Ground then Navigate: Language-guided Navigation in Dynamic Scenes

Anonymous ARR submission

Abstract

We investigate the problem of Vision-and-Language Navigation (VLN) in the context of autonomous driving in outdoor settings. We solve the problem by explicitly grounding the navigable regions corresponding to the textual command. At each timestamp, the model predicts a segmentation mask corresponding to the intermediate or the final navigable region. Our work contrasts with existing efforts in VLN, which pose this task as a node selection problem, given a discrete connected graph corresponding to the environment. We do not assume the availability of such a discretised map. Our work moves towards continuity in action space, provides interpretability through visual feedback and allows VLN on commands like 'park between the two cars", requiring finer manoeuvres. Furthermore, we propose a novel meta dataset CARLA-NAV to allow efficient training and validation. The dataset comprises pre-recorded training sequences and a live environment for validation and testing. We provide extensive qualitative and quantitive empirical results to validate the efficacy of the proposed approach.

1 Introduction

Humans have exceptional navigational abilities, which, combined with their visual and linguistic prowess, allow them to perform navigation based on the linguistic description of the objects of interest in the environment. This is in direct consequence of the human capability to associate visual elements with their linguistic descriptions. Referring Expression Comprehension (REC) (Rohrbach et al., 2016) and Referring Image Segmentation (RIS) (Hu et al., 2016a) are two tasks for associating the visual objects based on their linguistic descriptions using bounding-box-based and pixel-based localizations, respectively. However, it is non-trivial to utilize these localizations directly for a navigation task (Deruyttere et al., 2019). For example, consider the linguistic command "take a right turn from the intersection," an object-based localization is not usable for navigation as it does not answer the question "which region" on the road to navigate to. To solve the aforementioned issue, the task of Referring Navigable Regions (RNR) was proposed in (Rufus et al., 2021) to localize the navigable regions in a static front camera image on the road corresponding to the linguistic command.

Figure 1: A major limitation of single image based grounding methods is that they fail if the language command is not immediately visible, which restricts these methods to be used for VLN. Here, we show an example result using the RNR model and on an image from their dataset (Rufus et al., 2021). The model accurately localizes the black car (middle), however, the it completely confuses when asked to predict the left turn, which does not exist in the current view.

Although these single image-based visual grounding methods (Rufus et al., 2021; Hu et al., 2016a; Deruyttere et al., 2019) showcase the excellent ability of neural networks to correlate visual and linguistic data, they are still limited in many ways. These methods are trained on carefully paired data, assuming that the region to be grounded is always visible in the frame. They give an erroneous output when the region to be grounded is not currently visible, is occluded, or goes out of frame (as the carrier moves). Such scenarios are part of everyday language-guided navigation; for instance, consider a command "Take a right once you see the traffic signal" the traffic signal here may not be immediately visible. Figure 1 illustrates one such scenario, where the single

image-based RNR method gives an incorrect output that is not correlated with the linguistic command. As a second major limitation, single image-based predictions (Rufus et al., 2021) are devoid of any temporal context (short term or long term), which is crucial in successful navigation, especially in a dynamically changing environment. Finally, since single image methods are evaluated on frames from pre-recorded videos, they cannot be validated appropriately on their ability to complete the entire episode (from start to desired finish). Our work addresses these limitations and re-formulates the RNR approach to perform language-guided navigation in a dynamically changing environment by grounding intermediate navigable regions when the referred navigable region is not visible.

Our work also contrasts with the prior art for language-guided navigation in both indoor and outdoor environments. Most existing works on indoor navigation (Anderson et al., 2018; Shrestha et al., 2020; Zang et al., 2018) assume that the navigational environment is fully known. This allows them to discretize the known map into a graphical representation, where the nodes are the set of navigable regions (landmarks) that the agent can navigate given the linguistic command. However, such an approach is not practical for outdoor settings (as studied in our work) where the environment is unknown. Moreover, even if the environment is known, discretization of the maps is not feasible when more refined localization and manoeuvres are required (e.g. “stop beside the person with a red cap").

Finer control remains a challenge in indoor and outdoor VLN methods, which model navigation as a selection from a set of discrete actions (Schumann and Riezler, 2022; Zhu et al., 2021; Xiang et al., 2020) or as a reinforcement learning problem (Anderson et al., 2018; Fu et al., 2019). For instance, one of the commonly used Touchdown dataset (Chen et al., 2019) consists of pre-recorded google street view images and allows navigation across street views by choosing from a set of four discrete actions, i.e. FORWARD, RIGHT, LEFT and STOP. Discretizing the action space (and the environment) limits the type of navigational manoeuvres that can be performed. For instance, these methods (Schumann and Riezler, 2022; Zhu et al., 2021; Xiang et al., 2020) cannot be used for commands like "park between the two cars on the right", which require fine-grained control of the car.

