交通シーン画像からの歩行者の注視対象物推定

○村上大斗[1,†]，陳嘉雷[1]，出口大輔[1]，平山高嗣[2,1]，川西康友[3,1]，村瀬洋[1]

○ Hiroto MURAKAMI[1,†]，Jialei CHEN[1]，Daisuke DEGUCHI[1]，
Takatsugu HIRAYAMA[2,1]，Yasutomo KAWANISHI[3,1]，and Hiroshi MURASE[1]

1 ： 名古屋大学 大学院情報学研究科
2 ： 人間環境大学 環境科学部
3 ： 理化学研究所情報統合本部 ガーディアンロボットプロジェクト
†： hiroto.murakami@nagoya-u.jp

＜要約＞歩行者の注視対象物推定は，自動運転車両の実現に必要となる歩行者の行動予測のための重要な要素である．本発表では，交通シーンで歩行者が注視している物体（PEdestrians' Gaze Object: PEGO）を推定する PEGO Transformer V2 を提案する．従来手法の多くは，高解像度の頭部画像と視線ヒートマップを用いて歩行者の注視対象物を推定している．しかしながら，遠方にいる歩行者に対して，高解像度の頭部画像を得られないため，注視対象物の推定精度が低いという問題があった．そこで PEGO Transformer V2 は，歩行者全身の特徴に加え，歩行者位置情報も活用することで性能改善を図る．実験の結果，提案手法である PEGO Transformer V2 は，シーン中の歩行者ごとにそれぞれ正しい注視対象物を推定できることを確認した．
＜キーワード＞歩行者注視対象物，物体検出，視線検出，交通シーン

1 はじめに

歩行者の注視対象物（PEdestrians' Gaze Object: PEGO）推定は，自動運転車両等が歩行者の行動を予測する際に必要となる重要な情報であり，未だ解決されていない重要な技術課題である．例えば，図 1 のシーンにおいて，黄枠で示す歩行者は，奥の赤枠で示す車両を見ていることから，自車両の接近に気づかず横断するのではないか，といった行動予測が可能である．この例からも分かるように，PEGO の推定結果は歩行者が今後どのような行動をとるかを判断する重要な手がかりとなる．

本発表では，車載カメラ画像に写る歩行者に対し，同じ画像中のどの物体を見ているかの推定を目的とする．車載カメラ画像に写る歩行者の状態は主に以下の 3 つに分類できる．

- 画像撮影範囲内の特定の物体を見ている歩行者
- ぼんやりと進行方向を見ている歩行者
- 画像撮影範囲外を見ている歩行者

このうち，画像撮影範囲内の特定の物体を見ている歩行者に着目する．対象歩行者の注視対象物を認識することで，その歩行者の意図や行動予測に繋げることが可能である．注視対象物の候補には，自車両と車載カメラ画像に写る他の物体がある．ここで自車両を見ている歩行者は，その接近や車両の運転行動変化に気づけることから，比較的安全であると考えられる．畑ら [1] は歩行者が自車両を見ている場面を “アイコンタクト” と定義し，その検出に取り組んでいる．一方，自車両以外の物体を見ている歩行者に対し，その対象を推定することができればより詳細なシーン理解に繋がる．我々は，信号機を見ている歩行者は周囲を十分確認していることから比較的安全であり，ボールを追う子供は危険性が高いといった知識を獲得している．このような知見を活用するためには，各歩行者の注視対象物が何であるかを認識する必要がある．よって本発表では，畑らの研究を発展させ，自車両以外の物体を見ている歩行者の注視対象物の推定を行なう．

図 1 注視対象物に基づく行動予測の例

2 関連研究

2.1 既存の注視対象物推定手法

“目は口ほどにものを言う” ということわざがあるように，人物の視線には意図や興味の対象が強く表出される．視線はコミュニケーションにおいて重要な役割を担うことから，交通シーンに限らず様々なドメインで研究が行なわれている．

まず Recasens ら [2] が提案した GazeFollow は，推定対象の人物とその周囲の物体を写したシーン画像，高解像度な頭部画像，及びその位置を入力して，対象人物の視線を推定している．Recasens らは GazeFollow の学習に GazeFollow データセットを用いている．GazeFollow データセットは，複数の大規模データセットから人物画像を含むシーンを選別し，第三者のアノテータによって対象人物の目の位置と注視点をアノテーションしている．この研究は，最近の視線推定・追従タスクの先駆的な存在であり，GazeFollow データセットも数多くの研究（ [3, 4, 5] など）で用いられている．

