

# 修士学位論文

360°映像群からの Movie Map の  
構築とバーチャル探訪への応用

Building Movie Maps using 360°  
Videos and its Application to  
Virtual Exploration

令和3年1月28日

東京大学大学院 情報理工学系研究科

電子情報学専攻

48-196426 杉本 直樹

指導教員 相澤 清晴 教授

# 内容梗概

地図アプリケーションに映像メディアを紐づけ、地図上の移動に伴った映像の再生によりユーザーにまるでその場にいるかのような体験を提供するインターフェースを Movie Map と称する。過去の Movie Map はその撮影コスト、構築コストが高く汎用的ではなかったが、道案内やバーチャル探訪といったタスクでの有用性は現在台頭する他のアプリケーションよりも高いと言える。本論文では 360°映像を利用した Movie Map について、その撮影と構築のコストを低減し一般のあらゆる地域について Movie Map を自動的に構築する手法を提案する。提案手法においては撮影した映像群のどの映像のペアがどの部分で交差しているのかを検出し、各交差点についてそれぞれの映像からペアのもう片方の映像に移り変わる、すなわち交差点でターンを行う映像を合成することで、交差点における撮影コストと Movie Map の構築コストを従来のものと比較して低減している。そして、ユーザーが直感的に操作を行い地域内を探検できる Movie Map インターフェースを開発し、前述の交差点を基準として分割した映像群と合成された交差点ターン映像からなる映像データベースを用いて京都駅周辺や大阪難波町、東京大学本郷キャンパスなど様々な地域において Movie Map が自動的に構築できることを示す。本手法においては、提案する映像群を入力とした交差点の検出手法に visual SLAM を利用したものと深層学習を利用したものがあり、開発した Movie Map インターフェースに道案内を主軸としたものとバーチャル探訪を主軸にしたものがそれぞれ存在する。実験では、交差点におけるターン合成映像の主観評価による合成方法の比較実験、開発したバーチャル探訪型 Movie Map インターフェースのユーザビリティについての被験者実験を行い、提案手法それぞれの妥当性を示す。また映像群からの交差点検出には深層学習を用いたもう 1 つの手法を提案しており、その交差予測についても検出精度の比較実験を行っている。

# 目次

第 1 章	序論	1
1.1	研究背景	1
1.2	論文構成	2
第 2 章	関連技術・研究	3
2.1	360°動画像	3
2.1.1	360°画像を用いた深層学習	4
2.2	動画像メディアデータを紐づけた地図アプリケーション	4
2.3	Movie Map	8
2.4	シーンの 3 次元復元とカメラ位置推定	8
第 3 章	提案手法	10
3.1	360°映像の収集	10
3.2	交差点の検出と校正	11
3.2.1	vSLAM の適用	12
3.2.2	参照座標を用いた共通座標へのマッピング	14
3.2.3	座標情報に基づく交差点の検出	14
3.2.4	特徴量マッチングを用いた交差点フレームの校正	15
3.3	交差点情報を用いた映像データベースの構築	16
3.3.1	交差点を基準とした映像分割	16
3.3.2	交差点におけるターン映像の合成	16
3.4	映像データベースと交差点情報を用いた Movie Map インターフェース	17
3.4.1	道案内に特化した自由経路合成インターフェース	17
3.4.2	バーチャル探訪に特化した Movie Map 探索インターフェース	18
3.5	深層学習を用いた単一映像における交差点予測	19

第 4 章	実験	28
4.1	合成遷移映像についての主観評価実験 . . . . .	28
4.2	Movie Map インターフェースのユーザビリティについての主観評価実験	30
4.3	深層学習による交差点予測の精度比較実験 . . . . .	32
4.3.1	Direct Intersection Prediction (DIP) . . . . .	32
4.3.2	Movable Direction Prediction (MDP) . . . . .	32
4.3.3	Front Vanishing Point Prediction (FVPP) . . . . .	32
第 5 章	結論	35
5.1	まとめ . . . . .	35
5.2	本郷 Movie Map . . . . .	36
参考文献		37
発表文献		40

# 目次

2.1	<b>360°画像の例</b> . 一枚の画像中に全方位の情報が収められており, 回転処理を行う事で任意の方向を正面に捉えることが出来る. . . . .	4
2.2	<b>Google Street View の様子</b> . 選択位置に最も近い画像が表示され, その画像が属する経路にユーザーの位置は調整される. 画像上に表れる矢印は画像と自分の向きを考慮した経路の伸びている方向に対応した方向に出現する. . . . .	5
2.3	<b>Mapillary の様子</b> . ユーザーの投稿したメディアにアタッチされた GPS 座標に基づいて地図上に動画データが配置されており, ユーザーはそれらの再生と切り替えにより地図上を移動する. アタッチされた位置情報を基に映像間の切り替え候補となる矢印が出現するため, 実際の交差点とは大きくずれが生じる場合が多い. . . . .	6
2.4	<b>Aspen Movie Map のディスクへの保存設計</b> アスペン街の Movie Map システムにおいては, 1 本の道路について真っすぐ進む映像を光ディスクに収めた後, その交差点地点において左に曲がる映像が逆再生方向に, 右に曲がる映像が順再生方向に挿入されており, ジョイスティックの入力に応じて適切な場所, 方向にディスクを再生する事でリアルタイムに物理的な切り替えを可能にしている. (図は [1] より引用) . . . . .	7
2.5	<b>Bundle Adjustment の概要</b> . 画像上の特徴点の座標に基づき, カメラの位置, 向き, 三次元空間上の特徴点の位置を最適化する. カメラのパラメータや (図は [2] より引用) . . . . .	9

3.1	<p><b>Movie Map</b> システムフローチャート. 各映像について撮影された 360°映像は自動化された図中に緑色で示された Movie Map システムの入力となり, システムは交差点を基準とした映像群の位置関係情報と分割された映像, そして交差点における合成遷移映像を出力する. それらの情報は Movie Map インターフェースによって利用され, 一般のあらゆる地域について Movie Map が実現される. . . . .</p>	11
3.2	<p>東京大学本郷キャンパスにおける <b>Movie Map</b> ルートマップ. 東京大学本郷キャンパス構内における主要な経路を 27 本設定し, それらの端点に数字の ID を付けたルートマップ. 撮影の際にはこれらの経路に従い, 1 つの経路について双方向に進行する映像を収集する. その際撮影の移動速度や双方向に撮影する際の微小な座標のズレは許容される. . . . .</p>	12
3.3	<p><b>OpenVSLAM</b> における 360°画像からの <b>ORB</b> 特徴量抽出の例. 図中の赤い点で示された部分が抽出された特徴点である. 進行方向から見て左右方向の, 建物上に多くの特徴点が検出されていることが見て取れる. . . . .</p>	13
3.4	<p><b>OpenVSLAM</b> により推定された, 単一映像における三次元空間上の特徴点とカメラ軌跡. 緑色に示されているのが映像内の相対的なカメラ位置と向きの軌跡である . . . . .</p>	13
3.5	<p>参照座標をアンカーとした東京大学本郷キャンパス構内における地図上へのマッピング結果. 赤いマーカーが順方向の映像におけるキーフレームのカメラ位置を参照座標を基に回転, スケーリングさせることにより 2 次元地図上に投影したもので, 青いマーカーは逆方向の映像における投影点である. . . . .</p>	21
3.6	<p>カメラ軌跡のペアについての交差点検出の例. 図中に赤と緑のサークルで表されているのが各映像におけるキーフレームのカメラ位置座標である. 軌跡の一部分を矩形で囲み, それらの重なりを検出された場合その矩形内にさらに小さな領域の矩形を作成し重なりを検証する. 経路の端点においては採集フレームの位置座標を延長する事で, T 字路などにおける検出漏れを防いでいる. . . . .</p>	22

3.7	交差点フレーム校正の概略図。検出された交差点フレームの周辺フレーム画像を選び出し、vSLAM の推定したカメラ姿勢を基に全ての画像を同じ向きに回転させる。その後、ORB 特徴点のマッチングを行い最も近い画像のペアを選び出し、そのペアを正確な交差点フレームとする。図中においては、緑色のマーカーで表された経路のうち交差点フレームとして検出されたフレームの角度方向に全ての候補画像を回転させている。	22
3.8	交差点における分割映像、合成映像の管理。経路映像は双方向に撮影しているため、身体的な 1 つの交差点に対して検出される交差点は 4 つとなる。なお同経路を逆方向に進む映像同士とは、交差が検出されない例外処理が施されている。遷移映像の合成については、これらの映像間を遷移する全パターンの映像を合成する必要がある。	23
3.9	交差点における合成された遷移映像の例。2 つの 360°画像を重ね合わせ回転させながらその比率を変更していく手法を採用している。映像上では比較的自然な移り変わりが表現可能である。	24
3.10	道案内に特化した <b>Movie Map</b> インターフェースの概要。(a) に示す操作画面において、紫のサークルで表された交差点を選択していくことで自由な経路を決定する。そして経路映像の合成を行うと映像の再生画面が現れ、作成した経路に沿って移動する映像が再生される。映像中の現在位置は地図画面に赤いサークルで表示される。	25
3.11	探索型 <b>Movie Map</b> インタフェースの様子。2 つの 360°地図上でスタート地点と進行方向を定めることで、その方向に進む映像の再生が開始する。再生している映像の再生時間と向いている方向に応じて地図画面の赤い矢印が移動・回転する。	26
3.12	探索型 <b>Movie Map</b> インタフェースの交差点での再生画面の遷移。交差点に侵入すると再生画面上に、進める方向に応じた赤い矢印の UI がポップアップする。矢印を選択することで進行方向を決定し、ターンを行う場合は合成映像を挿入した上で再生映像の切り替えを行う。	26
3.13	設置したバーチャル看板の例。バーチャル看板はインターフェース上で映像を再生、閲覧しながら設置する事も、データとして事前に書き込むことでインターフェース上に正確に設置する事も可能である。クリックする事で紐づけられた web サイトにアクセスする事も出来る。	27

3.14	単一映像からの交差判定を利用した交差点範囲の絞りこみ。ルートマップ上で交差する 2 つの経路について，その映像内の各フレーム画像について交差点かどうか判定を行う。図中のグラフの横軸は映像の再生時間，縦軸は交差点の尤度を表しており，閾値を超えつづけた 1 範囲を 1 つの交差点候補とする。そして候補同士のペア全てについて校正作業を行い，全体を通じて最高のスコアを出したフレームのペアを交差点として検出する。 . . . . .	27
4.1	各交差点における映像切り替えの様子。2 つの交差点において，直接交差点フレームで映像を切り替えた場合 (手法 A)，片方の端点フレーム画像を回転させるだけの映像を挿入した場合 (手法 B)，提案する合成遷移映像を挿入した場合 (手法 C) の映像。交差点 2 は交差点 1 と比較してオブジェクトや風景の様子が切り替え前後で大きく変化している。 . . . .	29
4.2	実験タスクにおける探索エリアの図。赤枠で囲まれた 2 つの範囲を探索対象エリアとする。被験者はそれぞれ片方のエリアを Google Street View を用いて，もう片方のエリアを Movie Map を用いて探索を行う。 . . . .	31
5.1	<b>moviemap.jp</b> における初期画面。本郷キャンパス構内の Movie Map の他，いくつかの地域についても同様に Movie Map の構築を行っており，今後公開予定となっている。 . . . . .	36

# 表目次

4.1	合成遷移映像についての主観評価実験の結果. . . . .	30
4.2	Movie Map と GSV の比較主観評価実験の結果. . . . .	31
4.3	360°画像の交差点判定における各学習手法の性能比較実験の結果 . . .	33

# 第 1 章

## 序論

### 1.1 研究背景

ユーザーが地図アプリケーション上で移動を行う際、現在位置に対応した映像を再生する事で実際にユーザーが対象地域内を移動しているかのような体験を提供するインターフェースを Movie Map と呼ぶ [1]。Movie Map は過去に提唱されたコンセプトであり構築された事例も存在する [1] [3] [4] が、移動体験の連続性の担保のためにはルートの方岐、すなわち交差点のターンをすべてカバーする映像の撮影が必要であり、また地域の道の構造によって特有の物理的なシステムの構築方法を考えなくてはならず総じてコストが高かった。そのため、Movie Map が有用であるとされる道案内やバーチャル探訪といったタスクでは、後年に台頭した Google Street View [5] を始めとする Movie Map とは異なる地図アプリケーションに代替されていった。

