VLN-VIDEO: Utilizing Driving Videos for Outdoor Vision-and-Language Navigation

Jialu Li[1,2*], Aishwarya Padmakumar[2], Gaurav Sukhatme[2], Mohit Bansal[1]

[1]University of North Carolina, Chapel Hill
[2]Amazon Alexa AI
{jialuli, mbansal}@cs.unc.edu, {padmakua, sukhatme}@amazon.com

Abstract

Outdoor Vision-and-Language Navigation (VLN) requires an agent to navigate through realistic 3D outdoor environments based on natural language instructions. The performance of existing VLN methods is limited by insufficient diversity in navigation environments and limited training data. To address these issues, we propose VLN-VIDEO, which utilizes the diverse outdoor environments present in driving videos in multiple cities in the U.S. augmented with automatically generated navigation instructions and actions to improve outdoor VLN performance. VLN-VIDEO combines the best of intuitive classical approaches and modern deep learning techniques, using template infilling to generate grounded navigation instructions, combined with an image rotation similarity based navigation action predictor to obtain VLN style data from driving videos for pretraining deep learning VLN models. We pre-train the model on the Touchdown dataset and our video-augmented dataset created from driving videos with three proxy tasks: Masked Language Modeling, Instruction and Trajectory Matching, and Next Action Prediction, so as to learn temporally-aware and visually-aligned instruction representations. The learned instruction representation is adapted to the state-of-the-art navigator when fine-tuning on the Touchdown dataset. Empirical results demonstrate that VLN-VIDEO significantly outperforms previous state-of-the-art models by 2.1% in task completion rate, achieving a new state-of-the-art on the Touchdown dataset.

Introduction

Vision-and-Language Navigation (VLN) requires an agent to navigate through 3D environments based on natural language instructions (Mirowski et al. 2018; Anderson et al. 2018; Misra et al. 2018; Thomason et al. 2019; Nguyen and Daumé III 2019; Chen et al. 2019; Shridhar et al. 2020; Qi et al. 2020; Ku et al. 2020; Mehta et al. 2020). A critical bottleneck in improving models for VLN tasks is the limited availability of training data. Training data for VLN are usually collected with human annotation, where multiple annotators write instructions for a sampled trajectory in the environment. Additional annotators read these instructions and navigate accordingly to evaluate their quality and clarity. This annotation process is expensive and time consuming, which makes it hard to collect a large dataset for VLN.

Many methods have been proposed to solve the data scarcity problem in VLN. For indoor VLN, several works leverage pre-trained multi-modal representations from large datasets such as BookCorpus (Zhu et al. 2015) for natural language and Conceptual Captions (Sharma et al. 2018) for images, and further enhance the agent with specific domain knowledge by pre-training on an in-domain navigation dataset (Hao et al. 2020; Majumdar et al. 2020; Chen et al. 2021; Qiao et al. 2022), sometimes enhanced with additional out of domain images (Guhur et al. 2021; Li, Tan, and Bansal 2022; Liu et al. 2021; Chen et al. 2022b). Other works perform multimodal augmentation by training a speaker model to generate synthetic navigation instructions for unlabeled trajectories (Hao et al. 2020; He et al. 2021; Dou and Peng 2022; Fried et al. 2018; Tan, Yu, and Bansal 2019). Such augmentation methods can result in promising performance gains but prior results have focused primarily on indoor environments, in contrast to our focus on outdoor VLN. Additionally, most of these approaches do not introduce new training environments to improve generalization to unseen environments. Some add out of domain images (Guhur et al. 2021) but they lack the temporal information that could help an agent learn causality between actions and consequent observations. To address these problems, we propose to utilize large video datasets that have diverse new environments and both spatial and temporal information for pre-training to further enhance the ability of trained agents to reason. Additionally, if we can effectively utilize video for pretraining VLN agents, there exist a number of other video datasets (Chen et al. 2018; Grauman et al. 2022; Pirsiavash and Ramanan 2012; Fabian Caba Heilbron and Niebles 2015; Abu-El-Haija et al. 2016) that could be potentially useful for indoor or outdoor VLN tasks.

In this work, we propose VLN-VIDEO, a new data augmentation method that enhances outdoor VLN by learning from driving videos (Figure 1). We process driving videos from the BDD100k dataset (Chen et al. 2018) to obtain pre-training data for the Touchdown dataset (Chen et al. 2019). To utilize driving videos for pre-training, we need to generate language navigation instructions for each video and predict navigation actions - move forward, turn left or turn right - between each pair of consecutive frames. Learning the in-

*Work done during an internship at Amazon.

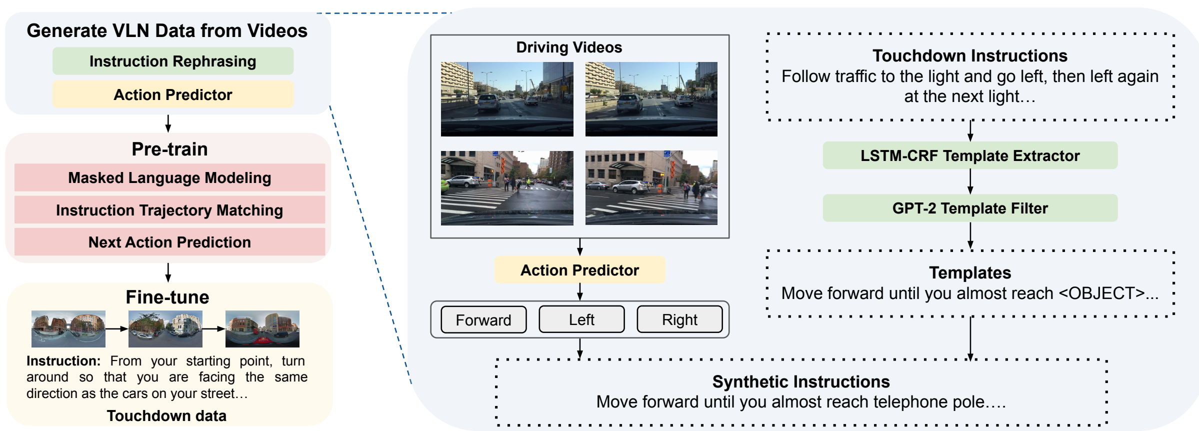


Figure 1: Overview of our proposed method VLN-VIDEO: We annotate driving videos with synthetic navigation instructions by extracting instruction templates from the Touchdown dataset and filling them with actions predicted using our image rotation similarity based navigation action predictor and objects detected using a pre-trained object detector. We pre-train VLN models on both the processed video data and Touchdown data to learn better domain knowledge with three proxy tasks. Lastly, we transfer the learned language representation to VLN via fine-tuning.

