2022年度
中部大学大学院工学研究科情報工学専攻

博士学位論文

周辺環境を考慮した経路予測に関する研究

箕浦 大晃

論 文 要 旨

経路予測とは，人間や自動車などの移動物体が未来にどのような経路を辿るかを過去の経路から予測する技術である．経路予測は，自動運転車，自律ロボット及び，監視カメラシステムなどで応用が期待され精力的に研究されている．これらのアプリケーションでは，歩行者や自動車といったクラスが異なる複数の対象の経路を同時に予測したり，対象間の衝突を避ける経路を予測したりする必要がある．特に後者は，人との接触の可能性があるシーンにおいて自動運転車や自律ロボット等の機械が衝突を避ける経路を予測しなければ事故に繋がる．人間は暗黙的なインタラクションにより，人間同士が互いに衝突を避けることを考慮するが，それを機械が考慮するためには人間が明示的に与えるか機械自身がモデル化する必要がある．そこで，人間同士の衝突を避けるインタラクションを深層学習でモデル化することで高性能な経路予測を実現できる．

深層学習で対象間のインタラクションをモデル化することで衝突回避が可能な経路予測が実現できる一方，従来の予測手法は以下 3 つの問題が挙げられる．1 つ目は，歩行者を対象とし暗黙的にクラスが異なる対象を同一クラスと扱う点である．例えば，歩行者は歩道や車道を歩き，自動車は車道を走ることが想定される．歩行者や自動車等が映るシーンで経路を同時予測する場合，既存の経路予測手法では予測対象のクラスに応じた経路を予測できない．これを解消するために対象のクラス毎にモデルを作成する必要があるが，対象のクラスが増加するにつれ扱うモデル数も増加し計算コストの面から現実的とは言えない．2 つ目は，正確な経路情報が必要となる点である．経路予測において必要となるのは，一連のフレームにおける予測対象の ID 情報で，これは検出と追跡技術により取得可能である．実社会で経路予測の実装を想定すると，混雑シーンでオクルージョン等により予測対象の正確な ID 情報の取得が困難で，既存の経路予測手法をそのまま適用できない．3 つ目は，混雑シーンで個々人のインタラクションを求めるコストがかかる点である．インタラクションを捉える既存研究は，個人レベルでインタラクションを求めており，対象数が増加するにつれ計算コストがかかる．そのようなシーンでは，対象間のインタラクションを効率的にモデル化する必要がある．

本研究では，上記問題に対し，インタラクションを扱うアプローチとインタラクション以外を扱うアプローチで対処する．まず，1 つ目では移動対象のクラスに応じた経路予測を目的に，予測対象の属性と周囲の環境情報を導入した経路予測を提案する．歩行者や自動車を属性とみなし，属性を one-hot vector とコンパクトな表現で扱うことで，対象のクラス毎にモデルを作成する必要がない．また，予測対象周囲の環境情報を用いることで，移動対象のクラスに応じた経路予測を実現する．2 つ目では，不正確な経路情報で適用可能な経路予測を目的に，シーンの各場所が将来どれだけ混雑しているかのマップ，すなわち群集密度マップを直接予測する手法を提案する．群衆をマップとして表現することで，正確な歩行者の検出と追跡を必要とせず直接群衆の将来の動きを予測できる．3 つ目では，混雑シーンにおける効率的な歩行者間のインタラクションを捉える経路予測を目的に，歩行者間のグループ間とグループ内のインタラクションをモデル化する Group-based Forecasting Module を提案する．Group-based Forecasting Module では，各グループのインタラクションに関連する重要な対象に着目させるために Attention 機構を導入する．各グループで独立した Attention 機構により，それぞれのグループの特性に応じた経路の予測及びインタラクションを効率的に捉えることができる．

目 次

図 目 次

表 目 次

第1章

序論

本章では，本研究の背景及び目的，本論文の構成について述べる．

1.1 研究の背景

経路予測とは，人間や自動車などの移動物体が未来にどのような経路を辿るかを過去の経路から予測する技術である．特に人間の経路を予測することは人間と機械が共存するために重要な技術になる．例えば，自動運転車 [30] [31] [32] や自律ロボット [33] [34] [35]，軽犯罪防止のための高度な監視カメラシステム [36] などは人間社会をサポートする技術として注目されている．これらの技術における正確な経路予測のためには，歩行者や自動車といったクラスが異なる複数の対象の経路を同時に予測したり，対象間の衝突を避ける経路を予測したりする必要がある．前者は，現実のシーンでは歩行者だけでなく自動車や自転車等の異なるクラスの移動物体がいる環境下で予測を行う必要があり，対象のクラスにより移動速度や移動領域が異なることが想定される．後者では，自動運転車では横断中の歩行者や道路に飛び出してくる人，自律ロボットでは群衆の中のナビゲーションなど，人との接触の可能性があるシーンでは機械が衝突を避ける経路を予測しなければ事故に繋がる．人間は暗黙的なインタラクションにより，人間同士が互いに衝突を避けることを考慮するが，それを機械が考慮するためには人間が明示的に与えるか機械自身がモデル化する必要がある．しかし，人間のインタラクション，特に人間同士の衝突を避けるパターンは複雑であり人間が機械に明示的に与えるのは困難である．このような背景より，人間同士の衝突を避けるインタラクションをモデル化する経路予測についてコンピュータビジョン分野で研究されている．

移動対象間の衝突を回避するインタラクションを実現するために，深層学習によるアプローチが数多く提案されている．インタラクションを考慮した予測手法はプーリングモデル [1] [3] とアテンションモデル [15] [10] 及び，その他のモデル [37] に分類される．まず，プーリングモデルは予測対象を中心とした周囲の空間を表現したグリッドを作成し，そのグリッドに予測対象周辺の個々の他対象に関する時系列特徴を埋め込むことでその対象と衝突回避する経路を予測する．アテンションモデルは予測対象と個々の他対象との関係を Softmax 関数で合計値が 1 となる連続的な重み値で表現し，この重みをそれぞれの他対象の特徴量と乗算している．最後に，その他のモデルはプーリングやアテンションモデルとは異なる方法でインタラクションを考慮している．

インタラクションを考慮した予測手法により，対象間で衝突回避が可能な経路予測を実現できる．しかし，これらの予測手法は，i) 歩行者を対象とし暗黙的に同一クラスの対象として予測する．また，ii) 正確な経路情報が必要，iii) 個人レベルのインタラクションを過去の経路から求めている．i) は，歩行者や自動車といったクラスが異なる複数の対象の経路を同時予測する場合，既存の予測手法では予測対象のクラスに応じた経路を予測することが困難である．予測対象のクラスに応じた経路を予測するには対象のクラス毎にモデルを作成することが考えられるが，対象のクラスが増加するにつれ，扱うモデルの数が増加するため，計算コストを抑えるアプローチが必要となる．ii) では，既存の経路予測手法を社会実装する場合，予測対象の正確な検出と追跡で捉えた経路及び ID 情報が入力と出力の両方で必要となる．しかし，混雑シーンではオクルージョン等により歩行者の正確な検出と追跡が困難で，既存の経路予測手法をそのまま適用できないため，不正確な歩行者データでも適用可能なアプローチが必要となる．iii) では，個人レベルのインタラクションは個々人の関係を捉えるため，混雑シーンでは各対象のインタラクションを求めるコストがかかる．一般的に混雑

シーンの歩行者は，衝突を避けたり周囲の集団に合わせるなどの暗黙的な社会ルールに従う傾向がある [38]．つまり，混雑シーンではグループで行動する歩行者が多く，正確な経路予測のためにグループレベルに基づく社会的なインタラクションをモデル化することが重要となる．また，社会的なインタラクションのモデル化にあたり，過去の経路情報から将来の他者の位置に基づくインタラクションを捉えるのが困難なため，数秒先に移動する可能性のある他対象の未来の経路からインタラクションを捉えるアプローチが必要になる．

1.2 研究目的

本研究では，以下の 3 つの項目を目的とする．

1. 予測対象のクラスに応じた経路予測.
2. 混雑シーンにおける不正確な歩行者データでの予測.
3. 将来の他者の位置及びグループレベルに基づく社会的なインタラクションを捉える経路予測.

1 つ目は，歩行者や自動車といったクラスが異なる複数の対象がいるシーンを想定し，予測対象が保有する潜在的な特徴を考慮することで，対象クラス毎の特徴的な動きを予測する．2 つ目と 3 つ目は，市街地やショッピングモールといった混雑シーンを想定し，混雑シーンにおける不正確な経路データから歩行者の経路を予測する手法及び，歩行者間の社会的なインタラクションの導入による経路予測手法を提案する．以下に，三項目における本研究の目的について述べる．

予測対象のクラスに応じた経路予測　自動運転やソーシャルロボット等のアプリケーションにおいて，高性能な経路予測を実現するために歩行者や自動車等の異なる移動物体の移動領域や移動速度の情報が必要となる．深層学習による経路予測手法は，これらの対象を同一クラスの予測対象として扱い，予測対象のクラスに応じた経路を予測できない．例えば，歩行者の場合に歩道や横断歩道を歩く，自動車の場合に車道を走る，といった各対象が潜在的に保有している特徴を無視する．これに対処するために，予測対象のクラス毎にモデル化し予測を行うことが考えられる．しかしながら，対象のクラスが増加するにつれ，扱うモデル数が増加するため，計算コストの面から現実的とは言えない．このような理由から，本研究では移動対象のクラスに応じた経路予測を目的に，予測対象の属性と環境情報を導入した経路予測を提案する．まず，歩行者や自動車等の予測対象を属性とみなし，属性を ohe-hot vector として表現する．これにより，対象のクラス毎にモデルを作成することはない．各対象は潜在的に保有している特徴が異なるため，各対象の移動領域が異なる．そこで，予測を行うシーンに付与されたシーンラベルを予測対象周囲の環境情報として表現する．予測対象の移動量と，one-hot vector で表現された属性情報及び，シーンラベルで表現された環境情報を用いて，移動対象のクラスに応じた経路予測を実現する．

混雑シーンにおける不正確な歩行者データでの予測　混雑シーンに対しても歩行者の正確な経路を予測することは，経路予測を社会実装する上で重要な課題である．歩行者の経路を予測するには，正

確な位置検出と追跡で取得したデータが経路予測ネットワークの入力と出力の両方に必要となる．しかし，混雑シーンではオクルージョン等により歩行者の正確な検出と追跡ができず，不正確な経路情報をネットワークへ入力し誤った経路を予測する問題がある．特に，追跡に誤りがあった場合，経路情報を取得する上で重要な歩行者 ID を割り当てることができず，そもそも経路予測ネットワークへ経路情報を入力することができない．歩行者の不正確な経路データでは既存の経路予測手法に適用できないため，不正確な経路データでも適用可能なアプローチが必要となる．本研究の目的は，混雑シーンにおける歩行者データでの予測を実現することである．そこで，本研究ではシーンの各場所が将来どれだけ混雑しているかのマップ，すなわち群集密度マップを直接予測する手法を提案する．群衆をマップとして表現することで，正確な歩行者の検出と追跡を必要とせず，群衆密度のダイナミクスを直接捉えることができる．しかし，広域で撮影された入力映像では独立して移動する複数の集団が含まれていることが多く，群衆密度マップの時空間的ダイナミクスは複雑になり予測が困難となる．これに対処するために，本研究はパッチベースの密度予測ネットワークを提案する．パッチベースでモデル化することで，様々な数の独立した群衆の複雑な時空間的ダイナミクスを効率的に捉えることが期待できる．

将来の他者の位置及びグループレベルに基づく社会的なインタラクションを捉える経路予測　経路予測は歩行者間の社会的インタラクションを考慮することで，衝突を回避した経路を予測できる．経路予測や社会的インタラクションのモデル化は，自動運転や自律ロボットなどのアプリケーションの基盤技術として重要になる．しかし，混雑シーンにおいて人の行動は多様な行動パターンの変化により，歩行者間の社会的インタラクションは複雑で捉えることが困難である．社会的インタラクションを捉える予測手法は個人レベルでインタラクションを捉えている．この方法では，各個人に対してインタラクションを求めるため，シーン内の歩行者数が多くなるにつれ計算コストが増加する．一般的に，群衆シーンでは歩行者の大半はグループで行動しており，同じ方向に歩いている人と自発的にグループを形成するなどして，暗黙的な社会的ルールに従う傾向がある．このことを念頭に置き，グループレベルに基づいてインタラクションを捉える予測手法が提案されている．しかし，これらは過去の経路から他者とのインタラクションを捉えるため，将来で予測対象同士が衝突する．我々人間は，反対方向から来る他の歩行者の将来の位置を予想することで，その対象と衝突を回避できる．本研究の目的は，混雑シーンにおける経路予測のために人間の社会的インタラクションを捉えることである．そこで，本研究では歩行者間の複雑なグループ間とグループ内の社会的インタラクションをモデル化する Group-based Forecasting Module を提案する．Group-based Forecasting Module では，社会的インタラクションに関連する重要な対象に着目させるために Attention 機構を導入する．グループ間の Attention 機構では，将来の他者の位置と現時刻の予測対象自身の位置からインタラクションを求める．グループ内の Attention 機構では，予測対象が属するグループ内の他対象とのインタラクションを求める．それぞれを独立して求めることで，グループレベルのインタラクションを効率的に捉える経路を予測できる．

1.3　本論文の構成

本論文は，図 1.1 に示すように 7 つの章で構成されている．1 章では，本研究の背景と目的を述べた．本研究では，予測対象のクラスに応じた経路予測，混雑シーンにおける不正確な歩行者データでの予測及び，将来の他者の位置及びグループレベルに基づく社会的なインタラクションを捉える経路予測について，それぞれの枠組みを提案する．2 章では，経路予測の関連研究について述べる．また，経路予測で用いられるデータセット及び評価指標についても調査し，体系的にまとめる．3 章では，移動対象の属性と環境情報を導入した経路予測について述べる．4 章では，混雑シーンにおける不正確な歩行者データの予測を目的に，歩行者集団の密度をマップで表現し，未来でどのようにマップが変化するかを視覚的に予測する群衆密度予測について述べる．5 章では，2 章で述べた経路予測手法の中からインタラクションを考慮する代表的な予測手法による性能調査について述べる．そして，6 章では 4 章と 5 章で得た知見から，グループレベルのインタラクションによる経路予測手法について述べる．7 章では，本論文の結論と展望について述べる．

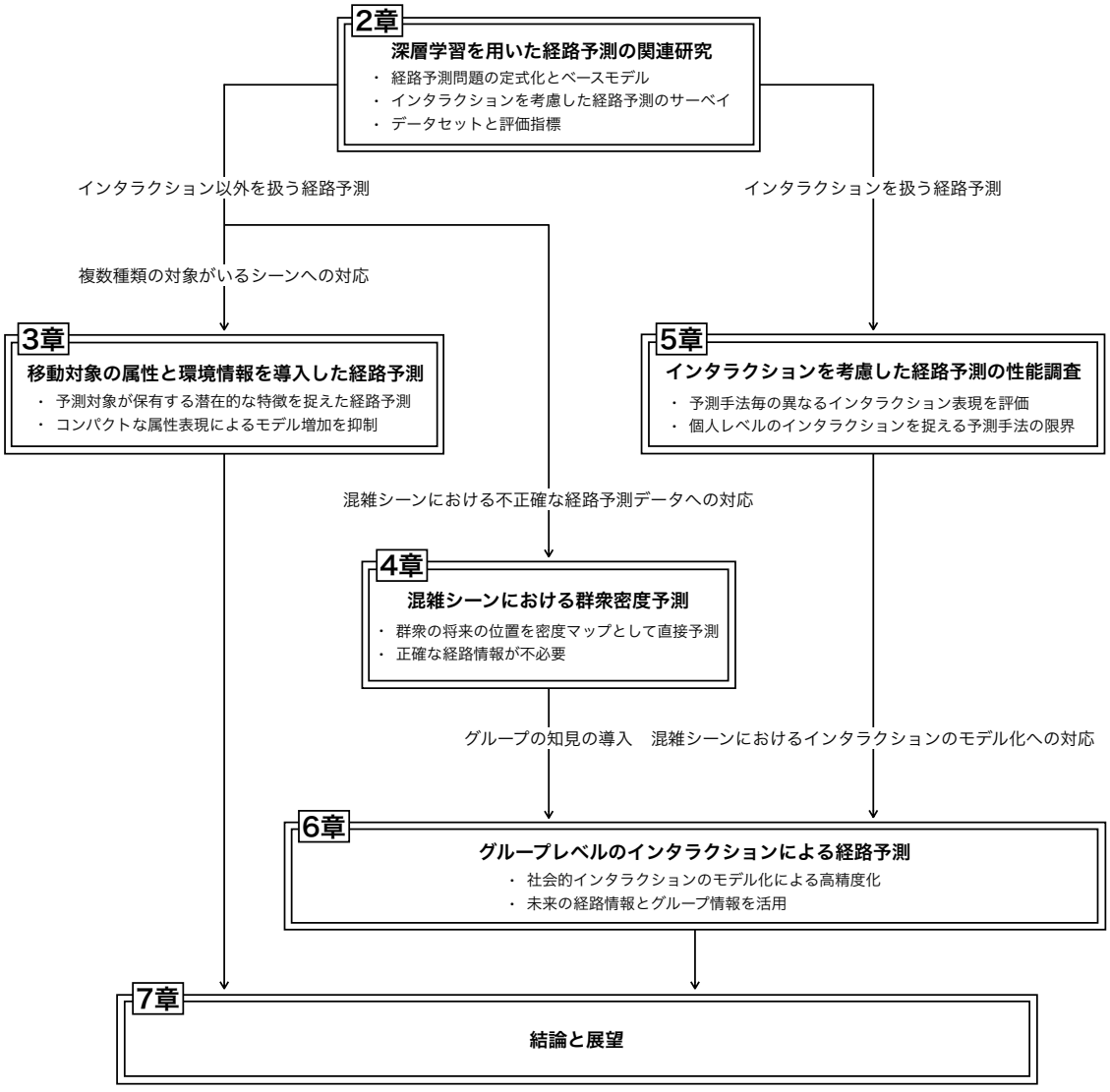

2章
深層学習を用いた経路予測の関連研究
・ 経路予測問題の定式化とベースモデル
・ インタラクションを考慮した経路予測のサーベイ
・ データセットと評価指標
インタラクション以外を扱う経路予測
インタラクションを扱う経路予測
複数種類の対象がいるシーンへの対応
3章
移動対象の属性と環境情報を導入した経路予測
・ 予測対象が保有する潜在的な特徴を捉えた経路予測
・ コンパクトな属性表現によるモデル増加を抑制
5章
インタラクションを考慮した経路予測の性能調査
・ 予測手法毎の異なるインタラクション表現を評価
・ 個人レベルのインタラクションを捉える予測手法の限界
混雑シーンにおける不正確な経路予測データへの対応
4章
混雑シーンにおける群衆密度予測
・ 群衆の将来の位置を密度マップとして直接予測
・ 正確な経路情報が不必要
グループの知見の導入
混雑シーンにおけるインタラクションのモデル化への対応
6章
グループレベルのインタラクションによる経路予測
・ 社会的インタラクションのモデル化による高精度化
・ 未来の経路情報とグループ情報を活用
7章
結論と展望

図 1.1: 本論文の構成.

第2章

深層学習を用いた経路予測の関連研究

経路予測とは，人間や自動車などの移動物体が未来にどのような経路を辿るかを過去の経路から予測する技術である．特に人間の経路を予測することは人間と機械が共存するために重要な技術になる．例えば，自動運転車やモバイルロボット，軽犯罪防止のための高度な監視カメラシステムなどは人間社会をサポートする技術として注目されている．これらの技術における正確な経路予測のためには，衝突を避ける経路を予測する必要がある．例えば，自動運転車では飛び出してくる人や，モバイルロボットでは群衆の中のナビゲーションなど，人との接触の可能性があるシーンでは機械が衝突を避ける経路を予測しなければ事故に繋がる．人間は暗黙的なインタラクションにより，人同士が互いに衝突を避けることを考慮するが，それを機械が考慮するためには人間が明示的に与えるか機械自身がモデル化する必要がある．しかし，人間のインタラクション，特に人同士の衝突を避けるパターンは複雑であり人間が機械に明示的に与えるのは困難である．このような背景より，人同士の衝突を避けるインタラクションをモデル化する経路予測についてコンピュータビジョン分野で研究が行われている．本章では，固定したカメラ映像内の複数の移動対象同士が相互に影響する経路を辿る，つまり衝突を避ける現象のことをインタラクションとして定義する．ただし，自己運動を通じてカメラ装着者と単一の対象の 2 者が相互に影響する場合，本章ではインタラクションと定義しないことに注意されたい．

経路予測は動画像内の予測対象の過去の経路をネットワークへ入力し，未来の経路を予測する．その際，経路にはピクセル座標系またはメートル座標系を用いる．ピクセル座標系は，画像座標系に基づいて画像の左上画素を原点とした座標系である．メートル座標系は，ホモグラフィ行列を用いてピクセル座標系から変換された座標系である．これらの座標系から，インタラクションをモデル化する経路予測が行われている．

経路予測のベースとなる方法を大きく 3 つで分類する．1 つ目は Physics ベースである．Physics ベースはカルマンフィルタ [39] や線形予測 [40] のようなベイズモデルがある．ベイズモデルに基づくアプローチでは，内部状態と呼ばれる変数から，内部状態にノイズが付与されたものが予測として現れる確率的モデルを定義し時刻間の内部状態を更新することで予測値を逐次推定する [41] [42] [30]．また，Physics ベースには Social Force Model [43] もある．Social Force Model は歩行者間や歩行者と建物のような物体との間に引力と斥力のようなエネルギー場があると仮定することで，物体との衝突を避けるインタラクションを捉えることができる [44] [45]．2 つ目は深層学習 (Deep Learning : DL) ベースである．DL ベースの経路予測手法では，歩行者などの移動物体の過去の経路情報から行動パターンを学習することで，長期的な予測を実現している．また，DL ベースの経路予測手法は人手で

移動対象間の複雑なインタラクションを明示的にモデル化する必要はなく，学習器が自動でモデル化する．3 つ目は Planning ベースである．Planning ベースは逆強化学習 [46] [47] やパスプランニング [48] のように予測開始地点と目標地点を設定するモデルがある．これらのモデルは，予測開始地点から目標地点までに得られる報酬値や距離などのコストを最適化することで，尤もらしい経路を獲得できる．これらのように経路予測は様々なベースとなる方法がある．本章では，2016 年以降爆発的に増え続けている DL ベースの経路予測手法問題について扱う．

DL ベースの経路予測で用いられるネットワークモデルは，主に Recurrent Neural Network (RNN) と Convolutional Neural Network (CNN) [49] を利用している．RNN による経路予測は，RNN を拡張した Long Short-Term Memory (LSTM) [50] 及び，Gated Recurrent Unit (GRU) [51] が主なネットワークモデルとして用いられている．RNN や RNN を拡張したモデルは，移動対象の経路情報に関する特徴を時間方向に伝播することで，未来の予測経路を取得する．RNN が再帰的に処理することで直近の特徴に強く影響すること，計算の並列化が困難なことから，2017 年自然言語処理で提案された Transformer [52] による経路予測が増加傾向にある．一方，CNN では映像内の静的物体情報を予測モデルに組み込むことで，静的物体との衝突を避ける経路を予測する．また，時間方向へ伝播する CNN [53] により RNN のように未来の予測経路を獲得できる．

ここで，DL ベースの経路予測手法を図 2.1 及び表 2.1 にまとめる．図 2.1 は移動対象間の衝突を回避するインタラクションを考慮した経路予測手法の傾向と分類を示したものである．図 2.1 より，インタラクションを考慮した手法はプーリングモデルとアテンションモデル及び，その他のモデルに分類する．一方で，インタラクション以外の課題を扱う手法もある．本章では，それぞれを次のように定義する．プーリングモデル: 予測対象を中心として周囲の空間を表現したグリッドを作成し，これに予測対象周辺の個々の他対象に関する系列特徴を埋め込むアプローチ．アテンションモデル: 予測対象と個々の他対象との関係を Softmax 関数で合計値が 1 となる連続的な重み値で表現し，この重みをそれぞれの他対象の特徴量と乗算するアプローチ．その他のモデル: プーリングやアテンションモデルとは異なる方法でインタラクションを考慮したアプローチ．インタラクション以外の課題を扱う手法: 周辺環境の読み取りをして建物といった静的な障害物との衝突回避を目的とした経路予測などインタラクション以外の課題を扱うアプローチ．

また，これらの経路予測手法はインタラクションの表現方法だけでなく，様々な基準で分類できる．表 2.1 より，映像視点は経路を予測する視点を表し，鳥瞰視点や車載カメラ視点，1 人称視点に分類できる．対象属性は各予測手法がどの移動物体を対象としているかを示す．複数経路は複数の未来の経路を予測する手法を示す．環境は映像内のシーンコンテキストやセマンティックラベルを用いた予測手法を示す．目的関数は，文献で用いられている損失関数を示す．経路予測には周辺環境の読み取り，複数経路の予測，インタラクションなどの様々な課題があるなかで，インタラクションの課題に取り組む論文が多いことが表 2.1 から読み取れる．

このように，経路予測は DL の発展に伴い，様々な予測手法が提案されている．特に，移動物体間の衝突を避けるインタラクションを考慮した予測手法が爆発的に増え続けている．本章では，コンピュータビジョン分野で爆発的に増え続けている DL ベースの経路予測手法について扱う．多くの手法で取り組まれている移動物体間のインタラクションのモデル化のアプローチについて着目し，経

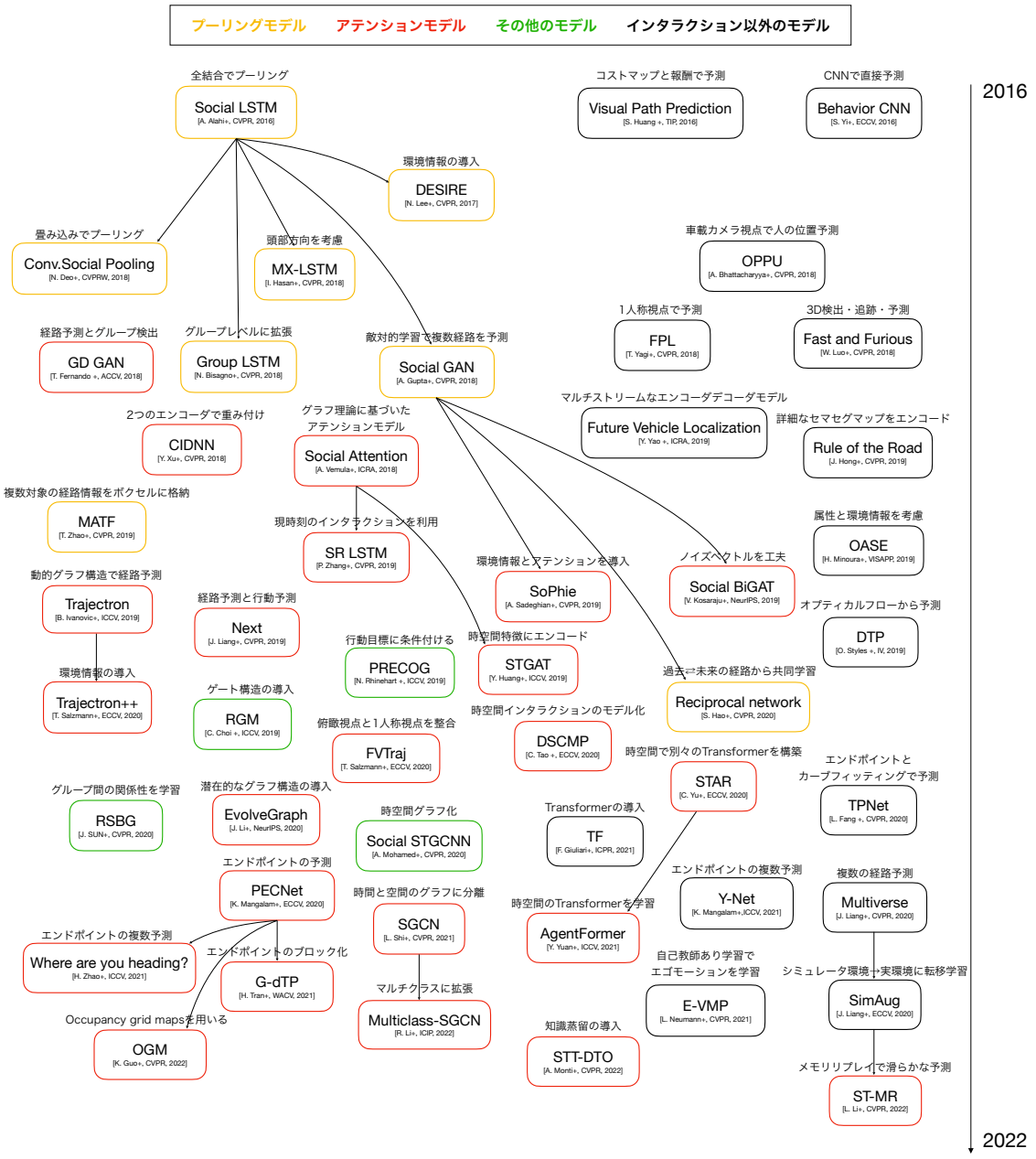


図 2.1: 移動対象間の衝突を回避するインタラクションを考慮した経路予測手法の傾向と分類．インタラクションを考慮した予測手法は各色の枠，インタラクション以外の課題を扱う手法を黒色の枠で示す．黄色の枠はプーリングモデルに基づくアプローチ，赤色の枠はアテンションモデルに基づくアプローチ，緑色の枠はプーリングやアテンションモデルとは異なる方法でインタラクションを考慮するその他のモデルに基づくアプローチを表す．

路予測手法を体系的に整理し述べる．

本章の構成は以下のとおりである．まず，2.1 節では経路予測のベースモデルと問題設定について述べる．2.2 節ではインタラクションを考慮した経路予測手法について幾つかのカテゴリに分類し，様々な予測手法について述べる．2.3 節では経路予測の性能を評価する際に用いられるデータセットについて述べる．2.4 節では経路予測で用いられる評価指標について述べる．最後に 2.5 節で本章をまとめる．

表 2.1: 経路予測手法の分類.

	文献	発表年	モデル	映像視点	対象属性	複数経路	環境	目的関数
プーリング	Social LSTM [1]	2016	LSTM	鳥瞰	歩行者			負の対数尤度
	DESIRE [54]	2017	GRU Enc.Dec.	鳥瞰	歩行者 自動車	✓	✓	Reconstruction KLD Cross-entropy Regression
	Conv. Social Pooling [55]	2018	LSTM	鳥瞰	自動車	✓		負の対数尤度
	MX-LSTM [2]	2018	LSTM	鳥瞰	歩行者			負の対数尤度
	Group LSTM [56]	2018	LSTM	鳥瞰	歩行者			負の対数尤度
	Social GAN [3]	2018	LSTM Enc.Dec.	鳥瞰	歩行者	✓		Adversarial L2
	MATF [4]	2019	LSTM Enc.Dec.	鳥瞰	歩行者 自動車	✓	✓	Adversarial Reconstruction
	Reciprocal network [57]	2020	LSTM Enc.Dec.	鳥瞰	歩行者	✓	✓	Reciprocal Adversarial
アテンション	Social Attention [15]	2018	LSTM	鳥瞰	歩行者			負の対数尤度
	CIDNN [5]	2018	LSTM	鳥瞰	歩行者			L2
	GD GAN [58]	2018	LSTM	鳥瞰	歩行者			Adversarial L1
	SR LSTM [6]	2019	LSTM	鳥瞰	歩行者			L2
	Next [7]	2019	LSTM Enc.Dec.	鳥瞰	歩行者		✓	Cross-entropy SmoothL1 L2
	SoPhie [8]	2019	LSTM Enc.Dec.	鳥瞰	歩行者	✓	✓	Adversarial L2
	STGAT [9]	2019	LSTM Enc.Dec.	鳥瞰	歩行者	✓		L2
	Social BiGAT [59]	2019	LSTM Enc.Dec.	鳥瞰	歩行者	✓	✓	Adversarial L2 KLD
	Trajectron [10]	2019	LSTM Enc.Dec.	鳥瞰	歩行者	✓		尤度最大化
	Trajectron++ [60]	2020	LSTM Enc.Dec	鳥瞰	歩行者 自動車	✓	✓	尤度最大化
	EvolveGraph [61]	2020	GRU	鳥瞰	歩行者 自動車	✓	✓	尤度最大化
	STAR [11]	2020	Transformer	鳥瞰	歩行者	✓		MSE
	PECNet [12]	2020	LSTM Enc.Dec.	鳥瞰	歩行者	✓		KLD L2
	DSCMP [62]	2020	LSTM Enc.Dec.	鳥瞰	歩行者	✓	✓	Regularization L2
	FVTraj [13]	2020	LSTM Enc.Dec.	一人称 鳥瞰	歩行者	✓		L2
	AgentFormer [63]	2021	Transformer	鳥瞰	歩行者 自動車	✓		尤度最大化
	G-dTP [64]	2021	GRU Enc.Dec.	鳥瞰	歩行者	✓		L2
	Where are you heading? [65]	2021	LSTM Enc.Dec.	鳥瞰	歩行者	✓		負の対数尤度
	SGCN [66]	2021	CNN	鳥瞰	歩行者	✓		負の対数尤度
	Multiclass-SGCN [67]	2022	CNN	鳥瞰	歩行者 自動車	✓		負の対数尤度
	OGM [68]	2022	CNN, RNN	鳥瞰	歩行者	✓	✓	負の対数尤度
	ST-MR [69]	2022	ConvLSTM, Transformer	鳥瞰	歩行者	✓	✓	負の対数尤度
	STT-DTO [14]	2022	Transformer	鳥瞰	歩行者	✓		L2 Distillation
その他	RGM [16]	2019	CNN, LSTM	鳥瞰	歩行者 自動車	✓	✓	L2
	PRECOG [70]	2019	CNN, GRU	鳥瞰	自動車	✓	✓	尤度最大化
	RSBG [17]	2020	LSTM Enc.Dec.	鳥瞰	歩行者	✓	✓	L2
	Social STGCNN [37]	2020	CNN	鳥瞰	歩行者	✓		負の対数尤度
インタラクション以外の課題を扱うアプローチ	Behavior CNN [71]	2016	CNN	鳥瞰	歩行者			L2
	Visual Path Prediction [72]	2016	CNN	鳥瞰	歩行者 自動車		✓	L2
	FPL [29]	2018	CNN	一人称	歩行者			MSE
	OPPU [73]	2018	RNN Enc.Dec.	車載	歩行者	✓	✓	MSE KLD
	Fast and Furious [74]	2018	CNN	鳥瞰	自動車			Cross-entropy SmoothL1
	OASE [75]	2019	LSTM	鳥瞰	歩行者 自動車		✓	MSE
	Rule of the Road [76]	2019	CNN, GRU	鳥瞰	自動車		✓	尤度最大化 L2
	Future Vehicle Localization [77]	2019	GRU	車載	自動車			-
	DTP [78]	2019	CNN	車載	歩行者			L2
	TPNet [79]	2020	CNN	鳥瞰	自動車	✓	✓	Cross-entropy L2
	Multiverse [24]	2020	CRNN	鳥瞰	歩行者	✓	✓	Cross-entropy SmoothL1
	SimAug [80]	2020	CRNN	鳥瞰	歩行者	✓	✓	Cross-entropy SmoothL1
	TF [81]	2021	Transformer	鳥瞰	歩行者	✓		L2
	Y-Net [82]	2021	CNN	鳥瞰	歩行者	✓	✓	Cross-entropy
	E-VMP [83]	2021	CNN, FC	車載	歩行者			L2 Photometric

2.1 経路予測問題の定式化とベースモデル

本節では，インタラクションを考慮した経路予測問題の定式化を行い，DL ベースの経路予測で主に用いられる LSTM 及び CNN について簡潔に説明する．

2.1.1 経路予測問題の定式化

インタラクションを考慮した経路予測手法の多くが [3, 8] に触発されて研究されている．これらは，シーン内に N 人の歩行者や移動物体がいると仮定する．各予測モデルには過去時刻 $T_{obs} = \{1, \ldots, t_{obs}\}$ の経路を入力し，未来時刻 $T_{pred} = \{t_{obs}+1, \ldots, t_{pred}\}$ の経路を予測する．ここで，t_{obs} は過去最終時刻，t_{pred} は未来最終時刻を示す．また，過去時刻で観測する経路を $X_i^{T_{obs}} = \{\mathbf{x}_i^t = (x_i^t, y_i^t)\}, \forall t \in T_{obs}$，未来時刻で予測する経路を $\hat{Y}_i^{T_{pred}} = \{\hat{\mathbf{y}}_i^t = (\hat{x}_i^t, \hat{y}_i^t)\}, \forall t \in T_{pred}$ で表す．ここで，i を予測対象，(x, y) を経路情報，$(\hat{x}, \hat{y})$ を予測の経路情報として表す．各予測モデルはシーン内の N 人の歩行者や移動対象に対し，各予測モデル内でプーリングやアテンションなどで対象同士の関係をモデル化することで，互いとの衝突を避ける経路を予測する．

2.1.2 LSTM

表 2.1 より，経路予測問題で主に扱われるネットワークモデルは LSTM である．経路予測問題における基本的な流れを図 2.2 に示す．経路予測問題では，過去時刻の経路情報 $\mathbf{x}_i^{t_{obs}}$ を入力層を通じて得た特徴 $\mathbf{e}_i^{t_{obs}}$ として LSTM へ入力する．前時刻の LSTM から出力された経路特徴 $\mathbf{z}_i^{t_{obs}-1}$ (内部状態) を用いて，次段落以降で詳細を述べる LSTM 内の各ゲートとメモリセルで経路特徴 $\mathbf{z}_i^{t_{obs}}$ (内部状態) を求め，次時刻の LSTM と出力層へ特徴を伝播する．LSTM で抽出された特徴を用いて，出力層で未来時刻の予測経路 $\hat{\mathbf{y}}_i^{t_{obs}+1}$ を出力する．このとき，過去時刻最後までネットワークに入力するのは過去時刻の経路情報であるが，予測開始時刻すなわち未来開始時刻以降では，ネットワークが出力した予測経路を次時刻の入力として使用する．予測経路は過去時刻の経路情報同様に入力層を通じて LSTM へ入力する．一連を繰り返し行うのが経路予測問題の基本的な流れである．

LSTM の内部構成を図 2.3 に示す．LSTM は，系列データを扱うモデルで過去の情報を記憶する内部状態の働きにより，長期にわたる記憶を可能にしている．LSTM の内部には，内部状態を記憶するメモリセルと 3 つのゲートで構成されている．3 つのゲートは，それぞれメモリセルの値が次時刻でどれだけ保持するかを調節する忘却ゲート，メモリセルに加算する値を調節する入力ゲート，メモリセルの値が次の層にどれだけ影響を及ぼすかを調節する出力ゲートがある．メモリセルと 3 つのゲートにより，長期の記憶の保持が可能となる．

LSTM の内部について説明する．メモリセル $\mathbf{s}^t$ は，式 (2.1) で更新する．式 (2.1) は，メモリセルにとってどの情報が必要になるかを決定する．第 1 項では，前時刻のメモリセル $\mathbf{s}^{t-1}$ と忘却ゲートの出力 $\mathbf{g}^{F,t} \in [0, 1]$ の乗算した値を出力するため，$\mathbf{g}^{F,t}$ が 1 に近いと必要な情報とみなし現状態をそのまま記憶，0 に近いと不必要な情報としてリセットする．第 2 項では，入力特徴 $\tanh(\mathbf{u}^t)$ と入力

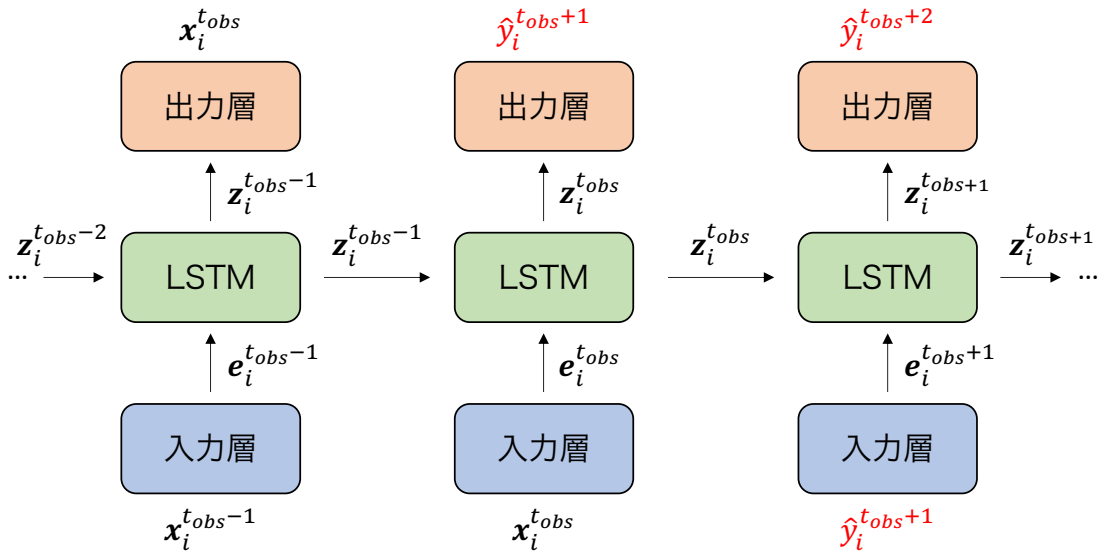


図 2.2: 経路予測問題における基本的な流れ.

ゲートの出力 $\mathbf{g}^{I,t} \in [0,1]$ の乗算した値を出力する．最終的に第 1 項と第 2 項を加算した値が現時刻のメモリセルへ伝播する．

$$\mathbf{s}^t = \mathbf{g}^{F,t}\mathbf{s}^{t-1} + \mathbf{g}^{I,t}\tanh(\mathbf{u}^t). \tag{2.1}$$

LSTM への入力 $\mathbf{u}^t$ は，入力層と前時刻の LSTM の出力から式 (2.2) で計算する．

$$\mathbf{u}^t = \mathbf{W}^{in}\mathbf{e}^t + \mathbf{b}^{in} + \mathbf{W}^{hid}\mathbf{z}^{t-1} + \mathbf{b}^{hid}. \tag{2.2}$$

ここで，$\mathbf{e}^t$ は時刻 t の入力層の出力，$\mathbf{z}^{t-1}$ は前時刻 $t-1$ の LSTM の出力 (内部状態)，$\mathbf{W}^{in}, \mathbf{b}^{in}$ は時刻 t の入力層から時刻 t の LSTM にかかる重みとバイアス，$\mathbf{W}^{hid}, \mathbf{b}^{hid}$ は前時刻 $t-1$ の LSTM から時刻 t の LSTM にかかる重みとバイアスを表す．これらの重みとバイアスの大きさはハイパーパラメータとして人が決定する．忘却ゲートの出力 $\mathbf{g}^{F,t}$ を式 (2.3) で求める．忘却ゲートは入力層の出力 $\mathbf{e}^t$，前時刻の LSTM の出力 $\mathbf{z}^{t-1}$，前時刻のメモリセルの出力 $\mathbf{s}^{t-1}$ を重みが共有されていない全結合で計算し，活性化関数 f を用いて最終的な値を計算する．

$$\mathbf{g}^{F,t} = f(\mathbf{W}^{F,in}\mathbf{e}^t + \mathbf{b}^{F,in} + \mathbf{W}^{F,z}\mathbf{z}^{t-1} + \mathbf{b}^{F,z} + \mathbf{W}^{F,s}\mathbf{s}^{t-1} + \mathbf{b}^{F,s}). \tag{2.3}$$

ここで，式 (2.3) の $\mathbf{W}^{F,in}, \mathbf{b}^{F,in}$ は時刻 t の入力層の出力にかかる重みとバイアス，$\mathbf{W}^{F,z}, \mathbf{b}^{F,in}$ は前時刻 $t-1$ の LSTM の出力にかかる重みとバイアス，$\mathbf{W}^{F,s}, \mathbf{b}^{F,s}$ は前時刻 $t-1$ のメモリセルの出力にかかる重みとバイアスを表す．入力ゲートの出力 $\mathbf{g}^{I,t}$ は式 (2.4) で求める．入力ゲートは忘却ゲートと同様に導出する．

$$\mathbf{g}^{I,t} = f(\mathbf{W}^{I,in}\mathbf{e}^t + \mathbf{b}^{I,in} + \mathbf{W}^{I,z}\mathbf{z}^{t-1} + \mathbf{b}^{I,z} + \mathbf{W}^{I,s}\mathbf{s}^{t-1} + \mathbf{b}^{I,s}). \tag{2.4}$$

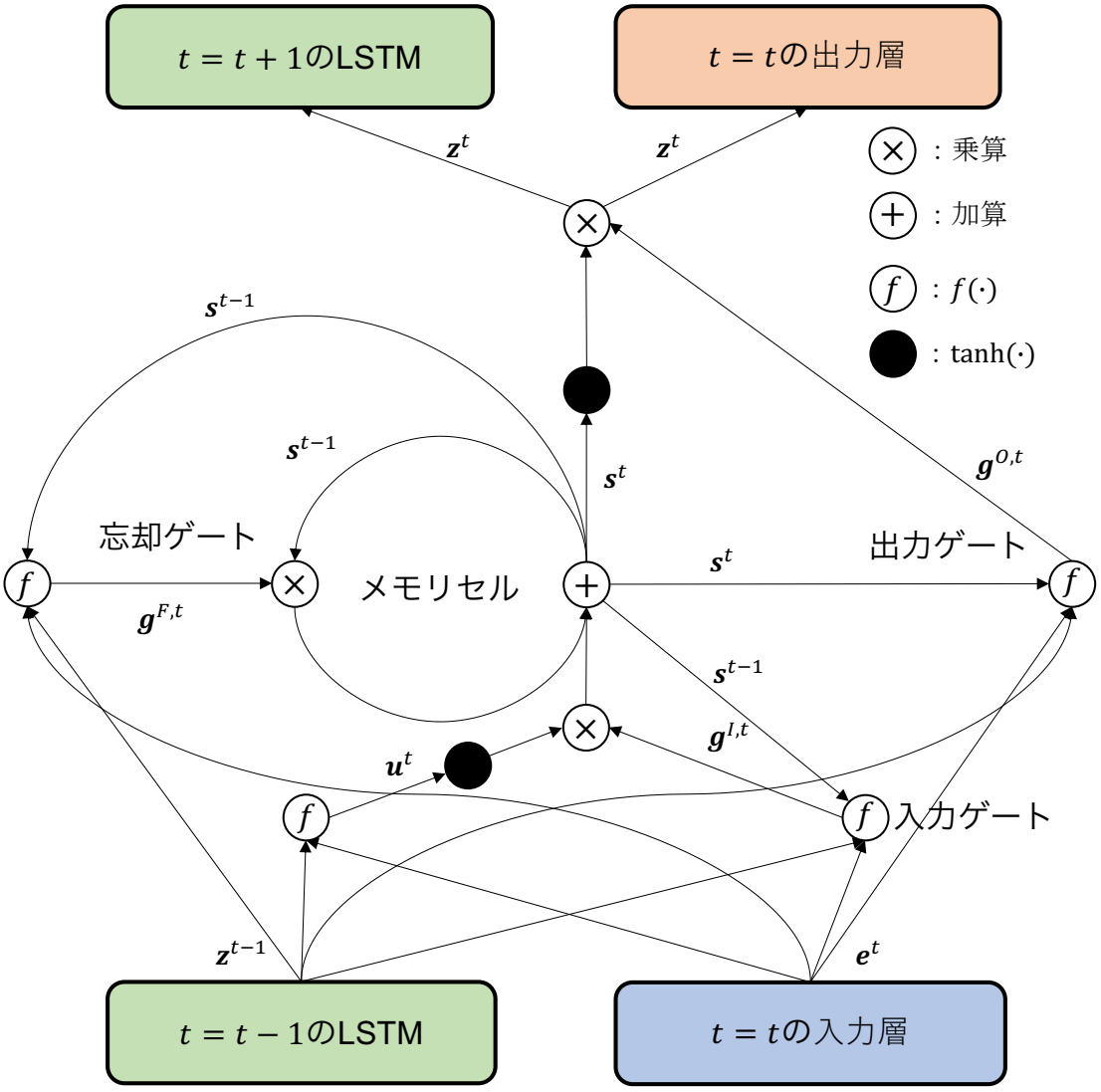


図 2.3: LSTM の内部構成.

ここで，式 (2.4) の $\mathbf{W}^{I,in}, \mathbf{b}^{I,in}$ は時刻 t の入力層の出力にかかる重みとバイアス，$\mathbf{W}^{I,z}, \mathbf{b}^{I,z}$ は前時刻 $t-1$ の LSTM の出力にかかる重みとバイアス，$\mathbf{W}^{I,s}, \mathbf{b}^{I,s}$ は前時刻 $t-1$ のメモリセルの出力にかかる重みとバイアスを表す．LSTM の出力 $\mathbf{z}^t$ は式 (2.5) で求める．LSTM の出力は，後述する出力ゲートの出力 $\mathbf{g}^{O,t}$ とメモリセルの出力 $\mathbf{s}^t$ の乗算で求められる．

$$\mathbf{z}^t = \mathbf{g}^{O,t}\tanh(\mathbf{s}^t). \tag{2.5}$$

出力ゲートの出力 $\mathbf{g}^{O,t}$ は式 (2.6) で求める．出力ゲートも忘却ゲートと同様に導出するが，出力ゲートのみ前時刻のメモリセルの出力ではなく，現時刻の出力 $\mathbf{s}^t$ を加算する．

$$\mathbf{g}^{O,t} = f(\mathbf{W}^{O,in}\mathbf{e}^t + \mathbf{b}^{O,in} + \mathbf{W}^{O,z}\mathbf{z}^{t-1} + \mathbf{b}^{O,z} + \mathbf{W}^{O,s}\mathbf{s}^t + \mathbf{b}^{O,s}). \tag{2.6}$$

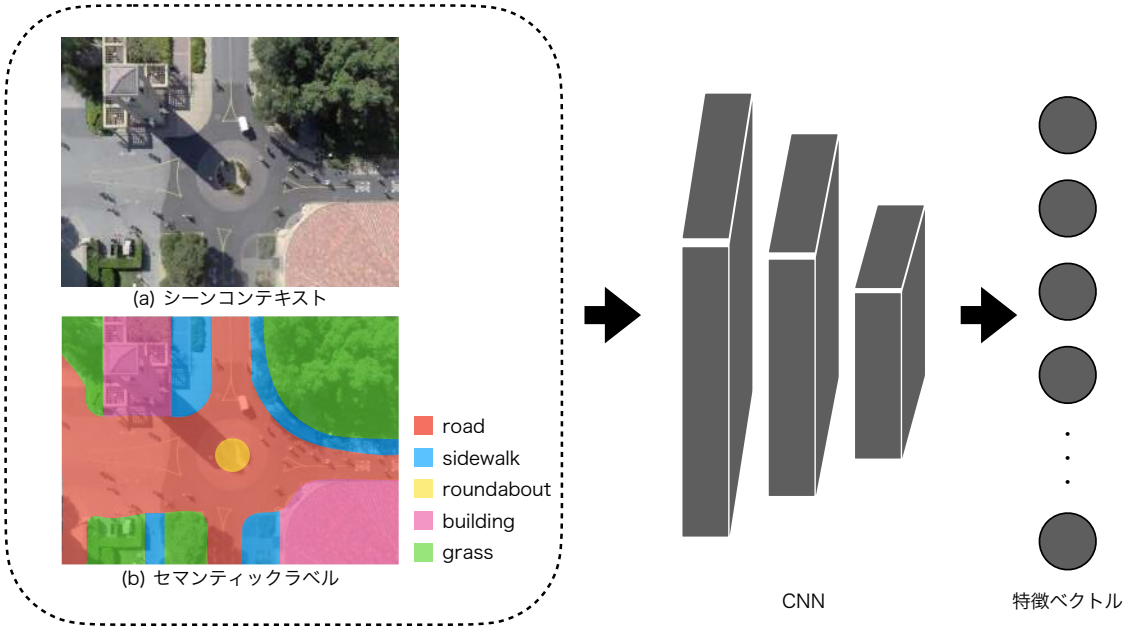


図 2.4: CNN による環境情報の抽出例.