しかし、Movie Map はその移動体験の連続性やデータ密度から、前述のタスクにおいて、近年利用されているアプリケーションには無い高い没入感を活かした高いユーザビリティを得ることが出来ると考えられる。

そこで本研究では、Movie Map の撮影や構築のコストを低減し、一般の様々な地域について汎用的に Movie Map を構築しユーザーが利用できるようにすることを目指す。一般に Movie Map 構築の正確性とコストはトレードオフとなっており、より正確に、すなわち手作業による構築をするほどその労力は大きくなり、より低コストに、すなわち自動化された構築をするほど正確性は失われてしまう。提案する Movie Map の構築手法では、対象地域において Movie Map に採用する道を進むストリート映像群と最低限の補助情報を入力とし、自動的に交差点を基準としたそれらの分割映像と交差点ごとの合成ターン映像を出力する。この際、入力する補助情報はより単純で簡単なものが望ましく、それ

でいて出力はより正確なものが求められる。本論文では、撮影を行った各映像についてその始点と終点の座標情報を補助情報として入力し visual SLAM を利用して交差点の検出を行う手法と、どの映像同士が交差するかというマップ情報を補助情報として入力し深層学習による映像内の交差点部分を推論する事で交差点の検出を行う手法を提案している。現時点では前者がより正確である一方、後者はより様々な地域に適用でき汎用性が高く補助情報の入力の手間も少ない。

## 1.2 論文構成

以下に本稿の構成を示す。

- 第 1 章 序論
- 第 2 章 関連研究
- 第 3 章 提案手法
- 第 4 章 実験
- 第 5 章 結論

第 1 章では、研究の背景及び本稿の構成を述べる。

第 2 章では、本研究に関連する研究を述べる。Movie Map の構築に関して必要となる要素技術の紹介と、既存のシステムに存在する技術的課題の提示を行う。

第 3 章では、提案する Movie Map 構築システムと Movie Map インターフェースについて述べる。

第 4 章では、交差点における合成ターン映像の挿入による連続的な移動体験の主観評価実験、Movie Map インターフェースのユーザビリティについての主観評価実験について述べる。加えて深層学習を利用した交差点検出の前段階となる、360°画像の交差点判定についての精度比較実験についても触れる。

第 5 章では、本稿の内容を再度まとめ、本研究の成果として公開されている東京大学本郷キャンパス Movie Map について紹介する。

## 第 2 章

# 関連技術・研究

本章では提案する Movie Map の自動構築に関連する各種技術について，その概要と現在の状況を紹介する．

### 2.1 360°動画像

提案する Movie Map インターフェースにおいては，360°映像が経路に沿って再生され，インターフェース上でユーザーは自由にあたりを見回すことが出来る．

ここで 360°映像とは，映像の各フレームが equirectangular 形式の画像として保存されたデータ形式の事を指す．equirectangular 画像は ERP 画像と略され，全方位の視覚情報を一枚の画像上に収めたものとなっている．図 2.1 にその例を示す．2 枚の画像は同じ場所についての情報を収めた ERP 画像であり，左に示した画像を鉛直軸方向に 90°回転させる事で右に示す画像に変換される．ERP 画像は上下方向についての情報を平面である画像上に収めるため歪みが生じてしまい，また左右方向についても画像上の中間地点に表現されるため直感的に理解が難しい場合も多い．そのため，一般に ERP 形式で保存されたデジタル 360°画像データはビューアプログラムを通じて天球にテクスチャとして張り付けた上でユーザーに提供されることが多い．その際，ユーザーは天球の内部から天球の一部を見る形でデータを閲覧するため天球のどの部分を見るかを操作する事で 360°画像が撮影された空間をあたかも見回しているかのような体験が得られる．

近年では 360°画像を撮影する事の出来る 360°カメラは一般向けに販売提供されており，ロボットやバーチャルリアリティ向けのアプリケーションのために利用されていることも多い [6] [7]．Movie Map は，この仕組みを 360°動画像データに適用する事で，ユーザーに経路を進みつつあたりを見回す，すなわち実際に対象地域を移動しているかのような



図 2.1: **360°**画像の例 . 一枚の画像中に全方位の情報が取められており, 回転処理を行う事で任意の方向を正面に捉えることが出来る.

な体験を提供するものである.

### 2.1.1 360°画像を用いた深層学習

画像を深層ニューラルネットワークの入力として様々なタスクを学習させる研究は幅広く行われており, とりわけ CNN [8] を利用した数多くの深層ネットワーク [9] [10] [11] が物体検出 [12], 画像分類 [13], セマンティックセグメンテーション [14] といったタスクで人間と同じ, あるいは人間を上回る性能を發揮している [15] [16] [17]. しかし, CNN のアーキテクチャは画像のピクセル同士の画像上の距離関係が位置, 方向に関わらず一定であることを前提としているため, 通常のパースペクティブ画像を入力として高性能を發揮するモデルに対して 360°画像をそのまま入力として学習, 適用した場合, 同様の性能が得られる保障は無い. そこで 360°画像を入力とした CNN を用いた深層学習においては, パースペクティブ画像に対して高い性能を發揮するモデルをそのまま適用する手法 [18] の他に, 入力画像をキューブマップと呼ばれる天球上のテクスチャを立方体の各面に投影する手法 [19] を用いて複数枚のパースペクティブ画像に変換しモデルに入力する手法 [20] や, CNN の畳み込み演算に用いるカーネルのサイズや動かし方, パラメータを画像の位置によって変化させることで 360°画像の歪みにモデルを対応させる手法 [21] などが存在する.

## 2.2 動画画像メディアデータを紐づけた地図アプリケーション

デジタルアプリケーションとしての地図には, その地域の地理的情報や数値データなど, 様々な情報を結び付けて表示する事で単なる地図よりもユーザーにより対象地域を分かりやすく伝える事が出来るという特色がある [22] [23] [24] [25] [26]. 中でも, 地図上



図 2.2: Google Street View の様子. 選択位置に最も近い画像が表示され, その画像が属する経路にユーザーの位置は調整される. 画像上に表れる矢印は画像と自分の向きを考慮した経路の伸びている方向に対応した方向に出現する.

の移動に対応して現在位置の画像や映像を再生するアプリケーションは, ユーザーに直感的に地域内での自分の立ち位置を理解させ高い没入感を与えることから道案内やバーチャル探訪といったタスクで広く用いられる [5] [27]. Google Street View (GSV) [5] は現在世界で広く使用されている地図アプリケーションであり, ユーザーの現在位置, あるいは選択した場所に対応する 360°画像が表示, 更に画像上にポップアップする矢印をタップ, またはクリックする事で経路に沿った移動が可能なインターフェースとなっている. 内部的には地図にあらかじめ経路が登録されており, 経路上にはある程度定期的なインターバルで撮影された 360°画像が登録されている. インターフェース上で地図の一点を選択すると, ユーザーは最も近くの座標に登録された画像が属する経路に移動し, 画像上では経路が存在する方向に矢印の UI が出現する. 画像が交差点に一定以上近い場合図 [reffig:related:gsv](#) のように現在の経路とは異なる経路に移りかわる方向にも矢印がポップし, 2 次元的な移動を可能としている.

GSV は直感的な操作が可能であり, インターフェースがシンプルで分かりやすいため前述のタスクで高いユーザービリティを誇る. ただし, その特性には課題も存在する. 実際に GSV を利用していると, 画像間の移動に距離の開きがあり自分の位置を見失い, 詳細な道順の確認や風景を見回しながらの移動が困難な場合が存在する. これは画像が離散的に撮影されているうえその間隔が一定でなく, 不連続な移動が起きてしまう事による. また気になった場所に丁度該当する場所の画像が存在しない場合, 1 つのランドマークや



図 2.3: Mapillary の様子。ユーザーの投稿したメディアにアタッチされた GPS 座標に基づいて地図上に動画データが配置されており、ユーザーはそれらの再生と切り替えにより地図上を移動する。アタッチされた位置情報を基に映像間の切り替え候補となる矢印が出現するため、実際の交差点とは大きくずれが生じる場合が多い。

建物をピックアップして詳細に確認する事が難しい。

この課題の単純な解決方法として、収集するメディアデータを 360°静止画像から 360°映像に変更する事が考えられる。しかしその場合、画像の位置を手入力によって正確に経路上に登録している GSV と同じ方法で映像を経路上にマッピングする事は困難である。

Mapillary [27] は、その問題に GPS による位置情報を利用する事で対処しようとした事例である。Mapillary はユーザー投稿型のサービスであり、投稿された映像データはパースペクティブ、360°を問わず GPS の位置情報に基づいて地図上にマッピングされる。これを利用するユーザーは、図 2.3 のように地図上に示される緑色の経路を選択し映像を再生し、経路同士が交差点する地点付近で映像中の遷移 UI を選択する事で再生する映像を切り替えて移動を行っていく。

しかし、それらの映像の位置は GPS の位置情報を参照して決定しているため、誤差が大きく、映像間の遷移が行える映像上の地点は実際にそれらの経路が交わっている地点と大きくずれている事が多い。結果として、映像間の遷移を行うと今までとは少し違った場所の映像が急に再生され始め、自分の位置のワープに戸惑う状況が多々発生する事になる。そのため、映像の再生と映像間の遷移を通じて移動の連続性が保たれているとは言えない。

このように、撮影した映像同士の位置関係、とりわけ 2 つの映像が正確にどのフレーム

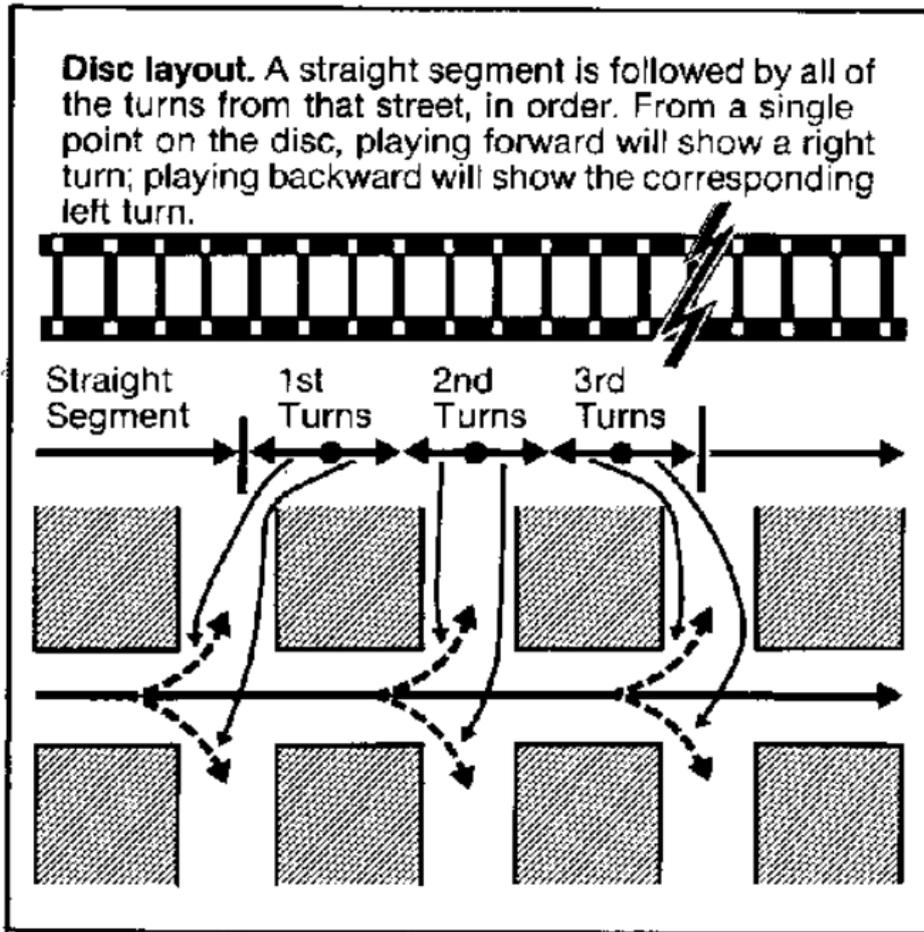


図 2.4: Aspen Movie Map のディスクへの保存設計アスペン街の Movie Map システムにおいては、1本の道路について真っすぐ進む映像を光ディスクに収めた後、その交差点地点において左に曲がる映像が逆再生方向に、右に曲がる映像が順再生方向に挿入されており、ジョイスティックの入力に応じて適切な場所、方向にディスクを再生する事でリアルタイムに物理的な切り替えを可能にしている。(図は [1] より引用)

