目 次

第1章　序論

1.1　研究背景および目的

近年, 日本をはじめとする先進国では少子高齢化や人口減少に伴う労働力不足が深刻な社会問題となっている. 特に製造業や物流業といった労働集約型の産業では, 人手不足による生産効率の低下やサービス品質の維持が課題として挙げられる. このような状況を打開するための方策として, ロボットを活用した自動化や省人化が注目を集めている. ロボットの導入により, 単純作業の効率化が可能になるだけでなく, 人間の代わりに危険な作業環境へ投入することで安全性の向上も期待されている.

一方で, ロボットを効果的に活用するには, ロボット自身が周囲の環境を適切に認識し, 自律的に移動できるナビゲーション技術の確立が不可欠である. ロボットナビゲーションの研究は, 初期にはセンサーやビーコンを利用した手法から始まり, 現在では光学計測を用いた LiDAR（Light Detection And Ranging）を活用することで, 周囲の障害物や地形の三次元計測が可能となっている.

LiDAR は高解像度で環境情報を取得できる利点があるが, 幾何学的情報のみを扱うため, 例えば草などの実際には通行可能な物体を障害物として誤認識する課題がある. この課題を克服するため, 近年では深層学習を活用したナビゲーション手法の研究が盛んに行われている. 特に, 外部環境の画像とゴール画像を入力し, 強化学習や模倣学習によってロボットの行動を学習する「ビジュアルナビゲーション」が注目されている.

このビジュアルナビゲーションにおいて, 革新的な手法として注目されているのが NoMaD（Navigation with Goal Masked Diffusion）である [1]. ナビゲーションタスクは, 「ゴール条件付きナビゲーション」と「未知環境の探索」に大別されるが, 従来の手法ではこれらを別々のモデルで処理していた. NoMaD はこれらを統合し, 1 つのモデルで両方のタスクを学習することで, モデルのサイズを削減しつつ, 高精度なナビゲーションと探索を実現している. この手法は高く評価され, ICRA 2024 において Best Paper Award を受賞した [2].

しかし, NoMaD にはいくつかの課題が残されている. 特に, NoMaD はゴールを画像でしか指定できないため, より柔軟なナビゲーションシステムの構築が求められている. 例えば, 音声やテキストによるゴール指定が可能になれば, ユーザーの意図をより直感的に反映でき, 多様な応用が可能となる. また, 環境の変化に対する適応力の向上も課題の一つであり, 特に動的な環境下での適用を考えると, 追加のセンサー情報との統合や学習アルゴリズムの改良が必要となる.

そこで本研究では, NoMaD のモダリティを拡張し, 音声や映像によってゴールを指定できるナビゲーションシステムの開発を目的とする. これにより, 従来の画像ベースの制約を克服し, より柔軟で直感的なナビゲーションを可能にすることを目指す.

1.2 論文構成

本論文の構成について述べる. 本論文では, 第 1 章に研究背景と本論文の構成, 第 2 章に本研究で必要な事前知識, 第 3 章に関連研究, 第 4 章に本研究で開発したシステムの概要, 第 5 章に実験内容と結果について述べる. 最後に, 第 6 章にて結論を述べる.

第2章　準備

本章では, 本研究において必要な事前知識に関して説明する.

2.1　NoMaD

2.1.1　モデル概要

NoMaD (Navigation with Goal Masked Diffusion) は, 2023 年 10 月にカリフォルニア大学の Ajay Sridhar らによって提案された, 視覚情報を用いたナビゲーションのための深層学習モデルである. 本手法の大きな特徴は, ゴールへのナビゲーションと未知環境における探索という二つのタスクを, 単一のモデルで統合的に処理できる点にある.

一方, 従来手法である ViNT (Visual Navigation Transformer) [3] では, 未知環境の探索時に以下の二段階の処理が必要であった.

1. サブゴール画像の生成: 画像生成モデルを用いて, 次の目的地（サブゴール）に相当する画像を生成する.
2. 経路計画: 生成されたサブゴール画像を入力として, ナビゲーションモデルが経路を決定する.

これに対して, NoMaD は観測画像のみから直接次の経路を生成するため, サブゴール画像の生成プロセスを省略できる. このアプローチにより, NoMaD のパラメータ数は ViNT に比べて約 15 分の 1 に削減され, NVIDIA Jetson Orin などのエッジデバイス上での実行が可能となった.

2.1.2 ナビゲーション時の推論フロー

NoMaD のモデル構造を図 2.1 に示す.

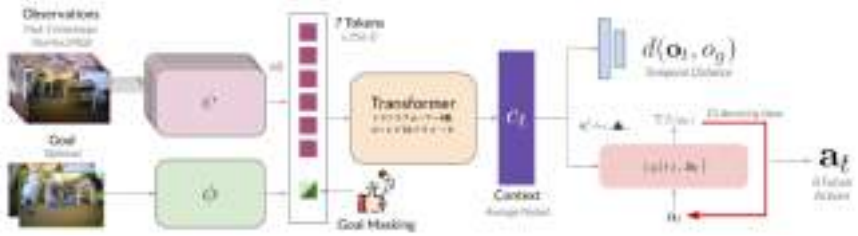

図 2.1: NoMaD のモデル構造（出典: [1]）

NoMaD は，時刻 t における 4 枚の観測画像 $\{o_{t-3}, o_{t-2}, o_{t-1}, o_t\}$ とゴール画像 o_g を入力とし，時刻 t における行動ベクトル

$$a_t = [v_t, \omega_t] \tag{2.1}$$

（v_t は並進速度，ω_t は角速度）を推定する. 本節では, ナビゲーション時の推論手順を概説する.