次に Wang ら [5] が提案した GaTector は，注視対象物を物体レベルで推定する手法である．GaTector の入力は GazeFollow と同様であり，シーン画像，高解像度な頭部画像，さらに頭部位置を組み合わせて視野ヒートマップを推定する．また同時に，シーン画像から注視対象物候補の検出を行なう．そして，検出した物体の各 Bounding Box (BBox) 内に含まれる視野ヒートマップのエネルギー総和を計算し，最も高い物体を注視対象物とする．GaTector のモデルの訓練には，小売店シーンの注視対象物検出に特化した GOO (Gaze On Object) データセット [6] が使われている．よって，GaTector の対象は小売店のように人物と対象物体が近いシーンであり，本研究が対象とする交通シーンとは人物と物体の位置関係や物体が配置される密度が大きく異なる．特に，車載カメラで撮影された歩行者の多くはカメラからの距離が遠く，解像度が低くなる．そのため，小売店シーンで撮影された歩行者よりも小さく不鮮明に写ることから，交通シーンでは注視領域の推定に必要な高解像度な頭部画像の獲得が困難である．また Wang らは，視野ヒートマップの推定精度が低く，性能のボトルネックになることを考察で述べている．よって，高解像度な頭部画像や視野ヒートマップに依存しない新しい手法が必要である．

畑ら [1] が提案した Eye-contact Transformer は，車載カメラ画像に写る歩行者が自車両を見ているか否か（アイコンタクト）を検出する手法である．畑らは，車載カメラ画像に写る歩行者の顔領域は不鮮明で既存の視線推定手法が使えないことを実験によって示している．そこで，歩行者の骨格と周囲環境の特徴を統合利用することで，アイコンタクト検出を実現している．さらに，既存の交通シーンデータセット [7] に追加のアノテーションを実施し，複数のアノテータによって，歩行者のアイコンタクトラベルを記録している．本発表は畑らの研究を発展させ，歩行者が自車両以外を見ている場合に，見ている物体が何であるかの推定を目的とする．

2.2 PEGO データセット

これまで我々は，歩行者の注視対象物推定のための PEGO データセット [8] を構築してきた．本データセットは，既存の交通シーンデータセット [9, 10] に対し，新たに歩行者の注視点をアノテーションすることで構築したデータセットである．このデータセットは，1,157 人の対象歩行者の ID と BBox 座標，注視点座標，歩行者の状態（自車両を見ている，後ろ向き），注視点に対応する物体の BBox 座標とそのクラス情報をアノテーションとして含んでいる．なお，人による判断の曖昧さを考慮し，各歩行者に対して 3 人のアノテータがアノテーションしている．さらに，このデータセットを用いて，交通シーンにおける歩行者の注視対象物（PEGO）を推定する PEGO Transformer を提案している．本発表では，PEGO Transformer に対し，歩行者と物体の位置情報と，推定対象か否かをそれぞれ明示的に入力する改良を加えた PEGO Transformer V2 を提案するとともに，PEGO データセットを用いてその有用性を示す．

3 PEGO Transformer V2

本発表では，歩行者の注視対象物推定を行なう Transformer ベースの手法 “PEGO Transformer V2” を提案する．従来の注視対象物推定手法とは異なり，本手法は高解像度な頭部画像を必要とせず，歩行者の全身から得られる画像特徴を用いて注視対象物を推定する．また，視線ヒートマップを陽に用いず，シーン画像から検出された物体に対して各歩行者の注視対象物の尤