ここで，式 (2.6) の $\mathbf{W}^{O,in}, \mathbf{b}^{O,in}$ は時刻 t の入力層の出力にかかる重みとバイアス，$\mathbf{W}^{O,z}, \mathbf{b}^{O,z}$ は前時刻 $t-1$ の LSTM の出力にかかる重みとバイアス，$\mathbf{W}^{O,s}, \mathbf{b}^{O,s}$ は時刻 t のメモリセルの出力にかかる重みとバイアスを表す．各ゲートの値 $\mathbf{g}^{F,t}$, $\mathbf{g}^{I,t}$, $\mathbf{g}^{O,t}$ は活性化関数 f をシグモイド関数にすることで値域を $[0,1]$ に制約する．メモリユニットの出力 $\mathbf{z}^t$ は次時刻の 3 つのゲートを制御する役割の他に，出力層への入力及び，次時刻 $t+1$ の LSTM へ入力する役割を持つ．

2.1.3 CNN

CNN は主に画像認識に利用するニューラルネットワークで，入力画像に対して畳み込みを複数回行うことで，認識に有効な特徴量を抽出する．経路予測では，周囲の静的環境との衝突を避けるための環境特徴の抽出に用いられている．図 2.4 に CNN による環境情報の特徴抽出例を示す．CNN には映像内のシーンコンテキストや，アノテーションしたセマンティックラベルを入れ，環境に関する特徴ベクトルを得る．この環境に関する特徴ベクトルが予測対象周辺やシーン全体の環境を表現したベクトルになっている．この特徴ベクトルと経路情報を予測モデルへ入力することで，静的物体との衝突を回避した経路を予測できる [8] [75]．一連の流れを式 (2.7) に示す．

$$\begin{aligned} \mathbf{v} &= CNN(I; W_{cnn}), \\ \hat{\mathbf{y}} &= F(\mathbf{v}, \mathbf{x}; W_F). \end{aligned} \tag{2.7}$$

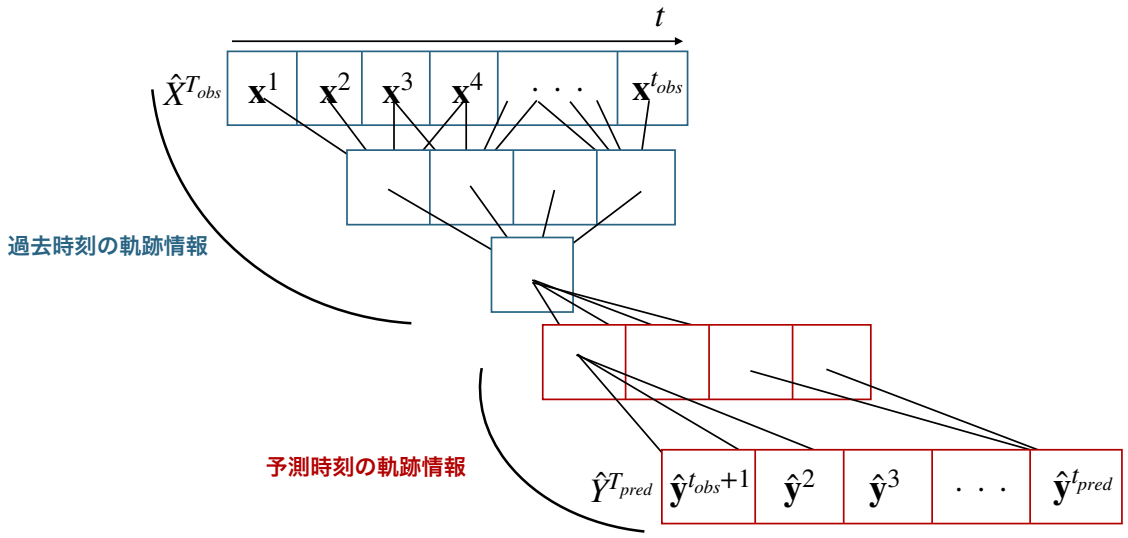


図 2.5: 時間方向に伝播する CNN の例.

ここで，I は入力画像，$\mathbf{v}$ は CNN で得た環境に関する特徴ベクトル，W_{cnn} は CNN の重みパラメータ，$\mathbf{x}$ は過去で観測された経路情報，$\hat{\mathbf{y}}$ は未来で予測する経路情報，$F(\cdot)$ は予測モデル，W_F は予測モデルの重みパラメータを示す.

環境特徴を抽出するために経路予測では CNN が用いられている一方，移動対象の経路の系列特徴を LSTM ではなく，CNN で捉える方法もある．系列特徴を CNN で捉える文献は表 2.1 のモデル欄「CNN」に該当する．時間方向に伝播する CNN [53] の例を図 2.5 に示す．この CNN の利点は，並列計算が可能な点である．RNN は過去時刻の入力特徴を再帰的にモデルへ入力し，未来時刻の特徴を逐次的に出力する必要がある．一方で，時間方向に伝播する CNN は過去時刻の入力をカーネルサイズなど予め設定したハイパーパラメータに従って，時間方向に対し畳み込みフィルタをかけて系列データの特徴量を抽出する．そして，抽出した特徴量をアップサンプリングして未来時刻の特徴として出力する．この構造によりモデルの並列処理が可能となる．そのため，RNN と比べて推論速度が速く，モデルパラメータを低減することができる [37].

2.2　インタラクションを考慮した経路予測手法

本節では，図 2.1 や表 2.1 に記載されている各カテゴリに属する経路予測手法及び，各予測手法の特徴について述べる．また，本節は各文献の数式等は各論文で記載される状態のまま説明する.

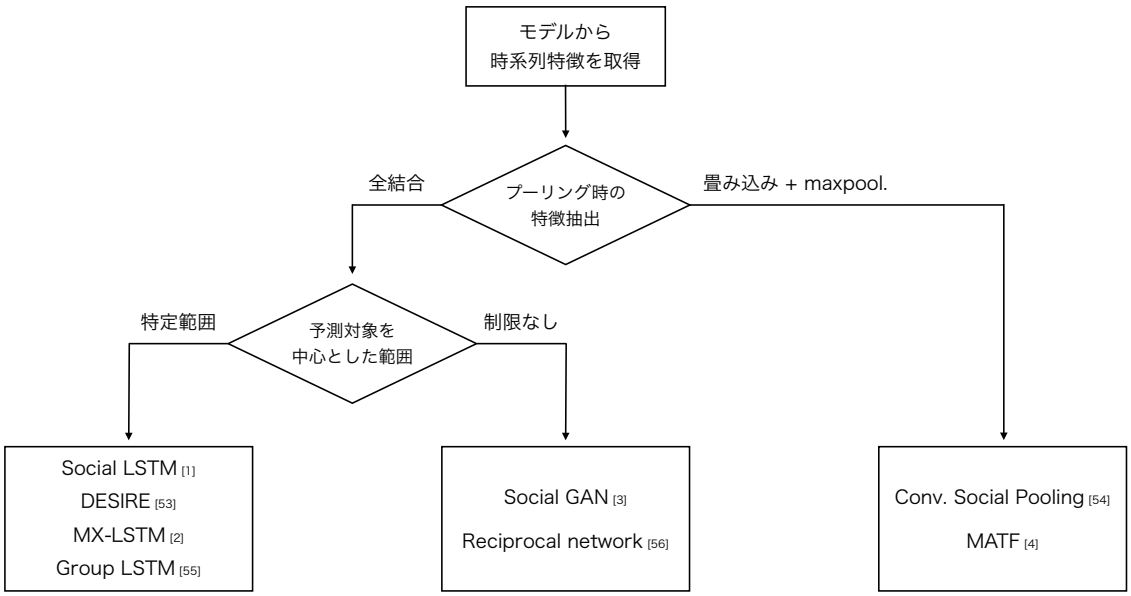


図 2.6: プーリングモデルに基づくアプローチの違い.

2.2.1 プーリングモデルに基づくアプローチ

プーリングモデルに基づくアプローチの違い及びプーリングの概略図をそれぞれ図 2.6 と図 2.7 に示す．プーリングモデルに基づくアプローチでは，予測対象を中心として周囲の空間を表現したグリッドを作成し，これに予測対象周辺の個々の他対象に関する特徴を埋め込む．埋め込んだ特徴は全結合または畳み込みで特徴抽出する．畳み込みの場合は畳み込み処理と maxpool. を繰り返しながら移動対象間のインタラクションに関する特徴を抽出する方法がある．全結合の場合は，特定範囲内の他対象のみの特徴を抽出する方法と，範囲の制限がない，つまりシーン全体にいる他対象の特徴を抽出する方法に分かれている．特定範囲内の他対象のみインタラクションを考慮する方法では，範囲内なら 1，範囲外なら 0 で定義する指示関数で明示的にインタラクション対象の情報を保持する．一方で，範囲の制限がない方法では他対象の特徴から maxpool. でインタラクションを保持する．予測対象周辺の個々の他対象に関する特徴をグリッドに埋め込んだ後，それぞれの予測モデルで提案する異なるプーリングが用いられる．本節ではこのようなプーリングモデルに基づく予測手法について述べる．

■ Social LSTM

Social LSTM [1] はインタラクションを考慮した DL ベースの代表的な手法である．Social LSTM の概略図を図 2.8(a) に示す．この手法では，映像内の複数の歩行者の未来の移動経路を同時に予測することを目的としており，歩行者同士の衝突を避けるために Social pooling layer (S-pooling) を提案

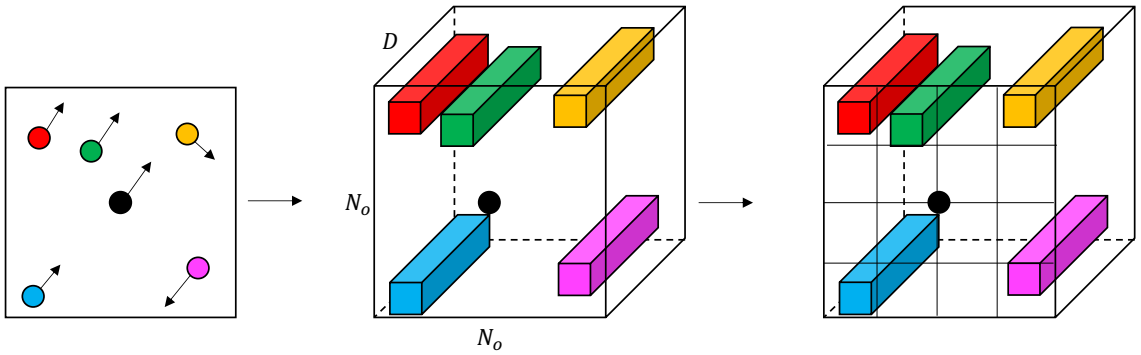


図 2.7: プーリングモデルの概略図．予測対象が黒点を中心に，様々な色で表現されている他対象に関する特徴を $N_o \times N_o \times D$ で表現したグリッドにそれぞれ埋め込む．

している．S-pooling の概略図を図 2.8(b)，S-pooling に他対象の特徴を埋め込む式は式 (2.8) に示す．

$$H_t^i(m,n,:) = \sum_{j \in N_i} \mathbf{1}_{mn}[x_t^i - x_t^j, y_t^i - y_t^j] h_{t-1}^j. \tag{2.8}$$

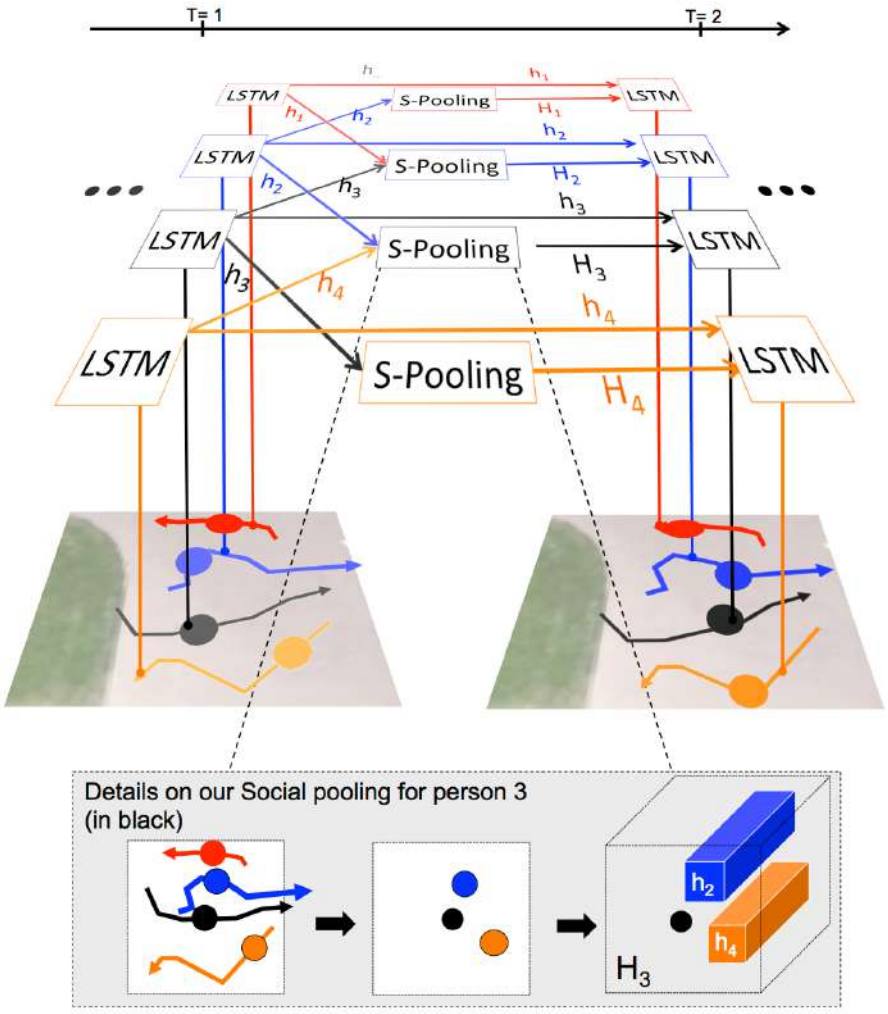
ここで，i は予測対象のインデックス，j は他対象のインデックス，t は時刻，mn は特定範囲のグリッドセル，(x,y) は絶対座標，$\mathbf{1}_{mn}[x,y]$ は指示関数，h は LSTM の内部状態，$H_t^i(m,n,:)$ は他対象の特徴が埋め込まれるグリッドを表す．S-pooling では，指示関数に従い予測対象 i を中心とした特定範囲内の全他対象 j の特徴 h_{t-1}^j をグリッド $H_t^i(m,n,:)$ に埋め込んでいる．図 2.8(b) の例では，黒点を予測対象とした時，特定範囲内にいる青色，黄色及び橙色の他対象の特徴量を空間 $\times$ 次元数で表現されたグリッドに特徴を埋め込んでいる．Social LSTM では，この他対象の特徴が埋め込まれたグリッド $H_t^i(m,n,:)$ を式 (2.9) の全結合層 ϕ へ入力することで，歩行者間のインタラクションを表現している．

$$a_t^i = \phi(H_t^i; W_a). \tag{2.9}$$

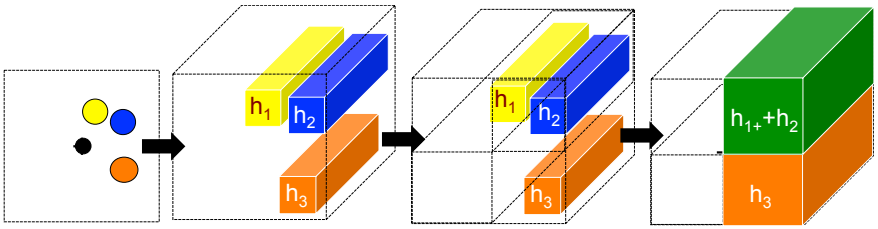
ここで，a_t^i はインタラクションを表現したベクトル，W_a は全結合層の重みを示す．そして，式 (2.10) のように，S-pooling 後のベクトル a_t^i と前時刻の LSTM の出力 h_{t-1}^i，座標情報が埋め込まれた特徴 e_t^i を連結し，連結した情報を LSTM へ入力する．

$$h_t^i = \mathrm{LSTM}(h_{t-1}^i, e_t^i, a_t^i; W_l). \tag{2.10}$$

ここで，W_l は LSTM の重みである．Social LSTM は，式 (2.10) より LSTM の内部状態に歩行者同士の空間的関係を保存することで，衝突を避けるような動きを考慮した経路予測を実現している．



(a) Social LSTMの概略図



(b) Social Poolingの概略図

図 2.8: Social LSTM の概略図．文献 [1] より引用及び改変.

■ DESIRE

Lee ら [54] は，GRU エンコーダ/デコーダによる経路予測手法 Deep Stochastic IOC RNN Encoder-decoder framework (DESIRE) を提案している．DESIRE は，移動対象間のインタラクションに加え，

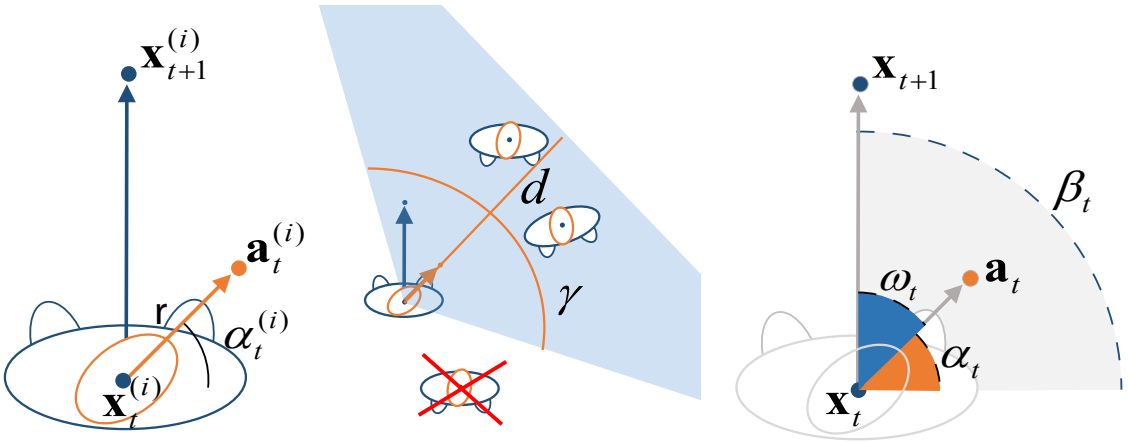


図 2.9: MX-LSTM のプーリング方法の概略図．文献 [2] より引用．

周囲の環境情報に関する特徴ベクトルをネットワークへ組み込むことで交差点や道沿い端の障害物領域を避ける経路を予測している．インタラクションには S-pooling と似た離散グリッドベースが使用されている．また，過去の経路に関する特徴を CVAE [84] にエンコードすることで，周囲の環境に対して尤もらしい複数の経路予測を実現している．さらに，予測経路にランキング付けする ranking module により，経路を反復的に改善することで予測精度をさらに向上させている．

■ Convolutional Social Pooling

Deo ら [55] は，高速道路上で隣接する自動車同士のインタラクション情報を考慮した自動車の経路予測手法であり，インタラクション情報に空間的意味合いを持つ Convolutional Social Pooling を提案している．Convolutional Social Pooling は，Social LSTM の S-Pooling の特徴抽出が単一の全結合層で行われているため，空間的な意味合いに関する特徴が欠落してしまう問題があることから，全結合層の部分を CNN に置換することでこの問題を解決している．

■ MX-LSTM

Hasan ら [2] は，歩行者の経路情報に加え，歩行者の頭部方向の情報を予測の手がかりとして取り入れた MiXing LSTM (MX-LSTM) を提案している．MX-LSTM のプーリング方法の概略図を図 2.9 に示す．MX-LSTM では，人間が視野角内に存在するものに対して着目する点から，頭部の向きを中心とした視野角内にいる他対象のみを考慮してプーリング処理を行っている．MX-LSTM のプーリング処理を式 (2.11) に示す．

$$H_t^i(m,n,:) = \sum_{j \in VFOA_i} \mathbf{h}_t^j. \tag{2.11}$$

ここで，$j \in VFOA_i$ は予測対象の視野角内にいる他対象のインデックスを示す．視野角内の他対象の推定方法は，予測対象の頭部方向及び，他対象までの距離情報で求めている．最終的に経路，頭部方向及び，インタラクション情報を 1 つの LSTM へ入力することで，視線情報内にいる他対象に対して衝突を避けるような経路を予測している．また，視線情報を任意に変更することで，任意方向に向かった経路予測を可能としている．

■ Group LSTM

Group LSTM [56] は友人やカップル，家族などの動きには一貫性があると仮定して運動傾向が類似している歩行者同士をグループとみなし経路を予測している．Group LSTM のインタラクションモデルは Social LSTM の S-pooling を改良している．S-pooling が予測対象周辺の個々人の情報をグリッドに埋め込んでいるのに対し，Group LSTM は予測対象が属するグループ以外の個人の情報をグリッドに埋め込むことで，異なるグループとの衝突を避ける経路を予測する．

■ Social GAN

Gupta ら [3] は，Generative Adversarial Networks (GAN) [85] を用いて実際の経路と予測経路を敵対的に学習させることで，予測経路を徐々に実際の経路に近似させる Social GAN を提案している．また，この手法では未来の経路は複数の可能性を持つとして潜在空間に正規分布を基にしたノイズベクトルを連結することで，複数の経路を予測している．Social GAN の概略図を図 2.10 に示す．Social GAN は，予測経路を生成する Generator (生成器) と実際の経路と予測経路の Real/Fake を判別する Discriminator (識別器) の 2 つのネットワークから構成されている．生成器内では LSTM エンコーダ/デコーダ及び，他対象との衝突を避ける Pooling Module (PM) の 2 つで構成されている．エンコーダでは過去の経路情報の特徴抽出を行う．PM では，単一の全結合層で予測対象と他対象間の相対距離に関する特徴ベクトルを取得した後，エンコーダで抽出した特徴とノイズベクトルを連結する．連結した特徴をデコーダへ入力し，この処理を反復的に繰り返すことで複数の経路を予測する．最後に，生成器から出力された予測経路と実際の経路を識別器で Real/Fake を判別するように敵対的に学習させる．ここで，Social GAN のプーリング処理を式 (2.12) に示す．

$$P_i^t = \max_{j \in N_i}(\phi_2([\phi_1(x_j - x_i, y_j - y_i); h_j^t])). \tag{2.12}$$

$[;]$ は連結記号，ϕ は全結合層，max は maxpool. を示す．式 (2.12) より，Social GAN のプーリング処理は予測対象と他対象間の相対距離と他対象の内部状態から，衝突回避に重要な要因となる他対象の特徴のみを抽出している．

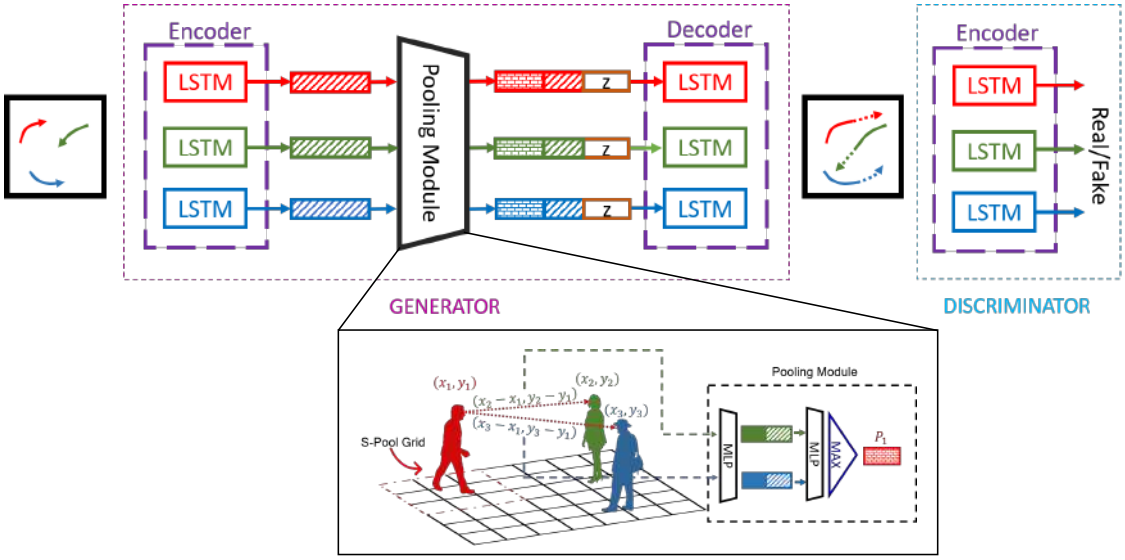


図 2.10: Social GAN の概略図．文献 [3] より引用及び改変.

■ MATF

現実シーンにおける経路予測には，移動物体の過去の動きや数，異なる対象とのインタラクション，シーンコンテキストの制約などが要因で困難になる．特に異なる対象とのインタラクションは多様かつ複雑になる．そのため，Zhao ら [4] は異なる対象とのインタラクション及びシーンコンテキスト情報を共同でモデル化する Multi-Agent Tensor Fusion (MATF) を提案している．MATF の構造を図 2.11 に示す．MATF では，まず CNN でシーンコンテキスト情報に関する特徴を得る．次に複数の異なる対象の過去の経路に関する特徴ベクトルを LSTM でエンコードする．その後，対象の位置情報を埋め込む空間的グリッドに LSTM の出力を埋め込む．そして，取得したシーンコンテキスト情報と空間的グリッドをチャンネル方向に連結し，CNN で特徴を結合する．最後に結合した特徴とエンコードした LSTM の出力を加算した後，ノイズベクトルと連結し Decoder LSTM への入力として用いる．この構造により，対象の位置情報とシーンコンテキストの空間構造を保持しながら，複数対象とのインタラクションを捉えることができる．

■ Reciprocal network

Hao ら [57] は，過去の経路から未来の経路を予測できるだけでなく，未来の経路から過去の経路を予測することができる点に着目し，Forward Prediction Network と Backward Prediction Network の 2 つのネットワークを結合した共同学習モデルを提案している．2 つのネットワークは同一モデルを使用しており，Social GAN に実際のシーン画像に関する特徴と深度マップを利用したモデルとなっている．これらはシーン画像におけるコンテキスト情報の重要性と人間の動きがカメラ位置によって異な

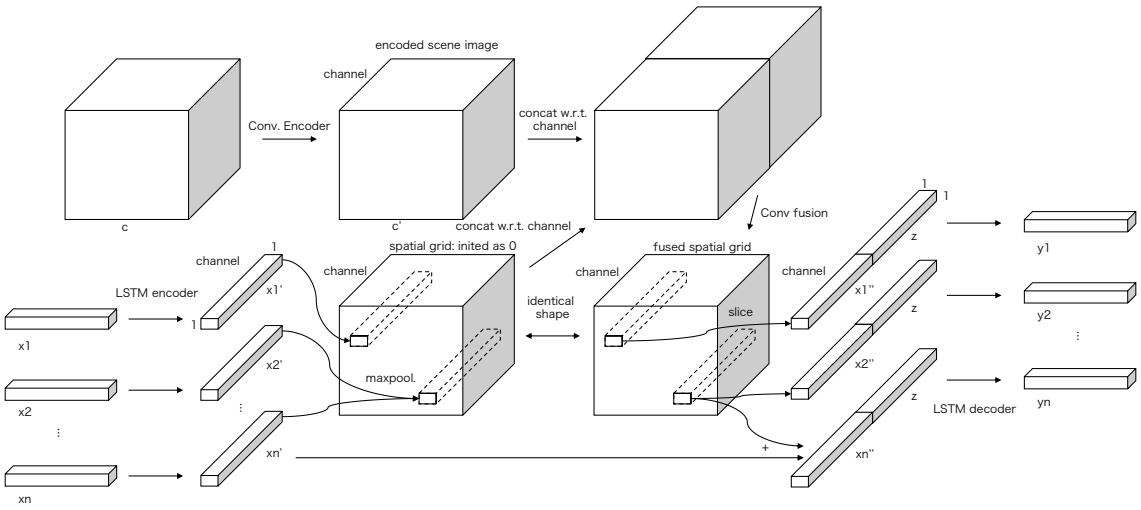


図 2.11: MATF の構造．文献 [4] より引用及び改変．

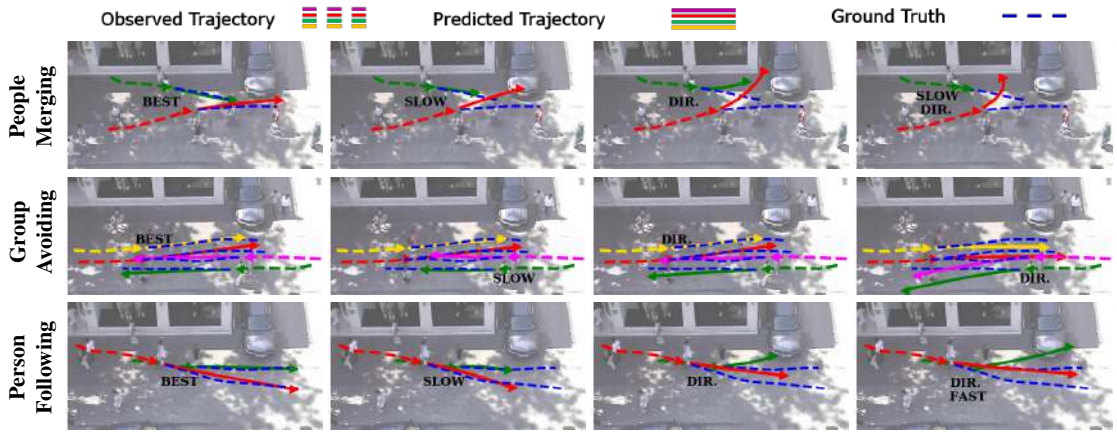


図 2.12: プーリングモデルによる予測結果例．文献 [3] より引用．横軸は複数の予測結果例で左図は最も真値と似た予測経路を辿った例を示す．縦軸はそれぞれ同じ目的地に向かう歩行者のシーン，前方の集団との衝突回避シーン及び，歩行者が他の歩行者を追い越すシーンを示す．

るという点から取り入れられている．また，この手法では学習後の後処理として Backward Prediction Network の学習済みモデルを利用して過去の経路から予測経路に敵対的攻撃を行うことで，与えられた入力またはネットワークの出力に一致するようにネットワークの入力を繰り返し変更する相互攻撃と呼ばれる新しい予測方法を提案している．

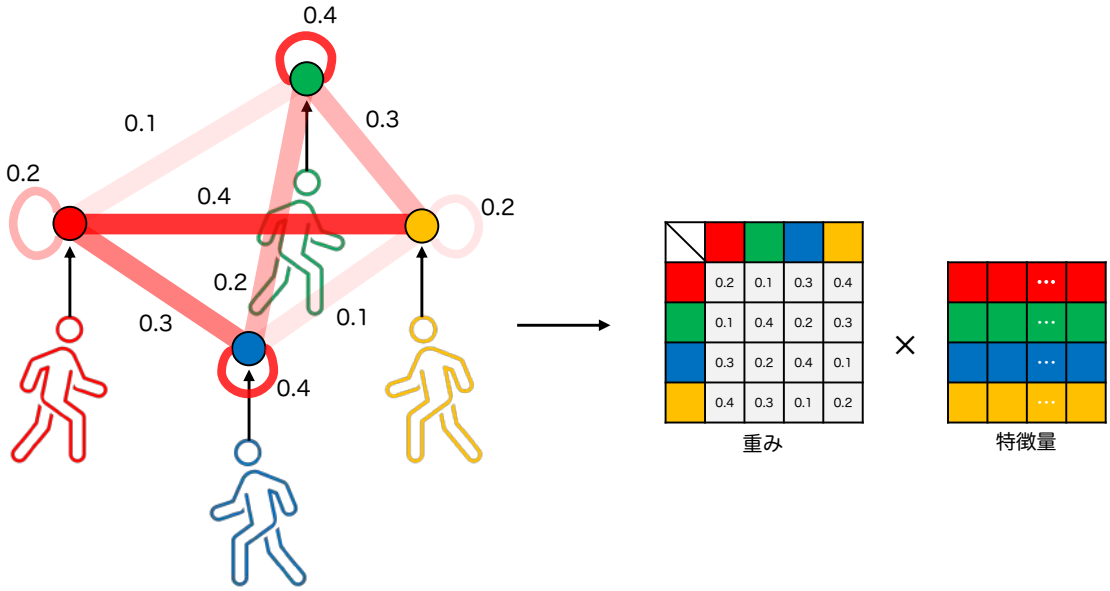


図 2.13: アテンションモデルに基づくアプローチの概略図．赤線は対象間の関係を表しており，色が濃いほど関係が深いことを意味している．赤線の上部の数値は連続的な重み値であり，各対象の特徴量と乗算することでインタラクションを考慮している．

■ プーリングモデルに基づく手法のまとめ

プーリングモデルに基づくアプローチでは，S-pooling のような離散グリッドに基づくアプローチや予測対象が衝突回避に重要な要因となる他対象の特徴を maxpool. で抽出するアプローチが提案されてきた．図 2.12 にプーリングモデルによる予測結果例を示す．図 2.12 より，歩行者が相互に影響を与えることで衝突を避けるために動きを遅くしたり方向を変えたりするなど，人間に近い動きをした経路を予測していることがわかる．しかしながら，プーリングモデルに基づくアプローチは予測対象周辺のグリッド内の他対象に限定したり，maxpool. が最も特徴が大きい他対象とのインタラクションしか考慮しない点から，衝突回避に重要な他対象の情報が欠落してしまう問題が挙げられる．また前者について，予測対象を中心とした特定範囲，つまりグリッドサイズに関するパラメータを予め人間が定義する必要がある．グリッドサイズは予測対象毎に異なり，複雑になるため致命的な欠点と言える．

2.2.2 アテンションモデルに基づくアプローチ

アテンションモデルに基づくアプローチの概略図を図 2.13 に示す．アテンションモデルに基づくアプローチでは，予測対象と予測対象自身を含む個々の他対象との関係を Softmax 関数で合計値が 1 となる連続的な重み値で表現している．この重みはそれぞれの対象の特徴から求められる．求めた重みを用いてそれぞれの対象の特徴量と乗算を行う．特徴量が重みと乗算されるため，重みが大き

い対象の特徴が多く含まれ，重みが小さい対象の特徴が僅かに含まれる．アテンションモデルでは，モデルが獲得した重みの大小から，予測対象が着目または僅かに考慮する対象をモデル自身が選択し，それらの対象との衝突を避ける経路予測が期待できる．アテンションモデルの利点は，移動対象間の複雑なインタラクションをモデル自身が表現できることにある．移動対象間のインタラクションは複雑で人間が自ら定義するのが困難であるが，これをモデル自身が連続的な数値としてインタラクションを表現できるのが利点である．本節ではこのようなアテンションモデルに基づく予測手法について述べる．

■ Social Attention

Vemula ら [15] は，グラフ理論を基盤として，各ノードを映像内の対象の位置情報，エッジを対象同士の空間情報と表現し，表現されたグラフを時空間方向に拡張した Spatial-Temporal Graph [86] で将来の経路を予測している．この手法では，Edge RNN，Attention Module 及び，Node RNN の 3 つのモデルで構成されている．Edge RNN では，空間的意味を持つエッジ (spatial edge) と，時間方向へ伝播するノード自身のダイナミクス (temporal edge) の 2 つをモデル化する．各モデルは，単一の全結合層で得た特徴を LSTM へ入力し，空間と時間の 2 つの特徴ベクトルを取得する．Attention Module では，Edge RNN の各出力から誰にどの程度着目しているかに関するアテンションを求める．Attention Module の処理を式 (2.13) に示す．

$$
\begin{aligned}
H_v^t &= \sum_{i=1}^{m} \frac{\exp(\mathrm{score}(h_{vv}^t, h_{vi}^t))}{\sum_{j=1}^{m} \exp(\mathrm{score}(h_{vv}^t, h_{vj}^t))} h_{vi}^t, \\
\text{where } \mathrm{score}(h_{vv}^t, h_{v\cdot}^t) &= \frac{m}{\sqrt{d_e}} (W_1 h_{vv}^t, W_2 h_{v\cdot}^t).
\end{aligned}
\tag{2.13}
$$

ここで，v は予測対象のインデックス，h_{vv} は temporal edge における LSTM の内部状態，$h_{v\cdot}$ は spatial edge における LSTM の内部状態，H_v は h_v^t の加重和で計算する出力ベクトル，W は全結合層の重み，m は spatial edge の数，d は LSTM の次元数を示す．$\mathrm{score}(h_{vv}^t, h_{v\cdot}^t)$ では LSTM の次元数が大きい内積注意をすると，誤差逆伝播する際に勾配値が極小値しか返さなくなり精度が悪化するのを防ぐために，[52] に触発されて $\sqrt{d}$ で内積値にスケーリングを行っている．アテンションを求める際は内積注意 [87] を適用し，Softmax 関数で他対象に関する重み付けを行う．そして，式 (2.14) に示すように temporal edge の特徴ベクトルと連結した後に単一の全結合層 ϕ で固定長のベクトル a_v^t を抽出する．

$$
a_v^t = \phi(\mathrm{concat}(h_{vv}^t, H_v^t); W_{node}^h). \tag{2.14}
$$

ここで，W_{node}^h は全結合層 ϕ の重みである．最後に，Node RNN でノード情報と Attention Module の出力を連結し，LSTM へ入力することで時空間のアテンションを考慮した経路予測を行う．

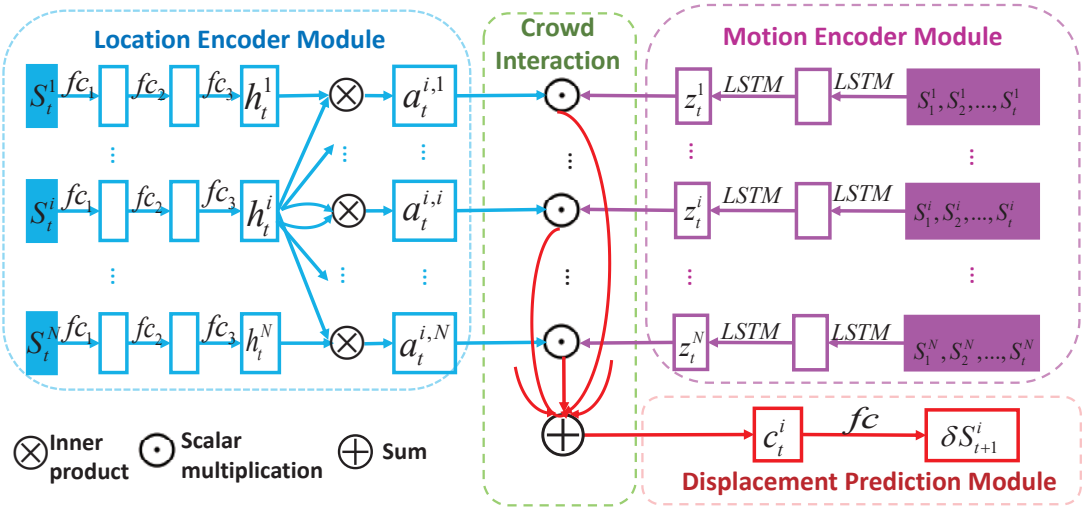


図 2.14: CIDNN の概略図．文献 [5] より引用．

■ CIDNN

Crowd Interaction Deep Neural Network (CIDNN) [5] は，密集した映像における移動対象の行動による危険度をアテンション機構で推定し，行動の特徴に重み付けしている．CIDNN の概略図を図 2.14 に示す．CIDNN は予測対象の位置情報をエンコードする Location Encoder Module，予測対象の移動情報をエンコードする Motion Encoder Module 及び，これらの特徴から将来の経路を予測する Displacement Prediction Module の 3 つのモジュールから構成される．まず，Location Encoder Module では，映像内の複数対象の過去最終時刻の位置情報から 3 層の全結合層で位置情報に関する特徴量を求める．次に，予測対象 (図 2.14 中の S_t^i) と全他対象に対し内積をとり，Softmax 関数にかけることで他対象の特徴に重み付けを行う．Motion Encoder Module では，複数対象の過去の経路を 2 層の LSTM でエンコードする．Location Encoder Module と Motion Encoder Module で得た特徴を連結し，予測対象と他対象に関する加重平均を行う．最後に Displacement Prediction Module で次時刻の予測経路を出力する．求めた予測経路を Location Encoder Module と Motion Encoder Module に逐次入力として用いることで，長期的予測を可能としている．

■ GD GAN

群衆シーンにおける移動対象間のインタラクションは，個別や群衆全体ではなくグループレベルで発生すると仮定して，Fernando ら [58] は経路予測とグループ検出を行う Group Detection GAN (GD GAN) を提案している．GD GAN は，モデルから出力された移動対象の予測経路を特徴抽出器に入れた後，t-SNE [88] で次元圧縮を行う．その後，DBSCAN [89] で近くにいる歩行者集団に対してグループ検出を行う．

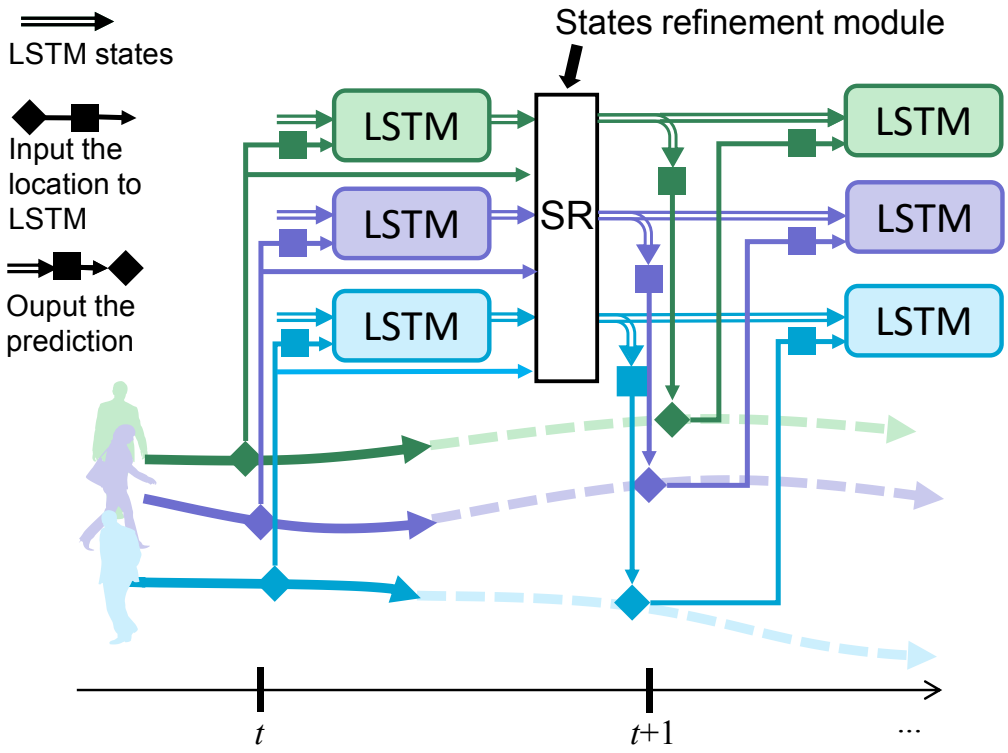


図 2.15: SR LSTM の概略図．文献 [6] より引用．

■ SR LSTM

Social LSTM や Social Attention など，既存研究は 1 時刻前のインタラクション情報を LSTM への入力として用いていた．しかし，現時刻のインタラクション情報を用いることが衝突回避に有効ではないかという主張から，LSTM の出力を State refinement module (SR) へ入力することで，現時刻のインタラクションを考慮する SR LSTM を提案している [6]．SR LSTM の概略図を図 2.15 に示す．SR では，予測対象を中心とした近傍サイズ内にいる他対象との衝突を防ぐためのアテンションを求める Pedestrian-aware attention，他対象の動きから有益な情報を受け取り予測対象自身が経路を選択する Motion gate の 2 つの機構により高精度な経路を予測している．また，SR では LSTM の出力を反復的に更新することで，予測経路の改善を図っている．Pedestrian-aware attention は式 (2.15) に示すように，予測対象と他対象の相対距離に関する特徴ベクトル r と LSTM の出力 $\hat{h}$ を連結して全結合層で特徴抽出した後，Softmax 関数で他対象に対する重み付けを行う．

$$
\begin{aligned}
r_{i,j}^{t,l} &= \phi_r(x_i^t - x_j^t, y_i^t - y_j^t; W^r), \\
u_{i,j}^{t,l} &= w^{a,\mathrm{T}}[r_{i,j}^{t,l}; \hat{h}_j^{t,l}; \hat{h}_i^{t,l}], \\
\alpha_{i,j}^{t,l} &= \frac{\exp(u_{i,j}^{t,l})}{\sum_k \exp(u_{i,k}^{t,l})}.
\end{aligned}
\tag{2.15}
$$

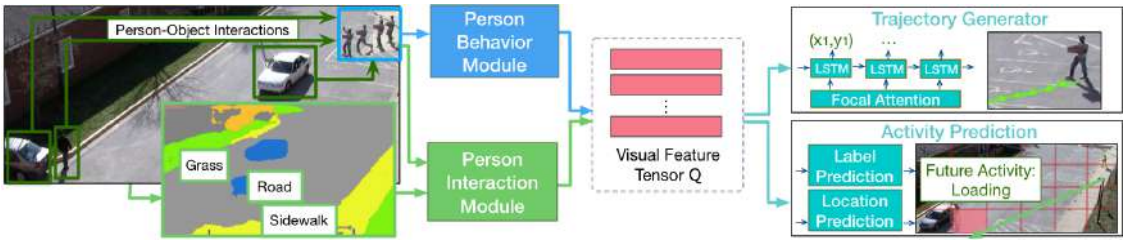


図 2.16: Next の概略図．文献 [7] より引用．

ここで，i は予測対象のインデックス，j は他対象のインデックス，t は時刻，l は反復回数，ϕ_r は全結合層，W^r はその全結合層の重み，w は W^r と異なる全結合層の重みを示す．Motion gate は式 (2.16) に示すように，予測対象と他対象の相対距離に関する特徴ベクトルと LSTM の出力から単一の全結合層で求める．

$$g_{i,j}^{m,t,l} = \delta(W^m[r_{i,j}^{t,l}; \hat{h}_j^{t,l}; \hat{h}_i^{t,l}] + b^m). \tag{2.16}$$

ここで，W, b は全結合層の重みとバイアス，δ はシグモイド関数を示す．また，SR LSTM は Pedestrian-aware attention や Motion gate の導入の有無，近傍サイズなど豊富な Ablation study の結果より提案手法の有効性を示している．

■ Next

人間が未来に何かしらの動作を起こす際，現在の状態に応じて異なる経路を辿ると予想される．例えば，荷物を持っている歩行者が荷物を車に積む際，最短の経路を辿って車のトランクに荷物を積み，自転車に乗っている人は周囲に気を配りながら走行する．このように，歩行者は“自転車に乗っている”や“荷物を運んでいる”など異なる現在の状態によって，それぞれに適した将来の経路を辿ると考えられることから，Liang ら [7] は現在の状態から将来の経路と動作を同時に予測する Next を提案している．Next の概略図を図 2.16 に示す．この手法では，5 つのモジュールで構成されており，それぞれ動作情報や周囲の環境とのインタラクションについての視覚的情報をエンコードする役割を持つ．1) Person Behavior Module では，Person Appearance Encoder と Person Keypoint Encoder の 2 つのエンコーダがあり，映像内の歩行者についての外見情報と骨格情報をそれぞれ事前学習済みモデルを用いてエンコードしている．2) Person Interaction Module では，Person-Scene Encoder と Person-Object Encoder の 2 つのエンコーダがある．Person-Scene Encoder では，歩行者と周囲の環境物体とのインタラクション情報を取得する．具体的には，映像内の歩道や自動車といった物体や環境情報を事前学習済みのセマンティックセグメンテーションモデルでシーンラベルを取得し，シーンラベルを CNN へ入力することで特徴マップを得る．得た特徴マップに対して，歩行者のクラスラベルでマスク処理を行うことで，歩行者以外の周囲の特徴を取得する．Person-Object Encoder では歩行者と物体との位置関係を対数関数で求める．3) Visual Feature Tensor Q では，過去の経路情報に関

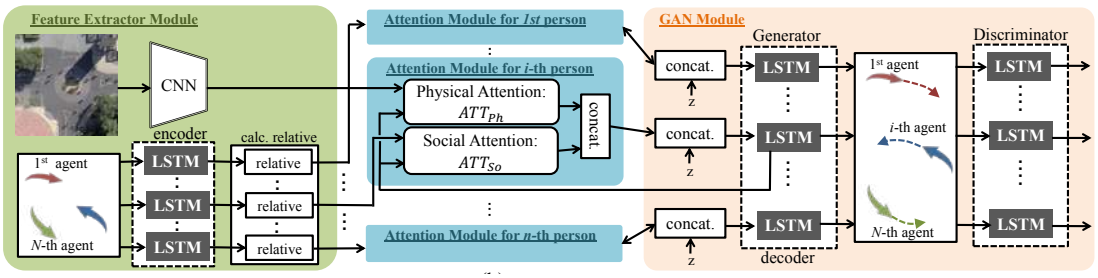


図 2.17: SoPhie の概略図．文献 [8] より引用．

する特徴ベクトルを LSTM エンコーダで求める．同時に 1,2 で得た 4 つのエンコード特徴をそれぞれの LSTM エンコーダで同じ次元にエンコードした後，全てのエンコーダ特徴量をテンソルとして埋め込みする．4) Trajectory Generation with Focal Attention では，3 で得たテンソルと 1 時刻前の経路情報を LSTM デコーダへ入力することで未来の経路を予測する．5) Activity Prediction では，2 で得たシーンラベルと 3 で得たテンソルを利用して，未来最終時刻での未来の動作を予測する．

■ SoPhie

Sadeghian ら [8] は，歩行者同士のインタラクションに加え，建物などの障害物特徴を既存の CNN モデルで抽出し静的物体とのインタラクション情報として用いることで，動的と静的環境とのインタラクションを同時に考慮した SoPhie を提案している．SoPhie の概略図を図 2.17 に示す．Sophie は Social GAN と同様に GAN を用いた経路予測手法であるため，生成器で経路を予測，識別器で Real/Fake の経路の判別を行う．生成器では，過去経路に関する特徴ベクトルの抽出に加え，環境との接触を回避する Physical Attention 及び，歩行者同士の衝突を回避する Social Attention の 2 つのアテンションモデルがある．Physical Attention では，環境に関する特徴ベクトル及び，未来時刻のデコーダの出力を soft-attention [90] に似た構造で特徴を抽出する．Social Attention では，予測対象と他対象とのユークリッド距離から衝突回避のためのアテンション，すなわちインタラクション情報を取得する．Physical Attention 同様に，soft-attention に似た構造で特徴を抽出する．各アテンションの出力と正規分布に従うノイズベクトルを連結しデコーダへ入力することで，将来の予測経路を導出する．導出された予測経路と実際の経路を識別器へ入力し Real/Fake を判別するように敵対的に学習させる．

■ STGAT

Ivanovic ら [9] は，歩行者の衝突を避ける空間情報を時間情報に伝播し，その時空間特徴をエンコードした Spatial-Temporal Attention Network (STGAT) を提案している．STGAT の概略図を図 2.18 に示す．この手法では，過去時刻と未来時刻における経路情報と，各過去時刻で得られる空間情報を伝播する 3 つの LSTM モデルで構成されている．まず，各過去時刻の歩行者の経路情報を LSTM