画像のエンコード

まず，4 枚の観測画像とゴール画像を EfficientNet-B0 エンコーダ（ψ と ϕ）および Transformer の Decoder（f）によって統合し，256 次元の観測ベクトル c_t にエンコードする（式 2.2）.

$$c_t = f(\psi(o_{t-3}), \psi(o_{t-2}), \psi(o_{t-1}), \psi(o_t), \phi(o_t, o_g)). \tag{2.2}$$

ここで, ψ はチャンネル数 3 の EfficientNet-B0 エンコーダ, ϕ はチャンネル数 6 の EfficientNet-B0 エンコーダ, f は Transformer の Decoder を指す.

このとき, ゴール画像のトークンを masking することで, ゴール条件付きのナビゲーションから, 未知環境における探索に切り替えることができる.

Diffusion Policy による行動系列の推論

つぎに，観測ベクトル c_t を Diffusion Policy に入力し, 将来の行動系列 $\{a_t, a_{t+1}, a_{t+2}, \ldots, a_{t+k}\}$ をサンプリングする.

Diffusion Policy[4] は，2023 年に Cheng Chi らにより提案された拡散過程に基づく方策学習手法であり, 外乱や不確実性を考慮した確率的な行動生成が可能である. 具体的には, 拡散過程を逆にたどるかたちでノイズ除去を行い，最終的にロボットの行動を得る.

NoMaD では, Transformer によって生成された観測ベクトル c_t を入力することで, 障害物の回避も含めたロバストな行動を学習している.

観測情報 c_t を条件とした時刻 t におけるロボットの行動 a_t は, 式 2.3 を ${a_t}^K, {a_t}^{K-1}, ..., {a_t}^0 = a_t$ まで K 回のノイズ除去を繰り返すことで, 外乱を考慮しながら得ることができる. ここで, α はスケーリング係数, γ はノイズ予測の調整パラメータである. また, ϵ_θ はノイズ予測ネットワークであり, 外乱 $\mathcal{N}(0, \sigma^2 I)$ を予測している.

$$ {a_t}^{k-1} = \alpha({a_t}^k - \gamma\epsilon_\theta(c_t, {a_t}^k, k) + \mathcal{N}(0, \sigma^2 I)) \tag{2.3} $$

2.1.3 ナビゲーション手法

NoMaD では，ゴール画像を条件とし，次に進むべき行動ベクトル a_t を逐次出力することで短距離のビジュアルナビゲーションを実現する. しかし，ゴール地点が遠距離にある場合には，一度に正確な経路を推論することは難しい.

そこで，本研究では事前にロボットを手動で操作してスタート地点からゴール地点までの観測画像を取得し, トポロジカルマップを構築したうえで，その間の画像を複数の「サブゴール画像」として順に辿ることで長距離のナビゲーションを可能としている.

Ajay Sridhar らは，2 地点間の時間的距離を画像から推定できる深層学習モデル（以降,「距離推定モデル」とする）を用いてマップ上での現在地点に最も近いノードを推定する. 具体的には，観測画像群 $\{o_{t-3}, \ldots, o_t\}$ とゴール画像 o_g を入力して得られるベクトル c_t を距離推定モデル *DistPredNet* に入力し，2 地点間の距離 $d(o_t, o_g)$ を推定する（式 2.4）.

$$ d(o_t, o_g) = DistPredNet(c_t) \tag{2.4} $$

この推定距離を用いることで，現在のサブゴールを更新しながらゴールまでの経路を移動する. ナビゲーションのアルゴリズムは以下の通りである.

Algorithm 1 NoMaD によるナビゲーション

Require: $sub\_goal\_images = [sg_0, sg_1, ..., sg_N]$ ▷ 一連のサブゴール画像

1: $current\_node \leftarrow 0$
2: **while** $current\_node \neq \text{len}(sub\_goal\_images) - 1$ **do**
3: $obs\_images \leftarrow [o_{t-3}, o_{t-2}, o_{t-1}, o_t]$ ▷ 直近 4 枚の観測画像
4: $waypoint \leftarrow \text{NoMaD}(obs\_images, sub\_goal\_images[current\_node + 1])$
5: *waypoint* に従って移動
6: $distances \leftarrow \text{d}(obs\_images, sub\_goal\_images)$
7: $current\_node \leftarrow argmin(distances)$
8: **end while**

2.2 ROS2

2.2.1 概要

ROS（Robot Operating System）[5] はロボットアプリケーションを構築するためのオープンソースの SDK（ソフトウェア開発キット）である. 開発者は, ROS が提供する豊富なツールやライブラリを活用することで, 複雑なロボットアプリケーションを効率的に開発することができる.

当初, ROS は ROS 1 として開発されたが, 普及が進むにつれて, 以下のような当初想定されていなかったユースケースへの対応が求められるようになった.

- マルチロボットシステム（複数のロボット間の通信が必要）
- リソース制約のある組み込みシステム（低消費電力・低メモリ環境での動作）
- セキュリティ要件が高い環境（暗号化や認証の必要性）
- リアルタイム性が要求されるシステム（即時応答が可能な制御）

しかしながら, ROS 1 の通信プロトコルは独自実装であったため, これらのユースケースに適応するのが困難であった. この問題を解決するために, 通信プロトコルとして標準規格である DDS (Data Distribution Service) [6] を採用した ROS 2 [7] が開発された.

DDS は, 分散システムにおけるデータ通信のための国際標準規格 であり, 以下のような利点を持つ.

- 高い信頼性（QoS: Quality of Service を利用した通信制御）
- スケーラビリティ（複数のロボット間での効率的なデータ共有）
- リアルタイム性の向上（低遅延通信のサポート）
- セキュリティの強化（認証やデータ暗号化のサポート）

このように DDS を基盤とすることで, ROS 2 は ROS 1 では対応が難しかった環境にも適応可能となった.

