A Priority Map for Vision-and-Language Navigation with Trajectory Plans and Feature-Location Cues

Jason Armitage
University of Zurich
Switzerland
jason.armitage@uzh.ch

Leonardo Impett
University of Cambridge
UK
li222@cam.ac.uk

Rico Sennrich
University of Zurich
Switzerland
sennrich@cl.uzh.ch

Abstract

In a busy city street, a pedestrian surrounded by distractions can pick out a single sign if it is relevant to their route. Artificial agents in outdoor Vision-and-Language Navigation (VLN) are also confronted with detecting supervisory signal on environment features and location in inputs. To boost the prominence of relevant features in transformer-based systems without costly preprocessing and pretraining, we take inspiration from priority maps - a mechanism described in neuropsychological studies. We implement a novel priority map module and pretrain on auxiliary tasks using low-sample datasets with high-level representations of routes and environment-related references to urban features. A hierarchical process of trajectory planning - with subsequent parameterised visual boost filtering on visual inputs and prediction of corresponding textual spans - addresses the core challenge of cross-modal alignment and feature-level localisation. The priority map module is integrated into a feature-location framework that doubles the task completion rates of standalone transformers and attains state-of-the-art performance for transformer-based systems on the Touchdown benchmark for VLN. We release code (https://github.com/JasonArmitage-res/PM-VLN) and data (https://zenodo.org/record/6891965.YtwoS3ZBxD8).

1. Introduction

Navigation in the world depends on attending to relevant cues at the right time. A road user in an urban environment is presented with billboards, moving traffic, and other people - but at an intersection will pinpoint a single light to check if it contains the colour red [12, 33]. An artificial agent navigating a virtual environment of an outdoor location is also presented with a stream of linguistic and visual cues. Action selections that move the agent closer to a final destination depend on the prioritisation of references that are relevant to the point in the trajectory. In the first example, human attention is guided to specific objects by visibility and the present objective of crossing the road. At a neurophysiological level, this process is mediated by a priority map - a neural mechanism that guides attention by matching low-level signals on salient objects with high-level signals on task goals. Prioritisation in humans is enhanced by combining multimodal signals and integration between linguistic and visual information [29, 4]. The ability to prioritise improves as experience of situations and knowledge of environments increases [41, 36].

We introduce a priority map module for Vision-and-Language Navigation (PM-VLN) that is pretrained to guide a transformer-based architecture to prioritise relevant information for action selections in navigation. In contrast to pretraining on large-scale datasets with generic image-text pairs [34], the PM-VLN module learns from small sets of samples representing trajectory plans and urban features. Our proposal is founded on observation of concentrations in location deictic terms and references to objects with high visual salience in inputs for VLN. Prominent features in the environment pervade human-generated language navigation instructions. Road network types (“intersection”), architectural features (“awning”), and transportation (“cars”) all appear with high frequency in linguistic descriptions of the visual appearance of urban locations. Learning to combine information in the two modalities relies on synchronising temporal sequences of varying lengths. We utilise references to entities as a signal for a process of cross-modal prioritisation that addresses this requirement.

Our module learns over both modalities to prioritise timely information and assist both generic vision-and-language and custom VLN transformer-based architectures to complete routes [22, 43]. Transformers have contributed to recent proposals to conduct VLN, Visual Question Answering, and other multimodal tasks - but are associated with three challenges: 1) Standard architectures lack mechanisms that address the challenge of temporal synchronisation over linguistic and visual inputs. Pretrained transformers perform well in tasks on image-text pairs but are challenged when learning over sequences without explicit alignments between modalities [24]. 2) Performance is dependent on pretraining with large sets of image-text pairs and a consequent requirement for access to enterprise-scale computational resources [28, 35]. 3) Visual learning relies on external models and pipelines - notably for object detection [23, 20]. The efficacy of object detection for VLN is low in cases where training data only refer to a small subset of object types observed in navigation environments.

We address these challenges with a hierarchical process of trajectory planning with feature-level localisation and low-sample pretraining on in-domain data. We use discriminative training on two auxiliary tasks that adapt parameters of the PM-VLN for the specific challenges presented by navigating routes in outdoor environments. High-level planning for routes is enabled by pretraining for trajectory estimation on simple path traces ahead of a second task comprising multi-objective cross-modal matching and location estimation on urban landmarks. Inputs in the final evaluation task represent locations and trajectories in large US cities and present an option to leverage real-world resources in pretraining. Our approach builds on this opportunity by sourcing data from the open web and the Google Directions API where additional samples may be secured at low cost.

This research presents four contributions to enhance transformer-based systems on outdoor VLN tasks:

- **Priority map module** Our novel PM-VLN module conducts a hierarchical process of high-level alignment of textual spans with visual perspectives and feature-level operations on inputs during navigation (see Figure 3).
- **Trajectory planning** We propose a new method for aligning temporal sequences in VLN comprising trajectory estimation on path traces and subsequent predictions for the distribution of linguistic descriptions over routes.
- **Two in-domain datasets and training strategy** We introduce a set of path traces for routes in two urban locations (TR-NY-PIT-central) and a dataset consisting of textual summaries, images, and World Geodetic System (WGS) coordinates for landmarks in 10 US cities (MC-10). These resources enable discriminative training of specific components of the PM-VLN on trajectory estimation and multi-objective loss for a new task that pairs location estimation with cross-modal sentence prediction.
- **Feature-location framework** We design and build a framework (see Figure 2) to combine the outputs from the PM-VLN module and cross-modal embeddings from a transformer-based encoder. The framework incorporates components for performing self-attention, combining embeddings, and predicting actions with maxout activation.

2. Background

In this section we define the Touchdown task and highlight a preceding challenge of aligning and localising over linguistic and visual inputs addressed in our research. A summary of the notation used below and in subsequent sections is presented in **SupMat:Sec.1**.