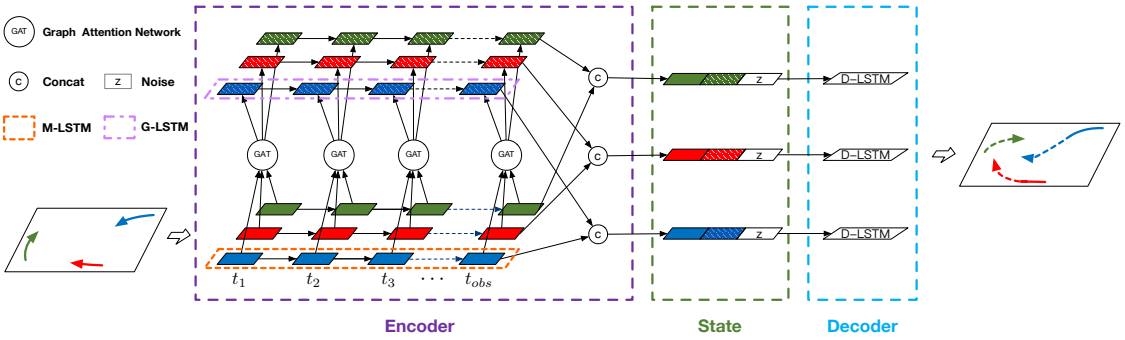


図 2.18: STGAT の概略図．文献 [9] より引用．

へ入力し，経路に関する内部状態を得る．各時刻の LSTM で得られる内部状態から Graph Attention Network [91] で，予測対象の他対象に対するアテンションを取得する．STGAT のアテンション処理を式 (2.17) で示す．

$$\alpha_{i,j}^{t} = f\left(\frac{\exp(\mathrm{LeakyReLU}(a^{\top}[\mathbf{W}m_i^t||\mathbf{W}m_j^t]))}{\sum_{k\in N_i}\exp(\mathrm{LeakyReLU}(a^{\top}[\mathbf{W}m_i^t||\mathbf{W}m_k^t]))}\right). \tag{2.17}$$

ここで，α は注意重み，$||$ は連結記号，$\top$ は転置記号，$\mathbf{W}\in\mathbb{R}^{F'\times F}$ は重み行列，F' は出力の次元数，F は m_i^t の次元数，$a\in\mathbb{R}^{2F'}$ は全結合層の重みベクトル，m は LSTM の内部状態を示す．各時刻 t のアテンションは予測対象と他対象間の空間情報を表し，式 (2.18) に示すように他対象の LSTM の出力と乗算し，全他対象について加算された後に単一の全結合層 σ により空間情報に関する特徴ベクトル $\hat{m}$ を得る．

$$\hat{m}_i^t = \sigma\left(\sum_{j\in N_i}\alpha_{i,j}^t\mathbf{W}m_j^t\right). \tag{2.18}$$

Graph Attention Network を介して得た空間情報に関する特徴ベクトルを時間方向へエンコードする LSTM へ入力することで，時空間インタラクション情報を捉えることができる．最後に，時空間インタラクション情報と経路に関する特徴ベクトルをノイズベクトルと連結し，デコーダへ入力することで将来の経路を予測する．

■ Social BiGAT

既存手法 [3] では，エンコーダの出力に対しノイズベクトルを連結することで複数の経路を予測している．複数の経路を予測できる一方で単純にノイズベクトルを付与するだけでは，分散の大きい経路を予測してしまう．これはランダムにノイズを生成しているため，モデルに予期しないノイズが加わると誤った経路，すなわち分散の大きい経路を予測してしまうためだと考えられる．そのため，Vineet ら [59] は予測した経路とノイズベクトルとの間の潜在的表現を学習する Social BiGAT を提案

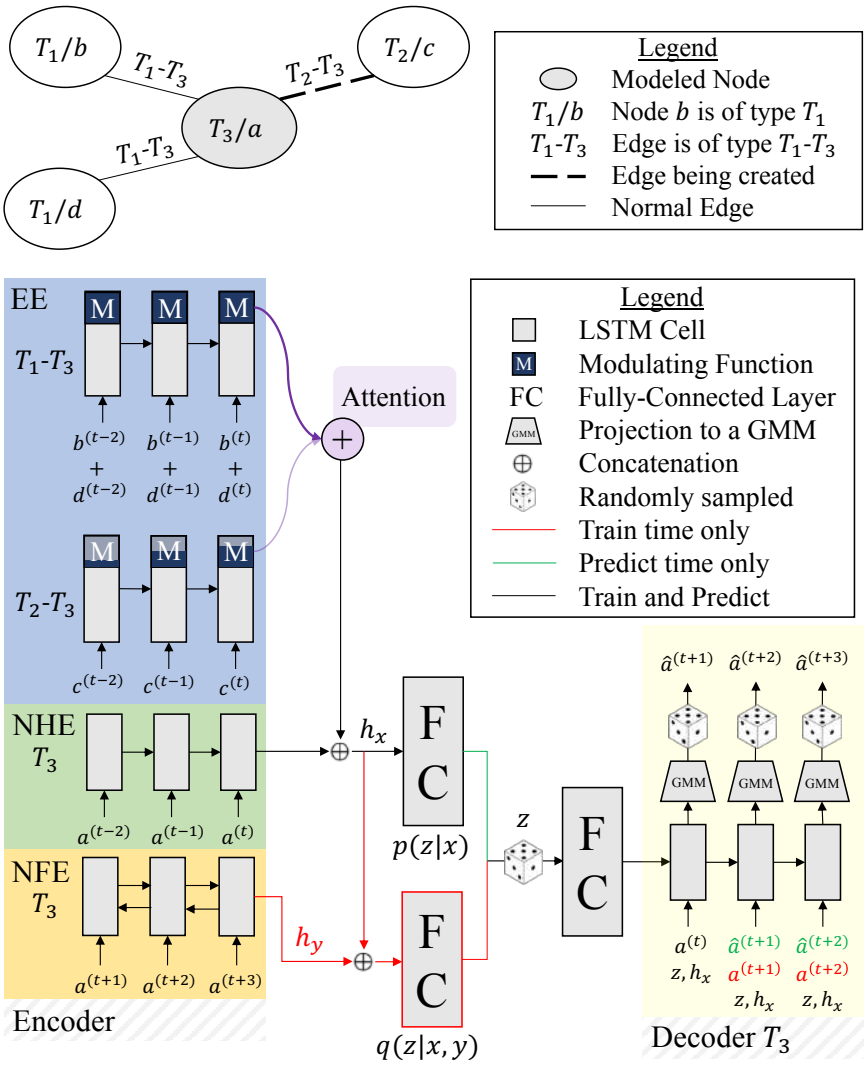


図 2.19: Trajectron の概略図．文献 [10] より引用．

している．具体的には，ノイズベクトルから生成した予測経路を LSTM エンコーダへ入れ，元のノイズベクトルと類似するようにマッピングしている．このようなマッピング方法は Bicycle GAN [92] で提案されており，Social BiGAT は Bicycle GAN のマッピング方法を経路予測タスクへ応用した手法となっている．

■ Trajectron

Huang ら [10] は，映像中の複数対象を動的なグラフ構造によって効率的にモデル化する手法を提案している．Trajectron の概略図を図 2.19 に示す．Trajectron は 3 つのモジュールで構成されている．Encoding Trajectory History (図 2.19 中の NHE) では，過去時刻のノード特徴を LSTM へ入

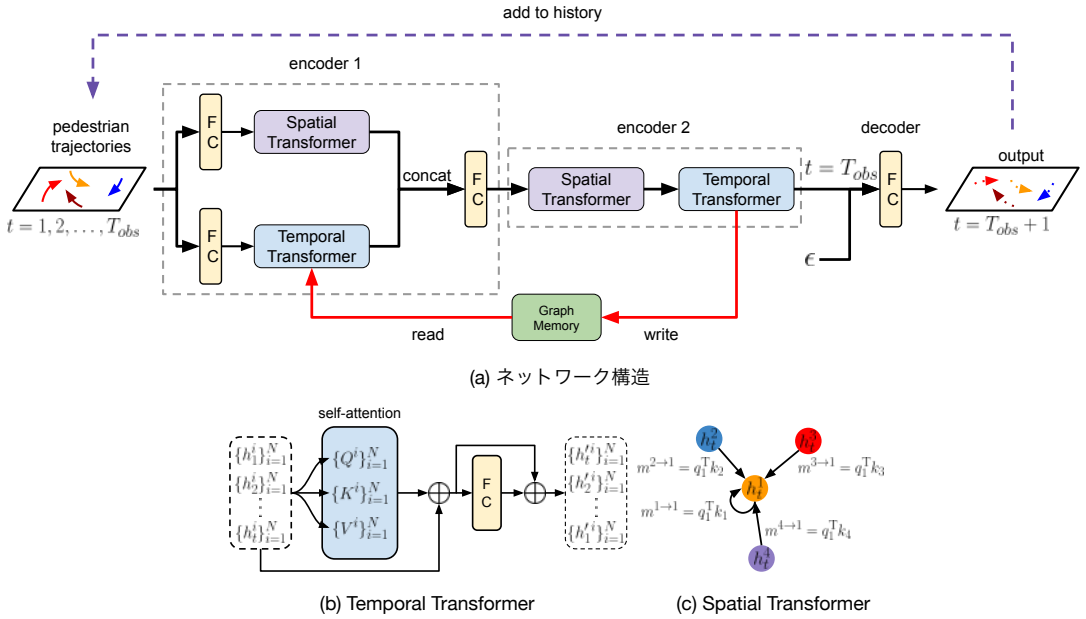


図 2.20: STAR の概略図．文献 [11] より引用及び改変．

力する．図 2.19 中の赤線で表す学習時にノードの未来の真値の経路をエンコードするために，Bi-directional LSTM (BiLSTM) を適用したモデル化を行っている (図 2.19 中の NFE)．Encoding Dynamics Influence from Neighbors (図 2.19 中の EE) では，特定の範囲内にいる全ての隣接ノードとエッジ特徴から，Attention 機構を用いることで重要度の高い，すなわち互いに影響し合うエッジ情報を取得する．Generating Distributions of Trajectories (図 2.19 中の Decoder) では，それぞれで得られた特徴ベクトルから将来の経路を予測する．このとき CVAE で複数の未来の経路の予測及び，Gaussian Mixture Model で学習が進む毎に予測経路を洗練する 2 つの要素を取り入れることで高精度な経路予測を実現している．また，Trajectron をベースに自動車のような異種対象及び，セマンティックな情報を取り入れた Trajectron++が同研究グループによって提案されている [60]．その他にも，移動物体間のインタラクションを潜在的なグラフ構造で捉える EvolveGraph [61] など，グラフ構造を取り入れた予測手法が提案されてきている．

■ STAR

LSTM を用いたアテンションモデルによる予測手法は，歩行者同士のインタラクションを完全にモデル化できていないこと，LSTM が複雑な時間依存性のモデル化が困難 [52] なことの 2 つの限界がある．そのため，Transformer [52] を時間的及び空間的なアテンションへ拡張し，経路予測タスクへ応用した Spatio-Temporal grAph tRansformer (STAR) を提案している [11]．STAR の概略図を図 2.20 に示す．Temporal Transformer では，Transformer と同様に経路のシーケンスを一般化したモデルで

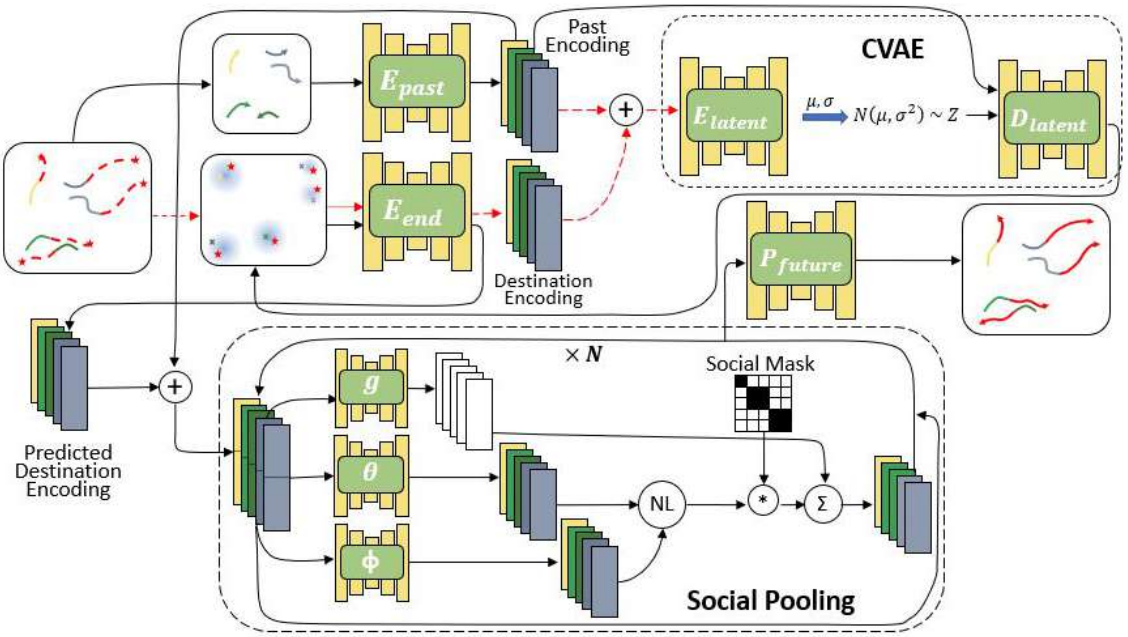


図 2.21: PECNet の概略図．文献 [12] より引用．

ある．具体的には経路情報を Query，Key，Value へ拡張し，各歩行者に対し独立して経路を更新する．Spatial Transformer では歩行者間の空間的インタラクションを抽出する．具体的には，予測対象に関する特徴量を Query，他対象に関する特徴量を Key と Value とし，Source-target-attention のような構造で時刻毎に独立した空間的インタラクションを抽出する．2 つの Transformer で得た情報を連結し全結合層へ入力する．その後，再度 Spatial Transformer と Temporal Transformer へ入れ特徴量を抽出する．抽出した特徴量をノイズベクトルと連結し全結合層へ入れることで次時刻の予測経路を出力する．そして，求めた予測経路を逐次過去経路への入力として使うことで，長期的な経路を予測している．結果より，Transformer を用いることで LSTM を用いた予測手法の予測精度を大幅に超えている．STAR のように空間と時間を別々で Transformer を構築するのではなく，時空間で共同に Transformer を学習することで，STAR の予測精度を上回る AgentFormer [63] が翌年の ICCV で提案されている．

■ PECNet

Mangalam ら [12] は，人間の左右前方など様々な経路の未来最終地点 (エンドポイント) を予測するのが重要であること，エンドポイントに向かう際に他対象の位置情報が重要なことの 2 点に着目し，長期的な経路予測のための Predicted Endpoint Conditioned Network (PECNet) を提案している．PECNet の概略図を図 2.21 に示す．ここで，赤線は学習時のみで伝播することを表す．PECNet では，E_{past} で過去経路に関する特徴量を算出する．次に CVAE 内の D_{latent} で Past Encoding の出力とノ

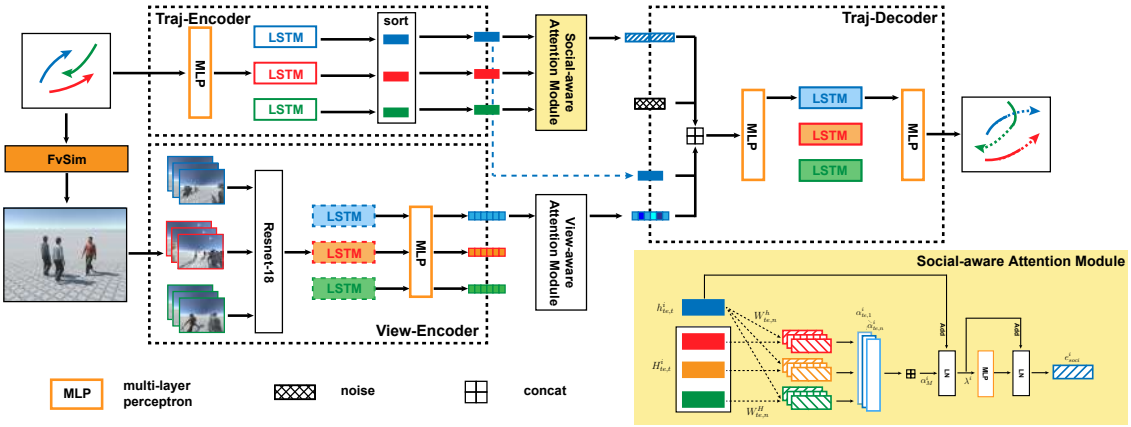


図 2.22: FVTraj の概略図．文献 [13] より引用．

イズベクトルを連結し，エンドポイントを予測する．E_{end} で予測したエンドポイントに関する特徴量を得ると，Past Encoding の出力と連結する (concat encoding)．連結した特徴を Social Pooling 内の g, θ, ϕ の 3 つのパラメータを求める全結合層へ入力する．θ と ϕ は行列演算した後に Softmax 関数で重み付けし，歩行者が隣接するか否かをバイナリ表現にした歩行者 $\times$ 歩行者の Social Mask と乗算する．最後に g と行列演算した後に concat encoding と加算し P_{future} で未来の経路を予測する．PECNet が提案されて以降，エンドポイントを予測する手法が数多く提案されていることから注目度が高いことが窺える [64, 65, 68].

■ DSCMP

Tao ら [62] は，予測対象と他対象間の空間的及び，時間的インタラクションの両方を明示的にモデル化し，将来の複数の経路を確率的に予測する Dynamic and Static Context-aware Motion Predictor (DSCMP) を提案している．DSCMP は他対象の関係性を self-attention [52] に似た構造で捉える Non-local block [93] で予測対象と他対象間のインタラクションを考慮している．また，映像中の静的環境との衝突を避ける経路を予測するために，事前学習された PSPNet [94] を使って映像のセマンティックラベルを抽出している．

■ FVTraj

Bi ら [13] は，First-person View based Trajectory predicting model (FVTraj) と呼ばれる俯瞰視点で撮影された歩行者の経路情報と 1 人称視点の画像から整合性をとる経路予測手法を提案している．FVTraj の概略図を図 2.22 に示す．FVTraj は，FvSim と呼ばれるシミュレータ環境を用いて俯瞰視点で撮影された動画像を 1 人称視点にレンダリングし，1 人称視点で捉えられるエゴモーションを伴う動的なシーンコンテキストが経路予測に有効なことを示している．また，Multi-Head Attention [52]

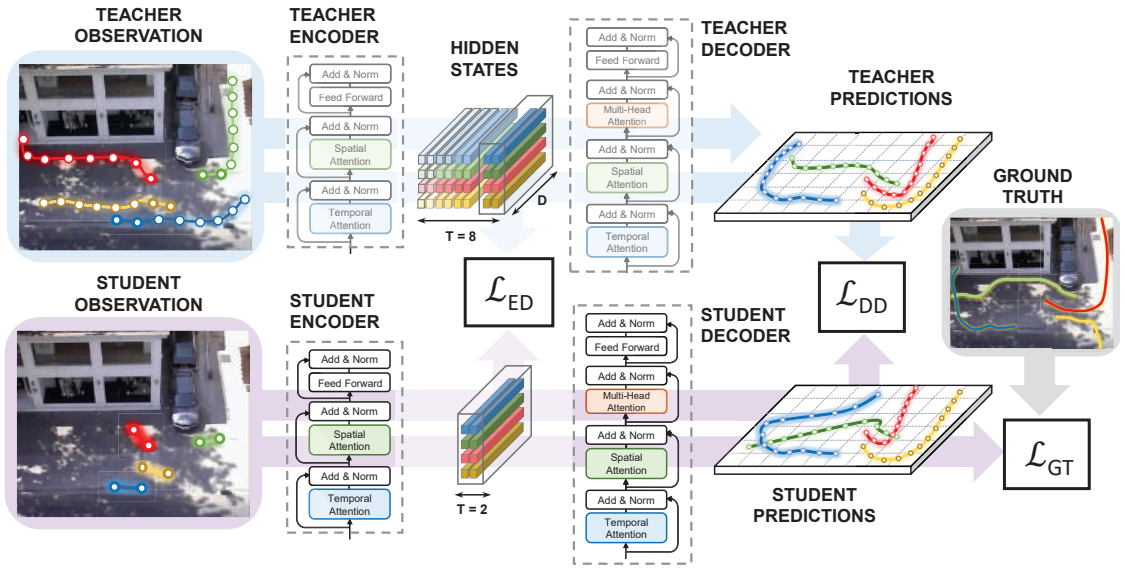


図 2.23: STT-DTO の概略図．文献 [14] より引用．

に基づいて，歩行者間のインタラクションをモデル化する Social-aware Attention Module と，一人称視点画像から過去経路と視覚的特徴との関係を捉える View-aware Attention Module を提案している．Social-aware Attention Module で求められるアテンションは式 (2.19) で表す．

$$
\begin{aligned}
a &= (h^i_{te,t} W^h_{te,n}), \\
b &= (H^i_{te,t} W^H_{te,n}), \\
\alpha^i_{te,n} &= \mathrm{softmax}(\frac{ab^{\mathrm{T}}}{\sqrt{d_{te}}})b.
\end{aligned} \tag{2.19}
$$

ここで，h_{te} は Traj-Encoder 内の予測対象 i に関する LSTM の内部状態，H_{te} は i に関する内部状態の集合，W は全結合層の重みを示す．View-aware Attention Module は式 (2.19) と同様の計算を行う．

■ SGCN

歩行者間の衝突に関するインタラクションにより，衝突回避した経路予測を実現できる一方で衝突を避けるために真値と異なる経路を予測する場合がある．これは，全ての歩行者とのインタラクションを密に捉えることで起こり得る．Shi ら [66] は，遠い歩行者同士や歩行向きが異なるなど，衝突の可能性のない歩行者同士のインタラクションを疎に捉えることが重要と仮定し，歩行者の疎なインタラクションと運動傾向を組み合わせた Sparse Graph Convolution Network (SGCN) による経路予測を提案している．予測対象の経路に影響を与える他の歩行者との疎なインタラクションにより，高性能な経路予測を実現した．SGCN は歩行者を対象としており，自動車や自転車といったマルチクラスに拡張した Multiclass-SGCN [67] が翌年の ICIP で提案されている．

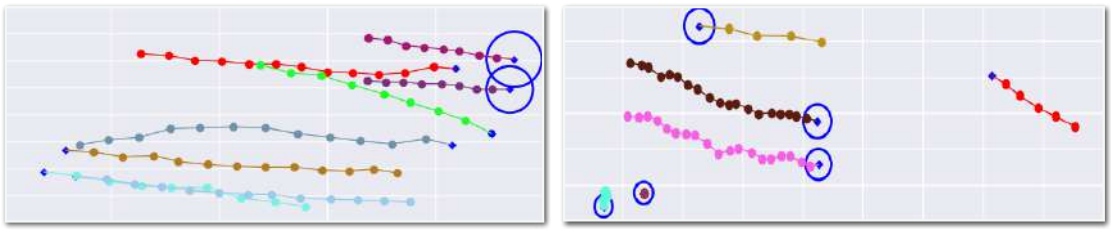


図 2.24: アテンションモデルによる予測結果例．文献 [15] より改変．各図の赤線が予測対象，その他の色が他対象，各対象の青色の点が現在地，青色の円が予測対象の他対象に対するアテンションの大きさを表し，円が大きい他対象に予測対象の予測経路が強く影響することを示す．

■ STT-DTO

先行研究の問題設定では，歩行者の正確に観測された経路が必要になる．監視カメラや自律走行といった実用的なアプリケーションでは，オクルージョン等による誤検出で経路予測を適用できない．適用したとしても，予測モデルにはノイズの多い入力データが与えられることで，予測性能に悪影響を及ぼす．悪影響を最小限に抑えることができる範囲で，観測する経路の長さを減らすことが考えられる．Monti ら [14] は，知識の蒸留 [95] を用いた経路予測手法 Spatio-Temporal Transformer Distilling the Observations (STT-DTO) を提案している．STT-DTO の概略図を図 2.23 に示す．教師ネットワークは，一般的な経路予測の問題設定と同様の長さの観測経路から予測する．生徒ネットワークは，教師ネットワークより少ない観察経路情報から教師ネットワークの経路を模倣するように学習する．各ネットワークは，Attention 機構を介して空間的インタラクションと時間的インタラクションの両方を Transformer ベースの構造で経路を予測する．

■ アテンションモデルに基づく手法のまとめ

アテンションモデルに基づくアプローチでは，予測対象と予測対象自身を含む個々の他対象との関係を連続的な重み値で表現し，この重みをそれぞれの対象の特徴量と乗算することで，インタラクションを表現している．図 2.24 にアテンションモデルによる予測結果例を示す．図 2.24 より，予測対象は前方にいる他対象や対向者に対してアテンション，つまり注意重みを自動で獲得することで衝突を回避する経路を予測できる．また，Transformer がコンピュータビジョンの幅広い分野で応用されており，経路予測でも STAR や FVTraj 及び，その他手法が応用している．さらに，シンプルな Transformer モデルが LSTM による予測手法の予測精度を上回ることができることから [81]，今後 Transformer による予測モデルが爆発的に増えることが予想できる．

2.2.3 その他のモデルに基づくアプローチ

プーリングモデルでは，予測対象周辺の他対象の特徴を空間を表現したグリッドに埋め込むことでインタラクションを表現している．アテンションモデルではモデルが獲得した重みから，予測対象が着目または僅かに考慮する対象をモデル自身が選択することでインタラクションを表現している．一方で，これらに属さないその他のモデルに基づくアプローチもいくつか提案されている．本節では，それらについて述べる．

■ RGM

Choi ら [16] は，歩行者や自動車などの移動対象間の関係と周辺の静的環境との関係の両方を捉えるために，LSTM 内部のゲート構造に触発された Relation Gate Module (RGM) を提案している．RGM の構造を図 2.25 に示す．RGM は，後述する時空間的特徴と LSTM の出力である予測対象の座標値に関する特徴ベクトルを用いる．RGM 内では，Sigmoid 関数及び Tanh 関数を通した特徴量間で乗算することで，時空間的特徴から予測対象に有用な情報を伝播するだけでなく，どの程度重みを与える必要があるかを決定することができる．これは，Sigmoid 関数と Tanh 関数に通す特徴量が同じで，それぞれの関数に通した後の値域がそれぞれ [0, 1] と [-1, 1] となりこれらを乗算すると，0 に近い場合不必要な情報とみなし，1 に近い場合必要な情報とみなされる．つまり，LSTM の忘却ゲートと似た構造となっているため，予測対象に有用な情報を伝播し不要な情報の伝播を防ぐことができる．時空間的特徴は移動対象間及び，移動対象と周辺の静的環境とのインタラクションを考慮したものである．時空間的特徴はプーリングモデルやアテンションモデルのインタラクションとは異なり，2D Convolution と 3D Convolution で各インタラクションに関する特徴量を求めている．まず，画像内を行動する移動対象の空間的行動から，移動対象間の衝突回避に関するインタラクション及び，移動対象と静的物体間のインタラクションに関する特徴量を 2D Convolution で求める．次にこれを全過去時刻で行い，チャンネル方向に各過去時刻で得た特徴量を連結した後に 3D Convolution で時空間的特徴を取得する．

■ PRECOG

Rhinehart ら [70] は，複数の予測対象の経路を予測するために，VAE や GAN とは異なりモデルから得られる尤度推論を用いて経路予測する Estimating Social-forecast Probabilities (ESP) を提案している．また，この手法は推論時に予測対象の“ 前に進む ”や“ 停止する ”などの目標を条件付けする PREdiction Conditioned on Goals (PRECOG) を提案している．PRECOG により，予測対象と他対象間のインタラクションをモデル化した後，予測対象の目標を条件付けると他対象の予測経路が予測対象の目標に従って変化する．

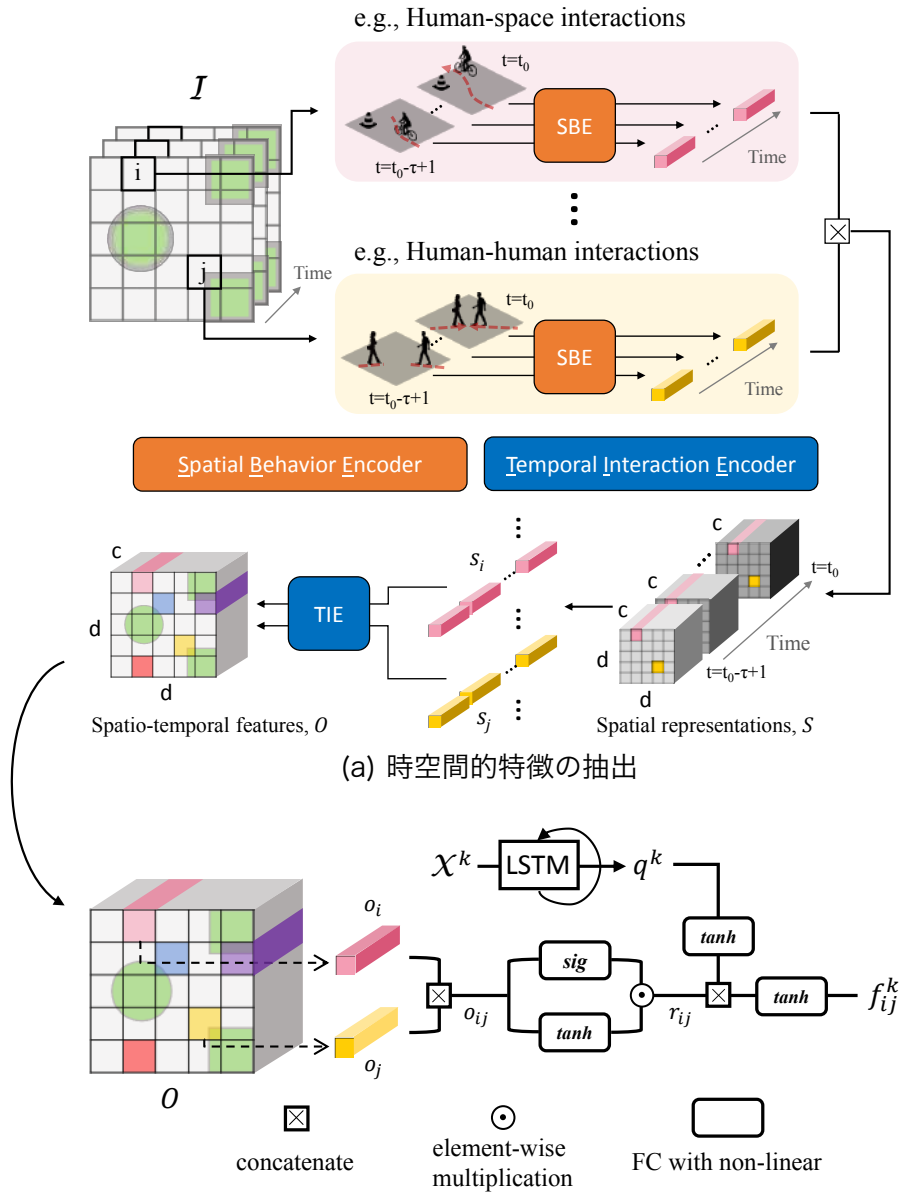


図 2.25: RGM の概略図．文献 [16] より引用及び改変.

■ RSBG

歩行者は同じ目的地や進む方向を考えると，いくつかの類似性を持つ対象が多いことから，Sun ら [17] は歩行者間の関係を調査するグループベースのインタラクションによる経路予測手法を提案している．この手法ではグループかどうかを判断するために，人によるアノテーションが施されている．また，グループのインタラクションには Graph Convolutional Neural Network (GCN) [96] を使ってグループ間の関係性を学習している．RSBG の概略図を図 2.26 に示す．この手法は大きく分けて 3

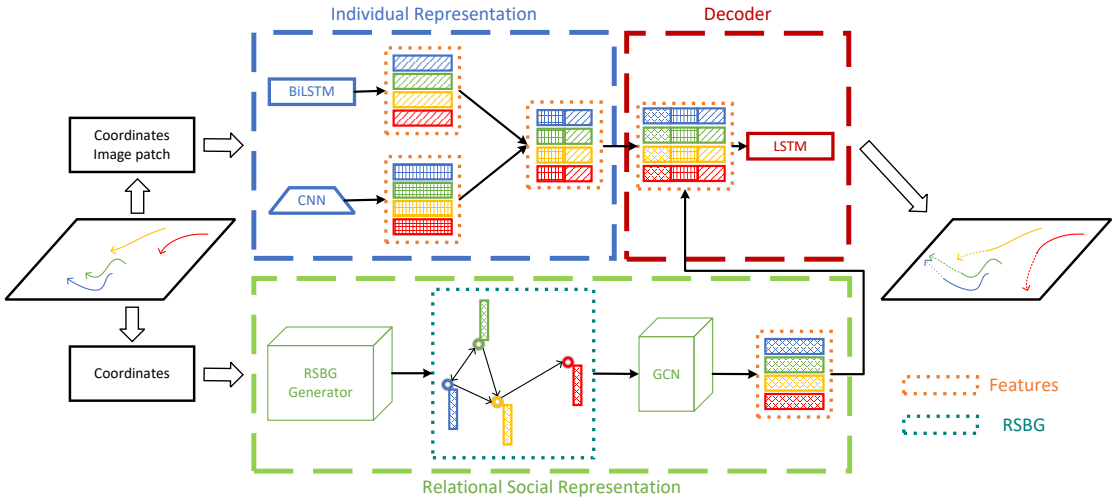


図 2.26: RSBG の概略図．文献 [17] より引用．

つのネットワーク表現で構成されている．1 つ目は Individual Representation である．ここでは予測対象の経路情報に関する特徴量を BiLSTM で取得する．同時に，シーン画像から予測対象を中心としたパッチ領域を取得し，Convolution 層に伝播することで，静的環境に関する特徴量を取得する．その後，それぞれで得た特徴量を連結する．2 つ目は Relational Social Representation である．ここでは，Recursive Social Behavior Graph (RSBG Generator) と呼ばれるモジュールで各人の位置情報から，グループか否かをバイナリ表現にした隣接行列を作成する．作成した隣接行列と位置情報を GCN へ入れ，対象間の特徴量を算出する．最後に Individual Representation と Relational Social Representation の特徴量を連結しデコーダの LSTM へ入力することで将来の経路を予測する．また，経路予測タスクにおいて，未来最終時刻の位置が移動対象の目標を示すのに重要である点から，損失計算時に未来最終時刻の誤差を強調する Exponential L2 Loss を提案することで予測精度をさらに向上させている．

■ Social STGCNN

Mohamed ら [37] は，GCN と Temporal Convolutional Neural Network (TCN) [53] を組み合わせることで，歩行者間の複雑なインタラクションを考慮した時空間グラフ化したモデルを提案している．この手法では，映像中の対象間の相対距離値の逆数を隣接行列とし，グラフのノードと隣接行列から GCN でインタラクションに関する特徴抽出を行い，その後 TCN で予測分布を出力している．また，従来の LSTM による予測手法では経路を予測するためにモデルから出力される予測値を逐次入力値として繰り返す必要があったが，2.1.3 節に述べたように TCN は予測経路を並列に出力することから，推論速度や計算コストを大幅に削減できる．

■ その他のモデルに基づく手法のまとめ

その他のインタラクションを考慮したモデルに基づくアプローチでは，2D や 3D Convolution で時空間的にインタラクションを求める方法や歩行者をグラフとして捉えて GCN でインタラクションを考慮する方法などが提案されている．これらは，プーリングやアテンションモデルに基づくアプローチと比べ発展途上にあるため，移動対象間の関係性を捉える別のアプローチが今後提案されていくと考えられる．

2.2.4 インタラクション以外の課題を扱うアプローチ

プーリングやアテンションといったモデルでは主に LSTM を用いた手法が多く提案されている．一方で CNN を用いた経路予測手法や上記で説明したモデルに分類されない予測手法，実環境における建物や道路などのシーンコンテキストを用いた予測手法といった様々な予測手法が提案されている．本節ではそれらについて述べる．

■ Behavior CNN

Yi ら [71] は過去の経路から CNN を用いて予測経路を直接出力するような Behavior CNN を提案している．Behavior CNN は，歩行者の過去数フレームの移動座標を各チャネルに格納したスパースな 3D データを作成し，Behavior CNN への入力とする．その後，Convolution 及び maxpool. を適用することで入力データをエンコードし，Deconvolution を適用しデコードすることで，予測経路を出力している．また，入り口や障害物などの存在により，特定のシーン中の位置によって異なる歩行者の振る舞いを考慮するために，location bias map と呼ばれるバイアスをエンコーダによってエンコードされたデータの各チャネルに組み込むで，予測精度を向上させている．

■ Visual Path Prediction

Huang ら [72] は，Spatial Matching Network と呼ばる CNN を提案しており，予測対象が存在しているパッチ画像と周辺の微小領域のパッチ画像の類似度を比較することで局所的な領域の報酬を推定している．この手法では，シーン中の微小領域毎に推定したコストからシーン全体のコストマップを作成し，Spatial Matching Network で獲得した報酬と組み合わせることで，衝突の危険が高い領域を回避した経路予測を可能としている．

■ FPL

Yagi ら [29] は，1 人称視点における対面の歩行者の将来の経路を予測する Future Person Localization in First-Person Videos (FPL) を提案している．Yagi らは一人称における経路予測の重要な要素として，

1) 将来のフレームでの対面の歩行者の位置に影響を与えるエゴモーション，2) 対面の歩行者のスケール，3) 将来の位置を予測するための歩行者の姿勢や向きが必要だと主張している．そのため，これら 3 つの情報を用いたマルチストリームの Convolution-Deconvolution モデルを提案している．

■ OPPU

Bhattacharyya ら [73] は，車載カメラ映像に映る人の Bounding Box 情報 (BB) と自車のステアリングや速度といった移動量を用いた予測モデルの On-board Pedestrian Prediction under Uncertainty (OPPU) を提案している．この手法では，上記の情報の他に車載カメラ画像を共に用いている．ネットワーク構造は Odometry Prediction Stream と Bayesian Bounding Box Prediction Stream の 2 つから構成される．まず，Odometry Prediction Stream で過去の移動量と車載カメラ画像から，未来の移動量を推定する．続いて，Bayesian Bounding Box Prediction Stream で過去の同時刻の Bounding Box と移動量から，現在までの状態を保持する summary を算出する．その後，summary と推定した移動量を用いて，未来の Bounding Box とそのバラツキ (不確実性) を予測する．CNN を用いて周辺の環境情報に関する特徴量を車載カメラ画像から抽出し，Bounding Box や移動量と共に LSTM へ入力することで，高性能な経路予測を実現している．

■ Fast and Furious

Luo ら [74] は，俯瞰視点の 3D 点群データを用いて自動車の 3D 検出，追跡及び経路予測を同時に推論するモデルを提案している．この手法では，3D 点群データから自動車を検出することに焦点を当てている．一般的に，追跡や予測といったタスクには自動車の固定 ID が必要になる．そのため，複数フレームにわたる 3D 点群データを使用することでロバストな自動車の検出，追跡及び予測を行っている．また，一般的に点群データは本質的に 3D 空間でスパースな特徴表現であり，ネットワークの計算コストを抑えることができるため，リアルタイムでこれらのタスクを同時に計算することができる．

■ OASE

既存の予測手法で歩行者や自動車などの異なる移動物体の経路を予測する際，異なる対象毎に予測モデルを作成する必要がある．しかしながら，対象の種類が増加するにつれ，扱うモデルの数が増加するため，計算コストの面から現実的とは言えない．そのため，Object Attributes and Semantic Environment (OASE) [75] では歩行者や自動車といった複数対象の種類を対象が保有する属性とみなし，対象の属性情報を one-hot vector としてコンパクトに表現した経路予測手法を提案している．同時に予測シーンに付与されたセマンティックラベルを CNN へ入力することで，予測対象周囲の環境に関する特徴を捉えている．これにより，例えば自動車なら車道を走る，歩行者なら歩道を歩くなど異なる対象に応じて移動する領域の違いを考慮した経路予測を実現している．詳細は 3 章で述べる．

■ Rules of the Road

Hong ら [76] は，俯瞰視点における自動車の将来の経路を予測するために，車線や信号の状態など含む正確かつ詳細なセマンティックマップをエンコードし，複雑なシーンコンテキストを考慮する CNN と GRU による経路予測手法を提案している．この手法では，オートエンコーダモデルを適用し，エンコーダは時間，空間，チャネルの 4D テンソルを幾つかの潜在的内部表現にマッピングし，デコーダではその表現を使用して，将来の自動車の予測分布を出力する．

■ Future Vehicle Localization

Yu ら [77] は，車載カメラ視点における他車の将来のスケールの大きさをマルチストリームな GRU エンコーダ/デコーダで予測している．この手法では，Location-Scale Encoding，Motion-Appearance Encoding，Future Location-Scale Decoding 及び，Future Ego-Motion Cue の 4 つで構成されている．Location-Scale Encoding では，他車の BB 情報から位置に関する特徴を全結合層で抽出した後に GRU へ入力する．Motion-Appearance Encoding では，隣接フレーム間の物体の動きをベクトルで表すオプティカルフロー画像から他車の BB 領域のみを取り出し，全結合層と GRU を介して他車のダイナミクスな特徴を抽出する．Future Location-Scale Decoding では，先述した 2 つの特徴と後述する自車のエゴモーションを GRU へ入力し，将来のスケールを予測する．Future Ego-Motion Cue では，自車のエゴモーションに関する特徴を全結合層で抽出した後に，Future Location-Scale Decoding へ入力する．

■ DTP

Styles ら [78] は，隣接フレーム間の物体の動きをベクトルで表すオプティカルフローから歩行者の行動特徴を予測する Dynamic Trajectory Predictor (DTP) を提案している．また，この論文では歩行者の位置情報を人手でアノテーションするには人的コストがかかる点から，既存の歩行者検出と歩行者追跡手法を利用して歩行者の位置情報を自動でアノテーションし，ラベルのないデータから事前学習を行っている．その後，ターゲットのデータセットへ fine-tuning することで，効率的に学習させている．

■ TPNet

DL を用いた従来手法は，経路を直接回帰することで将来の経路パターンを学習し，複数経路を予測することができる．このようなデータドリブンなアプローチでは，次の 2 つの問題点が挙げられる．1) 右左折などして移動対象が将来辿る可能性が複数あると合理的な経路の予測が困難なこと．2) 環境情報を考慮する際セマンティックマップがエンコードされるが，交通ルールに従うための安全性の保証がなく周囲の静的環境を効率的に組み込むことが困難なこと．これらを解決するために，Fang ら [79] は Trajectory Proposal Network (TPNet) と呼ばれる 2 段階のアプローチを提案している．1 段

階目は将来で辿る可能性のある経路を削減する目的として，CNN エンコーダ/デコーダでエンドポイントを予測し，予測したエンドポイントと過去の経路からカーブフィッティングをして将来候補となる経路を生成する．2 段階目は，1 段階目で候補に挙がった各経路に対してクラス分類を行い，最良な予測経路を出力する．

■ Multiverse

この論文では，人が辿る未来の目的地や経路は複数存在していると仮定し，複数の尤もらしい将来の経路を予測する Multiverse を提案している [24]．また，現実シーンでは単一の目的地や経路しか取得できないことから，CARLA シミュレータ上で現実シーンの情報をマッピングすることで複数の目的地や経路を取得した The Forking Paths Dataset も提案している．データセットの詳細は 2.3 節で説明する．また，同著者らは The Forking Paths Dataset，すなわちシミュレータ環境で作成した歩行者の経路を学習し，それを現実世界の歩行者の経路に転移させて将来の経路を予測する SimAug [80] を提案している．さらに，Li ら [69] は Multiverse が人間の行動パターンに反する時間的な整合性がないことを問題とし，メモリグラフで各時刻で得た経路情報を蓄積し，経路の滑らかさを保証するメモリリプレイアルゴリズムを提案している．

■ E-VMP

Ego-motion Vehicle Motion Prediction (E-VMP) [83] では，車載カメラ視点で歩行者の未来の位置の推定を行う．自車両及び歩行者は時刻毎に動的に動くことが想定されるため，E-VMP では自己教師あり学習の深度推定ネットワーク [97, 98] に基づき，異なる時刻間の入力画像の差分から自車両の将来の視点変化を予測するネットワークを提案している．そして，歩行者の過去の位置情報と組み合わせシンプルな線形モデルにより，将来の歩行者の位置を推定する．

■ Y-Net

歩行者の過去の経路やシーンセマンティックを条件としても，将来の人間の行動は複雑で行動がランダムに変化するため本質的に確率的 [99] である．これは，長期目標などの潜在的決定変数による認識の不確実性と環境要因などのランダムな決定変数から生じる偶然性の両方が原因となり引き起こされる [100]．この二律背反が，長期的予測では将来の不確実性が高くなる傾向がある．Mangalam らは，歩行者の長期的な潜在的目標が経路予測における認識の不確実性を表すと仮定し，認識の不確実性をモデル化により得られた推定値を条件とした Y-Net [82] で経路を予測する．Y-Net では，シーンと歩行者の過去の経路から長期的な目標，すなわちエンドポイントに対する確率分布を複数推定する．さらに，いくつか選択された将来の waypoint に関する分布を推定し，推定したエンドポイントと残りの中間地点に関する明示的な確率マップを推定する．これにより，経路予測における認識の不確実性を表現できる．認識可能なエンドポイントの確率分布，予測可能な waypoint 及び，過去の

経路の分布から将来の経路を予測している．

■ インタラクション以外の課題を扱う手法のまとめ

インタラクション以外の課題を扱うアプローチでは，CNN を用いた予測手法が多く提案されている．特にシーンコンテキストやセマンティックラベルなど，環境情報を捉える手法が初期から提案されている．今後は，次節で述べる経路予測のデータセットのリッチな追加情報より，点群情報やインフラストラクチャなどを利用した手法がより提案されると考えられる．

2.3 経路予測のデータセット

経路予測手法を定量的に評価するために，表 2.2 及び図 2.27 のような様々なデータセットが用いられている．本節では，これらのデータセットの特性について説明する．

2.3.1 鳥瞰視点

経路予測で最も用いられるのは，一般道や市街地を固定カメラやドローンなどを用いて鳥瞰視点で撮影，または鳥瞰視点に変換したデータセットである．これらのデータセットは広域にわたり様々な対象データや詳細なマップ情報を取得できるため，大規模なデータとなっている．

■ ETH/UCY

ETH/UCY [19] [18] は，市街地や大学構内の歩行者を鳥瞰視点で撮影したデータセットである．このデータセットは，現在も多くの手法が使用しており [12] [17]，経路予測の代表的なデータセットとなっている．

■ SDD

SDD [20] は，スタンフォード大学構内をドローンで空撮したデータセットである．SDD には 8 つの異なるシーンで撮影されている．また，pedestrian や car などの異なる物体クラスが 6 種類含まれている．

■ Argoverse

Argoverse [21] は一般道における自動運転用のデータセットである．このデータセットはアメリカのマイアミとピッツバーグの 2 シーンで撮影されており，自車両に取り付けた LiDAR カメラやステレオカメラで他車両の動きを収集している．鳥瞰視点に変換した経路情報以外に車線情報や地図デー

表 2.2: データセットの比較.