struction generator is challenging for three reasons. First, the entities referenced in outdoor navigation follows a long-tail distribution, where an object such as "intersection" appears in 29% of the sentences in the instructions. This when combined with the small size of the training set and long average instruction length of 89.6 words causes a speaker model to overfit to frequently occurring objects and generate repetitive instructions. Further, the entities mentioned in the instruction only take a small portion of the observation and are unlikely to be centered in the image, which make it difficult to learn the alignment between observations and instructions. Finally, the large domain gap between Touchdown images in Manhattan and BDD100K videos in a larger number of cities makes transferring a speaker model challenging.

To effectively generate synthetic instructions for driving videos, we propose a template infilling method that explicitly incorporates objects detected in environments to have richer entity references and less repetition. Due to the above challenges of training speaker models in our setting, we propose a template based method for generating navigation instructions - masking out noun phrases and navigation directions from the original instructions, and filling them with select objects detected in the observation by a pre-trained object detector and navigation actions predicted using an intuitive image rotation similarity based predictor.

With the augumented data from videos, we pre-train the agent with three proxy tasks: Masked Language Modeling, Instruction and Trajectory Matching, and Next Action Prediction to learn temporally-aware and visually-aligned instruction representations. We adapt the learned instruction representations to the state-of-the-art navigation agent and fine-tune it on the Touchdown dataset.

We show that VLN-VIDEO significantly improves over non-pre-training baselines by 2.1% in task completion rate (TC) on the Touchdown (Chen et al. 2019) test set, achieving the new state-of-the-art for Touchdown. We improve over pre-training with only in-domain data by 2.9% TC on the validation set and demonstrate that pre-training on the synthetic data generated with our template infilling method on StreetLearn dataset achieves better performance than the style-transferred Google Maps instructions (Zhu et al. 2021). We also show that our template infilling method and rotation similarity based action predictor work better than learning based methods, improving the performance by 7.4% and 3.2% TC respectively. Finally, we qualitatively show that our generated instructions align with the trajectory better.

Related Work

Vision-and-Language Navigation. Vision-and-Language Navigation is a task that requires an agent to navigate through a 3D environment based on natural language instructions and egocentric visual observations. Many datasets have been proposed for this task (Mirowski et al. 2018; Anderson et al. 2018; Misra et al. 2018; Chen et al. 2019; Shridhar et al. 2020; Qi et al. 2020; Ku et al. 2020; Mehta et al. 2020). While substantial progress has been made for indoor VLN (Chen et al. 2021, 2022b; Kim, Li, and Bansal 2021; Wang et al. 2019; Lin et al. 2022; Chen et al. 2022a; Fu et al. 2020), outdoor VLN is still an under-explored area (Zhu et al. 2021; Schumann and Riezler 2022; Xiang, Wang, and Wang 2020; Armitage, Impett, and Sennrich 2022). Prior work includes an LSTM navigation agent with cross modal attention to ground instructions in navigation history (Schumann and Riezler 2022), using trajectory traces as external knowledge to aid agents' navigation (Armitage, Impett, and Sennrich 2022), and using a transformer based architecture combined with a data augmentation method that transfers the style of the instructions in the StreetLearn dataset (Mirowski et al. 2019) subsequently used for directly pre-training the agent with the downstream navigation task (Zhu et al. 2021). We explore a new data source - driving videos - for data augmentation to enrich the navigation environments, which requires novel techniques for instruction generation and navigation action prediction.

Data Augmentation in Vision-and-Language Navigation. Data scarcity is a core problem in VLN. Many works attempt to learn deep learning based speaker models to generate synthetic instructions for unannotated paths in the Matterport 3D (Chang et al. 2017) environments to improve performance on indoor VLN datasets set in these environments (Tan, Yu, and Bansal 2019; Hao et al. 2020; Fried et al. 2018; Dou and Peng 2022; He et al. 2021; Zhao et al. 2021). Recent research aims to augment indoor navigation environments by mixing (Liu et al. 2021) or editing (Li, Tan, and Bansal 2022) existing environments and utilizing external room images (Guhur et al. 2021; Chen et al. 2022b). For outdoor VLN, prior work transfers the style of the instructions in the StreetLearn dataset (Mirowski et al. 2019) to match the instruction style on Touchdown dataset (Zhu et al. 2021). This also provides an augmented navigation environment as StreetLearn covers a larger geographical area compared to Touchdown. However, their approach is dependent on the existence of ground truth navigation instructions in the dataset used for augmentation. VLN-VIDEO proposes an instruction generation method that does not rely on ground truth navigation instructions, and is the first to explore the utilization of videos as a source for augmented data in outdoor VLN.

Utilizing Videos in Pre-training. Videos have been utilized during pre-training for downstream tasks such as text-to-video retrieval (Xu et al. 2016; Krishna et al. 2017; Rohrbach et al. 2015), video question answering (Xu et al. 2017; Jang et al. 2017; Lei et al. 2018), and video classification (Goyal et al. 2017; Kay et al. 2017). Recently, videos have been used in pre-training for non-video tasks such as playing a video game - Minecraft (Baker et al. 2022), and pre-training the policy representation for a driving agent (Zhang, Peng, and Zhou 2022). However, these works focus on uni-modal tasks. Our paper is the first to explore utilizing videos during pre-training for multi-modal instruction-guided navigation tasks.

