Human-like Guidance のための時空間シーングラフによる案内文生成

鈴木 颯斗† 下村 晃太† 平川 翼† 山下 隆義† 藤吉 弘亘†
大久保 翔太‡ 南里 卓也‡ 王 思源‡
†中部大学 ‡日産自動車株式会社
E-mail: hayato@mprg.cs.chubu.ac.jp

1 はじめに

自動車の運転時に利用されるナビゲーションシステムは，GPS と地図データから取得した情報を用いて目的地までの経路等を提示してナビゲーションを行う手法が一般的である．ナビゲーション情報はモニタに表示されるため，ドライバの注意散漫や誤解を招く可能性がある．一方で，同乗者である人によるナビゲーションは，瞬時に周辺状況を理解し，シーンの動的な情報をもとにした案内を行う．これにより，ドライバの認知負担を最小限にした案内が可能である．こうした人間のように振る舞うナビゲーションシステムを実現する Human-like Guidance の研究が注目されている．

Human-like Guidance の研究として，地図データからランドマークを選択する手法が検討されている．Apple の特許 [1] として開発された技術は，地図データから目標となる交差点付近の建物情報をもとに案内を行う．ランドマークをもとにしたナビゲーション手法は実用的であり，様々な改善手法が検討されている．Amanda ら [2] は，ドライバの好みに合わせたランドマーク選択の最適化方法を提案した．さらに，Jana ら [3] は，周辺に存在する選択可能なランドマークとの距離や方向を考慮して最適なランドマークを選択する方法を提案した．ランドマークを用いたアプローチは，地図データを最新の状態とある必要がある．また，地図データを参照せず，視界情報をもとにしたナビゲーションを行うアプローチが考えられる．その場合，周辺環境に存在するオブジェクトと位置関係を理解する必要がある．自動車の分野に関しては，自動運転の高性能化を目標として，走行シーンの動画像から周辺環境を人間のように詳細に理解することに対して焦点を当てた研究が盛んである．しかし，動画像から周辺の環境を理解したうえで，人間に対してナビゲーションする研究は行われていない．

本研究では，Human-like Guidance の実現に向け，視界情報をもとにした人間のようなナビゲーションを再現するアプローチを考える．本研究では，動画像から検出したオブジェクトの空間的な情報と時間的な変化をシーングラフとして表現し，効果的に周辺環境を理解した上で，動的オブジェクトを含むランドマークを用いた案内文の自動生成法を提案する．また，従来のナビゲーションシステムのようなテンプレート化された案内文の生成ではなく，言語モデルを用いることで高い表現力を持つ案内文の生成を実現する．

2 関連研究

本章では，Image Captioning，Graph Neural Network について述べる．

2.1 Image Captioning

Image Captioning とは，画像とテキストを関連づけ，画像に対する自然言語の説明文を生成するタスクである．Image Captioing は主に，画像認識モデルを用いて与えらえた画像の特徴抽出を行うステップと，その画像特徴量から言語モデルを用いたキャプション生成のステップで行う．

Image Captioning の初期手法である Show and Tell[4] は，Convolutional Neural Network (CNN) [5] を用いて画像の特徴を抽出し，Recurrent Neural Network (RNN) [6] によるキャプション生成を行う．これより，入力画像を詳細に説明することが可能となった．Show and Tell に Attention Mechanism を導入した Show, Attend and Tell[7] は，画像上の単語に関連する領域に注目を当てることでモデルの性能を向上させた．Bottom-Up and Top-Down Attention[8] は，物体検出器で検出した領域の物体ラベルから，直接言語モデルでキャプションを生成する手法である．このアプローチの導入により，画像内の物体や構造を考慮したキャプションの生成が可能となる．

Transformer[9] の登場以降，Transformer を用いたアプローチが主流である．Transformer は，自然言語処理において優れた性能を発揮し，その柔軟性と並列処理能力が Image Captining に適していることが示されている．CPTR[10] は，CNN と Transformer を組み合わせた手法であり，従来の Image Captioning 手法と比較して大幅に精度が向上した．