で交差しているかを検出するのは映像間の遷移において位置の変化の連続性を担保するために必要であるが、現在の一般的な地図アプリケーションに用いられる手法やその延長では、それが困難である事が分かる。

## 2.3 Movie Map

1980年にそのコンセプトが提唱された Movie Map [1] は、交差点間を移動する  $360^\circ$  映像と交差点をそれぞれの方向に進む  $360^\circ$  映像を車載カメラによって別々に撮影し、それぞれを別の光ディスクに保存、インターフェースとしてユーザーが物理的なジョイスティックを操作する事で映像の再生終了時に次に再生するディスクを選別する事で、まるでユーザーが実際に車に乗ってドライブをしているかのような体験を提供するものであった。

手作業で交差点までの進行映像と交差点におけるターン映像を正確に隔てているため映像の連続性が高く、ユーザーは高い没入感を得られる一方、その撮影と物理的なシステムのコストはとても高い。[1]において対象地域となったアスペン街は図 2.4 に示す通り道路構造が単純であるものの、複雑な構造を持つ値域について撮影した映像を交チェックし差点単位で分割していくのはスケラビリティの面で現実的ではない。[1]のあとに、いくつかの例において Movie Map の構築の自動化について手法が提案されているものの [3] [4]、対象の道路が円形の道路とそれらを繋ぐ道路から構成されていることを前提とするなど、その自動化手法は対象地域の構造に依存しており汎用的ではない。

本稿で提案する手法においては、既存の Movie Map 構築システムに存在する撮影と構築の高いコストの低減のため、交差点におけるターン映像の撮影を行わず、一般のあらゆる地域において撮影した映像群と最低限の補助情報を入力として自動的な Movie Map の構築を可能としている。

## 2.4 シーンの 3 次元復元とカメラ位置推定

対象地域内や単一の建造物など、1つのシーンを様々な場所、角度から撮影した静止画像や映像を用いて、3次元空間上にそのシーンを復元する技術として Structure from Motion (SfM) [28] [29] や visual SLAM [30] [31] が知られている。画像特徴点の3次元空間上での位置を推定する際には、同時にそれぞれの撮影場所と角度、すなわちカメラ姿勢も最適化される。従来は図 2.5 に示すような、Bundle Adjustment [32] [33] と呼ばれる、SfM であれば複数枚の画像内、visual SLAM であれば連続するフレーム画像内の特徴点の対応とそれらのカメラ位置の最適化によって3次元復元が行われていたが、近年はCNNの発展によって単一の画像について depth-map 表現を深層ニューラルネットの出力として近似的に得ることで、多くの画像間の特徴点について最適化をすることなく最低

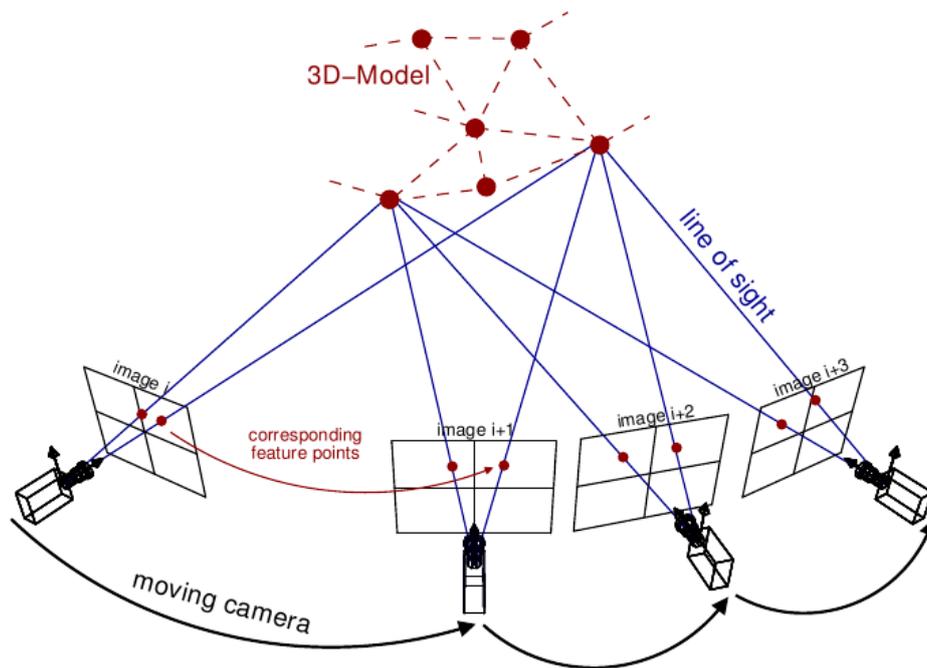


図 2.5: **Bundle Adjustment** の概要. 画像上の特徴点の座標に基づき, カメラの位置, 向き, 三次元空間上の特徴点の位置を最適化する. カメラのパラメータや (図は [2] より引用)

限の画像からシーンを 3 次元空間上に復元する事も可能となってきている [34] [35] [36].

Movie Map の構築において, 撮影した映像のどの部分が地図上のどこに対応しているのかを知ることが出来れば映像同士の交差するフレームを特定することが出来るためこれらの技術はとても有用である一方, 別々に撮影された映像群全体に対して共通の 3 次元空間を与え最適化を行う事は非常に長い計算時間を要し広い地域について Movie Map を構築する事を考えると現実的ではない. 提案手法においては, 撮影した 360°映像それぞれについて独立に visual SLAM を適用する事でその映像内に置ける相対的なカメラ姿勢の変化を推定し, 映像に不可情報として与えた参照座標を基に全てのカメラの軌跡を同一空間上にマッピングする事で映像間の位置関係を明らかにしている. その際, ソフトウェアとしてオープンソースとして公開されている OpenVSLAM [37] を利用している. OpenVSLAM は入力として 360°映像を受け付けており, パースペクティブ画像を用いた時よりも正確な推定を可能にしている.

## 第 3 章

# 提案手法

本章では、既存のシステムと比較して撮影、構築のコストを大幅に低減した新たな Movie Map 構築システムとそれを利用した各種 Movie Map インターフェースの紹介を行う。

360°映像の撮影からインターフェース上での利用までのデータの入出力関係を表現したフレームワーク全体のフローチャートを図 3.1 に、示す。図中の黄色で示された部分が処理の内容を、青色で示された部分がシステムが入出力とするデータを表現している。以下の節では、このそれぞれの処理内容についての提案手法について詳しく述べる。

### 3.1 360°映像の収集

まず始めに、Movie Map を構築する対象地域について 360°映像の撮影を行う。図 3.1 においては、Data Acquisition と表された部分と対応する。事前準備として、図 3.2 に示すような撮影計画地図（このような地図を、今後ルートマップと呼称する）を設計する。ルートマップには Movie Map 上で採用したいストリートについて、その端点に図中では数字で表されている固有の ID を設定する。撮影の際には 1 つのストリートについて、片方の端点からもう片方の端点に向かう 360°映像を双方向に撮影を行う。この際、どの地点で他の映像と交差するかについてや、移動速度の統一については考える必要が無く、既存のシステムにおける撮影作業と比較して格段に撮影者の負担が少ない。

撮影を行う際、あるいは各ストリアートの端点の共通の座標情報が地図上などで取得できる場合には全ての撮影を行った後に、それぞれのストリアートの端点の座標情報を取得しておく必要がある。

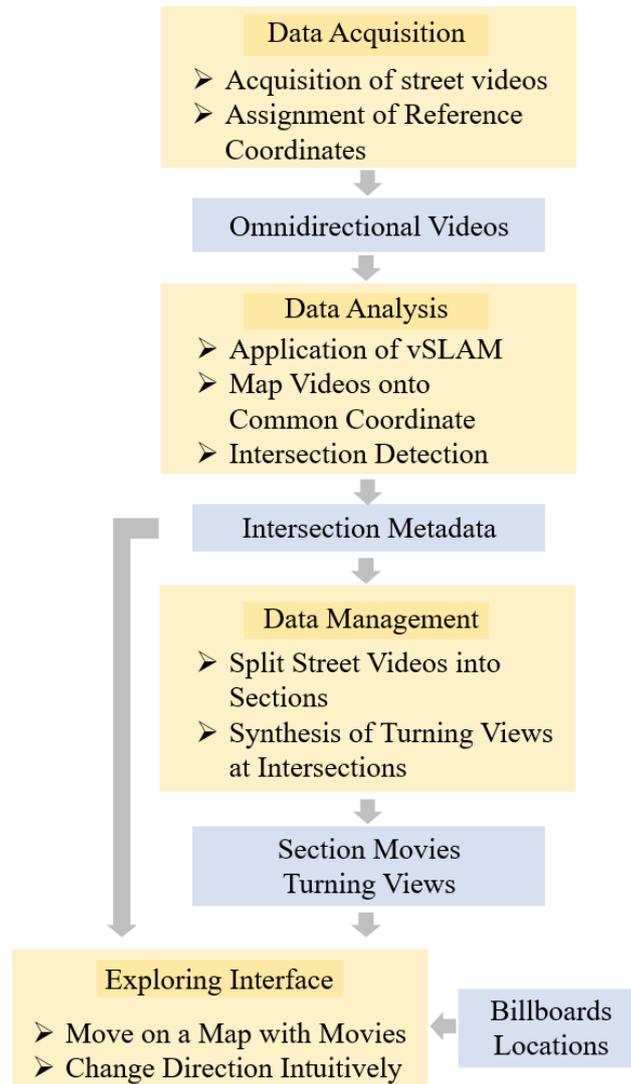


図 3.1: **Movie Map** システムフローチャート. 各映像について撮影された 360°映像は自動化された図中に緑色で示された **Movie Map** システムの入力となり, システムは交差点を基準とした映像群の位置関係情報と分割された映像, そして交差点における合成遷移映像を出力する. それらの情報は **Movie Map** インターフェースによって利用され, 一般のあらゆる地域について **Movie Map** が実現される.

## 3.2 交差点の検出と校正

撮影した 360°映像群と各ストリートの端点の参照座標を入力として, 映像間の黄砂情報の検出を行う. その手順は以下の通りである.