Generating VLN Data from Driving Videos

In this section, we explain the challenges in generating synthetic instructions for videos data. Next, we describe our approach that automatically generates large-scale VLN data from caption-less driving videos. Specifically, we use driving videos from the training set in BDD100k (Chen et al. 2018). Our template infilling based generation approach contains three parts: instruction template extractor, image rotation similarity based navigation action predictor, and object detector. We first extract instruction templates from the training data in the Touchdown dataset. We then predict navigation actions between consecutive frames, detect objects in the current observation, and use these to fill templates to obtain navigation instructions.

Challenges of Instruction Generation from Videos

Although many works have explored generating synthetic instructions from navigation trajectories for indoor VLN using deep learning models, such models fail the outdoor Touchdown dataset for three reasons:

Difficulty in Aligning Objects in Observations with Entities in Instructions. The objects referenced by entities in outdoor VLN instructions usually take up a small area in the full panoramic outdoor image observation. This makes it more difficult for a deep learning model to learn the cross-modality grounding between the instruction and the observation, which is crucial for learning a good instruction generator. VLN models trained on these datasets are also found to rely less on detected objects - it is shown that even when 100% of the objects are masked out in the instruction, an LSTM VLN model's performance only drops by less than 3% in task completion (Schumann and Riezler 2022). Further, although the navigation instructions in Touchdown (Chen et al. 2019) have rich references to entities in each instruction, these follow a long-tail distribution. This imbalance hinders the ability of a speaker model to learn the alignment between observations and instructions based on object information, and causes speaker models to overfit to objects that appear frequently.

Large Environment Gap between Navigation Environment and Driving Videos. There usually exists little to no gap between the training and inference environments for a speaker model for indoor VLN. Most works use the speaker model to generate synthetic instructions on un-annotated paths in the same environments (Tan, Yu, and Bansal 2019; Hao et al. 2020; Dou and Peng 2022; Fried et al. 2018; He et al. 2021). Given the large gap between the panorama observations in Touchdown environments and egocentric views in the driving videos, the performance of directly adapting a speaker model trained on Touchdown environments to driving videos is very low. Although the large domain gap poses challenges in generating synthetic instructions for videos, it also has the potential to enrich the navigation environments with more diverse visual observations from driving videos.

Long Instructions and Trajectories in Outdoor Navigation. The instructions in Touchdown are almost three times longer than the instructions in the indoor VLN dataset Room-to-Room (Anderson et al. 2018), with an average length of 89.6 words. Further, the trajectories in Touchdown are nearly six times longer than the trajectories in Room-to-Room (35.2 vs. 6 steps on average). However, Touchdown only contains 6,526 training examples in contrast to 14,052 for Room-to-Room. The long instruction and trajectory length and limited available training data makes it hard to learn a speaker model that generates faithful navigation instructions.

We show in analysis that a speaker trained on Touchdown fails to generate high quality instructions for BDD100K videos, while our template infilling approach can generate instructions with rich entity mentions and little repetition.

Instruction Template Extraction

We extract instruction templates from the training data of the Touchdown dataset. The template extraction method consists of two steps. First, we detect noun phrases and direction words in the sentence using an LSTM-CRF model with a pre-trained sequence labeling embedding (Akbik,

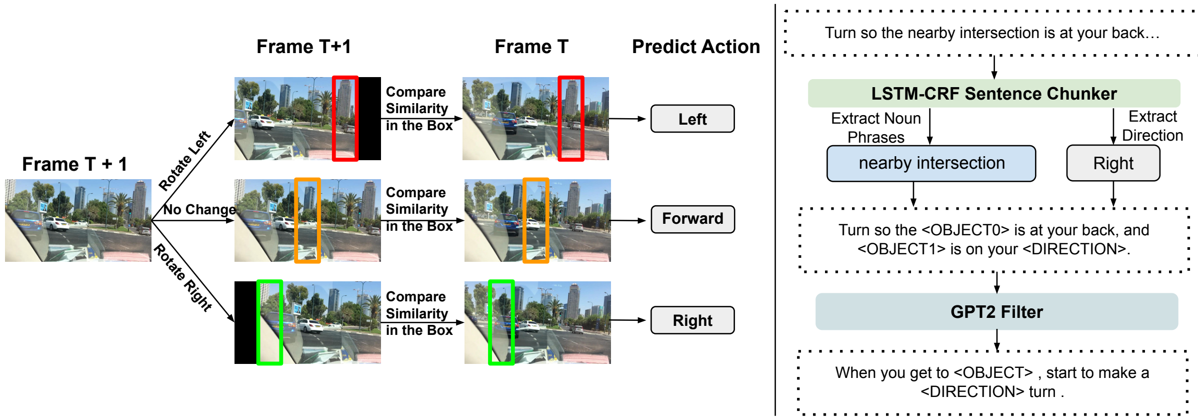


Figure 2: Overview of our proposed method to detect the turn between consecutive frames in driving videos, and generate synthetic instructions with template infilling method.

Blythe, and Vollgraf 2018) to chunk the instructions. The sequence labeling embedding is a contextualized character-level word embedding, which is learned by optimizing the next character prediction task. We mask noun phrases with <OBJECT> tokens and filter out templates that have multiple <OBJECT> tokens to reduce the probability of nonsensical sentences resulting from strange combinations of objects (for example we would want to avoid generating a sentence such as: <Street sign> should be <intersection> to your right). We then categorize templates into "turn", "forward" and "stop" templates by searching for the keywords "right", "left", "forward" and "stop" since humans tend to write instructions differently for each of these types. We filter out templates that contain multiple direction words so that each template matches with one specific action. We then use GPT-2 (Radford et al. 2019) to filter out templates with low generation probability. Specifically, for every template, we fill in the most frequently appearing objects (signboard, traffic light, awning, telephone pole) and the direction words ("left", "right", "forward", "stop"), calculate its generation loss with GPT-2, and filter out templates with high generation loss. We filter out half of the templates with GPT-2. In total, we extract 3,004 templates for turn actions, 269 templates for forward actions, and 92 templates for stop actions.