2.2 Graph Neural Network

グラフ構造データの解析や予測において，Graph Neural Network (GNN) [11] が用いられる．ノードとエッジ

から成るグラフに GNN を適用することで，ノードの隣接関係やグラフのトポロジーを考慮した処理が可能となる．GNN における代表的なモデルの 1 つである Graph Convolutional Networks (GCN) [12] は，畳み込み層をグラフに適用して特徴の抽出を行うことで，ノード分類やグラフ分類において高い性能を発揮した．GCN の拡張モデルの 1 つである Spatial Temporal Graph Convolutional Networks (ST-GCN) [13] は，空間的なグラフ構造に加えて，時系列データに対する畳み込み操作を組み合わせた手法である．本手法は，ノードの空間的な配置と，ノードの時間的変化を同時に考慮することで，時空間的な特徴を捉えることが可能となった．

また，画像上のオブジェクトをノードとして捉え，その情報を構造化したシーングラフとして表現し，GNN を用いた手法も存在する [14]．シーングラフは，画像中のオブジェクトを検出し，検出されたオブジェクト同士の位置関係およびオブジェクト同士の意味的な関係をグラフで表現したものである．Graph R-CNN[15] では，Mask R-CNN[16] によりオブジェクトを検出し，オブジェクト同士の関係性を推定してシーングラフを生成する．シーングラフは画像を効果的に理解をする手法として効果的であることが示されている．

3 提案手法

本研究では，Human-like Guidance の実現に向け，視界情報からナビゲーションを行うアプローチを考える．走行シーンにおける動画像は，ナビゲーションに必要となるオブジェクトに加え，建物などの不要となる情報が多数含まれており，動画像から周辺状況を理解する上で影響を与える可能性がある，こうした課題の解決において，画像全体のテクスチャ情報を用いるのではなく，画像に含まれるオブジェクトに着目して周辺状況の構造を捉えるアプローチを導入し，画像の空間情報をオブジェクトとその関係性に着目したシーングラフとして表現することを考える．また，人間がナビゲーションを行う場合，周囲の動的なオブジェクトの情報を考慮して最適なナビゲーションを行う．そこで本研究では，時系列情報を導入して拡張した時空間シーングラフを生成するアプローチを提案する．これにより，車両周辺オブジェクトの位置関係や動作の変化といった時間的な変化を考慮することが可能になる．次に，言語モデルを用いることで時空間シーングラフの入力に対応した自然な案内文を生成する．

3.1 時空間シーングラフの生成

時空間シーングラフの生成手法の概要について図 1 に示す．空間情報及び時系列情報を表現した時空間シーングラフ G を式 (1) として定義する．

表 1: YOLOv8 による検出対象クラス

Class ID	Class Name
0	person
1	bicycle
2	car
3	motorbike
5	bus
7	truck
9	traffic light

$$G = \{\mathbf{V}_t, \mathbf{E}_t \mid t = 1, \ldots, T\} \tag{1}$$

ここで，T はフレーム数，$\mathbf{V}_t$ はノード集合，$\mathbf{E}_t$ はエッジ集合である．

3.1.1 ノード集合 $\mathbf{V}_t$

ノード集合 $\mathbf{V}_t$ は，シーン画像に存在する自動車や歩行者，自車両などのオブジェクトを表す．自車両を表すセルフノードを $\mathbf{V}_{t,self}$ とし，その他のオブジェクトを表すオブジェクトノードの部分集合を $\mathbf{V}_{t,obj}$ とする．

ここで，シーン内に k 個のオブジェクトが存在すると仮定した場合，$\mathbf{V}_{t,obj}$ は式 (2) のように定義する．

$$\mathbf{V}_{t,obj} = \{\mathbf{N}^{obj}_{t,1}, \mathbf{N}^{obj}_{t,2}, \ldots, \mathbf{N}^{obj}_{t,k}\} \tag{2}$$

$\mathbf{N}$ は各オブジェクトを示すノードを表す．また，セルフノード $\mathbf{V}_{t,self}$ は式 (3) のように定義する．自車両を表すノードの追加を行うことで，シーン内のオブジェクトと自車両との関係性を考慮することが可能となる．

