IGNAVFM: A FOUNDATION MODEL FOR IMAGE-GOAL NAVIGATION

PREPRINT

Navigation Research Team under MSRA ML Group*
Contact: pushizhang@microsoft.com

February 19, 2025

ABSTRACT

Image-goal navigation is a fundamental problem in robot learning, requiring agents to reach a target location based solely on an image observation at that location. In this paper, we introduce the Image-Goal Navigation Foundation Model (IGNavFM), a novel framework that advances the state-of-the-art through several key innovations. First, IGNavFM introduces an early-fusion network that integrates observation and goal images at an early stage, significantly enhancing the model's navigational capabilities. Second, we leverage pretrained Vision Transformer (ViT) encoders to improve representation learning, allowing for more effective generalization. Third, we propose auxiliary loss functions that optimize both the pretraining and fine-tuning phases, leading to more robust and data-efficient adaptation. We evaluate IGNavFM across two game-based environments and a real-world robotic setup, demonstrating superior zero-shot generalization and fine-tuning efficiency compared to existing navigation foundation models. Notably, IGNavFM achieves competitive performance with significantly reduced fine-tuning data, underscoring its potential for real-world deployment with minimal labeled supervision.

1 Introduction

Vision-based navigation is the capability of an autonomous agent to navigate toward a specific location or object using visual observations and a reference image. This field has been a major research focus due to its broad practical applications, including home automation and search-and-rescue missions [Wu et al., 2022, Kim et al., 2023].

Traditional navigation methods primarily rely on Simultaneous Localization and Mapping (SLAM) or map-based strategies to guide robots using environmental information [Chaplot et al., 2020, Temeltas and Kayak, 2008, Milford and Wyeth, 2010]. However, in many scenarios where localization is difficult, or the agent interacts with dynamic environment and objects, agents must depend on their own exploration and navigation capabilities. Additionally, reinforcement learning (RL)-based approaches, which rely on interactions with the environment, often suffer from slow training times and limited generalizability [Zhu et al., 2017, Lobos-Tsunekawa et al., 2018, Yadav et al., 2023].

Recently, foundation models have emerged as a transformative paradigm in machine learning, enabling agents to generalize more effectively with less training data. As environments grow increasingly complex, traditional approaches struggle to scale due to their high variability and data demands. Navigation foundation models [Shah et al., 2023a,b, Sridhar et al., 2024, Zhang et al., 2024a,b] address these limitations by leveraging their large-scale generalization capabilities across diverse environments. These works train foundation agents for navigation that generalize across new environments and new tasks, either in the form of language instructions, or object goals and image goals.

To further extend the success of navigation foundation models, we raise IGNavFM, a foundation model for image-goal navigation. We advance the state-of-the-art in navigation foundation models through several key contributions. First, our network structure design enables the modeling of the low-level correspondence between image and goals with high-capacity networks. We achieve this by integrating a novel early fuse network structure, and pretrained Vision Transformer (ViT) encoder into our model. Also, we use auxiliary tasks that capture global decision making information

*Work in progress.

to train IGNavFM, which improves the navigation performance. To comprehensively evaluate the performance of IGNavFM, we conduct experiments on two simulation-based game environments and one real-world robotic environment, comparing our approach with multiple baselines. The results demonstrate the superiority of our model in terms of generalizability, downstream finetuning performance, and its utility as a foundation model.

We demonstrate the following findings through our experiments:

1. IGNavFM outperforms all baselines compared to former navigation foundation model including GNM and ViNT, showing an average improvement of 18.1% in navigation success rate for zero-shot generalization settings, and an average improvement of 40.7% for finetune settings across all tasks.
2. The pretraining of IGNavFM significantly enhances learning efficiency in downstream environments, reducing data requirements by approximately eightfold.
3. The usage of early-fuse network structure and appropriate pretrained Vision Transformer (ViT) improves the navigation performance significantly.
4. Auxiliary tasks that capture global decision making information for navigation improves the performance of the foundation model.

The remainder of this paper is organized as follows. Section 2 provides a brief review of the literature on image-goal navigation. Section 3 details the proposed model. Experiments and result analysis are presented in Section 4. Finally, conclusions are drawn in Section 5.

2 Related Work

Visual navigation for autonomous agents [Sun et al., 2024] involves guiding an agent from its current location to a target position specified by visual cues, typically derived from camera observations. This broad task can be divided into several subtasks, including object navigation, where the agent searches for an object identified by an image, and image goal navigation, where the goal is to navigate to a specific location represented by an input goal image. In this work, we focus on the image goal navigation task.