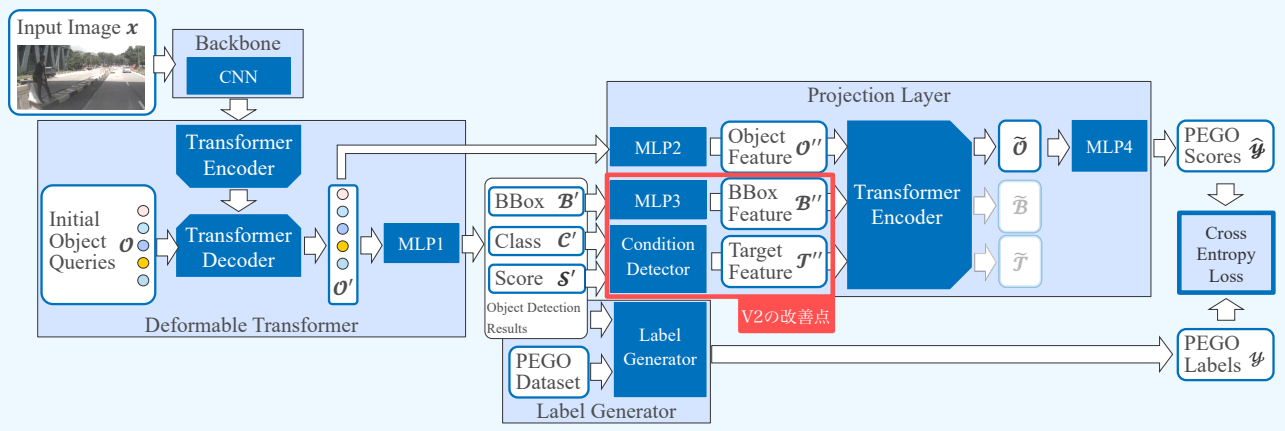

図 2　PEGO Transformer V2 のアーキテクチャ

度が高くなるように PEGO Transformer V2 を学習させる．図 2 に PEGO Transformer V2 のアーキテクチャを示す．PEGO Transformer V2 は入力画像から特徴を抽出する Backbone，物体に対応する特徴を捉える Deformable Transformer，特徴を利用して注視対象物の予測結果を生成する Projection Layer からなり，PEGO Transformer V2 の学習に利用する教師信号を生成する Label Generator と合わせて 4 モジュールから構成される．以下で各モジュールについて紹介する．

Backbone　Backbone は CNN で構成され，Deformable Transformer の入力に用いる画像特徴量の獲得を行なう．入力画像 $\boldsymbol{x} \in \mathbb{R}^{C\times H\times W}$ を CNN バックボーン（ResNet [11]）に入力し，式 (1) に示す画像特徴量 $\boldsymbol{f} \in \mathbb{R}^{C\times H\times W}$ を出力する．ただし，C はカラーチャンネル，H は画像の高さ，W は画像の幅である．

$$\boldsymbol{f} = \mathrm{CNN}(\boldsymbol{x}) \tag{1}$$

Deformable Transformer　Deformable Transformer は Deformable Transformer Encoder （DTE）と Deformable Transformer Decoder （DTD） [12] で構成され，画像特徴量 $\boldsymbol{f}$ から各物体に対応する特徴量の Object Query $\boldsymbol{o}' \in \mathbb{R}^D$ を出力する．Deformable Transformer は，まず式 (2) に示すように $\boldsymbol{f}$ を平坦化（flatten）し，Positional Embedding $\mathcal{P}$ を加えて DTE に入力する．

$$\boldsymbol{f}' = \mathrm{DTE}\big(\mathrm{flatten}(\boldsymbol{f}) + \mathcal{P}\big) \tag{2}$$

DTE は Self Attention モジュールによって各画像特徴同士がそれぞれ影響し合う相互関係を捉える．次に，DTE の出力 $\boldsymbol{f}'$ と Initial Object Query $\mathcal{O} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_N | \boldsymbol{o}_i \in \mathbb{R}^D\}$ を DTD に入力する．$\boldsymbol{o}_1 \sim \boldsymbol{o}_N$ はそれぞれ学習可能な D 次元のパラメータであり，ランダムな値に初期化して用いる．DTD の Cross Attention モジュールによって，$\mathcal{O}$ は $\boldsymbol{f}'$ の情報を統合的に取り入れ，物体と 1 対 1 で対応する特徴量 Object Query $\mathcal{O}'$ へ変換される．

$$\mathcal{O}' = \mathrm{DTD}(\boldsymbol{f}', \mathcal{O}) \tag{3}$$

DTD が出力する $\mathcal{O}' = \{\boldsymbol{o}'_1, \boldsymbol{o}'_2, \ldots, \boldsymbol{o}'_N | \boldsymbol{o}'_i \in \mathbb{R}^D\}$ を MLP_1 に入力すると，対応する物体の BBox 座標 $\mathcal{B}' = \{\boldsymbol{b}'_1, \boldsymbol{b}'_2, \ldots, \boldsymbol{b}'_N | \boldsymbol{b}'_i \in \mathbb{R}^4\}$，クラスラベル $\mathcal{C}' = \{c'_1, c'_2, \ldots, c'_N | c'_i \in \mathbb{R}\}$，及びその検出尤度スコア $\mathcal{S}' = \{s'_1, s'_2, \ldots, s'_N | s' \in \mathbb{R}\}$ が得られる．