$$\mathbf{V}_{t,self} = \{\mathbf{N}^{self}_t\} \tag{3}$$

従って，ノード集合 $\mathbf{V}_t$ は式 (4) のように定義する．

$$\mathbf{V}_t = \mathbf{V}_{t,obj} \cup \mathbf{V}_{t,self} \tag{4}$$

オブジェクトノード $\mathbf{N}^{obj}_k$ は物体検出器の出力情報を用いる．ここで，物体検出器は Microsoft COCO (2017) Dataset[17] で事前学習した You Only Look Once(YOLO)v8[18] を用いる．また，ナビゲーションに必要となる対象のみに焦点を当てるため，表 1 に示すクラスのみを検出対象とする．ここで，オブジェクトノード $\mathbf{N}^{obj}_k$ には，検出結果である Class ID と BBox 座標を保持する．さらに，BBox 領域を用いてオブジェクト画像を切り取り，CNN による特徴抽出を行う．抽出したオブジェクトの画像特徴量 $Image^{obj}_k \in \mathbb{R}^{d_{obj}}$ はノードに保持する．従って，オブジェクトノードは $\mathbf{N}^{obj}_k = \{ClassID_k, BBoxCoordinate_k, Image^{obj}_k\}$ となる．

セルフノードは $\mathbf{N}^{self} = \{ClassID, SelfCoordinate, Action\}$ と定義する．Class ID は，YOLOv8 の検出可能クラスに “*self*”クラスを追加することで対応し，そ

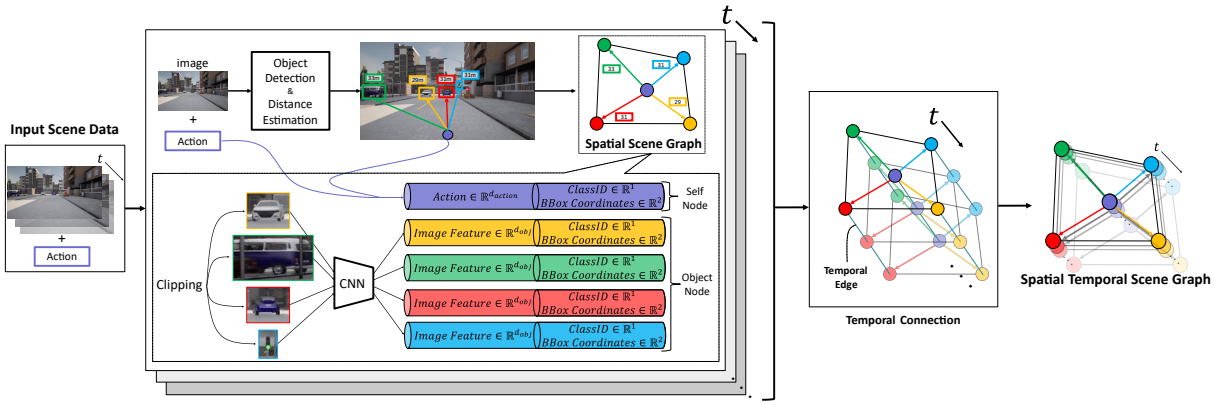


図 1: 時空間シーングラフの生成

の ID を用いる．自車両の座標は，画像の中心最下部の座標を暫定的に用いる．また，ナビゲーションを行うようなタスクにおいては，“右折”や “左折”，“直進”といった交差点に対する動作が既知の情報であることが前提である．よって，この情報をセルフノードの情報として与える．しかし，直接この情報を与えてしまうと，セルフノードとオブジェクトノードのサイズが一致しないため，GNN への適応が不可能となる．そこで，$Action = E(x) \in \mathbb{R}^{d_{action}}$ として埋め込み処理を行うことで対応する．ここで，$E(\cdot)$ は埋め込み処理を表し，x は交差点に対する動作情報を表す．従って，$d_{action} = d_{obj}$ と設定することで，オブジェクトノードとセルフノードは同一のサイズで扱うことができる．

3.1.2 エッジ集合 $\mathbf{E}_t$

ノード集合 $\mathbf{V}_t$ に含まれる各ノード間の関係はエッジ集合 $\mathbf{E}_t$ で表される．あるフレーム t が与えられたとき，$\mathbf{V}_{t,obj}$ に含まれる各ノード同士，及び $\mathbf{V}_{t,obj}$ と $\mathbf{V}_{t,self}$ を接続することで空間情報をシーングラフとして表現する．また，$\mathbf{V}_{t,obj}$ と $\mathbf{V}_{t,self}$ の間の空間エッジを表す部分集合を $\mathbf{E}_{t,self-obj}$，$\mathbf{V}_{t,obj}$ に含まれる各オブジェクト間の空間エッジを表す部分集合を $\mathbf{E}_{t,obj-obj}$ と表記する．