The aforementioned issues become apparent in a dynamically changing environment, where fine-grained control of the car’s navigation and a fully navigable environment is required to adapt to the dynamic surroundings and perform navigational manoeuvres based on the linguistic command. In this paper, we present a novel meta-dataset in the CARLA environment (Dosovitskiy et al., 2017) for outdoor navigation, which addresses the limitations associated with the existing navigation datasets. Additionally, the visual grounding-based approach combined with a planner allows us to have a fine-grained control over the vehicle as it enables navigation to any drivable region on the road.

Another concern with the current sequence-to-sequence and reinforcement learning approaches is that their predictions are not interpretable. It is non-trivial to understand their predictions as there is no feedback. Instead, in the visual-grounding-based approach, there is visual feedback associated with each prediction in terms of the segmentation mask corresponding to the navigable region on the road. We take a step forward and also predict the short term intermediate trajectories, using a novel multi-task network. Moreover, we perform live inference on the proposed meta-dataset in a dynamic environment. To the best of our knowledge, this is the first attempt towards live language-based navigation in outdoor environments. Lastly, we conduct comprehensive qualitative and quantitative ablations to validate the effectiveness of our approach. To summarize, the main contributions of this paper are the following:

- We present a vision language navigation tool, CARLA-NAV, on the CARLA simulator which provides fine-tuned control of the vehicle to execute various language-based navigational manoeuvres.
- We propose a novel multi-task network for trajectory prediction and per-frame RNR tasks in dynamic outdoor environments. The prediction for each task is explainable and interpretable in the form of segmentation masks.
- We perform real-time navigation in the CARLA environment with a diverse set of linguistic commands.
- Finally, extensive qualitative and quantitative ablations are performed to validate the practicality of our approach.

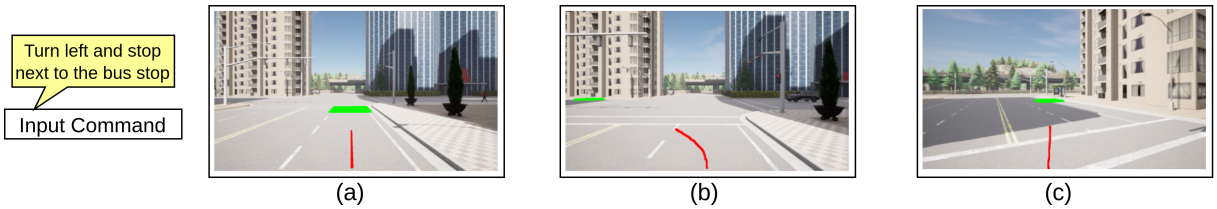

Figure 2: Ground truth annotations of three sampled frames from an episode of CARLA-NAV dataset. The textual command for the episode is shown on the top. The green rectangle illustrates the navigable region and the red curve corresponds to the short future trajectory. (a) At the start, the left turn or the final navigable region is not visible, so a straight path is chosen as intermediate mask; (b) the intermediate mask corresponding to the left turn; (c) final navigable region (stop next to bus stop).

2 Related Work

2.1 Visual Grounding

Visual grounding aims to help associate the linguistic description of entities with their visual counterparts by localizing them visually. There are two prevalent approaches for visual grounding based on the type of localization used. Proposal-based localization is formally referred to as Referring Expression Comprehension (REC). Most methods in REC follow a propose-then-rank strategy, where the ranking is done using similarity scores (Rohrbach et al., 2016; Hu et al., 2016b; Plummer et al., 2018; Rufus et al., 2020) or through attention-based methods (Deng et al., 2018; Zhang et al., 2018; Yang et al., 2019; Qiu et al., 2020). The other approach is to localize the objects by their pixel-level segmentation mask, formally known as Referring Image Segmentation (RIS). In RIS, methods use different strategies to fuse the spatial information of the image with the word-level information of the language query (Shi et al., 2018; Ye et al., 2019; Huang et al., 2020; Jain and Gandhi, 2022). Recently, (Rufus et al., 2021) proposed the Referring Navigable Region (RNR) task to directly localize the navigable regions on the road corresponding to the language commands. However, their work limits to predictions on static images in pre-recorded video sequences (Caesar et al., 2020). We propose reformulating the RNR task for dynamic outdoor settings and performing real-time navigation based on language commands.

2.2 Language-based Navigation

Majority of efforts on Vision Language Navigation (VLN) have focused on the indoor scenario. Availability of interactive synthetic environments has played a key role in indoor navigation research. The environments are either designed manually by 3D artists (Kolve et al., 2017; Wu et al., 2018) or are constructed using RGB-D scans of actual buildings (Chang et al., 2017). Existing methods have approached language guided navigation in variety of ways, including imitation learning (Nguyen et al., 2019; Nguyen and Daumé III, 2019), behavior cloning (Das et al., 2018), sequence-to-sequence translation (Anderson et al., 2018) and cross-modal attention (Cornia and Cucchiara, 2019). In these methods, the navigation is modelled as traversing an undirected graph, presuming known environment topologies. In recent work, (Krantz et al., 2020) suggest that the performance in prior 'navigation-graph' settings may be inflated by strong implicit assumptions. Hu *et al.* (Hu et al., 2019) questions the role of visual grounding itself by highlighting that models which only use route structure outperform their visual counterparts in unseen new environments. Most indoor VLN methods are also hindered by limiting the output to a discretized action space (Irshad et al., 2021).