2.2.2 メッセージ通信

ロボットには多くのセンサー, モーター, アクチュエーター が搭載されており, それらを異なる周期で制御する必要がある. 例えば, モーターの制御は数ミリ秒単位のリアルタイム処理が求められる一方, カメラ画像の処理は数十ミリ秒の遅延を許容できる. しかし, これらの異なる周期の制御を単一プロセスで実装すると, 処理負荷が高くなり, 柔軟性にも欠ける. そのため, ROS 2 では複数のプロセス（ノード）を用いて処理を分担し, それらを連携させるためにメッセージ通信を採用している.

ROS 2 のメッセージ通信は, パブリッシュ/サブスクライブ (Publish/Subscribe) 方式 に基づいている. この方式では, 各プロセスは 「ノード」(Node) として定義され, ノード間の通信は「トピック」(Topic) を介して行われる. ノードとトピックの関係は以下の通りである.

- パブリッシャー（Publisher）: トピックにデータを送信するノード
- サブスクライバー（Subscribe）: トピックからデータを受信するノード

図 2.2 に ROS 2 を用いた画像認識システムの例を示す. ここでは, /camera/image というトピックを介して, カメラノードが画像を送信し, 物体検出ノードが画像を受信する.

図 2.2: ROS2 による画像認識システム

このように, トピックを介してデータをやり取りすることで, 各ノードは互いの存在を意識することなく動作できる. これにより, システムの 柔軟性と拡張性が向上し, 新しい機能の追加や変更が容易になる.

第3章　関連研究

本章では, 本研究の関連研究である「LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action」について説明する.

3.1　LM-Nav

3.1.1　概要

本研究は, Dhruv Shah らによる先行研究「LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action」[8] に基づいている. 先行研究では, LLM（Large Language Model）, VLM（Visual Language Model）, および VNM（Visual Navigation Model）の３つの基盤モデルを統合することで, 言語指示に基づいたビジュアルナビゲーションが可能であることを示している. 具体的には, LLM には GPT-3, VLM には CLIP, VNM には ViNG を用いている. 以下では, LM-Nav を構成する各基盤モデルについて詳細に説明する.

GPT-3

GPT-3（Generative Pre-trained Transformer 3）[9] は, OpenAI によって 2020 年に発表された LLM（大規模言語モデル）である. 大量なインターネット上のテキストを用いた事前学習により, 言語の統計的性質や文脈依存性を獲得した. その結果, 汎用的な自然言語処理性能を有し, 翻訳や要約など, 特定のタスク向けにファインチューニングを行わなくても, 少数の例示（few-shot learning）や全く例示がない状況（zero-shot learning）で十分な性能を発揮する.

CLIP

CLIP（Contrastive Language-Image Pre-training）[10] は, 2021 年 2 月に OpenAI によって発表された, 言語と画像という異なるモダリティを統合的に扱うマルチモーダルモデルである. 従来の画像認識手法は, あらかじめ定義されたクラス（例：「犬」や「猫」など）に限定して分類を行っていたが, CLIP はテキストと画像のペアから成る大規模なデータセットを用い, 対照学習の枠組みに基づいて, 画像とテキストの間に共有される意味的表現空間を獲得する点において革新性を有する（図 3.1 参照）.

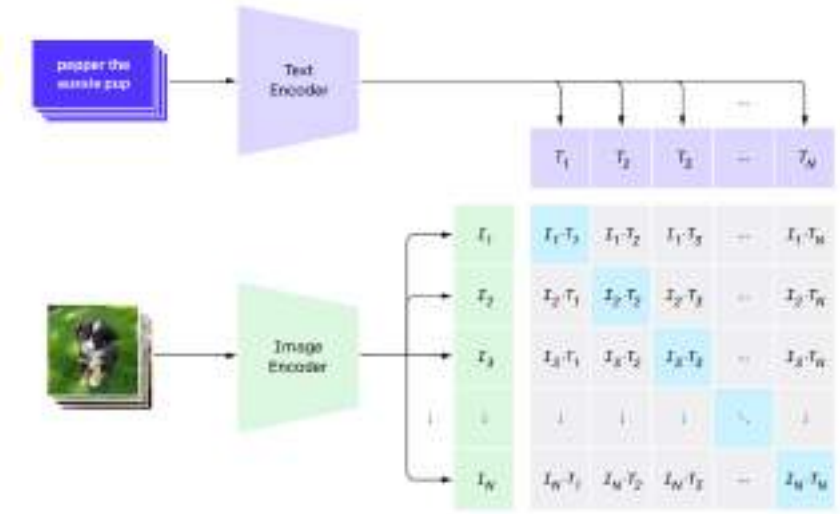

図 3.1: CLIP の事前学習（出典: [10]）

具体的には, CLIP は各画像に対して対応するテキスト記述をペアとして取り扱い, 対応する組み合わせ同士の距離を縮小し, 非対応の組み合わせ間の距離を拡大する学習戦略を採用している. このプロセスにより, 画像とテキストは同じ意味表現空間にマッピングされ, 自然言語で記述されたクラス情報を直接利用できるようになる. つまり, 学習時に明示的に定義されていなかった「走っている犬」や「魚をくわえた猫」といった新たなクラスに対しても, ゼロショット学習の枠組みを通じて高い分類性能が発揮できるのである.

さらにその汎用性の高さから, 画像検索, キャプション生成, さらには他のマルチモーダルタスクへの応用も期待される. 例えば, 画像検索においては, ユーザが入力する自然言語のクエリと画像の表現との類似性を計算することで, 関連性の高い画像を迅速に抽出することが可能となる. また, キャプション生成の分野では, 画像内容の意味的理解に基づいた

自然言語の出力が実現できるため, 従来手法では困難であった柔軟な記述が可能になると考えられる.

