1907.00953*, 2019.
[25] D. Yarats, A. Zhang, I. Kostrikov, B. Amos, J. Pineau, and R. Fergus, "Improving sample efficiency in model-free reinforcement learning from images," *arXiv preprint arXiv:1910.01741*, 2019.
[26] A. Zhang, R. McAllister, R. Calandra, Y. Gal, and S. Levine, "Learning invariant representations for reinforcement learning without reconstruction," *arXiv preprint arXiv:2006.10742*, 2020.
[27] G. Yang, A. Zhang, A. Morcos, J. Pineau, P. Abbeel, and R. Calandra, "Plan2vec: Unsupervised representation learning by latent plans," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 935–946.
[28] I. Sorokin, A. Seleznev, M. Pavlov, A. Fedorov, and A. Ignateva, "Deep attention recurrent q-network," *arXiv preprint arXiv:1512.01693*, 2015.
[29] K. Choromanski, D. Jain, J. Parker-Holder, X. Song, V. Likhosherstov, A. Santara, A. Pacchiano, Y. Tang, and A. Weller, "Unlocking pixels for reinforcement learning via implicit attention," *arXiv preprint arXiv:2102.04353*, 2021.
[30] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "Deep reinforcement learning framework for autonomous driving," *Electronic Imaging*, vol. 2017, no. 19, pp. 70–76, 2017.
[31] X. Pan, Y. You, Z. Wang, and C. Lu, "Virtual to real reinforcement learning for autonomous driving," *arXiv preprint arXiv:1704.03952*, 2017.
[32] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8248–8254.
[33] J. Hawke, R. Shen, C. Gurau, S. Sharma, D. Reda, N. Nikolov, P. Mazur, S. Micklethwaite, N. Griffiths, A. Shah, *et al.*, "Urban driving with conditional imitation learning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 251–257.
[34] M. Riedmiller, M. Montemerlo, and H. Dahlkamp, "Learning to drive a real car in 20 minutes," in *2007 Frontiers in the Convergence of Bioscience and Information Technologies*, 2007, pp. 645–650.
[35] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1343–1350.
[36] S. Wang, D. Jia, and X. Weng, "Deep reinforcement learning for autonomous driving," *arXiv preprint arXiv:1811.11329*, 2018.
[37] B. Wymann, E. Espié, C. Guionneau, C. Dimitrakakis, R. Coulom, and A. Sumner, "Torcs, the open racing car simulator," *Software available at http://torcs. sourceforge. net*, vol. 4, no. 6, 2000.
[38] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "Safe, multi-agent, reinforcement learning for autonomous driving," *arXiv preprint arXiv:1610.03295*, 2016.
[39] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, *et al.*, "An empirical evaluation of deep learning on highway driving," *arXiv preprint arXiv:1504.01716*, 2015.
[40] T. Manderson, J. C. Gamboa, S. Wapnick, J.-F. Tremblay, D. Meger, and G. Dudek, "Vision-based goal-conditioned policies for underwater navigation in the presence of obstacles," in *Proceedings of Robotics: Science and Systems*, July 2020. [Online]. Available: https://arxiv.org/abs/2006.16235
[41] D. C. Burr, M. C. Morrone, and D. Spinelli, "Evidence for edge and bar detectors in human vision," *Vision research*, vol. 29, no. 4, pp. 419–431, 1989.
[42] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
[43] V. Simoncini, "Computational methods for linear matrix equations," *SIAM Review*, vol. 58, no. 3, pp. 377–441, 2016.
[44] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3828–3838.
[45] Y. Pang, T. Wang, R. M. Anwer, F. S. Khan, and L. Shao, "Efficient featurized image pyramid network for single shot detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7336–7344.
[46] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.
[47] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.