For outdoor VLN, Sriram *et al.* (Sriram et al., 2019) use CARLA environment to perform navigation as waypoint selection problem, however, their work limits to only turning actions. The Talk2Car dataset limits to localizing the referred object (Deruyttere et al., 2019). Another line of work focuses on interactive navigation environment of Google Street View (Mirowski et al., 2018). The Touchdown dataset (Chen et al., 2019) proposes a task of following instructions to reach a goal (identifying a hidden teddy bear). Map2Seq dataset (Schumann and Riezler, 2020) learns to generate navigation instructions that contain visible and salient landmarks from human natural language instructions. The navigation on both datasets is modelled as node selection in a discrete connectivity graph. Most methods (Schumann and Riezler,

2022; Zhu et al., 2021; Xiang et al., 2020) using these datasets, solve outdoor VLN as sequence to sequence translation in a discrete action space. The role of vision modality remains illusive when tested in unseen areas (Schumann and Riezler, 2022). In this work, we propose a paradigm shift towards utilizing RNR-based approaches for VLN. The explicit visual grounding forces the network to utilize visual information. Integrating with a local planner, the navigation is performed in a continuous space, without any reliance on the map information.

3 Dataset

The proposed CARLA-NAV dataset was curated using the open-source Carla Simulator for autonomous driving research. It contains episodic level data, where each episode consists of a language command and the corresponding video from Carla Simulator of navigation towards the final goal region described by the command. Example ground truth annotations from an episode from the CARLA-NAV dataset are shown in Figure 2. The ground truth segmentation mask for each frame either corresponds to the final or an intermediate navigable region. Each frame is additionally annotated with a plausible future trajectory of the vehicle in the next few frames.

The dataset includes video sequences captured in 8 different maps, 14 distinct weather conditions, and a diverse range of vehicles and passengers in the environment. The language commands in our dataset contain detailed visual descriptions of the environment and describe a wide range of manoeuvres. In some cases, there are multiple manoeuvres in a single command, e.g., "stop for the traffic light, then take a right turn and park near the bus stand." Overall, the training split of the dataset consists of 500 episodes, the validation split consists of 25 episodes, and the test split consists of 34 episodes. During data collection, in each episode, the vehicle is spawned in a randomly selected map at a random position. During the training phase, we use the pre-recorded sequence for the network training. However, during the inference phase on validation and test splits, for each episode, we spawn the vehicle at the corresponding starting location, and the navigation is performed based on network prediction and not on the pre-recorded sequences.

3.1 Dataset Creation

We created a data-collection toolkit on top of Carla's API and plan to open-source it upon acceptance. The data collection process is illustrated in Figure 3. It happens in a two step manual process: (a) providing a language command through a text prompt and (b) navigating the Carla environment based on the language command through mouse clicks. The 2D point corresponding to the mouse click in the front view of the car is transformed into the 3D world coordinates using Inverse Projective transform; and this 3D position is passed as input to the local planner to navigate the CARLA environment. We use CARLA's default rule-based planner for our case; however, this can easily be replaced with more sophisticated planners like RRT* or end-to-end imitation learning models like NEAT (Chitta et al., 2021). An episode comprise of multiple mouse clicks, until the final navigable region is not visible in the front view. These intermediate mouse clicks signify the intermediate navigable regions, and the last mouse click depicts the final goal region corresponding to the command. The mouse clicks are converted into segmentation masks by