Touchdown Navigation in the Touchdown benchmark ϕ_{VLN} is measured as the completion of N predefined trajectories by an agent in an environment representing an area of central Manhattan. The environment is represented as an undirected graph composed of nodes O located at WGS latitude / longitude points. At each step t, the agent selects an edge ξ_t to a corresponding node. The agent's selection is based on linguistic and visual inputs. A textual instruction τ composed of a varying number of tokens describes the overall trajectory. We use ς to denote a span of tokens from τ that corresponds to the agent's location in the trajectory. Depending on the approach, ς can be the complete instruction or a selected sequence. The visual representation of a node in the environment is a panorama drawn from a sequence $Route$ of undetermined length. The agent receives a specific perspective ψ of a panorama determined by the heading angle $\angle$ between (o_1, o_2). Success in completing a route is defined as predicting a path that ends at the node designated as the goal - or one directly adjacent to it.

In a supervised learning paradigm (see Figure 1), an embedding e_η is learned from inputs ς_t and ψ_t. The agent's next action is a classification over e_η where the action α_t is one of a class drawn from the set A$\{Forward, Left, Right, Stop\}$. Predictions $\alpha_t = Forward$ and $\alpha_t = \{Left, Right\}$ result respectively in a congruent or a new $\angle$ at edge ξ_{t+1}. A route in progress is terminated by a prediction $\alpha_t = Stop$.

Align and Localise We highlight in Figure 1 a preceding challenge in learning cross-modal embeddings. As in real-world navigation, an agent is required to align and match cues in instructions with its surrounding environment. A strategy in human navigation is to use entities or landmarks to perform this alignment [4]. In the Touchdown benchmark, a relationship between sequences τ and $Route$ is assumed from the task generation process outlined in Chen *et al.* [5] - but the precise alignment is not known. We de-

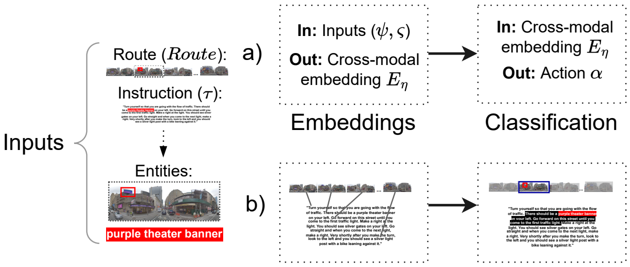
Route ($Route$):
a)
In: Inputs (ψ, ς)
Out: Cross-modal embedding E_η
In: Cross-modal embedding E_η
Out: Action α
Instruction (τ):
"Turn yourself so that you are going with the flow of traffic. There should be a purple theater banner on your left. Go forward on this street until you come to the first traffic light. Make a right at the light. You should see silver gates on your left. Go straight and when you come to the next light, make a right. Very shortly after you make the turn, look to the left and you should see a silver light post with a bike leaning against it."
Embeddings
Classification
Inputs
Entities:
purple theater banner
b)
Align
Localise

Figure 1: Outline of VLN as a supervised classification task a). Linguistic and visual inputs both refer to entities indicated in red. We address a challenge to align and localise over unsynchronised inputs b) by focusing on entities represented in both modalities.

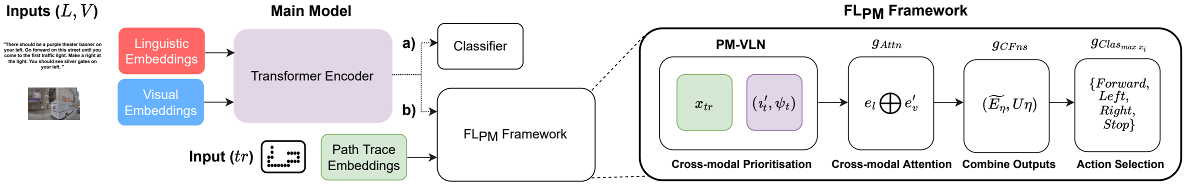

Figure 2: Prior work on transformer-based systems for VLN follows the above pipeline from inputs to the main model concluding with a) a classifier to predict actions. We propose a feature-location framework (FL$_{\text{PM}}$) to enhance the performance of a main model as in b). Here path traces are an additional input to assist the PM-VLN to align linguistic and visual sequences. Submodule g_{CFns} combines embeddings from the main model U_η and the PM-VLN $\widetilde{E_\eta}$ ahead of action prediction with maxout activation.

fine the challenge as one of aligning temporal sequences $\tau = \{\varsigma_1, \varsigma_2, \ldots, \varsigma_n\}$ and $Route = \{\psi_1, \psi_2, \ldots, \psi_n\}$ with the aim of generating a set of cross-modal embeddings E_η where referenced entities correspond. At a high level, this challenge can be addressed by an algorithm that maximises the probability P of detecting S signal on entity-related references in the linguistic and visual inputs. Formally, assuming $P_S|E_\eta$

$$g(X_t) \to \underset{subject\,to}{max} \;\; P[\tau, Route] = \prod p_{x_\varsigma x_\psi} \qquad (1)$$

where the aim of maximising P_S is equivalent to the product of probabilities in pairings between ς_t and ψ_t that contain corresponding entities.

3. Method

We address the challenge of aligning and localising over sequences with a computational implementation of cross-modal prioritisation. Diagnostics on VLN systems have placed in question the ability of agents to perform cross-modal alignment [42]. Transformers underperform in problems with temporal inputs where supervision on image-text alignments is lacking [6]. This is demonstrated in the case of Touchdown where transformer-based systems complete less than a quarter of routes. Our own observations of lower performance when increasing the depth of transformer architectures motivates moving beyond stacking blocks to an approach that compliments self-attention.

Our PM-VLN module modulates transformer-based encoder embeddings in the main task ϕ_{VLN} using a hierarchical process of operations and leveraging prior learning on auxiliary tasks (ϕ_1, ϕ_2) (see Figure 3). In order to prioritise relevant information, a training strategy for PM-VLN components is designed where training data contain samples that correspond to the urban grid type and environment features in the main task. The datasets required for pretraining contain less samples than other transformer-based VLN frameworks [43, 28] and target only specific layers of the PM-VLN module. The pretrained module is integrated in a novel feature-location framework FL$_{\text{PM}}$ shown in Figure 2. Subsequent components in the FL$_{\text{PM}}$ combine cross-modal embeddings from the PM-VLN and a main transformer model ahead of predicting an action.