データセット名	年代	対象数	映像視点	シーン数	対象種類	追加情報
UCY [18]	2007	0.7K	鳥瞰	3	pedestrian	-
ETH [19]	2009	0.7K	鳥瞰	2	pedestrian	-
SDD [20]	2016	11K	鳥瞰	8	pedestrian, car, biker bus, skateboarder, cart	-
Argoverse [21]	2019	300K	鳥瞰	113	car	車線情報 地図データ センサー情報
Lyft Level 5 [22]	2019	3B	鳥瞰	170,000	pedestrian, car，cyclist	航空画像 セマンティックラベル
inD [23]	2019	13K	鳥瞰	4	pedestrian, car，cyclist	-
The Forking Paths [24]	2020	0.7K	鳥瞰，監視カメラ	7	pedestrian	セマンティックラベル 複数経路情報
PIE [25]	2019	1.8K	車載	53	pedestrian	車両情報 インフラストラクチャ
Apolloscape [26]	2019	81K	車載	100,000	pedestrian, car，cyclist	-
TITAN [27]	2020	645K	車載	700	pedestrian, car，cyclist	対象の行動ラベル 歩行者の年齢グループ
nuScenes [28]	2020	1.4M	車載	1,000	truck, bicycle, car, 20 of others	センサー情報 地図データ 点群情報 エゴモーション
First-Person Locomotion [29]	2018	5K	一人称	87	歩行者	姿勢情報 エゴモーション

タを含むリッチマップが提供されている他，3D tracking タスク用に車載カメラ視点における物体の位置情報が公開されている．

■ Lyft Level 5

Lyft Level 5 [22] は交差点や一般道を中心に自動車や自転車，歩行者などの物体の動きが含まれているデータセットである．対象数は約 30 億と非常に大規模なデータセットで，特に自動車は約 92% のデータが含まれている．また，追加情報として道路情報にセマンティックラベルがあり，交差点の右左折レーンなど交通状況を視覚化したマップが提供されている．

■ inD

inD dataset [23] は，ドイツの交差点の自動車や歩行者の動きをドローンで空撮したデータセットである．このデータセットは 4 つの地点で撮影されており，歩行者や自動車などの複数対象の物体情報がある．

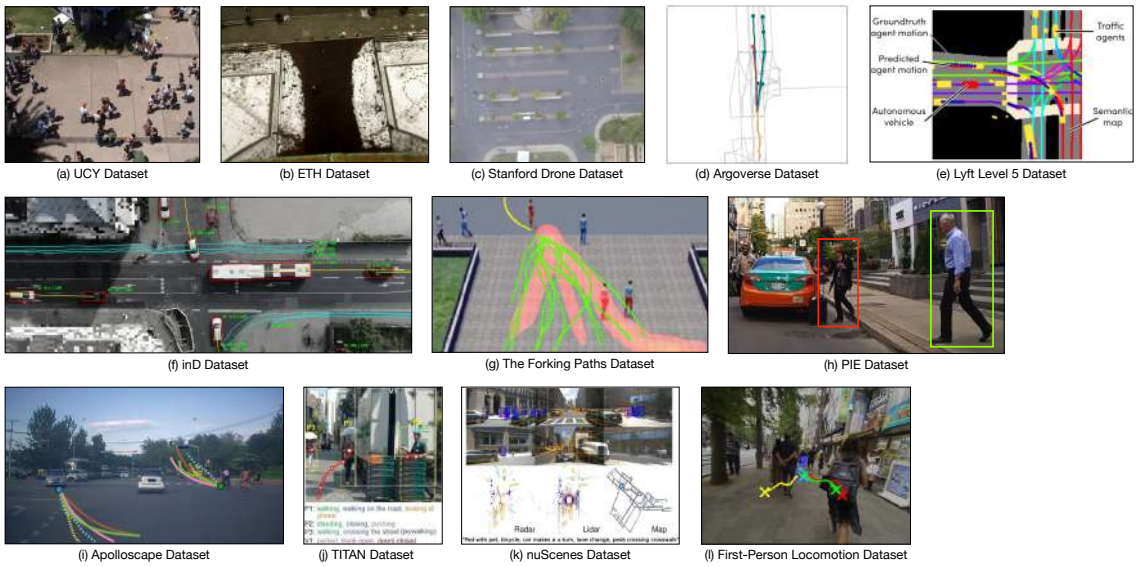


図 2.27: 経路予測のデータセット及び，予測結果例．文献 [18]， [19]， [20]， [21]， [22]， [23]， [24]， [25]， [26]， [27]， [28]， [29] より引用及び改変．

■ The Forking Paths

ActEV/VIRAT [101] と ETH/UCY Dataset [19] [18] を拡張した The Forking Paths Dataset [24] がある．このデータセットは，CARLA シミュレータ上でデータを作成することで，現実シーンが人間の移動経路を 1 つしか観測できないのに対し，複数の移動経路や目的地の設定が可能になっている．また，データ作成のツールも公開されているため，新しい環境を自身で作ることができる．

2.3.2　車載カメラ視点

車載カメラ視点は自動車前方を撮影した映像中の歩行者や自動車の経路を予測することを目的としている．

■ PIE

PIE [25] は，一般道における歩行者の動きを撮影したデータセットである．このデータセットは歩行者情報の他に，カメラと同期した on-board diagnostics センサーを利用して GPS 座標や正確な速度などの車両情報と，信号機や横断歩道などのインフラストラクチャの関連要素を提供している．

■ **Apolloscape**

Apolloscape [26] は中国の北京で様々な照明条件と交通密度を撮影したデータセットであり，車載カメラ映像と 3D 点群データ及び，手動でアノテーションされた経路情報で構成されている．このデータセットには小型車，大型車，歩行者など異なる属性が提供されている．

■ **TITAN**

TITAN [27] は東京で撮影されたデータセットである．このデータセットの特徴は経路情報の他に歩行者や自動車に関する行動ラベルが付与されている点である．例えば，歩行者なら“ walking ”や“ closing ”，自動車なら“ parking ”など各対象が保有する特徴の行動ラベルが付与されている．また，子供や高齢者のような歩行者の属性情報や，二輪車及び四輪車といった自動車の種類に関する情報も提供されている．

■ **nuScenes**

nuScenes [28] は交通量が多いボストンとシンガポールを合計 1,000 の運転シーンを 360 度で撮影した大規模な自動運転用のデータセットである．nuScenes には，約 140 万の RGB カメラ画像と 3D の BB が含まれており，合計 23 種類の物体クラスがアノテーションされている．また，センサー情報や点群情報などの追加情報も豊富で，経路予測の他に点群データによる物体検出や物体追跡などのタスクに有用なデータが提供されている．

2.3.3 一人称視点

一人称視点は，被験者にウェアラブルカメラを装着し，上記の車載カメラのように前方の対象の移動経路を予測することを目的としている．

First-Person Locomotion Dataset [29] は胸部に装着したカメラで，様々な環境で歩き回る歩行者を撮影したデータセットである．撮影された歩行者は約 5,000 人以上で非常に大規模な 1 人称視点によるデータセットとなっている．このデータセットには，エゴモーション，姿勢及び経路データが提供されている．論文中では生の画像データの結果があるが，生の画像データは未公開である．そのため，1 人称視点の画像ベースで経路予測手法を行う際には，自身でデータセットの作成を行う必要がある．

2.4 経路予測の評価指標

本節では，経路予測手法を定量的に評価するために使用する評価指標を簡潔に説明する．

2.4.1 Displacement Error

インタラクションを考慮した経路予測の評価指標は，Alahi ら [1] が提案した予測対象毎に未来時刻間における真値と予測値間のユークリッド距離誤差で評価している．この評価指標は未来時刻間の平均距離誤差の Average Displacement Error (ADE)，未来最終時刻の距離誤差の Final Displacement Error (FDE) の 2 つで評価している．ADE と FDE の概略図を図 2.28(a)，算出式を式 (2.20)，式 (2.21) に示す．ここで，予測対象のインデックスを i，サンプル内の対象数を N，時刻を t，未来最終時刻を T，真値を (x, y)，予測値を $(\hat{x}, \hat{y})$ で示す．

$$ADE = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\sqrt{(\hat{x}_i^t - x_i^t)^2 + (\hat{y}_i^t - y_i^t)^2} \tag{2.20}$$

$$FDE = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(\hat{x}_i^T - x_i^T)^2 + (\hat{y}_i^T - y_i^T)^2} \tag{2.21}$$

2.4.2 Minimum Displacement Error

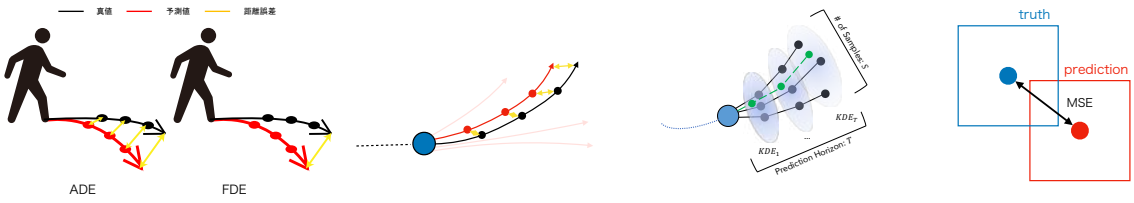
複数経路を予測する評価指標は，Gupta ら [3] が提案している．これは，k 個の複数の経路の中から最良の予測経路，すなわち真値と最も似た予測経路で評価する方法である．最良の経路と真値間の評価には ADE と FDE が用いられ，これらは Minimum ADE (mADE) 及び，Minimum FDE (mFDE) と呼ばれている [24]．mADE と mFDE の概略図を図 2.28(b)，算出式を式 (2.22)，式 (2.23) に示す．ここで，真値を $\mathbf{y} = (x, y)$，予測値を $\hat{\mathbf{y}} = (\hat{x}, \hat{y})$，複数の経路のインデックスを k，複数の経路数を K で示す．

$$mADE = \frac{1}{NT}\min_{\{k=1,\ldots,K\}}\sum_{i=1}^{N}\sum_{t=1}^{T}||\hat{\mathbf{y}}_i^{k,t} - \mathbf{y}_i^t||^2 \tag{2.22}$$

$$mFDE = \frac{1}{N}\min_{\{k=1,\ldots,K\}}\sum_{i=1}^{N}||\hat{\mathbf{y}}_i^{k,T} - \mathbf{y}_i^T||^2 \tag{2.23}$$

2.4.3 Negative log-likelihood

複数経路を予測する場合には，複数の経路を生成し続けるため将来の予測分布とみなすことができる．一方で，ADE と FDE は単一の予測経路を比較するために有用な指標であるが，複数経路の予測で行う場合には，Minimum Displacement Error で述べたように真値と最も似た予測経路のみで評価を行うため，複数の経路を予測する利点を無視してしまう問題がある [102]．そのため，[10][103] では図 2.28(c) に示す Negative log-likelihood (NLL) を用いることで予測分布に対して公平に評価している．



(a) Displacement Error (b) Minimum Displacement Error (c) Negative log-likelihood (d) Mean Square Error

図 2.28: 各評価指標の概略図．文献 [10] より引用及び改変．

2.4.4 Mean Square Error

将来の Bounding Box (BB) を予測する手法では，Mean Square Error (MSE) で予測された BB と真の BB の中心座標で精度比較を行う [73]．MSE の概略図を図 2.28(d) に示す．また，将来の BB を予測する方法では，予測された BB と真の BB の重なり率から適合率と再現率を容易に求めることができるため，モデルの予測精度を評価する F 値で評価することもある．

2.4.5 衝突率

Displacement Error では全てのデータに対して平均を求めている．これらの評価指標では予測値が真値にどれだけ近似できているかを示すことはできるが，衝突回避に関するインタラクション情報がどの予測経路に効果的なのか正確に評価できない．そのため，動的物体と静的物体の 2 つの衝突率に関する評価指標によりインタラクション情報を評価できる．

■ 動的物体との衝突率に関する評価指標

動的物体との衝突率は，映像内の他対象と衝突したか否かを評価する．動的物体との衝突率を分析することで，インタラクションの処理が効果的なのかを定量的に確認できる．動的物体との衝突率は式 (2.24) で求める．

$$
\begin{aligned}
\text{動的物体との衝突率} &= \frac{\sum_i \sum_j c^{ij}}{N\,(N-1)} \\
c^{ij} &= \begin{cases} 1 & \text{if } ||\hat{\mathbf{x}}_t^i - \hat{\mathbf{x}}_t^j||_2 < \theta_d, \forall t \in T \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{2.24}
$$

式 (2.24) の i はサンプル内にいる予測対象のインデックス，c は衝突回数，t は時刻を表している．式 (2.24) より，予測対象 $\exists i \in \mathbb{N}^+, 2 \leq i$ と他対象 $j \neq i$ の各未来時刻 t の予測値 $\hat{\mathbf{x}}_t = (x_t, y_t)$ が 1 時刻でも距離値 θ_d 以内にいる場合に衝突したとみなす．この処理を全サンプルで繰り返し行うことで，動的物体との衝突率を算出する．

複数の経路を予測する際の動的物体との衝突率は，mADE の予測経路を用いて式 (2.25) として表

す．式 (2.25) の k と l は最良の予測経路のインデックスを示す．

$$\text{動的物体との衝突率 (複数経路)} = \frac{\sum_i \sum_j c^{ij}}{N\,(N-1)}$$
$$c^{ij} = \begin{cases} 1 & \text{if } ||\hat{\mathbf{x}}_t^{i,k} - \hat{\mathbf{x}}_t^{j,l}||_2 < \theta_d, \forall t \in T \\ 0 & \text{otherwise} \end{cases} \tag{2.25}$$

■ 静的物体との衝突率に関する評価指標

静的物体との衝突率では，建物などの障害物と衝突したか否かを評価する．静的物体との衝突率に関する評価を行うことで，予測経路の正誤の判別が可能になる．静的物体との衝突率は式 (2.26) となる．

$$\text{静的物体との衝突率} = \frac{\sum_i c^i}{N}$$
$$c^i = \begin{cases} 1 & \text{if } \hat{\mathbf{x}}_t^i \in \mathbf{o}, \forall t \in T \\ 0 & \text{otherwise} \end{cases} \tag{2.26}$$

式 (2.26) より，未来時刻間の予測対象 i の予測値 $\hat{\mathbf{x}}$ が 1 時刻でも障害物領域 $\mathbf{o}$ にいる場合に衝突したとみなす．この処理を全サンプルで繰り返し行うことで，静的物体との衝突率を算出する．

複数の経路を予測する際の静的物体との衝突率は，mADE の予測経路を用いて式 (2.27) として表す．式 (2.27) の k は最良の予測経路のインデックスを示す．

$$\text{静的物体との衝突率 (複数)} = \frac{\sum_i c^i}{N}$$
$$c^i = \begin{cases} 1 & \text{if } \hat{\mathbf{x}}_t^{i,k} \in \mathbf{o}, \forall t \in T \\ 0 & \text{otherwise} \end{cases} \tag{2.27}$$

2.5 まとめ

本章では，経路予測の問題設定及びベースモデル，深層学習をベースとした移動対象間のインタラクションを考慮した経路予測手法，評価に用いられるデータセット及び，評価指標について述べた．深層学習を経路予測に応用した際の，対象間のインタラクションを考慮した手法を 3 つのグループに分割した．1 つ目のプーリングモデルでは，予測対象を中心として周囲の空間を表現したグリッドを作成し，このグリッドに予測対象周辺の個々の他対象に関する位置特徴を埋め込む．その後，maxpool. や特徴抽出など各手法で異なるプーリング処理で他対象との関係を捉えることで，対象同士の衝突を避けるインタラクションを考慮した経路予測が可能になる．一方で，グリッドに他対象に関する位置特徴を maxpool. などで計算すると，衝突回避に関係のない他対象の位置特徴が最大である場合にその他対象のみをプーリング処理してしまうため，本来衝突回避に重要となる他対象の情報が欠

落する場合がある．また，プーリングモデルはグリッドサイズを予め人間が定義する必要があるが，グリッドサイズの適切な大きさは予測対象毎に異なり，複雑になり得るため致命的な欠点と言える．2 つ目のアテンションモデルでは，映像内の予測対象と個々の他対象との関係が連続的な重み値で表現されている．この重みの大小から，予測対象が着目または僅かに着目する対象をモデル自身が選択し，それらの対象との衝突を避ける経路予測が可能となる．3 つ目のその他のインタラクションを考慮する方法では，GCN や Convolution で移動対象間のインタラクションを捉えている．また，本章ではインタラクション以外の課題を扱う方法についても説明した．この方法の多くは CNN をベースとしており，建物といった静的な障害物との衝突回避を目的とした経路予測や 1 人称や車載カメラ映像，異なる入力表現など従来とは異なる問題設定による経路予測手法を提案している．

続いて，経路予測手法の評価に用いられるデータセットについて調査した．深層学習の発展により，経路予測のために歩行者以外の対象やリッチな追加情報を提供するデータセットが増加しつつある．また，シミュレーション環境を使うことで，複数の移動経路や目的地の設定が可能になり，現実シーンでは撮影が困難なデータを作成できるため，自身で問題設定に合わせたデータ構築が可能である．

最後に，経路予測手法で用いられる評価指標について調査した．インタラクションを考慮した経路予測手法では，真値と予測値との距離誤差で精度比較を行っている．複数経路を予測する場合には，複数の予測経路に対して公平に評価するために真値に最良な予測経路による評価や NLL による評価を行っている．将来の BB を予測する方法では，予測された BB と真の BB の中心座標から MSE で評価をすることで予測精度の比較を行っている．さらに，距離誤差が移動対象間のインタラクション情報の効果を評価できない点から，衝突率に関する評価指標が提案されている．

第3章

移動対象の属性と環境情報を導入した経路予測

本章では，歩行者や自動車といった予測対象の種類及び，予測対象の周囲の環境情報を導入した経路予測を提案する．2 章で説明したように，深層学習の発達により CNN 及び LSTM を用いた経路予測が数多く提案されている．正確な予測経路の獲得には様々な情報が必要になる．例えば，歩行者同士の衝突を避けるインタラクションの導入 [1] やシーン中のセマンティックな情報 [54] を用いることで予測精度の向上を図っている．

しかしながら，これらの予測手法では全ての予測対象を同一クラスの予測対象として扱う問題がある．現実のシーンでは歩行者だけでなく，自動車や自転車などの異なる移動物体が存在する環境下で予測を行う必要がある．その際，対象のクラスにより移動速度や移動する領域が異なることが考えられる．そのようなシーンにおいて，歩行者や自動車といったクラスが異なる複数の対象の経路を同時に予測する場合，上記の予測手法では同一クラスの対象として予測しているため，予測対象のクラスに応じた経路を予測することが困難となる．この問題を解決するためには，対象のクラス毎にモデルを作成し，予測を行うことが考えられる．しかしながら，対象のクラスが増加するにつれ，扱うモデルの数が増加するため，計算コストの面から現実的とは言えない．

本章では，予測対象のクラスおよび，予測対象の周囲の環境情報を導入した LSTM による経路予測手法を提案する．具体的には，歩行者や自動車といった予測対象のクラスを対象が保有する属性とみなし，対象の属性情報を one-hot vector としてコンパクトに表現する．そして，予測を行うシーンに付与されたシーンラベルを CNN へ入力することで，予測対象の周囲の環境に関する特徴ベクトルを抽出する．対象の移動情報に加え，属性および環境に関する特徴ベクトルを LSTM へ入力することで，出力として次の時刻の対象が存在する移動情報を得る．予測時には，ネットワークの出力を次の時刻の入力として逐次的に入力することで，クラスが異なる対象に対する速度の違いや移動する領域の違いを考慮した経路予測を実現する．また，移動情報に 2 つの連続する座標の差から得られる相対座標を使用する．相対座標をネットワークへ入力することで，予測結果が学習したシーンに対する依存を防ぐことができるため，複数の異なるシーンの経路を予測することができる．評価実験において，予測対象のクラスと周囲の環境情報を導入することによる精度の変化を検証する．また，属性毎の予測精度の違いや入力するシーンラベルを変更した比較実験を行う．

本章の構成は以下の通りである．まず，3.1 節では提案するネットワークの概要を述べる．3.2 節では公開されているデータセットを用いて従来手法と提案手法の比較結果を述べる．最後に 3.3 節で本章をまとめる．

3.1　属性と環境情報を導入したネットワーク

本節では，予測対象のクラスおよび，対象周囲の環境情報を導入した経路予測手法について述べる．提案手法のネットワーク構成を図 3.1 に示す．まず，歩行者や自動車といった予測対象の情報から，その属性を表現するベクトルを生成する．次に，対象の移動情報を抽出する．本研究では，前の時刻の位置から現在までの移動量，すなわち相対座標をネットワークへの入力として用いる．そして，シーン中の予測対象を中心とする静的なシーンラベルを CNN に入力し，予測対象周囲の環境に関する特徴ベクトルを抽出する．予測対象の属性情報，移動情報および，環境に関する特徴ベクトルを連結し，LSTM へ入力する．LSTM の出力として，次の時刻の移動情報を出力する．LSTM は内部状態を記憶するメモリセルの働きにより，過去の情報を保持することができる．そのため，LSTM は上記の情報を逐次的に入力することで，予測対象の将来の位置を予測することが可能となる．提案手法では，対象の属性情報，過去の移動情報および，環境情報を抽出するための表現方法が重要となる．以下に，各入力情報の表現方法および，ネットワークへの入力について説明する．

3.1.1　問題設定

本研究では，シーン中のクラスが異なる複数の対象を予測することを目的とする．ネットワークには予測対象 $i = \{1, 2, \ldots, N\}$ の観測時刻 $t = \{1, 2, \ldots, T_{obs}\}$ の移動経路を入力し，予測時刻 $t = \{T_{obs}+1, T_{obs}+2, \ldots, T_{pred}\}$ における予測経路を出力する．

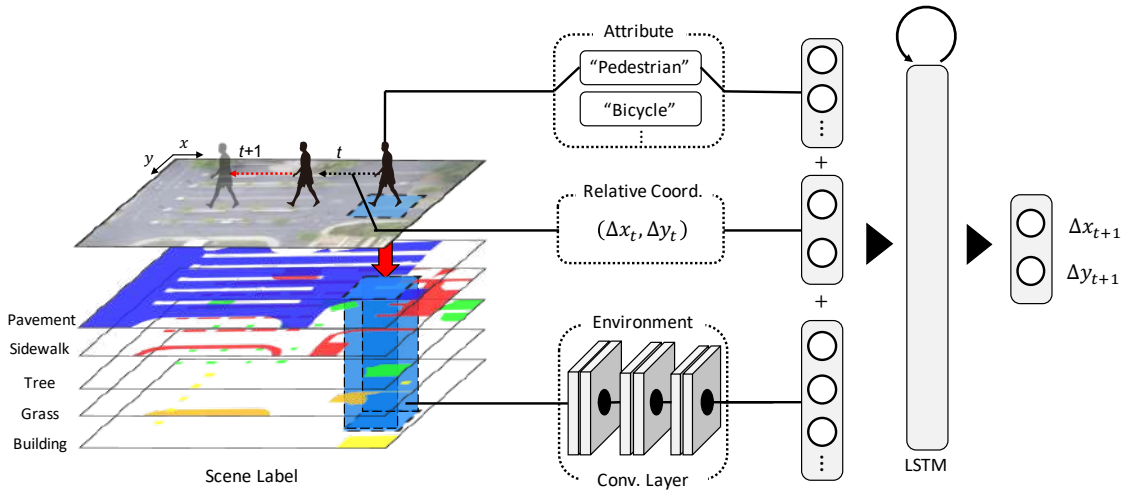


図 3.1: 提案手法のネットワーク構造．提案手法は予測対象の属性と相対座標及び，対象周囲の環境をネットワークへの入力として用いる．one-hot vector で埋め込まれた属性と畳み込み層を介してセマンティックなシーンラベルから抽出した特徴ベクトル及び，相対座標を LSTM へ入れ次時刻の相対座標を出力する．

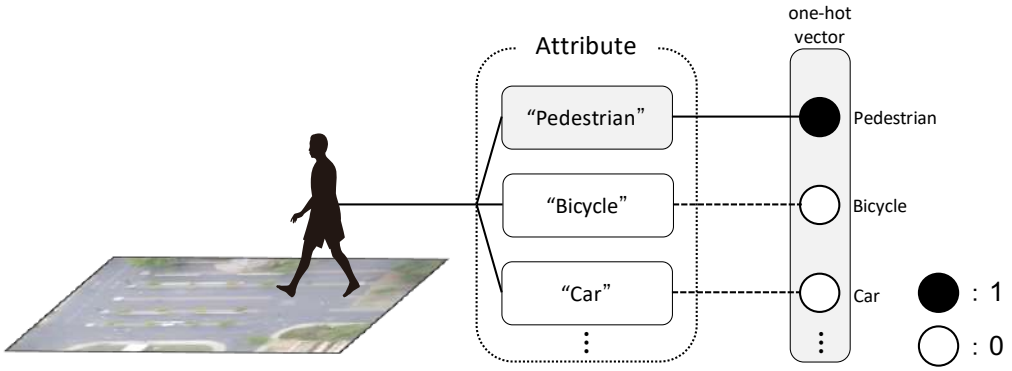


図 3.2: 予測対象の属性表現．ここでは，予測対象が歩行者の例を示す．対象の属性情報から one-hot vector を取得する．

3.1.2 属性情報

クラスが異なる複数の対象の経路を予測するために，入力に対象のクラスに関する追加情報を導入する．対象のクラスをネットワークへ入力するために，本研究では図 3.2 のように対象の属性情報を one-hot vector で表現する．具体的には，与えられた対象の属性情報をベクトル $\mathbf{e}$ に埋め込まれる．具体的な処理は，属性数を n と表した時，第 n 成分のベクトルの要素 e_n を 1，それ以外を 0 とすることで予測対象の属性を表現する．例えば，図 3.2 のように属性情報が pedestrian の場合，第 1 成分のベクトルの要素 e_1 に 1，それ以外の要素を 0 とする．この処理により，対象の属性毎に LSTM を構築する必要がなく，コンパクトな表現にできる．また，このベクトルを入力すると速度と方向に関して一意の経路を予測することが可能になる．さらに，この one-hot vector と静的環境を表す特徴ベクトルと組み合わせることにより，対象が移動する傾向がある領域も考慮できる．そのため，対象のクラスに応じた経路を予測することが期待できる．

3.1.3 移動情報

複数のシーンの予測を行う場合，予測シーン毎に建物や木などの障害物，歩道や車道などの移動領域が異なる．そのため，移動情報に過去から現在までの位置情報，すなわち相対座標を使用する．相対座標は式 (3.1) で定まる．

$$\begin{aligned} \delta x_t &= x_t - x_{t-1} \\ \delta y_t &= y_t - y_{t-1} \end{aligned} \tag{3.1}$$

ここで，t は現時刻，$t-1$ は前時刻を示す．式 (3.1) で求めた相対座標を LSTM に入力することで，次の時刻の相対座標を取得することができる．相対座標を用いることで，対象の現在地を常に基点

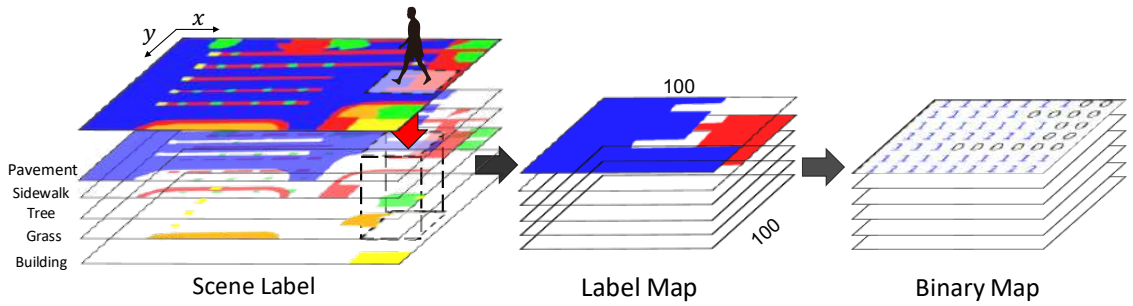


図 3.3: 予測対象周囲の環境情報の表現．シーンラベルから予測対象を中心とした領域を切り出しラベルマップを抽出する．ラベルマップはバイナリマップに変換した後，畳み込み層を介して特徴マップを抽出する．

とすることができるため，建物などの障害物や歩道などの移動領域の位置情報に依存することなく，複数のシーンでの経路予測を行うことが可能となる．

3.1.4 環境情報

環境情報は予測精度を向上させるためにも不可欠な要素である．従って，歩道や建物などのシーンに付与されたセマンティックなシーンラベルを用いて環境情報に関する特徴マップの抽出を行う．環境情報の導入手順を図 3.3 に示す．はじめに，シーンラベルから予測対象を中心とする 100×100[pixel] の領域ラベルを抽出する．次に，抽出したシーンラベル毎に分割したラベルマップを作成する．そして，ラベルマップを 0 または 1 で表現されたバイナリマップに変換する．作成したバイナリマップを CNN へ入力することで，環境に関する特徴ベクトル V_t を抽出する．環境情報に関する特徴は式 (3.2) で定まる．

$$V_t = CNN(I_t; W_{cnn}) \tag{3.2}$$

ここで，I_t はクロップしたバイナリマップ，W_{cnn} は重みパラメータを示す．求めた環境に関する特徴ベクトルをネットワークへ入力することで，障害物の有無や対象の属性に応じた領域を考慮した経路予測が可能になる．

3.1.5 ネットワークへの入力方法

上記の属性情報，移動情報および，環境情報を LSTM へ入力することで，次の時刻における対象の位置を予測する．具体的には，予測対象の各情報を観測データとして用いることで予測を行う．観測データは予測対象が実際に移動した真値を用いる．LSTM には観測データを予測開始直前のフレーム

表 3.1: ネットワーク構成の詳細．Convolution layer は環境に関する入力を受け取る．そして，Convolution layer から抽出された特徴ベクトルと属性に関するベクトルおよび，経路情報を LSTM へ入力する．

layer	kernel size	output size	remarks
input (attribute)		6	
input (coordinate)		2	
input (environment)		(100, 100, 7)	
conv1	(5, 5)	(48, 48, 16)	ReLU stride=2
norm1		(48, 48, 16)	batch norm.
pool1	(2, 2)	(24, 24, 16)	max pool.
conv2	(5, 5)	(20, 20, 32)	ReLU stride=1
norm2		(20, 20, 32)	batch norm.
pool2	(2, 2)	(10, 10, 32)	max pool.
conv3	(5, 5)	(6, 6, 32)	ReLU stride=1
pool3	(2, 2)	(3, 3, 32)	max pool.
concat		296	
LSTM		128	
output		2	

まで逐次的に入力する．予測時には LSTM の出力である予測値を次の時刻の入力として逐次的に入力する．その処理を予測終了時刻まで行い，予測の実現をする．式 (3.3) に予測経路の算出式を示す．

$$\begin{aligned} h_t &= LSTM((\delta x_t, \delta y_t), \mathbf{e}, V_t, \mathbf{h}_{t-1}; W_{LSTM}) \\ [\delta x_{t+1}, \delta y_{t+1}] &= \phi(h_t; W_c) \end{aligned} \tag{3.3}$$

ここで，$\mathbf{h}$ は LSTM の出力ベクトル，$\phi(\cdot)$ は単一の全結合層を示す．表 3.1 にネットワーク構成を示す．表 3.1 より提案手法のネットワーク構成は 3 層の CNN と 1 つの LSTM および，単一の全結合層から成るシンプルな構造である．convolution layer で環境に関する入力を受け取る．そして，convolution layer から抽出された特徴ベクトルと属性に関するベクトルおよび，経路情報を LSTM へ入力し，次時刻の予測経路を出力する．

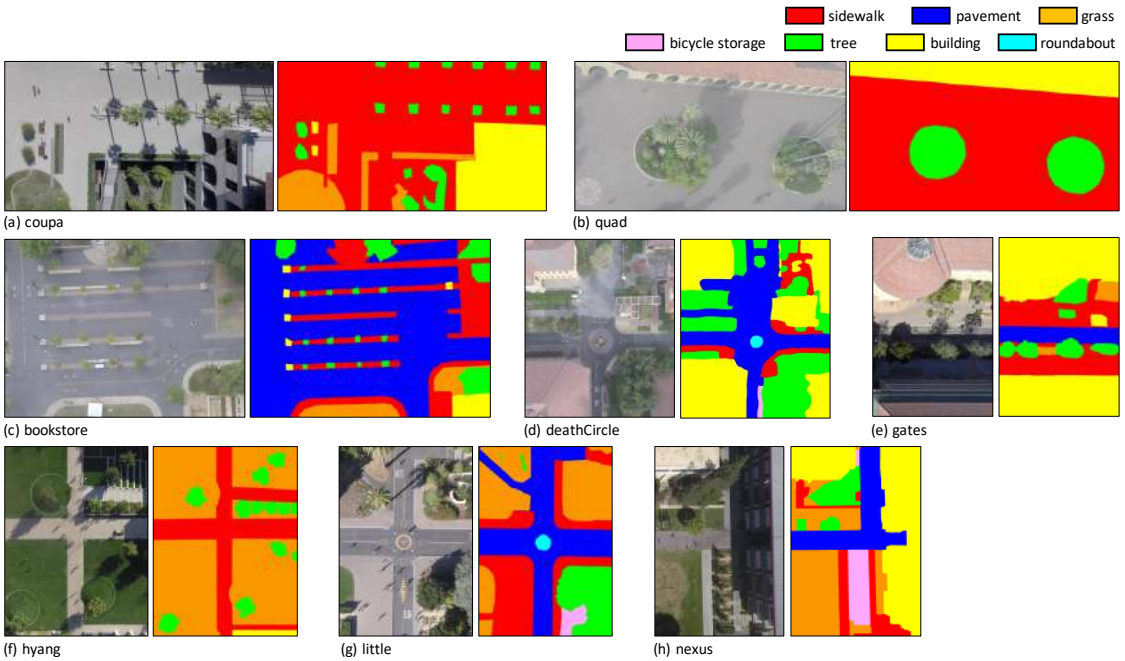


図 3.4: SDD のシーン例．各シーンの左図が実シーン画像，右図がアノテーションされたシーンラベル例を示す．

3.2 評価実験

提案手法の有効性を評価実験により検証する．具体的には，属性および，環境情報の有無による予測精度の変化について検証する．

3.2.1 データセット

データセットには，Stanford Drone Dataset (SDD) [20] を用いる．SDD は bookstore, coupa などの 8 つの異なる予測シーンから構成される．各シーンには異なる日時で撮影された動画が複数含まれており，合計で 60 本の異なる動画から構成される．図 3.4 に予測シーンの例を示す．

提案手法では，環境情報を表現するために，シーンラベルを入力として用いる．しかしながら，SDD にはシーンラベルの情報は含まれていないため，全てのシーンに対してシーンラベルを付与した．付与したシーンラベルの種類は sidewalk，pavement，grass，bicycle storage，tree，building，roundabout の合計 7 種類である．図 3.4 に付与したシーンラベル例を示す．各図の左はオリジナルのシーン例，右図はオリジナルのシーン例をアノテーションしたシーンラベル例を示す．シーンラベルはシーンの視覚的な外観だけでなく，歩行者の移動経路などに基づいて，右上の注釈に従ったクラスをシーン毎に慎重にアノテーションを行った．

SDD にはいくつかの不正確な経路が含まれている．図 3.5 に不正確な経路例を示す．緑線はアノ

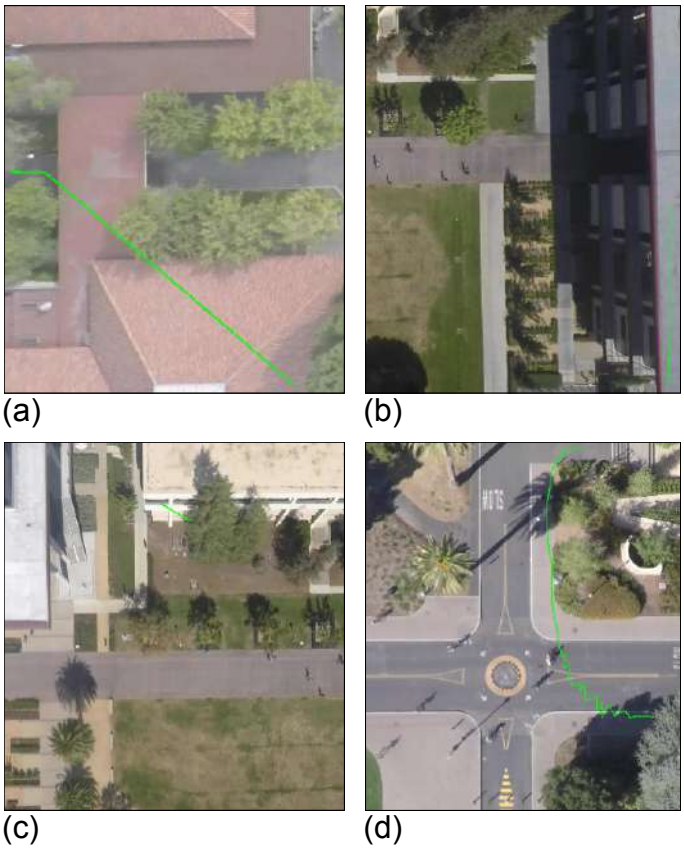


図 3.5: SDD の不正確な経路サンプル例．各シーン緑線がアノテーションされている．

テーションされた経路を示す．これらの例では，建物上または建物を横切る経路 (図 3.5(a, b, c)) や不正確にアノテーションされた経路 (図 3.5(d)) を示す．このような経路を学習および評価に使用すると，予測精度が大きく低下し，公正な比較が困難になる．そのため，本研究では正確で正しい経路のみを選択して使用する．

SDD は bicycle，pedestrian，cart，car，bus，skateboarder の異なる 6 クラスの移動対象と座標データから構成される．予測対象の数は学習用に 5,365，評価用に 1,082 を使用する．使用するデータの内訳を表 3.2 に示す．SDD はフレームレート 30fps で撮影されており，本実験では 20 フレーム毎の座標データを用いて実験を行う．すなわち，予測の 1 ステップは約 0.66 [s] に対応する．本実験では，このうちの 5 ステップ，すなわち約 3.3 秒間を観測として用いる．また，8 ステップ，すなわち約 5.3 秒間を予測として用いる．

3.2.2 評価指標と比較手法

定量的評価のために，本実験では 2 つの評価指標を用いる．1 つ目は予測の最終フレームにおける真値と予測値のユークリッド距離である Final Displacement Error (FDE)，2 つ目は各予測フレームに

表 3.2: 学習と評価データの内訳.

		train	test
Number of scenes		52	8
attribute	bicycle	2,369	545
	pedestrian	2,696	500
	cart	71	15
	car	75	5
	bus	17	2
	skateboarder	137	15

表 3.3: 各予測手法の定量的評価結果．単位は [pixel] である．属性と環境情報をネットワークへ導入することで予測精度が向上している．また，属性と環境情報の両方をネットワークへ導入することで，どちらの評価指標も提案手法の性能が最良である．

Metric	FDE	ADE
KF	174.42	116.02
S-LSTM	206.22	125.41
trajectory	196.13	86.42
trajectory + attribute	173.04	76.32
trajectory + environment	172.12	76.32
trajectory + attr. + env.	**109.44**	**53.20**

おける真値と予測値のユークリッド距離の平均である Average Displacement Error (ADE) を用いる．ADE および，FDE は式 (2.20)，式 (2.21) で表される．

また，比較手法として，状態空間モデルを用いて内部の状態を効率的に推定するカルマンフィルター (KF) [39] 及び，深層学習モデルの代表的な手法である Social LSTM (S-LSTM) [1] をベースラインとして用いる．

3.2.3 実験条件

学習条件として，最適化手法に RMSprop [104] を用いる．RMSprop の初期学習率を 0.01，α=0.99，ε=10e-8 として学習する．また，全ての予測モデルはバッチサイズを 10 に設定し，100 エポックで学習する．学習時には各時刻，すなわち観測の開始時刻から予測最終時刻までを通して過去の対象の

経路として真値をネットワークに入力する．予測時には，観測最終時刻で得られた最初の予測時刻を逐次ネットワークへ入力し予測値を得る．損失関数を真値と予測値との平均二乗誤差とする．フレームワークは Chainer，GPU に Nvidia Titan Xp を利用し End-to-end で学習および，評価する．

3.2.4 従来手法との評価結果

本節では，予測対象の属性および，環境情報を導入した場合の予測精度の変化について確認する．予測精度の比較結果を表 3.3 に示す．表 3.3 より，属性と環境情報のどちらか一方をネットワークへ入れることで予測精度が向上している．また，属性と環境情報の両方をネットワークへ導入した提案手法が最も予測精度が向上していることが確認できる．図 3.6 に予測結果例を示す．KF の予測経路では，過去の経路が観測とみなされるため，予測結果は障害物領域の有無に限らず線形の予測となる．LSTM ベースの予測手法では，pedestrian の予測経路はどの手法も真値と似た経路が得られた (図 3.6(d)，(e)，(f))．しかしながら，経路のみを利用した予測結果では，KF の予測結果より予測経路が正確に獲得できていないことが確認できる (図 3.6(a)，(c)，(g))．また，表 3.3 より S-LSTM は KF より予測精度が低下している．結果を再現するために，パラメータを慎重に選択したが，妥当な結果を得ることができなかったため，図 3.6 には S-LSTM の予測結果を記載していない．S-LSTM の結果について，同じ報告が [2] で報告されている．

表 3.3 に示すように，LSTM に他の補助情報を導入することで，予測精度の向上が確認できる．特に，静的な環境情報を導入することで，障害物を避けた経路を正確に予測することができる (図 3.6(h))．属性と環境情報のどちらかを追加した手法では，定量的評価の観点からの改善は KF と比較しても小さいが，提案手法である trajectory+attr.+env. は，他の手法と比較し精度向上が確認できる．また，提案手法では，図 3.6(a)，(b)，(c) 及び (g) で真値に類似した経路を予測していることが確認できる．図 3.6(d)，(e) 及び (f) は pedestrian の経路を示している．これらで得られた結果は，全ての経路予測手法が bicycle よりも歩行間隔が狭く pedestrian の経路を容易に予測できるため，真値に似た経路を追跡できることを示している．図 3.6(g) は，予測対象が車道に沿って進行する car の経路を示している．KF，trajectory，trajectory+attribute，trajectory+environment がネットワークの入力として使用される場合，予測結果は直進している．一方で，環境と属性を同時に導入することで，真値に似た経路を予測することができる．ただし，図 3.6(h) と (i) に示すように，環境情報を導入すると，予測結果は真値とは異なる予測経路を獲得する場合もある．

以上の結果から，提案手法は比較した経路予測手法の中で最も精度が高いことが確認できる．KF は線形に辿る経路の予測の精度は良いが，障害物回避の場合のような非線形経路を予測することは困難である．経路を正確に予測するには，属性と環境情報を対象の経路に導入する必要がある．

3.2.5 異なる属性毎の評価結果

表 3.4 に，属性と環境情報を導入した場合の各対象に関する予測誤差を示す．表 3.4 は属性と環境情報を両方考慮した結果を示す．表 3.4 より，bicycle，car および，skateboarder は他の属性より速く

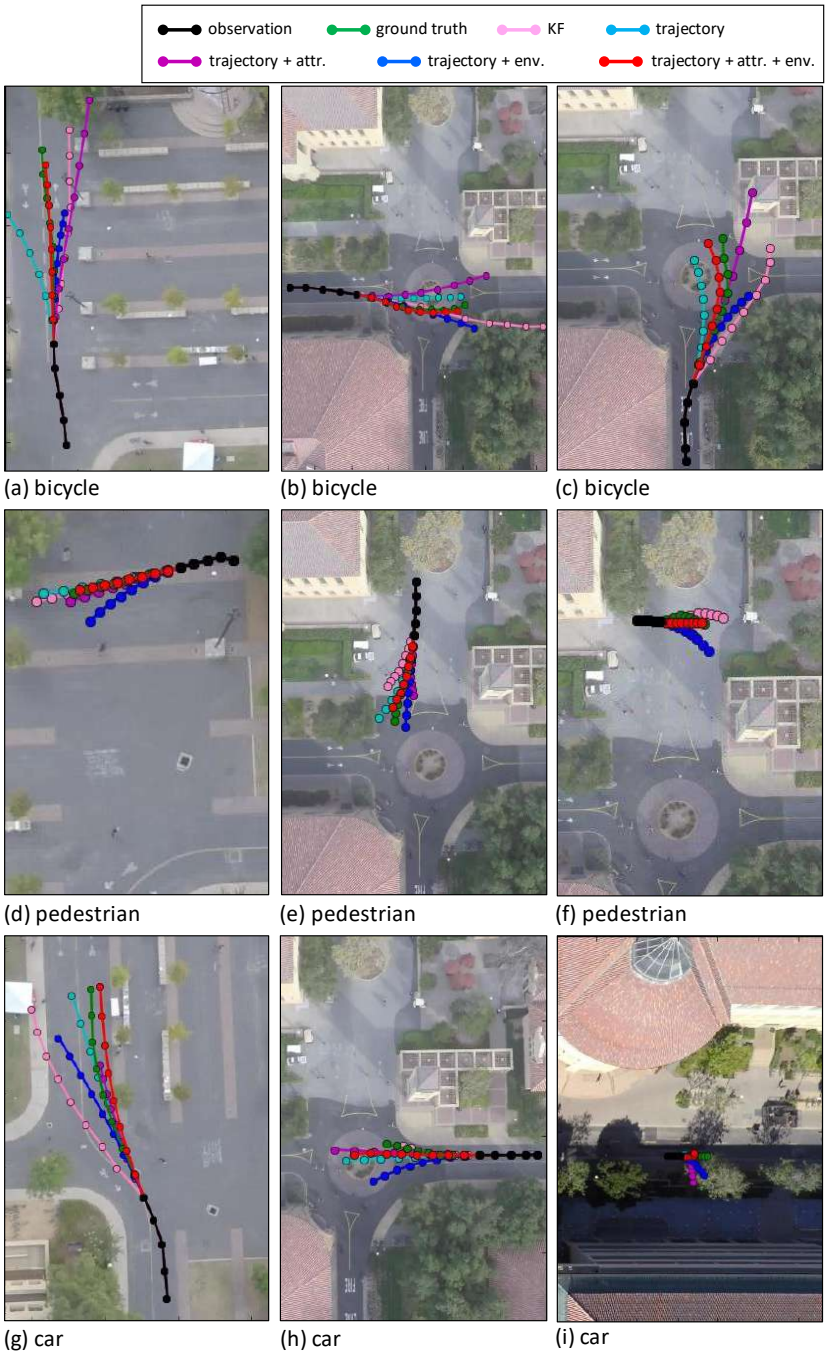


図 3.6: 各予測手法の予測結果例．各行のサブグラフは属性毎の予測結果例で，上から順に bicycle, pedestrian，car を示す．

移動するため，予測誤差は大きい．また，表 3.2 に示すように，cart，car，bus，skateboarder のサンプルは他のサンプルよりも少ないため，予測誤差が大きくなる傾向が見られる．

表 3.4: 属性毎の定量的評価結果．単位は [pixel] である．表は属性と環境情報を両方考慮した結果を示す．car や bicycle のような動きが速い対象の予測誤差が大きい．また対象のデータが少ない場合も予測誤差が大きい．

input	FDE	ADE
bicycle	113.82	51.25
pedestrian	43.22	23.14
cart	85.52	53.68
car	129.53	58.68
bus	151.74	76.34
skateboarder	132.79	61.67

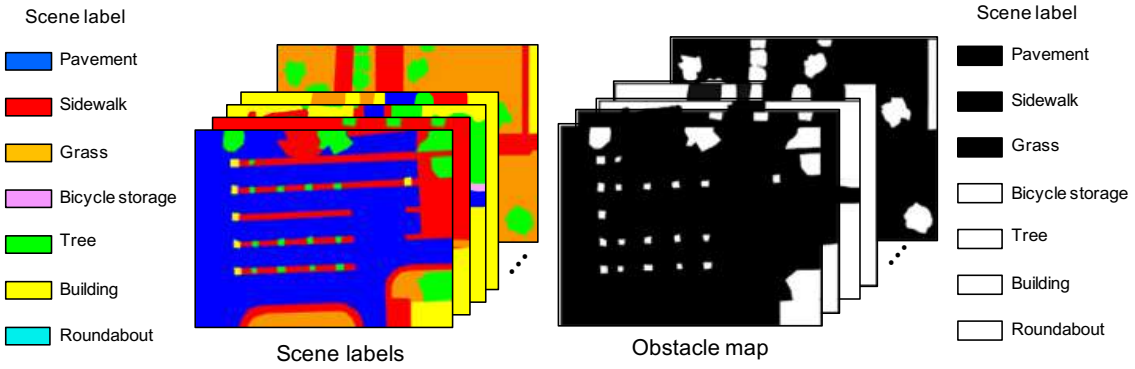


図 3.7: 障害物マップの例．シーンラベルの障害物領域を白，移動可能領域を黒とした障害物マップを作成する．異なる環境情報をネットワークの入力として用いることで，将来の経路を予測するのに適した環境情報を分析する．

3.2.6　入力が異なるシーンラベルを用いた検証実験

環境情報としてアノテーションされたシーンラベルを使用して実験を行った．セマンティックなラベルの有効性を評価するために，アノテーションされたシーンラベルを障害物エリアとして設定した bicycle storage，tree，building および，roundabout と他の領域に分割して作成した障害物マップを使用する実験を行う．図 3.7 に障害物マップの例を示す．障害物マップは 1 に設定された障害物領域と 0 に設定された移動可能領域のバイナリマップとして表現される．異なる環境情報をネットワークへの入力として用いることで，将来の経路を予測するのに適した環境情報を分析する．

入力が異なるシーンラベルを用いた定量的評価結果を表 3.5 に示し，予測結果例を図 3.8 に示す．表 3.5 は属性と環境情報を両方考慮した結果を示す．表 3.5 より，シーンラベルが障害物マップよりも優れていることを示す．図 3.8(a) の car の予測結果では，障害物マップを使用すると，sidewalk に向かって移動する経路を予測していることを示す．また，図 3.8(b) の bicycle の予測結果では，grass

表 3.5: 障害物マップとシーンラベルの定量的評価結果．単位は [pixel] である．

input	FDE	ADE
obstacle map	130.12	59.42
scene label	**109.44**	**53.20**

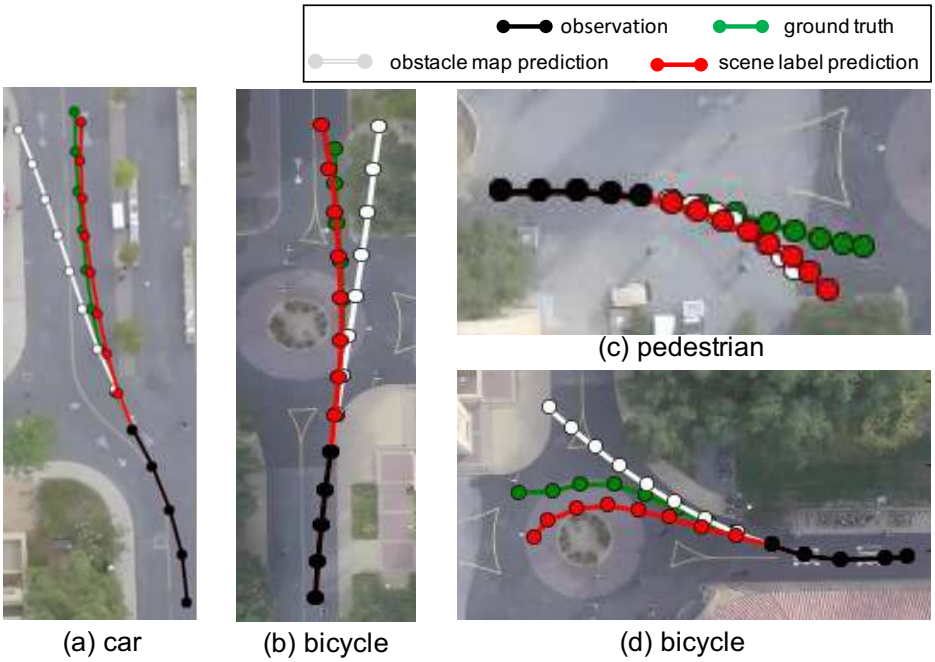


図 3.8: 異なる環境情報を導入した場合の予測結果例 .