ViNG

ViNG（Learning Open-World Navigation with Visual Goals）[11] は, 2020 年 12 月に Dhruv Shah らによって提案された Visual Navigation Model（VNM）であり, 画像のみを用いて複雑なオープンワールド環境内を自律的にナビゲートすることを目的としている.

本モデルは, 目標地点の画像が与えられると, 現在位置からその地点へ向かうための移動経路を生成する能力を有する. しかし, 本モデルだけでは複雑な長距離経路を直接計画することが困難であるため, Dhruv Shah らはスタート地点からゴール地点までの経路上に複数の中間目標地点を設定し, それらを基にトポロジカルマップを構築した（図 3.2 参照）. さらに, 2 つの画像間の距離を推定可能な自己教師あり学習モデル（以下「距離推定モデル」とする）を導入することで, スタートからゴールまでの最短経路を求め, より効率的なナビゲーションを実現している.

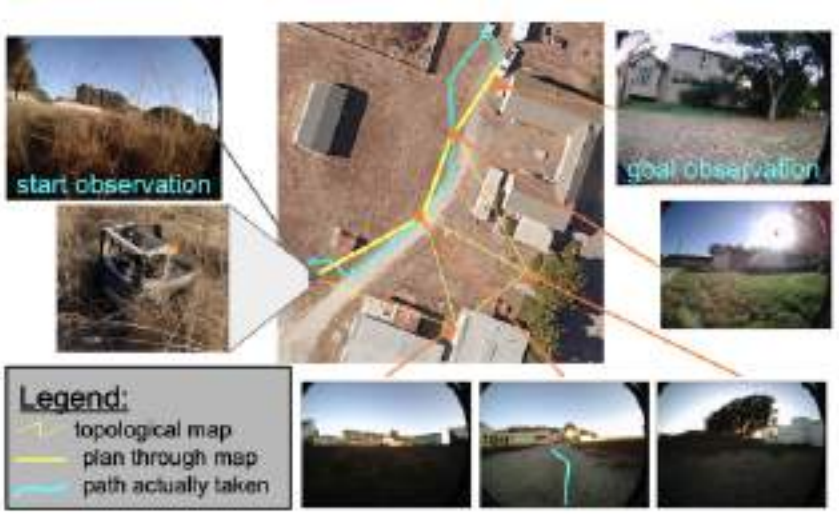

図 3.2: ViNG によるナビゲーション（出典: [11]）

3.1.2　処理フロー

LM-Nav は, 図 3.3 に示すように, 自由形式の文章から複数のランドマーク指示を抽出し, その順序に従ってビジュアルナビゲーションを実現するシステムである. LM-Nav の処理フローについて以下に説明する.

図 3.3: LM-Nav の概要（出典: [8]）

トポロジカルマップの構築

対象環境内を走行して一連の観測画像を取得し, ViNG の距離推定モデルを用いてこれらの画像からトポロジカルグラフを構築する.

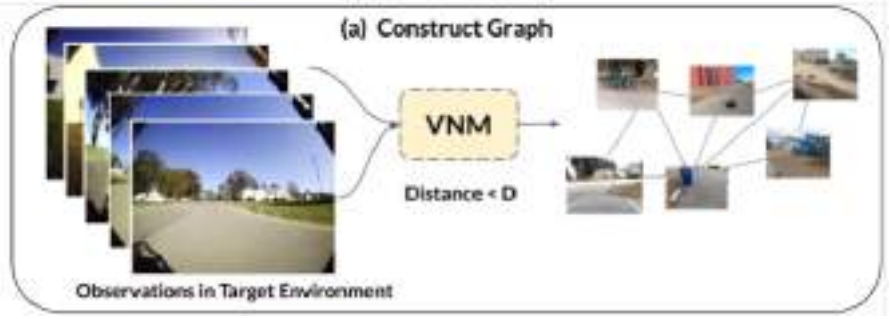

図 3.4: トポロジカルマップの構築（出典: [8]）

言語指示からランドマーク列の抽出

自由形式の文章から, 巡回すべきランドマークの順序を GPT-3 を利用して抽出する.

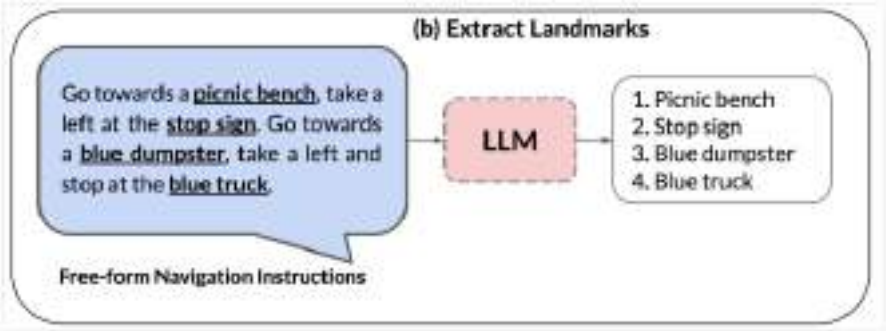

図 3.5: 言語指示からランドマーク列の抽出（出典: [8]）

各ノードとランドマークの関連度合いを算出

CLIP を用い, グラフ上の各ノードに対応する画像とランドマークのテキスト間の類似度を算出することで, 各ノードとランドマークの関連度を評価する.