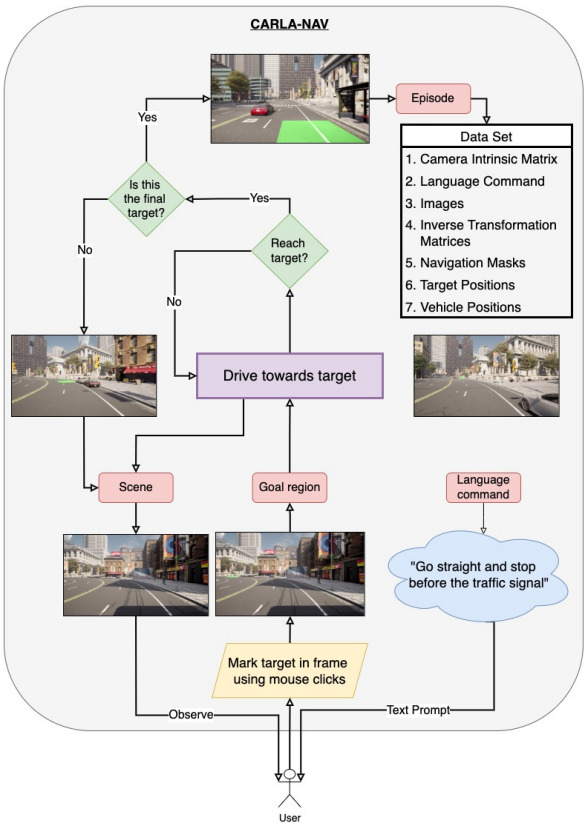

Figure 3: Pipeline for the data collection procedure. The user is provided with a language command conforming to the current environment state. Based on the linguistic command, the user successively selects the navigable region on the road until the final destination corresponding to the command is reached.

drawing a 3m $\times$ 4m rectangle (approx size of a car) in the top view, centered at the mouse click. The rectangle is then projected in the front-camera. Similarly, for the future trajectory prediction task, we take the 3D position of the vehicle in the successive frames and project the 3D positions to the front-camera image using the projective transformation. We treat trajectory prediction as a dense prediction task which makes it interpretable during navigation. Overall, only the language commands and mouse clicks require manual effort, the rest of the data-collection process is fully automated.

4 Problem Statement

Given an input of video frames $V = \{v_{t-k}, v_{t-k+1}, ...v_t\}$, contextual historical trajectory P and a language command $L = \{l_1, l_2, ...l_N\}$, where t is the current timestamp, k is the window size for historical frames and N is the maximum number of words in the linguistic expression, the goal is to predict the navigable mask y_t and the future trajectory mask z_t corresponding to the frame from current timestamp, i.e. v_t. The contextual trajectory P is utilized to ensure that the network gets the contextual information necessary to identify which part of the linguistic command has been executed. For example, if the linguistic command is "turn left and park near the blue dustbin", the contextual trajectory will provide information regarding the trajectory already taken by the vehicle, i.e. whether the "left turn" has been taken or not. The spatial location of the navigation mask should determine the trajectory path's direction; similarly, the orientation of the trajectory path should determine the location of the navigable regions. In the next section we describe the network architecture and the training process.

5 Methodology

We propose a novel multi-task network for navigation region prediction and future trajectory prediction tasks. Both tasks are treated as dense prediction tasks to make them interpretable for practical scenarios. We convert the dense pixel points to 3D world coordinates using inverse projective transformation during real-time inference. The architecture for our model is illustrated in Figure 4. In this section, we describe the feature extraction process and the architecture in detail.

We utilize CLIP (Radford et al., 2021) to extract both linguistic and visual features. For the linguistic expression $L = \{l_1, l_2, ...l_N\}$, where l_i is the i^{th} word of the expression, we tokenize the linguistic command using CLIP tokenizer and pass it through CLIP architecture to compute word-level feature representation $F^l = \{f_1^l, f_2^l, ..., f_N^l\}$ of shape $\mathbb{R}^{B\times N\times C_l}$. For visual frames $V = \{v_{t-k}, v_{t-k+1}, ...v_t\}$, the CLIP architecture encodes the video features as $F^v = \{f_{t-k}^v, f_{t-k+1}^v, ...f_t^v\}$. Finally, for the trajectory context P, we project the past trajectory on an image having same size as the input video frames v_t's and pass it through convolution and pooling layers to get feature map F^p with the same feature size as video frame features f_t^v's.

The input to our network are the video frames V, historical trajectory context P and the language command L. Specifically, for video V, we get visual features F^v of shape $\mathbb{R}^{C_v\times T\times HW}$, where H, W, T, and C represent the height, width, time, and channel dimensions, respectively. The trajectory context feature $F^p \in \mathbb{R}^{C_v\times 1\times HW}$ contains information about the past trajectory taken by the vehicle. Following feature extraction, we concatenate the trajectory context feature F^p with video features F^v along the temporal dimension resulting in joint feature $F^{vp} \in \mathbb{R}^{C_v\times(T+1)\times HW}$ capturing the video and trajectory related contextual information. Finally, we apply multi-head self-attention over the joint contextual feature F^{vp} and linguistic feature F^l in the following manner,

$$\begin{aligned} F &= F^{vp} \odot F^l \\ A &= \text{Mhead}(F, F, F) \\ M &= \text{Conv3D}(A * F) \end{aligned} \tag{1}$$

Here, $\odot$ represents the length-wise concatenation of the word-level linguistic features F^l and the joint feature F^{vp}, Mhead is the multi-head self-attention over the multi-modal features F and $*$ represents the matrix multiplication. Conv3D represents 3D convolution operation and is used to collapse the temporal dimension, M is the final multi-modal contextual feature with information from both visual and linguistic modalities.