$$\boldsymbol{b}'_i, c'_i, s'_i = \mathrm{MLP}_1(\boldsymbol{o}'_i) \tag{4}$$

Projection Layer　Projection Layer は各物体の情報を統合利用して，それぞれの物体の注視対象物尤度を示す PEGO スコア $\hat{\boldsymbol{y}} \in \mathbb{R}^N$ を出力する．Projection Layer は，主に Transformer Encoder (TE) と MLP で構成される．その処理手順を 図 3 に示す．Projection Layer の入力は各物体に対応する Object Query $\boldsymbol{o}'_1 \sim \boldsymbol{o}'_N$，BBox 座標 $\boldsymbol{b}'_1 \sim \boldsymbol{b}'_N$，クラスラベル $c'_1 \sim c'_N$，検出尤度スコア $s'_1 \sim s'_N$ である．まず各 $\boldsymbol{o}'_i$ は MLP_2 によって Object Feature $\boldsymbol{o}''_i \in \mathbb{R}^D$ に変換する．MLP_2 は $\boldsymbol{o}'_i$ から注視対象物検出に必要な特徴量の獲得を目指す．

$$\boldsymbol{o}''_i = \mathrm{MLP}_2(\boldsymbol{o}'_i) \tag{5}$$

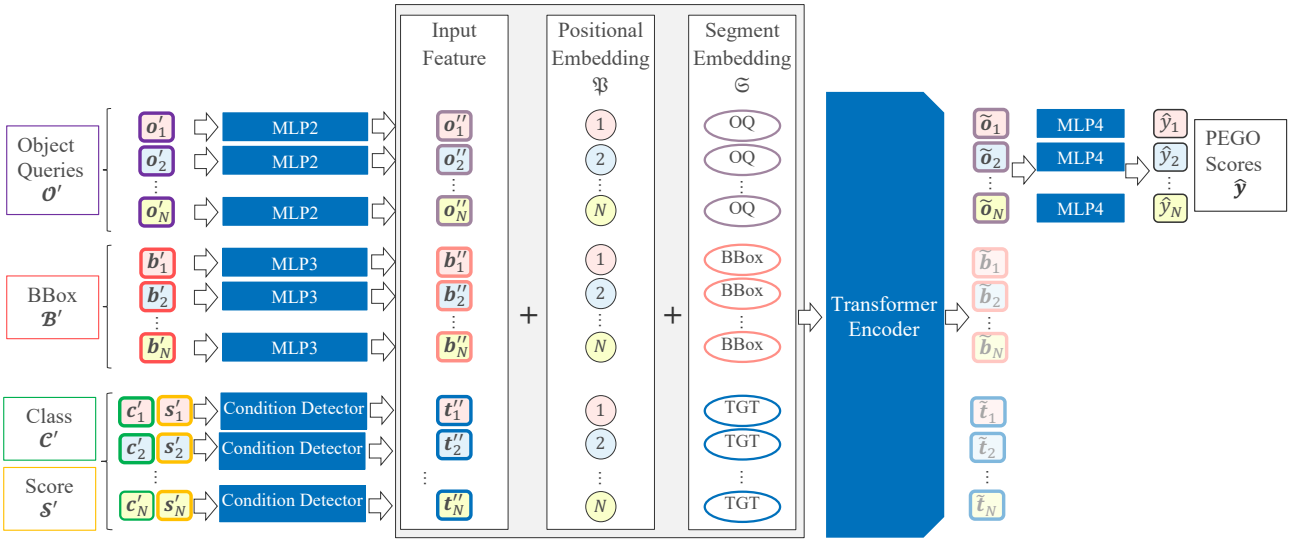

図 3　Projection Layer の処理手順

次に $\boldsymbol{b}'_i$ は MLP_3 によって TE に入力可能な特徴量の BBox Feature $\boldsymbol{b}''_i \in \mathbb{R}^D$ に変換する．

$$\boldsymbol{b}''_i = \mathrm{MLP}_3(\boldsymbol{b}'_i) \tag{6}$$

そして c'_i と s'_i を Condition Detector に入力し，対象歩行者と注視対象物候補を区別する Target Feature $\boldsymbol{t}''_i \in \mathbb{R}^D$ を得る．Condition Detector は 式 (7) に示す定義によって，対象歩行者を示す特徴量 $\boldsymbol{t}_p$，注視対象物候補を示す特徴量 $\boldsymbol{t}_o$，無視する物体を示す特徴量 $\boldsymbol{t}_\phi$ を出力する．

$$\boldsymbol{t}''_i = \begin{cases} \boldsymbol{t}_p & \left((s'_i > \delta) \wedge (c'_i = \sigma_p) \wedge (\boldsymbol{t}_p \notin \mathcal{T}'')\right) \\ \boldsymbol{t}_o & \left((s'_i > \delta) \wedge (\boldsymbol{t}_i \neq \boldsymbol{t}_p)\right) \\ \boldsymbol{t}_\phi & \left(s'_i \leq \delta\right) \end{cases} \tag{7}$$