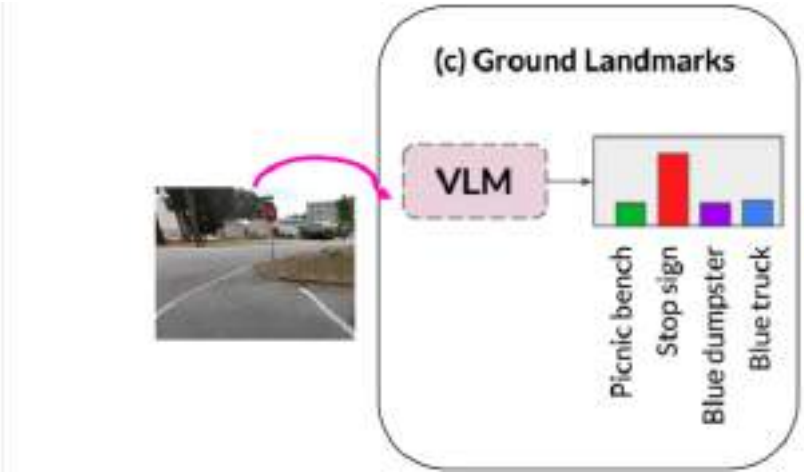

図 3.6: CLIP によるランドマーク関連度合いの算出（出典: [8]）

トポロジカルマップ上でのプランニング

手順 3 で算出した各ノードのランドマーク関連度とトポロジカルマップを組み合わせ, 指示されたランドマークの順序に沿いながらマップ上で最短の経路を, ダイクストラ法をベースとするアルゴリズムでプランニングする.

サブゴールに向かってナビゲーション

トポロジカルマップ上での現在位置から, 手順 4 で算出した経路に沿って次のノードに ViNG を用いて移動する.

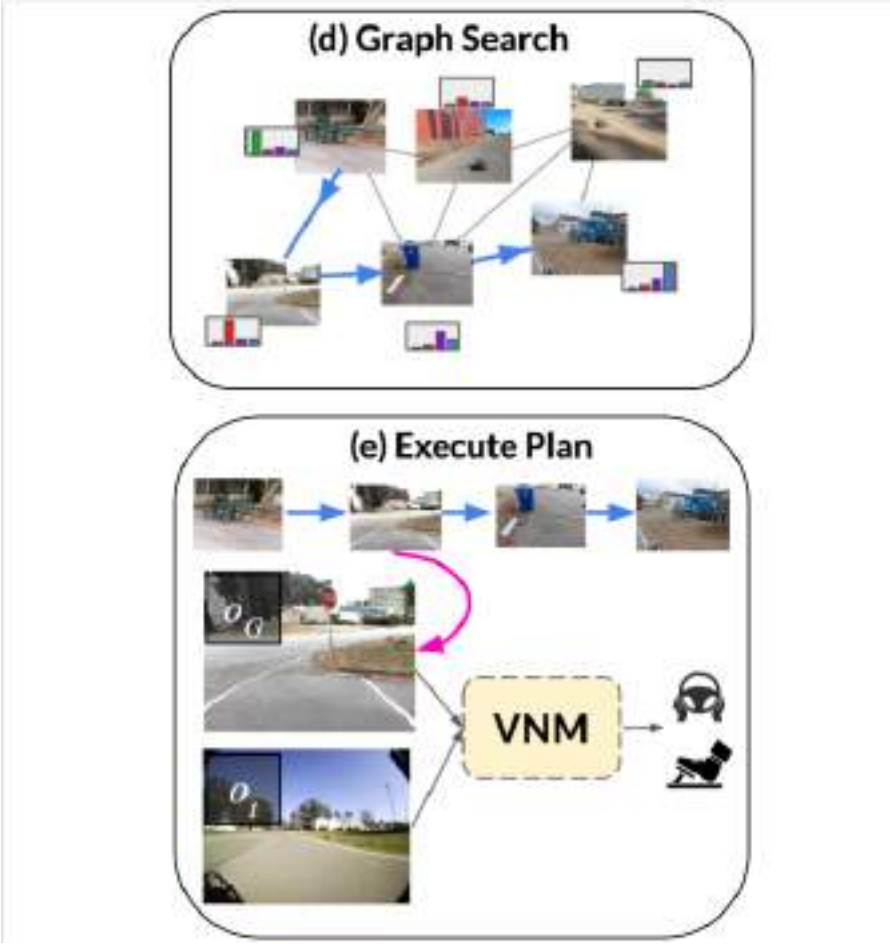

図 3.7: 最短経路の算出と ViNG による移動（出典: [8]）

第4章　提案手法

本章では, 本研究で開発したナビゲーションシステムについて説明する.

4.1　概要

NoMaD は, ICRA において Best Paper Award を受賞するなど, 画期的なナビゲーションモデルとして注目されている. しかし, 現行仕様にはいくつかの課題が存在する.

第一の課題は, ゴール地点の指定が画像のみで行われる点である. 画像による指定は一般的な方法ではあるが, 自然言語による指示が可能となれば, ユーザーの利用体験は大いに向上すると考えられる.

第二の課題は, 複雑な経路のナビゲーションに十分に対応できていない点である. Ajay Sridhar らの実験では, NoMaD を長距離ナビゲーションに対応させるためにトポロジカルマップの作成が試みられたが, 現行仕様では経路に分岐が存在するような複雑なマップに対応できず, 環境によっては目的地まで最短経路で到達できない場合がある.

本研究では, 上記の NoMaD の課題を解決することを目的として, NoMaD を用いた音声指示可能なナビゲーションシステムの開発に取り組んだ. 本システムは, LM-Nav から着想を得つつ, LLM, VLM, VNM の 3 つの基盤モデルを組み合わせることで構築された. 具体的には, LLM として Gemini 2.0 Flash Experimental, VLM として CLIP, VNM として NoMaD を採用している.