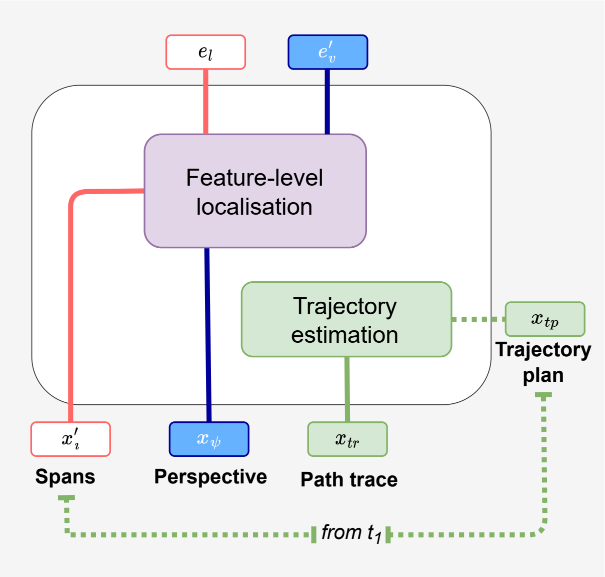

Figure 3: A Priority Map module performs a hierarchical process of high-level trajectory planning and feature-level localisation. Submodules inside the white box are learned together and a helper function generates a trajectory plan to predict spans from step t_1.

3.1. Feature-location Framework with a Priority Map Module

Prior work on VLN agents has demonstrated reliance for navigation decisions on environment features and location-related references [43]. In the definition of ϕ_{VLN} above, we consider this information as the supervisory signal contained in both sets of inputs $(x_\varsigma, x_\psi)_t$. As illustrated in Figure 2, our PM-VLN module is introduced into a framework FL$_{\text{PM}}$. This framework takes outputs from a transformer-based main model Enc_{Trans} together with path traces ahead of cross-modal prioritisation and classification with maxout activation $Clas_{max\ x_i}$. Inputs for Enc_{Trans} comprise cross-modal embeddings proposed by [43] $\bar{e}_\eta$ and a concatenation of perspectives up to the current step ψ_{cat}.

Algorithm 1 Priority Map Module

```
Input: Datasets $\mathcal{D}_{\phi_1}$,$\mathcal{D}_{\phi_2}$, and $\mathcal{D}_{\phi_{VLN}}$ with inputs $(x_l, x_v)$ for tasks $\Phi$. Initial parameters in all layers at $\Theta_j^l \sim Normal(\mu_j, \sigma_j)$.
Output: $(e_l, e_v')$
while not converged do
    for $x_{tr_i}$ in $\phi_1$ do
        $\Theta'_{g_{PMTP}} \leftarrow g_{\phi_1}(X_i, \Theta)$.
    end for
end while
while not converged do
    for $(x_{l_i}, x_{v_i})$ in $\phi_2$ do
        $\Theta'_{g_{PMF}} \leftarrow g_{\phi_2}(X_i, \Theta)$.
    end for
end while
while not converged do
    Sample $x_{tr_t}$ from $D^{Train}$.
    $x_{tp_t} \leftarrow g_{PMTP}(x_{tr_t})$.
    Sample $(\imath_t', \psi_t)$ from $D^{Train}$.
    $e_v \leftarrow g_{USM}(\psi_t)$.
    $e_v' \leftarrow g_{VBF}(e_v)$.
    $e_l \leftarrow g_{PrL}(g_{Cat}(\imath_t', e_v'))$.
end while
return $(e_l, e_v')$
```

3.1.1 Priority Map Module

Priority maps are described in the neuropsychological literature as a mechanism that modulates sensory processing on cues from the environment. Salience deriving from the physical aspects of objects in low-level processing is mediated by high-level signals for the relevance of cues to task goals [8, 18, 41]. Prioritisation of items in map tasks with language instructions indicate an integration between linguistic and visual information and subsequent increases in salience attributed to landmarks [4].

Our priority map module (PM-VLN) uses a series of simple operations to approximate the prioritsation process observed in human navigation. These operations avoid dependence on initial tasks such as object detection. Alignment of linguistic and visual inputs is enabled by trajectory estimation on simple path traces forming high-level representations of routes and subsequent generation of trajectory plans. Localisation consists of parameterised visual boost filtering on the current environment perspective ψ_t and cross-modal alignment of this view with selected spans from subsequent alignment (see Algorithm 1). This hierarchical process compliments self-attention by accounting for the lack of a mechanism in transformers to learn over unaligned temporal sequences. A theoretical basis for cross-modal prioritisation is presented in **SupMat:Sec.2**.

High-level trajectory estimation Alignment over linguistic and visual sequences is formulated as a task of predicting a set of spans from the instruction that correspond to the current step. This process starts with a submodule g_{PMTP} that estimates a count cnt of steps from a high-level view on the route (see Figure 4). Path traces - denoted as tr_T - are visual representations of trajectories generated from the coordinates of nodes. At t_0 in tr_T initial spans in the instruction are assumed to align with the first visual perspective. From step t_1, a submodule containing a pretrained ConvNeXt Tiny model [25] updates an estimate of the step count in cnt_{tr_T}. A trajectory plan tp_t is a Gaussian distribution of spans in τ within the interval $[x_{left}, x_{right}]$. At each step, samples from this distribution serve as a prediction for relevant spans. The final output $\imath_t'$ is the predicted span $\imath_t$ combined with $\imath_{t-1}$.

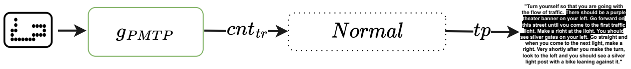
g_{PMTP}
cnt_{tr}
$Normal$
tp
"Turn yourself so that you are going with the flow of traffic. There should be a purple theater banner on your left. Go forward on this street until you come to the first traffic light. Make a right at the light. You should see silver gates on your left. Go straight and when you come to the next light, make a right. Very shortly after you make the turn, look to the left and you should see a silver light post with a bike leaning against it."