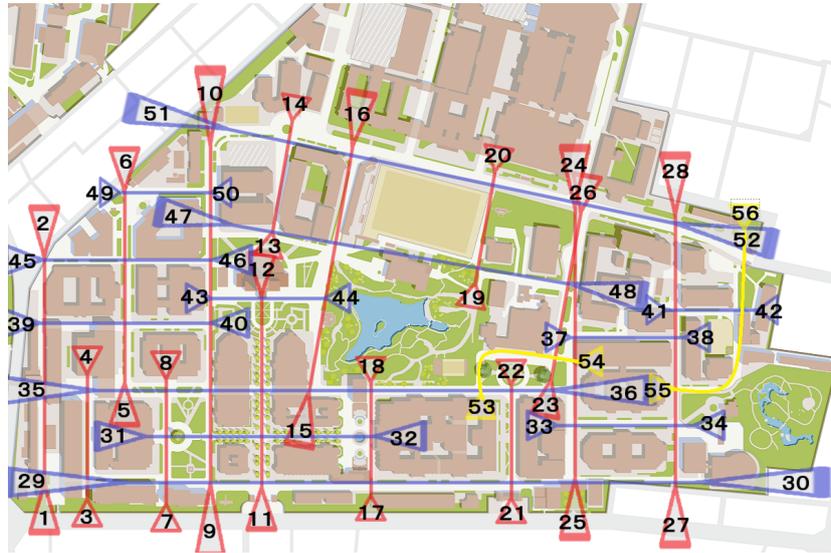


図 3.2: 東京大学本郷キャンパスにおける Movie Map ルートマップ。東京大学本郷キャンパス構内における主要な経路を 27 本設定し、それらの端点に数字の ID を付けたルートマップ。撮影の際にはこれらの経路に従い、1 つの経路について双方向に進行する映像を収集する。その際撮影の移動速度や双方向に撮影する際の微小な座標のズレは許容される。

### 3.2.1 vSLAM の適用

まず、撮影した映像それぞれについて独立に、vSLAM を適用し映像内での相対的なカメラ姿勢の変化を推定する。提案手法においては vSLAM のアプリケーションとして、OpenVSLAM [37] を使用する。OpenVSLAM は従来から広く一般に用いられていた、ORB 特徴量を用いた vSLAM アプリケーションである ORB-SLAM [38] を使いやすいようにオープンソース化したソフトウェアであり、360°画像を入力として受け付ける事が可能である。

vSLAM において、1 枚の画像から得られる特徴点の数はその最適化精度に大きな影響を与えるが、一般に屋外において正面を撮影した映像を vSLAM に適用した場合、その進行方向については安定した構造情報が少ないために特徴点が多く捉えられない事が多く、また時間と共に変化する風景の割合が屋内の環境と比較して高いために、vSLAM の動作が不安定になりがちである [39]。しかし 360°画像においては、進行方向の左右に当たる方向には建物や壁など時間によって変化しない構造情報が含まれている事が多

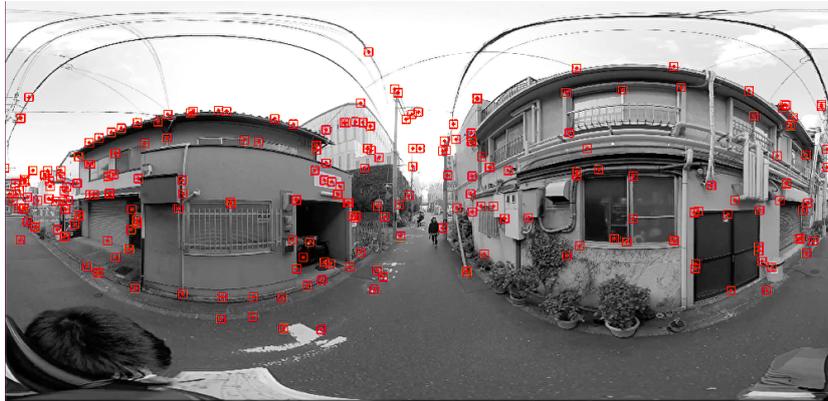


図 3.3: OpenVSLAM における 360°画像からの ORB 特徴量抽出の例. 図中の赤い点で示された部分が抽出された特徴点である. 進行方向から見て左右方向の, 建物上に多くの特徴点が検出されていることが見て取れる.

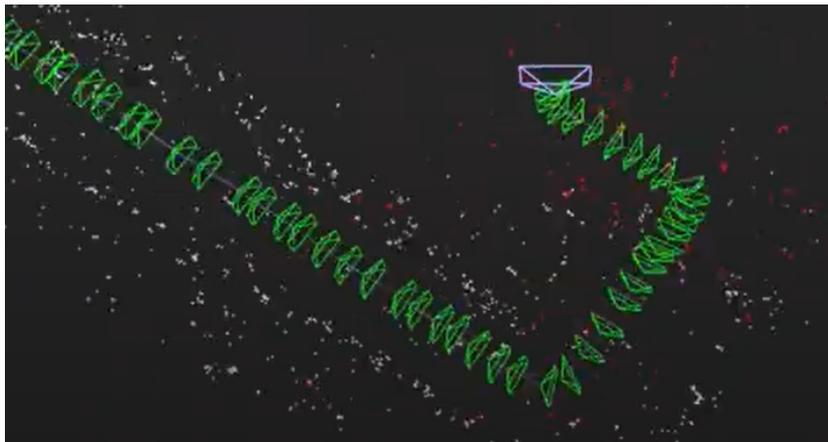


図 3.4: OpenVSLAM により推定された, 単一映像における三次元空間上の特徴点とカメラ軌跡. 緑色に示されているのが映像内の相対的なカメラ位置と向き軌跡である

く, 特徴点の検出がパースペクティブ画像を用いるよりも遥かに多く成功する. 図 3.3 に OpenVSLAM を用いて 1 枚の 360°画像から画像特徴量を抽出した例を示す. 正面方向にはまっすぐ伸びる道と空しか映っていないため, 正面を撮影したパースペクティブ画像を入力として vSLAM を適用しようとした場合その動作は不安定になる危険がある. 360°画像を入力とすることで, その精度は大きな向上が見込める.

### 3.2.2 参照座標を用いた共通座標へのマッピング

vSLAMによってそれぞれの映像について独立にカメラ位置の軌跡を推定した後は、それらを共通の座標空間上にマッピングする。そのために、撮影時に取得したストリーットの端点の座標情報を用いる。図 3.4 に示すように OpenVSLAM の出力するカメラ位置の軌跡は 3 次元のものであり、またそのスケールが各映像間で共通している保証はない。そこで、映像全体を通してのカメラ位置の移動ベクトルがストリーットの端点の参照座標同士を結ぶベクトルに合致するように、推定されたカメラ軌跡全体に回転とスケーリングを施す。この処理を全ての映像群に対して行う事により、全ての映像のカメラ軌跡が図 3.5 に示すように共通の地図座標空間上にマッピングされる。

注意すべきなのは、この共通の座標系は必ずしもグローバルな地図座標である必要は無く、例えば図 3.2 のルートマップに用いられているような縮尺が正確ではない地域の概略図におけるピクセル位置としての座標でも構わないという点である。仮に地図座標上で正確な参照座標を入力したとしても vSLAM の誤差と参照座標のエラーによりマッピングの時点で軌跡の交差と実際の映像上での交差点が正確に合致する事は期待できないため、この時点で必要とするのはマッピングの結果交差している映像の軌跡のペアが実際に交差しているフレームの近くで交差する事としている。なおここまでの処理において大きく処理時間を要するのは vSLAM の適用であり、360°映像への vSLAM の適用はその解像度が 1440\*720 である場合 1 分の映像に対して 75 秒ほどの処理時間がかかる\*1ものの、Movie Map の構築はユーザーがインターフェースとしての Movie Map に触れる前に行われるため現実的に実行可能なスケールであればその処理時間には要求が無く、また以上のマッピングまではそれぞれの映像について独立に行うため、対象地域が拡大した場合にも並列に処理を行う事で処理時間がスケールする事は無い。

### 3.2.3 座標情報に基づく交差点の検出

全てのカメラ軌跡が共通座標空間上へマッピングされた後、そのカメラ位置を用いて交差点の検出を行う。図 3.6 に 1 つの映像ペアにおける交差点検出の様子を示す。まずカメラ軌跡をある程度の範囲で区切り、区切られた一範囲の位置座標を全て含む大きさの矩形を考える。その後、軌跡ペア間で重なりが生じる矩形のペアを検索する。検索がヒットした場合、その矩形内を更に細かな範囲に分割し、再度矩形を定義、重なるペアを検索し

---

\*1 Intel Core i7-8700K 3.70GHz プロセッサを Ubuntu16.04.6 で用いた場合

ていく。最終的に矩形に含まれるフレーム数が限られたものになった後に、その範囲内に含まれる全てのフレームの組み合わせについて座標が最も近いフレームのペアをその経路ペアの交差点として検出する。

この際、2つの経路は1点でのみ交差している事を前提とする。ルートマップを計画する際にある経路のペアが2箇所以上の点で交差してしまっている場合には、どちらかの経路を2つに分割し、前提条件を満たすようルートマップを変更する事によって対応する。提案する検出方法においては経路の交差する角度に制限は無いため、1つの経路を2つに分割した場合は検出の際にそれぞれの経路の端点で交差が検出されることとなる。

経路の端点を含む矩形は、経路の開始フレーム、終点フレームの位置座標をその1つ後、あるいは前のキーフレームの位置座標と逆の方向にある程度延長する事で、T字路などにおける検出漏れの発生を防ぐ。ただしこの延長の距離は長すぎる場合本来は交差しない経路のペアについて交差点が検出されてしまう恐れがあるため、映像の撮影と参照座標の付与の段階でT字路などにおいてはしっかりと経路が交差するように留意する事が望ましい。

### 3.2.4 特徴量マッチングを用いた交差点フレームの校正

前述の交差点検出によって推定された交差点フレームは、次の理由により少なからず理想的なフレームから誤差を含む可能性が高い。

- 映像に付与した参照座標の実際の映像とのズレによる影響
- vSLAMのスケールドリフト [40] による蓄積誤差の影響
- 映像の交差が長時間に渡ることによる、映像上の交差の中心と座標上の最近傍フレームとのズレによる影響

そのため、交差点の検出後に正確なフレームを選定するために特徴点マッチングを用いた校正作業を行う。

図 3.7 に校正作業の概要を示す。交差点検出によって交差点フレームが推定された後、その周辺の十数フレームを正確交差点フレームの候補フレームとしてピックアップする。検出された片方の交差点フレームの角度方向に選び出されたすべての候補画像を回転させたうえで、経路間の全てのフレームのペアについて ORB 特徴量 [41] のマッチングを行い、マッチングの取れた特徴点の数の多さを画像同士の距離の近さとして最も距離の近い画像のペアを正確な交差点フレームとする。

### 3.3 交差点情報を用いた映像データベースの構築

撮影した映像群全体に対して交差点の検出と校正を行った後、それらの交差点情報に基づいて Movie Map インターフェースで用いる映像データベースの構築を行う。具体的には、収集した各映像を交差点を基準とした分割映像群とし、各交差点についてペアを成す映像間でそれぞれの方向に曲がる遷移映像の合成が必要となる。

#### 3.3.1 交差点を基準とした映像分割

各映像を交差点を基準とした分割映像群としてデータベースに保存しておくことで、インターフェース上での簡便な取り扱いと再生場所を移動するのに比べて比較的高速でなめらかな再生映像の切り替えを可能にする。この際、1つの身体的な交差点について、各経路映像は双方向に撮影されているため図 3.8 に示すように4つの映像の交差が生じる。よって通常の場合、とある経路を左右方向に横切る経路映像は映像上でそれぞれ少しズレた位置を横切るため、交差点の内部に短い分割映像が生成されることとなる。

#### 3.3.2 交差点におけるターン映像の合成

Movie Map インターフェース上において、交差点を曲がる、すなわちストリート映像の切り替えを行う際、実際に視点を回転させて移動するようなターン映像の挿入が必要となる。既存のシステムではこれらの遷移映像は全てのパターンを別々に撮影していたものの、これにはとても大きい撮影コストがかかってしまう。そこで、撮影されている映像が360°映像であることを活用し、撮影されたストリート映像から交差点を曲がっているかのような遷移映像を合成する事を考える。

具体的には、正確な交差点フレームとして推定された切り替え前の映像の交差点フレームと、切り替え後の映像の交差点フレームを取り出し、vSLAMの推定した角度情報を基に切り替え前の方向に回転させた上で重ね合わせ、そのブレンド比率を変えながら切り替え後の映像の角度へと回転をさせるというものである。

詳細を式 3.1 に示す。

$$F(i) = \left(1 - \frac{i}{N}\right) \cdot \text{Rotate}(I_{\text{before}}, \frac{i}{N}q) + \frac{i}{N} \cdot \text{Rotate}(I_{\text{after}}, (1 - \frac{i}{N})q^{-1}) \quad (3.1)$$

式 1 において、 $F(i)$  が合成映像の  $i$  フレームを、 $I$  が切り替え前後の交差点フレーム画像を表し、 $q$  は 2 画像間の相対的な回転角である。  $N$  は合成映像の総フレーム数であり、これにより切り替え前の画像が徐々に回転しながら切り替え後の画像に切り替わっていく映像が合成される。

遷移映像の合成例を、図 3.9 に示す。 遷移映像の再生中は、重ね合わせた 2 舞の画像が同方向に回転していくため、あたかも自分が交差点において視点を回転しているかのような映像を合成する事が可能である。 また、この手法による合成映像は切り替え前の映像の交差点フレームで始まり切り替え後の映像の交差点フレームで終了するため、交差点におけるターンにおいて対象の切り替え前後の分割映像の切り替え時にこの合成映像を挿入する事により途切れることの無い連続的な映像の切り替えが可能になる。