Early approaches to image goal navigation relied on handcrafted features to compare the goal image with the agent's current view and estimate the direction towards the goal. However, these methods were effective in simple environments, and they often struggled in complex or dynamic scenes due to their sensitivity to changes in lighting, occlusions, and scene dynamics. To address these limitations, some studies integrated geometric models of the environment. For example, [Kwon et al., 2023] proposed creating a 3D map of the environment and projecting the goal image onto this map to calculate the navigation path. Although this approach improved accuracy in structured environments, it required significant prior knowledge and was computationally expensive, limiting its scalability.

Another promising direction combines deep learning with reinforcement learning (RL). In these methods [Yadav et al., 2023, Sun et al., 2024], agents learn to navigate towards the goal image through interaction with the environment, guided by a reward function. [Yadav et al., 2023] introduced an RL-based approach where agents mapped visual inputs (current and goal images) to actions that maximized cumulative rewards. While this approach enables adaptability to diverse environments, it often suffers from slow convergence and challenges in designing effective reward functions.

Also, a lot of vision-language-action model inspired by the success of large language models (LLMs) in natural language processing [Liu et al., 2024, Zhang et al., 2024c], researchers have begun developing large-scale, pretrained foundation models for visual navigation [Shah et al., 2023a,b] or invite LLMs directly [Zhou et al., 2024]. Foundation models aim to encapsulate extensive prior knowledge across diverse navigation scenarios, allowing for fine-tuning on specific tasks. This approach has the potential to reduce training costs, improve generalization, and enhance performance in various operational environments. However, pretraining such large-scale foundation model is not easy, which requires collecting tremendous navigation data and effective network design.

Another direction is vision-language navigation [Zhou et al., 2024, Chen et al., 2024, Zhang et al., 2024b], Zhou et al. [2024] introduce NavGPT-2, integrates vision language models (VLMs) to align visual observations with language instructions, to enable the system to generate navigation actions while producing interpretable navigation decisions. However, due to the significant computational demands, its real-time application remains challenging, and the VLM's scene understanding has not yet met deployment standards.

With the passion of training a generalist model in embodied AI area, quantities of Vision-Language-Action (VLA) models as [Brohan et al., 2022, Kim et al., 2024, Black et al., 2024] were mentioned as foundation models that understands real world environment observations, language instruction and takes agent action. While these works have shown great generalization capabilities on robot arm manipulations, we discuss multiple key design choices that is

key component to affect the foundation model's performance on the image-goal settings which is less studied by the mainstream VLA models.

For further extension with combination of vision, language and action. The VLA foundation model as Kim et al. [2024] can also be applied in short term navigation tasks. OpenVLA [Kim et al., 2024] introduces the vision-language-action model which obtains the strong performance for embodiment robot control based on diverse interaction inputs as image and language instructions.

3 Proposed Model: IGNavFM

In this section, we provide a detailed explanation of our proposed model IGNavFM, including its architecture and training methodology, which incorporates auxiliary training tasks. Figure 1 presents an overview of IGNavFM, which is pretrained on diverse indoor navigation, outdoor navigation and autonomous driving datasets. This work begins by collecting diverse trajectory data consisting of RGB images and corresponding positional and angle information. Using these datasets, we pretrain our image-goal navigation foundation model to learn generalizable navigation representations. Subsequently, we adapt the pretrained foundation model to downstream tasks by finetuning it on additional environment-specific datasets. In details, we introduce the whole model architecture of IGNavFM in Section 3.1, and then follows the training explanation in Section 3.2, which details the main training task and the design of auxiliary tasks.