Figure 4: Submodule g_{PMTP} estimates a step count (cnt_{tr}) on a path trace. A trajectory plan (tp) is a Gaussian distribution ($Normal$) over the instruction and predicts a span for every step $\imath_t$. This is concatenated with the prediction for the previous step.

Feature-level localisation Predicted spans are passed with ψ_t to a submodule g_{PMF} that is pretrained on cross-modal matching in ϕ_2 (see Figure 5). Feature-level operations commence with visual boost filtering. We opt for a simple and efficient implementation from [3] where the level of boosting is reduced to a single learned term (see **SupMat:Sec.2** for additional details).

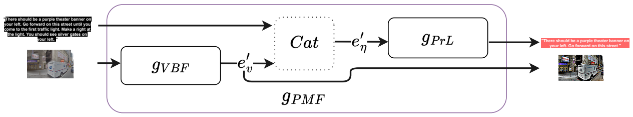
"There should be a purple theater banner on your left. Go forward on this street until you come to the first traffic light. Make a right at the light. You should see silver gates on your left."
Cat
e_η'
g_{PrL}
g_{VBF}
e_v'
g_{PMF}
"There should be a purple theater banner on your left. Go forward on this street"

Figure 5: Submodule g_{PMF} commences feature-level operations by boosting visual features in the perspective. The next operation (Cat) is a concatenation of the output from g_{VBF} and the linguistic output $\imath_t'$ from the alignment process above. A precise prediction for the relevant span e_l is returned by g_{PrL}.

Selection of a localised span e_l proceeds with a learned cross-modal embedding e_η' composed of e_v' and the linguistic output $\imath_t'$ from the preceding alignment operation. A binary prediction over this linguistic pair is performed on the output hidden state from a single-layer LSTM, which receives e_η' as its input sequence. Function g_{PrL} returns a precise localisation of relevant spans w.r.t. prominent features in the perspective:

$$g_{PrL}(e_l) = g_{Cat}(\imath_t', e_v') \triangleq \begin{cases} 0, \ if \langle w, x \rangle + b < 0 \\ 1, \ otherwise \end{cases} \qquad (2)$$

3.1.2 Cross-modal Attention and Action Prediction on Combined Outputs

Resuming operations subsequent to the PM-VLN, outputs e'_{vt} from $Conv_{VBF}$ are passed together with e_{lt} to a VisualBERT embedding layer. Embeddings for both modalities are then processed by 4 transformer encoder layers with a hidden size of 256 and self-attention $\bigoplus$ is applied to learn alignments between the pairs

$$\widetilde{e}_{\eta} = \bigoplus(e_l \Longleftrightarrow e'_v) = Soft\left(\sum_{k=1}^{\mathcal{E}} \mathcal{M}_k \text{Ł}(\mathcal{E}_k, \widetilde{\mathcal{E}}_k)\right) \quad (3)$$

where $Soft$ is the softmax function, k is the number of elements in the inputs, $\mathcal{M}_{k=1}$ is a masked element over the cross-modal inputs, Ł is the loss, $\mathcal{E}_k$ is an element in the input modality, and $\widetilde{\mathcal{E}}_k$ is the predicted element. Cross-modal embeddings resulting from this attention operation are processed by concatenating over layer outputs $g(\widetilde{e_{\eta}}') = (\widetilde{e}^1_{\mathcal{L}}, \widetilde{e}^2_{\mathcal{L}}, \widetilde{e}^3_{\mathcal{L}}, \widetilde{e}^4_{\mathcal{L}})$.

Architectural and embedding selections for our frameworks aim to enable comparison with benchmark systems on ϕ_{VLN}. The Enc_{Trans} in the best performing framework uses a standard VisualBERT encoder with a hidden size of 256 and 4 layers and attention heads. As noted above, inputs for Enc_{Trans} align with those used in prior work [43]. A submodule g_{CFns} combines U_{η} from $\mathcal{L}^4$ of the Enc_{Trans} and outputs from the cross-modal attention operation $g(\widetilde{E}'_{\eta})$ ahead of applying dropout. Predictions for navigation actions are the outputs of a classifier block consisting of linear layers with maxout activation. Maxout activation in a block composed of linear operations takes the $maxz_{ij}$ where z_{ij} are the product of $x_{ij}W_{n*}$ for k layers. In contrast to ReLU, the activation function is learned and prevents unit saturation associated with performing dropout [11]. We compare a standard classifier to one with $max\ x_i$ in Table 2. Improvements with $max\ x_i$ are consistent with a requirement to offset variance when training with the high number of layers in the full FL$_{\text{PM}}$ framework.

3.2. Pretraining Strategy

A data-efficient pretraining strategy for the PM-VLN module consists of pretraining submodules of the PM-VLN on auxiliary tasks (ϕ_1, ϕ_2). We denote the two datasets as $(\mathcal{D}_{\phi_1}, \mathcal{D}_{\phi_2})$ and a training partition as $\mathcal{D}^{Train}$. In ϕ_1, the g_{PMTP} submodule is pretrained on TR-NY-PIT-central - a new set of path traces. Path traces in $D^{Train}_{\phi_1}$ are generated from 17,000 routes in central Pittsburgh with a class label for the step count in the route. The distribution of step counts in $D^{Train}_{\phi_1}$ is 50 samples for routes with $\leq$7 steps and 300 samples for routes with $>$7 steps (see **SupMat:Sec.3** for further details). During training, samples from $D^{Train}_{\phi_1}$ are presented in standard orientation for 20 epochs and rotated 180° ahead of a second round of training. This rotation policy is preferred following empirical evaluation using standalone versions of the g_{PMTP} submodule receiving two alternate preparations of $D^{Train}_{\phi_1}$ with random and 180° rotations. Training is formulated as multiclass classification with cross-entropy loss on a set of M=66 classes

$$g_{\phi_1}(x_{tr}, \Theta) = B_0 + \underset{i}{argmax} \sum_{j=1}^{M} B_i(x_{tr}, W_j) \quad (4)$$

where a class is the step count, B is the bias, and i is the sample in the dataset.