交差点のターン補間映像の合成手法については他にもいくつかの手法が考えられる。 それらと比較した際の提案手法の優位性については後述の、実際にそれらの映像を挿入した際のストリート映像の切り替えが自然かどうかという主観評価実験によって示す。

### 3.4 映像データベースと交差点情報を用いた Movie Map インターフェース

以上の自動化された Movie Map の構築手法によって、撮影した 360°映像群と参照座標情報の入力から、分割、合成映像からなる映像データベースと映像の交差情報を得ることが出来た。 これらの出力を用いて、道案内、バーチャル探訪それぞれに特化した web ブラウザ上で動作する Movie Map インターフェースの実装を行った。 どちらのインターフェースにおいても、再生中の 360°映像についてはマウスの操作と UI ボタンの入力によってあたりを見回したり再生場所、再生速度の変更を行うことが可能となっている。

#### 3.4.1 道案内に特化した自由経路合成インターフェース

地域において目的地となるランドマークに向かうさいのナビゲーションとして地図アプリケーションを用いる際には、自分の現在位置や駅などのスタート地点から目的地まで移動する映像の連続的な再生がとても有用である。 図 3.10 に示すインターフェースはその機能を実現したものであり、ユーザーは以下の操作によって目的地までの映像と共に移動することが出来る。

まず、インターフェースの地図画面において紫色のサークルで表された交差点を選択すると、隣接する交差点へと伸びる矢印の UI が現れる。 矢印を選択して交差点同士をつな

いしていくと、それらのサークルが黒い線で結ばれ1つの経路が作成される。この機能により、ユーザーはスタート地点から目的地までの自由経路を作成する。経路の作成が完了した後、映像の合成ボタンを押すことで映像データベースが参照され、必要な部分映像と遷移映像を繋ぎ合わせることで入力した道順に対応した自由経路映像が合成される。その後図 3.10(b) に示す再生画面がポップアップし、合成された 360°映像が周囲の見回しや基本的な映像再生機能の実行が可能な状態で再生される。再生中の映像の再生時間と地図画面上の経路の部分は交差点の区分単位で紐づけられており、現在どのあたりを映像が進行しているのかが把握できるようになっている。

### 3.4.2 バーチャル探訪に特化した Movie Map 探索インターフェース

対象地域をバーチャル探訪などの目的で探索したい場合には、道を進みながら、分かれ道が現れた場合に進行方向を決定し連続的にストリート映像を再生していくことが有用となる。図 3.11 に示すインターフェースはその機能を実現した者であり、ユーザーは以下の操作によって現在の自分の位置に対応した映像を再生しながら地域内を自由に散策することが出来る。

まず、道案内型のインターフェースと同様に、探索の開始地点となる交差点と進行方向を決定する。すると、再生画面で選択した領域のストリート分割映像の再生が開始し、地図上の赤い矢印で現在の位置と見ている方向を表したカーソルが移動し始める。ユーザーが交差点から一定の距離に侵入すると、直後の交差点において進行可能な方向に応じて、図 3.12 中にあるような赤い矢印が再生画面中にポップアップする。交差点の中央に到達する前に矢印を選択しておくことで、分割映像の再生が終了し次第その方向へターンを行う合成遷移映像が再生され、次なるストリート分割映像の再生が開始する。進行方向の決定が交差点侵入までになされていない場合は映像の再生が一時停止するが、設定としてどの方向も設定していない場合にはランダムに進行方向を決定し、放置する事で対象地域を無作為に歩き回るように動作させることも可能になっている。結果としてオリジナルの経路を進む映像が連続的に再生されるため、ユーザーはあたかも実際に自分がその地域を歩いているかのような感覚を得ることが出来る。

このインターフェースはその基本的な機能に加えて、地図画面の紫色で表された交差点にアクセスする事で適時離散的な移動を可能としているほか、場所、画像、情報を入力する事でインターフェース上に図 3.13 に示すようなバーチャル看板を表示する機能を備える。データとして入力する場合にはどのストリート映像上に表示するのか、道のどちら側、あるいは正面、背後方向に表示するのか、どんなテクスチャを持つのか等を詳細に入

力する必要があるが、インターフェース上で看板を設置したい地点を視点の中心にとらえ「広告を追加」ボタンを押し、現れるファイル選択 UI において画像を一枚選択する事でその画像をテクスチャとするバーチャル看板の設置が簡単に完了する。またその際表示される文字列入力欄に URL を記述しておくことで、インターフェース上でその看板をクリックした場合その web サイトにアクセスするように設定することが出来る。

これらのインターフェースは Movie Map 構築システムが出力した交差点情報と映像データベースを入力とすることで様々な地域について適用することが可能で、地域に対して特別な処理を行う必要は無い。

### 3.5 深層学習を用いた単一映像における交差点予測

以上が Movie Map の構築とその映像データベースを利用したインターフェースの全容である。提案手法は一般の地域に敵よ可能であり、連続した映像を伴う経路移動体験をユーザーに提供することが出来る。

しかし一方で本手法は、その実行性を vSLAM に依存している点、撮影するストリートそれぞれについて参照座標を与えなくてはならない点でコストと汎用性について課題が残っている。参照座標の付与は人手で入力を行う上数値は目視によって間違いが検出しにくいいため、その数が多いと致命的なエラーが含まれてしまう事が多々あり、それが発覚するたびにエラーの修正と Movie Map 構築システムのやり直しが必要になってしまう。また OpenVSLAM による vSLAM の適用は、多くの場合で成功するものの、繰り返し構造の多い地下通路や揺れ動く木々に囲まれた空間など、自己位置をロストしたり現在位置を以前の位置に誤認してしまう事例が発生してしまうことも事実である。

そこで、撮影した映像群から交差点を検出する手法として、vSLAM の適用と参照座標による共通空間へのマッピングに代わる新たな手法を提案する。それが、深層学習を用いた交差点予測である。

図 3.14 にその概略を示す。どの経路同士が交差するかという情報はルートマップ情報から取得できる既知のものとして、それぞれの映像についてどの部分が交差点かを学習モデルによって予測を行う。その後、閾値判定によって 1 つの範囲として検出された交差点らしいパートの映像間のペアを交差点の候補として、全ての交差点候補について前述の校正作業を行う。そして、全ての項補間で最も近いフレーム画像のペアをその経路ペアの交差点フレームとする。

この際、映像から交差点の候補となる範囲を選び出すためのフレーム画像の交差点としての尤度予測に用いるのが深層学習による予測器である。映像内各フレームの 360°画像

について交差点かどうかを予測器により判定する。 naïveな手法として、交差点と非交差点の 360°交差点画像からなるデータセットを用意し、教師あり学習によって直接交差点かどうかの判定をネットワークに出力させる方法が考えられるが、交差点という概念は非常に高次元であり、その定義や基準は地域や場合によって曖昧である。そのため、ネットワークに学習させる入出力関係は直接交差点かどうかを判定するのではなく、その基準となる画像特徴的な要素であることが望ましい。

そこで提案する手法は、ネットワークには 360°画像の正面に消失点が存在するかどうかを判別する学習をさせ、推論時には入力画像を回転させつつネットワークを通すことでどの方向に消失点が存在するかを予測し、その結果を用いてヒューリスティックな関数により画像が交差点かどうか判定するというものである。

一般に交差点においては 3 つ以上の方向に無限遠点が存在するため、実際の推論時には 8 方向に画像を回転させ正面方向の消失点判定を行い、3 方向以上で消失点が検出された場合にその画像を交差点として判定する。

ただし、以上の単一映像からの交差点部分の抽出については未だ前段階である 360°画像の交差点判定の精度を高めている段階であり、実際の Movie Map の構築においては未実装となっている。

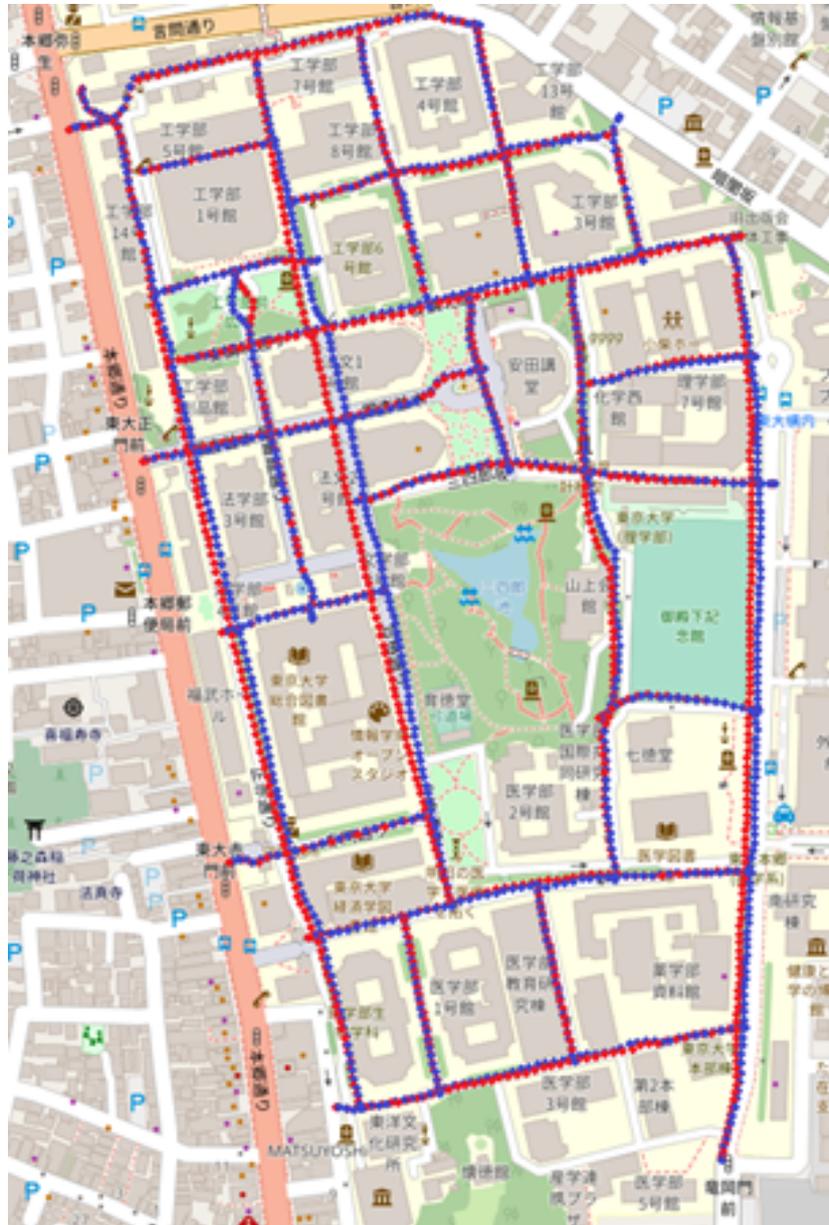


図 3.5: 参照座標をアンカーとした東京大学本郷キャンパス構内における地図上へのマッピング結果. 赤いマーカーが順方向の映像におけるキーフレームのカメラ位置を参照座標を基に回転, スケーリングさせることにより 2 次元地図上に投影したもので, 青いマーカーは逆方向の映像における投影点である.