に向かった経路を予測していることが確認できる．これらのような予測結果となったのは，障害物マップが障害物領域のみを区別するため，対象によっては sidewalk や grass など通常移動することがない領域を移動可能領域として予測してしまう．その結果，正確な経路の予測が困難になったと考えられる．

3.2.7 Failure cases

図 3.9 に，適切な予測が行えなかった例を示す．図 3.9(a) より，観測データが移動間隔の狭い移動経路から急激に速度が変化する場合では，急速な動きに対応することができず真値と異なる動きを予測している．また，図 3.9(b) では観測データが直進しているため，予測結果も直進した動きを予測している．このことから，分岐した経路の予測は困難であると考えられる．予測対象が cart の場合を図 3.9(c) と (d) に示す．図 3.9(d) では，真値は左折しているが，予測結果は建物方面に直進していることが確認できる．これは，cart のデータサンプルが少なく，学習が不充分だったため建物に衝

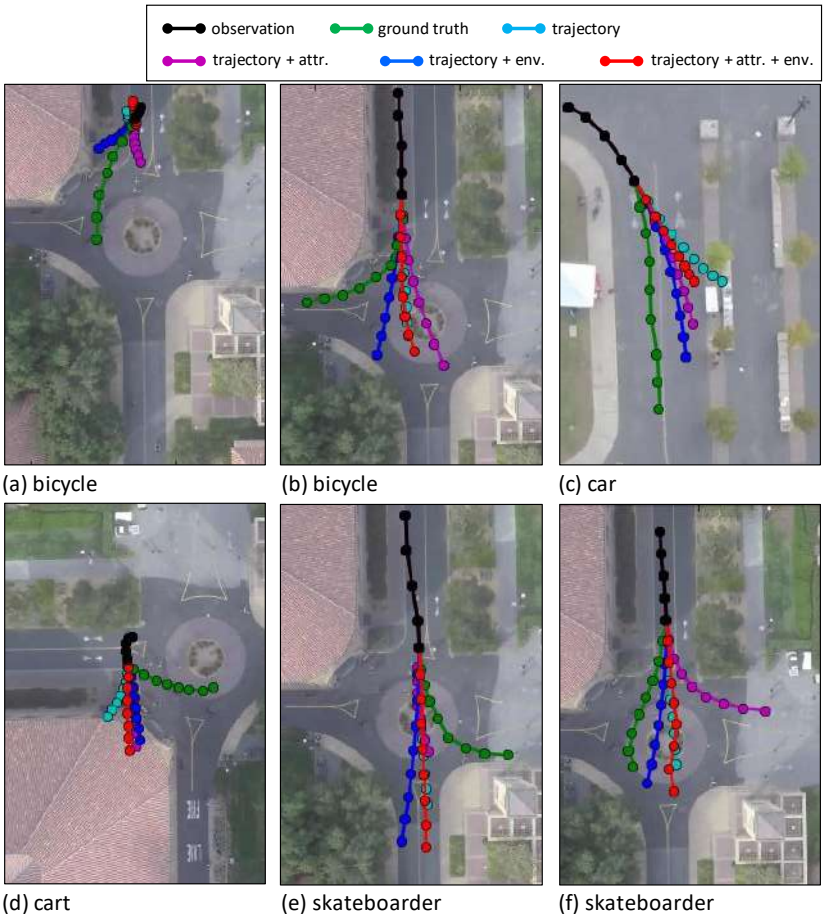


図 3.9: 誤った予測結果例．(a) の例は，対象の動きが急速に変化する場合の予測経路を示す．(b)，(c) の例は，将来の経路に複数の候補があり，実際の経路と異なる経路を予測している．(d)，(e) 及び (f) は表 3.2 より，対象数が少ないデータは障害物に衝突した経路を予測した．

突する動きを予測したと考えられる．予測対象が skateboarder の場合を図 3.9(e) と (f) に示す．どちらの予測結果も障害物上を移動していることが確認できる．これは，cart の場合と同様でデータのサンプル数が少なかったため，学習が不充分で障害物に衝突する動きとなったと考えられる．

以上より，移動対象の突発的な動きや分岐した経路の予測には対応できず，真値とは異なる予測結果となる問題が確認された．突発的な動きについて，本研究の手法が予測対象毎に特徴的な動きを考慮した経路予測を実現できるが，予測対象同士の衝突に関するインタラクション情報は考慮していないため，突発的な行動の予測に対応できないためだと考えられる．そのため，予測対象同士の衝突を防ぐインタラクション情報をネットワークへ導入することで，これに対処する．また，データ数が少ない属性の場合における経路予測は学習が不充分であったため，予測するのは困難となる問題が確認された．このようなデータに対応するためには，幾何変換などでデータを増やす Data Augmentation を活用することで，上記の問題を解決できると考えられる．

3.3 まとめ

本章では，予測対象の属性，及び周囲の環境情報を導入した経路予測手法を提案した．提案手法では，予測対象の属性を one-hot vector で表現し，周囲の環境情報のシーンラベルを CNN へ入力し，各情報に関する特徴ベクトルを抽出した．これらの情報を LSTM へ逐次的に入力することで，予測対象のクラスに応じた経路予測を実現した．SDD を用いた評価実験により，位置情報のみを入力する場合と比較して属性及び，環境情報を導入した場合における予測精度が高い結果となった．

これらの結果より，移動対象の属性と環境情報を導入した経路予測の有効性が確認できた．一方で，突発的な行動やデータサンプルが少ない属性では，十分な学習を行うことができず，適切な経路予測が困難であった．今後の課題として，予測対象同士の衝突を回避するインタラクション情報，すなわち動的な環境を考慮することが挙げられる．

第4章

混雑シーンにおける群衆密度予測

本章では，群衆の密度が未来でどのように変化するかを視覚的に予測する群衆密度予測を提案する．市街地やショッピングモールといった混雑シーンにおいて人間の経路を予測することは，自律ロボット [105] [106] のナビゲーションやドローンといった様々な実世界のアプリケーションに有益となる．経路予測を社会実装する場合，予測対象の正確な検出と追跡で捉えた経路情報と ID 情報が入力と出力の両方で必要となる．しかし，図 4.1(c) のような混雑シーンでは，オクルージョン等により一人一人を正確に検出・追跡が困難で，既存の経路予測手法をそのまま適用できない．

本章の目的は正確な歩行者データを用いた予測ではなく，不正確な歩行者データでも将来の予測を実現することである．そこで，シーンの各場所が将来どれだけ混雑しているかのマップ，すなわち群集密度マップを直接予測する手法を提案する．図 4.1(b) のように，群集密度推定 [107] [108] [109] により，各人を検出するよりも群集密度を推定する方が容易でコストが低い．そこで，観測から現在までのフレームから抽出した群集密度マップから，将来のフレームに対するマップを予測するモデルを学習させる．これにより，予測モデルは正確な歩行者の検出と追跡を必要とせず，群衆密度のダイナミクスを直接捉えることで将来の群衆密度を予測できる．提案手法の技術的困難は，群衆密度マップを効率的に予測するためにどのようにモデル化するかということである．特に，広域で撮影される入力映像では独立して移動する複数の集団が含まれていることが多い．さらに，観測される集団の数やシーン全体の混雑度はシーンの種類，例えば賑やかなショッピングモールと静かな街角によっては群衆が広範囲に変化する可能性がある．このため，群集密度マップの時空間ダイナミクスは多様かつ複雑で予測が困難となる．この困難に対処するためにパッチベースの密度予測ネットワーク (patch-based density forecasting networks : PDFN) を提案する．PDFN は，CNN の受容野の範囲で空間的あるいは時空的に重なり合ったパッチに基づき，シーン全体の多様で複雑な群集密度ダイナミクスをモデル化する．

提案手法の有効性を示すために，群衆密度予測に利用可能な様々なモデル [110] [111] を FDST [112] や UCY [18] で用いて評価実験を行った．実験結果より，混雑したシーンにおいて既存の経路予測手法 [1] [10] より PDFN が将来の群衆密度を正確に予測できることを示す．

本章の構成は以下のとおりである．まず，4.1 節では群衆密度予測の問題設定と提案手法について述べる．4.2 節では提案手法の有効性を確認するための評価実験について述べる．最後に 4.3 節で本章をまとめる．

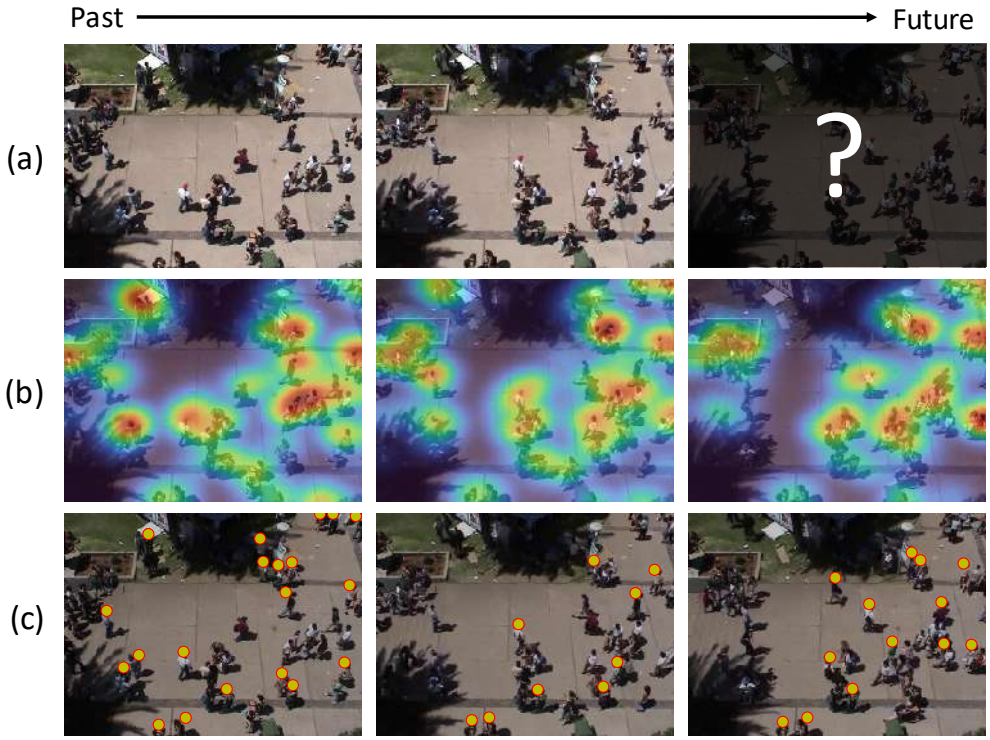


図 4.1: 群衆密度予測の概略図．(a) それぞれの群衆が行動するシーンで，(c) 黄色の円で示される各人の将来の位置を検出，追跡及び予測するのではなく，(b) 各場所がどれだけ混雑するかのマップ，すなわち群衆密度マップで将来のフレームで群衆がどう動くかを予測する．

4.1　群衆密度予測

本節では，まず群集密度予測問題を定式化する．そして，図 4.2 に示すように，提案するパッチベースのアプローチ PDFN-S と PDFN-ST の 2 つを含む幾つかの予測モデルを説明する．

4.1.1　問題設定

俯瞰視点カメラで撮影されたシーンを想定すると，様々な数の人のグループが形成されており，それぞれ独立に行動している．本章は最初の数フレームを入力とし，その後のフレームで群衆密度がどのように変化するかを予測する．提案手法では各フレームの各場所がどれだけ混雑しているかを群集密度マップで得るために，既存の群集密度推定 [113] [110] を利用して前処理を行う．画像サイズ (W, H) の t 番目のフレームから抽出した群集密度マップを $c_t \in [0,1]^{W \times H}$ とする．フレームの長さ T_{in} と T_{out} の群集密度マップの入力と出力をそれぞれ $C_{in} = [c_{t-T_{in}+1}, \ldots, c_t]$ と $C_{out} = [c_{t+1}, \ldots, c_{t+T_{out}}]$ で表す．提案手法の課題は，C_{in} から C_{out} へのマッピングを学習することである．ビデオベースの群集密度推定 [112] [114] とは異なり，C_{out} に対応するビデオフレームは評価時に入力として利用できない．さらに提案手法の問題設定は，予測対象として群集密度の推定結果を利用するため，人の位置が正確にアノテーションされたデータを想定していない．これは混雑したシーンにおいて，一人一人の位置を手動でアノテーションすることはコストがかかるためである．しかし，公平な比較実験を行うために，公開されているデータセットを用いて提案手法の予測結果が実際の人々の位置とどれだけ一致するかを評価する．

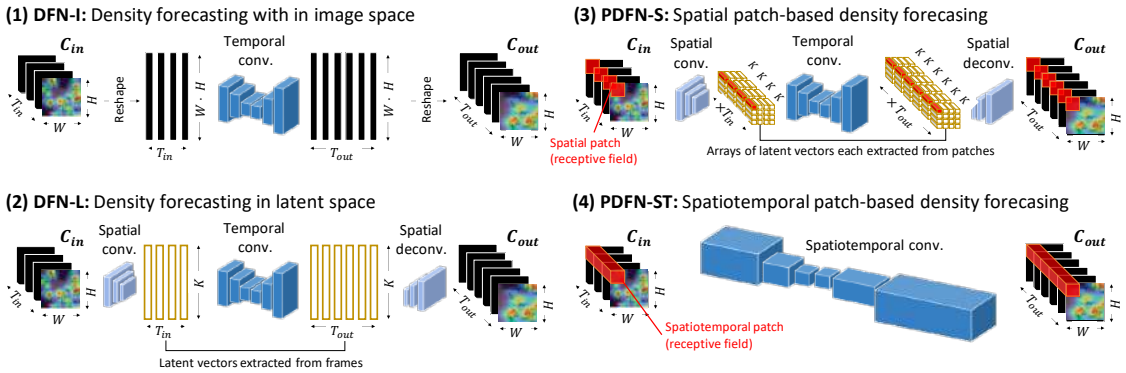


図 4.2: 提案手法の概略図．観測の群衆密度マップ C_{in} を入力とし，将来の群衆密度マップ C_{out} を予測する．PDFN-S と PDFN-ST は，空間的または時空的なパッチ毎に独立して予測するパッチベースの予測モデルを提案する．パッチは赤で強調され，CNN の受容野の範囲で予測される．

4.1.2 従来のネットワークモデルでの群衆密度予測

群衆密度を直接予測する従来手法がない．そのため，群衆密度マップを直接予測する方法と潜在空間上における予測を従来手法とし，本節ではそれらについて説明する．

■ Density Forecasting in Image Space

まず，観測の群衆密度マップ C_{in} から将来の群衆密度マップ C_{out} を直接予測する方法について説明する．この方法を画像空間で学習した密度予測ネットワーク (density forecasting network trained in the image space : DFN-I) とする．図 4.2(a) のように，DFN-I はまず C_{in} の各入力マップを $W \times H$ 次元のベクトルに変換する．次に，長さ T_{in} の入力ベクトルを temporal convolution に与え，長さ T_{out} の将来の群集密度マップを得る．これは $W \times H$ 次元のベクトルとなる．出力を元の画像空間として変換することで，将来の群衆密度マップ C_{out} が得られる．DFN-I は単純であるが，画素間の局所的な相関を無視して多様な群集密度マップを単一のベクトルで記述するため，効率的でない．

■ Density Forecasting in Latent Space

群集密度マップを直接入力とする代わりに，マップをコンパクトな潜在ベクトルに符号化する密度予測ネットワーク (density forecasting in latent space : DFN-L) を提案する．潜在空間における予測は，モデルベースの強化学習 [115] や模倣学習 [116] で広く用いられる．図 4.2(b) のように，DFN-L は temporal convolution と auto-encoder 型の convolution/deconvolution で構成される．エンコーダは spatial convolution で，群衆密度マップの空間的なパターンをモデル化できる．具体的には，まず入力マップをエンコーダに通し，K 次元の潜在ベクトルを得る．次に，潜在ベクトルを temporal convolution に与え，T_{out} 後のフレームの潜在ベクトルを出力する．最後に，出力された潜在ベクトルをデコーダを通じ，未来の群集密度マップを予測する．

4.1.3 パッチベースの群衆密度予測

本節では，提案するパッチベースの群衆密度予測手法について説明する．4 章の初めに説明したように，群衆の密度のダイナミクスが多様で複雑であっても，局所的な領域のダイナミクスは単純である．そのため，パッチベースによる群衆密度予測を提案する．具体的には，図 4.2(c) のような空間パッチに基づく密度予測ネットワーク (spatial patch-based density forecasting : PDFN-S) を提案する．PDFN-S には，群衆密度マップの空間的なパターンをモデル化する auto-encoder 型のネットワーク及び，密度の時間的ダイナミクスをモデル化する temporal convolution で構成される．マップを 1 つの特徴量として埋め込み予測する DFN-L とは異なり，PDFN-S はパッチレベルで群衆の密度を予測する．

■ 空間的パターンをモデル化する auto-encoder

まず，PDFN-S は入力された群集密度マップを空間的に重複するパッチから抽出された K 次元の潜在ベクトルを変換する auto-encoder 型の convolution/deconvolution を学習する．図 4.2(c) のように，エンコーダでは群衆密度マップ C_{in} を空間的に重なり合う複数のパッチに分解し，コンパクトな特徴表現を convolutional network で学習する．デコーダでは，temporal convolution で予測した特徴ベクトルから，deconvolutional networks で特徴マップをアップサンプリングし，将来の群衆密度マップ C_{out} を生成するように学習する．空間的に重なり合う複数のパッチを K 次元の潜在空間に auto-encoder に埋め込むことで，シーンの小さい領域で観測される群衆密度マップの時空間パターンをシンプルに学習できる．

■ 時間的ダイナミクスをモデル化する temporal convolution

auto-encoder を学習した後に，DFN-L で行われたように temporal convolution で潜在空間での予測を行うが，各潜在ベクトルについて独立に行う．これはパッチ単位の予測に相当し，各パッチはシーン全体よりも単純な群集密度パターンの予測を意味する．具体的には，事前学習した auto-encoder$\mathcal{E}$ のエンコーダの特徴マップからデコーダの特徴マップを予測することを考える．エンコーダで埋め込まれる特徴マップを Z_{in}，デコーダで予測する特徴マップを Z_{out} とすると，以下式で表される．

$$Z_{in} = [\mathcal{E}(c_{t-T_{in}+1}), \ldots, \mathcal{E}(c_t)] \in \mathbb{R}^{W' \times H' \times K \times T_{in}} \tag{4.1}$$

$$Z_{out} = [\mathcal{E}(c_{t+1}), \ldots, \mathcal{E}(c_{t+T_{out}})] \in \mathbb{R}^{W' \times H' \times K \times T_{out}} \tag{4.2}$$

観測の特徴マップ Z_{in} から予測の特徴マップ Z_{out} は temporal convolution でモデル化する．temporal convolution で観測の特徴マップ Z_{in} の T_{in} 方向をダウンサンプリングしながら畳み込み，Z_{out} の T_{out} 分をアップサンプリングしながら畳み込む．すなわち，temporal convolution で予測する特徴マップは $Z'_{out} = \mathcal{M}(Z_{in}) \in \mathbb{R}^{W' \times H' \times K \times T_{out}}$ となる．これらの temporal convolution は (W', H') の各位置に対して独立で行うため，群衆密度のダイナミクスを小さな空間でモデルを学習できる．予測結果の特徴マップ Z'_{out} は潜在ベクトルの列であり，これをデコーダに通すことで，将来の群集密度マップとして予測する．

4.1.4 時空間パッチベースの群衆密度予測

PDFN-S は，各パッチの空間情報と時間のダイナミクスをそれぞれのネットワークで学習する．本節では，PDFN-S の拡張として時空間パッチベースの群衆密度予測手法について説明する．具体的には，時空間パッチとみなす 3 次元の受容野における局所的な群集密度のダイナミクスを学習するために，3D convolution と deconvolution で構成される，時空間パッチベースに基づく密度予測ネットワーク (spatio-temporal patch-based density forecasting : PDFN-ST) を提案する．PDFN-ST の具体的構造は図 4.2(d) に示す．空間的な特徴を捉える auto-encoder 型の convolution/deconvolution と群衆密度

のダイナミクスを捉える temporal convolution を別々に学習する PDFN-S とは異なり，PDFN-ST は単一のネットワークで群集密度マップの時空間的パターンを学習する．

4.1.5 ネットワーク構成

上記で説明した DFN-I，DFN-L，PDFN-S 及び，PDFN-ST は以下のように実装した．

- **Temporal Convolutional Network**．図 4.2 に示すように，PDFN-ST を除くすべてのモデルは予測を行うために temporal convolution を用いている．これは，3 つの convolution layer と 3 つの deconvolution layer から構成され，それぞれが活性化関数 ReLU [117] を持つ．チャンネル数，カーネルサイズ，ストライドはそれぞれ，$(64, 4, 2), (128, 4, 2), (256, 2, 1), (128, 3, 1), (64, 4, 2), (K, 4, 2)$ とする．K は DFN-L と PDFN-S では潜在ベクトルの次元，DFN-I では $W \times H$ であり，長さ T_{in} の入力系列を変換して長さ T_{out} の出力系列にするように設定した．

- **Auto-encoder**．DFN-L と PDFN-S は，群衆密度マップを K 次元の潜在ベクトルに変換するためにエンコーダとデコーダ構造を持つ．PDFN-S では，4 つの convolution layer と 3 つの deconvolution layer から構成される．convolution layer のチャンネル数，カーネルサイズ，ストライドをそれぞれ，$(32, 4, 2), (64, 4, 2), (64, 4, 2), (K, 1, 1)$，deconvolution layer は $(64, 4, 2), (64, 4, 2), (1, 4, 2)$ で設定する．また，各層で活性化関数 ReLU を持つ．パッチサイズ，つまり受容野の大きさはピクセルで 22×22 に設定した．DFN-L は，単一の潜在ベクトルを出力するために，最後の convolution layer の前後にパラメータ $(64, 4, 2)$ を持つ 2 つの convolution layer と deconvolution layer を持つ．層数と潜在ベクトルの次元数は，検証データセットで性能を最大化するために設定した．convolution layer と deconvolution layer の数は $(4, 3)$ と $(3, 2)$ から選択し，潜在ベクトル K は $K \in \{8, 16, 32, 64, 128\}$ から選択した．

- **Spatiotemporal Convolution Network**．PDFN-ST は 3 つの 3D convolution layer と 3 つの deconvolution layer から構成され，これらは全て活性化関数 ReLU を用いている．チャンネル数，カーネルサイズ，空間的及び時間的なストライドは，入力の長さ T_{in} から T_{out} を適切に出力するために，convolution layer では $(32, 4, 2, 2)$, $(64, 4, 2, 2)$, $(256, 4, 2, 2)$，deconvolution layer では $(64, 4, 2, 1)$, $(32, 4, 2, 2)$, $(1, 4, 2, 2)$ に設定した．そして，時空間パッチサイズは画素数で 22×22，フレーム数 22 に設定した．auto-encoder と同様に，検証用データセットの性能に基づきこのネットワーク構造とした．

- **Training**．すべてのモデルは最適化手法 Adam [118] で学習を行う．バッチサイズ，反復回数，学習率は，FDST では $(16, 1k, 0.001)$，UCY データセットでは $(16, 100k, 0.005)$ とした．損失関数は，DFN-L と PDFN-S の temporal convolution では平均二乗誤差，DFN-I の temporal convolution，PDFN-ST の spatial-temporal convolution 及び，DFN-L と PDFN-S の auto-encoder ではバイナリクロスエントロピーで学習を行う．さらに，混雑した領域と混雑していない領域の両方を学習させるために，c_t の代わりに $\sqrt{c_t}$ を入力とした．

4.2 評価実験

本節では，公開されているデータセットを用いて，提案手法と従来の予測手法を比較する．

4.2.1 データセット

本実験では，俯瞰視点で撮影された屋内外の様々なシーンから成る以下のデータセットを使用する．

- **FDST** [112] は crowd counting タスクのためのデータセットである．FDST には合計 100 つの様々な場所で撮影された 15 シーンの動画像が含まれている．各動画像は 30fps の 150 フレームで構成され，各フレームについて歩行者の位置がアノテーションされている．FDST は 15 シーンの内 10 シーンが 5 つの動画像，他の 5 シーンが 10 つの動画像となっているが，元論文では 13 シーンであると報告されている．

- **UCY (Crowds-by-Example Dataset)** [18] は経路予測のための一般的なデータセットである．UCY には，ZARA01(9,031 frames)，ZARA02(10,519 frames)，UCY(5,405 frames; all recorded at 25 fps) の斜めの俯瞰視点で撮影された群衆の異なるシーンが 3 つある．FDST データセットとは異なり， このデータセットでは歩行者の位置情報は 10 フレーム毎にアノテーションされている．

4.2.2 データの前処理

- **群集密度マップの生成**．群集密度予測では，動画像から群集密度マップを求めることが重要になる．本実験では，2 つの異なる入力方法を採用し比較した．具体的には，入力動画像から直接ピクセル毎の混雑の度合いを推定するために GCC データセットで事前学習した C3 フレームワーク [113] による群集密度推定 [110] 及び，MS COCO データセット [119] で事前学習した物体検出器 (ChainerCV [120] による feature pyramid network [111]) に基づく人検出である．これらを用いることで，4.1 節で述べたような実用的な場面で必要となる動画像中の歩行者一人一人に対するアノテーションを行わずに，群集密度予測モデルの学習を行うことができる．人検出を用いた場合，検出された各 bounding box の上端中心位置を入力画像空間にマッピングし，群集密度推定で得られたものと互換性のある群集密度マップを形成する．群衆密度推定と人検出は 640×480 のサイズの動画像で行ったが，予測モデルの学習時間を短縮するために群衆密度マップを 80×80 にリサイズする．

- **真値の群衆密度マップの作成**．群衆密度予測モデルの学習に使用する群衆密度マップに加え，モデルの性能を評価するために各データセットの真値から真値の群衆密度マップを作成する．上記の人検出結果の処理方法と同様に，全人物の位置を入力画像空間にマッピングし，モデルの出力とマッチするようにサイズの変更を行う．しかし，FDST と ZARA02 では人物の頭部，

ZARA01 と UCY では足元がアノテーションされているためデータの不一致が発生する．そこで，人検出の結果から人物の垂直位置と身長を対応付けるサポートベクター回帰で，ZARA01 と UCY の足元にアノテーションされているデータを頭部になるように修正した．

- **データセットの分割**．一般的な経路予測ベンチマークに従い，データセットから T_{in} と T_{out} を設定する．具体的には，FDST は 6fps，UCY は 5fps の連続した 20 フレームから成るサンプルを時間方向に 1 時刻毎にスライドする．そのため，最初の 8 フレームを観測時刻，その後 12 フレームを予測時刻とした．すなわち，$T_{in} = 8, T_{out} = 12$ となる．UCY では，3 つのシーンに対して leave-one-out を行い，ETH データセット [19] から HOTEL シーン (25fps で撮影された 19,350 フレーム) を学習データとして leave-one-out の学習として追加することで，学習した群集密度ダイナミクスの多様性を増加させるようにした．ETH データセットの ETH シーンも経路予測手法のベンチマークとして利用されるが，予備実験の結果，真値のアノテーションが不完全であり，群衆密度予測タスクへの適用が困難であることが判明したため，ETH シーンは利用しない．FDST では，100 つの動画像から学習を 60，評価を 40 に分割し，それぞれ 10 と 5 の異なるシーンが含まれるように設定した．FDST は，異なる群衆の行動から成る 60 の学習用動画像と 40 の評価用動画像を同じ 15 のシーンで作成している．そこで，学習したモデルが未知シーンで正確に予測できるかを調査する目的で，同じ数の学習用動画像と評価用動画像でシーンの重複がないように分割した．

4.2.3 評価方法

提案手法の有効性を評価するために，予測された群衆密度マップと真値の群衆密度マップを比較する．これを行うために，事前定義したカーネルサイズ σ のガウシアンフィルタでこれらのマップに対して平滑化を行う．このカーネルサイズは，物体検出タスクの IoU スコアに与えられる閾値と同様に，予測された群衆密度マップと真値の群衆密度マップとの差をどの程度厳密に測定するかを制御するために使用した [119]．σ を小さく設定すると，予測結果がより厳密に真値の群衆密度マップと一致することが期待される．これに対して，大きな σ を設定することで，予測モデルが近似的な予測を行うことが期待される．実験を通して $\sigma = 3$ と設定した．

予測された群衆密度マップと真値の群衆密度マップそれぞれについて，予測最終時刻 T_{out} と予測開始時刻から最終時刻 $t+T_{out}$ までを平均した recall(再現性能) と precision(精度性能) で評価する．この 2 つの評価の差は，時刻の経過とともに予測が困難になることを示す．すなわち，予測最終時刻の方が予測が困難となる．c_τ と g_τ を τ 番目のフレームにおける $W \times H$ サイズの予測の群衆密度マップと真値の群衆密度マップとする．真値マップ g_τ が予測マップ c_τ によってどの程度正確に予測されたかという再現性能を評価するために，Kullback-Leibler (KL) divergence を計算した．KL divergence は $D_{KL}(g_\tau||c_\tau) = \frac{1}{W \cdot H} \sum_{i,j} \bar{g}_\tau(i,j) \log \left(\frac{\bar{g}_\tau(i,j)}{\bar{c}_\tau(i,j)} \right)$ のように計算される．ここで $\bar{c}_\tau = c_\tau / \sum_{i,j} c_\tau(i,j)$ と $\bar{g}_\tau = g_\tau / \sum_{i,j} g_\tau(i,j)$ は，確率分布になるように正規化した予測マップと真値マップ，i, j はマップの各位置に付与されたインデックスを表す．また、Inverse KL divergence : $D_{RKL}(g_\tau||c_\tau) = D_{KL}(c_\tau||g_\tau)$

を用いて精度性能，すなわち予測マップ c_τ がどれだけ正確に真値マップ g_τ を予測しているかを評価する．最後に，$D_{JS}(g_\tau||c_\tau) = \frac{1}{2}(D_{KL}(g_\tau||\frac{g_\tau+c_\tau}{2}) + D_{KL}(c_\tau||\frac{g_\tau+c_\tau}{2}))$ として定義される Jensen-Shannon (JS) divergence による性能評価も行う．これは，精度性能と再現性能のバランスを直感的に示す指標である．これらの divergence は非負であるため，スコアが低いほど性能が良いことを意味する．

4.2.4 比較手法

群衆密度予測タスクに関する先行研究がないため，ベースラインとして以下に示すいくつかの経路予測手法を拡張した．まず，4.2.2 節で得られた人物検出結果を，[121] [122] などの多くの実用的な予測タスクで用いられている sort tracking [123] でフレーム毎に追跡し，フレーム間を線形補間して学習と評価の経路データを作成した．予測した経路をマップ化し，ガウシアンフィルタによって平滑化を行う．これにより，4.2.2 節で真値マップに対して行ったように，未来の群集密度マップを形成する．

- **ConstVel** [40]．RNN を経路予測に応用した研究が行われているにもかかわらず，速度に基づいて経路予測を行う単純なアプローチが強力であることが判明している．[40] に従い，観測時刻最後の 2 フレームにおける速度情報から将来の経路を線形に外挿した．

- **Social LSTM (S-LSTM)** [1]．深層学習を用いた経路予測で最も使用されるベースラインである．S-LSTM は，各個人の経路を予測するために LSTM で学習し，LSTM に周囲の歩行者の hidden state が追加される．S-LSTM の学習には，元論文で提案されているハイパーパラメータの設定で実装した．

- **Trajectron** [10]．Trajectron は，歩行者グループを閾値に基づいて動的なグラフ構造でモデル化した経路予測手法である．S-LSTM とは異なり，Trajectron は複数の未来の予測経路をサンプリングできる．この問題設定においてより良い性能を発揮するために，ハイパーパラメータの選択を元論文から若干変更した．具体的には，640×480 サイズの動画像で動作するように，歩行者間の閾値距離を変更した．評価時には，各フレームの各歩行者の予測経路を 100 個サンプリングし，それらの経路にガウシアンフィルタを適用して予測の群集密度マップを生成した．

4.2.5 実験結果

■ ベースラインとの比較

表 4.1 と表 4.2 に，FDST と UCY データセットに対する定量的な評価結果を示す．提案手法のパッチベースモデルである PDFN-S と PDFN-ST は，全てのデータセットにおいてベースラインを上回る性能を示す．また，人物の検出は必ずしも安定していないにも関わらず，検出ベースの入力を用いた PDFN-S と PDFN-ST は ZARA02 を除く全てのシーンでベースラインの性能を上回る．従って，歩

行者の経路ではなく，群衆密度マップを直接予測することの重要性が示される．さらに，群衆密度推定による入力を用いることで，予測性能はさらに向上している．DFN-I と DFN-L は，歩行者の集団が多い FDST データセットにおいて性能が低下している．これらの結果より，パッチベースの予測手法を用いることが有効であることを示す．特に UCY では，PDFN-ST が PDFN-S よりも性能が向上していることから，群集密度マップの空間的パターンと時間的ダイナミクスを単一のネットワークで学習することの有効性が確認できる．

■ 予測結果

図 4.3 に群衆密度推定を入力とした場合の Trajectron と PDFN-ST の予測結果例を示す．図 4.3(a)(b) に示すように，Trajectron は数人の将来の位置を正確に予測できるが，図 4.3(c)(d) では正確な検出や追跡ができず正確な予測ができていない．一方で，提案手法の PDFN-ST は個人 (図 4.3(c))，小さな集団 (図 4.3(b)(d)) 及び，群集 (図 4.3(a)) のダイナミクスを予測できている．

■ ガウシアンカーネルの影響

群衆密度予測タスクでは，入出力の群集密度マップにガウシアンフィルタを適用した．そのカーネルサイズ σ で予測結果をどれだけ厳密に評価されるかについて Ablation study を行った．表 4.3

表 4.1: FDST の定量的評価結果．左が予測時刻の平均，右が最終時刻における divergence のスコアを示す．PDFN-S と PDFN-ST がパッチベースの提案手法を示す．

	D_{KL}	D_{RKL}	D_{JS}
ConstVel [40]	1.83 / 2.30	0.92 / 1.47	0.13 / 0.18
S-LSTM [1]	2.23 / 2.80	1.89 / 2.50	0.20 / 0.24
Trajectron [10]	1.60 / 1.90	0.94 / 1.59	0.12 / 0.16
Inputs given by pedestrian detection			
DFN-I	1.02 / 1.01	2.40 / 2.66	0.22 / 0.23
DFN-L	1.86 / 1.88	5.35 / 5.65	0.33 / 0.33
PDFN-S	0.45 / 1.19	0.68 / 1.41	0.07 / 0.16
PDFN-ST	0.18 / 0.43	**0.38 / 0.87**	0.05 / 0.11
Inputs given by crowd density estimation			
DFN-I	1.01 / 0.88	4.58 / 4.95	0.25 / 0.23
DFN-L	1.63 / 1.68	6.67 / 7.01	0.34 / 0.34
PDFN-S	**0.15 / 0.36**	0.44 / 1.06	**0.04 / 0.10**
PDFN-ST	0.16 / 0.37	0.45 / 1.08	**0.04 / 0.10**

表 4.2: UCY の定量的評価結果.

ZARA01	D_{KL}	D_{RKL}	D_{JS}
ConstVel [40]	7.60 / 8.52	3.26 / 4.52	0.33 / 0.40
S-LSTM [1]	10.9 / 12.3	9.07 / 10.5	0.50 / 0.56
Trajectron [10]	7.19 / 7.67	4.29 / 5.36	0.34 / 0.39
Inputs given by person detection			
DFN-I	3.73 / 3.87	7.95 / 8.77	0.45 / 0.48
DFN-L	4.82 / 5.55	8.24 / 9.31	0.48 / 0.52
PDFN-S	2.75 / 3.69	5.32 / 6.07	0.32 / 0.38
PDFN-ST	2.66 / 3.36	5.39 / 5.87	0.31 / 0.36
Inputs given by crowd density estimation			
DFN-I	1.80 / 2.10	5.11 / 6.56	0.33 / 0.38
DFN-L	0.93 / 1.65	2.73 / 4.06	0.20 / 0.31
PDFN-S	0.88 / 1.57	2.53 / 3.54	0.19 / 0.28
PDFN-ST	**0.76 / 1.11**	**2.29 / 3.01**	**0.17 / 0.22**
ZARA02	D_{KL}	D_{RKL}	D_{JS}
ConstVel [40]	7.76 / 8.59	3.52 / 4.41	0.34 / 0.40
S-LSTM [1]	9.58 / 10.8	7.48 / 8.44	0.48 / 0.52
Trajectron [10]	7.26 / 7.41	3.69 / 3.77	0.35 / 0.34
Inputs given by pedestrian detection			
DFN-I	3.56 / 3.24	7.92 / 8.00	0.45 / 0.44
DFN-L	4.04 / 3.59	7.53 / 7.36	0.45 / 0.44
PDFN-S	5.65 / 5.88	7.40 / 7.72	0.45 / 0.46
PDFN-ST	5.92 / 5.91	7.00 / 7.38	0.44 / 0.45
Inputs given by crowd density estimation			
DFN-I	1.97 / 2.36	5.43 / 6.78	0.34 / 0.40
DFN-L	0.99 / 1.90	2.58 / 3.70	0.20 / 0.30
PDFN-S	0.98 / 1.84	2.67 / 3.81	0.20 / 0.30
PDFN-ST	**0.85 / 1.47**	**2.35 / 3.26**	**0.18 / 0.26**
UCY	D_{KL}	D_{RKL}	D_{JS}
ConstVel [40]	8.92 / 8.99	3.86 / 4.13	0.39 / 0.41
S-LSTM [1]	8.96 / 9.70	6.21 / 6.99	0.47 / 0.52
Trajectron [10]	8.63 / 8.34	3.69 / 3.93	0.37 / 0.39
Inputs given by pedestrian detection			
DFN-I	8.62 / 9.02	6.01 / 5.78	0.52 / 0.54
DFN-L	9.07 / 8.82	4.90 / 5.13	0.48 / 0.48
PDFN-S	3.46 / 4.28	2.90 / 3.31	0.31 / 0.36
PDFN-ST	3.45 / 3.78	2.72 / 2.61	0.30 / 0.33
Inputs given by crowd density estimation			
DFN-I	1.84 / 1.80	4.21 / 4.47	0.32 / 0.33
DFN-L	1.04 / 1.35	2.49 / 2.98	**0.21 / 0.26**
PDFN-S	**1.04 / 1.32**	2.45 / 2.83	**0.21 / 0.25**
PDFN-ST	1.07 / 1.37	**2.28 / 2.72**	**0.21 / 0.26**

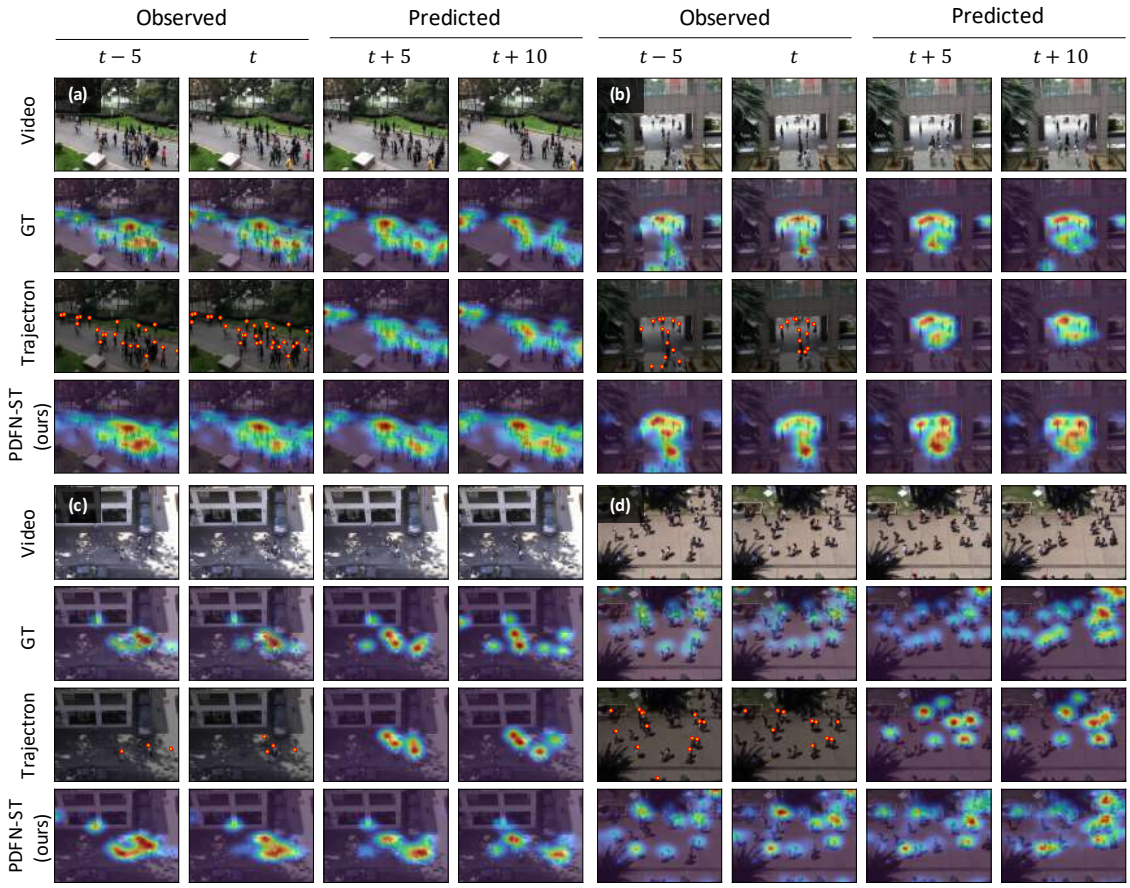


図 4.3: 予測結果例．入力，真値 (GT)，Trajectron と群集密度推定結果を入力した PDFN-ST による予測された群集密度マップを示す．ここでは，群衆密度マップの変化を分かりやすく可視化するため，$t-5, t, t+5, t+10$ フレーム目を選択して予測結果例を示す．Trajectron で検出・追跡された人物は黄色の丸で囲まれている．

は，UCY データセットにおいて σ の値を変えた場合に予測性能がどれだけ影響されるかを示している．σ が小さくなるにつれ評価基準が厳しくなり，全ての予測手法の性能が低下している．しかし，PDFN-ST は Trajectron の性能を上回っていることから，群衆密度マップを直接予測することが重要であることがわかる．

■ Failure cases

図 4.4 に群衆密度推定を入力した場合の PDFN-ST の誤った予測結果例を示す．図 4.4(a) のように，提案手法は後半のフレームで示すような歩行方向を急変させる群衆の予測は失敗している．また，図 4.4(b) のように，将来のフレームで新しい人がフレームイン/アウトするシーンを含む，パッチサイズを超える動きの予測は困難である．しかし，これらは既存の経路予測手法でもよく見られ

表 4.3: ガウシアンカーネルサイズ σ の効果．数値は UCY の平均/最終時刻の divergence のスコアを 3 つのシーンで平均したものを表す．

	σ	D_{KL}	D_{RKL}	D_{JS}
Trajectron [10]	1	13.5 / 13.4	9.67 / 11.1	0.54 / 0.56
	3	7.54 / 7.71	3.90 / 4.37	0.35 / 0.37
	6	3.69 / 3.83	1.96 / 2.19	0.23 / 0.24
PDFN-ST	1	2.19 / 2.68	9.31 / 10.7	0.43 / 0.47
	3	0.87 / 1.32	2.31 / 3.05	0.18 / 0.25
	6	0.95 / 1.30	1.26 / 1.59	0.16 / 0.20

る困難な例である．これに対処するためには，シーンの特徴 [8]，または目標地点の情報 [124] などの追加の入力情報が必要となる．

■ 処理速度

最後に、単一の GPU (Nvidia Tesla V100) 上で PDFN-ST がどの程度高速に予測を行うかを測定した．群集密度推定は，1 枚の画像から入力マップを作成するのに 10.4 [ms] 必要となる．そして，PDFN-ST は観測から現在までの 8 フレームを処理し，8.1 [ms] で未来の 12 フレームを予測する．従って，PDFN-ST をオンライン方式で実行すれば，18.5 [ms] 毎に新しい予測結果を得ることができる．すなわち，54fps の速度で実行可能である．なお，実用的なシステムではアプリケーションによって異なるハードウェアやタスクの性能が必要であり，これらはすべての予測手法の実行時間に影響を与えることが想定される．

4.3 まとめ

本章では，混雑シーンにおける監視カメラシステムのための新しい視覚的予測手法である群衆密度予測を提案した．提案手法はパッチベースでモデル化することで，様々な数の独立した群衆の多様で複雑なダイナミクスを効率的に捉えることができる．さらに，予測対象の正確な検出と追跡に依存する既存の経路予測手法とは異なり，提案手法は群集密度推定手法の結果をマップで入力することで，正確な予測を実現した．群集密度予測を監視カメラだけでなく，ウェアラブルカメラ [125, 29] やオートカメラ [105, 106] に拡張すると，ナビゲーションシステム，運転支援など他のアプリケーションにつながると予想される．このようなアプリケーションには，複数のカメラにまたがる群衆の再識別や，長期的な群衆ダイナミクスのモデリングと予測のための新しい技術も必要となると考えられる．また，衝突を回避するための歩行者間のインタラクションが必要になると考えられる．しかし，提案手法では群衆をマップとして表現しているため，インタラクションを導入できない．従って，群

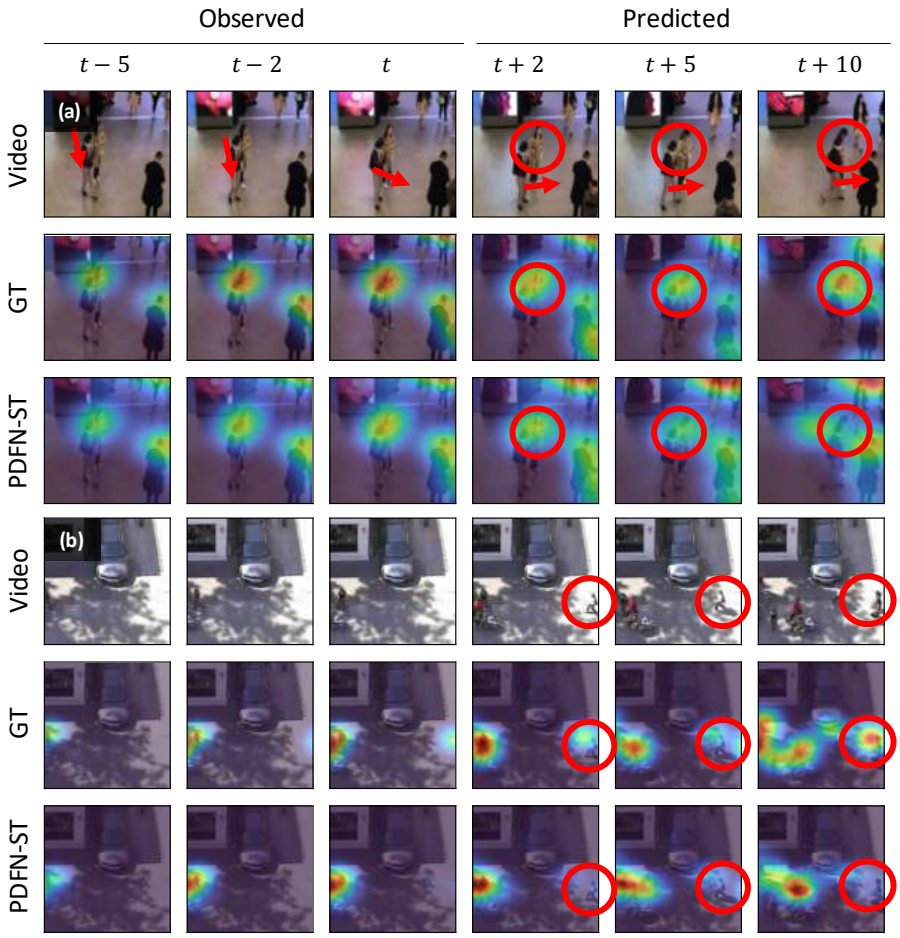


図 4.4: 誤った予測結果例．(a) は急に歩く方向が変わった場合のサンプル，(b) は新しい人がシーンに現れる場合のサンプルを示す．

衆といったグループのインタラクションをどのようにモデル化するかが重要になると考えられる．

第5章

インタラクションを考慮した経路予測の性能調査

本章では，2 章で説明した代表的な予測モデルについて，経路予測で主に使われるデータセットを用いて各モデルの精度及び予測結果の傾向について議論を行う．本章では，5.1 節で経路予測の評価の問題点を述べる．5.2 節で精度比較を行うモデルの詳細とハイパーパラメータの設定を述べる．5.3 節で学習及び評価で使用するデータセットや評価方法について述べる．5.4 節及び 5.5 節で各データセットによる各予測モデルの比較結果を述べる．5.6 節で各予測モデルの計算時間とパラメータの比較を行う．5.7 節でプーリングモデルとアテンションモデルの違いについて考察を述べる．最後に，5.8 節で本章をまとめる．

5.1 経路予測の評価の問題点

一般的に経路予測モデルの評価には，Displacement Error と Minimum Displacement Error が用いられている．しかしながら，これらの評価指標を用いてモデルを評価すると，元の論文値と異なる結果になることがある．[10] でも述べられているように，[3] と [1] で同じ著者が出した結果が異なるため，経路予測分野でどのモデルが最先端で矛盾のない公平な評価になっているかを判断することが困難である．また，[3][1]や [37][2]の公開コードの issue で指摘されているように，他手法と比較するための評価コードが異なっており，コードを揃えると結果が他手法の予測性能を下回ることが確認されている．そのため，本章では公平な比較及び，DL ベースの経路予測手法において，インタラクションのモデル化の効果の検証のために，いくつかの最先端の手法の公開コードを用いて，それぞれのモデルサイズやハイパーパラメータを論文と同じ条件にした元で学習と評価を行う．

5.2 精度比較を行うモデル

使用するモデルを表 5.1 に示す．深層学習 (DL) ベースの経路予測手法は，用いる映像視点や CNN や LSTM などのネットワークのベースになるアーキテクチャにより条件が異なるため，全ての予測手法を一様に評価することが困難である．本実験では，最も広く用いられている LSTM に基づく経路予測の中で代表的な手法を用いる．各モデルの詳細とハイパーパラメータの設定は以下である．

- **LSTM** は入力層-LSTM-出力層の 3 層のモデルで構成される．推論時は，過去最終時刻まで入出力を行い，予測開始時刻以降は出力層で得た経路情報を入力層に逐次入力する．LSTM に入れる前の埋め込み層の次元数を 64，LSTM の次元数を 64，入力層と出力層の次元数を 2 で設

表 5.1: 精度比較を行うモデル．

モデル名	インタラクション	環境	データセット
LSTM	-	-	ETH/UCY, SDD
RED [126]	-	-	ETH/UCY, SDD
Social LSTM[3]	Pooling	-	ETH/UCY, SDD
Social GAN[4]	Pooling	-	ETH/UCY, SDD
STGAT[5]	Attention	-	ETH/UCY, SDD
Trajectron[6]	Attention	-	ETH/UCY, SDD
Env LSTM	-	✓	SDD

[1] https://github.com/agrimgupta92/sgan/issues/8
[2] https://github.com/abduallahmohamed/Social-STGCNN/issues/47
[3] https://github.com/quancore/social-lstm
[4] https://github.com/agrimgupta92/sgan
[5] https://github.com/huang-xx/STGAT
[6] https://github.com/StanfordASL/Trajectron

定する．エポック数を 300，バッチサイズを 64，最適化手法を Adam [118]，学習率を 0.001 で設定する．損失関数は予測値と真値間の L2 loss を使用する．

- **RED** は，Recurrent Encoder-Decoder [126] で構成されている．エンコーダとデコーダはどちらも LSTM を使用する．各 LSTM に入れる前の埋め込み層の次元数を 64，各 LSTM の次元数を 64，入力層と出力層の次元数を 2 で設定する．エポック数を 300，バッチサイズを 64，最適化手法を Adam，学習率を 0.001 で設定する．損失関数は予測値と真値間の L2 loss を使用する．

- **Social LSTM** は，周囲の他対象との衝突を避ける S-Pooling を導入した予測モデルである．Social LSTM のベースモデルは LSTM であるため，推論時は過去最終時刻まで入出力を行い，予測開始時刻以降は出力層で得た経路情報を入力層に逐次入れる．LSTM に入れる前の埋め込み層の次元数を 64，LSTM の次元数を 128，プーリング処理内の全結合層の次元数を 128，入力層と出力層を 2 で設定する．ETH/UCY と SDD のどちらも，S-Pooling 内のパラメータの予測対象を中心とした周囲の範囲を 32，グリッドサイズを 8 で設定する．エポック数を 200，バッチサイズを 64，最適化手法を RMSprop [104]，学習率を 0.003 で設定する．損失関数は負の対数尤度 [127] を使用する．

- **Social GAN** は，GAN を用いて実際の経路と予測経路を敵対的に学習させる予測モデルである．また，S-Pooling を改良した Pooling Module により周囲の他対象との衝突回避を期待できる．さらに，正規分布を基にしたノイズベクトルを予測モデルに加えることで複数の経路を予測する．Social GAN のベースモデルは RED である．生成器と識別器の各 LSTM に入れる前の埋め込み層の次元数を 16，生成器の各 LSTM の次元数を 32，識別器の LSTM の次元数を 64，プーリング処理内の全結合層の次元数を $[48 \times 512 \times 32]$，入力層と出力層の次元数を 2 で設定する．ノイズベクトルの次元数を 8 で設定する．エポック数を 200，バッチサイズを 64，最適化手法を Adam，学習率を 0.001 で設定する．損失関数は予測値と真値との Adversarial loss 及び，k 個のサンプリングした予測値から真値に最も似た予測値のみを選択し，その予測値と真値間の L2 loss を用いる．ここで，k は 20 で設定する．

- **STGAT** は，Graph Attention Network で得た空間情報を LSTM で時間方向に伝播することで，時空間特徴をエンコードする予測モデルである．STGAT のベースは RED である．各 LSTM に入れる前の埋め込み層の次元数を 16，エンコーダの LSTM の次元数を 32，デコーダの LSTM の次元数を 64，アテンション処理の次元数を $[16 \times 32]$，入力層と出力層の次元数を 2 で設定する．ノイズベクトルの次元数を 8 で設定する．エポック数を 400，バッチサイズを 64，最適化手法を Adam，学習率を 0.001 で設定する．損失関数は k 個のサンプリングした予測値から真値に最も似た予測値のみを選択し，その予測値と真値間の L2 loss を用いる．ここで，k は 20 で設定する．

- **Trajectron** は，シーン内の複数対象を動的なグラフ構造で効率的にモデル化する．Trajectron のベースは RED である．各 LSTM に入れる前の埋め込み層の次元数を 8，エンコーダの LSTM の次元数を 32，デコーダの LSTM の次元数を 128，アテンション処理の次元数を 8，入力層と

出力層の次元数を 2 で設定する．ノイズベクトルの次元数を 10 で設定する．Gaussian Mixture Model のコンポーネント数を 16 で設定する．ETH/UCY では，予測対象を中心とした周囲 1.5 [m] 以内の他対象とのグラフを構築する．SDD では，1.5 [m] を 40 ピクセルで設定する．反復数を 20,000，バッチサイズを 256，最適化手法を Adam，学習率を 0.001 で設定する．損失関数は尤度最大化損失を用いる．

- **Env LSTM** は，予測対象周辺の環境情報を考慮することで，周辺の障害物との衝突回避した経路予測を行うモデルである．Env LSTM のベースは LSTM であるため，推論時は過去最終時刻まで入出力を行い，予測開始時刻以降は出力層で得た経路情報を入力層に逐次入力する．予測対象を中心とした周囲 100×100 [pixel] のセマンティックなラベル情報を 2 層の CNN へ入れる．1 層目の CNN のチャネルサイズを 32，カーネルサイズを 5，ゼロパディングを 0，ストライドを 2 で設定する．2 層目の CNN のチャネルサイズを 32，カーネルサイズを 5，ゼロパディングを 0，ストライドを 1 で設定する．各 CNN から出た特徴マップは batch normalization と活性化関数 ReLU を介した後に 2×2 の maxpool. を行う．最後に，1 次元の特徴ベクトルに変換する．LSTM には CNN から出た特徴ベクトルと座標値を連結して入力する．LSTM の次元数を 128 で設定し，出力層の次元数を 2 で設定する．エポック数を 300，バッチサイズを 64，最適化手法を Adam，学習率を 0.001 で設定する．損失関数は予測値と真値間の L2 loss を使用する．

5.3 学習及び評価の設定

データセットは経路予測で最も使用される ETH/UCY データセット及び，Stanford Drone Dataset (SDD) [20] を用いる．ETH/UCY は ETH, HOTEL などを含む 5 つのシーンがある．学習及び評価には，leave-one-out アプローチを用いる．ETH/UCY で観測された歩行者の経路の可視化を図 5.1 に示す．図 5.1 の青色は歩行者の移動経路を示す．図 5.1 のように，ETH 以外のシーンは左右に行動する予測対象の割合が多く，ETH は上下に行動する予測対象の割合が多い．そのため，ETH の x, y 座標を変換した C-ETH を評価に用いる．実験で使用した ETH/UCY の歩行者のデータ数及び，密度を表 5.2 に示す．

SDD では bookstore，coupa などを含む 8 つのシーンがある．学習や評価，経路情報などの設定は ETH/UCY と同条件で行う．また，SDD は歩行者以外にも自動車などの複数対象のデータがあるが，本実験では歩行者のみを対象とする．実験で使用した SDD の歩行者のデータ数及び，密度を表 5.3 に示す．

実験では，過去約 3.2 秒，未来約 4.8 秒間の経路情報を各モデルの入出力として用いる．評価指標は 2.4 節で説明した Displacement Error，Minimum Displacement Error 及び，各衝突率を用いる．Minimum Displacement Error はサンプリング数を 20 とする．動的物体との衝突率の閾値は ETH/UCY で 0.1 [m]，SDD で 10 [pixel] とする．静的物体との衝突率では，各データセットに付与したシーンラベルを用いて予測値が障害物領域の範囲内にいた場合，接触したとみなす．シーンラベルは sidewalk,

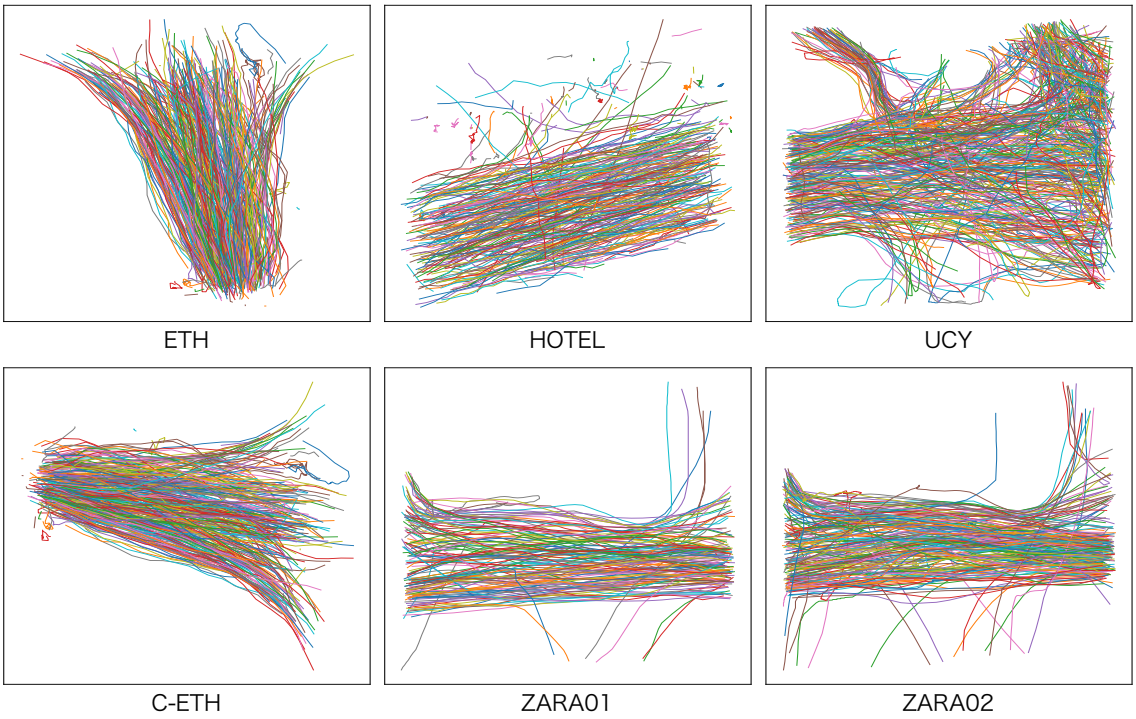


図 5.1: ETH/UCY で観測された歩行者の経路の可視化.