Pretraining on ϕ_2 for the feature-level localisation submodule g_{PMF} is conducted with the component integrated in the framework FL$_{\text{PM}}$ and the new MC-10 dataset. Samples in $D^{Train}_{\phi_2}$ consist of 8,100 landmarks in 10 US cities. To demonstrate the utility of open source tools in designing systems for outdoor VLN, the generation process leverages free and accessible resources that enable targeted querying. Entity IDs for landmarks sourced from the Wikidata Knowledge Graph are the basis for downloading textual summaries and images from the MediaWiki and WikiMedia APIs. Additional details on MC-10 are available in **SupMat:Sec.3**. The aim in generating the MC-10 dataset is to optimise $\Theta_{g_{PMF}}$ such that features relating to $Y_{\phi_{VLN}}$ are detected in inputs $X_{\phi_{VLN}}$. We opt for open A multi-objective loss for ϕ_2 consists of cross-modal matching over the paired samples (x_l, x_v) - and a second objective comprising a prediction on the geolocation of the entity. In the first objective, g_{PMF} conducts a binary classification between the true x_l matching x_v and a second textual input selected at random from entities in the mini-batch. A limit of 540 tokens is set for all textual inputs and the classification in g_{PMF} is performed on the first sentence for each entity. Parameters $\Theta_{g_{PMF}}$ are saved and used subsequently for feature-level localisation in ϕ_{VLN}.

4. Experiments

Our starting point in evaluating the PM-VN module and FL$_{\text{PM}}$ is performance in relation to benchmark systems (see Table 1). Ablations are conducted by removing individual operations (see Table 2) and the role of training data is assessed (see Table 3). To minimise computational cost, we implement frameworks with low numbers of layers and attention heads in transformer models.

4.1. Experiment Settings

Metrics We align with [5] in reporting task completion (TC), shortest-path distance (SPD), and success weighted edit distance (SED) for ϕ_{VLN}. All metrics are derived using the Touchdown navigation graph. TC is a binary measure of success $0, 1$ in ending a route with a prediction $c^o_{t-1} = y^o_{t-1}$ or $c^o_{t-1} = y^{o-1}_{t-1}$ and SPD is calculated as the mean distance between c^o_{t-1} and y^o_{t-1}. SED is the Levenshtein distance

		Development			Test		
		TC↑	SPD↓	SED↑	TC↑	SPD↓	SED↑
Inputs (L, V) (non-transformer based)	GA [a]	12.1	20.2	11.7	10.7	19.9	10.4
	RCONCAT [a]	11.9	20.1	11.5	11.0	20.4	10.5
	ARC+L2STOP* [c]	19.5	17.1	19.0	**16.7**	18.8	16.3
Inputs (L, V) (transformer based)	VisualBERT(8l)	10.4	21.3	10.0	9.9	21.7	9.5
	VisualBERT(4l)	14.3	17.7	13.7	11.8	18.3	11.5
	VLN Transformer(4l) [b]	12.2	18.9	12.0	12.8	20.4	11.8
	VLN Transformer(8l) [b]	13.2	19.8	12.7	13.1	21.1	12.3
	VLN Transformer(8l) + M50 + style * [b]	15.0	20.3	14.7	**16.2**	20.8	15.7
Inputs (L, V) + JD / HT** (non-transformer based)	ORAR (ResNet pre-final)* [d]	26.0	15.0	-	25.3	16.2	-
	ORAR (ResNet 4th-to-last)* [d]	29.9	11.1	-	**29.1**	11.7	-
Inputs (L, V) + Path Traces (transformer based)	VLN Transformer(8l)	11.2	23.4	10.7	11.5	23.9	10.8
	VisualBERT(4l)	16.2	18.7	15.7	15.0	20.1	14.5
	FL_{PM}(4l) + VLN Transformer(8l)	29.9	23.4	26.8	28.2	23.8	25.6
	FL_{PM}(4l) + VisualBERT(4l)	33.0	23.6	29.5	**33.4**	23.8	29.7

Frameworks from [a] [5], [b] [43], [c] [39], and [d] [31].

* Results reported by the authors.

** Systems receive two types of features - Junction Type and Heading Delta - as inputs.

Table 1: Performance on the Touchdown benchmark ranked by TC on the test partition. Systems are grouped by input types during VLN and the use of transformer blocks in architectures. Contributions of the FL_{PM} framework and path traces to improved performance are demonstrated with results for systems with two baseline transformer-based architectures - VisualBERT and VLN Transformer. These baselines also are assessed in two sizes to test the benefits of adding transformer blocks.

between the predicted path in relation to the defined route path and is only applied when TC = 1.

Hyperparameter Settings Frameworks are trained for 80 epochs with batch size=30. Scores are reported for the epoch with the highest SPD on $\mathcal{D}^{Dev}_{\phi_{VLN}}$. Pretraining for the PM-VLN module is conducted for 10 epochs with batch sizes $\phi_1 = 60$ and $\phi_2 = 30$. Frameworks are optimised using AdamW with a learning rate of 2.5 x 10^{-3} [26].