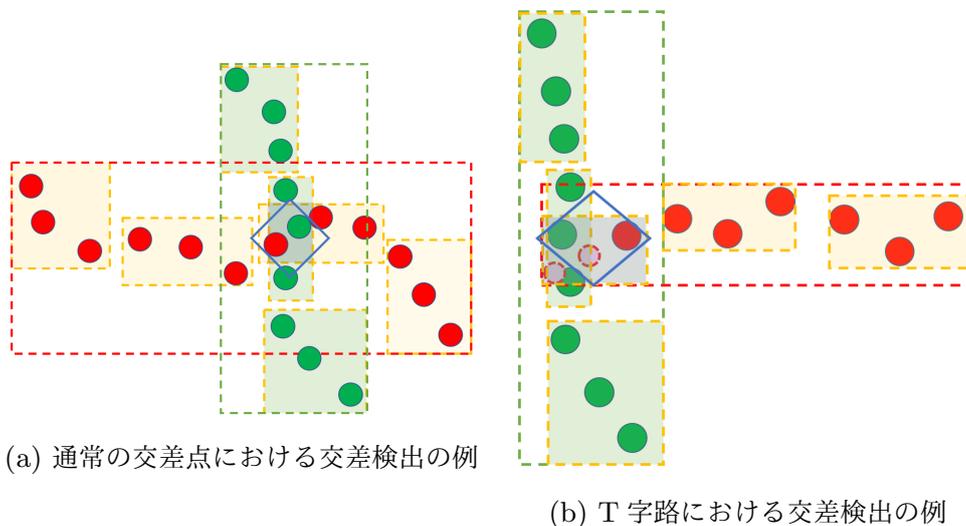


図 3.6: カメラ軌跡のペアについての交差点検出の例. 図中に赤と緑のサークルで表されているのが各映像におけるキーフレームのカメラ位置座標である. 軌跡の一部を矩形で囲み, それらの重なりを検出された場合その矩形内にさらに小さな領域の矩形を作成し重なりを検証する. 経路の端点においては採集フレームの位置座標を延長する事で, T字路などにおける検出漏れを防いでいる.

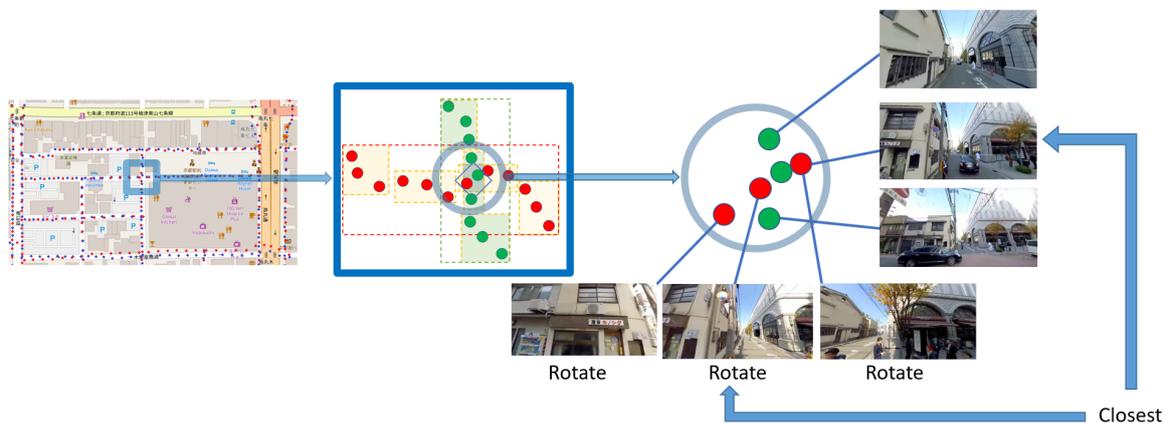


図 3.7: 交差点フレーム校正の概略図. 検出された交差点フレームの周辺フレーム画像を選び出し, vSLAM の推定したカメラ姿勢を基に全ての画像を同じ向きに回転させる. その後, ORB 特徴点のマッチングを行い最も近い画像のペアを選び出し, そのペアを正確な交差点フレームとする. 図中においては, 緑色のマーカーで表された経路のうち交差点フレームとして検出されたフレームの角度方向に全ての候補画像を回転させている.

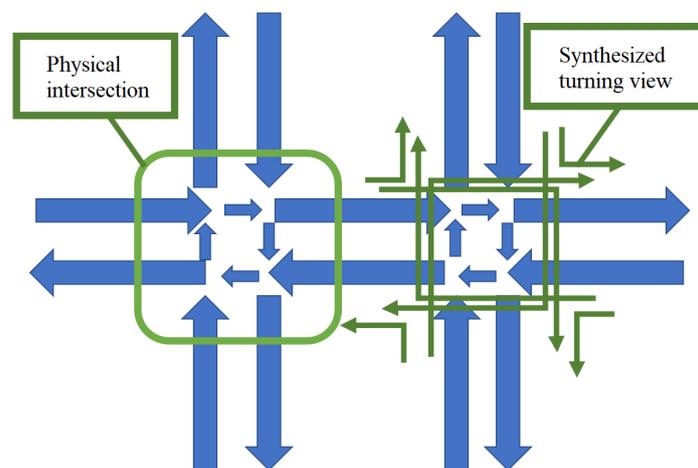


図 3.8: 交差点における分割映像, 合成映像の管理. 経路映像は双方向に撮影しているため, 身体的な 1 つの交差点に対して検出される交差点は 4 つとなる. なお同経路を逆方向に進む映像同士とは, 交差が検出されない例外処理が施されている. 遷移映像の合成については, これらの映像間を遷移する全パターンの映像を合成する必要がある.

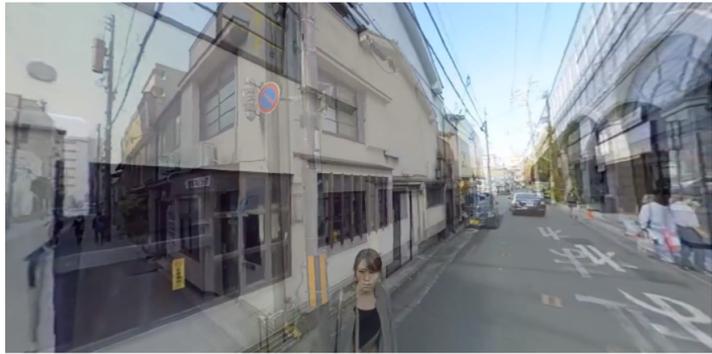
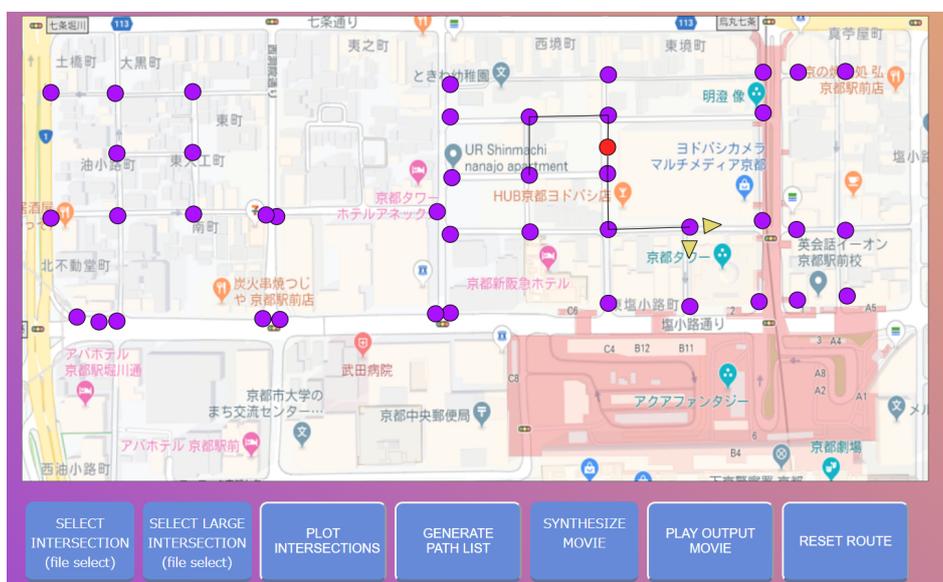


図 3.9: 交差点における合成された遷移映像の例。2つの360°画像を重ね合わせ回転させながらその比率を変更していく手法を採用している。映像上では比較的自然な移り変わりが表現可能である。



(a) 道案内 Movie Map インターフェース・地図画面



(b) 道案内 Movie Map インターフェース・再生画面

図 3.10: 道案内に特化した **Movie Map** インターフェースの概要. (a) に示す操作画面において、紫のサークルで表された交差点を選択していくことで自由な経路を決定する. そして経路映像の合成を行うと映像の再生画面が現れ、作成した経路に沿って移動する映像が再生される. 映像中の現在位置は地図画面に赤いサークルで表示される.

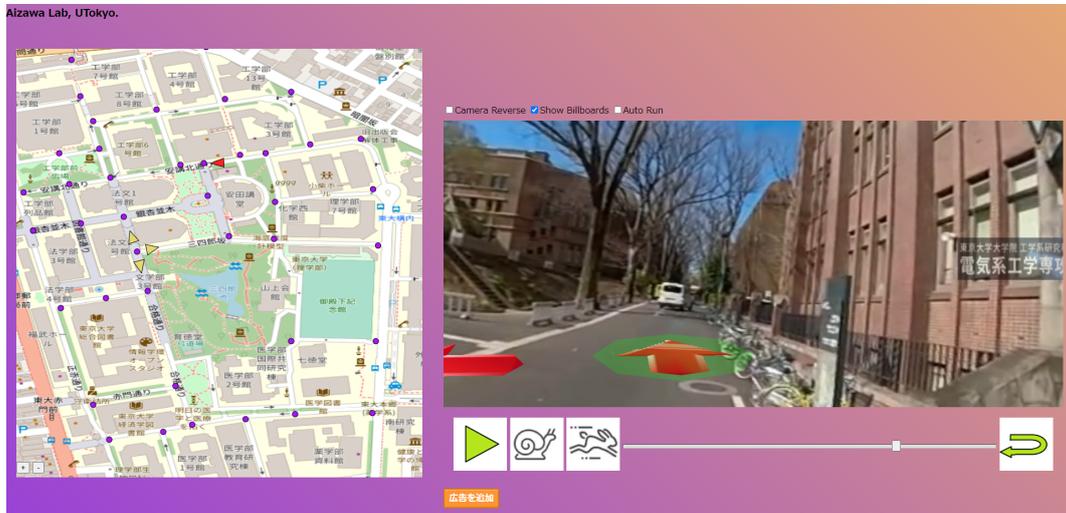


図 3.11: 探索型 Movie Map インタフェースの様子. 2つの 360°地図上でスタート地点と進行方向を定めることで, その方向に進む映像の再生が開始する. 再生している映像の再生時間と向いている方向に応じて地図画面の赤い矢印が移動・回転する.



図 3.12: 探索型 Movie Map インタフェースの交差点での再生画面の遷移. 交差点に侵入すると再生画面上に, 進める方向に応じた赤い矢印の UI がポップアップする. 矢印を選択することで進行方向を決定し, ターンを行う場合は合成映像を挿入した上で再生映像の切り替えを行う.



(a) バーチャル看板・遠景

(b) バーチャル看板・近景

図 3.13: 設置したバーチャル看板の例. バーチャル看板はインターフェース上で映像を再生, 閲覧しながら設置する事も, データとして事前書き込むことでインターフェース上に正確に設置する事も可能である. クリックする事で紐づけられた web サイトにアクセスする事も出来る.

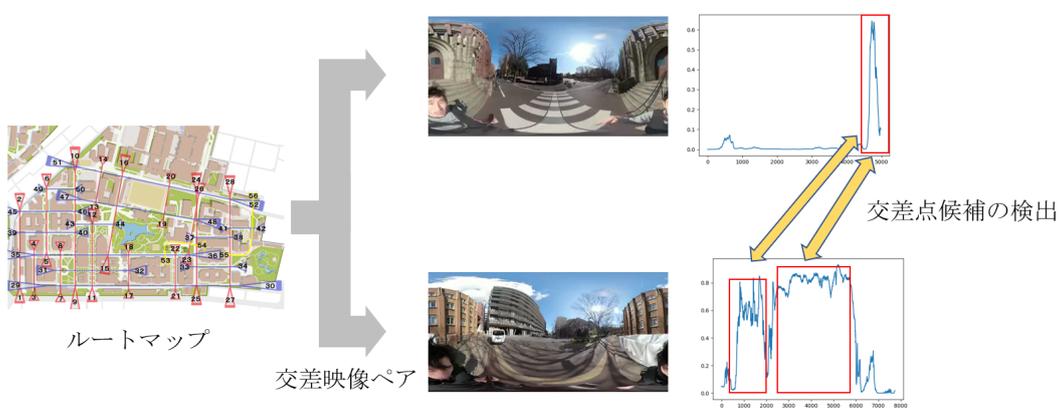


図 3.14: 単一映像からの交差判定を利用した交差点範囲の絞りこみ. ルートマップ上で交差する2つの経路について, その映像内の各フレーム画像について交差点かどうか判定を行う. 図中のグラフの横軸は映像の再生時間, 縦軸は交差点の尤度を表しており, 閾値を超えつづけた1範囲を1つの交差点候補とする. そして候補同士のペア全てについて校正作業を行い, 全体を通じて最高のスコアを出したフレームのペアを交差点として検出する.