Next, we describe the procedure for predicting the navigation and trajectory prediction masks. We want the future trajectory and the navigable region for the current time-step to be correlated with each other, i.e. the future trajectory should point in the direction of the predicted navigable region. Consequently, we utilize the multi-modal contextual

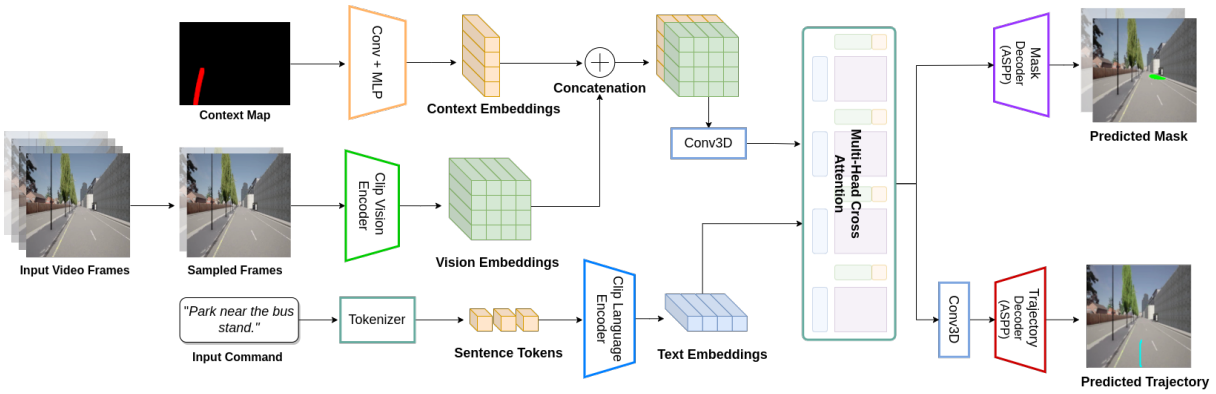

Figure 4: Overall pipeline of the proposed approach. Given the visual frames, already executed past trajectory (context map) and the textual command, the network predicts a segmentation map corresponding to the navigable region and a plausible future trajectory.

feature M to predict the segmentation masks corresponding to the navigation and trajectory prediction tasks. For each task, we have a separate segmentation head, where each segmentation head comprises of sequence of convolution layers with upsampling operation. For training the segmentation masks, we utilize combo loss (Taghanaki et al., 2019) which is a combination of binary cross-entropy loss and dice loss:

$$\begin{aligned} L_{bce} &= -(y_t \log(\hat{y_t}) + (1 - y_t) \log(1 - \hat{y_t})) \\ L_{dice} &= 2 * \frac{\hat{y_t} \cap y_t}{\Sigma \hat{y_t} + \Sigma y_t} \\ L_{combo} &= \lambda L_{bce} - (1 - \lambda) L_{dice} \end{aligned} \tag{2}$$

The proposed approach is end-end trainable and the predicted trajectory is highly correlated with the predicted navigation mask, as a result the predicted trajectory is interpretable in the sense that it suggests the future route to be taken by the autonomous vehicle.

6 Experiments

Implementation Details: We utilize CLIP backbone (Radford et al., 2021) for feature extraction. The frames are selected with a stride of 10 and are resized to 224×224 resolution. After feature extraction, we get per-frame visual features of spatial resolution $H = W = 7$ and channel dimension $C_v = 512$. For the historical contextual trajectory, we plot the trajectory from the starting location of episode to the current timestamp and resize it to 680×480 spatial resolution image, this is passed as input through convolution + MLP layers to obtain trajectory features with same resolution as per-frame visual features. For linguistic features, we use the CLIP tokenizer followed by the CLIP language encoder to compute the word-level features corresponding to the linguistic command. Maximum length of command is set to $N = 20$ and the channel dimension is $C_l = 512$. We use batch size of 32 and our network is trained using AdamW optimizer, the initial learning rate is set to $1e^{-4}$ and polynomial learning rate decay with power of 0.5 is used. For the combo loss, we set $\lambda = 0.3$.

Live-Navigation: In order to utilize the segmentation mask corresponding to the navigable region directly for navigation, we first need to sample a point from the predicted region. We take the largest connected component from the predicted mask and use its centroid as the target point for the local planner. As we move closer to the final navigable region, the distance between the current car location and the centroid target location consistently decreases. Simultaneously, the area of the predicted mask should increase as we move closer to the target region due to the perspective viewpoint of the front camera. Consequently, we use an area-based threshold to determine if the predicted navigation mask corresponds to the final navigable region or not. If the area of the predicted navigation mask is higher than the threshold for five consecutive times, we treat the predicted region as the final goal region corresponding to the linguistic command and stop the navigation.