ここで，$\mathbf{E}_{t,obj-obj}$ は，オブジェクトノード $\mathbf{N}_{t,k}$ を用いて式 (5) のように定義する．

$$\mathbf{E}_{t,\text{obj-obj}} = \{(\mathbf{N}_{t,i}^{obj}, \mathbf{N}_{t,j}^{obj}) \mid \mathbf{N}_{t,i}^{obj}, \mathbf{N}_{t,j}^{obj} \in \mathbf{V}_{t,obj}\} \quad (5)$$

セルフノードと各オブジェクトノードを接続するエッジ $\mathbf{E}_{t,self-obj}$ は，式 (6) のように定義する．

$$\mathbf{E}_{t,self-obj} = \{((\mathbf{N}_{t}^{self}, \mathbf{N}_{t,i}^{obj}), \omega_{distance}) \mid \mathbf{N}_{t,i}^{obj} \in \mathbf{V}_{t,obj}\} \quad (6)$$

ここで，$\omega_{distance}$ は各オブジェクトノードとセルフノードを結ぶエッジに対する重みを表す．この重みは，オブジェクトとの物体間距離を用いる．物体距離は，式 (7) を用いて算出する．

$$\omega_{distance} = \frac{f^{camera} \times h^{real}}{h^{BBox} \times h^{image}} \quad (7)$$

f^{camera} はカメラの焦点距離，h^{real} は現実のオブジェクトの高さ，h^{BBox} は BBox の高さ，h^{image} は入力画像の高さをを示す．物体間距離をエッジの重みとして用いることで，オブジェクトとの距離関係や位置関係といった特徴をシーングラフとして表現することが可能になる．

次に，時系列方向に接続するエッジを定義する．時空間シーングラフでは，与えられたフレーム $(t = 1, \ldots, T)$ 毎に，セルフノードと各オブジェクトノードを接続する．$\mathbf{E}_{t,temporal^{obj}}$ を各 $\mathbf{V}_{t,obj}$ の時間的エッジの部分集合，$\mathbf{E}_{t,temporal^{self}}$ を各 $\mathbf{V}_{t,self}$ の時間的エッジの部分集合とし，これらは式 (8) のように定義する．

$$\begin{aligned} \mathbf{E}_{t,temporal^{obj}} &= \{(\mathbf{N}_{t,i}^{obj}, \mathbf{N}_{t+1,i}^{obj}) \mid \mathbf{N}_{t,i}^{obj}, \mathbf{N}_{t+1,i}^{obj} \in \mathbf{V}_{t,obj}\} \\ \mathbf{E}_{t,temporal^{self}} &= \{(\mathbf{N}_{t}^{self}, \mathbf{N}_{t+1}^{self})\} \end{aligned} \quad (8)$$

従って，エッジ集合 $\mathbf{E}_t$ は式 (9) のように定義する．

$$\mathbf{E}_t = \mathbf{E}_{t,\text{obj-obj}} \cup \mathbf{E}_{t,\text{self-obj}} \cup \mathbf{E}_{t,\text{temporal}^{\text{obj}}} \cup \mathbf{E}_{t,\text{temporal}^{\text{self}}} \quad (9)$$

3.2 時空間シーングラフに対応した文章生成

時空間シーングラフの入力に対応した文章生成モデルをグラフエンコーダと文章生成デコーダにより構成する．モデルの詳細を図 2 に示す．グラフエンコーダでは，入力された時空間シーングラフを解析し，グラフ特徴の抽出を行う．ここでは，時空間シーングラフに含まれる空間情報及び時系列情報を考慮することが可能である ST-GCN を用いる．文章生成デコーダでは，グラフエンコーダで得られたグラフ特徴量から文章の生成を行う．ここでは，言語モデルとして Transformer-Decoder を用いる．学習時，Transformer-Decoder は入力シーンに対応した案内文と抽出したグラフ特徴量から単語列を逐次的に推測してデータセットに含まれる案内文の統計的な特徴を獲得する．推論時，グラフ特徴量と文

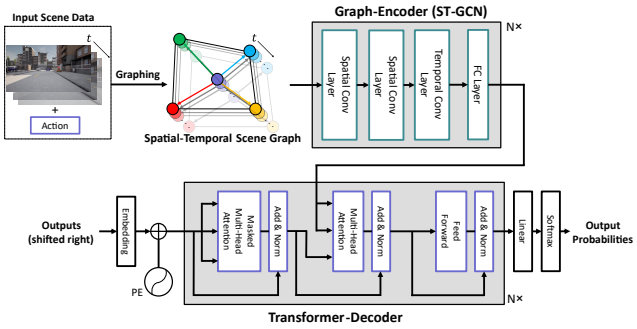


図 2: 時空間シーングラフに対応した文章生成

Ground Truth:
Turn right at the intersection where the blue car in front just turned a moment ago.