## 第4章

# 実験

本章では、提案する合成遷移映像が他に考えられる遷移映像の合成方法と比較して自然に感じられるかどうかの主観評価実験、探検型 Movie Map の GSV と比較した場合のユーザビリティについての主観評価実験、そして新たに提案している交差点予測について学習する入出力関係を変更した場合の精度比較実験について記述する。

### 4.1 合成遷移映像についての主観評価実験

提案する合成遷移映像の交差点における映像切り替え時の挿入について、撮影者の位置、風景の変化の連続性が保たれているか、そしてどの方向に曲がったかが正しく認識できるかをユーザスタディによって確かめた。実験の対象となったのは16人の研究内容を知らない東京大学の学生である。被験者は図4.1に示す6通りの交差点における映像の切り替えを好きな回数鑑賞する。交差点は2箇所あり、それぞれに映像のパターンが3通りある。それぞれ交差点フレームにおいて直接映像の接続を行うパターン、片方のフレーム画像を回転させる映像を挿入するパターン、そして提案する合成遷移映像を挿入するパターンである。質問する内容として、撮影者の位置、周りの風景の変化が連続的、かつ自然に感じたかどうか、そして曲がった方向が正しく認識できたかといった項目を用意し、5をポジティブ、1をネガティブとする5段階評価で回答してもらった。

その結果を表4.1に示す。

どちらの交差点についても、撮影者の位置の変化の連続性と自然さ、曲がった方向の認識のしやすさについては合成遷移映像を挿入する手法が高い評価を得られた。一方周囲の風景の変化についての連続性、自然さについての評価は、交差点2では提案手法が最も高い評価を得られた一方、交差点1では片方の交差点フレームを回転させた映像を挿入した

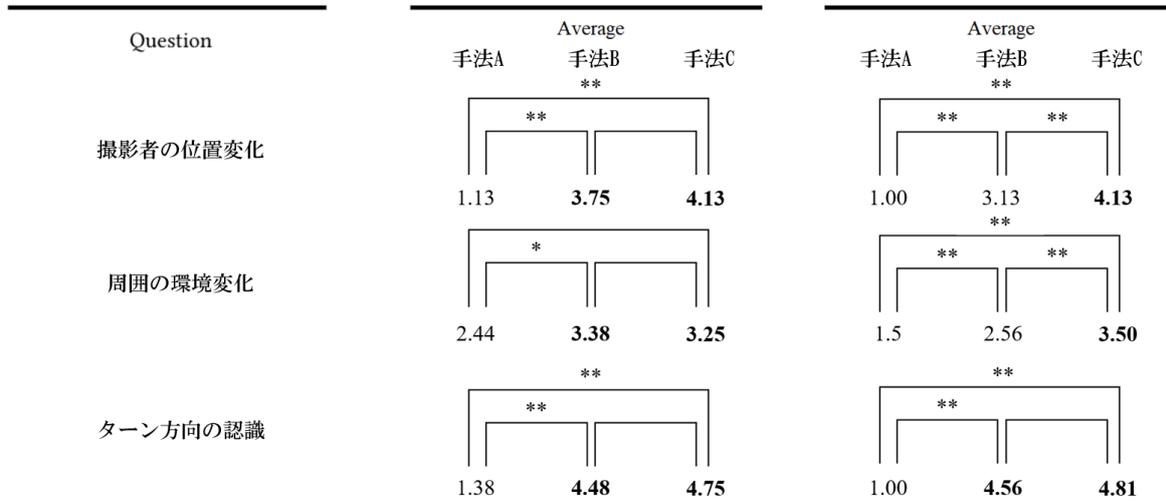


図 4.1: 各交差点における映像切り替えの様子. 2つの交差点において, 直接交差点フレームで映像を切り替えた場合(手法 A), 片方の端点フレーム画像を回転させるだけの映像を挿入した場合(手法 B), 提案する合成遷移映像を挿入した場合(手法 C)の映像. 交差点 2 は交差点 1 と比較してオブジェクトや風景の様子が切り替え前後で大きく変化している.

方が高い評価を得られている.

これは, 交差点 1 の切り替え前後の映像に辺りの明るさやオブジェクトの有無といった風景の変化が少ないために, 片方の交差点フレーム画像を回転させた画像ともう片方の交差点フレーム画像の間に大きな違いが無くそれらを繋げた際の不連続性よりも 2 枚のフレーム画像をブレンディングする事による合成感が自然さの点で評価を下げたことが原因だと考えられる.

よって実際の Movie Map の構築の際には, 推定された交差点の最も近いフレーム画像



(a) 交差点 1

(b) 交差点 2

\* :  $p < 0.05$ , \*\* :  $p < 0.01$

表 4.1: 合成遷移映像についての主観評価実験の結果.

のペアについて、その近さに応じた補間映像の生成方法の場合分けをする事が望ましいと言える。すなわち、ペア画像が非常に近い場合は合成をすることなく片方のフレーム画像を回転させた映像を挿入し、そうでない場合は提案する合成遷移映像を挿入する事によってそれぞれの場合に応じてより自然な経路映像が作成できると考えられる。

## 4.2 Movie Map インターフェースのユーザビリティについての主観評価実験

京都駅周辺のエリアについて Movie Map を構築し、そのユーザビリティについて GSV との比較実験を行った。図 4.2 に示す赤枠内のエリアについて、GSV、Movie Map それぞれにおいて 1 枚の画像で表されたランドマークを探索するタスクを行い、その成否に関わらず双方のインターフェースを用いたときの使いやすさ、没入感の 2 つの観点において 5 をポジティブ、1 をネガティブとした 5 段階の主観評価を行ってもらった。被験者は合成遷移映像の評価と同一の 16 人の学生である。

探索は図 4.2 に示す京都駅周辺のエリアで行った。探索の条件を揃えるため、GSV の利用時には操作を再生している画像上に表示されている UI をクリックする事による移動と周囲の見回しのみを許容し、Movie Map の利用時には再生中の映像上に表示される交



図 4.2: 実験タスクにおける探索エリアの図。赤枠で囲まれた 2 つの範囲を探索対象エリアとする。被験者はそれぞれ片方のエリアを Google Street View を用いて、もう片方のエリアを Movie Map を用いて探索を行う。

Question	Average	
	GSV	提案手法
ユーザビリティ	3.81	3.44
探検の没入感	3.19	<b>4.38</b>

\*\* :  $p < 0.01$

表 4.2: Movie Map と GSV の比較主観評価実験の結果。

差点における矢印の選択と周囲の見回し，再生速度や再生向きの変更のみを許容した。

その結果を表 4.2 に示す。

没入感の観点で，Movie Map が GSV を上回る評価を得た。

一方使いやすさの観点で Movie Map が高い評価を得られなかった理由として，Movie Map において気になった点を詳細に確認するためには再生中の映像を停止して視点を合わせるという 2 つのステップを踏む必要があり，GSV においては気になった点の確認にすぐに移れるという違いが原因として考えられる。これが優位性として表れるのは，映像よりも静止画像をベースとしたインターフェースの方が今回のタスクを達成しようとしたときに適しているからである。そのため，他のタスク，例えば目的地までの道順を確認するであったり，対象地域の概要を把握するといったより地域の探索や情報の連続性に重きを置いたタスクにおいてはこの評価は覆る可能性がある。

## 4.3 深層学習による交差点予測の精度比較実験

新たな提案手法として提示した単一映像からの交差点となる範囲の予測について、その前段階となる 360°画像の交差点判定についてその学習手法による精度の比較を行った。

本実験においては、以下の 3 つの入出力関係の学習を行った。

### 4.3.1 Direct Intersection Prediction (DIP)

360°画像と、その画像が交差点かどうかのラベルをデータセットとし直接交差点判定を学習する手法。本手法のデータセットとして、Movie Map 構築に利用した東京大学本郷キャンパス、大阪難波町の映像群についてそのフレーム画像を等間隔にサンプリングし、その交差点からの距離によってラベル付けを行ったデータ群を採用した。

### 4.3.2 Movable Direction Prediction (MDP)

360°画像と、画像において前後左右斜め 8 方向について進めるかどうかのラベルをデータセットとし進める方向をネットワークに予測させ、その出力をヒューリスティックな関数に通すことで交差点判定を行う手法。データセットとして、500 枚の市街交差点 360°画像を京都市について収集し、それらについて前述のラベル付けを行った。交差点判定の関数としては、交差点においてはそうでない画像に比べ進める方向が多く、また進める方向と進めない方向がハッキリと分かれている傾向があることから、出力の合計値が 4 を超えるかどうかを  $F_1$ 、出力の分散が 0.4 を超えるかどうかを  $F_2$  とし、それぞれにおける結果を求めた。

### 4.3.3 Front Vanishing Point Prediction (FVPP)

360°画像と、その画像の正面方向に消失点が存在するかどうかの判定をネットワークに学習させ、推論時には入力画像を回転させつつネットワークを通すことでどの方向に消失点があるかを判定し、その情報をヒューリスティックな関数を通して交差点判定の基準とする手法。MDP と同じ 500 枚の 360°交差点画像に、消失点の場所をアノテーションしたものを学習データセットとする。回転は前後左右斜め方向に対応する 8 方向であり、出力に消失点が 3 つ以上含まれていれば交差点と判定するようにした。

Learn Method	Accuracy			
	DIP	Hongo Test Set		China Town Test Set
DIP (Hongo Dataset)		0.65		0.54
DIP (Namba Dataset)		0.52		0.81
DIP (Hongo and Namba Dataset)		0.58		0.65
MDP	$F_1$	$F_2$	$F_1$	$F_2$
MDP (Kyoto Dataset)	0.48	0.52	0.54	0.52
FVPP				
FVPP (Kyoto Dataset)		0.65		0.54

表 4.3: 360°画像の交差点判定における各学習手法の性能比較実験の結果 .

評価は各手法について東京大学本藤キャンパス構内, 横浜中華街の交差点, 非交差点 360°画像各種 50 枚ずつから構成されるテストセットについて, 交差点判定を行った際の精度によって行った.

その結果を表 4.3 に示す.

DIP においては, 難波町の学習データセットと横浜中華街のテストデータセットが近いドメインであるため, それぞれ同じドメインで学習した場合は高い精度を発揮している一方, 全ての学習データによって学習した場合にはどのドメインにおいてもあまり高くない精度となってしまっている.

これはネットワークが交差点の判定基準として, 横断歩道や信号機の有無, 空間の開け具合など, 地域特有の交差点に共通する特徴を学習しているために単一のドメイン内でのみ高精度の判別が可能となってしまっていることが原因と考えられる.

一方 MDP においてはテストデータセット間で精度に大きな差は見られないものの, 交差点かどうかという 2 値分類においてランダムとさして変わらない精度しか出ておらず, 交差点を判別できるほどの正確な予測が出来ていないとは言えない.

FVPP においては, MDP や, 全てのデータを用いて学習した DIP と比較して高い精度を発揮した. これは, 消失点という交差点を構成する低次元な画像特徴を学習したことにより, ドメインに依存しない交差点の判定が可能となった点, またネットワークが行う学習が正面方向の消失点の有無のみという単純タスクであった点が理由として考えられる.