Image Rotation Similarity Based Navigation Action Prediction from Driving Videos

Touchdown (Chen et al. 2019) contains a graph of the navigation environments, where each node contains a panorama observation and its heading, and edges indicate connectivity between panoramas. Navigation actions can be derived from heading changes between two consecutive panoramas. However, video datasets do not have explicit heading annotation for each frame to derive navigation actions between consecutive frames. Thus, we need to generate actions for driving videos as pseudo labels for pre-training tasks such as next action prediction. We additionally utilize the predicted actions to generate more accurate instructions.

One standard way to predict an action on the driving video dataset is to learn an action predictor on the Touchdown dataset. However, the action distribution in Touchdown is imbalanced with 91% of the actions as "forward" actions. We experiment with three models for navigation action prediction: a multi-layer perceptron which takes in observations at two consecutive steps, an LSTM based decoder and a transformer based decoder that takes in the full navigation history. We also use under-sampling and weighted cross-entropy loss to mitigate the data imbalance. Though the best model achieves 82% macro average accuracy on Touchdown validation data, when adapted to driving videos, they predict the "forward" action for almost all pairs of frames, likely due to the domain gap between Touchdown environments and driving videos in BDD100k.

Thus, we propose a more intuitive and effective *image rotation similarity* based method to predict navigation actions. We hypothesize that two consecutive frames in a video contain mostly the same information, which makes it hard for deep learning models to learn the small differences between consecutive frames to predict the action. Thus, we instead rotate the frame to mimic the car turning action, and compare the similarity between the rotated image and the target frame to determine the turning direction. As shown in Figure 2, given two consecutive frames sampled from a video, we rotate the frame at time $t + 1$ in both left and right directions. We hypothesize that if the frame rotated left has a higher similarity with frame t than the frame rotated right or remain unchanged, it indicates that the car is turning left. Specifically, given a frame with size $H \times W$, and a window with size D, we calculate the similarity inside the window between rotated frame and frame t. The window will pass through the image from left to right, excluding the rotated part with width R, which gives $(W - R)/D$ similarity scores. We compare the similarity scores for frames rotated left, rotated right and unchanged, using the similarity score in rightmost window to predict a left turn, middle window to predict moving forward, and leftmost window to predict right turn. Mean squared error over pixel values is used to represent the similarity between two frames, and lower mean squared error indicates higher similarity.

Object Detection

We utilize a Mask-RCNN model (He et al. 2017) from the Detectron2 (Wu et al. 2019) package pre-trained on the LVIS dataset (Gupta, Dollar, and Girshick 2019) to detect objects in video frames. The LVIS dataset contains a moderate number of object classes (around 1200 classes), which detects more diverse objects such as "trash can" compared with MSCOCO (91 classes), and is not too detailed as YOLO9000 which divides car into different classes such as "hotrod". We manually filter out some object classes that appear frequently (i.e., "bus", "car(automobile)"), and items that are not helpful for navigation (i.e., "license plate", "wheel", "rearview mirror", "taillight")

Instruction Generation

Each video in the BDD100K dataset is a 40 seconds driving video. We sample frames with an interval of 1 second. To increase the variance of the trajectory length, we sample a trajectory length between 25 to 40. We predict actions between consecutive frames using our rotation-similarity based action predictor, and further merge predicted successive forward actions together with a maximum of 6 forwards. Similarly, we merge successive left or right actions together since one turn action might happen across multiple frames. After merging, we generate one sentence for each action and concatenate them as the final instruction.

VLN Model and Training Procedures

Stage 1: Pre-training

We pre-train the VLN-Transformer model (Zhu et al. 2021) on the Touchdown dataset (Chen et al. 2019), Manh50 dataset (Zhu et al. 2021) and BDD100K dataset (Chen et al. 2018). Specifically, we use our VLN-VIDEO to generate synthetic instructions for both Manh50 and BDD100K. On Manh50, we use the heading change between nodes on the graph to fill in the actions, while on BDD100K, we use our action predictor to predict the action sequence.

VLN-Transformer (Zhu et al. 2021) is a multi-modal transformer model. In this model, words in the instruction are encoded using BERT (Devlin et al. 2018) and averaged to obtain a sentence embedding for each sentence in the instruction. View representations of observations are obtained using the fourth-to-last layer features from ResNet (He et al. 2016) pre-trained on ImageNet (Deng et al. 2009), flattened using another convolutional layer. Sentence and view embeddings are concatenated, passed through cross modal transformer layers and finally concatenated and passed through a fully connected layer to obtain the next action prediction.

We use three proxy tasks to pre-train the model: Masked Language Modeling (MLM), Instruction and Trajectory Matching (ITM), and Next Action Prediction (NAP). In Masked Language Modeling, the agent needs to recover masked words given surrounding words and visual information from the full navigation trajectory. In Instruction and Trajectory Matching, we create four negative paths for each positive trajectory and instruction pair, and the agent needs to identify the positive pair. Two of the negative pairs are randomly sampled from the batch, and another two are created by shuffling the sequence of the viewpoints in the trajectory. In this case, the model learns to align both the semantics between instruction and visual observations, and also be aware of the order information of the trajectory. In Next Action Prediction, given full navigation instruction and the history observation, the agent predicts the next step action.

Stage 2: Fine-tuning.