Ground Truth:
Straight at the intersection where the red car just passed.

図 3: 作成したシーンと正解案内文の例

章の開始を示すトークンが入力され，学習の際に獲得した文章表現の特徴を基に単語列を逐次的に推測することで案内文の生成を行う．

3.3 モデルの学習

提案手法のモデルは，ST-GCN と Transformer-Decoder からなり，End-to-End で学習する．学習の目的は，モデルが生成した文章と正解の文章との誤差を評価し，最小化することである．そのため，Cross-entropy 損失を用いる．学習では，各ミニバッチにおいてモデルが予測した単語の確率分布と，正解の単語を表す確率分布との Cross-entropy 損失を式 (10) より算出する．

$$L = -\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{T} targets_{i,j} \cdot \log(outputs_{i,j}) \qquad (10)$$

ここで，M はミニバッチ内のサンプル数，T は文章の長さ（単語数），$targets_{i,j}$ はサンプル i において文章内の各単語の位置 j における正解単語のインデックス，$outputs_{i,j}$ はモデルが出力したサンプル i において文章内の各単語の位置 j における各単語の予測確率である．この Cross-entropy 損失を最小化するようにモデルのパラメータが更新され，学習プロセスを通じて，モデルはデータセット内の文章の統計的な特徴を学習する．

4 データセット

学習に用いるデータセットは，走行車両の車載カメラ映像と案内文としてアノテーションされた文章のペアが必要となる．車載カメラの映像を中心としたデータセットは数多く存在し，代表例として Waymo Open Dataset[19] などが公開されている．これらのデータセットは物体検出やトラッキング，セマンティックセグメンテーションなどの画像認識タスクを想定しており，案内文のアノテーションは含まれていない．また，案内文生成タスクでは交差点付近の走行シーンが対象であるが，既存のデータセットは交差点のない走行シーンが多く含まれるため不適切である．そこで，新たに案内文生成タスクに特化したデータセットをシミュレータ環境を用いて作成する．

4.1 走行シーンの撮影

データセット作成のツールとして CARLA Simulator[20] を用いる．CARLA Simulator とは，自動運転システムの開発と検証に向けたシミュレーターソフトである．車両や歩行者といったトラフィックの生成機能や自動運転を行う機能が搭載されており，これらを活用してダッシュボードカメラの視点において走行シーンの撮影を行った．撮影条件は 10fps とし，おおよそ交差点 50m 手前から交差点通過直後，注目対象となるオブジェクトが 1 つ以上存在するシーンを選定した．

4.2 アノテーション

正解案内文のアノテーションは人間により行う．“Turn right”や “Turn left”といった交差点に対しての動作を先頭に配置し，それ以降は注目対象を中心とした案内方法になるよう表現方法を統一した．アノテーションを行ったシーンの例を図 3 に示す．

4.3 作成したデータセットの概要

作成したデータセットは，合計 160 シーン，計 10219 フレームで構成される．全てのシーンに対し 1 つの正解案内文と，動作情報（“right”，“left”，“straight”のテキスト形式）が含まれる．また，学習用データは 136 シーン，評価用データは 24 シーンとする．

5 評価実験

本章では，評価実験により，時空間シーングラフを用いた手法の有効性を検証する．

5.1 時系列情報の有無による評価

時空間シーングラフを用いたアプローチの有効性を検証するため，シーン画像を直接用いた手法をベースラインとして比較する．また，時系列情報を含む場合の有効性を検証するため，複数のフレームを入力した場合と，1 時刻のフレームのみを入力した場合で比較実験を行う．