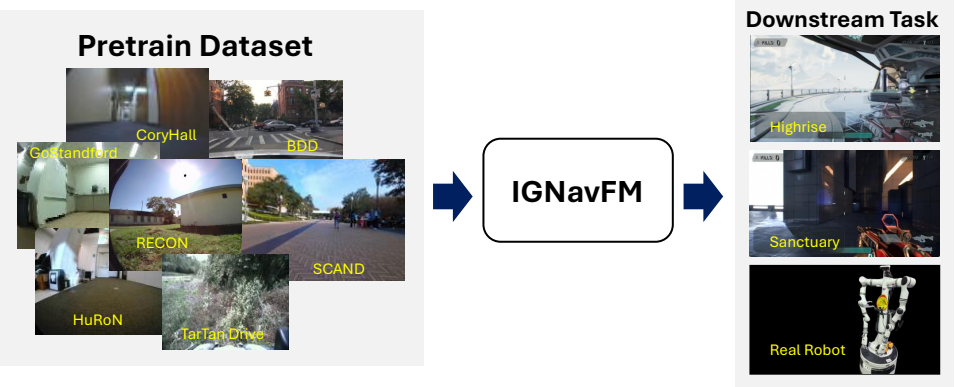


Figure 1: Overview of IGNavFM. Pretrain datasets come from seven open source navigation video datasets. Downstream environments include two Unreal-Engine-based FPS game environment Highrise and Sanctuary and one real world environment with robot.

3.1 Model Architecture

Image-goal navigation tasks require the agent to model the nuanced spatial relationships between observations and goal images. Motivated by this, our design of NavFM models the low-level correspondence between image and goals with high-capacity networks. To achieve this, we integrate a novel early fuse network structure for IGNavFM to capture a joint distribution of observation and goal images, and leverage pretrained Vision Transformer (ViT) encoder to enhance representation learning.

Figure 2 illustrates the network architecture of our model. The current observation image and the goal image are processed into patches embeddings separately, and then fuse together into multiple patches of visual tokens. These patch tokens are added by the learnable tokens for observation and goal image respectively, and then jointly fed into the ViT encoder, which aims to find the correspond between the low-level features of the observation and goal images. We refer this design as early-fuse model architecture as the transformer encoder takes the low-level features of both observation and goals as input.

The learnable [CLS] token is used as an additional input of the transformer encoder to extract the joint representations of the image inputs, where the output embedding of the token is subsequently fed into separate MLP layers as the contextual embeddings to generate waypoint action outputs and auxiliary outputs. The action heads decode the next $N_{\text{waypoint}} = 10$ actions, where we use 4-dimensional waypoint action space $[\Delta x, \Delta y, \cos \Delta\psi, \sin \Delta\psi]$, where $\Delta x, \Delta y, \Delta\psi$ are the 2-dimensional movements and rotations of the agent. The prediction targets of auxiliary tasks are detailed in Section 3.2.

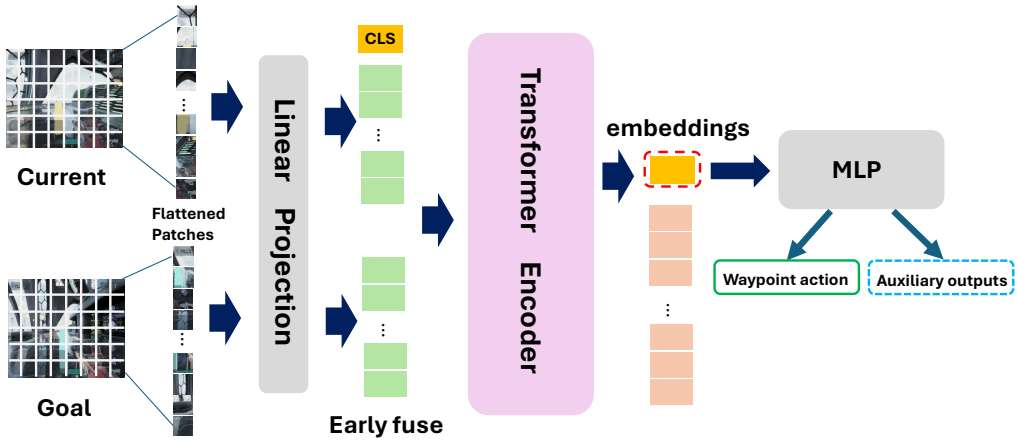


Figure 2: IGNavFM Network Architecture.

We also propose to leverage pretrained ViT encoders that preserves the strengths of powerful self-supervised models including DINOv2 [Oquab et al., 2023] and Masked Auto Encoding (MAE) [He et al., 2022] while enabling effective joint processing of observation and goal images. In our experiments, we validate that the self-supervised pretrained ViT encoders trained by MAE most significantly enhance the navigation performance. We highlight that the pretrained vision encoders trained by self-supervised learning can help extracting low-level features of images, and subsequently benefit the training of navigation foundation models.