本システムには, 2024 年 12 月に Google が発表した Gemini 2.0 Flash Experimental[12]（以降、Gemini）を導入した. このモデルはテキスト・音声・画像を統合的に理解できるマルチモーダル LLM であり, ストリーミング音声や映像の入力に対応しており, リアルタイムで音声応答が可能である. 従来のナビゲーションシステムでは, 音声認識のみを用いることが一般的であったが, 本システムでは音声に加えてカメラ映像の情報も考慮することで, より正確な目的地の指示が可能となる.

例えば, ある商品の場所までのナビゲーションを指示する場面を考える. 音声認識のみのシステムでは, 目的の商品の特徴を言葉で説明する必要があり, ユーザーが詳細な情報を提供しなければならない. 一方で, 本システムではカメラ映像を活用することで, ユーザーが商品画像をカメラに映すだけで目的地を特定することができる. この仕組みにより, 従来の音声認識ベースのナビゲーションと比較して, より直感的かつ正確な案内が可能となる.

4.2 システム構成

本システムは ROS2 を用いて設計した. 図 4.1 は本システムにおける ROS2 のネットワーク図である.

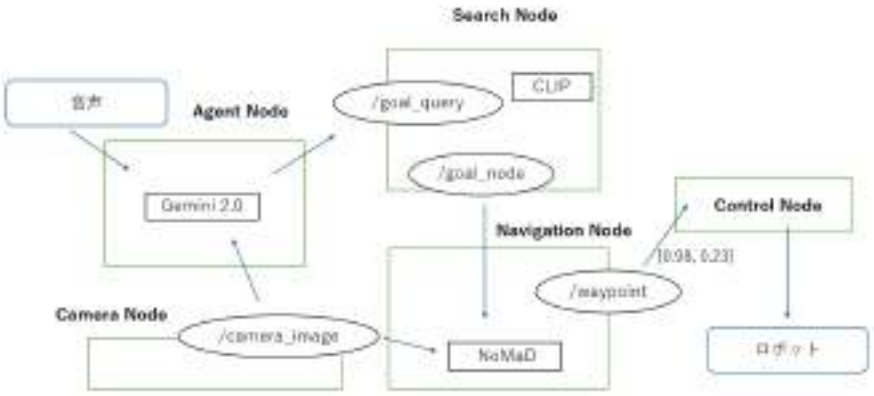

図 4.1: 開発したシステムの ROS2 ネットワーク図

システムの各機能は ROS2 におけるノードとして設計されており, トピックを通してデータを共有することで分散処理を行っている. 各トピックが扱うデータは以下の通りである.

- /goal_query(std_msgs/String): ユーザーが要求した目的地のテキスト.
- /goal_node(std_msgs/UInt32): ゴール地点のノード番号.
- /camera_image(sensor_msgs/Image): ロボットに搭載されたカメラの画像.
- /waypoint(std_msgs/Float32MultiArray): NoMaD が推論した進路 ([並進速度, 角速度]).

4.3 処理フロー

ナビゲーションを行う手順は以下の通りである.

1. トポロジカルマップの構築
2. ユーザーとの対話により目的地を決定
3. 目的地のノード番号を検索
4. 最短経路の算出
5. 目的地へのナビゲーション

各手順について詳しく説明する.

4.3.1 トポロジカルマップの構築

まず, 手動でロボットを操作し, ターゲット環境内の観測画像を取得する. そして, NoMaD の距離推定モデルを用い, 観測画像をノードとした重み付き有向グラフとしてトポロジカルマップを構築する.

4.3.2 ユーザーとの対話により目的地を決定

Agent Node において, Gemini がユーザーからナビゲーションの指示を受けると, 音声と画像情報から目的地を抽出し, Function Calling により/goal_query に目的地のテキストを送信する関数を呼び出す. このとき, Gemini のシステムプロンプトには, 「あなたは案内ロボットです。目的地を指示されたらその場所まで移動してください。また、人が手に画像を持っている場合、その画像に移っている物体をできるだけ詳細に説明し、その場所まで移動してください。」と設定した.

4.3.3 目的地のノード番号を検索

Search Node が/goal_query から目的地のクエリーを受け取ったら, CLIP により各ノードに対応する画像とクエリーの類似度を算出し, 最も類似度が高いノードをゴール地点のノードとする. そして, そのノード番号を/goal_node に送信する.

4.3.4　最短経路の算出

Navigation Node が/goal_node からゴール地点のノード番号を受け取ったら, 距離推定モデルによって, トポロジカルマップ上で最も現在地に近いノードを求め, そのノードからゴールノードまでの最短経路をダイクストラ法により算出する.

4.3.5　目的地へのナビゲーション

Navigation Node において, NoMaD により算出した最短経路上のサブゴールを現在のノードから順番に移動していく.

第5章　実験

本章では, ナビゲーションシステムの性能を検証するために行った実験について説明する.

5.1　実験環境

実験環境には, 以下のオブジェクトを配置した（各図参照）.

- Unitree Go2 の箱（図 5.1）
- 赤色のスーツケース（図 5.2）
- パソコンモニター（図 5.3）
- 段ボール製の筒（図 5.4）

図 5.1: Unitree Go2 の箱

図 5.2: 赤色のスーツケース

図 5.3: パソコンモニター

図 5.4: 段ボールの筒

本研究で用いたナビゲーションシステムは, Unitree Robotics 社が開発した 4 足歩行ロボット「Unitree Go2」[13]（「Unitree Go1」の後継機）に搭載された計算機上で実行した. Unitree Go2 は, 歩行性能およびバッテリー稼働時間が向上しており, 険しい段差が存在する環境や広範囲の自律走行に対応可能である. また, 背面には NVIDIA Jetson Orin Nano 搭載のドッキングステーションを装備することができる.

Unitree Go2 の主なスペックは以下の通りである.

- モデル名: Unitree Go2（R&D）
- サイズ（立脚時）: 70x31x40 cm
- 走行速度: 0 3.7 m/s
- 搭載カメラの解像度: 1280x720 ピクセル

ナビゲーションシステムの実行環境は以下に示す通りである.