表 5.2: ETH/UCY のサンプル数及び，密度の内訳．密度は 1 [m] × 1 [m] の四角形の中点をある歩行者の x, y 座標とし，四角形内に他の歩行者が位置する場合の最大の密度を記載.

シーン名	歩行者数	密度 [人数/m^2]
ETH	358	8
HOTEL	389	5
UCY	434	8
ZARA01	148	4
ZARA02	204	5

pavement，grass，bicycle storage，tree，building，roundabout の 7 種類で，そのうち tree，building，roundabout の 3 種類を障害物領域とする．Displacement Error 及び各衝突率による評価は単一経路の予測手法と複数経路の予測手法で実験条件が異なるため，公平な比較ができない．そのため，複数経路を予測するモデル (Social GAN, STGAT, Trajectron) はノイズベクトルなどを含めない条件でも学習と評価を行う．これにより，単一経路の予測手法と同様の条件で各評価指標を公平に比較できる．

表 5.3: SDD のサンプル数及び，密度の内訳．密度は 50 [pixel] ×50 [pixel] の四角形の中点をある歩行者の x, y 座標とし，四角形内に他の歩行者が位置する場合の最大の密度を記載.

シーン名	歩行者数	密度 [人数/pixel^2]
bookstore	357	8
coupa	147	8
deathCircle	346	8
gates	379	8
hyang	819	8
little	93	8
nexus	716	8
quad	20	8

表 5.4: ETH/UCY における予測誤差．単位は [m].

	Method	Scene						
		C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	AVG
Single Model	LSTM	0.57 / **1.20**	0.71 / 1.36	**0.21** / **0.38**	0.63 / 1.37	0.59 / 1.26	0.42 / 0.84	0.52 / 1.08
	RED	**0.56** / 1.24	0.67 / 1.36	**0.21** / 0.40	0.61 / 1.34	0.52 / 1.19	0.42 / 0.89	**0.50** / **1.07**
	Social LSTM	0.93 / 1.62	1.19 / 2.27	0.49 / 0.97	0.98 / 1.94	1.02 / 1.82	0.71 / 1.38	0.89 / 1.67
	Social GAN	0.62 / 1.35	0.73 / 1.59	0.48 / 1.07	0.78 / 1.62	0.62 / 1.36	0.50 / 1.10	0.62 / 1.35
	STGAT	0.60 / 1.29	**0.61** / **1.24**	0.23 / 0.45	0.65 / 1.40	0.52 / 1.11	0.42 / 0.90	0.51 / **1.07**
	Trajectron	0.58 / 1.27	0.79 / 1.80	0.25 / 0.46	**0.58** / **1.28**	**0.40** / **0.89**	**0.38** / **0.76**	**0.50** / 1.08
20 Outputs	Social GAN	0.49 / 1.05	0.53 / 1.03	0.35 / 0.77	0.79 / 1.63	0.47 / 1.01	0.44 / 0.94	0.51 / 1.07
	STGAT	**0.44** / **0.77**	**0.48** / **0.94**	**0.16** / **0.29**	0.57 / 1.23	**0.35** / **0.74**	0.31 / 0.67	**0.39** / **0.77**
	Trajectron	0.52 / 1.14	0.68 / 1.34	0.19 / 0.40	**0.55** / 1.24	**0.35** / 0.82	**0.27** / **0.65**	0.43 / **0.77**

5.4 ETH/UCY における精度比較

ETH/UCY における予測誤差を表 5.4 に示す．表 5.4 の Single Model は Displacement Error，20 Outputs は Minimum Displacement Error の結果を示す．また，各シーンのスラッシュの左を ADE，右を FDE による評価結果を示す．表 5.4 より，ETH/UCY において Single Model では，各シーンを平均した ADE で RED と Trajectron，FDE で RED と STGAT が最も予測誤差を低減していることがわかる．HOTEL シーンは，歩行者の密度が低く線形的な動きになりやすいため，インタラクションを考慮しない LSTM や RED だけで十分な予測精度を得ることができる．一方で，UCY などの歩行者密度が高いとインタラクションを考慮した Trajectron が最も予測誤差を低減していることから，インタラクションを考慮した経路予測は歩行者密度が高いシーンで有効であると言える．また，Social

表 5.5: ETH/UCY における動的物体との衝突率．単位は [%].

	Method	Scene						
		C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	AVG
Single Model	LSTM	0.81	**0.81**	0.44	0.25	0.24	0.28	0.47
	RED	0.81	2.41	0.75	0.21	**0.15**	0.29	0.77
	Social LSTM	1.61	4.03	1.34	0.38	0.76	0.69	1.47
	Social GAN	0.81	**0.81**	**0.22**	0.38	0.34	0.44	0.50
	STGAT	**0.0**	**0.81**	0.44	**0.20**	0.19	0.26	**0.32**
	Trajectron	0.81	**0.81**	0.60	**0.20**	0.24	**0.20**	0.48
20 Outputs	Social GAN	1.61	**0.0**	0.60	0.23	0.17	0.26	0.48
	STGAT	**0.0**	0.81	**0.44**	0.25	0.27	**0.24**	**0.34**
	Trajectron	1.61	**0.0**	0.60	**0.18**	**0.15**	0.26	0.47

LSTM の予測誤差が大きいのは，[3][9] でも述べられているように [127] の負の対数尤度関数を最小化するように学習すると，サンプリングプロセスが微分不可能になるため誤差逆伝播が困難になり，より良いパラメータを学習できなかったためだと考えられる．20 Outputs では，各シーンを平均した ADE で STGAT，FDE で STGAT と Trajectron が最も予測誤差を低減していることがわかる．Social GAN の予測精度が低い要因は，Pooling Module が過去最終時刻におけるインタラクションのみを考慮しており，他対象が“近づく”あるいは“遠ざかる”などの時間的ダイナミクスを考慮できていないためと考えられる．一方で，STGAT や Trajectron は過去時刻全てのインタラクションを時間方向にも捉えるため，精度が向上したと考えられる．

次に，動的物体との衝突率を表 5.5，静的物体との衝突率を表 5.6 に示す．表 5.5 及び表 5.6 より，どちらの衝突率もほとんどのモデルが約 2%を下回る結果となっていることがわかる．動的物体との衝突率の結果より，STGAT が最も衝突率が低いことがわかる．表 5.4 では RED が最良ではあったが，表 5.5 では STGAT が最良なことから，STGAT は歩行者間の複雑なインタラクションを捉えるあまり予測誤差が大きくなったと考えられる．また静的物体との衝突率では，ほとんどのモデルが衝突しないことがわかる．これは ETH/UCY が複雑な地形をしておらず，障害物が歩行者の前方に存在する状況がないため，衝突率が全体的に減少したと考えられる．

最後に，予測結果例を図 5.2 に示す．図 5.2 の緑色の実線を過去経路，緑色の波線を真値，赤色の実線を予測経路として表す．複数経路を予測する手法は，表 5.4 の Single Model の予測結果であることに注意されたい．図 5.2(a) は，群衆のシーン例である．図 5.2(a) より，Social GAN や STGAT などインタラクションを考慮する予測手法は真値と似た経路を予測する．特に，中央の並列する歩行者は RED だと衝突の可能性がある経路を予測するが，Social GAN などはインタラクションが考慮されているため真値と似た経路を予測している．図 5.2(b) は，2 人の歩行者が並行して動くシーン例である．図 5.2(b) より，ほとんどの予測手法が真値と似た経路を辿っている．ETH Dataset には，このような線形に動く歩行者のデータが多く含まれるため表 5.4 の LSTM や RED の予測誤差が低下し

表 5.6: ETH/UCY における静的物体との衝突率．単位は%．

	Method	Scene						AVG
		C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	
Single Model	LSTM	1.02	1.97	3.71	3.95	2.88	0.57	2.35
	RED	**0.17**	2.99	2.88	0.46	1.22	0.26	1.33
	Social LSTM	1.26	1.63	2.88	2.60	3.19	2.53	2.34
	Social GAN	2.04	0.92	0.19	0.32	0.11	0.61	0.70
	STGAT	0.46	**0.46**	**0.05**	0.13	**0.03**	**0.10**	**0.21**
	Trajectron	1.14	2.99	0.11	**0.11**	0.17	0.46	0.83
20 Outputs	Social GAN	1.26	**0.92**	0.38	0.82	0.30	0.52	0.61
	STGAT	**0.46**	**0.92**	**0.05**	0.11	0.03	0.10	**0.28**
	Trajectron	1.14	2.68	0.19	**0.11**	**0.17**	**0.32**	0.77

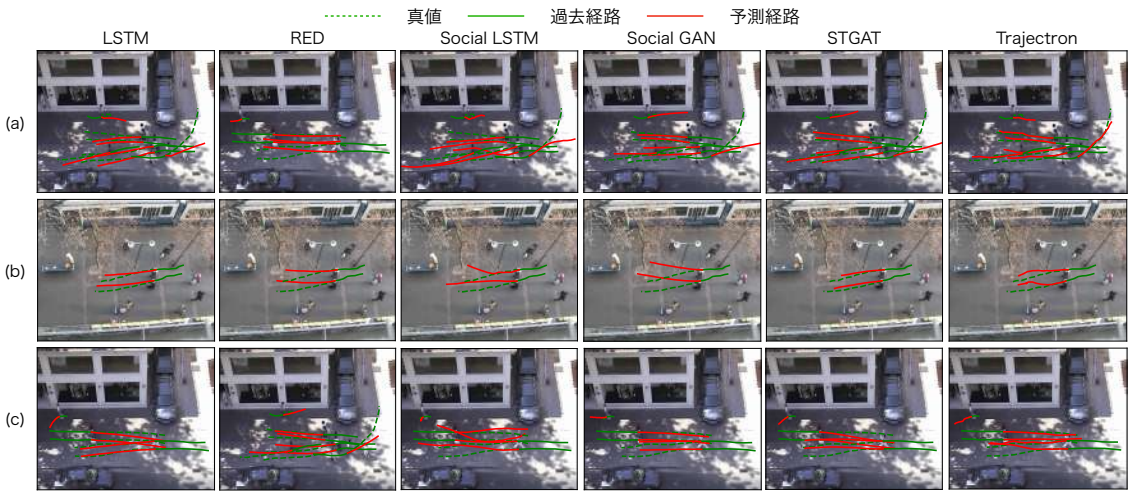


図 5.2: ETH/UCY における各予測モデルの予測結果例.

たと考えられる．図 5.2(c) は，対向者との衝突回避のシーン例である．図 5.2(c) より，インタラクションを考慮しない予測手法は線形的に予測していることがわかる．STGAT や Trajectron などは対向者との衝突を避けるような経路を予測していることがわかる．特に，STGAT は対向者との衝突を避けるために真値と異なる経路を予測してしまっている．この結果と表 5.4 及び，表 5.5 より，インタラクションを考慮した予測手法は歩行者同士が互いとの衝突を回避するために，真値に沿わない経路を予測し予測精度が落ちる一方，動的物体との衝突率を減少できることがわかる．

5.5 SDD における精度比較

SDD における予測誤差を表 5.7 に示す．表 5.7 より，Single Model では各シーンを平均した ADE で RED，FDE で RED と STGAT の予測誤差が低いことがわかる．また，20 Outputs では STGAT の予測誤差が最も低いことがわかる．SDD で RED の予測誤差が低くなった要因はドローンで撮影した位置に関係があると考えられる．SDD は ETH/UCY に比べ，高所で撮影されている．そのため，歩行者の動きが直線的になり過去の動きからインタラクションを考慮せず，線形的に予測する RED の予測誤差を低減したと考えられる．

次に，動的物体との衝突率を表 5.8，静的物体との衝突率を表 5.9 に示す．動的物体との衝突率の結果より，STGAT が Single model と 20 Outputs の両方で衝突率が最も低いことがわかる．ETH/UCY と同様に，STGAT は歩行者間の複雑なインタラクションを捉えていると言える．静的物体との衝突率の結果より，Single Model で Env LSTM，20 Outputs で STGAT が衝突率が最も低いことがわかる．Env LSTM の衝突率が低いことから，環境情報を導入することで周囲の障害物との衝突を避ける経路の予測に成功していると言える．

最後に，予測結果例を図 5.3 に示す．複数経路を予測する手法は，表 5.7 の Single Model の予測結果であることに注意されたい．図 5.3(a) は，線形の経路を辿る歩行者のシーン例である．図 5.3(a) より，ほとんどの予測手法は真値に似た経路を予測していることがわかる．特に，STGAT は歩行者同士が互いとの衝突を回避した経路を予測している．図 5.3(b) は，前方に街灯がある場合のシーン例を示す．図 5.3(b) より，環境情報を考慮した Env LSTM は前方の街灯との衝突を避ける経路を予測していることがわかる．図 5.3(c) は，混雑した十字路のシーン例である．図 5.3(c) より，ほとんどの予測手法が真値と似た経路を予測している一方，Social LSTM は真値と離れている経路を予測している．Social LSTM は図 5.3 の全シーンで真値と離れた経路を予測している．これは ETH/UCY の結

表 5.7: SDD における予測誤差．単位は [pixel].

		Scene								
	Method	bookstore	coupa	deathCircle	gates	hyang	little	nexus	quad	AVG
Single Model	LSTM	9.64 / 20.9	10.4 / 22.3	9.24 / 19.6	7.80 / 16.7	10.3 / 21.8	12.3 / 25.8	8.97 / 19.2	8.52 / 18.7	9.65 / 20.6
	RED	**6.66 / 13.5**	7.96 / 16.2	**7.42** / 14.8	**5.80 / 11.7**	**9.13 / 17.4**	**10.9** / 22.9	7.65 / 14.9	5.57 / 9.74	**7.64 / 15.1**
	Social LSTM	22.8 / 47.2	24.1 / 49.3	29.5 / 61.6	24.5 / 49.3	35.4 / 75.9	24.3 / 50.5	22.9 / 45.7	24.1 / 46.6	26.0 / 53.3
	Env LSTM	14.3 / 31.5	20.6 / 43.5	16.7 / 34.4	16.2 / 34.4	12.6 / 27.3	12.7 / 26.6	10.9 / 23.8	9.36 / 21.2	14.2 / 30.3
	Social GAN	18.4 / 37.2	19.1 / 38.4	18.1 / 36.5	18.1 / 36.5	19.4 / 39.1	20.8 / 42.5	18.6 / 37.4	21.1 / 41.4	19.2 / 38.6
	STGAT	7.58 / 14.6	9.00 / 17.4	7.57 / **14.4**	6.33 / **11.7**	9.17 / 17.9	**10.9 / 22.8**	7.37 / 14.0	**4.83 / 7.95**	7.84 / **15.1**
	Trajectron	8.79 / 19.2	**7.24 / 15.5**	14.3 / 33.0	6.29 / 13.0	9.55 / 21.6	12.4 / 29.0	**6.02 / 12.8**	6.80 / 15.1	8.92 / 19.9
20 Outputs	Social GAN	5.03 / 8.96	5.39 / 9.57	5.20 / 9.13	4.66 / 8.25	5.74 / **10.2**	6.51 / 11.9	5.23 / 9.11	4.08 / 7.15	5.23 / 9.28
	STGAT	**4.09 / 7.57**	**4.83 / 9.05**	**4.59 / 8.43**	**2.86 / 4.69**	**5.37 / 10.2**	**6.25** / 12.1	**4.20 / 7.58**	2.34 / **3.62**	**4.32 / 7.91**
	Trajectron	4.51 / 7.99	5.69 / 9.72	5.12 / 8.87	4.43 / 8.06	5.72 / 10.3	6.38 / **10.8**	5.55 / 9.43	**2.26** / 4.61	4.96 / 8.72

表 5.8: SDD における動的物体との衝突率．単位は [%].

	Method	Scene								AVG
		bookstore	coupa	deathCircle	gates	hyang	little	nexus	quad	
Single Model	LSTM	**3.62**	5.95	**39.76**	4.48	10.28	3.48	23.21	**0.0**	11.35
	RED	**3.62**	6.90	41.02	3.96	10.35	**2.43**	23.96	**0.0**	11.53
	Social LSTM	9.57	14.67	50.75	6.00	16.06	7.20	28.26	**0.0**	16.56
	Env LSTM	4.22	5.69	44.22	5.04	10.82	3.76	25.10	**0.0**	12.36
	Social GAN	3.86	6.12	40.50	4.09	10.96	4.11	24.09	**0.0**	11.72
	STGAT	**3.62**	**5.22**	**39.76**	**3.66**	10.28	3.66	**20.09**	**0.0**	**10.79**
	Trajectron	4.85	9.65	48.87	4.70	12.06	3.32	26.38	**0.0**	13.73
20 Outputs	Social GAN	3.62	5.66	38.26	**3.96**	10.88	3.96	22.98	**0.0**	11.17
	STGAT	**2.85**	**4.98**	**36.77**	4.57	**9.28**	**3.32**	**20.09**	**0.0**	**10.23**
	Trajectron	3.62	7.32	44.22	4.70	10.01	**3.32**	22.98	**0.0**	12.02

果で議論したように，負の対数尤度関数を最小化するように学習すると，サンプリングプロセスが微分不可能になるため誤差逆伝播が困難になり，良いパラメータを学習できなかったため予測に失敗したと考えられる．

5.6　各モデルの計算時間とパラメータの比較

最後に，モデルの計算時間とパラメータ数の比較を表 5.10 に示す．表 5.10 のパラメータ数より，動的物体との衝突回避に関するインタラクションを考慮するモデル及び静的環境特徴を抽出する Env LSTM のパラメータ数が各インタラクションを考慮しない LSTM，RED と比べ必然的に多くなっている．動的物体との衝突回避に関するインタラクションを考慮するモデルでは，Social LSTM が最もパラメータ数が多いことがわかる．これは，LSTM 及び出力の次元数が他の手法と比べて多いのが原因と考えられる．LSTM の次元数が 128 と多い，Env LSTM，Social LSTM，Trajectron のパラメータ数が多いこともわかる．これは，2.1.2 節で述べた LSTM の内部には重みパラメータが複数あり，これが原因で各モデルのパラメータが多くなっていると考えられる．そのため，経路予測のパラメータ数は LSTM の次元数に大きく依存すると考えられる．プーリングモデルの Social GAN とアテンションモデルの STGAT では，Social GAN の方がパラメータ数が多い．これは，Social GAN のプーリング処理の全結合層の次元数が多いためと考えられる．

次に計算時間では，インタラクションを考慮するモデルは人数が増加する毎に計算時間が増加している．Social LSTM 及び，Env LSTM の時間が他と比べて多いのは，過去と未来の両時刻で予測対

表 5.9: SDD における静的物体との衝突率．単位は [%].

	Method	Scene								AVG
		bookstore	coupa	deathCircle	gates	hyang	little	nexus	quad	
Single Model	LSTM	**1.90**	2.12	**5.19**	1.83	4.12	3.58	6.78	**0.0**	3.19
	RED	2.44	1.30	5.29	2.27	6.43	**2.56**	5.67	5.88	3.96
	Social LSTM	6.44	5.22	6.70	6.70	4.66	5.06	3.23	2.94	5.12
	Env LSTM	2.19	1.79	4.27	**1.73**	**0.90**	3.13	1.17	**0.0**	**1.90**
	Social GAN	3.74	9.44	6.70	8.17	2.06	3.94	3.94	8.59	5.82
	STGAT	2.36	**1.22**	5.30	2.21	7.49	4.58	7.46	**0.0**	3.83
	Trajectron	1.93	2.77	4.63	2.21	5.57	3.35	5.87	2.94	3.66
20 Outputs	Social GAN	3.74	6.55	5.30	7.66	**2.06**	3.94	3.13	2.94	4.42
	STGAT	2.36	**1.22**	**4.63**	**1.98**	6.43	3.94	6.78	**0.0**	**3.42**
	Trajectron	**1.93**	2.12	**4.63**	4.28	5.57	**3.13**	**5.87**	2.94	3.81

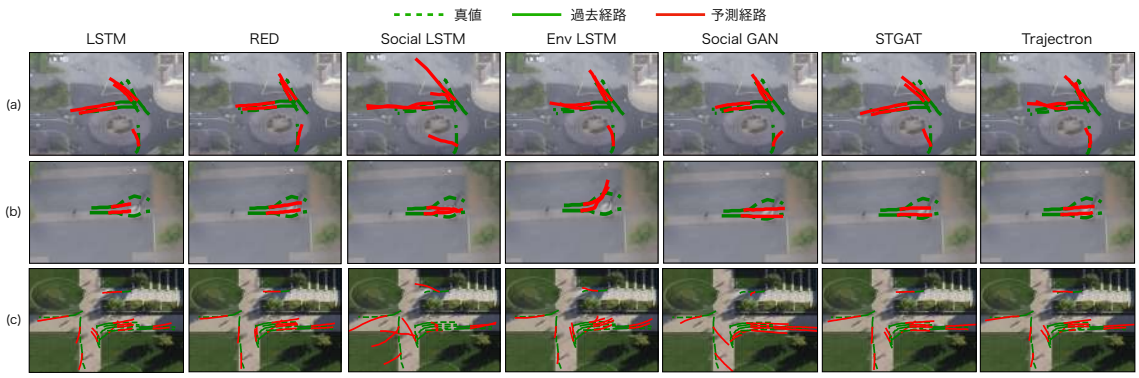


図 5.3: SDD における各予測モデルの予測結果例.

象を中心としてグリッドの作成や，セマンティックラベルを逐次読み込む必要があるためだと考えられる．また，未来時刻に予測対象周囲を見ていない Social GAN，STGAT，Trajectron は人数が増加しても計算時間に僅かな差しかないことがわかる．この理由は，過去時刻でそれぞれのインタラクションを表現しており，Social LSTM のように未来時刻でインタラクションを逐次表現していないためと考えられる．

表 5.10: 各モデルの計算時間及びパラメータ比較．計算時間はシーンに歩行者が 10 人・50 人・100 人・1,000 人いた場合の結果である．実験環境は GPU が Quadro RTX 8000，CPU が Intel Xeon Gold 6226，動作クロック 2.7GHz，コア数 12，スレッド数 24，RAM メモリ 196GB である．

モデル名	計算時間 [sec.]				パラメータ数
	10 人	50 人	100 人	1,000 人	
LSTM	0.067	0.071	0.072	0.076	8,610
RED	0.079	0.081	0.085	0.087	17,154
Env LSTM	0.250	1.737	2.486	6.864	456,930
Social LSTM	0.641	4.308	6.701	24.688	264,069
Social GAN	0.232	0.479	0.505	0.876	60,234
STGAT	0.179	0.239	0.556	0.987	44,630
Trajectron	0.115	0.282	0.458	0.897	138,164

5.7 プーリングモデルとアテンションモデルの違いに対する考察

動的物体及び静的物体それぞれとのインタラクションを表現したモデル化をすることで，予測精度の向上やそれぞれの物体との衝突率を減少できることが実験より示された．特に，動的物体とのインタラクションを表現した予測モデルでは，アテンションモデルの方がプーリングモデルより精度が高い．また，アテンションモデル同士では STGAT の方が Trajectron より予測誤差の低減及び動的物体との衝突率が低い．これらは以下の理由だと考えられる．

- プーリングモデル：Social LSTM は予測対象を中心に，予め定めたグリッドサイズに従いグリッド内の他対象に関する特徴を埋め込むことでインタラクションを表現している．一方で，範囲外の他対象の特徴は埋め込まれないため，インタラクションを十分に表現できず予測精度が低下したと考えられる．Social GAN はグリッドサイズの制限がないため，シーン内の全他対象とのインタラクションを式 (2.12) で表現している．一方で，他対象の特徴から maxpool. で最大の特徴のみを選択するため，衝突回避に重要な他対象の情報が欠落する．結果，動的物体との衝突率が高くなり，予測精度も低下したと考えられる．

- アテンションモデル：アテンションモデルは，重みの大小で予測対象と他対象間の関係を表現し，この重みをそれぞれの対象の特徴量と乗算を行うことで，全ての対象とのインタラクションを表現している．そのため，プーリングモデルより精度が高くなったと考えられる．また，アテンションモデル同士で比較すると，Trajectron が STGAT と比較して僅かに精度が低いことがわかる．これは，Trajectron が過去時刻に予測対象を中心とした周囲 1.5[m] または 40[pixel] 以内の他対象とのグラフを構築するため，STGAT と比べ予測精度の低下と動的物体との衝突率が高くなったと考えられる．

5.8　まとめ

本章では，インタラクションを考慮した代表的な予測モデルの精度の検証を行った．歩行者間のインタラクションを考慮した手法は歩行者密度が高いシーンで予測誤差が低くなることがわかった．また，インタラクションを考慮することで，動的物体との衝突率が低くなることがわかった．プーリングモデルと比べアテンションモデルによるインタラクション表現が予測精度を向上しつつ衝突率を減少させることがわかった．一方で，歩行者密度が低いシーンでは，インタラクションを考慮しない予測手法で十分な予測精度を獲得できることがわかった．予測結果より，インタラクションを考慮した手法は移動対象同士が互いとの衝突を回避するために，真値に沿わない経路を予測し予測精度が落ちる一方，他対象との衝突回避する経路を予測した．また，環境情報を考慮することで，静的物体との衝突率を減少させ，障害物との衝突を避ける経路を予測することを確認した．さらに，動的物体及び静的物体のそれぞれの衝突回避に関するインタラクションを考慮する予測手法はパラメータ数や計算時間が増加することを確認した．

第6章

グループレベルのインタラクションによる経路予測

本章では，混雑シーンにおける人間の社会的インタラクションを捉えるために，4 章で得た知見を活かしてグループレベルのインタラクションによる経路予測手法を提案する．1 章で述べたように，経路予測は歩行者や自動車等の対象物間の社会的インタラクションを考慮することで，衝突を回避した経路を予測できる．経路予測や社会的インタラクションのモデル化は，自動運転や自律ロボットなどのアプリケーションの基盤技術として活発に研究されている [33, 34, 31, 32, 35]．特に，人間の将来の行動をモデル化することは，人間社会を支える安全な自律システムの開発にとって重要なステップである．しかし，人間と人間の社会的インタラクションは複雑であり，捉えることが困難な場合が多い．特に，混雑シーンにおける人の行動は多様な行動パターンの変化が起こるため，より困難となる．

一般的に混雑シーンの歩行者は，衝突を避けたり周囲の集団の動きに合わせるなどの暗黙の社会的ルールに従う傾向がある．多くの場面で歩行者の大半はグループで歩いている [128] こと，歩行者の 7 割は家族や友人などのグループで歩いており [38]，同じ方向に歩いている人と自発的にグループを形成しているという研究結果がある．このような社会的インタラクションをモデル化することは，正確な経路予測を行う上で非常に重要になる．図 6.1 に示す歩行者グループの簡単な例を示す．黒色の歩行者が予測対象とする．この歩行者の将来の経路は，自分が属するグループと同じ方向に進み，他のグループとの衝突を避けるといったグループ間及び，グループ内のインタラクションの影響を強く受ける．しかし，古典的なアプローチ [39, 32, 30, 129, 40] はこのような社会的なインタラクションを無視して経路予測を行う．個人レベルのインタラクションを捉える予測手法 [1, 20, 3, 9, 8, 60, 11] もあるが，歩行者の複雑なグループレベルの関係を捉えることができない．

混雑シーンにおけるグループベースの社会的インタラクションをモデル化した経路予測手法はいくつか提案されている [56, 17]．しかし，これらの方法は社会的インタラクションの限られた側面しか考慮しておらず，現実的な経路予測が困難である．図 6.2(b) にグループベースの予測手法 [56] の予測結果例を示す．右側のグループの青色の対象はグループ間のインタラクションしか考慮していないため，左側のグループとの衝突を避け真値 (図 6.2(a)) と異なる経路を予測する．そして，インタラクション対象ではない緑色の対象と衝突するような経路の予測，右側のグループの赤色の対象と黄色の対象もそれぞれがインタラクション対象とされないため，将来で衝突するような経路を予測する．また，歩行者は反対方向から来た人の将来の位置を予想することで衝突を回避するが，従来手法では過去の経路のみを利用しているため，将来の他者の位置に基づく社会的インタラクション

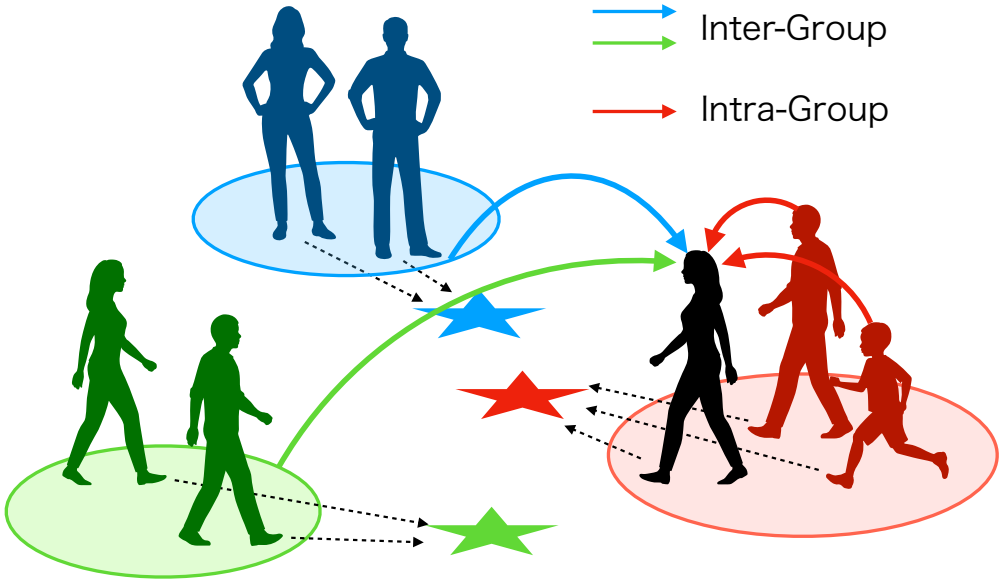


図 6.1: 歩行者間の社会的インタラクションの例．グループ内のインタラクションは星マークのように同じ目的地に向かう歩行者グループ，グループ間のインタラクションは他のグループとの衝突を避ける経路を辿る．

図 6.2: 先行研究及び，提案手法の予測結果例．破線は過去の経路，実線は未来の経路を表す．

をモデル化していない．直感的には，歩行者は数秒先に移動しそうな他対象の位置を予想することで，衝突を回避できる．5 章で述べたように，従来の個人レベル及びグループレベルの予測手法では過去時刻で社会的インタラクションをモデル化するため，シーンによってはインタラクションをモデル化しないシンプルな予測モデルの方が予測誤差や衝突率が低減する．

本研究の目的は，混雑シーンにおける経路予測のために人間の社会的インタラクションを捉えることである．この目的のために，歩行者間の複雑なグループ間とグループ内の社会的インタラクションをモデル化する Group-based Forecasting Module を提案する．Group-based Forecasting Module では，社会的インタラクションに関連する重要な対象に着目させるために，Attention 機構 [52] を用いる．また，人間は反対方向から来る人の速度や位置などから数秒先に移動しそうな位置を事前に予想す

ることで，衝突を回避できる．そこで，将来の他者の位置を事前に予測し，未来の位置に基づく社会的インタラクションを捉えるために，Prospection Module を導入する．さらに，グループ情報の動的な変化を反映するために，Group-based Forecasting Module の予測経路から Clustering Module でグループラベルを逐次更新する．従来の予測手法のほとんどが，真値と予測値間の L2 距離の loss を最小化するため，未来の経路を平均的に予測する．そのため，L2 loss では歩行者の将来の複数の経路を予測する際，1 つの予測経路として制限される可能性がある．そのため，提案手法では L2 loss に加え，Generative Adversarial Network (GAN) [85] のような構造で現実的な経路を予測する．各対象に対して，Group-based Forecasting Module はノイズベクトルの追加により複数の経路を予測する．この予測経路と真値を Discriminator で敵対的に学習させる．これにより図 6.2(c) に示すように，先行研究と比べて正確な経路を予測できる．

評価実験において，Group-based Forecasting Module を導入することによる精度の変化を検証する．具体的には，3 つの公開されているデータセットを用いて提案手法と従来手法で比較実験を行った．実験結果より，提案手法は全ての予測手法より正確に経路を予測できることを示す．

本章の構成は以下の通りである．まず，6.1 節では問題設定と提案手法の肝となる Group-based Forecasting Module について述べる．6.2 節では提案手法の有効性を確認するための評価実験について述べる．最後に 6.3 節で本章をまとめる．

6.1 Group-based Forecasting Module による経路予測

本節では，Group-based Forecasting Module で歩行者間の社会的インタラクションを考慮した経路予測を実現する．初めに提案手法の大まかな概要及び問題設定を述べ，その後 Group-based Forecasting Module の詳細を述べる．

6.1.1 Overview

提案手法のネットワーク構造を図 6.3，本章で用いる数式記号を表 6.1 に示す．先行研究 [3, 8] に従い，シーン内に歩行者が N 人いると仮定する．本研究は観測時刻期間 $T_{obs} = \{1, \ldots, t_{obs}\}$ の経路を入力し，予測時刻期間 $T_{pred} = \{t_{obs}+1, \ldots, t_{pred}\}$ の経路を予測する．ここで，t_{obs} と t_{pred} はそれぞれ最終観測時刻と予測最終時刻を表す．

まず，Trajectory Encoder は i 番目の予測対象の観測経路 $X_i^{T_{obs}} = \{\boldsymbol{x}_i^t = (x_i^t, y_i^t)\}, \forall t \in T_{obs}$ から，各対象の hidden state $\boldsymbol{h}_i^{enc}$ を求める．複数の経路を予測するために $\boldsymbol{h}_i^{enc}$ は多変量正規分布からのノイズベクトル $\boldsymbol{z}$ と結合し，1 層の全結合 $\phi(\cdot)$ を介して Internal State $\boldsymbol{h}_i$ を求める．この Internal State を次の Decoder Module の入力として利用する．

デコーダは，Prospection Module，Group-based Forecasting Module 及び Clustering Module の 3 つで構成される．Prospection Module は，他対象との将来のインタラクションを捉えるのが 1 つの予測モデルではモデル化できないため導入される．Prospection Module は，予想時刻期間 $T_{pros} = \{t_{obs}+1+\lambda, \ldots, t_{pred}+\lambda\}$ における予想経路 $\bar{Y}_i^{T_{pros}} = \{\bar{\boldsymbol{y}}_i^t = (\bar{x}_i^t, \bar{y}_i^t)\}, \forall t \in T_{pros}$ を出力する．Prospection Module の他グループに対する予想経路を Group-based Forecasting Module に伝播することで，未来の位置に基づく社会的インタラクションを捉える経路を予測できる．λ は Prospection Module がどの

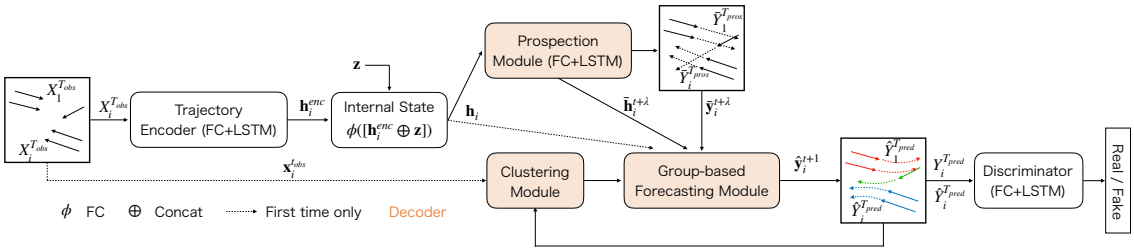


図 6.3: 提案手法のモデル構造．提案手法は Trajectory Encoder，Internal State，Prospection Module，Clustering Module，Group-based Forecasting Module 及び，Discriminator の 6 つのモジュールで構成される．Group-based Forecasting Module は，Internal State と Prospection Module から出力される経路情報及び，Clustering Module で対象毎にグループとしてクラスタリングされた結果を用いて，次時刻の経路を予測する．グループは時間と共に常に変化しているため，Group-based Forecasting Module の予測経路は Clustering Module に逐次入力される．Discriminator は実際の経路と予測された経路を判別するように敵対的に学習される．

表 6.1: 本章で用いる数式記号.

Symbol	Description
t_{obs}	end of the observation time step
t_{pred}	end of the prediction time step
λ	control parameter of future prospection
i	target pedestrian
j	other pedestrians
k	group ID
G	set of target indices whose group label
T	transpose
$\oplus$	concatenation operation
(x, y)	past and future ground-truth trajectory
$(\bar{x}, \bar{y})$	prospective trajectory of prospection module
$(\hat{x}, \hat{y})$	predicted trajectory of group-based forecasting module
$\boldsymbol{h}$	hidden state
$\boldsymbol{z}$	noise vector
$\boldsymbol{r}, \boldsymbol{o}$	spatial feature vector between targets
$\boldsymbol{q}, \boldsymbol{p}$	query vector
$\boldsymbol{k}, \boldsymbol{s}$	key vector
$\boldsymbol{v}, \boldsymbol{u}$	value vector
d	number of query's dimension
α, β, γ	attention weight
$\boldsymbol{b}, \boldsymbol{w}, \boldsymbol{l}$	aggregated values
$\phi.$	FC layer

程度先を予想するかを制御するパラメータである．すなわち，$t + \lambda$ は数秒先の時刻を表し，T_{pros} は予測時刻期間 T_{pred} から未来に λ フレーム分シフトされた時刻である．Clustering Module は，最初に最終観測時刻の座標 $\boldsymbol{x}_i^{t_{obs}}$ から初期グループラベルを取得する．次に，Group-based Forecasting Module では複数の経路 $\hat{Y}_i^{T_{pred}} = \{\hat{\boldsymbol{y}}_i^t = (\hat{x}_i^t, \hat{y}_i^t)\}, \forall t \in T_{pred}$ を予測する．予測対象に対応する hidden state $\boldsymbol{h}_i$ と共に，Clustering Module でクラスタリングされたグループラベルと他の歩行者の予想経路 $\bar{Y}_j^{T_{pros}}, \forall j \neq i$ とその hidden state $\{\bar{\boldsymbol{h}}_j^t\}, \forall t \in T_{pros}$ が Group-based Forecasting Module へ入力される．Group-based Forecasting Module で求めた予測座標 $\hat{\boldsymbol{y}}_i^t$ が Clustering Module に逐次入力する．これは，グループが集結，解散により時間と共に変化しグループ情報を動的に変更されるためである．Discriminator では，予測経路 $\hat{Y}_i^{T_{pred}}$ と真値 $Y_i^{T_{pred}} = \{(x_i^t, y_i^t)\}$ から，全結合層と LSTM で実際の経路か予測経路かを分類する．

6.1.2 Group-based Forecasting Module

本節では，Group-based Forecasting Module の詳細について述べる．Group-based Forecasting Module は図 6.4 のように，全結合層，LSTM，グループ間及びグループ内の Attention 機構で構成されている．Group-based Forecasting Module は現時刻の座標 $\hat{\boldsymbol{y}}_i^t$ を全結合層に埋め込み，LSTM で hidden state $\hat{\boldsymbol{h}}_i^t$ を求める．グループ間の Attention 機構への入力は，λ 分フレームシフトされた Prospection Module の予想経路 $\bar{\boldsymbol{y}}_j^{t+\lambda}$ と hidden state $\bar{\boldsymbol{h}}_j^{t+\lambda}$，予測対象の現時刻の経路 $\hat{\boldsymbol{y}}_i^t$ と hidden state $\hat{\boldsymbol{h}}_i^t$ である．グループ間の Attention 機構は，Source-Target Attention [52] で他グループに対応する hidden state に注意重みを割り当てる．同様に，グループ内の Attention 機構は同じグループに属する他の歩行者を対象とし，これら個人の hidden state に注意重みを割り当てる．Group-based Forecasting Module は，LSTM の hidden state $\hat{\boldsymbol{h}}_i^t$ と各 Attention 機構の出力 $\boldsymbol{b}_i^t$ と $\boldsymbol{w}_i^t$ を連結し，2 つの全結合層を介して未来の hidden state $\hat{\boldsymbol{h}}_i^{t+1}$ と座標 $\hat{\boldsymbol{y}}_i^{t+1}$ を出力する．hidden state $\hat{\boldsymbol{h}}_i^{t+1}$ は Group-based Forecasting Module の LSTM に再帰的に入力される．このように，グループ間及びグループ内の Attention 機構を動的に割り当てることで，予測時刻毎にインタラクション情報を逐次更新し，将来の経路を予測する．同様に，予測座標 $\hat{\boldsymbol{y}}_i^{t+1}$ を Clustering Module に逐次入力することで，グループ情報を動的に変化させる．Group-based Forecasting Module のアルゴリズムを Algorithm 1 に示す．

■ グループ間の Attention 機構

i 番目の予測対象に対し，グループ間の Attention 機構では予想された経路情報を用いて他グループに対応する注意重みを抽出する．グループ間の Attention 機構では，初めに全結合層 $\phi_{Br}(\cdot)$ を介して

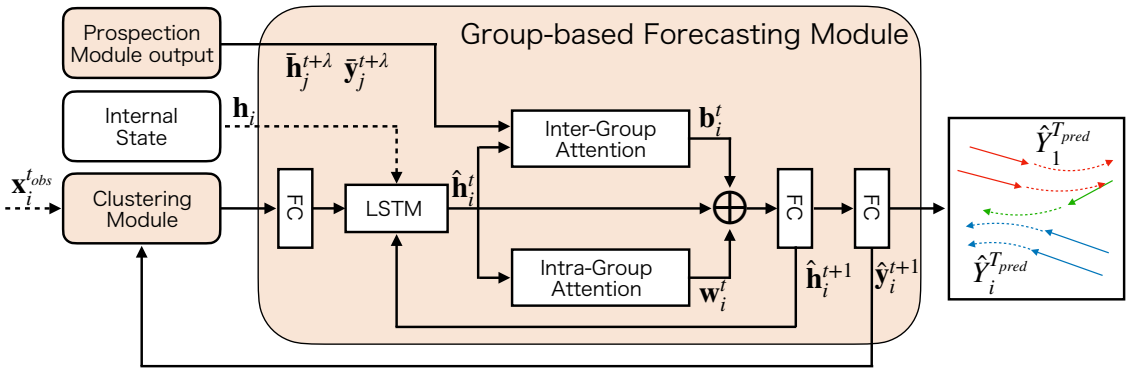


図 6.4: Group-based Forecasting Module の内部構成．Group-based Forecasting Module は，デコーダの LSTM とグループ間及び，グループ内の Attention 機構により社会的なインタラクションを考慮した経路を予測する．

Algorithm 1 Group-based Forecasting Module のアルゴリズム.