3.2 Training Tasks

IGNavFM features leveraging auxiliary tasks that capture global decision making information to improve navigation performance. These global decision making information serves as high-level sub-goals (or macro-actions) that provide useful decision making signals for low-level navigation actions. This approach of predicting macro-actions is also employed by recent works [Bachmann and Nagarajan, 2024, Hu et al., 2024] as an improvement from predicting only the next actions.

Specifically, we use three types auxiliary tasks: global path to the goal position, distance to the goal position, and relative pose to the goal position. We detail these task as well as waypoint action prediction tasks as follows.

Waypoint Action Prediction. We define the provided navigation trajectory

$$\tau = (o_0, o_1, \cdots, o_T; p_0, p_1, \cdots, p_T)$$

with total T steps, where the first observation $o_{cur} = o_0$ is the current image and the last observation is the goal image $o_{goal} = o_T$. For each observation o_k, it is a RGB image which shows the field of view within 90 degrees directly in front of the agent. Each pose p_k consists of three elements (x, y, ψ), where the first two are coordinates x, y that measure the current location and the last one is yaw ψ that measures the current direction. The target of waypoint action prediction is the next $N_{\text{waypoint}} = 10$ consecutive steps in the provided navigation trajectory, where the model predict the target by using the current image observation o_{cur} and goal image observation o_{goal}. The waypoint action loss is defined as:

$$L^{\text{waypoint}}(\tau) = \sum_{k=1}^{N_{\text{waypoint}}} D^{\text{pos_yaw}}(f_\theta^{\text{waypoint}}(o_{cur}, o_{goal})_k, p_{0\to k})$$

Here the waypoint function $f_\theta^{\text{waypoint}}(o_{cur}, o_{goal})_k$ denotes the predicted waypoint action generated by IGNavFM, and $p_{0\to k}$ denotes the pose of $t = k$ under the coordinates of $t = 0$.

The position-yaw L_2 measurement

$$D^{\text{pos_yaw}}(p_1, p_2) = \|(x_2 - x_1, y_2 - y_1, \cos(\psi_2) - \cos(\psi_1), \sin(\psi_2) - \sin(\psi_1))\|_2^2$$

evaluates how the predicted positions and yaws are close to the ground truth positions and yaws.

Relative Pose to Goal. For relative pose prediction task, the model estimates the pose of the goal image with respect to the observation image. This task helps the model grasp the global task information as the spatial relations of the observation and the goal image.

The following loss, which evaluates how the predicted relative positions and rotations are close to the ground truth relative positions and rotations, is used for relative position prediction:

$$L^{\text{relative}}(o_{cur}, o_{goal}; p_0, p_T) = D^{\text{pos_yaw}}(f_\theta^{\text{relative}}(o_{cur}, o_{goal}), p_{0\to T})$$

Navigation Distance Prediction. For the navigation distance prediction task, our model is trained to predict the total distance that takes the agent to navigate from the current state to goal state. When the loss is optimized, our model learns to estimate the connectivity and traversibility between different images of the environment.

We define the navigation distance between current position and goal position as follows:

$$\text{nav_distance}(\tau) = \sum_{k=0}^{T-1} \|(x_{k+1} - x_k, y_{k+1} - y_k)\|_2^2$$

Then the navigation distance loss function is defined as:

$$L^{\text{nav_distance}}(\tau) = (f_\theta^{\text{nav_distance}}(o_{cur}, o_{goal}) - \text{nav_distance}(\tau))^2$$

Global Path Prediction. To further increase agent's prediction on long position prediction, we also propose global path prediction task. We predict $N_{\text{global}} = 10$ intermediate points which are equally spaced in time from current time to the total path length T and compare them to the output of IGNavFM outputs. The loss term can be defined as:

$$L^{\text{global}}(\tau) = \sum_{k=1}^{N_{\text{global}}} D^{\text{pos_yaw}}(f_\theta^{\text{global}}(o_{cur}, o_{goal})_k, p_{0\to\lfloor \frac{k\times T}{N_{\text{global}}} \rfloor})$$

In details, this loss term measures predicted future step performance based on uniform sampled k points in the prediction path. The overall training loss is the weighted sum of the above losses.