4.2. Touchdown

Experiment Design: [5] define two separate tasks in the Touchdown benchmark: VLN and spatial description resolution. This research aligns with other studies [43, 42] in conducting evaluation on the navigation component as a standalone task. **Dataset and Data Preprocessing:** Frameworks are evaluated on full partitions of Touchdown with $D^{Train} = 6,525$, $D^{Dev} = 1,391$, and $D^{Test} = 1,409$ routes. Trajectory lengths vary with $D^{Train} = 34.2$, $D^{Dev} = 34.1$, and $D^{Test} = 34.4$ mean steps per route. Junction Type and Heading Delta are additional inputs generated from the environment graph and low-level visual features [31]. M-50 + style is a subset of the StreetLearn dataset with $D^{Train} = 30,968$ routes of 50 nodes or less and multimodal style transfer applied to instructions [43]. **Embeddings:** All architectures evaluated in this research receive the same base cross-modal embeddings x_η proposed by [43], which are learned by a combination of the outputs of a pretrained BERT-base encoder with 12 encoder layers and attention heads. At each step, a fully connected layer is used for textual embeddings ς_t and a 3 layer CNN returns the perspective ψ_t. FL_{PM} frameworks also receive an embedding of the path trace tr_t at step t. As this constitutes additional signal on the route, we evaluate a VisualBERT model (4l) that also receives tr_t, which in this case is combined with ψ_t ahead of inclusion in x_{η_t}. **Results:** In Table 1 the first block of frameworks consists of architectures composed primarily of convolutional and recurrent layers. VLN Transformer is a framework proposed by [43] for the Touchdown benchmark and consists of a transformer-based cross-modal encoder with 8 encoder layers and 8 attention heads. VLN Transformer + M50 + style is a version of this framework pretrained on the dataset described above. To our knowledge, this was the transformer-based framework with the highest TC on Touchdown preceding our work. ORAR (ResNet 4th-to-last) [31] is from work published shortly before the completion of this research and uses two types of features to attain highest TC in prior work. Standalone VisualBERT models are evaluated in two versions with 4 and 8 layers and attention heads. A stronger performance by the smaller version indicates that adding self-attention layers is unlikely to improve VLN predictions. This is further supported by the closely matched results for the VLN Transformer(4l) and VLN Transformer(8l). FL$_{\mathrm{PM}}$ frameworks incorporate the PM-VLN module pretrained on auxiliary tasks (ϕ_1, ϕ_2) - and one of VisualBERT (4l) or VLN Transformer(8l) as the main model. Performance on TC for both of these transformer models doubles when integrated into the framework. A comparison of results for standalone VisualBERT and VLN Transformer systems with path traces supports the use of specific architectural components that can exploit this additional input type. Lower SPD for systems run with the FL$_{\mathrm{PM}}$ framework reflect a higher number of routes where a stop action was predicted prior to route completion. Although not a focus for the current research, this shortcoming in VLN benchmarks has been addressed in other work [39, 2].

4.3. Assessment of Specific Operations

Ablations are conducted on the framework with the highest TC *i.e.* FL$_{\mathrm{PM}}$ + VisualBERT(4l). The tests do not provide a direct measure of operations as subsequent computations in forward and backward passes by retained components are not accounted for. Results indicate that initial alignment is critical to cross-modal prioritisation and support the use of in-domain data during pretraining.

	Development		
	TC↑	**SPD↓**	**SED↑**
FL$_{\mathrm{PM}}$ + VisualBERT(4l)	33.0	23.6	29.5
PM-VLN			
- g_{PMTP} (a)	7.1	26.8	6.8
- g_{PMF} minus $\mathrm{g}_{\mathrm{VBF}}$ (b)	27.9	25.7	24.9
- g_{PMF} minus $\imath_{t-1}$ (c)	29.8	21.8	27.2
FL$_{\mathrm{PM}}$			
- g_{Attn} with g_{Cat} (d)	18.8	30.5	16.4
- $g_{Clas_{max\ x_i}}$ with g_{Clas} (e)	31.7	21.9	28.2

Table 2: Ablations on core operations in the PM-VLN (variants (a-(c)) and the FL$_{\mathrm{PM}}$ framework (variants (d) and (e)).

Ablation 1: PM-VLN Prioritisation in the PM-VLN module constitutes a sequential chain of operations. Table 2 reports results for variants of the framework where the PM-VLN excludes individual operations. Starting with g_{PMTP}, trajectory estimation is replaced with a fixed count of 34 steps for each route tr_t (see variant (a)). This deprives the PM-VLN of a method to take account of the current route when synchronising τ and sequences of visual inputs. All subsequent operations are impacted and the variant reports low scores for all metrics. Two experiments are then conducted on g_{PMF}. In variant (b), visual boost filtering is disabled and feature-level localisation relies on a base ψ_t. A variant excluding linguistic components from g_{PMF} is then implemented by specifying $\imath_t$ as the default input from τ_t (see variant (c)). In practice, span selection in this case is based on trajectory estimation only.

Ablation 2: FL$_{\mathrm{PM}}$ Ablations conclude with variants of FL$_{\mathrm{PM}}$ where core functions are excluded from other submodules in the framework. Results for variant (d) demonstrate the impact of replacing the operation defined in Equation 3 with a simple concatenation on outputs from PM-VLN e_l and e'_v. A final experiment compares methods for generating action predictions: in variant (e), $g_{Clas_{max\ x_i}}$ is replaced by the standard implementation for classification in VisualBERT. Classification with dropout and a single linear layer underperforms our proposal by 1.3 points on TC.

4.4. Assessment of Training Strategy

A final set of experiments is conducted to measure the impact of training data for auxiliary tasks (ϕ_1, ϕ_2).

	Development		
	TC↑	**SPD↓**	**SED↑**
FL$_{\mathrm{PM}}$ + VisualBERT(4l)	33.0	23.6	29.5
Pretraining for g_{PMTP}			
- $g_{PMTP} + D_{\phi_1}^{Train} V2$ (f)	11.9	20.1	11.5
- $g_{PMTP} + D_{\phi_1}^{Train} V3$ (g)	13.6	20.5	13.1
- g_{PMTP} no pretraining (h)	4.7	27.6	1.9
Pretraining for g_{PMF}			
- $g_{PMF} + D_{\phi_2}^{Train} V2$ (i)	19.8	23.2	17.2
- $g_{PMF} + D_{\phi_2}^{Train} V3$ (j)	23.9	20.8	20.3
- g_{PMF} no pretraining (k)	6.3	25.1	4.6

Table 3: Assessment of the pretraining strategy for individual PM-VLN submodules g_{PMTP} (variants (f) to (h)) and g_{PMF} (variants (i) to (k) using alternative datasets for auxiliary tasks. Variants are also run with no pretraining of g_{PMTP} and g_{PMF}.