In the second stage, we fine-tune the state-of-the-art ORAR model (Schumann and Riezler 2022) on the navigation task with imitation learning. The ORAR model is an LSTM based encoder-decoder agent. The instructions are encoded with a bi-directional LSTM over word embeddings learned from an embedding layer. The decoder is a two-layer LSTM, where the first layer encodes the trajectory representation and the second layer learns to predict action. The input $i_t = v_t||a_{t-1}||j_t||d_t$ to the first layer LSTM is a concatenation of encoded history visual observations v_t, action embeddings at previous step a_{t-1}, junction embedding indicates the number of outgoing edges at each step j_t, and heading embedding that indicates the heading of each step d_t. Soft dot attention is utilized to calculate weighted instruction representation c_l based on trajectory representations, and calculate weighted trajectory representation c_i based on weighted instructions. The policy making LSTM layer predicts the next action given c_l, c_i, e_t, and h_t^{first}, where e_t is the time embedding of the current step, and h_t^{first} is the output at step t from the first LSTM layer.

We extract the learned contextualized instruction representation from the pre-trained VLN-Transformer, and use it as the input word embedding to the LSTM based ORAR agent. As the original ORAR paper (Schumann and Riezler 2022) utilized a randomly initialized embedding layer to learn the word embedding, we additionally compare to using pre-trained BERT-base embeddings (Devlin et al. 2018) for fair comparison.

Experiments

Datasets and Evaluation Metrics

We evaluate our agent on the Touchdown dataset (Chen et al. 2019). Touchdown is set in Manhattan and contains 9,326 instruction-trajectory pairs, with 6,526 examples in the training set, 1,391 examples in the validation set, and 1,409 examples in the test set.

The Manh50 dataset (Zhu et al. 2021) we use during pre-training is extracted from the StreetLearn dataset (Mirowski et al. 2019) by sampling trajectories in the Manhattan area with length less than 50. The Manh50 dataset contains 31k training trajectories, and each training trajectory is paired with one instruction automatically generated by the Google Map API. Prior work transfers the style of these instructions to more natural instructions (Zhu et al. 2021).

The driving videos we utilized during pre-training come from the BDD100K dataset (Chen et al. 2018). We use a

Models	Validation Set			Test Set		
	TC↑	SPD↓	SED↑	TC↑	SPD↓	SED↑
RConcat	10.6	20.4	10.3	11.8	20.4	11.5
GA	12.0	18.7	11.6	11.9	19.0	11.5
ARC-L2S	19.5	17.1	19.0	16.7	18.8	16.3
VLN-Trans	15.0	20.3	14.7	16.2	20.8	15.7
ORAR	30.1	11.1	29.5	29.6	11.8	28.9
PM-VLN	33.0	23.6	29.5	**33.4**	23.8	29.7
ORAR-BERT	30.6	10.3	29.9	30	11.3	29
Ours	**34.5**	**9.6**	**33.5**	31.7	**11.2**	**31**

Table 1: Comparison of agent performance on Touchdown dataset on validation set and test set.

Models	Validation Set			Test Set		
	TC↑	SPD↓	SED↑	TC↑	SPD↓	SED↑
ORAR-BERT	30.6	10.3	29.9	30	11.3	29.0
+TD	31.6	10.6	30.9	31.4	11.2	30.6
+TD+M50-ori	32.2	10.0	31.4	30.4	10.8	29.5
+TD+M50-style	31.6	10.0	30.7	**32.7**	**10.5**	**32.0**
+TD+M50-our	34.3	10.7	**33.7**	31.2	11.1	30.4
Ours	**34.5**	**9.6**	33.5	31.7	11.2	31.0

Table 2: Comparison of agent performance when trained on different datasets on Touchdown validation set and test set.

subset of 6K videos in the BDD100K training data, where each driving video is captured in a city environment with clear weather during daytime.

We use three metrics to evaluate agents' performance on Touchdown: Task Completion (TC), Shortest Path Distance (SPD), and Success weighted by Edit Distance (SED). Details can be found in Appendix.

Results and Analysis

Comparison with SotA Agents

In this section, we compare our agent with previous state-of-the-art approaches. As shown in Table 1, our VLN-VIDEO agent pre-trained on Touchdown and processed BDD100K videos achieves the new state-of-the-art on validation set and test set. Note that PM-VLN agent (Armitage, Impett, and Sennrich 2022) has access to extra ground truth information - it uses the path trace image of the full navigation trajectory for trajectory planning, where the agent can foresee all future actions implicitly at the start of navigation. Though VLN-VIDEO is 1.7% lower than the PM-VLN agent in task completion on test set, our agent significantly improves the SPD score by 12.6%, and 1.3% in SED, demonstrating that our agent follows the instruction better while navigating correctly. ORAR-BERT is the agent that utilizes pre-trained BERT as the embedding input to the ORAR model. VLN-VIDEO improves over ORAR-BERT in task completion by 3.9% on validation set and 1.7% on test set, demonstrating that our agent is able to learn better instruction representation through pre-training on both the Touchdown dataset and processed driving videos.