4 Experiments

We first introduce the environment we use for evaluating our model and present the results and analysis. We conduct the experiments to answer the following research questions:

- How does IGNavFM compare to other navigation foundation models regarding to the zero-shot generalization performance and finetuning performance in new environments?
- How does the pretraining of IGNavFM benefit the data efficiency of finetuning in new environment?
- How does early fuse of the low-level image features of observation and goal affect the IGNavFM performance?
- How does initialization from pretrained visual encoder by self-supervised learning (MAE, DINO-v2) affect the IGNavFM performance?
- How does the auxiliary tasks for predicting global decision making information impact the performance of IGNavFM?

4.1 Experiment Setups

We conducted evaluation experiments on three environments including two simulation environments (Highrise and Sanctuary) from a shooter game and one real robot environment. For simulation environments, we tested IGNavFM by deploying the models in these two game environments with three difficulties directly, which are consists of easy, medium and hard tasks with less and less information support.

Environment. Highrise and Sanctuary are two environments from ShooterGame, which is a quintessential representation of a PC multiplayer First-Person Shooter (FPS) Game by Unreal Engine 4, providing a robust framework. The ShooterGame contains Sanctuary and Highrise environment, both features a very large space of approximately $10000m^2$ area. The two environments also feature a large number dynamic assets and backgrounds, with a large number of obstacles that can be challenging for the navigation policy. For real world environment, we use a wheeled robot for traversing an indoor floor of about $2000m^2$ area.

Fine-tune Dataset. We collect 2100 navigation episodes on Highrise and Sanctuary by human player, with an average episode length of 34 and 47 on Highrise and Sanctuary respectively, and about 0.7 and 0.9 total number of frames in Highrise and Sanctuary respectively. For real robot dataset, we collect 113 episodes with an average of 100 episode length and a total 10K frames. In all of the dataset, we let the human player or human teleoperator to traverse the whole environment, and trim the straightforward trajectory from start to goal afterwards.

Task Specification. In all of Highrise, Sactuary and real robot environment, we split our collected dataset into 90% of fine-tuning dataset and 10% of the validation dataset. We randomly sample start and goal points from evaluation on the corresponding environment. Additionally, we split the evaluation into three difficulty levels: Easy, Medium and Hard, which are determined by the fractions of length between start point and goal point in the validation dataset. On Highrise, the Easy, Medium and Hard difficulty has an average of $(19, 37, 55)$ timesteps between start and goal position, while on Sanctuary, the Easy, Medium and Hard difficulty has an average of $(28, 56, 83)$ timesteps between start and goal position. We sample 50 tasks for each of the difficulty level. On real robot environment, we don't separate the difficulty level. Constrained by the hardware, we only evaluate the waypoint action prediction loss on validation dataset in this work for the real robot dataset.

Evaluation Measurement. For the evaluation of IGNavFM, we use two common metrics, success rate (SR) and success weighted by path length (SPL), to measure its performance of navigation tasks. In details, SR measures whether the navigation task can be completed without mistakes, and SPL measures the efficiency of navigation. We define

$$SR = \frac{1}{N}\sum_{i=1}^{N} S_i, \quad SPL = \frac{1}{N}\sum_{i=1}^{N} S_i \frac{d_i}{p_i}$$

where S_i equals 1 if navigation is successful in i-th evaluation episode and 0 otherwise, N is the total number of evaluation episodes, d_i is the shortest distance from current to goal position, and p_i is the navigation path length done by agent in evaluation.

Experiment Parameters. Our model employs ViT-Base as the image encoder and integrates multiple MLP heads, resulting in approximately 100 million parameters. We train the model using the Adam optimizer with a learning rate of 5×10^{-5} and a batch size of 128. Pretrained models are trained for 1,000 epochs, requiring approximately five days on a 4x A100 GPUs setup. Fine-tuned models are trained for 200 epochs, taking about two days on a single A100 GPU.

4.2 Results & Analysis

4.2.1 Comparison with Baselines for Zero-Shot Generalization and Finetuning

In this experiment, we compare our model with other two baselines including GNM and ViNT in both tasks as regarding generalizability to unseen situations and performance after fine-tuning in diverse environments. Results are demonstrated in Table 1. Our method IGNavFM outperforms in both zero-shot and fine-tune tasks for all three tasks compared to GNM and ViNT, in benefits from the design of early fuse structure and the desgin of auxiliary loss.