とはいえ，映像内の交差点範囲を抽出する際に各フレームの交差点判定の精度が7割程度では心もとないため，実用化のためにはより高精度の予測が必要になると考えられる．

## 第5章

# 結論

### 5.1 まとめ

本稿では、既存の Movie Map システムに存在した撮影と構築のコストを低減し、あらゆる一般の地域に対して自動的に Movie Map を構築する手法の提案と、その実際のインターフェース実装の紹介を行った。

また実験においては、交差点での映像切り替えにおける合成遷移映像の挿入が、位置の変化や環境風景の変化を自然かつ連続的とする効果を持つこと、実装した Movie Map インターフェースが現在広く用いられている GSV と比較して高い没入感をユーザーに提供可能であることを示した。

加えて新たに提案する、より入力コストと対象地域の制限を低くした交差点の推定手法である映像の交差点範囲の抽出においては、その前段階である 360°画像の交差点判定について画像の正面に消失点が存在するかの 2 値分類をネットワークに学習させ入力画像を回転させて推論を行う事で 7 割を超える精度での画像の交差点判定を行う事が出来ることを示した。

今後の課題としては、まず第一に 360°画像の交差点判定精度を高めて映像内の交差点範囲の抽出、そしてその後の範囲候補ペアでの校正作業を可能とすることが挙げられる。

そしてもう一点、交差点の校正作業に当たる特徴点マッチングによる画像ペアの近さの計算部分を、候補として相応しくないと判明した場合に弾けるような計算方法に変えることが必要になると考えられる。

現時点では校正作業が必要となる交差点の周辺フレームは少ないため特徴点によるマッチングが現実的な時間で行えるものの、映像内の交差点部分の抽出により映像のうちの数百フレームがいくつかの交差点候補として抽出された場合、2 経路間のそれぞれの交差点



図 5.1: `moviemap.jp` における初期画面. 本郷キャンパス構内の Movie Map の他, いくつかの地域についても同様に Movie Map の構築を行っており, 今後公開予定となっている.

範囲の候補同士で全てのフレームペアについて同様の手法で近さを計算するのは計算時間の面で非現実的である.

そのため, フレーム単位でのマッチングの前に抽出された交差点範囲の候補同士での大まかなマッチングと切り捨てを可能とするフレームワークが必要となる. その際には, SfM や SLAM を用いた 3 次元復元技術の活用が有効であると考えられる.

## 5.2 本郷 Movie Map

本研究の成果として, 東京大学本郷キャンパス構内を対象地域として構築した Movie Map が <https://moviemap.jp> にて公開されている. 図 5.1 にそのトップページを示す. 学内限定の公開ではあるが, 図中の本郷キャンパスのボタンを選択する事で, 本郷キャンパスについての探索型 Movie Map インターフェースがブラウザ上で動作する. 現在はバーチャル看板として工学部 2 号館の電気系についてのもののみ設置されている状態であるが, 今後拡張を施しより実在感のある本郷キャンパスが Movie Map 上に再現される事が期待される.

## 参考文献

- [1] Andrew Lippman. Movie-maps: An application of the optical videodisc to computer graphics. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '80, pp. 32–42, New York, NY, USA, 1980. ACM.
- [2] C. Sweeney. Theia multiview geometry library: Tutorial & reference. <https://www.google.com/maps/>.
- [3] M. Naimark. *Field Recording Studies*. MIT Press, Cambridge, 1996.
- [4] Michael Naimark. A 3d moviemap and a 3d panorama. *Proc. SPIE*, Vol. 3012, , 01 2004.
- [5] Google. Google street view. <https://www.google.co.jp/intl/ja/streetview/>, 2005.
- [6] Hou-Ning Hu, Yen-Chen Lin, Ming-Yu Liu, Hsien-Tzu Cheng, Yung-Ju Chang, and Min Sun. Deep 360 pilot: Learning a deep agent for piloting through 360 sports videos. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1396–1405. IEEE, 2017.
- [7] Yu-Chuan Su and Kristen Grauman. Making 360deg video watchable in 2d: Learning videography for click free viewing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6476–6484, 2017.
- [8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, Vol. 25, pp. 1097–1105, 2012.
- [9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- [11] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.
- [12] Jan Hendrik Hosang, Rodrigo Benenson, and Bernt Schiele. How good are detection proposals, really? *CoRR*, Vol. abs/1406.6962, , 2014.

- [13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, Vol. 25, pp. 1097–1105, 2012.
- [14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.
- [15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.
- [16] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10687–10698, 2020.
- [17] Huikai Wu, Junge Zhang, Kaiqi Huang, Kongming Liang, and Yizhou Yu. Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation. *arXiv preprint arXiv:1903.11816*, 2019.
- [18] Chuhan Zou, Alex Colburn, Qi Shan, and Derek Hoiem. Layoutnet: Reconstructing the 3d room layout from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2051–2059, 2018.
- [19] James McCrae, Igor Mordatch, Michael Glueck, and Azam Khan. Multiscale 3d navigation. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games*, pp. 7–14, 2009.
- [20] Fu-En Wang, Hou-Ning Hu, Hsien-Tzu Cheng, Juan-Ting Lin, Shang-Ta Yang, Meng-Li Shih, Hung-Kuo Chu, and Min Sun. Self-supervised learning of depth and camera motion from 360 degree videos. In *Asian Conference on Computer Vision*, pp. 53–68. Springer, 2018.
- [21] Chao Zhang, Stephan Liwicki, William Smith, and Roberto Cipolla. Orientation-aware semantic segmentation on icosahedron spheres. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3533–3541, 2019.
- [22] M. Tanaka and T. Ichikawa. A visual user interface for map information retrieval based on semantic significance. *IEEE Transactions on Software Engineering*, Vol. 14, No. 5, pp. 666–670, May 1988.
- [23] Bing. Bing maps. <https://www.bing.com/maps/>, 2009.
- [24] MapFan. Mapfan. <https://mapfan.com/>, 1997.
- [25] iosphere GmbH. Offmaps 2. <http://www.offmaps.com/>, 2011.
- [26] Japanese Geographical Survey Institute. Maps. <https://www.gsi.go.jp/tizu-kutyu.html>, 2015.
- [27] Mapillary AB. Mapillary. <https://www.mapillary.com/>, 2014.
- [28] Matthew J Westoby, James Brasington, Niel F Glasser, Michael J Hambrey, and Jen-

- nifer M Reynolds. ‘structure-from-motion’ photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, Vol. 179, pp. 300–314, 2012.
- [29] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [30] P. Lothe, S. Bourgeois, F. Dekeyser, E. Royer, and M. Dhome. Towards geographical referencing of monocular slam reconstruction using 3d city models: Application to real-time accurate vision-based localization. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2882–2889, June 2009.
- [31] M. Tamaazousti, V. Gay-Bellile, S. N. Collette, S. Bourgeois, and M. Dhome. Nonlinear refinement of structure from motion reconstruction by taking advantage of a partial knowledge of the environment. In *CVPR 2011*, pp. 3073–3080, June 2011.
- [32] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pp. 298–372. Springer, 1999.
- [33] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *European conference on computer vision*, pp. 29–42. Springer, 2010.
- [34] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2041–2050, 2018.
- [35] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283*, 2014.
- [36] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2002–2011, 2018.
- [37] Shinya Sumikura, Mikiya Shibuya, and Ken Sakurada. OpenVSLAM: A Versatile Visual SLAM Framework. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM ’19, pp. 2292–2295, New York, NY, USA, 2019. ACM.
- [38] Raul Mur-Artal and Juan D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *CoRR*, Vol. abs/1610.06475, , 2016.
- [39] Zheyuan Lu, Zhencheng Hu, and Keiichi Uchimura. Slam estimation in dynamic outdoor environments. *International Journal of Humanoid Robotics*, Vol. 7, No. 02, pp. 315–330, 2010.
- [40] Jakob Engel, Thomas Schoeps, and Daniel Cremers. Lsd-slam: large-scale direct monocular slam. Vol. 8690, pp. 1–16, 09 2014.
- [41] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pp. 2564–2571, Nov 2011.

# 発表文献

## 国際会議

- [1] Naoki Sugimoto, Yoshihito Ebine, Kiyoharu Aizawa. Building Movie Map - A Tool for Exploring Areas in a City. ACM Multimedia, pp. 3330-3338, 2020. (Full Paper)
- [2] Naoki Sugimoto, Yuko Inuma, Kiyoharu Aizawa. Walker's Movie Map: Route View Synthesis Using Omni-directional Videos. ACM Multimedia, 1050-1052, 2019. (Demo)
- [3] Naoki Sugimoto, Toru Okubo, Kiyoharu Aizawa. Urban Movie Map for Walkers: Route View Synthesis using 360 ° Videos. ACM ICMR, pp. 502-508, 2020. (Oral)

## 国内会議

- [4] 杉本直樹, 飯沼宥光, 相澤清晴. Movie Map のための全方位映像を用いた自由経路合成. 第 22 回 画像の認識・理解シンポジウム (MIRU). 2019. (デモ)
- [5] 杉本直樹, 海老根義人, 相澤清晴. Movie Map による街の探索: 全方位映像データベースと探索インターフェース. 第 23 回 画像の認識・理解シンポジウム (MIRU). 2020. (ポスター)
- [6] 杉本直樹, 相澤清晴. 360°映像を利用した Movie Map インターフェースによるバーチャル探訪の実現. 映像情報メディア学会 創立 70 周年記念大会. 2020. (オーラル)
- [7] 杉本直樹, 相澤清晴. 街探索のための Movie Map の自動構築と評価. 第 25 回バーチャルリアリティ学会大会. 2019. (オーラル)

## 本研究に関係のない発表

- [8] 杉本直樹, 鶴岡慶雅. 戦略の動的推定による 2 人対戦ゲーム接待 AI の提案. ゲームプログラミングワークショップ 2018 論文集, pp. 114-119, 2018. (ポスター)

# 謝辞

鶴岡研究室での学部研究生生活を終え相澤山崎研究室に配属されて2年間、東京大学に在籍した6年間の中で圧倒的に密度の高い日々を送らせていただきました。始めは研究室の前提知識の水準の高さに戦々恐々とする毎日でしたが、勉強会や密なミーティング、そしてメンバー皆それぞれが研究者として高い意識を持った研究室全体の雰囲気のおかげもあり研究者として何弾もレベルアップすることが出来たように思います。また、豊富な研究資源と技術的な疑問に対して即座に対応していただける先輩方の存在のおかげもあり、情報技術者としての能力も幾分か会得できました。そういった意味でとても価値ある2年間でした。

指導教員である相澤先生にはゲームが好きな自分に合うかもしれないという文脈でこのMovie Mapというテーマを与えていただき、ミーティングや面談の際には次に何をしたら面白いだろうか、何を膨らませたら発展があるだろうかという提案をしてくださいました。論文投稿やプレゼンの準備の際には何度も、加えて迅速に内容や表現の推敲を手伝っていただきました。海外発表の際には初めての海外渡航となった自分のために、それこそ飛行機に乗って出発から日本への帰還まであらゆる面でサポートしていただきました。他にも研究道具の調達や実験の連絡サポート、論文投稿、登録のお勧めなど自分のために割いてもらった時間と労力はとても大きなものですが、何よりも有難かったのは、お話をする際にいつも、自分の提案や発表を面白いと褒めていただいた上ですらなる提案をしてくださった事です。1人で研究や実装をしていると本当に自分は正しいことをしているのか、成果を出せているのかと不安になってしまう事が多いのですが、ミーティング等を迎えて先生とお話をすると、自分のしていたことが肯定された気がしてまた頑張る事が出来ました。

山崎先生、松井先生、そして3D班の池畑さんには、ミーティングの際に同分野の研究者の視点、そして分野外の間人としての始点療法においてクリティカルな指摘を多くいただき、「面白そう」で始めたMovie Mapの研究をより技術的に発展性があり、論理的に課

題を説明できるものとするための大きな助けとなりました。実装は楽しいけれど、論文などとして形になるのか不安であった本研究が、こうして多くの外部発表や本修士論文として形を残せたのは先生方のおかげです。ありがとうございました。

その他、学術支援職員の松林さんには主に金銭面のやり取りについて、初めての事ばかりで何も分からない自分に面倒が無いように説明や書類の要請をしてくださり円滑な事務手続きのためご協力いただきました。3D 班の先輩である金子先輩には、現在の Movie Map の基幹技術である SLAM についての知見の共有の他、研究における論理的な進め方、考え方について学ばせていただきました。

また同期や同じ 3D 班の後輩の皆とはコロナウイルスの流行のせいもあり中々顔を合わせて話をする機会は持てませんでしたが、ミーティングでの進捗報告を通していつも刺激を貰い、自分も頑張らなければと気持ちを奮い立たせる助けになっていました。

最後に、金銭面や精神面で 6 年間の東京大学での生活をサポートしてくれた親族の皆様方に感謝を述べ、謝辞とさせていただきます。本当に、ありがとうございました。

2021 1 月 28 日

杉本 直樹