Training Strategy 1: Exploiting Street Pattern in Trajectory Estimation We conduct tests on alternate samples to examine the impact of route types in $D_{\phi_1}^{Train}$. The module for FL$_{\mathrm{PM}}$ frameworks in Table 1 is trained on path traces drawn from an area in central Pittsburgh (see **SupMat:Sec.3**) with a rectangular street pattern that aligns with the urban grid type [27] found in the location of routes in Touchdown. Table 3 presents results for modules trained on routes selected at random outside of this area. In variants (f) and (g), versions V2 and V3 of $D_{\phi_1}^{Train}$ each consist of 17,000 samples drawn at random from the remainder of a total set of 70,000 routes. Routes that conform to curvilinear grid types are observable in outer areas of Pittsburgh. Lower TC for these variants prompts consideration of street patterns when generating path traces. A variant (h) where the g_{PMTP} submodule receives no pretraining underlines - along with variant (a) in Table 2 - the importance of the initial alignment step to our proposed method of cross-modal prioritisation.

Training Strategy 2: In-domain Data and Feature-Level Localisation We conclude by examining the use of in-domain data when pretraining the g_{PMF} submodule ahead of feature-level localisation operations in the PM-VLN. In Table 3, versions of FL_{PM} are evaluated subsequent to pretraining with varying sized subsets of the Conceptual Captions dataset [32]. This resource of general image-text pairs is selected as it has been proposed for pretraining VLN systems (see below). Samples are selected at random and grouped into two training partitions equivalent in number to 100% (variant (i)) and 150% of $D^{Train}_{\phi_2}$ (variant (j)). In place of the multi-objective loss applied to the MC-10 dataset, $\theta_{g_{PMF}}$ are optimised on a single goal of cross-modal matching. Variant (k) assesses FL_{PM} when no pretraining for the g_{PMF} submodule is undertaken. Lower results for variants (i), (j), and (k) support pretraining on small sets of in-domain data as an alternative to optimising VLN systems on large-scale datasets of general samples.

5. Related Work

This research aims to extend cross-disciplinary links between machine learning and computational cognitive neuroscience in the study of prioritisation in attention. This section starts with computational methods used to explore this subject in these two disciplines. Our training strategy is positioned in the context of prior work on pretraining for VLN tasks and research related to the alignment and feature-level operations performed by the PM-VLN module is reviewed.

Computational Implementations of Prioritisation in Attention [7] proposed a model that generates saliency maps where feature selection is dependent on high-level signals in the task. The full system was evaluated on computer vision tasks where the aim is to track targets in video. A priority map computation was implemented in object detection models by [38] to compare functions in these systems to those observed in human visual attention. [1] used a Support Vector Machine classifier to model visual attention in human participants traversing four terrains. Priority maps were then generated to study the interaction of prioritised features and a high-level goal of maintaining smooth locomotion. A priority map component was incorporated into a CNN-based model of primate attention mechanisms by [40] to prioritise locations containing classes of interest when performing visual search. Studies on spatial attention in human participants have explored priority map mechanisms that process inputs consisting of auditory stimuli and combined linguistic and visual information [10, 4]. To our knowledge, our work is the first to extend neuropsychological work on prioritisation over multiple modalities to a computational implementation of a cross-modal priority map for machine learning tasks.

Pretraining for VLN Tasks Two forms of data samples - in-domain and generic - are used in pretraining prior to conducting VLN tasks. In-domain data samples have been sourced from image-caption pairs from online rental listings [13] and other VLN tasks [43]. In-domain samples have also been generated by augmenting or reusing in-task data [9, 17, 14, 15, 31]. Generic samples from large-scale datasets designed for other Vision-Language tasks have been sourced to improve generalisation in transformer-based VLN agents. [28] conduct large-scale pretraining with 3.3M image-text pairs from Conceptual Captions [32] and [30] initialise a framework with weights trained on four out-of-domain datasets. In contrast our training strategy employs datasets with a few thousand samples of in-domain data derived from resources where additional samples are available at low cost.

Methods for Aligning and Localising Features in Linguistic and Visual Sequences Alignment in multimodal tasks is often posited as an implicit subprocess in an attention-based component of a transformer [37, 43]. [17] identified explicit cross-modal alignment as an auxiliary task that improves agent performance in VLN. Alignment in this case is measured as a similarity score on inputs from the main task. In contrast, our PM-VLN module conducts a hierarchical process of trajectory planning and learned localisation to pair inputs. A similarity measure was the basis for an alignment step in the Vision-Language Pretraining framework proposed by [21]. A fundamental point of difference with our work is that this framework - along with related methods [19] - is trained on a distinct class of tasks where the visual input is a single image as opposed to a temporal sequence. Several VLN frameworks containing components that perform feature localisation on visual inputs have been pretrained on object detection [28, 35, 16]. In contrast, we include visual boost filtering in g_{PMF} to prioritise visual features. Our method of localising spans using a concatenation of the enhanced visual input and cross-modal embeddings is unique to this research.

6. Conclusion

We take inspiration from a mechanism described in neurophysiological research with the introduction of a priority map module that combines temporal sequence alignment enabled by high-level trajectory estimation and feature-level localisation. Two new resources comprised of in-domain samples and a tailored training strategy are proposed to enable data-efficient pretraining of the PM-VLN module ahead of the main VLN task. A novel framework enables action prediction with maxout activation on a combination of the outputs from the PM-VLN module and a transformer-based encoder. Evaluations demonstrate that our module, framework, and pretraining strategy double the performance of standalone transformers in outdoor VLN.

References

[1] Nantheera Anantrasirichai, Katherine AJ Daniels, Jeremy F Burn, Iain D Gilchrist, and David R Bull. Fixation prediction and visual priority maps for biped locomotion. *IEEE Transactions on Cybernetics*, 48(8):2294–2306, 2017.

[2] Valts Blukis, Dipendra Misra, Ross A Knepper, and Yoav Artzi. Mapping navigation instructions to continuous control actions with position-visitation prediction. In *Conference on Robot Learning*, pages 505–518. PMLR, 2018.