Evaluation Metrics: Like previous approaches to VLN (Schumann and Riezler, 2022; Xiang et al., 2020; Chen et al., 2019), we use the gold standard ***Task Completion*** metric to measure the success ratio for the navigation task. In addition, we use ***Frechet Distance*** and ***normalized Dynamic Time Warping (nDTW)*** metrics to compare the pre-

Method	Task Completion	
	Val	Test
RNR-S	0.44	0.29
RNR-SC	0.52	0.32
CLIP-S	0.48	0.47
CLIP-SC	0.52	0.50
CLIP-M	0.56	0.55
CLIP-MC	**0.72**	**0.68**

Table 1: Results on the *Task Completion* metric. The superior performance of proposed approach CLIP-MC, showcases the effectiveness of historical context for the navigation task.

Method	Frechet Distance ↓		nDTW ↑	
	Val	Test	Val	Test
RNR-S	28.14	42.45	0.35	0.16
RNR-SC	21.64	44.65	0.45	0.33
CLIP-S	40.30	42.53	0.23	0.24
CLIP-SC	35.58	38.49	0.36	0.39
CLIP-M	32.92	53.10	0.39	0.26
CLIP-MC	**13.54**	**15.06**	**0.54**	**0.59**

Table 2: Experimental results on the *Frechet Distance* and *nDTW* metrics. ↓ indicates lower value is better and ↑ indicates that the higher value is better.

Method	Split	Task Completion				
		n=1	n=2	n=4	n=6	n=8
CLIP-MC	val	0.52	0.48	0.52	0.68	0.72
	test	0.50	0.53	0.56	0.62	0.68
CLIP-M	val	0.48	0.44	0.48	0.52	0.56
	test	0.47	0.47	0.50	0.53	0.55

Table 3: Ablation on the number of frames for multi-frame models for the Task Completion metric.

dicted navigation path during live inference with the ground truth navigation path.

6.1 Experimental Results

We compare our proposed approach CLIP-MC against the RNR-based approach proposed in (Rufus et al., 2021). We use their proposed approach with CLIP-based backbone, CLIP-S as the baseline for our experimental results. The original RNR approach is limited to using a static scene with linguistic commands for navigation, which fails in a dynamically changing environment where the scene can change drastically when we start the navigation. Additionally, we motivate the benefits of contextual trajectory and multiple frames by presenting two variant baselines, (1) multiple frames without a contextual trajectory CLIP-M and (2) single frame with contextual trajectory CLIP-SC. Table 1 presents the results on the gold standard *Task Completion* metric and Table 2 presents the results on *Frechet Distance* and *nDTW* metrics.

We observe that our proposed approach CLIP-MC outperforms all the other variants. Introducing historical contextual trajectory consistently helps improve performance as it increases by 4% and 16% in cases of single-frame approaches (CLIP-SC, CLIP-S) and multi-frame approaches (CLIP-MC, CLIP-M), respectively on the validation split. Furthermore, the multi-frame approach CLIP-M gives an improvement of 8% on both the validation and test splits, respectively, over the single-frame approach CLIP-S. These results indicate that a combination of multiple frames and contextual trajectory are required to effectively tackle the VLN task.

In Table 2, we present experimental results on the *Frechet Distance* and *nDTW* metrics. Our reformulated approach CLIP-MC outperforms all other variants by significant margins. However, we would like to stress that these metrics are not indicative of the performance on the actual navigation task, as one outlier can drastically affect the final score on these metrics. For example, if "a left turn" is taken instead of "a right turn," the predicted trajectory will diverge from the ground truth trajectory, and the score will be heavily penalized.

Effect of feature extraction backbone: Additionally, we compare our CLIP-based single frame approaches with the original non-CLIP RNR approach proposed in (Rufus et al., 2021), referred to as RNR-S and RNR-SC (single frame without and with context, respectively) in Table 1. Both RNR-S and RNR-SC are trained from scratch on the proposed CARLA-NAV dataset. The results showcase the advantage of superior multi-modal features captured by the CLIP-based approaches over non-CLIP approaches, as the performance consistently increases on the challenging test split in case of both with context (RNR-SC, CLIP-SC) and without context (RNR-S, CLIP-S).

Effect of Number of Frames: In Table 3, we study the impact of the number of video frames on the multi-frame models for the Task Completion metric. As the number of video frames increases, the visual modality's contextual information also increases. We hypothesize that the network should utilize this additional contextual information and employ it effectively for the VLN task. The results in Table 3 indeed corroborate our hypothesis, as we observe consistent performance gains as the number of video frames increases. The networks with $n = 1$ frame give the same performance as the