ベースラインは，画像を用いて直接特徴を抽出する手法であるため，入力から文章生成デコーダまでの間

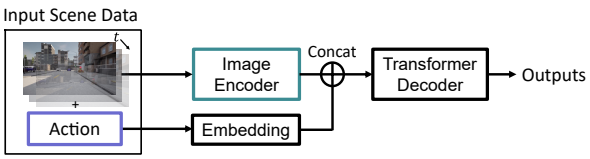


図 4: ベースラインモデルの概略図

表 2: 各モデルの詳細

	シーングラフ	時系列情報	Encoder	Decoder
Baseline			ResNet-18	Transformer Decoder
		✓	ResNet3D-18	
Ours	✓		GCN	
	✓	✓	ST-GCN	

を画像エンコーダに置き換えたモデルを用いる．また，提案手法と同様に，データセットに含まれる交差点の動作情報を入力に含める．その際，$Action = E(x) \in \mathbb{R}^{d_{action}}$ として埋め込み処理を行った動作特徴量を，画像エンコーダで得られた画像特徴量と結合する形で文章生成デコーダへ入力を行う．文章生成デコーダは提案手法とベースラインの比較の公平性を保つため，提案手法と同一の Transformer-Decoder を用いる．ベースラインモデルの概略図を図 4 に示す．画像エンコーダでは，時系列情報を含まない場合は ResNet-18[21] を用い，時系列情報を含む場合は時間的な変化を捉えることができる ResNet3D-18[22] を用いる．

提案手法において時系列情報を含まない場合，空間情報のみを表す空間シーングラフ，すなわち，式 (1) の t=1 を生成する．空間シーングラフ G は式 (11) として定義される．

$$
\begin{aligned}
G &= \{\mathbf{V}, \mathbf{E}\} \\
\mathbf{V} &= \mathbf{V}_{1,obj} \cup \mathbf{V}_{1,self} \\
\mathbf{E} &= \mathbf{E}_{1,obj-obj} \cup \mathbf{E}_{1,self-obj}
\end{aligned}
\tag{11}
$$

また，式 (1) で表される空間シーングラフは時系列情報を考慮する必要はないため，グラフエンコーダは空間情報の特徴を抽出する GCN に置き換える．

提案手法及びベースラインの各モデルの詳細をまとめたものを表 2 に示す．

5.2 実験条件

Transformer-Decoder の設定は，層数を 8，Multi-Head Attention の head 数を 4，Feed Forward Networks の次元数を 1024，エンコーダから文章生成デコーダへの入力次元数を 512 とする．提案手法におけるグラフエンコーダの ST-GCN，及び時系列情報を考慮しない場合の GCN はそれぞれ 2 層とする．動作情報の埋め込みサイズ d_{action}，及び時空間シーングラフ生成時の BBox 領域画像特徴の抽出処理における出力サイズ d_{obj} は，同一の 32 とする．

提案手法とベースラインにおいて，時系列情報を含める場合，入力するフレーム数は 5 として固定する．また，時系列情報を含めない場合においては入力するフレーム数は 1 となる．

学習における設定は，学習率を 1.0×1.0^{-4}，エポック数を 100，Dropout 率を 0.3，バッチサイズを 32 とする．学習の最適化アルゴリズムには AdamW[23] を用いる．

5.3 評価指標

各モデルで生成された案内文と正解案内文の生成精度を定量的に評価する評価指標として，自然言語処理の分野で広く用いられる BLEU[24]，METEOR[25]，ROUGE[26]，BERT Score[27] を用いる．BLEU は，生成された文章と正解文章の類似度を評価する．METEOR は，単語の一致，ステミングによる一致，同義語による一致を評価する．ROUGE は，生成された文章と正解の文章の最長シーケンスを用いて評価する．BERT Score は，言語モデルである BERT[28] を用いた評価指標であり，文脈を理解した評価が可能である．

5.4 定量的評価

定量的評価を行い，提案手法とベースライン手法の案内文の生成精度について比較を行う．まず，テストデータの各シーンの中で最も精度が高いフレームのスコアを用いて評価した結果を表 3 に示す．ここでは，最高スコアを記録したフレームを参照することで，正解案内文の表現方法との一致度を分析する．表 3 より，ベースライン手法は時系列情報の有無に関わらず同程度の精度となった．しかし，提案手法において時系列情報を考慮することで最も優れた精度を達成し，正解案内文の一致度が高くなることを確認した．