[3] Jose Carranza-Rojas, Saul Calderon-Ramirez, Adán Mora-Fallas, Michael Granados-Menani, and Jordina Torrents-Barrena. Unsharp masking layer: injecting prior knowledge in convolutional networks for image classification. In *International Conference on Artificial Neural Networks*, pages 3–16. Springer, 2019.

[4] Federica Cavicchio, David Melcher, and Massimo Poesio. The effect of linguistic and visual salience in visual world studies. *Frontiers in Psychology*, 5:176, 2014.

[5] Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 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, 2019.

[6] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10638–10647, 2020.

[7] Misha Denil, Loris Bazzani, Hugo Larochelle, and Nando de Freitas. Learning where to attend with deep architectures for image tracking. *Neural Computation*, 24(8):2151–2184, 2012.

[8] Jillian H Fecteau and Douglas P Munoz. Salience, relevance, and firing: a priority map for target selection. *Trends in Cognitive Sciences*, 10(8):382–390, 2006.

[9] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 31, 2018.

[10] Edward J Golob, Kristen Brent Venable, Jaelle Scheuerman, and Maxwell T Anderson. Computational modeling of auditory spatial attention. In *CogSci*, 2017.

[11] Ian Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. In *International Conference on Machine Learning*, pages 1319–1327. PMLR, 2013.

[12] Jacqueline Gottlieb, Michael Cohanpour, Yvonne Li, Nicholas Singletary, and Erfan Zabeh. Curiosity, information demand and attentional priority. *Current Opinion in Behavioral Sciences*, 35:83–91, 2020.

[13] Pierre-Louis Guhur, Makarand Tapaswi, Shizhe Chen, Ivan Laptev, and Cordelia Schmid. Airbert: In-domain pretraining for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1634–1643, 2021.

[14] Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. 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*, pages 13137–13146, 2020.

[15] Keji He, Yan Huang, Qi Wu, Jianhua Yang, Dong An, Shuanglin Sima, and Liang Wang. Landmark-rxr: Solving vision-and-language navigation with fine-grained alignment supervision. *Advances in Neural Information Processing Systems*, 34, 2021.

[16] Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. Are you looking? grounding to multiple modalities in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6551–6557, Florence, Italy, July 2019. Association for Computational Linguistics.

[17] Haoshuo Huang, Vihan Jain, Harsh Mehta, Alexander Ku, Gabriel Magalhaes, Jason Baldridge, and Eugene Ie. Transferable representation learning in vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7404–7413, 2019.

[18] Laurent Itti and Christof Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10-12):1489–1506, 2000.

[19] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021.

[20] Tung Le, Khoa Pho, Thong Bui, Huy Tien Nguyen, and Minh Le Nguyen. Object-less vision-language model on visual question classification for blind people. In *ICAART (3)*, pages 180–187, 2022.

[21] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34, 2021.

[22] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

[23] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020.

[24] Yan-Bo Lin and Yu-Chiang Frank Wang. Audiovisual transformer with instance attention for audio-visual event localization. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[25] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[27] Kevin Lynch. A theory of good city form. ma. *Cambridge: Massachusetts Institute of Technology*, 1981.

[28] Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter An-

derson, Devi Parikh, and Dhruv Batra. Improving vision-and-language navigation with image-text pairs from the web. In *European Conference on Computer Vision*, pages 259–274. Springer, 2020.

[29] Radek Ptak. The frontoparietal attention network of the human brain: action, saliency, and a priority map of the environment. *The Neuroscientist*, 18(5):502–515, 2012.

[30] Yuankai Qi, Zizheng Pan, Yicong Hong, Ming-Hsuan Yang, Anton van den Hengel, and Qi Wu. The road to know-where: An object-and-room informed sequential bert for indoor vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1655–1664, 2021.

[31] Raphael Schumann and Stefan Riezler. 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, May 2022. Association for Computational Linguistics.

[32] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 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)*, pages 2556–2565, 2018.

[33] Hiroyuki Shinoda, Mary M Hayhoe, and Anurag Shrivastava. What controls attention in natural environments? *Vision Research*, 41(25-26):3535–3545, 2001.

[34] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. VL-BERT: pre-training of generic visual-linguistic representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[35] Alessandro Suglia, Qiaozi Gao, Jesse Thomason, Govind Thattai, and Gaurav Sukhatme. Embodied bert: A transformer model for embodied, language-guided visual task completion. *arXiv preprint arXiv:2108.04927*, 2021.

[36] Benjamin W Tatler, Mary M Hayhoe, Michael F Land, and Dana H Ballard. Eye guidance in natural vision: Reinterpreting salience. *Journal of Vision*, 11(5):5–5, 2011.

[37] Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2019, page 6558. NIH Public Access, 2019.

[38] Zijun Wei, Hossein Adeli, Minh Hoai Nguyen, Greg Zelinsky, and Dimitris Samaras. Learned region sparsity and diversity also predicts visual attention. *Advances in Neural Information Processing Systems*, 29, 2016.

[39] Jiannan Xiang, Xin Wang, and William Yang Wang. 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, Nov. 2020. Association for Computational Linguistics.

[40] Gregory J Zelinsky and Hossein Adeli. Learning to attend in a brain-inspired deep neural network. *Journal of Vision*, 19(10):282d–282d, 2019.

[41] Gregory J Zelinsky and James W Bisley. The what, where, and why of priority maps and their interactions with visual working memory. *Annals of the New York Academy of Sciences*, 1339(1):154, 2015.

[42] Wanrong Zhu, Yuankai Qi, Pradyumna Narayana, Kazoo Sone, Sugato Basu, Xin Eric Wang, Qi Wu, Miguel Eckstein, and William Yang Wang. Diagnosing vision-and-language navigation: What really matters. *arXiv preprint arXiv:2103.16561*, 2021.

[43] Wanrong Zhu, Xin Wang, Tsu-Jui Fu, An Yan, Pradyumna Narayana, Kazoo Sone, Sugato Basu, and William Yang Wang. 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, 2021.