- ハードウェア: NVIDIA Jetson Orin Nano（8 GB）
- OS: Ubuntu 20.04
- CPU: Arm® Cortex®-A78AE
- GPU: 1024-core NVIDIA Ampere architecture GPU with 32 tensor cores
- メモリー: 8GB 128-bit LPDDR5

なお, 本実験では, サンワサプライ製 Web カメラ（モデル：CMS-V41BK）に内蔵された音声マイクを使用した.

図 5.5: 実験時の Unitree Go2

5.2 実験方法

本研究では，ナビゲーションシステムの性能評価を目的として，下記の 2 種類の実験を実施した.

1. ナビゲーション性能の評価
2. ゴール地点の検索性能の評価

5.2.1 ナビゲーション性能の評価

本実験では, 対象環境内に配置された各オブジェクトに対するナビゲーションの成功率を測定するとともに, Diffusion Policy におけるノイズ除去プロセスの反復回数（10 回，20 回，30 回）がナビゲーション成功率に及ぼす影響を検証した.

トポロジカルマップの構築

まず，手動操作によりロボットを移動させ，対象環境のトポロジカルマップを構築した, この際, 環境内の画像は「Unitree Go」アプリケーションを用い, Unitree Go2 視点で動画を撮影しながら, ナビゲーション経路上を移動し, 撮影した動画から 1 fps の間隔でフレームを抽出することにより取得した.

ナビゲーションの実施

構築したトポロジカルマップ上において, 対象オブジェクト（1: 段ボールの筒, 2: Unitree Go2 の箱, 3: 赤色のスーツケース, 4: パソコンモニター）に対し，スタート地点（ノード 0）から各オブジェクトへのナビゲーションを各 10 回実施した. 本実験では，Diffusion Policy のノイズ除去反復回数を 10 回，20 回，30 回の各条件下で，ナビゲーションの成功率を測定した.

また, ナビゲーションの指示方法として, 以下の 2 種類の手法を実施した.

・音声指示: 各オブジェクトに対して, 以下のような音声指示を与えた.

1. 「段ボールの筒の前まで案内して」
2. 「Unitree Go2 の箱の前まで案内して」
3. 「赤色のスーツケースの前まで案内して」
4. 「パソコンモニターの前まで案内して」

・画像提示による指示: 対象オブジェクトの画像（図 5.6 参照）を提示しながら,「この場所まで案内してください」と指示した.

図 5.6: 画像提示の例

5.2.2 ゴール地点の検索性能の評価

本実験では,「ナビゲーション性能の評価」と同様に音声指示と画像提示による指示を行い, トポロジカルマップ上から正しいゴールノードを検索する成功率を測定した．各条件につき 10 回の試行を行い，得られた成功率を比較検討した.

5.3 評価方法

5.3.1 ナビゲーション性能の評価

本実験では, ゴール画像とゴール到達時の観測画像間の類似度を評価するため, SSIM（Structural Similarity Index Measure）を用いた. 具体的には, SSIM の値が 0.7 以上であればナビゲーション成功と判断する.

SSIM は画像間の類似性を評価する指標であり, 従来広く用いられている平均二乗誤差（MSE）やピーク信号対雑音比（PSNR）が画素ごとの差異のみを測定するのに対し, 以下の 3 つの要素に着目して, より人間の視覚特性に近い評価を行う.

- 輝度: 画像全体の明るさの平均値を比較する.
- コントラスト: 画像内の明暗のばらつき（分散や標準偏差）を比較する.

- 構造: 画像中の局所的なパターン（エッジやテクスチャなど）の相関を評価する.

SSIM の一般式は式 5.1 に示す通りである.

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{5.1}$$

ここで,

- μ_x, μ_y は画像 x と y の輝度平均,
- σ_x^2, σ_y^2 はそれぞれの分散,
- σ_{xy} は共分散,
- C_1, C_2 は分母がゼロになるのを防ぐための小さな定数である.

一般に, SSIM の値は 0 から 1 の範囲を取り, 1 に近いほど画像間の類似性が高いと評価される.

5.4 実験結果

5.4.1 ナビゲーション性能の評価

表 5.1 および表 5.2 に示すのは, ナビゲーション性能評価のための実験結果である. なお,「衝突率」とは, ナビゲーション中に少なくとも 1 回障害物に衝突した試行の割合のことである.

表 5.1: 音声指示によるナビゲーションの成功率と衝突率

拡散モデルの反復回数	段ボールの筒		Unitree Go2 の箱		赤色のスーツケース		パソコンモニター	
	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)
10	80	20	70	30	20	80	20	80
20	100	0	80	20	20	80	30	70
30	100	0	90	10	20	80	30	70

表 5.2: 画像提示指示によるナビゲーションの成功率と衝突率

拡散モデルの反復回数	段ボールの筒		Unitree Go2 の箱		赤色のスーツケース		パソコンモニター	
	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)	成功率 (%)	衝突率 (%)
10	60	0	70	30	20	80	20	80
20	60	0	70	30	20	80	30	70
30	70	0	90	10	30	70	30	70

また, ナビゲーションが失敗した試行において, 図 5.7 のように足元の障害物に衝突した割合は約 80%であった.

図 5.7: 足元の障害物に衝突する例

5.4.2 ゴール地点の検索性能の評価

表 5.3 に示すのは, ゴール地点の検索性能評価のための実験結果である.

表 5.3: ゴール検索の成功率（%）

指示形式	段ボールの筒	Unitree Go2 の箱	赤色のスーツケース	パソコンモニター
音声	90	80	100	100
画像	50	60	100	100

ゴール探索が失敗した事例として, 以下の状況が確認された.