次に，各フレームで得られたスコアをシーンごとに平均化した際の定量的評価を表 4 に示す．ここでは，各シーンの全体的な推論精度を比較することが目的である．各時系列のスコアを平均することで，時間の経過に伴う変動や安定性に着目し，高品質な文章生成の一貫性を分析する．表 4 より，時系列情報の有無に関わらずベースライン手法と比較して提案手法はスコアが向上することを確認した．また，提案手法において時系列情報を持つ場合に最もスコアが高くなり，時空間シーングラフは高品質な案内文生成の一貫性に寄与することが示された．

5.5 定性的評価

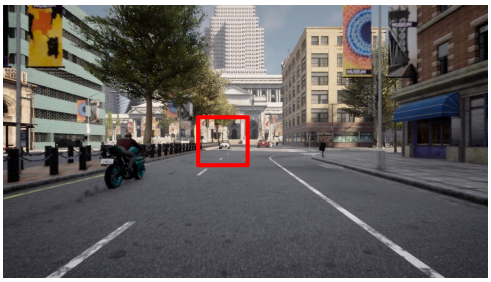
提案手法とベースライン手法によって生成された案内文の比較を行うことで，提案手法の有効性を調査する．提案手法とベースライン手法の各モデルの案内文生成結果の例を図 5 及び図 6 に示す．生成結果より，提案手法では画像内に存在するオブジェクトをを基準とした案内文が生成されていることが確認できる．時系列情報を含む場合においては，提案手法はベースライ

表 3: 最も精度が高いフレームのスコアを用いた評価結果

	シーングラフ	時系列情報	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	ROUGE	BERT
Baseline			0.625	0.511	0.430	0.377	0.678	0.679	0.948
		✓	0.612	0.490	0.404	0.302	0.662	0.688	0.949
Ours	✓		0.574	0.457	0.357	0.297	0.670	0.672	0.949
	✓	✓	**0.682**	**0.569**	**0.461**	**0.389**	**0.748**	**0.742**	**0.954**

表 4: 各フレームで得られたスコアを平均して評価した結果

	シーングラフ	時系列情報	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	ROUGE	BERT
Baseline			0.457	0.333	0.257	0.202	0.512	0.553	0.928
		✓	0.435	0.316	0.234	0.180	0.514	0.556	0.928
Ours	✓		0.478	0.356	0.252	0.209	0.575	0.598	0.929
	✓	✓	**0.522**	**0.392**	**0.287**	**0.224**	**0.603**	**0.624**	**0.933**



Action : right

Ground Truth:
Turn right at the intersection where the white car passed.

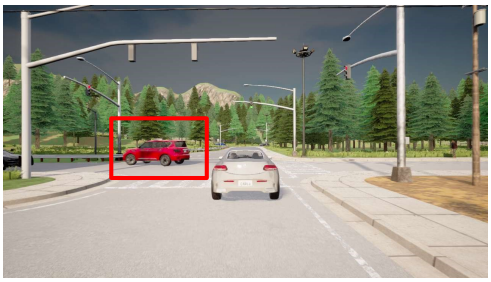
Baseline(Without Temporal Information):
Turn right at the intersection where the white car is located.

Baseline(With Temporal Information):
Turn right at the intersection just before the red car.

Ours(Without Temporal Information):
Turn right at the intersection just before the white car.

Ours(With Temporal Information):
Turn right at the intersection where the white car passed.

図 5: 入力シーンと案内文の生成結果の例．Ours(With Temporal Information) では Ground Truth と同一の文章が生成されたことを確認できる．



Action : left

Ground Truth
Turn left at the next intersection, just like the red car.

Baseline(Without Temporal Information)
Turn left in the direction where the black car passed.

Baseline(With Temporal Information)
Turn left following the red car.

Ours(Without Temporal Information)
Turn left at the intersection where the red car is located.

Ours(With Temporal Information)
Turn left at this intersection, following the direction of the red car.

図 6: 入力シーンと案内文の生成結果の例．Ours(With Temporal Information) では Ground Truth と近しい文章が生成されており，オブジェクトの動作を考慮した表現となり詳細な説明がされていることが確認できる．

ンと比較してオブジェクトの進行方向や位置が考慮されていることが確認できる．動的なオブジェクトの時間的な変化を捉え，生成する案内文の表現力の向上に寄与することが示された．

また，提案手法において入力する動作情報を変更した際のそれぞれの生成結果の例を図 7 に示す．図 7 より，入力した動作情報に合わせて適切なオブジェクトが選択されており，そのオブジェクトを中心とした案内文が生成されていることが確認できる．