Pre-training with Different Datasets

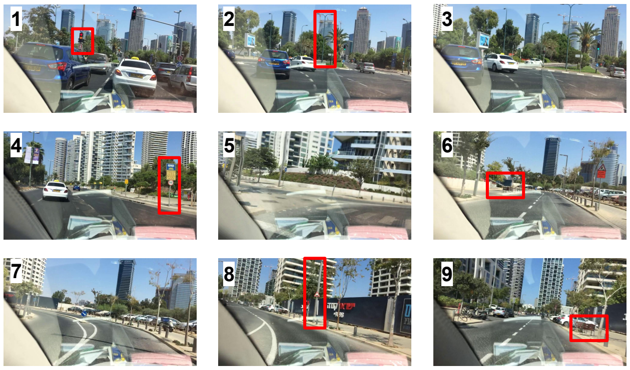
We evaluate the agents' performance when pre-trained on different datasets. As shown in Table 2, pre-training on Touchdown only ("+TD") achieves 1.0% improvement in TC on the validation set, demonstrating that the agent benefits from the in-domain knowledge learned during pre-training. Pre-training on Touchdown and Manh50 ("+TD+M50-ori") improves the baseline model by 1.6% in TC and 0.3% in SPD on the validation set, suggesting that the agent benefits from learning from more in-domain instruction-trajectory pairs. Then we evaluate the influence of instruction quality on performance by comparing a model pre-trained on automatically generated instructions in Manh50 dataset ("+TD+M50-ori"), transferred style instructions provided by (Zhu et al. 2021) ("+TD+M50-style"), and instructions generated by our method ("+TD+M50-our"). Table 2 shows that pre-training with instructions generated by our method achieves the best performance, outperforming the others by more than 2% in TC on the validation set. Training with transferred style instructions achieves the best performance on test set, which we hypothesize is due to the instruction style transfer model being trained on data from test splits in Touchdown dataset. Finally, pre-training on both Touchdown and driving videos ("Ours") improves the baseline by 3.9% in TC on the validation set, indicating that the agent benefits from learning from the diverse visual observations and temporal information in the driving videos.

Models	Validation Set			Test Set		
	TC↑	SPD↓	SED↑	TC↑	SPD↓	SED↑
Baseline	30.6	10.3	29.9	30.0	11.3	29.0
Ours	**34.5**	**9.6**	**33.5**	**31.7**	11.2	**31.0**
-speaker	27.1	11.7	26.4	28.2	11.7	27.6
-GPT2	31.6	10.3	30.8	29.8	**10.3**	29.1
-action	31.3	10.0	30.7	30.7	11.0	30.0

Table 3: Ablations of our proposed instruction rephrasing method and action prediction method on Touchdown dataset. The agents are pre-trained on Touchdown dataset and driving videos.

Speaker and Turn Predictor Effectiveness

We compare our template infilling method and image rotation similarity based navigation action prediction method with popular deep learning based methods. Specifically, we train a LSTM based speaker model (Tan, Yu, and Bansal 2019) on Touchdown and use it to generate synthetic instructions for driving videos. When combined with actions predicted using our rotation similarity based action predictor, as shown in Table 3, using this learned speaker ("-speaker") decreases the performance by 3.5% in TC on the validation set, demonstrating that the learned speaker fails to generate good quality instructions for the driving videos. Then, we remove the GPT-2 template filtering step during data generation ("-GPT2") and observe a decrease in performance by 2.9% in TC on validation set, demonstrating that it's crucial to filter out templates with low quality. We further experiment with utilizing a deep learning based turn predictor trained on Touchdown to predict actions between frames in the videos. The turn predictor is a multi-layer perceptron over visual features encoded with pre-trained ResNet and



Template Infilling (Ours): Orient yourself with street_sign and proceed forward. Walk forward down street_sign. Go to traffic_light and turn left. Move forward to streetlight. Move forward. street_sign will be on the right corner. Walk forward past street_sign. Turn right at the intersection. Move forward. Take bench to the left. Move forward. Turn left at the intersection. Head forward and go straight through pole. turn around and align yourself so that pole are on your left. Walk forward into signboard. Just prior to bench, stop.

Speaker: Turn so the red brick building is on your left. go straight through the first intersection. turn left at the next intersection. you will see a red awning on your left. go straight through the first intersection. you will see a red awning on your right. go straight through the next intersection. you will see a red awning on your right.

Figure 3: Qualitative Analysis of our proposed VLN-VIDEO in generating synthetic instructions for videos in BDD100K dataset. The object entities mentioned in the synthetic instructions generated with our VLN-VIDEO are in red, and their corresponding location is bounded in the red box in the frames. The turn actions are in blue.

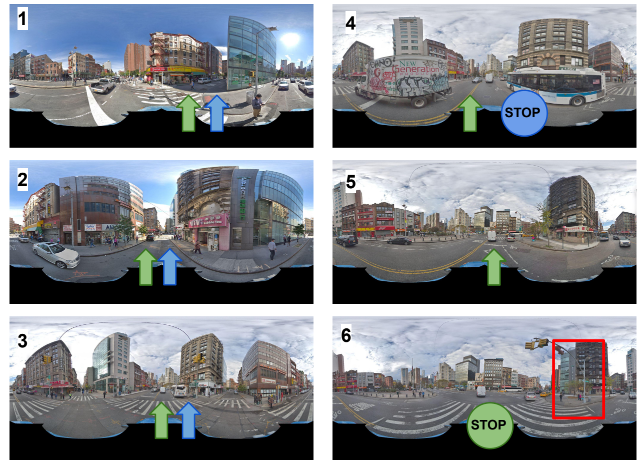
additional learned CNN layers. When combined with our template infilling method to generate the instructions, the learned turn predictor improves over the baseline by 0.7% in TC on the validation set, but is 3.2% lower than using the actions predicted with our rotation-similarity based approach. We also include a human evaluation to show the effectiveness of our action predictor in Appendix.

Qualitative Analysis of Synthetic Instructions

We show a qualitative example of our generated synthetic instructions on the BDD100K dataset. As shown in Figure 3, given the driving video shown here, VLN-VIDEO is able to generate long instructions with rich entity mentions. The synthetic instructions mention four different objects (street sign, traffic light, bench, pole) which appear in the videos. Furthermore, our rotation based action prediction method successfully predicts all the three turns made in the instruction. In comparison, the instruction generated by the speaker method is repetitive and mentions objects like "awning" that do not appear in the views.

Case Study

We show a qualitative example to compare the navigation trajectory predicted by VLN-VIDEO and the baseline. As shown in Figure 4, VLN-VIDEO successfully identifies the



Align yourself so that the english and chinese yellow and red sign are in front to your right. you'll want to make a right to back up in to this street. follow it down and make another right. a few steps down to your right is a silver/metal covered trash can next to the traffic pole.