Function Inter-Group($\hat{x}_i^t, \hat{y}_i^t, \bar{x}^{t+\lambda}, \bar{y}^{t+\lambda}, \hat{\boldsymbol{h}}_i^t, \bar{\boldsymbol{h}}^{t+\lambda}, G^t$)
if Group ID k to which the prediction target i does *not* belong **then**
  $\boldsymbol{r}_{i,k}^t \leftarrow \phi_{Br}(\hat{x}_i^t, \hat{y}_i^t, \bar{x}_j^{t+\lambda}, \bar{y}_j^{t+\lambda})$ # $j \in G_k^t$
  $\boldsymbol{q}_i^t \leftarrow \phi_{Bq}(\hat{\boldsymbol{h}}_i^t)$
  $\boldsymbol{k}_{i,k}^t \leftarrow \phi_{Bk}(\boldsymbol{r}_{i,k}^t, \bar{\boldsymbol{h}}_j^{t+\lambda})$
  $\boldsymbol{v}_{i,k}^t \leftarrow \phi_{Bv}(\boldsymbol{r}_{i,k}^t, \bar{\boldsymbol{h}}_j^{t+\lambda})$
  $\alpha_{i,k}^t \leftarrow \text{attn}(\boldsymbol{q}_i^t, \boldsymbol{k}_{i,k}^t)$
  $\boldsymbol{b}_i^t \leftarrow \text{weighted average}(\alpha_{i,k}^t, \boldsymbol{v}_{i,k}^t)$
  return $\boldsymbol{b}_i^t$
end if
Function Intra-Group($\hat{x}^t, \hat{y}^t, \hat{\boldsymbol{h}}^t, G^t$)
if Group ID k to which the prediction target i does belong **then**
  $\boldsymbol{o}_{i,j}^t \leftarrow \phi_{Wr}(\hat{x}_i^t, \hat{y}_i^t, \hat{x}_j^t, \hat{y}_j^t)$ # $j \in G_k^t$
  $\boldsymbol{p}_i^t \leftarrow \phi_{Wq}(\hat{\boldsymbol{h}}_i^t)$
  $\boldsymbol{s}_{i,k}^t \leftarrow \phi_{Wk}(\boldsymbol{r}_{i,j}^t, \hat{\boldsymbol{h}}_j^t)$
  $\boldsymbol{u}_{i,j}^t \leftarrow \phi_{Wv}(\boldsymbol{r}_{i,j}^t, \hat{\boldsymbol{h}}_j^t)$
  $\beta_{i,j}^t \leftarrow \text{attn}(\boldsymbol{p}_i^t, \boldsymbol{s}_{i,j}^t)$
  $\boldsymbol{w}_i^t \leftarrow \text{weighted average}(\beta_{i,j}^t, \boldsymbol{u}_{i,j}^t)$
  return $\boldsymbol{w}_i^t$
end if
for $t \in T_{pred}$ **do**
  $G^t \leftarrow \varnothing$
  if $t =$ start prediction time **then**
    $G^t \leftarrow$ Clustering Module($x^{t_{obs}}$)
  else
    $G^t \leftarrow$ Clustering Module($\hat{\boldsymbol{y}}^t$)
  end if
  for $i \in N$ **do**
    if $t =$ start prediction time **then**
      $\boldsymbol{b}_i^t \leftarrow$ Inter-Group($x_i^{t_{obs}}, y_i^{t_{obs}}, \bar{x}^{t+\lambda}, \bar{y}^{t+\lambda}, \hat{\boldsymbol{h}}_i^t, \bar{\boldsymbol{h}}^{t+\lambda}, G^t$)
      $\boldsymbol{w}_i^t \leftarrow$ Intra-Group($x^{t_{obs}}, y^{t_{obs}}, \hat{\boldsymbol{h}}^t, G^t$)
    else
      $\boldsymbol{b}_i^t \leftarrow$ Inter-Group($\hat{x}_i^t, \hat{y}_i^t, \bar{x}^{t+\lambda}, \bar{y}^{t+\lambda}, \hat{\boldsymbol{h}}_i^t, \bar{\boldsymbol{h}}^{t+\lambda}, G^t$)
      $\boldsymbol{w}_i^t \leftarrow$ Intra-Group($\hat{x}^t, \hat{y}^t, \hat{\boldsymbol{h}}^t, G^t$)
    end if
    $cat \leftarrow [\hat{\boldsymbol{h}}_i^t \oplus \boldsymbol{b}_i^t \oplus \boldsymbol{w}_i^t]$
    $\hat{\boldsymbol{h}}_i^{t+1} \leftarrow \phi_h(cat)$
    $\hat{\boldsymbol{y}}_i^{t+1} \leftarrow \phi_y(\hat{\boldsymbol{h}}_i^{t+1})$
  end for
end for

予測対象の位置と他グループの平均位置との空間的な相対距離から特徴ベクトルを下式で計算する.

$$\boldsymbol{r}_{i,k}^t = \phi_{Br}(\frac{1}{|G_k^t|}\sum_{j \in G_k^t} \bar{x}_j^{t+\lambda} - \hat{x}_i^t, \frac{1}{|G_k^t|}\sum_{j \in G_k^t} \bar{y}_j^{t+\lambda} - \hat{y}_i^t). \tag{6.1}$$

ここで，k はグループ ID，G_k^t は時刻 t におけるグループラベルが k である対象のインデックスの集合を表す．なお，ここでは予測対象 i のグループは考慮しないため，i 番目の対象はいずれのグループ k にも属されない．$(\hat{x}_i^t, \hat{y}_i^t)$ 及び $(\bar{x}_j^{t+\lambda}, \bar{y}_j^{t+\lambda})$ はそれぞれ予測対象の現在位置，他グループの対象の予想座標である．

次に各全結合層を介して，Source-Target Attention の特徴ベクトルを計算する．Target は予測対象に対応し，Query は d 次元のベクトル $\boldsymbol{q}_i^t = \phi_{Bq}(\hat{\boldsymbol{h}}_i^t)$ として定義される．Source は他グループに対応し，Key $\boldsymbol{k}_{i,k}^t$ と Value $\boldsymbol{v}_{i,k}^t$ は Prospection Module からの hidden state の合計と特徴ベクトル $\boldsymbol{r}_{i,k}^t$ を連結して次のように計算される．

$$\boldsymbol{k}_{i,k}^t = \phi_{Bk}([\boldsymbol{r}_{i,k}^t \oplus \sum_{j \in G_k^t} \bar{\boldsymbol{h}}_j^{t+\lambda}]), \tag{6.2}$$

$$\boldsymbol{v}_{i,k}^t = \phi_{Bv}([\boldsymbol{r}_{i,k}^t \oplus \sum_{j \in G_k^t} \bar{\boldsymbol{h}}_j^{t+\lambda}]). \tag{6.3}$$

ここで $\oplus$ は連結記号，$\bar{\boldsymbol{h}}_j^{t+\lambda}$ は Prospection Module の hidden state を表す．グループ間の Attention 機構の注意重み $\alpha_{i,k}^t$ と最終出力 $\boldsymbol{b}_i^t$ は，それぞれの特徴量を用いて次のように計算される．

$$\alpha_{i,k}^t = \frac{\exp(\boldsymbol{q}_i^t \boldsymbol{k}_{i,k}^{t\mathrm{T}})}{\sqrt{d}\sum_k \exp(\boldsymbol{q}_i^t \boldsymbol{k}_{i,k}^{t\mathrm{T}})}, \tag{6.4}$$

$$\boldsymbol{b}_i^t = \sum_k \alpha_{i,k}^t \boldsymbol{v}_{i,k}^t. \tag{6.5}$$

ここで，$\alpha_{i,k}^t$ は時刻 t における他のグループラベル k から i 番目の予測対象に対する注目重みである．Query と Key の内積から，予測対象と他グループ間の関連性 $\alpha_{i,k}^t$ を Softmax 関数で計算する．算出した関連性に Value を乗算することで，予測対象と他グループとの空間的関係を考慮した出力 $\boldsymbol{b}_i^t$ が得られる．

■ グループ内の Attention 機構

グループ間の Attention 機構と異なり，グループ内の Attention 機構では現時刻の経路情報を用いて予測対象と同じグループ内の他対象に対する注意重みを計算する．グループ内の Attention 機構は，初めに全結合層 $\phi_{Wr}(\cdot)$ を介して予測対象とグループ内の他対象の現在位置の空間的関係を次のように計算する．

$$\boldsymbol{o}_{i,j}^t = \phi_{Wr}(\hat{x}_j^t - \hat{x}_i^t, \hat{y}_j^t - \hat{y}_i^t). \tag{6.6}$$

ここで j は予測対象自身を含む同じグループ内の対象の ID である．

Target が予測対象に対応し，Query が LSTM の hidden state を用いてグループ間の Attention 機構と同様に $\boldsymbol{p}_i^t = \phi_{Wq}(\hat{\boldsymbol{h}}_i^t)$ として計算される．Source はグループ内の他対象に対応し，Key $\boldsymbol{s}_{i,j}^t$ と Value

$\boldsymbol{u}_{i,j}^t$ はそれぞれ次のように計算される．

$$\boldsymbol{s}_{i,j}^t = \phi_{Wk}([\boldsymbol{o}_{i,j}^t \oplus \hat{\boldsymbol{h}}_j^t]), \tag{6.7}$$

$$\boldsymbol{u}_{i,j}^t = \phi_{Wv}([\boldsymbol{o}_{i,j}^t \oplus \hat{\boldsymbol{h}}_j^t]). \tag{6.8}$$

グループ内の Attention 機構の注意重み $\beta_{i,j}^t$ と最終出力 $\boldsymbol{b}_i^t$ は次のように計算される．

$$\beta_{i,j}^t = \frac{\exp(\boldsymbol{p}_i^t \boldsymbol{s}_{i,j}^{t\mathrm{T}})}{\sqrt{d}\sum_j \exp(\boldsymbol{p}_i^t \boldsymbol{s}_{i,j}^{t\mathrm{T}})}, \tag{6.9}$$

$$\boldsymbol{w}_i^t = \sum_j \beta_{i,j}^t \boldsymbol{u}_{i,j}^t. \tag{6.10}$$

ここで，$\beta_{i,j}^t$ は時刻 t におけるグループ内の他対象 j から予測対象 i への注意重みである．$\boldsymbol{w}_i^t$ は予測対象 i に対する集合値で同じグループ内の他対象からの空間的関係が含まれている．グループ内の Attention 機構により，同じグループ内の対象同士が相互に影響を与える経路予測に期待できる．

■ 学習方法と損失計算

提案手法は 2 段階アプローチで学習を行う．まず，Trajectory Encoder と Decoder の Prospection Module を学習する．次に，学習済みの Trajectory Encoder の重みを固定し，Decoder の Group-based Forecasting Module を学習する．

Prospection Module は真値と予想経路間の L2 loss $\mathcal{L}_p$ で学習する．

$$\mathcal{L}_p = \sum_{i=1}^{N} \sum_{t \in T_{pros}} ||\bar{\boldsymbol{y}}_i^t - \boldsymbol{y}_i^t||_2. \tag{6.11}$$

Group-based Forecasting Module は，L2 loss と Adversarial loss で学習する．Discriminator D は，予測された経路を本物または偽物として分類し，Generator G が現実的な経路を予測できるように学習する．Adversarial loss $\mathcal{L}_{GAN}$ と L2 loss は次のように計算される．

$$\mathcal{L}_g = \sum_{i=1}^{N} (\mathcal{L}_{GAN} + \sum_{t \in T_{pred}} ||\hat{\boldsymbol{y}}_i^t - \boldsymbol{y}_i^t||_2). \tag{6.12}$$

ここで，Adversarial loss を次のように定義する．

$$\begin{aligned} \mathcal{L}_{GAN} = & \min_G \max_D \mathbb{E}_{Y_i^{T_{pred}} \sim p(Y_i^{T_{pred}})} [\log D(Y_i^{T_{pred}})] \\ & + \mathbb{E}_{X_i^{T_{obs}} \sim p_{(X_i^{T_{obs}})}, \boldsymbol{z} \sim p_{(\boldsymbol{z})}} [\log(1 - D(G(X_i^{T_{obs}}, \boldsymbol{z})))]. \end{aligned} \tag{6.13}$$

■ 個人レベルの Attention 機構

各グループの Attention 機構の有効性を調査するために，個人レベルの Attention 機構をモデル化する．図 6.5 に個人レベルの Attention 機構の内部構造を示す．Group-based Forecasting Module (図 6.4) と異なり，1 つの Attention 機構で構成される．

個人レベルの Attention 機構は，全結合層 $\phi_{Lr}(\cdot)$ を介して他対象の未来の予想位置と予測対象の現時刻の位置から空間的な関係を次のように計算する．

$$\boldsymbol{r}_{i,j}^{t} = \phi_{Lr}(\bar{x}_{j}^{t+\lambda} - \hat{x}_{i}^{t}, \bar{y}_{j}^{t+\lambda} - \hat{y}_{i}^{t}). \tag{6.14}$$

ここで，j はシーン内の他対象を表す．$(\bar{x}_{j}^{t+\lambda}, \bar{y}_{j}^{t+\lambda})$ と $(\hat{x}_{i}^{t}, \hat{y}_{i}^{t})$ は，それぞれ他対象の未来の予想位置と予測対象の現在の位置を表す．

個人レベルの Attention 機構においても Source-Target Attention を用いる．Target は予測対象に対応し，Query は全結合層を介して $\boldsymbol{q}_{i}^{t} = \phi_{Lq}(\hat{\boldsymbol{h}}_{i}^{t})$ として計算される．Source は他対象に対応し，Key $\boldsymbol{k}_{i,j}^{t}$ と Value $\boldsymbol{v}_{i,j}^{t}$ はそれぞれ次のように計算される．

$$\boldsymbol{k}_{i,j}^{t} = \phi_{Lk}([\boldsymbol{r}_{i,j}^{t} \oplus \hat{\boldsymbol{h}}_{j}^{t+\lambda}]), \tag{6.15}$$

$$\boldsymbol{v}_{i,j}^{t} = \phi_{Lv}([\boldsymbol{r}_{i,j}^{t} \oplus \hat{\boldsymbol{h}}_{j}^{t+\lambda}]). \tag{6.16}$$

個人レベルの Attention 機構の注意重み $\gamma_{i,j}^{t}$ と最終出力 $\boldsymbol{l}_{i}^{t}$ は次のように計算される．

$$\gamma_{i,j}^{t} = \frac{\exp(\boldsymbol{q}_{i}^{t}\boldsymbol{k}_{i,j}^{t\mathrm{T}})}{\sqrt{d}\sum_{j}\exp(\boldsymbol{q}_{i}^{j}\boldsymbol{k}_{i,j}^{t\mathrm{T}})}, \tag{6.17}$$

$$\boldsymbol{l}_{i}^{t} = \sum_{j}\gamma_{i,j}^{t}\boldsymbol{v}_{i,j}^{t}, \tag{6.18}$$

ここで，$\gamma_{i,j}^{t}$ は時刻 t における他対象 j から予測対象 i への注意重みである．$\hat{\boldsymbol{l}}_{i}^{t}$ は予測対象 i に対する集合値でシーン内の全他対象との空間的関係が含まれている．

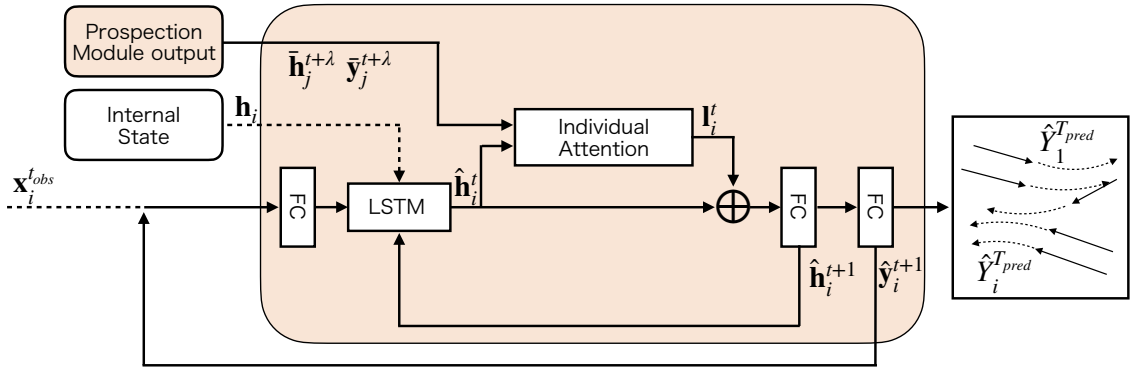


図 6.5: 個人レベルの Attention 機構による Forecasting Module.

個人レベルの Attention 機構 (図 6.5) 及び Group-based Forecasting Module (図 6.4) の内部構成は大きな括りでは Attention 機構が 1 つまたは 2 つかの違いである．一般的なモデルとして考えると，後者の方が Attention 機構にかかる計算コストが大きいように思える．個人レベルの Attention 機構にかかる計算コストは式 (6.19) で計算される．

$$MSA = 3NC^2 + 2N^2C. \tag{6.19}$$

ここで，N はシーン内の歩行者数，C は次元数を表す．右式の第 1 項は Query，Key 及び Value の全結合層にかかる計算コスト，第 2 項は Query と Key の内積にかかる計算コストを表す．ここでは全ての全結合層が 1 層の場合としている．例えば，$N = 8, C = 32$ とした場合，MSA にかかる計算コストは式 (6.19) より，

$$\begin{aligned} MSA &= 3 \times 8 \times 32^2 + 2 \times 8^2 \times 32, \\ &= 24,576 + 4,096, \\ &= 28,672, \end{aligned} \tag{6.20}$$

となる．

Group-based Forecasting Module のグループ間の Attention 機構にかかる計算コストは式 (6.21) のように計算される．

$$MSA(BG) = 3BC^2 + 2B^2C. \tag{6.21}$$

ここで，$B = n(G^t)$ は時刻 t におけるグループの数を示す．グループ内の Attention 機構にかかる計算コストは式 (6.22) のように計算される．

$$MSA(WG) = 3WC^2 + 2W^2C. \tag{6.22}$$

ここで，$W = n(J^t), J = \{1, 2, \ldots, j\}$ は時刻 t における予測対象自身を含む同じグループ内の対象の数を示す．例えば，$N = 8, C = 32$ の条件下で形成されるグループの最大数はシーンの対象全てがグループとして形成されない場合に $B = 8$ となるが，グループ間の Attention 機構では予測対象自身のグループは考慮しないため $B = 7$ となる．一方，最小数は対象全てがグループとして形成された場合に $B = 1$ となる．この $B = 1$ の条件下では $W = 8$ となり，個人レベルの Attention 機構の計算コスト (式 (6.20)) と変わらない．$B \geq 2$ で最大の計算コストとなるのは $B = 2, W = 7$ 及び $B = 8, W = 1$ の場合である．前者は 8 人いる中で 7 名と 1 名の合計 2 グループが形成され，後者は全員がグループとして形成されない場合である．前者では，7 名の中から任意の対象に対する計算コストが式 (6.21) と式 (6.22) より，

$$\begin{aligned} MSA(BG) &= 3 \times (2-1) \times 32^2 + 2 \times (2-1)^2 \times 32, \\ &= 3,072 + 64, \\ &= 3,136, \end{aligned} \tag{6.23}$$

$$
\begin{aligned}
MSA(WG) &= 3 \times 7 \times 32^2 + 2 \times 7^2 \times 32, \\
&= 21,504 + 3,136, \\
&= 24,640,
\end{aligned} \tag{6.24}
$$

となる．式 (6.23) の $(2-1)$ はグループ間の Attention 機構では予測対象自身のグループを考慮しないためグループ数から 1 を引く．これらを合計すると 27,776 となる．後者は式 (6.23) と式 (6.24) の $(2-1)$ と 7 が $(8-1)$ と 1 に変わるだけで，値は前者と変わらない．そのため，$B \geq 2$ の条件では Group-based Forecasting Module は個人レベルの Attention 機構 (式 (6.20)) より計算効率が良い．

■ Clustering Module

Group-based Forecasting Module で予測した経路を Clustering Module へ逐次入力することで，グループ情報を動的に変化させグループラベルを更新する．ここでは，シーン内で運動傾向が類似している歩行者をグループと定義し，位置，方向及び速度の 3 種類の情報からグループをクラスタリングする．まず，歩行者の位置をクラスタリングするために，DBSCAN アルゴリズム [89] を適用する．DBSCAN の入力として， [130] で定義された人間のインタラクションにおける対人距離に基づいて，ETH/UCY では 1.2 [m] 以下の歩行者のみをクラスタリングする．SDD では，20 [pixel] 以下の歩行者のみをクラスタリングする．しかし，位置情報のみを考慮した場合，シーンの動的特性を考慮できず，反対方向から来た人や異なる速度で歩く人を誤って同じグループとしてクラスタリングする可能性がある．そこで，位置ベースのクラスタリング結果から，歩行者の方向と速度に基づくフィルタリングを行う．具体的には以下の条件でフィルタリングを行う．i) 移動方向のコサイン類似度が 0.8 以上，ii) 歩行速度の差が 0.2 [m/s] 未満．上記の条件を満たさない歩行者はクラスタから除外する．ただし，歩行者はグループベースの予測処理により，全ての歩行者に固有のグループ ID が付与されている．つまり，1 人のグループとして扱われる場合もある．

■ ネットワーク構造

エンコーダの LSTM の hidden state $\boldsymbol{h}^{enc}$ とデコーダの LSTM の hidden state ($\bar{\boldsymbol{h}}$, $\hat{\boldsymbol{h}}$) の次元数を 64 と設定した．エンコーダとデコーダの全結合層に入力される経路は 64 次元の特徴として埋め込まれる．Discriminator も 64 次元の全結合層と LSTM で設定した．Internal State $\boldsymbol{h}$ は 2 つの全結合層 [96 $\times$ 128 $\times$ 64] を介して計算される．グループ間の Attention 機構の空間的な関係 $\boldsymbol{r}$ は 1 層の全結合層を介して 64 次元として設定した．Query $\boldsymbol{q}$ の次元数は 2 つの全結合層 [64 $\times$ 128 $\times$ 64] を介して 64 に設定した．Key $\boldsymbol{k}$ と Value $\boldsymbol{v}$ の次元数は 2 つの全結合層 [128 $\times$ 128 $\times$ 64] を介して 64 に設定した．グループ内の Attention では，グループ間の Attention と同様である．具体的には，グループ内の Attention も同じ設定とした．各 Attention 機構の出力 ($\boldsymbol{w}$, $\boldsymbol{b}$) と LSTM デコーダの hidden state $\hat{\boldsymbol{h}}$ を連結した特徴量の次元は，2 つの全結合層 [192 $\times$ 128 $\times$ 64] を介して 64 に設定した．複数経路を予測するためのノイズベクトル $\boldsymbol{z}$ の次元数は 32 とした．

Discriminator を含むモデルは，初期学習率 0.0001 で最適化手法に Adam [118] で学習した．全ての予測モデルはバッチサイズ 64 で 300 エポック学習させた．経験的に Prospection Module の λ を 3 で設定し，このパラメータの効果を Ablation study で検証した．

6.2 評価実験

3 つの公開されているデータセットを用いて，提案手法とインタラクションを考慮した従来手法と比較実験を行う．また，Ablation study で提案手法の有効性を確認する．

6.2.1 実験条件

■ データセット

データセットには，ETH [19], UCY [18] 及び SDD [20] を用いる．ETH と UCY は，俯瞰視点映像で撮影されており，世界座標系で 0.4 秒毎にアノテーションされている．ETH データセットは ETH と HOTEL の 2 つのシーンがあり，UCY データセットは ZARA01，ZARA02，UCY の 3 つのシーンがある．これら 5 つのシーンをベースとして，提案手法は先行研究 [1, 9, 57] と同様に leave-one-out アプローチを用いる．具体的には 4 つのシーンでモデルを学習し，残りの 1 つのシーンで評価する．先行研究同様，観測時刻を 3.2 秒 (8 ステップ) の経路をネットワークへ入力し，その後の予測時刻 4.8 秒 (12 ステップ) の経路を予測する．5.3 節で述べたように，ETH シーンのみ歩行者は縦に移動することが多く，他のシーンは歩行者が横に移動している．一般的に経路予測は leave-one-out アプローチで学習と評価を行うため，縦移動が多い ETH シーンで適切な評価を行うことは困難である．実際，多くの先行研究 [3, 9, 10, 66] では，ETH シーンの予測誤差が他のシーンよりも大きい．そのため，5 章と同様に ETH の (x, y) 座標を転置した C-ETH を定量的評価に用いることで，適切な評価を行う．

SDD は，スタンフォード大学構内をドローンで撮影したピクセル座標のデータセットで，bookstore、coupa、deathCircle などの 8 種類のシーンで構成されている．8 つのシーンをベースに leave-one-out アプローチを行う．SDD には，自転車，歩行者，カート，車，バス及びスケートボーダーの 6 クラスの対象の移動経路がアノテーションされている．本実験では，歩行者のみを対象とし，12 フレーム毎の経路の座標情報を実験に用いた．SDD は 30fps で撮影されているため，各タイムステップは約 0.4 [s] に相当する．ETH/UCY と同様に，観測 3.2 秒 (8 ステップ) 間の経路をネットワークへ入力し，その後の予測時刻 4.8 秒 (12 ステップ) の経路を予測する．

■ 評価指標

モデルの性能を評価するために，3 つの評価指標を用いた．まず，Average Displacement Error (ADE) は全予測時刻における真値と予測経路間のユークリッド距離誤差の平均である．次に，Final Displacement Error (FDE) は最終予測時刻における真値と予測経路間のユークリッド距離誤差である．これら

の計算式は式 (2.20)，式 (2.21) で表される．また，複数経路を予測する手法では，2.4.2 節の Minimum Displacement Error [3] の mADE と mFDE を用いる．mADE と mFDE の計算式は式 (2.22)，式 (2.23) で表される．

最後に， [8] のように Prediction collision を評価する．この指標は，シーン内で歩行者同士が衝突する割合の衝突率から評価される．衝突率は，2 人の歩行者間のユークリッド距離が閾値より低い場合に衝突が発生したと設定する．ETH/UCY では 0.1 [m]，SDD では 10 [pixel] とした．算出式は式 (2.24) である．複数経路を予測する手法では，式 (2.25) のように衝突率を求める．

■ ベースモデル

比較手法として以下のベースライン手法を用いる．Vanilla-LSTM は入力層-中間層-出力層の 3 層で構成されるベースモデルである．RED [126] は単純な LSTM のエンコーダ/デコーダモデルである．これら 2 つは歩行者間のインタラクションを考慮しない予測手法である．Group-LSTM [56] は，予測対象とは異なる方向を持つ歩行者集団をプーリングする予測手法である．Group-LSTM はコードが公開されていないため，Social-LSTM [1] の S-Pooling を修正したものを利用した．その他の予測手法については，コードが公開されている予測手法を利用した．これらの予測手法のパラメータは元論文に従って設定し，ETH/UCY 及び SDD で学習と評価を行った．

■ ベースモデルの問題点

5.1 節で述べたように，経路予測の評価は論文間で一貫性がなく，報告された性能を公平に比較できないことがコミュニティで指摘されている [10]．例えば，ADE と FDE は絶対値で評価しなければならないのに対し，PECNet [12] は出発点からの相対値で評価している．このため，上記ベースラインを再実装し，同じ条件で学習と評価を行った．

6.2.2 ETH/UCY における実験結果

表 6.2 に ETH/UCY データセットにおける ADE と FDE，表 6.3 に Prediction Collision の定量的評価結果を示す．評価は，Single model と 20 outputs に分ける．Single model とは，評価時に予測経路を 1 つ出力する方法である．表 6.2 の上段は Single model の出力の誤差を示す．20 outputs とは，先行研究 [3] のように，評価時に複数の予測経路を出力し，その中から最良の予測経路を 1 つ選択する方法である．これは，順伝播処理を繰り返すことで複数の経路を予測する．従って，1 回の順伝播処理で複数の経路を直接予測する [131]，すなわち N 個のサンプルの将来の (x, y) 座標を出力する方法とは異なる．表 6.2 の下段は 20 個の予測経路の中で最も性能が良いものを示す．全体として，提案手法は従来手法を大きく上回っていることがわかる．さらに，C-ETH のほとんどのモデルの予測誤差は ETH の予測誤差より低減している．これにより，ほとんどの予測モデルが水平方向の動きに対して頑健であることから，データの偏りが ETH シーンの予測誤差を増加させる要因であることを示

表 6.2: ETH/UCY における提案手法と従来手法の ADE/FDE の定量的評価結果．単位は [m] で，値が低い程性能が良いことを示す．Single model は 1 つの経路を予測するモデル，20 outputs は複数の経路を予測するモデルである．

	Method	Scene						
		C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	AVG
Single model	LSTM	0.57 / 1.20	0.71 / 1.36	0.21 / 0.38	0.63 / 1.37	0.59 / 1.26	0.42 / 0.84	0.52 / 1.08
	RED [126]	0.56 / 1.24	0.67 / 1.36	0.21 / 0.40	0.61 / 1.34	0.52 / 1.19	0.42 / 0.89	0.50 / 1.07
	Social LSTM [1]	0.93 / 1.62	1.19 / 2.27	0.49 / 0.97	0.98 / 1.94	1.02 / 1.82	0.71 / 1.38	0.89 / 1.67
	Group-LSTM [56]	0.83 / 1.65	1.03 / 2.12	0.48 / 0.94	0.87 / 1.73	0.94 / 1.72	0.69 / 1.25	0.81 / 1.57
	SR-LSTM [6]	0.59 / 1.32	0.66 / 1.42	0.53 / 1.35	0.66 / 1.45	0.65 / 1.63	0.45 / 1.00	0.59 / 1.36
	Social-GAN [3]	0.62 / 1.35	0.73 / 1.59	0.48 / 1.07	0.78 / 1.62	0.62 / 1.36	0.50 / 1.10	0.62 / 1.35
	STGAT [9]	0.60 / 1.29	0.61 / 1.24	0.23 / 0.45	0.65 / 1.40	0.52 / 1.11	0.42 / 0.90	0.51 / 1.07
	Trajectron [10]	0.58 / 1.27	0.79 / 1.80	0.25 / 0.46	**0.58** / 1.28	**0.40** / **0.89**	0.38 / **0.76**	0.50 / 1.08
	Social STGCNN [37]	0.66 / 1.18	1.30 / 2.30	0.29 / 0.48	1.03 / 1.74	0.83 / 1.53	0.66 / 1.22	0.80 / 1.41
	PECNet [12]	0.90 / 1.70	0.72 / 1.41	0.20 / 0.44	0.66 / 1.33	0.49 / 1.05	0.40 / 0.87	0.56 / 1.13
	SGCN [66]	0.59 / 1.20	0.70 / 1.33	0.25 / 0.49	0.67 / 1.38	0.51 / 1.05	0.43 / 0.92	0.53 / 1.06
	Ours-single-model	**0.50** / **1.08**	**0.55** / **1.08**	**0.17** / **0.33**	0.59 / **1.27**	0.47 / 1.02	**0.36** / 0.79	**0.44** / **0.93**
20 outputs	Social-GAN [3]	0.49 / 1.05	0.53 / 1.03	0.35 / 0.77	0.79 / 1.63	0.47 / 1.01	0.44 / 0.94	0.51 / 1.07
	STGAT [9]	0.44 / **0.77**	**0.48** / 0.94	0.16 / 0.29	0.57 / 1.23	**0.35** / **0.74**	0.31 / 0.67	**0.39** / **0.77**
	Trajectron [10]	0.52 / 1.14	0.68 / 1.34	0.19 / 0.40	**0.55** / 1.24	**0.35** / 0.82	**0.27** / **0.65**	0.43 / **0.77**
	Social-STGCNN [37]	0.51 / 1.10	0.90 / 1.64	0.20 / 0.42	0.87 / 1.47	0.58 / 1.02	0.52 / 0.94	0.60 / 1.10
	PECNet [12]	0.58 / 0.99	0.64 / 1.07	**0.14** / **0.21**	0.61 / **1.19**	0.39 / 0.78	0.36 / 0.67	0.45 / 0.82
	SGCN [66]	0.45 / 0.91	0.51 / 0.89	0.17 / 0.27	0.60 / 1.21	0.39 / 0.76	0.35 / 0.71	0.41 / 0.79
	Ours	**0.43** / 0.93	0.49 / **0.94**	**0.14** / 0.27	**0.55** / **1.19**	0.40 / 0.90	0.31 / 0.71	**0.39** / 0.82

唆している．

■ Single model

Single model の定量的評価結果を表 6.2 の上段に示す．Our-single-model はノイズベクトルを連結せずに，1 つの経路を予測している．提案手法は，ほとんどのシーンで従来手法より予測誤差を大きく低減している．特に，提案手法は同じグループレベルのインタラクションを捉える Group-LSTM より低い予測誤差である．Group-LSTM は，同じグループ内の歩行者間のインタラクションを無視しており，他グループとの社会的なインタラクションを 1 つのグループとして捉える．その結果，Group-LSTM は図 6.2(b) に示すように正確な経路予測に失敗している．一方で，提案手法では，予測対象が属するグループ内の他対象とのインタラクション及び，他グループとのインタラクションの両方をそれぞれの Attention 機構で捉えることで，図 6.2(c) のように正確な経路を予測できる．

■ 20 Outputs

20 Outputs の定量的評価結果を表 6.2 の下段に示す．提案手法は全てのシーンで ADE スコアが最良で，いくつかのシーンで FDE スコアが最良である．しかし，シーンで平均した FDE スコアは Trajectron，STGAT，SGCN と比較して高い．これは，表 6.3 に示すように，提案手法が将来的に他

表 6.3: ETH/UCY における提案手法と従来手法の Prediction Collision の結果．単位は [%] で，値が低い程性能が良いことを示す．歩行者間のユークリッド距離が閾値 0.1 [m] 以下であれば衝突が発生したとみなす．

	Method	Scene						
		C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	AVG
Single model	LSTM	0.81	0.81	0.44	0.25	0.24	0.28	0.47
	RED [126]	0.81	2.41	0.75	0.21	**0.15**	0.29	0.77
	Social-LSTM [1]	1.61	4.03	1.34	0.38	0.76	0.69	1.47
	Group-LSTM [56]	1.61	2.41	0.97	0.38	0.64	0.67	1.11
	SR-LSTM [6]	0.81	0.81	**0.22**	0.24	0.20	0.26	0.47
	Social-GAN [3]	0.81	0.81	**0.22**	0.38	0.34	0.44	0.50
	STGAT [9]	**0.0**	0.81	0.44	**0.20**	0.19	0.26	0.32
	Trajectron [10]	0.81	0.81	0.60	**0.20**	0.24	**0.20**	0.48
	Social-STGCNN [37]	0.81	2.42	2.23	0.51	0.71	0.67	1.23
	PECNet [12]	0.81	4.03	0.59	0.28	0.17	0.44	1.05
	SGCN [66]	**0.0**	3.22	1.56	0.34	0.48	0.54	1.02
	Ours-single-model	**0.0**	**0.0**	**0.22**	**0.20**	0.17	0.26	**0.14**
20 outputs	Social-GAN [3]	1.61	0.0	0.60	0.23	0.17	0.26	0.48
	STGAT [9]	0.0	0.81	0.44	0.25	0.27	0.24	0.34
	Trajectron [10]	1.61	0.0	0.60	0.18	**0.15**	0.26	0.47
	Social-STGCNN [37]	2.42	2.42	1.72	0.46	0.56	0.80	1.42
	PECNet [12]	0.81	2.42	0.59	0.28	0.20	0.67	0.83
	SGCN [66]	**0.0**	2.42	1.34	0.28	0.34	0.26	0.77
	Ours	**0.0**	**0.0**	**0.11**	**0.16**	0.17	**0.22**	**0.11**

グループとの衝突を避けるための経路を予測した可能性があるためである．

■ Prediction Collision

表 6.3 より，提案手法は Single model と 20 outputs 両方で，ほぼ全てのシーンで衝突率を抑え最良のスコアである．これは，Group-based Forecasting Module により，グループ内の他対象や将来の他グループとの社会的インタラクションを捉えた機構の導入によるものである．これにより，将来の経路情報を用いて社会的インタラクションを考慮することがシーン内の歩行者との衝突回避に有効であることが示された．その結果，表 6.2 のように，ほとんどのシーンで予測誤差も低減した．

6.2.3 Ablation study

提案手法の各モジュールの効果を確認するために Ablation study を行った．提案手法の各モジュールについて，ADE と FDE で定量的評価した結果を表 6.4 に示す．値はメートル単位である．BG と WG はそれぞれ，Group-based Forecasting Module におけるグループ間及びグループ内の Attention 機

表 6.4: 各モジュールの Ablation study．IA: 個人レベルの Attention 機構，WG: グループ内の Attention 機構，BG: グループ間の Attention 機構，λ: Prospection Module における未来のインタラクションを制御するパラメータ．単位は [m] であり，ADE/FDE スコアで評価している．

Variant	Condition				Scene						
ID	IA	WG	BG	λ	C-ETH	ETH	HOTEL	UCY	ZARA01	ZARA02	AVG
1	✓	-	-	0	0.58 / 1.22	0.63 / 1.27	0.17 / 0.33	0.63 / 1.37	0.51 / 1.07	0.40 / 0.87	0.49 / 1.02
2	-	✓	-	0	0.55 / 1.15	0.58 / 1.09	0.17 / 0.32	0.61 / 1.32	0.46 / 1.04	0.39 / 0.86	0.46 / 0.96
3	-	-	✓	0	0.52 / 1.20	0.55 / 1.12	0.16 / 0.31	0.60 / 1.31	0.49 / 1.08	0.37 / 0.84	0.45 / 0.98
4	-	✓	✓	0	0.53 / 1.15	0.56 / 1.06	0.16 / 0.30	0.60 / 1.30	0.46 / 1.00	0.36 / 0.82	0.45 / 0.94
5	✓	-	-	1	0.53 / 1.17	0.56 / 1.10	0.16 / 0.29	0.59 / 1.27	0.44 / 0.98	0.36 / 0.82	0.44 / 0.94
6	-	-	✓	1	0.51 / 1.07	0.54 / 1.09	0.20 / 0.39	0.59 / 1.29	0.45 / 1.07	0.38 / 0.84	0.45 / 0.96
7	-	✓	✓	1	0.49 / 1.07	0.50 / 0.99	0.15 / 0.29	0.57 / 1.24	0.42 / 0.92	0.35 / 0.79	0.41 / 0.88
8	✓	-	-	2	0.57 / 1.20	0.56 / 1.11	0.17 / 0.32	0.58 / 1.26	0.44 / 0.98	0.35 / 0.76	0.46 / 0.94
9	-	-	✓	2	0.48 / 1.03	0.53 / 1.08	0.15 / 0.28	0.55 / 1.20	0.41 / 0.91	0.34 / 0.75	0.41 / 0.88
10	-	✓	✓	2	0.50 / 1.08	0.51 / 0.98	**0.14** / **0.26**	**0.54** / **1.17**	**0.38** / 0.85	0.33 / 0.73	0.40 / 0.85
11	✓	-	-	3	0.49 / 1.01	0.54 / 1.07	0.18 / 0.33	0.57 / 1.24	0.43 / 0.99	0.33 / 0.74	0.42 / 0.90
12	-	-	✓	3	0.48 / 1.05	0.52 / 1.04	0.15 / 0.29	0.56 / 1.20	**0.38** / **0.84**	0.32 / 0.72	0.40 / 0.86
13	-	✓	✓	3	**0.43** / **0.93**	**0.49** / 0.94	**0.14** / 0.27	0.55 / 1.19	0.40 / 0.90	**0.31** / **0.71**	**0.39** / **0.82**
14	✓	-	-	4	0.49 / 1.02	0.52 / 1.04	0.15 / 0.28	0.57 / 1.24	0.41 / 0.89	0.34 / 0.76	0.41 / 0.87
15	-	-	✓	4	0.44 / 0.92	0.54 / 1.03	0.16 / 0.32	0.57 / 1.24	0.40 / 0.87	0.33 / **0.71**	0.41 / 0.85
16	-	✓	✓	4	0.47 / 0.99	**0.46** / **0.91**	0.17 / 0.32	0.58 / 1.26	0.40 / 0.88	0.34 / 0.74	0.40 / 0.85
17	✓	-	-	5	0.50 / 1.04	0.55 / 1.10	0.16 / 0.29	0.57 / 1.23	0.43 / 0.93	0.33 / 0.73	0.42 / 0.89
18	-	-	✓	5	0.47 / 1.00	0.52 / 1.04	0.17 / 0.32	0.59 / 1.29	0.39 / 0.85	0.33 / 0.72	0.41 / 0.87
19	-	✓	✓	5	0.50 / 1.07	0.48 / 0.93	0.18 / 0.30	0.60 / 1.28	0.39 / 0.85	0.34 / 0.75	0.42 / 0.86
20	✓	-	-	6	0.50 / 1.07	0.57 / 1.13	0.19 / 0.37	0.62 / 1.30	0.42 / 0.88	0.38 / 0.81	0.45 / 0.93
21	-	-	✓	6	0.49 / 1.00	0.53 / 1.03	0.19 / 0.35	0.62 / 1.29	0.41 / 0.88	0.37 / 0.79	0.44 / 0.89
22	-	✓	✓	6	0.48 / 0.98	0.52 / 1.03	0.18 / 0.35	0.61 / 1.28	0.41 / 0.87	0.35 / 0.76	0.43 / 0.88

構を表す．IA は，各グループの Attention 機構の有効性を調査するための個人レベルの Attention 機構を表す．表 6.4 の個人レベル (IA) とグループレベル (BG と WG) のインタラクションモデルを比較すると，個人レベルのインタラクションのモデル化よりグループレベルのインタラクションのモデル化が予測誤差を低減している．提案手法では，グループ間の Attention とグループ内の Attention を両方組み合わせることで，さらに予測誤差を低減する．

次に，将来のインタラクションを捉えるために，グループ間の Attention 機構で用いられる Prospection Module から伝播される予想経路が与える影響について調査した．表 6.4 の 5 列目は，Prospection Module の予想経路における未来のシフトされるフレーム λ を示す．なお，$\lambda = 0$ は，Prospection Module の予想経路を用いない．このモデルでは，グループ間の Attention 機構に現時刻の経路情報を用いている．Variant ID 13 は，6.1.2 節で述べた提案手法の λ を設定したもので，このモデルの性能が最良である．表 6.4 より，将来で衝突するリスクの高い他グループとの衝突を回避するために，Prospection Modue から伝播される予想経路を用いることが人間の社会的インタラクションに有効であることが

表 6.5: パーソナルスペースを変化させた ADE/FDE の精度比較．全てのモデルはグループ間及びグループ内の Attention 機構を導入しており，$\lambda = 3$ とした場合の結果を示す．

	Personal space [m]			
Scene	0.45	1.2	3.6	7.6
C-ETH	0.53/1.08	**0.43/0.93**	0.44/**0.93**	0.47/1.00
ETH	0.59/1.15	**0.49/0.94**	0.50/0.96	0.51/1.03
HOTEL	0.17/0.34	**0.14/0.27**	0.16/0.32	0.19/0.38
UCY	0.56/1.22	**0.55/1.19**	0.64/1.37	0.66/1.40
ZARA01	0.42/0.90	**0.40**/0.90	0.41/**0.89**	0.43/0.92
ZARA02	0.38/0.84	**0.31/0.71**	0.34/0.76	0.36/0.80
AVG	0.44/0.92	**0.39/0.82**	0.42/0.87	0.44/0.92

表 6.6: 各損失関数の有効性．Adversarial loss と L2 loss の両方を導入することで，予測性能が向上する．

	Loss function		
Scene	w/ Adversarial	w/ L2	w/ Adv. and L2
C-ETH	1.76/3.04	0.45/0.95	**0.43/0.93**
ETH	1.46/2.50	**0.47/0.88**	0.49/0.94
HOTEL	1.04/1.98	0.17/0.30	**0.14/0.27**
UCY	1.73/3.09	0.60/1.28	**0.55/1.19**
ZARA01	3.11/5.74	0.43/0.94	**0.40/0.90**
ZARA02	2.06/3.79	0.36/0.75	**0.31/0.71**
AVG	1.34/3.36	0.41/0.85	**0.39/0.82**

示された．興味深いことに，λ が大きくなるにつれ予測誤差が徐々に大きくなっている．これは，将来の不確実な予測結果に依存していることを意味する．

表 6.5 は，パーソナルスペースの設定，つまり DBSCAN アルゴリズムにおける距離の閾値を変更した場合の精度比較を示す．パーソナルスペースが小さいほど，5 章で説明したように歩行者密度の高い混雑したシーンである UCY において，性能が向上している．一方，あまり混雑していない ETH や ZARA01 では，パーソナルスペースが大きい程，性能が向上している．このように，シーンによって適切なパーソナルスペースは予測性能に影響を与える可能性があり，性能を向上させるためにはシーンの特徴に応じてパラメータを設定する必要がある．

表 6.6 は，各損失関数の有効性を調査した定量的評価結果である．Adversarial loss がない場合は，式 (6.12) の第 2 項において L2 loss のみを用いている．なお，全ての結果は表 6.4 の Variant ID 13 のモデルを用いている．表 6.6 の 2 列目は，全てのシーンにおいて Adversarial loss のみを用いた場合の ADE/FDE スコアを示す．Adversarial loss は予測経路が本物か偽物かを学習するため，真値と予測経路間の距離を最小化することができない．表 6.6 の 3 列目は，全てのシーンにおいて L2 loss を用いた場合の ADE/FDE スコアを示す．L2 loss は 2 列目の Adversarial loss のみより ADE/FDE スコアが良いことを示す．これは，L2 loss が真値と予測経路間の距離を最小化するように学習するためであ

表 6.7: SDD における提案手法と従来手法の ADE/FDE の定量的評価結果．単位は [pixel] で，値が低い程性能が良いことを示す．

	Method	Scene								
		bookstore	coupa	deathCircle	gates	hyang	little	nexus	quad	AVG
Single model	LSTM	9.64 / 20.9	10.4 / 22.3	9.24 / 19.6	7.80 / 16.7	10.3 / 21.8	12.3 / 25.8	8.97 / 19.2	8.52 / 18.7	9.65 / 20.6
	RED [126]	6.66 / 13.5	7.96 / 16.2	7.42 / 14.8	**5.80** / 11.7	9.13 / 17.4	10.9 / 22.9	7.65 / 14.9	5.57 / 9.74	7.64 / 15.1
	Social-LSTM [1]	22.8 / 47.2	24.1 / 49.3	29.5 / 61.6	24.5 / 49.3	35.4 / 75.9	24.3 / 50.5	22.9 / 45.7	24.1 / 46.6	26.0 / 53.3
	Group-LSTM [56]	19.6 / 39.9	22.8 / 43.9	24.4 / 50.5	20.9 / 42.8	28.1 / 44.0	22.2 / 46.1	19.9 / 40.2	22.1 / 42.9	22.5 / 43.8
	SR-LSTM [6]	8.11 / 16.2	7.69 / 15.9	7.61 / 14.7	6.18 / **11.2**	9.22 / 18.3	11.8 / 24.1	6.80 / 13.1	5.31 / 8.78	7.84 / 15.3
	Social-GAN [3]	18.4 / 37.2	19.1 / 38.4	18.1 / 36.5	18.1 / 36.5	19.4 / 39.1	20.8 / 42.5	18.6 / 37.4	21.1 / 41.4	19.2 / 38.6
	STGAT [9]	7.58 / 14.6	9.00 / 17.4	7.57 / 14.4	6.33 / 11.7	9.17 / 17.9	10.9 / 22.8	7.37 / 14.0	**4.83 / 7.95**	7.84 / 15.1
	Trajectron [10]	8.79 / 19.2	**7.24** / 15.5	14.3 / 33.0	6.29 / 13.0	9.55 / 21.6	12.4 / 29.0	**6.02 / 12.8**	6.80 / 15.1	8.92 / 19.9
	Social-STGCNN [37]	18.9 / 38.6	21.9 / 42.0	25.7 / 53.2	19.4 / 40.0	26.5 / 41.9	25.1 / 52.0	20.4 / 42.6	21.8 / 41.2	22.4 / 43.9
	PECNet [12]	9.01 / 21.2	8.91 / 16.9	8.92 / 17.3	6.52 / 13.8	9.88 / **14.1**	11.2 / 23.2	6.91 / 13.8	5.99 / 16.2	10.1 / 17.1
	SGCN [66]	9.45 / 18.6	7.37 / **13.8**	16.8 / 34.5	9.11 / 18.6	11.8 / 23.7	16.2 / 34.2	8.02 / 15.3	10.3 / 17.6	11.1 / 22.0
	Ours-single-model	**6.41 / 12.4**	7.45 / 14.9	**7.11 / 13.7**	6.10 / **11.2**	**8.54** / 16.8	**10.2 / 21.2**	6.88 / 13.9	5.07 / 8.77	**7.22 / 14.1**
20 outputs	Social-GAN [3]	5.03 / 8.96	5.39 / 9.57	5.20 / 9.13	4.66 / 8.25	5.74 / **10.2**	6.51 / 11.9	5.23 / 9.11	4.08 / 7.15	5.23 / 9.28
	STGAT [9]	4.09 / 7.57	**4.83 / 9.05**	4.59 / 8.43	2.86 / **4.69**	**5.37 / 10.2**	6.25 / 12.1	**4.20 / 7.58**	2.34 / 3.62	4.32 / 7.91
	Trajectron [10]	4.51 / 7.99	5.69 / 9.72	5.12 / 8.87	4.43 / 8.06	5.72 / 10.3	6.38 / **10.8**	5.55 / 9.43	2.26 / 4.61	4.96 / 8.72
	Social-STGCNN [37]	7.13 / 14.2	7.41 / 16.1	7.26 / 14.0	5.94 / 12.8	10.5 / 21.3	8.99 / 15.3	6.34 / 13.2	5.55 / 8.90	7.39 / 14.5
	PECNet [12]	4.33 / 7.52	5.31 / 9.42	5.11 / 8.76	3.99 / 6.83	6.10 / 11.6	6.67 / 11.0	4.85 / 8.01	3.82 / 4.43	5.02 / 8.45
	SGCN [66]	7.45 / 13.8	5.37 / 9.28	14.2 / 28.3	6.22 / 11.1	9.82 / 19.2	13.6 / 27.6	6.08 / 11.0	5.57 / 7.55	8.53 / 16.0
	Ours	**3.96 / 7.34**	4.88 / 9.11	**4.15 / 8.29**	**2.84** / 4.74	5.38 / 10.7	**6.14** / 11.8	4.28 / 7.71	**2.20 / 3.48**	**4.23 / 7.90**

る．表 6.6 の最後の列に示すように，Adversarial loss と L2 loss の両方を導入することで，ADE/FDE は最良のスコアとなる．これは，L2 loss が予測経路が真値からどれだけ離れているかを制御するのに対し，Adversarial loss が予測経路が本物か偽物かを敵対的に学習するためである．

6.2.4 SDD における実験結果

表 6.7 に SDD における ADE/FDE スコアを示す．提案手法は表 6.4 の Variant ID 13 のモデルを用いている．提案手法は，Single model，20 outputs でほぼ全てのシーンで従来手法より性能が向上している．これにより，複雑な歩行者間の社会的インタラクションを捉える経路予測において，2 つの Attention 機構の導入が効果的であることが示される．また，インタラクションを考慮しない RED が，Single model で提案手法の次に性能が良い．これは，SDD が ETH/UCY データセットより高所で撮影されることで歩行者の動きが線形的になり，他手法のインタラクションのモデル化が上手く機能しなかったことが原因だと考えられる．

表 6.8 に SDD における提案手法と従来手法の Prediction Collision の結果を示す．提案手法は，Single model と 20 output 共にほぼ全てのシーンで最良のスコアである．これらの結果より，提案手法の有効性が確認された．

表 6.8: SDD における提案手法と従来手法の Prediction Collision の結果．単位は [%] で，値が低い程性能が良いことを示す．歩行者間のユークリッド距離が閾値 10 [pixel] 以下であれば衝突が発生したとみなす．

	Method	Scene								
		bookstore	coupa	deathCircle	gates	hyang	little	nexus	quad	AVG
Single model	LSTM	**3.62**	5.95	39.76	4.48	10.28	3.48	23.21	**0.0**	11.35
	RED [126]	**3.62**	6.90	41.02	3.96	10.35	12.43	23.96	**0.0**	11.53
	Social-LSTM [1]	9.57	14.67	50.75	6.00	16.06	7.20	28.26	**0.0**	16.56
	Group-LSTM [56]	8.18	12.75	44.24	5.84	14.73	8.87	25.98	**0.0**	15.07
	SR-LSTM [6]	4.11	5.96	42.88	3.96	11.62	3.63	22.18	**0.0**	11.79
	Social-GAN [3]	3.86	6.12	40.50	4.09	10.96	4.11	24.09	**0.0**	11.72
	STGAT [9]	**3.62**	5.22	39.76	3.66	10.28	3.66	20.09	**0.0**	10.79
	Trajectron [10]	4.85	9.65	48.87	4.70	12.06	3.32	26.38	**0.0**	13.73
	Social-STGCNN [37]	9.71	15.90	49.62	8.96	17.25	7.33	**20.04**	**0.0**	16.10
	PECNet [12]	4.98	10.43	44.09	4.96	11.12	4.58	24.69	**0.0**	13.10
	SGCN [66]	5.32	8.12	45.26	4.22	11.56	5.61	27.12	**0.0**	13.40
	Ours-single-model	3.92	**4.87**	**38.86**	**2.78**	**10.12**	**3.22**	20.46	**0.0**	**10.53**
20 outputs	Social-GAN [3]	3.62	5.66	38.26	3.96	10.88	3.96	22.98	**0.0**	11.17
	STGAT [9]	**2.85**	4.98	36.77	4.57	9.28	3.32	20.09	**0.0**	10.23
	Trajectron [10]	3.62	7.32	44.22	4.70	10.01	3.32	22.98	**0.0**	12.02
	Social-STGCNN [37]	7.32	11.86	43.93	6.88	14.58	5.99	19.89	**0.0**	13.81
	PECNet [12]	4.46	8.55	42.20	4.22	10.96	4.22	22.54	**0.0**	12.14
	SGCN [66]	4.85	7.84	43.10	3.96	10.88	5.20	22.12	**0.0**	12.24
	Ours	3.49	**4.21**	**35.49**	**2.50**	**9.03**	**3.04**	**19.39**	**0.0**	**9.64**

6.2.5 予測結果

次に，各データセットにおける予測結果例と注意重みの可視化，Failure cases を述べる．

■ ETH/UCY における予測結果例

図 6.6 に ETH/UCY における提案手法と従来手法の予測結果例を示す．提案手法は，グループ間とグループ内のインタラクションを捉えることで，より正確な経路を予測できる．図 6.6(a) において，個人レベルのインタラクションに基づく Social-GAN や STGAT では，正確な経路を予測できていない．一方，提案手法はグループレベルのインタラクションにより，同じグループに属する対象同士が類似した経路の予測及び，前方のグループとの衝突を回避する経路を予測している．図 6.6(b) において，Group-LSTM，STGAT，Social-GAN はグループ内のインタラクションや過去のインタラクションを考慮するため，正確な経路を予測できていない．提案手法は，グループ内の Attention 機構の出力を次時刻の LSTM デコーダへ逐次入力されるため，未来の時刻で常にインタラクションを計算する．その結果，予測対象自身と他対象との関連性を捉えつつ正確な経路を予測できる．

歩行者グループのダイナミクスを変化させた提案手法の予測結果例を図 6.7 に示す．図 6.7(c) では，前方のグループとの衝突を避けるために，グループを解散させるような経路を予測した．図 6.7(d) は，赤色と青色の対象が同じ方向に進んでいるのか判断が困難なシーンである．提案手法は，同じ

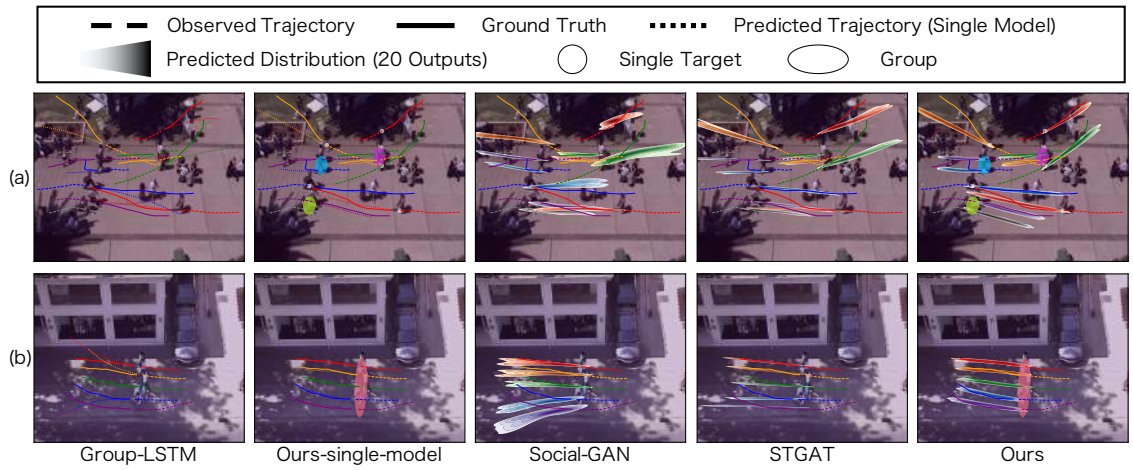


図 6.6: ETH/UCY における予測結果例．Group-LSTM，Ours-single-model は 1 つの経路を予測するモデル，Social-GAN，STGAT 及び Ours は 20 つの予測経路をサンプリングするモデルである．粗い破線は観測経路，実線は真値，細い破線は 1 つの経路を予測するモデルの予測経路，分布は 20 つの予測経路の分布である．提案手法 (Ours-single-model, Ours) は検出したグループをクラスタリングしている．

図 6.7: ETH/UCY における歩行者グループが時間と共に動的に変化する例．$t_{obs}+1$ が予測開始時刻，t_{pred} が最終予測時刻を表す．Group-based Forecasting Module は，グループ情報を時間と共に動的に変化させることで，変化に対応した経路を予測する．

図 6.8: ETH/UCY における Group-based Forecasting Module の各 Attention 機構の注意重みの可視化．(a) (d) は図 6.6，図 6.7 の各図に対応する．色のついた円と矢印はグループラベルを表し，同じグループ内の歩行者は同じ色で可視化される．各円の半径は注目度を表し，円が大きいほど重要度が高い．予測対象は黒い矢印で表す．従って，各結果は予測対象の他対象に対する注目度を表している．

目的地に向かう場合はグループとして，異なる目的地に向かう場合は離れていく経路を複数予測している．

■ ETH/UCY における注意重みの可視化結果例

提案手法の各注意重み，すなわち式 (6.4) と式 (6.9) の可視化例を図 6.8 に示す．色付いた円と矢印はグループラベルを表し，同じグループの対象は同じ色で表している．各円の半径はその注目度，予測対象は黒い矢印で表されている．つまり，これらの結果は予測対象の他対象に対する注目度を表している．図 6.8(a) と (c) はグループ間の Attention 機構の注意重みの可視化例で，予測対象の手前にいる赤色のグループが最も注目度が高い．これは図 6.6(a) と図 6.7(c) のように，将来で衝突する可能性の高いグループを強調することで，提案手法が正確に経路を予測できることを示す．また，グループ内の Attention 機構の注意重みの可視化例では，図 6.8(d) では予測対象自身が強調されるが，図 6.8(d) ではグループ内の上から 1 番目と 4 番目の対象に注目している．これらの結果は，状況に