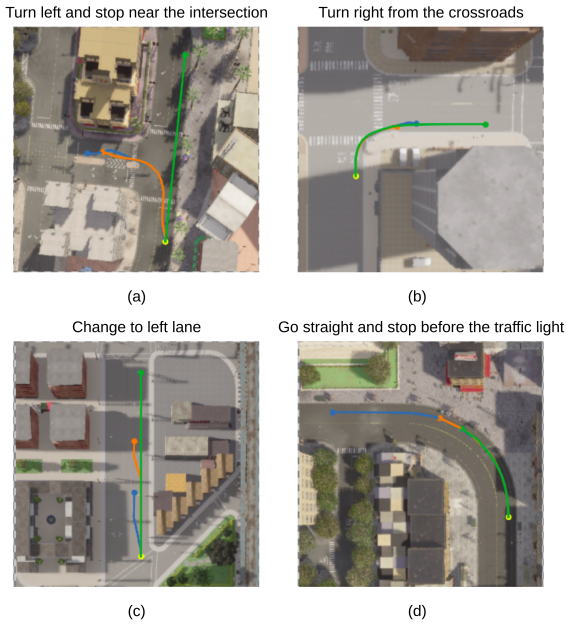

Figure 5: Qualitative navigation results in the CARLA-NAV dataset. Yellow represents the starting point for the navigation. Orange is used to depict the navigational path taken by CLIP-MC network, green denotes RNR-S network's navigational path and blue represents the ground-truth path.

corresponding single-frame variants. We obtain the best performance with $n = 8$ frames with CLIP-MC on both validation and test splits.

6.2 Qualitative Results

In Figure 5, we qualitatively compare the proposed approach CLIP-MC with the RNR approach (RNR-S) proposed in (Rufus et al., 2021). We juxtapose the entire navigation path taken by each approach during live inference for a given linguistic command and overlay it on the aerial map of the CARLA environment. We showcase successful navigation scenarios of CLIP-MC in (a) , (b) and (c). With additional contextual information from multiple frames and historical trajectory, CLIP-MC can successfully perform "turning" and "stopping" based navigational manoeuvres. While the RNR approach, without any contextual information and trained on static images, fails. For the command "change to the left lane", RNR-S fails to change the lane and continues in a straight line. While CLIP-MC manages to change the lane with a slight delay. For the example in the bottom-right corner, the road is curved in left direction and both the CLIP-MC and RNR-S stop much before the traffic light, as they mistake the curve with an intersection.

7 Conclusion

This paper proposes a language-guided navigation approach in dynamically changing outdoor environments. We reformulate the RNR approach, designed for static scenes to make it amenable for dynamic scenes. Our approach explicitly utilizes visual grounding directly for the navigation task. Along the same lines, we propose a novel meta-dataset CARLA-NAV, containing realistic scenarios of language-based navigation in dynamic outdoor environments. Additionally, we propose a novel multi-task grounding network for the tasks of navigable region and future trajectory prediction. The predicted navigable regions are explicitly used for navigating the vehicle in the dynamic environment. The predicted future trajectories bring interpretability to our approach and correlate with the predicted navigable region, i.e., they indicate the vehicle's navigational route. Furthermore, the proposed approach allows us to perform live navigation in a dynamic CARLA environment. Finally, quantitative and qualitative results validate our approach's effectiveness and practicality.

8 Limitations & Future Work

A major limitation of our approach that limits the practicality of our approach in real-world scenarios is the synthetic nature of our dataset. Future work should explore domain adaptation techniques like (Kundu et al., 2021; Kang et al., 2020) to ensure adaptability to real-world scenes. Stopping criteria is another aspect that future work can focus on. In this work, we utilize a rule-based stopping criterion; however, learning the stopping criteria like (Xiang et al., 2020) is more feasible for real-world scalability. Finally, we employ the predicted future trajectory to bring interpretability to our approach; future work should incorporate the predicted trajectory directly for end-to-end navigation.

References

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan,

Yu Pan, Giancarlo Baldan, and Oscar Beijbom. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*.

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 2019. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12538–12547.

Kashyap Chitta, Aditya Prakash, and Andreas Geiger. 2021. Neat: Neural attention fields for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15793–15803.

Federico Landi Lorenzo Baraldi Marcella Cornia and Massimiliano Corsini Rita Cucchiara. 2019. Perceive, transform, and act: Multi-modal attention networks for vision-and-language navigation.

Abhishek Das, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. 2018. Neural modular control for embodied question answering. In *Conference on Robot Learning*, pages 53–62. PMLR.

Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan. 2018. Visual grounding via accumulated attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7746–7755.

Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Luc Van Gool, and Marie Francine Moens. 2019. Talk2car: Taking control of your self-driving car. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2088–2098.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR.

Justin Fu, Anoop Korattikara, Sergey Levine, and Sergio Guadarrama. 2019. From language to goals: Inverse reinforcement learning for vision-based instruction following. *arXiv preprint arXiv:1902.07742*.

Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. 2019. Are you looking? grounding to multiple modalities in vision-and-language navigation. *arXiv preprint arXiv:1906.00347*.

Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. 2016a. Segmentation from natural language expressions. In *European Conference on Computer Vision*, pages 108–124. Springer.

Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. 2016b. Natural language object retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4555–4564.

Shaofei Huang, Tianrui Hui, Si Liu, Guanbin Li, Yunchao Wei, Jizhong Han, Luoqi Liu, and Bo Li. 2020. Referring image segmentation via cross-modal progressive comprehension. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10488–10497.