Figure 4: Qualitative Analysis of our proposed VLN-VIDEO in learning richer visual objects to help decision making during navigation. Symbols in green are the actions made by our method, and symbols in blue are the actions made by the baseline method.

"trash can" and "traffic pole" in the view and stops correctly, whereas the baseline fails and stops earlier. This suggests that diverse visual data seen in the videos helps the agent learn the semantics of objects better. More examples can be found in Appendix.

Conclusion

In this paper, we propose VLN-VIDEO - a method to process driving videos to obtain augmented data for outdoor Vision-and-Language Navigation by utilizing intuitive classical techniques for navigation action prediction and instruction generation, which when combined with recent deep learning models for VLN obtains a new state-of-the-art performance on the Touchdown dataset. We propose an image rotation similarity based method to predict navigation actions between consecutive video frames and a template infilling based method for instruction generation that does not require any human annotation. Our method can more generally be applied to preprocess any video dataset for pretraining any VLN model. Our experiments on the Touchdown dataset show that pre-training the agent with our generated synthetic navigation data helps the agent learn contextualized language representations that ground better to the navigation environments, which greatly improves the downstream navigation performance. VLN-VIDEO achieves the new state-of-the-art performance on Touchdown dataset, and provides a good starting point for future work that aims to exploit the rich visual and temporal information in videos for data augmentation in VLN tasks.

References

Abu-El-Haija, S.; Kothari, N.; Lee, J.; Natsev, P.; Toderici, G.; Varadarajan, B.; and Vijayanarasimhan, S. 2016. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*.

Akbik, A.; Blythe, D.; and Vollgraf, R. 2018. Contextual String Embeddings for Sequence Labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, 1638–1649.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and van den Hengel, A. 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*, 3674–3683.

Armitage, J.; Impett, L.; and Sennrich, R. 2022. A Priority Map for Vision-and-Language Navigation with Trajectory Plans and Feature-Location Cues. *arXiv preprint arXiv:2207.11717*.

Baker, B.; Akkaya, I.; Zhokhov, P.; Huizinga, J.; Tang, J.; Ecoffet, A.; Houghton, B.; Sampedro, R.; and Clune, J. 2022. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *arXiv preprint arXiv:2206.11795*.

Chang, A.; Dai, A.; Funkhouser, T.; Halber, M.; Niessner, M.; Savva, M.; Song, S.; Zeng, A.; and Zhang, Y. 2017. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*.

Chen, H.; Suhr, A.; Misra, D.; Snavely, N.; and Artzi, Y. 2019. Touchdown: Natural Language Navigation and Spatial Reasoning in Visual Street Environments. In *Conference on Computer Vision and Pattern Recognition*.

Chen, P.; Ji, D.; Lin, K.; Zeng, R.; Li, T. H.; Tan, M.; and Gan, C. 2022a. Weakly-supervised multi-granularity map learning for vision-and-language navigation. *arXiv preprint arXiv:2210.07506*.

Chen, S.; Guhur, P.-L.; Schmid, C.; and Laptev, I. 2021. History aware multimodal transformer for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34: 5834–5847.

Chen, S.; Guhur, P.-L.; Tapaswi, M.; Schmid, C.; and Laptev, I. 2022b. Learning from Unlabeled 3D Environments for Vision-and-Language Navigation. In *European Conference on Computer Vision*, 638–655. Springer.

Chen, X. W. W. X. Y.; Darrell, F. L. V. M. T.; Yu, F.; and Chen, H. 2018. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. *arXiv preprint arXiv: 1805.04687*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dou, Z.-Y.; and Peng, N. 2022. FOAM: A Follower-aware Speaker Model For Vision-and-Language Navigation. *arXiv preprint arXiv:2206.04294*.

Fabian Caba Heilbron, B. G., Victor Escorcia; and Niebles, J. C. 2015. ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 961–970.

Fried, D.; Hu, R.; Cirik, V.; Rohrbach, A.; Andreas, J.; Morency, L.-P.; Berg-Kirkpatrick, T.; Saenko, K.; Klein, D.; and Darrell, T. 2018. Speaker-follower models for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 31.

Fu, T.-J.; Wang, X. E.; Peterson, M. F.; Grafton, S. T.; Eckstein, M. P.; and Wang, W. Y. 2020. Counterfactual vision-and-language navigation via adversarial path sampler. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, 71–86. Springer.

Goyal, R.; Ebrahimi Kahou, S.; Michalski, V.; Materzynska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; et al. 2017. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, 5842–5850.

Grauman, K.; Westbury, A.; Byrne, E.; Chavis, Z.; Furnari, A.; Girdhar, R.; Hamburger, J.; Jiang, H.; Liu, M.; Liu, X.; et al. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18995–19012.

Guhur, P.-L.; Tapaswi, M.; Chen, S.; Laptev, I.; and Schmid, C. 2021. Airbert: In-domain pretraining for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1634–1643.

Gupta, A.; Dollar, P.; and Girshick, R. 2019. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5356–5364.

Hao, W.; Li, C.; Li, X.; Carin, L.; and Gao, J. 2020. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13137–13146.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Huang, Y.; Wu, Q.; Yang, J.; An, D.; Sima, S.; and Wang, L. 2021. Landmark-RxR: Solving Vision-and-Language Navigation with Fine-Grained Alignment Supervision. *Advances in Neural Information Processing Systems*, 34: 652–663.

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

Jang, Y.; Song, Y.; Yu, Y.; Kim, Y.; and Kim, G. 2017. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2758–2766.

Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; et al. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

Kim, H.; Li, J.; and Bansal, M. 2021. Ndh-full: Learning and evaluating navigational agents on full-length dialogue. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6432–6442.

Krishna, R.; Hata, K.; Ren, F.; Fei-Fei, L.; and Carlos Niebles, J. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, 706–715.