	Highrise						Sanctuary					
	Easy		Medium		Hard		Easy		Medium		Hard	
	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL
GNM (ZS)	0.86	0.78	0.58	0.47	0.32	0.20	0.33	0.27	0.11	0.05	0.00	0.00
ViNT (ZS)	0.82	0.78	0.46	0.43	0.26	0.22	0.41	0.37	0.36	0.32	0.18	0.17
IGNavFM (ZS)	**0.90**	**0.86**	**0.72**	**0.68**	**0.46**	**0.38**	**0.84**	**0.80**	**0.48**	**0.45**	**0.30**	**0.26**
GNM (FT)	0.88	0.86	0.42	0.41	0.28	0.27	0.61	0.55	0.25	0.21	0.04	0.02
ViNT (FT)	0.88	0.83	0.66	0.56	0.50	0.42	0.32	0.28	0.14	0.08	0.08	0.03
IGNavFM (FT)	**1.00**	**0.97**	**1.00**	**0.93**	**0.90**	**0.81**	**1.00**	**0.96**	**0.84**	**0.75**	**0.76**	**0.68**

Table 1: SR and SPL Performance across two environments and three difficulty levels compared with baseline methods (ViNT, GNM). Experiments run for zero-shot setting (first three rows) and fine-tune setting (last three rows). Values are rounded to the nearest hundredth.

4.2.2 Fine-Tune Efficiency of IGNavFM on New Environment

In this experiment, we evaluate the fine-tune efficiency of pretrained IGNavFM on Highrise, Sanctuary and real world environment. We reduce the amount of finetune data to several levels including $1, 1/2, 1/4, 1/8$ and $1/16$ and test the model performance in these corresponding settings on the average of three difficulty levels. As illustrated in Figure 3, it is obvious that as the amount of data decreases, all three models performance decrease. However, IGNavFM with pretraining phase shows the most robust performance even with significatly less finetuning, reducing data requirements by approximately eightfold, demonstrating its utility as a foundation model. It is also worth mentioning that the zero-shot generalization performance of IGNavFM outperform the $1/16$ and $1/8$ finetuning setting without pretraining, which highlights the IGNavFM's zero-shot generalization performance.

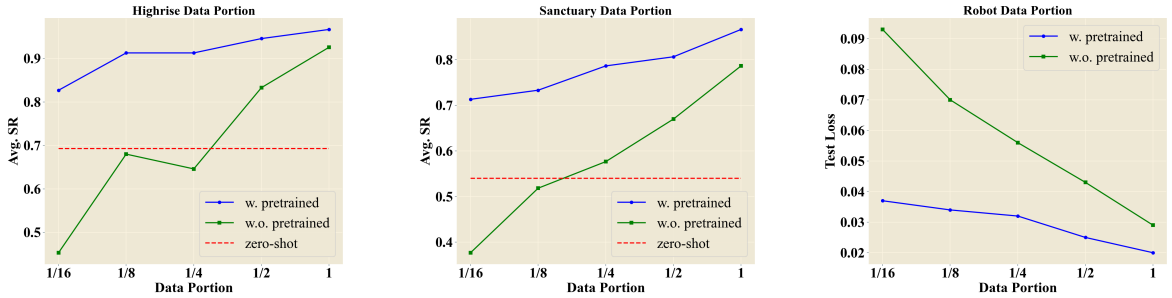
4.2.3 Early Fuse VS. Non-Early Fuse

To prove the effectiveness of our proposed early fuse structure, we experiment IGNavFM with two versions including early fuse and Non-early fuse, combined with MAE-pretrained ViT image encoder, in both Highrise and Sanctuary game environments. In the non-early fuse architecture, the observation and goal images are separately fed into the MAE-pretrained ViT encoder to obtain the image representation, and then the difference of their representation is subsequently fed into separate MLP layers as the contextual embeddings to generate waypoint action outputs and auxiliary outputs.

Table 2 shows the difference of model performance in these two structures. The results show that early fuse network architecture achieves better navigation performance on both zero-shot and fine-tuning settings, suggesting that early-fusion of low-level features for observation and goal images improve the performance for image-goal navigation.

	Highrise						Sanctuary					
	Easy		Medium		Hard		Easy		Medium		Hard	
	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL
Non-Fuse (ZS)	0.74	0.69	0.48	0.45	0.24	0.19	0.56	0.44	0.20	0.15	0.08	0.06
Early Fuse (ZS)	**0.90**	**0.86**	**0.72**	**0.68**	**0.46**	**0.38**	**0.84**	**0.80**	**0.48**	**0.45**	**0.30**	**0.26**
Non-Fuse (FT)	**1.00**	0.96	0.94	0.88	0.84	0.74	**1.00**	0.94	**0.84**	**0.75**	0.72	0.62
Early Fuse (FT)	**1.00**	**0.97**	**1.00**	**0.93**	**0.90**	**0.81**	**1.00**	**0.96**	**0.84**	**0.75**	**0.76**	**0.68**

Table 2: SR and SPL performance across two environments and three difficulty levels with different model structure. Experiments run for zero-shot setting (first two rows) and fine-tune setting (last two rows). Two types of Model structure include non-early fuse and early fuse. Values are rounded to the nearest hundredth.