- 「段ボールの筒」の画像を提示した際, Gemini は筒の背後に配置された箱を目的対象と誤認識した.
- 「Unitree Go2 の箱」の画像を提示した際, 目的対象を「白い箱」として誤認識する事例が発生した.

5.5 考察

表 5.1 および表 5.2 の結果から, ゴール先のオブジェクトが「段ボールの筒」および「Unitree Go2 の箱」の場合, 拡散モデルの反復回数を増加させると成功率が向上することが確認できた. 一方, 「赤色のスーツケース」および「パソコンモニター」では, 反復回数の増加による成功率の変化は認められなかった.

この差異は, 前者のケースでは経路上に障害物がほとんど存在しないのに対し, 後者のケースでは図 5.7 のようにカメラの画角外に位置する足元の障害物が存在するためと考えられる. 実際, 足元の障害物との衝突によりナビゲーションが失敗した事例は, 失敗した試行全体の約 80%を占めることが確認されている. したがって, 観測画像のみを用いて経路を推論する NoMaD では, 視界外の障害物を認識できず, 反復回数を増やして精度を高めても, 結果として経路生成に誤りが生じたと考えられる. この点に関して, LiDAR などの追加センサを併用することで, カメラでは捉えきれない障害物の検出が可能となり, ナビゲーションの安全性が向上することが期待される.

また, 表 5.3 の結果から, 画像提示指示を用いたゴール検索において, 「段ボールの筒」と「Unitree Go2 の箱」で特に成功率が低いことが判明した. これは, 「実験結果」でも述べた通り, Gemini による対象物認識が安定して行われなかったことが主な要因である. 具体的には, 「段ボールの筒」の画像提示時, 背景にある箱を誤って対象物と認識する事例が多発した. 原因としては, Gemini に入力するシステムプロンプト内で対象物を明確に指定していなかったことが挙げられる. そこで, システムプロンプトに「画像の中央にある物体」など, 対象物を明確に特定する文言を付加することで, 認識精度の向上が期待できる.

第6章　結論

本研究では, NoMaD を用いたマルチモーダル指示に対応するナビゲーションシステムを開発し, その性能評価を行った. 性能検証の結果, 障害物が少ない経路であれば高い精度でナビゲーションを行えることが確認できたが, 観測できない障害物があるなど, 経路の状況によってナビゲーション精度が大きく変わることが予想される. また, カメラ映像を通した目的地の指示も十分な精度とは言えない. そのため, プロンプトの修正や補助センサーの活用を行うことで, 本システムの性能をさらに高めることが期待される.

謝辞

本研究を遂行するにあたり, 終始ご指導と温かい励ましをいただきました王森レイ講師に深く感謝申し上げます.

また, 本論文の作成に際し, 多大なるご助言とご支援を頂きました高橋寛教授並びに, 甲斐博准教授, 王森レイ講師に深く感謝申し上げます.

並びに, ご審査いただきました甲斐博准教授, 樋上喜信教授, 一色正晴講師に厚く御礼申し上げます.

さらに, 共同研究先である太陽誘電様には, 研究に必要な機材の提供や研究内容に対するご助言など手厚いご支援を頂きました. 本研究の発展に欠かせないご支援をしていただきましたことに, 深く感謝申し上げます.

最後になりましたが, 日頃から切磋琢磨してきた同期にも心より感謝申し上げます. 彼らとの意見交換や情報共有は, 研究を進める上で大いに刺激となり, 常に前向きに取り組む原動力となりました.

以上の皆様の温かいご支援・ご指導がなければ, 本研究を無事に完成させることはできませんでした. ここに改めて深く感謝申し上げます.

関連図書

[1] Ajay Sridhar, Dhruv Shah, Catherine Glossop, Sergey Levine, ”Goal Masked Diffusion Policies for Navigation and Exploration”, https://arxiv.org/abs/2310.07896

[2] IEEE, ”IEEE ICRA Best Conference Paper Award”, https://2024.ieee-icra.org/awards-and-finalists/

[3] Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, Sergey Levine, ”ViNT: A Foundation Model for Visual Navigation”, https://arxiv.org/abs/2306.14846

[4] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, Shuran Song, ”Diffusion Policy: Visuomotor Policy Learning via Action Diffusion, ” The International Journal of Robotics Research (IJRR) 2024: arXiv:2303.04137v5

[5] ”ROS - Robot Operating System”: https://www.ros.org/, 閲覧日:2025/02/10

[6] Kaleem Peeroo, Peter Popov, Vladimir Stankovic, ”A Survey on Experimental Performance Evaluation of Data Distribution Service (DDS) Implementations,” https://doi.org/10.48550/arXiv.2310.16630

[7] ROS2 Documentation.: https://docs.ros.org/en/foxy/index.html, 閲覧日:2025/02/10

[8] Dhruv Shah, Blazej Osinski, Brian Ichter, Sergey Levine, ”LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action”, Conference on Robot Learning (CoRL) 2022 Auckland, New Zealand

[9] Tom B. Brown, et. al, ”Language Models are Few-Shot Learners,” https://doi.org/10.48550/arXiv.2005.14165

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever, ”Learning Transferable Visual Models From Natural Language Supervision”, https://doi.org/10.48550/arXiv.2103.00020

[11] D. Shah, B. Eysenbach, G. Kahn, N. Rhinehart and S. Levine, ”ViNG: Learning Open-World Navigation with Visual Goals”, 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi’an, China, 2021, pp. 13215-13222, doi: 10.1109/ICRA48506.2021.9561936.

[12] Gemini 2.0 Flash.: https://deepmind.google/technologies/gemini/flash/, 閲覧日:2025/02/10

[13] Unitree Go2.: https://www.unitree.com/go2, 閲覧日:2025/02/10