Ku, A.; Anderson, P.; Patel, R.; Ie, E.; and Baldridge, J. 2020. Room-Across-Room: Multilingual Vision-and-Language Navigation with Dense Spatiotemporal Grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4392–4412.

Lei, J.; Yu, L.; Bansal, M.; and Berg, T. L. 2018. Tvqa: Localized, compositional video question answering. *arXiv preprint arXiv:1809.01696*.

Li, J.; Tan, H.; and Bansal, M. 2022. EnvEdit: Environment Editing for Vision-and-Language Navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15407–15417.

Lin, C.; Jiang, Y.; Cai, J.; Qu, L.; Haffari, G.; and Yuan, Z. 2022. Multimodal transformer with variable-length memory for vision-and-language navigation. In *European Conference on Computer Vision*, 380–397. Springer.

Liu, C.; Zhu, F.; Chang, X.; Liang, X.; Ge, Z.; and Shen, Y.-D. 2021. Vision-language navigation with random environmental mixup. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1644–1654.

Majumdar, A.; Shrivastava, A.; Lee, S.; Anderson, P.; Parikh, D.; and Batra, D. 2020. Improving vision-and-language navigation with image-text pairs from the web. In *European Conference on Computer Vision*, 259–274. Springer.

Mehta, H.; Artzi, Y.; Baldridge, J.; Ie, E.; and Mirowski, P. 2020. Retouchdown: Adding touchdown to streetlearn as a shareable resource for language grounding tasks in street view. *arXiv preprint arXiv:2001.03671*.

Mirowski, P.; Banki-Horvath, A.; Anderson, K.; Teplyashin, D.; Hermann, K. M.; Malinowski, M.; Grimes, M. K.; Simonyan, K.; Kavukcuoglu, K.; Zisserman, A.; et al. 2019. The streetlearn environment and dataset. *arXiv preprint arXiv:1903.01292*.

Mirowski, P.; Grimes, M.; Malinowski, M.; Hermann, K. M.; Anderson, K.; Teplyashin, D.; Simonyan, K.; Zisserman, A.; Hadsell, R.; et al. 2018. Learning to navigate in cities without a map. In *Advances in Neural Information Processing Systems*, 2419–2430.

Misra, D.; Bennett, A.; Blukis, V.; Niklasson, E.; Shatkhin, M.; and Artzi, Y. 2018. Mapping Instructions to Actions in 3D Environments with Visual Goal Prediction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2667–2678.

Nguyen, K.; and Daumé III, H. 2019. Help, Anna! Visual Navigation with Natural Multimodal Assistance via Retrospective Curiosity-Encouraging Imitation Learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Pirsiavash, H.; and Ramanan, D. 2012. Detecting activities of daily living in first-person camera views. In *2012 IEEE conference on computer vision and pattern recognition*, 2847–2854. IEEE.

Qi, Y.; Wu, Q.; Anderson, P.; Wang, X.; Wang, W. Y.; Shen, C.; and van den Hengel, A. 2020. REVERIE: Remote Embodied Visual Referring Expression in Real Indoor Environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Qiao, Y.; Qi, Y.; Hong, Y.; Yu, Z.; Wang, P.; and Wu, Q. 2022. HOP: History-and-Order Aware Pre-training for Vision-and-Language Navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15418–15427.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Rohrbach, A.; Rohrbach, M.; Tandon, N.; and Schiele, B. 2015. A dataset for movie description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3202–3212.

Schumann, R.; and Riezler, S. 2022. Analyzing Generalization of Vision and Language Navigation to Unseen Outdoor Areas. *arXiv preprint arXiv:2203.13838*.

Sharma, P.; Ding, N.; Goodman, S.; and Soricut, R. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2556–2565.

Shridhar, M.; Thomason, J.; Gordon, D.; Bisk, Y.; Han, W.; Mottaghi, R.; Zettlemoyer, L.; and Fox, D. 2020. ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tan, H.; Yu, L.; and Bansal, M. 2019. Learning to navigate unseen environments: Back translation with environmental dropout. *arXiv preprint arXiv:1904.04195*.

Thomason, J.; Murray, M.; Cakmak, M.; and Zettlemoyer, L. 2019. Vision-and-Dialog Navigation. In *Conference on Robot Learning (CoRL)*.

Wang, X.; Huang, Q.; Celikyilmaz, A.; Gao, J.; Shen, D.; Wang, Y.-F.; Wang, W. Y.; and Zhang, L. 2019. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6629–6638.

Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.-Y.; and Girshick, R. 2019. Detectron2. https://github.com/facebookresearch/detectron2.

Xiang, J.; Wang, X. E.; and Wang, W. Y. 2020. Learning to stop: A simple yet effective approach to urban vision-language navigation. *arXiv preprint arXiv:2009.13112*.

Xu, D.; Zhao, Z.; Xiao, J.; Wu, F.; Zhang, H.; He, X.; and Zhuang, Y. 2017. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, 1645–1653.

Xu, J.; Mei, T.; Yao, T.; and Rui, Y. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5288–5296.

Zhang, Q.; Peng, Z.; and Zhou, B. 2022. Learning to Drive by Watching YouTube Videos: Action-Conditioned Contrastive Policy Pretraining. *ECCV*, 2(4): 5.

Zhao, M.; Anderson, P.; Jain, V.; Wang, S.; Ku, A.; Baldridge, J.; and Ie, E. 2021. On the evaluation of vision-and-language navigation instructions. *arXiv preprint arXiv:2101.10504*.

Zhu, W.; Wang, X.; Fu, T.-J.; Yan, A.; Narayana, P.; Sone, K.; Basu, S.; and Wang, W. Y. 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*, 1207–1221. Online: Association for Computational Linguistics.

Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; and Fidler, S. 2015. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In *The IEEE International Conference on Computer Vision (ICCV)*.