Muhammad Zubair Irshad, Chih-Yao Ma, and Zsolt Kira. 2021. Hierarchical cross-modal agent for robotics vision-and-language navigation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13238–13246. IEEE.

Kanishk Jain and Vineet Gandhi. 2022. Comprehensive multi-modal interactions for referring image segmentation. *Findings of the Association for Computational Linguistics: ACL*.

Guoliang Kang, Yunchao Wei, Yi Yang, Yueting Zhuang, and Alexander Hauptmann. 2020. Pixel-level cycle association: A new perspective for domain adaptive semantic segmentation. *Advances in Neural Information Processing Systems*, 33:3569–3580.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. 2017. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. 2020. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pages 104–120. Springer.

Jogendra Nath Kundu, Akshay Kulkarni, Amit Singh, Varun Jampani, and R Venkatesh Babu. 2021. Generalize then adapt: Source-free domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7046–7056.

Piotr Mirowski, Matt Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, Andrew Zisserman, Raia Hadsell, et al. 2018. Learning to navigate in cities without a map. *Advances in Neural Information Processing Systems*, 31.

Khanh Nguyen and Hal Daumé III. 2019. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. *arXiv preprint arXiv:1909.01871*.

Khanh Nguyen, Debadeepta Dey, Chris Brockett, and Bill Dolan. 2019. Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12527–12537.

Bryan A Plummer, Paige Kordas, M Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. 2018. Conditional image-text embedding networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 249–264.

Heqian Qiu, Hongliang Li, Qingbo Wu, Fanman Meng, Hengcan Shi, Taijin Zhao, and King Ngi Ngan. 2020. Language-aware fine-grained object representation for referring expression comprehension. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4171–4180.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer.

Nivedita Rufus, Kanishk Jain, Unni Krishnan R Nair, Vineet Gandhi, and K Madhava Krishna. 2021. Grounding linguistic commands to navigable regions. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

Nivedita Rufus, Unni Krishnan R Nair, K Madhava Krishna, and Vineet Gandhi. 2020. Cosine meets softmax: A tough-to-beat baseline for visual grounding. In *European Conference on Computer Vision*, pages 39–50. Springer.

Raphael Schumann and Stefan Riezler. 2020. Generating landmark navigation instructions from maps as a graph-to-text problem. *ACL\IJCNLP*.

Raphael Schumann and Stefan Riezler. 2022. Analyzing generalization of vision and language navigation to unseen outdoor areas. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7519–7532, Dublin, Ireland. Association for Computational Linguistics.

Hengcan Shi, Hongliang Li, Fanman Meng, and Qingbo Wu. 2018. Key-word-aware network for referring expression image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 38–54.

Amar Shrestha, Krittaphat Pugdeethosapol, Haowen Fang, and Qinru Qiu. 2020. High-level plan for behavioral robot navigation with natural language directions and r-net.

NN Sriram, Tirth Maniar, Jayaganesh Kalyanasundaram, Vineet Gandhi, Brojeshwar Bhowmick, and K Madhava Krishna. 2019. Talk to the vehicle: Language conditioned autonomous navigation of self driving cars. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5284–5290. IEEE.

Saeid Asgari Taghanaki, Yefeng Zheng, S Kevin Zhou, Bogdan Georgescu, Puneet Sharma, Daguang Xu, Dorin Comaniciu, and Ghassan Hamarneh. 2019. Combo loss: Handling input and output imbalance in multi-organ segmentation. *Computerized Medical Imaging and Graphics*, 75:24–33.

Yi Wu, Yuxin Wu, Georgia Gkioxari, and Yuandong Tian. 2018. Building generalizable agents with a realistic and rich 3d environment. *arXiv preprint arXiv:1801.02209*.

Jiannan Xiang, Xin Wang, and William Yang Wang. 2020. Learning to stop: A simple yet effective approach to urban vision-language navigation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 699–707, Online. Association for Computational Linguistics.

Sibei Yang, Guanbin Li, and Yizhou Yu. 2019. Dynamic graph attention for referring expression comprehension. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4644–4653.

Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. 2019. Cross-modal self-attention network for referring image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10502–10511.

Xiaoxue Zang, Ashwini Pokle, Marynel Vázquez, Kevin Chen, Juan Carlos Niebles, Alvaro Soto, and Silvio Savarese. 2018. Translating navigation instructions in natural language to a high-level plan for behavioral robot navigation.

Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. 2018. Grounding referring expressions in images by variational context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4158–4166.

Wanrong Zhu, Xin Wang, Tsu-Jui Fu, An Yan, Pradyumna Narayana, Kazoo Sone, Sugato Basu, and William Yang Wang. 2021. Multimodal text style transfer for outdoor vision-and-language navigation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1207–1221, Online. Association for Computational Linguistics.