より同じグループ内の他対象を強調及び自身を強調するなど，提案手法がグループ内の注意重みを柔軟に割り当てることで，グループの特性に応じた経路を予測できることを示している．

■ SDD における予測結果例

SDD における提案手法と従来手法の予測結果例を図 6.9 に示す．全ての予測手法は 20 outputs のモデルの予測結果を可視化している．個人レベルのインタラクションに基づく予測手法は，両シーンにおいて正確な経路の予測ができていない．Social-GAN は，両シーンにおいて正確な経路を予測できず，他対象との衝突が発生している．STGAT は前方のグループと衝突する経路を予測している．一方，グループレベルのインタラクションに基づく提案手法では，同じグループに属する対象同士は類似した経路を予測し，前方のグループとの衝突を回避する経路を予測している．

歩行者グループのダイナミクスを変化させた提案手法の予測結果例を図 6.10 に示す．図 6.10(c), (d)，(e) は図 6.9(b) の各グループに対応する．図 6.10(c) では，周囲の歩行者の影響を受けながらも，グループとして最初から最後まで一貫して同じような経路を予測している．図 6.10(d) では，黄色と紫色の予測対象が時刻 $t_{obs}+1$ でグループとして同じ目的地へ向かっている．それぞれの予測対象は時刻 $t_{obs}+6$ でグループを分散させ，紫色の対象はその地点で停止する経路を予測する．これは，紫色の対象がグループ間の Attention 機構により，黄色の対象と衝突しないように速度を落としているためである．図 6.10(e) は，時刻 $t_{obs}+6$ で赤色，灰色，紫色のグループが橙色と紫色の対象のグループ及び，赤色と灰色の対象のグループの 2 つに分散されるシーンを示している．

■ SDD における注意重みの可視化結果例

各 Attention 機構の注意重みの可視化例を図 6.11 に示す．図 6.11(a)(b) のグループ間の Attention 機構で獲得した注意重みの可視化結果例では，予測対象の前方にいる緑色のグループの注目度が最も高い．また，図 6.11(b) のグループ間の Attention 機構の注意重みは緑色のグループだけでなく，紫色やベージュ色で表現された他対象にも注目していることがわかる．これらは図 6.9(a)(b) のように，将来で衝突するリスクがある他グループや他対象に注目することで，提案手法が正確に経路を予測できることを示す．図 6.11(a) のグループ内の Attention 機構で得られる注意重みの可視化結果例では，予測対象自身が強調されているが，図 6.11(b) ではグループの中央にいる対象の注目度が高い．各対象は図 6.9 に示すように，強調された対象の影響を受けることで，同じグループで類似した経路を予測する．

■ Failure cases

ETH/UCY における誤った予測結果例を図 6.12 に示す．図 6.12(a) は経路予測タスクの典型的な失敗例で，急激な行動変化をする対象の経路予測に失敗した例である．図 6.12(b)(c) は提案手法特有の失敗例である．図 6.12(b) は，前方のグループと衝突を回避するために，予測分布が真値と離れた結

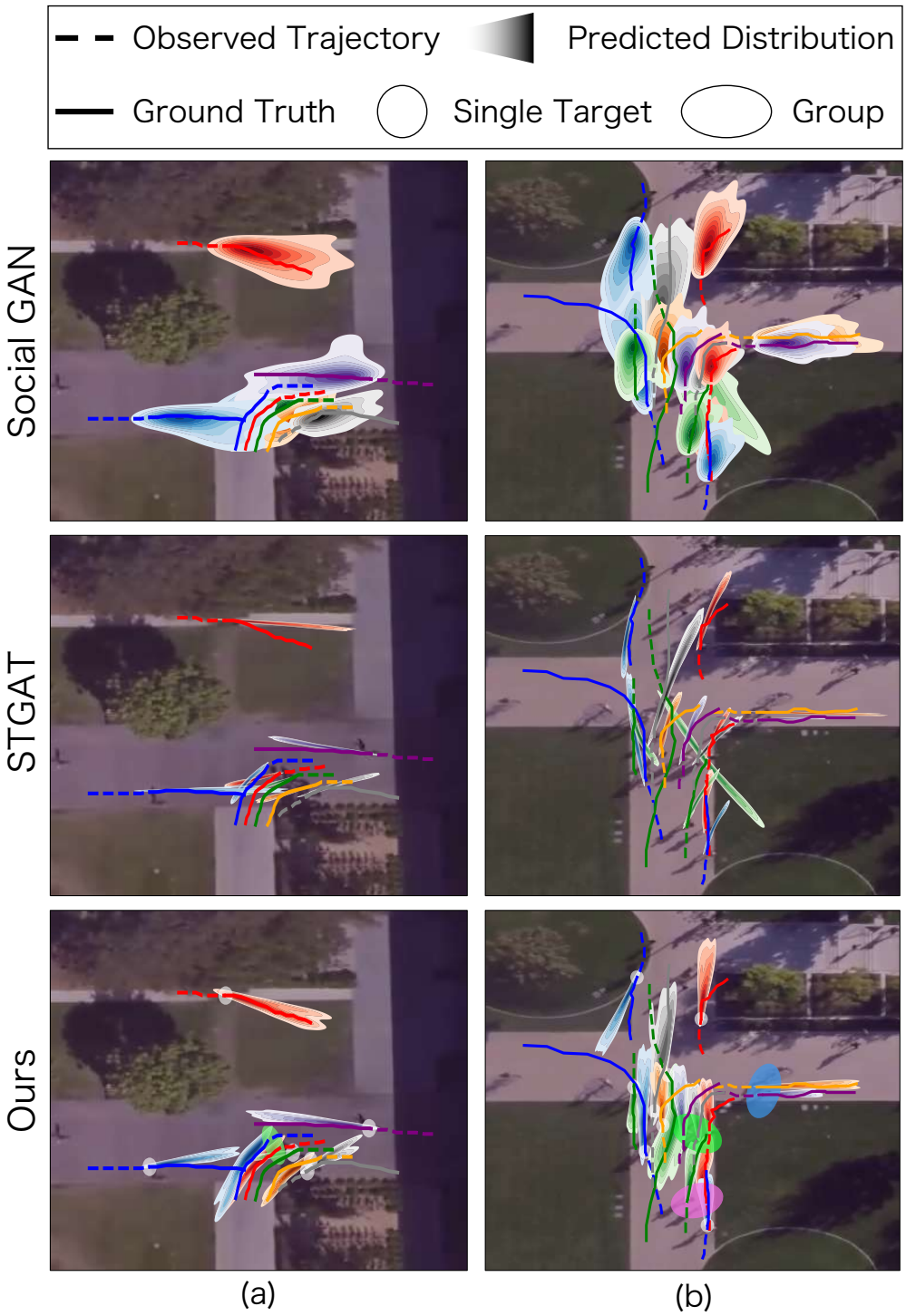


図 6.9: SDD における Social-GAN，STGAT 及び，Ours の予測結果例．粗い破線は観測経路，実線は真値，分布は 20 つの予測経路の分布である．Ours は検出したグループをクラスタリングしている．

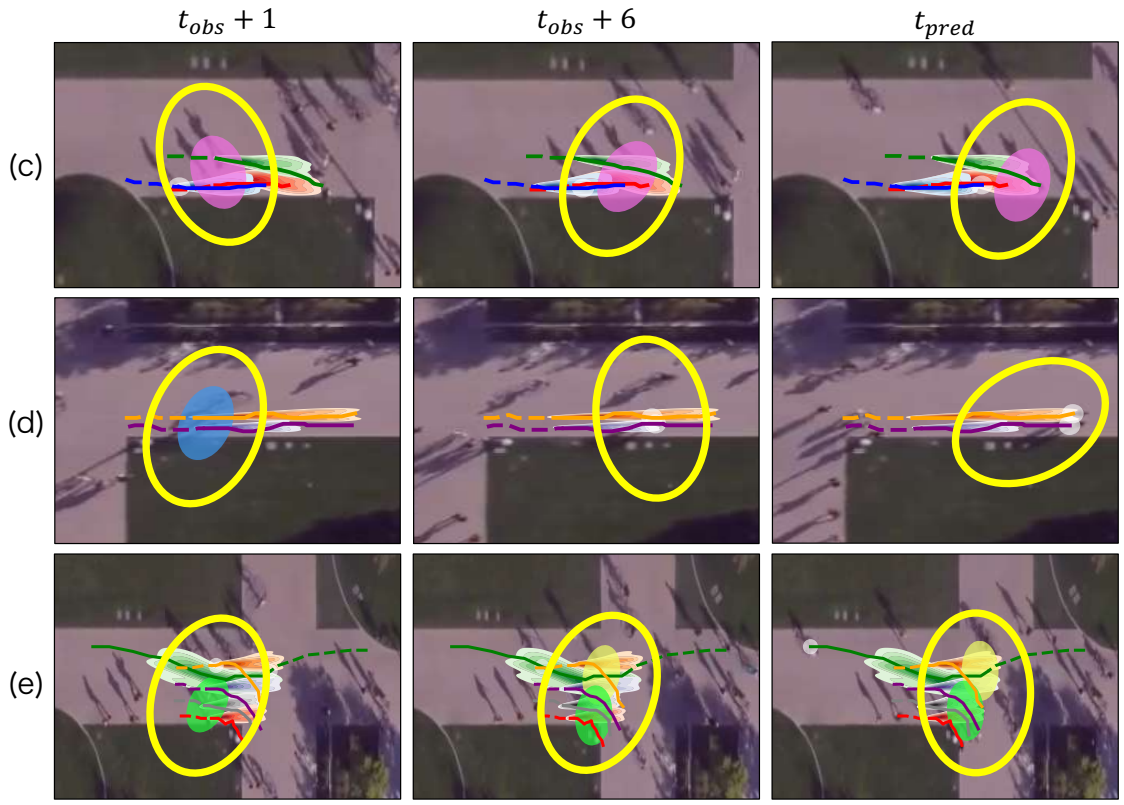


図 6.10: SDD における歩行者グループが動的に変化する例．$t_{obs}+1$ が予測開始時刻，t_{pred} が最終予測時刻を表す．(c)，(d) 及び (e) は図 6.9(b) の各グループに対応する．

果例である．図 6.12(c) は，Prospection Module が同じ上方向への予想経路を Group-based Forecasting Module へ伝播したため，反対方向から来る青色と赤色の対象同士が誤って衝突した例である．

図 6.13 は，SDD における誤った予測結果例である．図 6.13(a) は，図 6.12(a) のように急激な行動変化をする対象の経路予測に失敗した例である．図 6.13(b) は，図 6.12(b) のように前方のグループと衝突を回避するために，予測分布が真値と離れた結果例である．図 6.13(c) は，提案手法では建物といった静的障害物に関する特徴を捉えていないため，そのような障害物に衝突した経路を予測した例である．これについては，3 章や 5 章で述べた環境情報を導入することで対処可能だと考えられる．

6.3 まとめ

混雑したシーンにおけるグループの社会的インタラクションを各 Attention 機構でモデル化するために，グループ内とグループ間の関係を捉える Group-based Forecasting Module を提案した．グループ間の Attention は，Prospection Module から伝播される将来の予想経路をグループ間のインタラクションの手がかりとして用いる．グループ内の Attention は，グループ内の他対象のインタラクショ

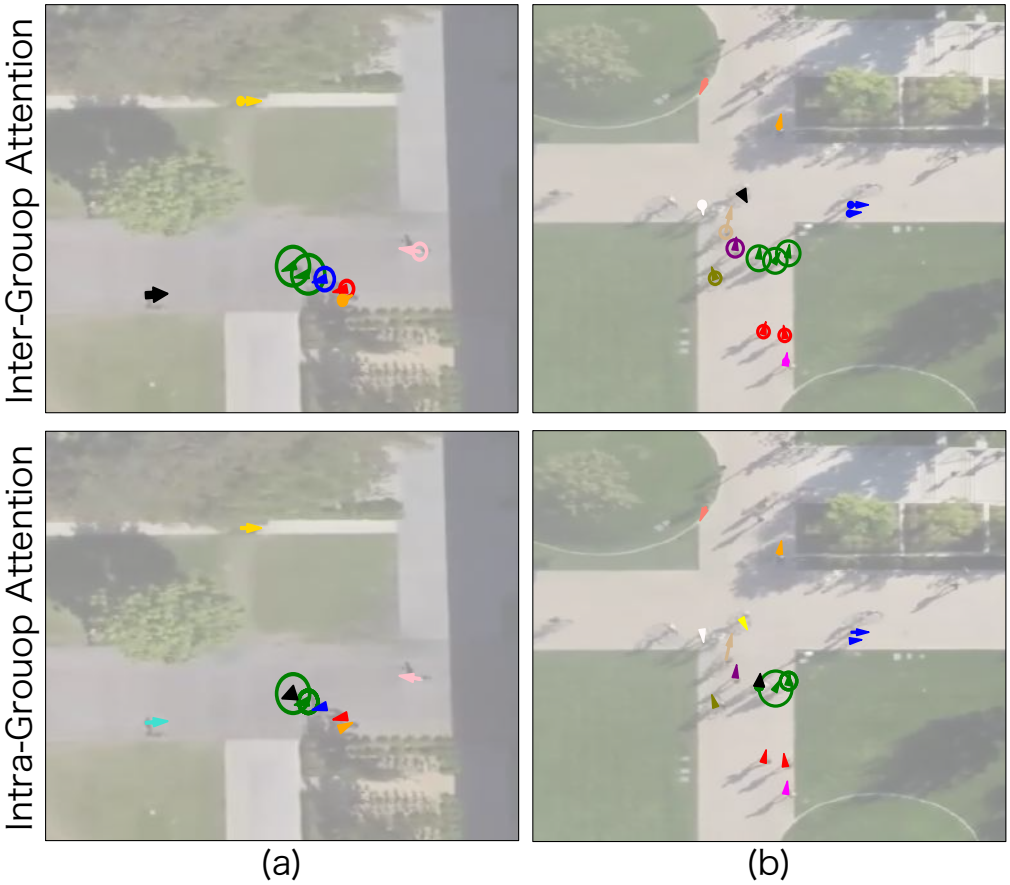


図 6.11: SDD における Group-based Forecasting Module の各 Attention 機構の注意重みの可視化．(a) と (b) は図 6.9 のそれぞれに対応する．

ンをモデル化する．実験結果より，ほとんどのシーンで提案手法は従来手法より予測誤差及び衝突率が減少した．提案手法の有効性を調査するための Ablation study より，個人レベルよりグループレベルでインタラクションを捉える場合が性能面や効率面で良いことを確認した．また，シーン毎に異なる歩行者密度によって，提案手法の適切なパーソナルスペースの設定が予測性能に影響を与えることを確認した．予測結果より，提案手法はグループレベルで歩行者間の社会的インタラクションを考慮することで，他のグループとの衝突を回避する経路予測及び，同じグループで類似する経路を予測した．また，各グループの Attention 機構の注意重みを可視化することで，予測対象が他グループまたは，他対象及び予測対象自身に着目して経路予測したかを分析した．しかし，提案手法では建物や木等の静的障害物を捉えるようにモデル化していないため，静的障害物との衝突を回避することができない．そのため，今後は静的環境を捉えるネットワークの構築が挙げられる．

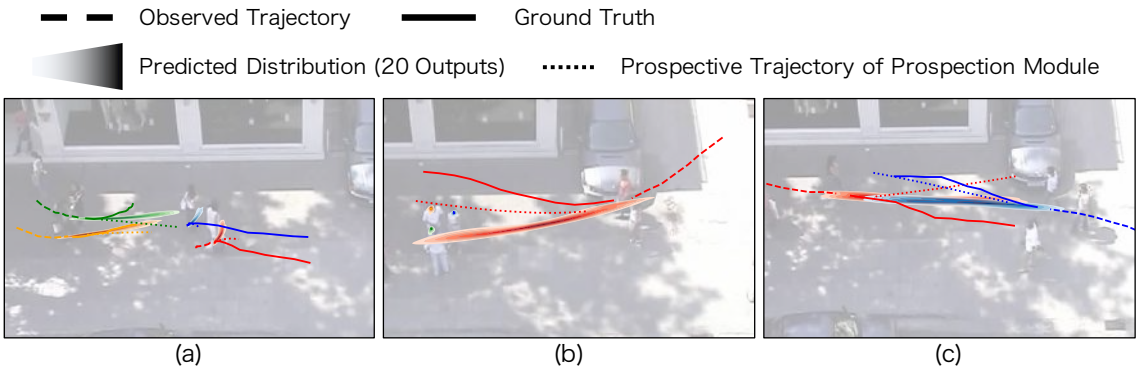


図 6.12: ETH/UCY における誤った予測結果例．(a) は行動を突然変える場合のサンプル，(b) は前方のグループと衝突回避するために真値と離れる経路を予測した場合のサンプル，(c) は Prospection Module から出力される経路がどちらも上方向だった場合のサンプル例である．粗い破線は観測経路，実線は真値，細い破線は Prospection Module の予想経路，分布は 20 つの予測経路の分布である．

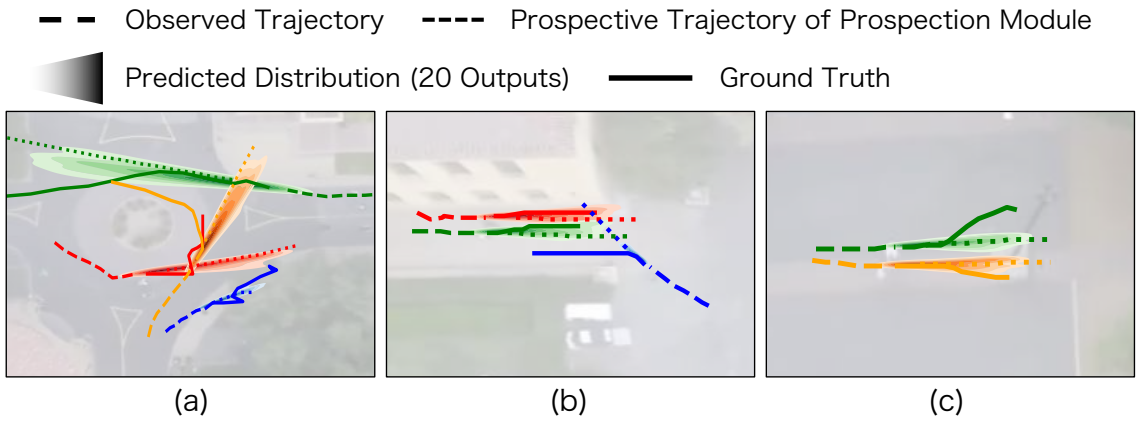


図 6.13: SDD における誤った予測結果例．(a) は行動を突然変える場合のサンプル，(b) は前方のグループと衝突回避するために真値と離れる経路を予測した場合のサンプル，(c) は静的な障害物への衝突した場合のサンプル例を示す．

第7章

結論と展望

本稿では，混雑シーンにおいて複雑な歩行者集団に対して高精度な経路予測を実現するために，グループレベルに基づいたグループ間とグループ内の社会的インタラクションを捉える Group-based Forecasting Module を提案した．以下に，本論文の結論と今後の展望について述べる．

7.1 結論

各章のまとめは以下の通りである．2 章では，経路予測の具体的な社会実装例や処理の流れを述べた後，深層学習を用いた経路予測についての関連研究をまとめた．深層学習を用いた経路予測は，移動対象間の衝突を避けるインタラクションを考慮する経路予測及び，インタラクション以外の課題を扱う経路予測手法の 2 つに大別された．前者はさらにプーリング，アテンション及びその他の 3 つのモデルに分類され，それぞれのモデルについて利点や欠点を述べつつ，各手法について体系的にまとめた．後者は，アイレベルのカメラ映像視点で経路を予測するなど，インタラクションを考慮する予測手法とは異なる問題設定による経路予測手法が提案された．また，経路予測手法で用いられるデータセットや評価指標についても体系的にまとめた．

3 章では，歩行者や自動車等のクラスが異なる複数の予測対象を属性とみなし，予測対象の属性と環境情報を導入した経路予測手法を提案した．既存の経路予測手法では属性数に応じてモデル数が増え計算コストが増加するが，提案手法では属性を one-hot vector としてコンパクトに表現することで計算コストの問題を解消した．コンパクトに表現した属性情報と周囲の環境情報を表現したシーンラベルをネットワークに組み込むことで，属性毎に保有する潜在的な特徴を捉えた経路予測を実現した．

4 章では，混雑シーンにおいて不正確な歩行者データから高精度に予測するために，シーンの各場所が将来どれだけ混雑しているかのマップ，つまり群衆密度マップとして直接予測する手法を提案した．提案手法はパッチベースでモデル化することで，各場所で独立して行動する群衆密度の時空間的ダイナミクスを効率的に捉える．既存の経路予測手法では，不正確な歩行者データを用いると誤った経路を予測するのに対し，提案手法は群衆密度マップを直接予測するため，ほぼ全てのシーンでベースラインより高精度であることを確認した．また，パッチベースに基づく群衆密度予測手法において，空間的パターンと時間的ダイナミクスを単一のネットワークで学習することが有効である結果が得られた．

5 章では，2 章のインタラクションを考慮した経路予測手法の中から代表的な手法を用いて，精度

検証を行った．予測対象を歩行者に限定し実験した結果，歩行者間のインタラクションを考慮する予測手法は歩行者密度が高いシーンで予測誤差が低くなる傾向が確認された．一方で，歩行者密度が低いシーンでは，インタラクションを考慮しない予測手法で十分な予測精度を得られることを確認した．プーリングモデルと比べアテンションモデルによるインタラクション表現が予測精度を向上させつつ対象間の衝突率を減少させることがわかった．

6 章では，4 章と 5 章で得た知見から，混雑シーンにおける歩行者間の社会的インタラクションをグループレベルでモデル化する Group-based Forecasting Module を提案した．Group-based Forecasting Module はグループレベルに基づいたインタラクションを捉えるために，グループ間とグループ内の Attention 機構を導入した．評価実験において，提案手法は個人レベルでインタラクションを捉える予測手法より予測誤差を低減しつつ衝突率を減少させた．また，予測結果より，提案手法は前方のグループとの衝突を避ける経路予測やグループで同じ目的地に向かう経路予測を実現した．

7.2 展望

本論文では，各章で予測対象周辺の環境を考慮した経路予測に関する研究を行った．対象の経路予測時に，建物や歩道などの静的物体及び他の移動物体など周辺環境に依存して予測対象のその後の経路が決定されるため，経路予測ではこれらの情報をうまく取り込むことが重要になる．本論文は，俯瞰視点から取得される情報により Advanced Driving Assistant System や自律型ロボットを含むソーシャルロボットに有用であると考えられる．例えば，LiDAR を搭載した車両から撮影された車載カメラ視点を俯瞰視点に視点変換させることで，歩行者の将来の経路を予測できる．また，交差点や天井に設置された俯瞰カメラを介して自車両やソーシャルロボットに情報を伝播させることで，車載視点及び 1 人称視点カメラからは見えない歩行者の位置を事前に捉えることができる．それらで得た歩行者情報から，各章で提案した手法を上記アプリケーションに実用できると考えている．

経路予測に必要な要素として，対象の属性情報や ID 情報，そしてグループの情報が挙げられる．しかし，これらの情報は実環境で撮影されたデータであり，しばしセンシティブに扱う必要がある．例えば，対象の属性は自転車や自動車から降りた人を歩行者と扱うか否か，運動傾向が類似している歩行者をグループとするのが正確なのか等，提案手法含む経路予測の社会実装には問題が複数生じる可能性がある．特にグループは運動傾向が類似する歩行者集団ではなく，家族や同僚及び友人等の異なる親密な関係を持つグループとしても考えられる．データにそのようなグループ情報のラベリングがあれば 6 章の提案手法を適用できるが，データには親密な関係に関するラベリングがされておらず，1 からデータを作成する必要がある．しかし，現実的な問題としてアノテーションコストがかかること，そして実環境で撮影されるため倫理関係やプライバシー問題が発生する．そのため，シミュレーション環境によるラベリング等が必要になると考えられる．従って，本論文はシミュレーション環境で作成したデータを用いて現実的な経路予測問題に対処することを今後の研究課題とする．

謝　辞

本研究の遂行にあたり，常日頃ご指導を賜りました中部大学工学部情報工学科 山下隆義教授に深く感謝の意を表します．本論文をまとめるにあたり，有益なご討論，ご助言を賜りました中部大学工学部情報工学科 山内康一郎教授，中部大学工学部ロボット理工学科 藤吉弘亘教授，金沢大学高度モビリティ研究所 菅沼直樹教授に謹んで感謝いたします．本研究において，貴重なご意見，ご指導を頂きました中部大学工学部ロボット理工学科 藤吉弘亘教授，中部大学 AI 数理データサイエンスセンター 平川翼講師，東京大学生産技術研究所 菅野裕介准教授，オムロンサイニックエックス株式会社 米谷竜氏，同企業 牛久祥孝氏，同企業 西村真衣氏に心から厚く御礼申し上げます．最後に，本研究にご協力して頂いた山下研究室と藤吉研究室の皆様に感謝致します．

参考文献

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, “Social lstm: Human trajectory prediction in crowded spaces”, Computer Vision and Pattern Recognition, pp.961–971, 2016.

[2] I. Hasan, F. Setti, T. Tsesmelis, A. Del Bue, F. Galasso, and M. Cristani, “MX-LSTM: mixing tracklets and vislets to jointly forecast trajectories and head poses”, Computer Vision and Pattern Recognition, pp.6067-6076, 2018.

[3] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, “Social gan: Socially acceptable trajectories with generative adversarial networks”, Computer Vision and Pattern Recognition, pp.2255–2264, 2018.

[4] T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu, “Multi-agent tensor fusion for contextual trajectory prediction”, Computer Vision and Pattern Recognition, pp.12126–12134, 2018.

[5] Y. Xu, Z. Piao, and S. Gao, “Encoding crowd interaction with deep neural network for pedestrian trajectory prediction”, Computer Vision and Pattern Recognition, pp.5275–5284, 2018.

[6] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, “Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction”, Computer Vision and Pattern Recognition, pp.12085–12094, 2018.

[7] J. Liang, L. Jiang, J. Carlos Niebles, A. G. Hauptmann, and L. Fei-Fei, “Peeking into the future: Predicting future person activities and locations in videos”, Computer Vision and Pattern Recognition, pp.5725–5734, 2019.

[8] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, “Sophie: An attentive gan for predicting paths compliant to social and physical constraints”, Computer Vision and Pattern Recognition, pp.1349–1358, 2019.

[9] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, “Stgat: Modeling spatial-temporal interactions for human trajectory prediction”, International Conference on Computer Vision, pp.6272–6281, 2019.

[10] B. Ivanovic, and M. Pavone, “The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs”, International Conference on Computer Vision, pp.2375–2384, 2019.

[11] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, “Spatio-temporal graph transformer networks for pedestrian trajectory prediction”, European Conference on Computer Vision, pp.507–523, 2020.

[12] K. Mangalam, H. Girase, S. Agarwal, K. H. Lee, E. Adeli, J. Malik, and A. Gaidon, “It is not the journey but the destination: Endpoint conditioned trajectory prediction”, European Conference on Computer Vision, pp.759–776, 2020.

[13] H. Bi, R. Zhang, T. Mao, Z. Deng, and Z. Wang, “How can i see my future? fvtraj: Using first-person view for pedestrian trajectory prediction”, European Conference on Computer Vision, pp.576–593, 2020.

[14] A. Monti, A. Porrello, S. Calderara, P. Coscia, L. Ballan, and R. Cucchiara, “How many observations are enough? knowledge distillation for trajectory forecasting”, Computer Vision and Pattern Recognition, pp.6553–6562, 2022.

[15] A. Vemula, K. Muelling, and J. Oh, “Social attention: Modeling attention in human crowds”, International Conference on Robotics and Automation, pp.1–7, 2018.

[16] C. Choi, and B. Dariush, “Looking to relations for future trajectory forecast”, International Conference on Computer Vision, pp.921–930, 2019.

[17] J. Sun, Q. Jiang, and C. Lu, “Recursive social behavior graph for trajectory prediction”, Computer Vision and Pattern Recognition, pp.660–669, 2020.

[18] A. Lerner, Y. Chrysanthou, and D. Lischinski, “Crowds by example”, Computer Graphics Forum, vol.26, no.3, pp.655–664, 2007.

[19] S. Pellegrini, A. Ess, K. Schindler, and L. V. Gool, “You’ll never walk alone: Modeling social behavior for multi-target tracking”, International Conference on Computer Vision, pp.261–268, 2009.

[20] R. Alexandre, S. Amir, A. Alexandre, and S. Silvio, “Learning social etiquette: Human trajectory understanding in crowded scenes”, European Conference on Computer Vision, pp.549–565, 2016.

[21] M. F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, “Argoverse: 3d tracking and forecasting with rich maps”, Computer Vision and Pattern Recognition, pp.8748–8757, 2019.

[22] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, A. Jain, S. Omari, V. Iglovikov, and P. Ondruska, “One thousand and one hours: Self-driving motion prediction dataset”, Conference on Robot Learning, 2020.

[23] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, “The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections”, arXiv, 2019.

[24] J. Liang, L. Jiang, K. Murphy, T. Yu, and A. Hauptmann, “The garden of forking paths: Towards multi-future trajectory prediction”, Computer Vision and Pattern Recognition, pp.10508–10518, 2020.

[25] A. Rasouli, I. Kotseruba, T. Kunic, and J. K. Tsotsos, “Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction”, International Conference on Computer Vision, pp.6262–6271, 2019.

[26] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, “Trafficpredict: Trajectory prediction for heterogeneous traffic-agents”, Association for the Advancement of Artificial Intelligence, pp.6120–6127, 2019.

[27] S. Malla, B. Dariush, and C. Choi, “Titan: Future forecast using action priors”, Computer Vision and Pattern Recognition, pp.11186–11196, 2020.

[28] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, “nuscenes: A multimodal dataset for autonomous driving”, Computer Vision and Pattern Recognition, pp.11621–11631, 2020.

[29] T. Yagi, K. Mangalam, R. Yonetani, and Y. Sato, “Future person localization in first-person videos”, Computer Vision and Pattern Recognition, pp.7593–7602, 2018.

[30] N. Schneider, and D. M. Gavrila, “Pedestrian path prediction with recursive bayesian filters: A comparative study”, German Conference on Pattern Recognition, pp.174–183, 2013.

[31] C. G. Keller, and D. M. Gavrila, “Will the pedestrian cross? a study on pedestrian path prediction”, Transactions on Intelligent Transportation Systems, vol.15, no.2, pp.494–506, 2014.

[32] J. F. P. Kooij, N. Schneider, F. Flohr, and D. M. Gavrila, “Context-based pedestrian path prediction”, European Conference on Computer Vision, pp.618–633, 2014.

[33] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, “Planning-based prediction for pedestrians”, International Conference on Intelligent Robots and Systems, pp.3931–3936, 2009.

[34] V. Karasev, A. Ayvaci, B. Heisele, and S. Soatto, “Intent-aware long-term prediction of pedestrian motion”, International Conference on Robotics and Automation, pp.2543–2549, 2016.

[35] A. Vemula, K. Muelling, and J. Oh, “Modeling cooperative navigation in dense human crowds”, International Conference on Robotics and Automation, pp.1685–1692, 2017.

[36] O. Styles, T. Guha, and V. Sanchez, “Multi-camera trajectory forecasting with trajectory tensors”, Transactions on Intelligent Transportation Systems, vol.44, no.11, pp.8482–8491, 2022.

[37] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, “Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction”, Computer Vision and Pattern Recognition, pp.14424–14432, 2020.

[38] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, “The walking behaviour of pedestrian social groups and its impact on crowd dynamics”, Public Library of Science, vol.5, no.4, p.e10047, 2010.

[39] E. Kalman, Rudolf, “A new approach to linear filtering and prediction problems”, journal of basic engineering, vol.82, no.1, pp.35–45, 1960.

[40] C. Schöller, V. Aravantinos, F. Lay, and A. Knoll, “What the constant velocity model can teach us about pedestrian motion prediction”, Robotics and Automation Letters, vol.5, no.2, pp.1696–1703, 2020.

[41] A. Møgelmose, M. M. Trivedi, and T. B. Moeslund, “Trajectory analysis and prediction for improved pedestrian safety: Integrated framework and evaluations”, Intelligent Vehicles Symposium, pp.330–335, 2015.

[42] A. Elnagar, “Prediction of moving objects in dynamic environments using kalman filters”, Computer Intelligence in Robotics and Automation, pp.414–419, 2001.

[43] D. Helbing, and P. Molnar, “Social force model for pedestrian dynamics”, Physical review E, vol.51, no.5, p.4282, 1995.

[44] G. Ferrer, and A. Sanfeliu, “Behavior estimation for a complete framework for human motion prediction in crowded environments”, International Conference on Robotics and Automation, pp.5940–5945, 2014.

[45] S. Oli, B. L’Esperance, and K. Gupta, “Human motion behaviour aware planner (hmbap) for path planning in dynamic human environments”, International Conference on Robotics and Automation, pp.1–7, 2013.

[46] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, “Activity forecasting”, European Conference on Computer Vision, pp.201–214, 2012.

[47] R. Nicholas, and K. K. Makoto, “First-person activity forecasting with online inverse reinforcement learning”, International Conference on Computer Vision, pp.3716–3725, 2017.

[48] S. M. LaValle, and J. J. Kuffner, “Randomized kinodynamic planning”, International Journal of Robotics Research, vol.20, no.5, pp.378–400, 2001.

[49] Y. LeCun, L. Bottou, Y. Bengio, and P. Ha ner, “Gradient-based learning applied to document recognition”, Proceedings of the IEEE, vol.86, no.11, pp.2278–2324, 1998.

[50] S. Hochreiter, and J. Schmidhuber, “LONG SHORT-TERM MEMORY”, Neural Computation, vol.9, no.8, pp.1735–1780, 1997.

[51] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling”, arXiv, 2014.

[52] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need”, Advances in Neural Information Processing Systems, pp.5998–6008, 2017.

[53] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling”, arXiv, 2018.

[54] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker, “Desire: Distant future prediction in dynamic scenes with interacting agents”, Computer Vision and Pattern Recognition, pp.336-345, 2017.

[55] N. Deo, and M. M. Trivedi, “Convolutional social pooling for vehicle trajectory prediction”, Computer Vision and Pattern Recognition Workshop, pp.1581–1589, 2018.

[56] N. Bisagno, B. Zhang, and N. Conci, “Group lstm: Group trajectory prediction in crowded scenarios”, European Conference on Computer Vision Workshop, pp.213–225, 2018.

[57] S. Hao, Z. Zhiqun, and H. Zhihai, “Reciprocal learning networks for human trajectory prediction”, Computer Vision and Pattern Recognition, pp.7416–7425, 2020.

[58] T. Fernando, S. Denman, S. Sridharan, and C. Fookes, “Gd-gan: Generative adversarial networks for trajectory prediction and group detection in crowds”, Asian Conference on Computer Vision, pp.314–330, 2018.

[59] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, “Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks”, Advances in Neural Information Processing Systems, pp.137–146, 2019.

[60] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, “Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data”, European Conference on Computer Vision, pp.683–700, 2020.

[61] J. Li, F. Yang, M. Tomizuka, and C. Choi, "Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning", Advances in Neural Information Processing Systems, vol.33, pp.19783–19794, 2020.

[62] C. Tao, Q. Jiang, L. Duan, and P. Luo, "Dynamic and static context-aware lstm for multi-agent motion prediction", European Conference on Computer Vision, pp.547–563, 2020.

[63] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting", International Conference on Computer Vision, pp.9813–9823, 2021.

[64] H. Tran, V. Le, and T. Tran, "Goal-driven long-term trajectory prediction", Winter Conference on Applications of Computer Vision, pp.796–805, 2021.

[65] H. Zhao, and R. P. Wildes, "Where are you heading? dynamic trajectory prediction with expert goal examples", International Conference on Computer Vision, pp.7629–7638, 2021.

[66] L. Shi, L. Wang, C. Long, S. Zhou, M. Zhou, Z. Niu, and G. Hua, "Sgcn: Sparse graph convolution network for pedestrian trajectory prediction", Computer Vision and Pattern Recognition, pp.8994–9003, 2021.

[67] L. Ruochen, K. Stamos, and P. H. S. Hubert, "Multiclass-sgcn: Sparse graph-based trajectory prediction with agent class embedding", International Conference on Image Processing, 2022.

[68] K. Guo, W. Liu, and J. Pan, "End-to-end trajectory distribution prediction based on occupancy grid maps", Computer Vision and Pattern Recognition, pp.2242–2251, 2022.

[69] L. Li, M. Pagnucco, and Y. Song, "Graph-based spatial transformer with memory replay for multi-future pedestrian trajectory prediction", Computer Vision and Pattern Recognition, pp.2231–2241, 2022.

[70] N. Rhinehart, R. McAllister, K. Kitani, and S. Levine, "Precog: Prediction conditioned on goals in visual multi-agent settings", International Conference on Computer Vision, pp.2821–2830, 2019.

[71] S. Yi, H. Li, and X. Wang, "Pedestrian behavior understanding and prediction with deep neural networks", European Conference on Computer Vision, pp.263–279, 2016.

[72] S. Huang, X. Li, Z. Zhang, Z. He, F. Wu, W. Liu, J. Tang, and Y. Zhuang, "Deep learning driven visual path prediction from a single image", IEEE Transactions on Image Processing, vol.25, no.12, pp.5892–5904, 2016.

[73] A. Bhattacharyya, M. Fritz, and B. Schiele, "Long-term on-board prediction of people in traffic scenes under uncertainty", Computer Vision and Pattern Recognition, pp.4194–4202, 2018.

[74] W. Luo, B. Yang, and R. Urtasun, “Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net”, Computer Vision and Pattern Recognition, pp.3569–3577, 2018.

[75] H. Minoura, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, “Path predictions using object attributes and semantic environment”, International Conference on Computer Vision Theory and Applications, pp.19–26, 2019.

[76] J. Hong, B. Sapp, and J. Philbin, “Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions”, Computer Vision and Pattern Recognition, pp.8454–8462, 2019.

[77] Y. Yao, M. Xu, C. Choi, J. D. Crandall, M. E. Atkins, and B. Dariush, “Egocentric vision-based future vehicle localization for intelligent driving assistance systems”, International Conference on Robotics and Automation, pp.9711–9717, 2019.

[78] O. Styles, A. Ross, and V. Sanchez, “Forecasting pedestrian trajectory with machine-annotated training data”, Intelligent Vehicles Symposium, pp.716–721, 2019.

[79] L. Fang, Q. Jiang, J. Shi, and B. Zhou, “Tpnet: Trajectory proposal network for motion prediction”, Computer Vision and Pattern Recognition, pp.6797–6806, 2020.

[80] J. Liang, L. Jiang, and A. Hauptmann, “Simaug: Learning robust representations from simulation for trajectory prediction”, European Conference on Computer Vision, pp.275–292, 2020.

[81] G. Francesco, H. Irtiza, C. Marco, and G. Fabio, “Transformer networks for trajectory forecasting”, International Conference on Pattern Recognition, pp.10335–10342, 2021.

[82] K. Mangalam, Y. An, H. Girase, and J. Malik, “From goals, waypoints & paths to long term human trajectory forecasting”, International Conference on Computer Vision, pp.15233–15242, 2021.

[83] L. Neumann, and A. Vedaldi, “Pedestrian and ego-vehicle trajectory prediction from monocular camera”, Computer Vision and Pattern Recognition, pp.10204–10212, 2021.

[84] K. Sohn, H. Lee, and X. Yan, “Learning structured output representation using deep conditional generative models”, Advances in Neural Information Processing Systems, pp.3483–3491, 2015.

[85] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets”, Advances in Neural Information Processing Systems, pp.2672–2680, 2014.

[86] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting”, International Joint Conferences on Artificial Intelligence, pp.3634–3640, 2018.

[87] M. T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation”, Empirical Methods in Natural Language Processing, pp.1412–1421, 2015.

[88] L. van der Maaten, and G. Hinton, “Visualizing data using t-SNE”, Journal of Machine Learning Research, vol.9, pp.2579–2605, 2008.

[89] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise”, Knowledge Discovery and Data Mining, vol.96, pp.226–231, 1996.

[90] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention”, International Conference on Machine Learning, pp.2048–2057, 2015.

[91] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks”, International Conference on Learning Representations, 2018.

[92] J. Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, “Toward multimodal image-to-image translation”, Advances in Neural Information Processing Systems, pp.465–476, 2017.

[93] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks”, Computer Vision and Pattern Recognition, pp.7794–7803, 2018.

[94] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network”, Computer Vision and Pattern Recognition, pp.2881–2890, 2017.

[95] H. Geoffrey, V. Oriol, and D. Jeffrey, “Distilling the knowledge in a neural network”, NIPS Deep Learning and Representation Learning Workshop, 2015.

[96] T. N. Kipf, and M. Welling, “Semi-supervised classification with graph convolutional networks”, International Conference on Learning Representations, 2017.

[97] R. Garg, V. K. B.G., G. Carneiro, and I. Reid, “Unsupervised cnn for single view depth estimation: Geometry to the rescue”, European Conference on Computer Vision, pp.740–756, 2016.

[98] C. Godard, O. Mac Aodha, and G. J. Brostow, “Unsupervised monocular depth estimation with left-right consistency”, Computer Vision and Pattern Recognition, pp.270–279, 2017.

[99] H. Dirk, “Stochastische methoden, nichtlineare dynamik und quantitative modelle sozialer prozesse”, Shaker, 1993.

[100] D. K. Armen, and D. Ove, “Aleatory or epistemic? does it matter?”, Structural Safety, vol.31, no.2, pp.105–112, 2009.

[101] A. George, B. Asad, C. Keith, L. Yooyoung, F. Jonathan, G. Afzad, J. David, D. Andrew, S. Alan, G. Yvette, K. Wessel, Q. Georges, M. Joao, S. David, and B. Saverio, “Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search”, In TRECVID, 2018.

[102] N. Rhinehart, K. M. Kitani, and P. Vernaza, “R2p2: A reparameterized pushforward policy for diverse, precise generative path forecasting”, European Conference on Computer Vision, pp.772–788, 2018.

[103] O. Makansi, E. Ilg, O. Cicek, and T. Brox, “Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction”, Computer Vision and Pattern Recognition, pp.7144–7153, 2019.

[104] T. Tieleman, and G. Hinton, “Lecture 6.5-rmsprop: Divide the gradint by a running average of its recent magnitude”, COURSERA: Neural networks for machine learning, vol.4, no.2, pp.26–31, 2012.

[105] F. Altché, and A. de La Fortelle, “An lstm network for highway trajectory prediction”, International Conference on Intelligent Transportation Systems, pp.353-0359, 2017.

[106] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, “Vehicle trajectory prediction based on motion model and maneuver recognition”, International Conference on Intelligent Robots and Systems, pp.4363–4369, 2013.

[107] D. Kang, Z. Ma, and A. B. Chan, “Beyond counting: Comparisons of density maps for crowd analysis tasks - counting, detection, and tracking”, Transactions on Circuits and Systems for Video Technology, vol.29, no.5, pp.1408–1422, 2018.

[108] V. A. Sindagi, and V. M. Patel, “A survey of recent advances in cnn-based single image crowd counting and density estimation”, Pattern Recognition Letters, vol.107, pp.3–16, 2018.

[109] M. S. Zitouni, H. Bhaskar, J. Dias, and M. E. Al-Mualla, “Advances and trends in visual crowd analysis: A systematic survey and evaluation of crowd modelling techniques”, Neurocomputing, vol.186, pp.139–159, 2016.

[110] Q. Wang, J. Gao, W. Lin, and Y. Yuan, “Learning from synthetic data for crowd counting in the wild”, Computer Vision and Pattern Recognition, pp.8198–8207, 2019.

[111] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection", Computer Vision and Pattern Recognition, pp.2117–2125, 2017.

[112] Y. Fang, B. Zhan, W. Cai, S. Gao, and B. Hu, "Locality-constrained spatial transformer network for video crowd counting", International Conference on Multimedia and Expo, pp.814–819, 2019.

[113] J. Gao, W. Lin, B. Zhao, D. Wang, C. Gao, and J. Wen, "C^3 framework: An open-source pytorch code for crowd counting", arXiv, 2019.

[114] W. Liu, K. M. Lis, M. Salzmann, and P. Fua, "Geometric and physical constraints for drone-based head plane crowd density estimation", International Conference on Intelligent Robots and Systems, 2019.

[115] D. Ha, and J. Schmidhuber, "Recurrent world models facilitate policy evolution", Advances in Neural Information Processing Systems, pp.2450–2462, 2018.

[116] K. H. Zeng, W. B. Shen, D. A. Huang, M. Sun, and J. Carlos Niebles, "Visual forecasting by imitating dynamics in natural sequences", International Conference on Computer Vision, pp.2999–3008, 2017.

[117] V. Nair, and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines", International Conference on Machine Learning, pp.807–814, 2010.

[118] D. P. Kingma, and J. Ba, "Adam: A method for stochastic optimization", International Conference on Learning Representations, 2015.

[119] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", European Conference on Computer Vision, pp.740–755, 2014.

[120] Y. Niitani, T. Ogawa, S. Saito, and M. Saito, "Chainercv: a library for deep learning in computer vision", ACM International Conference on Multimedia, 2017.

[121] D. Roy, T. Ishizaka, C. K. Mohan, and A. Fukuda, "Vehicle trajectory prediction at intersections using interaction based generative adversarial networks", International Conference on Intelligent Transportation Systems, pp.2318–2323, 2019.

[122] P. Chakraborty, A. Sharma, and C. Hegde, "Freeway traffic incident detection from cameras: A semi-supervised learning approach", International Conference on Intelligent Transportation Systems, pp.1840–1845, 2018.

[123] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking", International Conference on Image Processing, pp.3464-3468, 2016.

[124] N. Deo, and M. M. Trivedi, “Scene induced multi-modal trajectory forecasting via planning”, arXiv, 2019.

[125] H. Soo Park, J. J. Hwang, Y. Niu, and J. Shi, “Egocentric future localization”, Computer Vision and Pattern Recognition, pp.4697–4705, 2016.

[126] S. Becker, R. Hug, W. Húbner, and M. Arens, “Red: A simple but effective baseline predictor for the trajnet benchmark”, European Conference on Computer Vision Workshop, pp.138–153, 2018.

[127] A. Graves, “Generating sequences with recurrent neural networks”, arXiv, 2013.

[128] F. A. Aveni, “The not-so-lonely crowd: Friendship groups in collective behavior”, Sociometry, vol.40, no.1, pp.96–99, 1977.

[129] D. Xie, S. Todorovic, and S. C. Zhu, “Inferring “dark matter” and “dark energy” from videos”, International Conference on Computer Vision, pp.2224–2231, 2013.

[130] E. T. Hall, The Hidden Dimension, Anchor Books, 1966.

[131] H. Cui, V. Radosavljevic, F. C. Chou, T. H. Lin, T. Nguyen, T. K. Huang, J. G. Schneider, and N. Djuric, “Multimodal trajectory predictions for autonomous driving using deep convolutional networks”, International Conference on Robotics and Automation, pp.2090-2096, 2019.

研究業績一覧

学術論文

[1] Hiroaki Minoura, Tsubasa Hirakawa, Yusuke Sugano, Takayoshi Yamashita, Hironobu Fujiyoshi, “Utilizing Human Social Norms for Multimodal Trajectory Forecasting via Group-based Forecasting Module,” IEEE Transactions on Intelligent Vehicles (T-IV), 2022.

[2] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “Deep Learning を用いた移動物体間のインタラクションを考慮した経路予測の研究動向,” 電子情報通信学会, Vol.J105-D, No.5, pp.372–404, 2022.

[3] Hiroaki Minoura, Ryo Yonetani, Mai Nishimura, Yoshitaka Ushiku, “Crowd Density Forecasting by Modeling Patch-based Dynamics,” IEEE Robotics and Automation Letters (RA-L), Vol.6, Issue.2, pp.287–294, 2021.

[4] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “移動対象の属性と環境情報を導入した LSTM による経路予測,” 精密工学会誌, Vol.86, No.12, pp.961–968, 2020.

国際会議発表論文

[1] Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, “Understanding of Feature Representation in Convolutional Neural Networks and Vision Transformer,” The 18th International Conference on Computer Vision Theory and Application (VISAPP, Oral), 2022.

[2] Yuzhi Shi, Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, Mitsuru Nakazawa, Yeongnam Chae, Björn Stenger, “Action Spotting in Soccer Videos Using Multiple Scene Encoders,” The 26th International Conference on Pattern Recognition (ICPR, Oral), 2022.

[3] Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, Mitsuru Nakazawa, Yeongnam Chae, Björn Stenger, “Action Spotting and Temporal Attention Analysis in Soccer Videos,” The 17th International Conference on Machine Vision Applications (MVA, Oral), 2021.

[4] Masaki Miyata, Katsutoshi Shiraki, Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, “Relational Subgraph for Graph-based Path Prediction,” The 17th International Conference on Machine Vision Applications (MVA, Poster), 2021.

[5] Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, “Path predictions using object attributes and semantic environment,” The 14th International Conference on Computer Vision Theory and Application (VISAPP, Oral), 2019.

国内会議発表論文

[1] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “ViT の派生手法はどのような特徴表現を獲得しているか?,” 第 25 回 画像の認識・理解シンポジウム (MIRU, Poster), 2022.

[2] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “Group-based Forecasting Module による歩行者の社会的相互作用を考慮したマルチモーダルな軌跡予測,” 自動車技術会 2022 年春季大会, 2022.

[3] 伊佐 稜, 白木 克俊, 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “物体追跡による経路データセットの自動生成,” 電気・電子・情報関係学会 東海支部連合大会, 2021.

[4] 箕浦 大晃, 平川 翼, 菅野 裕介, 山下 隆義, 藤吉 弘亘, “Group-based Forecasting Module による歩行者の社会的相互作用を考慮したマルチモーダルな軌跡予測,” 第 24 回 画像の認識・理解シンポジウム (MIRU, Short Oral), 2021.

[5] 宮田 昌樹, 白木 克俊, 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “関係性部分グラフを用いた Graph Convolutional Network による経路予測,” 第 24 回 画像の認識・理解シンポジウム (MIRU, Poster), 2021.

[6] Yuzhi Shi, Hiroaki Minoura, Tsubasa Hirakawa, Takayoshi Yamashita, Hironobu Fujiyoshi, “Action Spotting in Soccer Videos via Transformer with Past and Future Encoders,” The 24rd Meeting on Image Recognition and Understanding (MIRU, Poster), 2021.

[7] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “Deep Learning を用いた経路予測の研究動向,” 電子情報通信学会技術報告, パターン認識・メディア理解研究会 (PRMU, Presentation), 2020.

[8] 箕浦 大晃, 米谷 竜, 西村 真衣, 牛久 祥孝, “パッチベースモデルによる群衆予測,” 第 23 回 画像の認識・理解シンポジウム (MIRU, Poster), 2020.

[9] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “衝突インタラクションを考慮した経路予測における評価指標の提案,” 第 23 回 画像の認識・理解シンポジウム (MIRU, Poster), 2020.

[10] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “Dilated Causal Convolution を用いた移動物体の経路予測に関する研究,” 新学術領域研究 生物移動情報学 (Poster), 2019.

[11] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “移動対象の属性と環境情報を導入した LSTM による経路予測,” 第 24 回 画像センシングシンポジウム (SSII, Poster), 2018.

[12] 箕浦 大晃, 平川 翼, 山下 隆義, 藤吉 弘亘, “移動対象の属性と環境情報を導入した LSTM による経路予測,” 新学術領域研究 生物移動情報学 (Poster), 2018.

学術表彰

[1] 2022 年　電子情報通信学会東海支部 学生研究奨励賞.
題目：Deep Learning を用いた移動物体間のインタラクションを考慮した経路予測の研究動向

書籍

[1] 箕浦大晃 (3 章, 6 章担当)　Vision Transformer 入門, 技術評論社 ISBN 978-4-297-13058-9, 2022.

[2] 箕浦大晃 (4 章担当)　コンピュータビジョン最前線 Summer 2022, 共立出版 ISBN 9784320125445, 2022.