(a) (b) (c)

Figure 3: SR and Validation Loss performance across different model training and finetune dataset size. Experiments run for zero-shot, with-pretrained and without-pretrained settings. (a) for Highrise, (b) for Sanctuary and (c) for real robot. Average SR is used for (a) (b) measurement, with higher value meaning better model performance. Test Loss is used for (c) measurement, with lower value meaning better model performance.

4.2.4 Integration of Pretrained Visual Encoders

We hypothesize that though DINOv2 encoder structure provides with high level semantic representations, MAE focus more on fine-grained details so that it is more helpful in image-goal navigation task. To find the better image encoder structure used for NavFM, we also experiment the effect of different ViT encoders, and the results show in Table 3.

Experiments show that the DINOv2 encoder situates in the high level area to measure the correspondence and lacks of the ability to capture the possible change or action information, while the MAE encoder can extract the low level embeddings relationship better and contains more low level details among.

	Highrise						Sanctuary					
	Easy		Medium		Hard		Easy		Medium		Hard	
	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL
ViT-raw	0.76	0.73	0.26	0.25	0.12	0.11	0.46	0.43	0.14	0.13	0.08	0.07
ViT-DINOv2	0.82	0.81	0.70	0.67	0.28	0.25	0.70	0.67	0.32	0.30	0.22	0.21
ViT-MAE	**0.90**	**0.86**	**0.72**	**0.68**	**0.46**	**0.38**	**0.84**	**0.80**	**0.48**	**0.45**	**0.30**	**0.26**

Table 3: SR and SPL performance across two environments and three difficulty levels with different pretrained model encoder. Experiments run for zero-shot setting. Three model encoders including ViT-raw, ViT-DINOv2 and ViT-MAE apply in experiments. Round to the nearest hundredth.

4.2.5 Effect of Auxiliary Loss

We also investigate the effect of each auxiliary loss on the performance of IGNavFM. The experiments are divided into five different settings. The first one only acquires the waypoint information as normal navigation tasks. In the following three experiments, we remove the goal, distance and global information separately to show these part effects in auxiliary loss. We show the auxiliary losses are all useful to achieve the succss rate of IGNavFM based on the outcome in Table 4. This experiment also suggests that predicting global decision making information are beneficial to the performance of navigation foundation models.

	Highrise						Sanctuary					
	Easy		Medium		Hard		Easy		Medium		Hard	
	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL	SR	SPL
Waypoint Only	0.82	0.77	0.70	0.64	0.32	0.28	0.78	0.72	0.42	0.38	0.14	0.13
No Goal	0.82	0.79	0.66	0.63	0.40	0.35	0.78	0.76	0.38	0.35	0.16	0.14
No Distance	0.86	0.84	0.72	**0.69**	0.38	0.34	0.82	0.79	0.46	0.43	0.18	0.16
No Global	0.88	0.84	**0.74**	0.68	0.42	0.34	0.80	0.76	0.46	0.42	0.18	0.17
All	**0.90**	**0.86**	0.72	0.68	**0.46**	**0.38**	**0.84**	**0.80**	**0.48**	**0.45**	**0.30**	**0.26**

Table 4: SR and SPL performance across two environments and three difficulty levels with different auxiliary loss design. Experiments run for zero-shot setting. Five types of design includes waypoint only, without goal, without distance, without global and all. Zero-shot are experimented. Values are rounded to the nearest hundredth.

5 Conclusions

In this paper, we introduce IGNavFM, a novel image-goal navigation foundation model that advances state-of-the-art navigation models through innovations in architecture and loss design. Our proposed early fusion network, which integrates pretrained ViT encoders, enhances both efficiency and performance. Additionally, we explore auxiliary loss functions to optimize pretraining and fine-tuning. Extensive experiments in both simulated game environments and real-world robotic settings demonstrate the effectiveness of our approach. These results underscore IGNavFM's potential to enhance autonomous navigation, paving the way for future advancements in robotic perception and decision-making.