6 おわりに

本研究では，Human-like Guidance の実現を目的に，視界情報から案内文を生成する手法を提案した．提案手法では，空間情報および時間情報を表現可能な時空間シーングラフを生成し，文章生成モデルを用いて案内文を生成するアプローチを導入した．独自に作成した Human-like Guidance に適したデータセットを用いて実験を行った結果，画像から直接特徴を抽出する方法と比べ，時空間シーングラフとして特徴表現することで案内文の生成精度が向上することを確認した．また，時系列情報の有無による比較実験を行った結果，時系列情報を持つ時空間シーングラフを用いた場合，オブジェクトの時間的な変化を考慮した案内文が生成されたことを確認した．また，各シーンの全てのフレームで評価した結果，時系列情報を含む時空間シーングラフは高品質な案内文を一貫して生成できることを確認した．従って，時空間シーングラフは，周辺環境の理解において有効であり，案内文の表現力向上に寄与することを示した．

しかし，本研究で作成したデータセットに含まれるシーンは限定的であり，より複雑な道路環境において

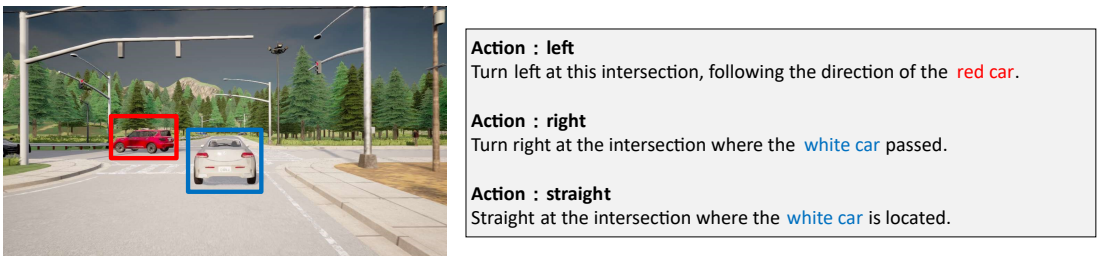


図 7: 提案手法において入力する動作情報を変更した際のそれぞれの生成結果の例．Action が “left”の場合は “red car”，“right”と “straight”の場合は “white car”を注目対象とした案内文が生成され，入力した Action に合わせて適切な案内文が生成されていることを確認できる．

の有効性は分析が不十分である．そのため，今後はデータセットの拡張により様々な状況を検証し，提案手法の応用性について確認する必要がある．また，提案手法のアプローチは実環境データにも適応することが可能であり，実環境シーンでの検証を行う予定である．

参考文献

[1] Anil K. Kandangath and Xiaoyuan Tu. Humanized navigation instructions for mapping applications, April 2015. US Patent application.

[2] Amanda Cercas Curry, Dimitra Gkatzia, and Verena Rieser. Generating and evaluating landmark-based navigation instructions in virtual environments. In *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*, pages 90–94, 2015.

[3] Jana Götze and Johan Boye. Deriving salience models from human route directions. In *Proceedings of the IWCS 2013 Workshop on Computational Models of Spatial Language Interpretation and Generation (CoSLI-3)*, pages 7–12, Potsdam, Germany, 2013. Association for Computational Linguistics.

[4] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164, 2015.

[5] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[6] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[7] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2048–2057, 2015.

[8] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6077–6086, 2018.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017.

[10] Wei Liu, Sihan Chen, Longteng Guo, Xinxin Zhu, and Jing Liu. Cptr: Full transformer network for image captioning, 2021.

[11] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

[12] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 5425–5434, 2017.

[13] Shijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[14] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 690–706, 2018.

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. *arXiv preprint arXiv:1405.0312*, 2014.

[18] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023.

[19] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, and Zhifeng Chen. The waymo open dataset: High resolution sensor data for autonomous driving. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[20] Alexey Dosovitskiy, Thomas Brox, Nikolay Stoyanov, Tom Erez, Eddy Ilg, Alexander Pishchulin, Maxim Yankovskiy, and Daniel Cremers. Carla: An open-source simulator for autonomous driving, 2017.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[22] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2018.

[24] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics (ACL)*, 2002.

[25] Michael Denkowski and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

[26] Chin-Yew Lin. Rouge: A package for automatic evaluation of summarization quality. *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004.

[27] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7971–7981, 2020.

[28] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2019.