For future work, an important direction is to assess IGNavFM's generalization across more diverse and complex environments, including dynamic and partially observable settings. Another promising avenue is self-supervised learning to reduce reliance on large-scale annotated datasets. Additionally, integrating IGNavFM with diverse real-world or game video datasets from various online platforms may further improve its capabilities.

References

Qiaoyun Wu, Jun Wang, Jing Liang, Xiaoxi Gong, and Dinesh Manocha. Image-goal navigation in complex environments via modular learning. *IEEE Robotics and Automation Letters*, 7(3):6902–6909, 2022.

Nuri Kim, Obin Kwon, Hwiyeon Yoo, Yunho Choi, Jeongho Park, and Songhwai Oh. Topological semantic graph memory for image-goal navigation. In *Conference on Robot Learning*, pages 393–402. PMLR, 2023.

Devendra Singh Chaplot, Ruslan Salakhutdinov, Abhinav Gupta, and Saurabh Gupta. Neural topological slam for visual navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12875–12884, 2020.

Hakan Temeltas and Demiz Kayak. Slam for robot navigation. *IEEE Aerospace and Electronic Systems Magazine*, 23 (12):16–19, 2008.

Michael Milford and Gordon Wyeth. Persistent navigation and mapping using a biologically inspired slam system. *The International Journal of Robotics Research*, 29(9):1131–1153, 2010.

Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 3357–3364. IEEE, 2017.

Kenzo Lobos-Tsunekawa, Francisco Leiva, and Javier Ruiz-del Solar. Visual navigation for biped humanoid robots using deep reinforcement learning. *IEEE Robotics and Automation Letters*, 3(4):3247–3254, 2018.

Karmesh Yadav, Arjun Majumdar, Ram Ramrakhya, Naoki Yokoyama, Alexei Baevski, Zsolt Kira, Oleksandr Maksymets, and Dhruv Batra. Ovrl-v2: A simple state-of-art baseline for imagenav and objectnav. *arXiv preprint arXiv:2303.07798*, 2023.

Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. Gnm: A general navigation model to drive any robot. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7226–7233. IEEE, 2023a.

Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. Vint: A foundation model for visual navigation. *arXiv preprint arXiv:2306.14846*, 2023b.

Ajay Sridhar, Dhruv Shah, Catherine Glossop, and Sergey Levine. Nomad: Goal masked diffusion policies for navigation and exploration. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 63–70. IEEE, 2024.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *arXiv preprint arXiv:2402.15852*, 2024a.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024b.

Xinyu Sun, Peihao Chen, Jugang Fan, Jian Chen, Thomas Li, and Mingkui Tan. Fgprompt: fine-grained goal prompting for image-goal navigation. *Advances in Neural Information Processing Systems*, 36, 2024.

Obin Kwon, Jeongho Park, and Songhwai Oh. Renderable neural radiance map for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9099–9108, 2023.

Yang Liu, Weixing Chen, Yongjie Bai, Xiaodan Liang, Guanbin Li, Wen Gao, and Liang Lin. Aligning Cyber Space with Physical World: A Comprehensive Survey on Embodied AI, August 2024.

Yue Zhang, Ziqiao Ma, Jialu Li, Yanyuan Qiao, Zun Wang, Joyce Chai, Qi Wu, Mohit Bansal, and Parisa Kordjamshidi. Vision-and-Language Navigation Today and Tomorrow: A Survey in the Era of Foundation Models, December 2024c.

Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. NavGPT-2: Unleashing Navigational Reasoning Capability for Large Vision-Language Models, September 2024.

Jiaqi Chen, Bingqian Lin, Xinmin Liu, Lin Ma, Xiaodan Liang, and Kwan-Yee K Wong. Affordances-oriented planning using foundation models for continuous vision-language navigation. *arXiv preprint arXiv:2407.05890*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0 : A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

Gregor Bachmann and Vaishnavh Nagarajan. The pitfalls of next-token prediction. *arXiv preprint arXiv:2403.06963*, 2024.

Edward S Hu, Kwangjun Ahn, Qinghua Liu, Haoran Xu, Manan Tomar, Ada Langford, Dinesh Jayaraman, Alex Lamb, and John Langford. Learning to achieve goals with belief state transformers. *arXiv preprint arXiv:2410.23506*, 2024.