なお，σ_p は歩行者のクラスラベルを表し，δ は検出尤度スコアのしきい値である．ここで，1 つのシーン画像内に複数の歩行者が存在する場合がある．このとき，1 人の歩行者に $\boldsymbol{t}_p$ を割り当て，その他の歩行者は $\boldsymbol{t}_o$ を割り当てる．そして，この処理を全ての歩行者に順に繰り返す．各歩行者に対する処理は独立なので，複数の歩行者で並列して注視対象物を推定できる．こうして得られた各 $\boldsymbol{o}''_i, \boldsymbol{b}''_i, \boldsymbol{t}''_i$ に対し，同じ物体に対応する特徴量であることを示すために，Positional Embedding $\mathfrak{P} = \{\mathfrak{p}_1, \mathfrak{p}_2, \ldots, \mathfrak{p}_3 N | \mathfrak{p}_i \in \mathbb{R}^D\}$ を加算する．Positional Embedding $\mathfrak{P}$ は，N 種類の学習可能な D 次元のパラメータで構成され，同じ値を 3 つずつ含む．同じ物体に対応する $\boldsymbol{o}''_i, \boldsymbol{b}''_i, \boldsymbol{t}''_i$ に対してそれぞれ同じ値を加算して，同じ物体から派生した特徴量であることを表現する．さらに，入力特徴量の種類を区別するために，BERT [13] に倣い Segment Embedding $\mathfrak{S} = \{\mathfrak{s}_1, \mathfrak{s}_2, \ldots, \mathfrak{s}_3 N | \mathfrak{s}_i \in \mathbb{R}^D\}$ を加算する．Segtment Embedding $\mathfrak{S}$ は，3 種類の学習可能な D 次元のパラメータで構成され，同じ値を N 個ずつ含む．同じ種類の特徴量に対応する $\boldsymbol{o}''_1 \sim \boldsymbol{o}''_N$，$\boldsymbol{b}''_1 \sim \boldsymbol{b}''_N$，$\boldsymbol{t}''_1 \sim \boldsymbol{t}''_N$ に対してそれぞれ同じ値を加算して，その特徴量が Object Feature，BBox Feature，Target Feature のどれを表すのかを表現する．よって TE への入力は，$\mathcal{O}''$，$\mathcal{B}''$，$\mathcal{T}''$ と，それらに $\mathfrak{P}$，$\mathfrak{S}$ を足し合わせた特徴量となる．これらの処理の具体的な例を 図 4 に示す．各物体で $\mathcal{O}' = \{\boldsymbol{o}'_1, \boldsymbol{o}'_2, \boldsymbol{o}'_3\}$ が得られ，それぞれ Object Feature $\mathcal{O}'' = \{\boldsymbol{o}''_1, \boldsymbol{o}''_2, \boldsymbol{o}''_3\}$，BBox Feature $\mathcal{B}'' = \{\boldsymbol{b}''_1, \boldsymbol{b}''_2, \boldsymbol{b}''_3\}$，Target Feature $\mathcal{T}'' = \{\boldsymbol{t}''_1, \boldsymbol{t}''_2, \boldsymbol{t}''_3\}$ が生成される．そして，歩行者から生成された特徴量 $\boldsymbol{o}''_1, \boldsymbol{b}''_1, \boldsymbol{t}''_1$ に対し，それぞれ同じ Positinal Embedding $\mathfrak{p}_1$ を加算する．さらに，$\boldsymbol{o}''_1, \boldsymbol{o}''_2, \boldsymbol{o}''_3$ に Object Feature であることを示す $\mathfrak{s}_o$ を，$\boldsymbol{b}''_1, \boldsymbol{b}''_2, \boldsymbol{b}''_3$ に BBox Feature であることを示す $\mathfrak{s}_b$ を，$\boldsymbol{t}''_1, \boldsymbol{t}''_2, \boldsymbol{t}''_3$ に Target Feature であることを示す $\mathfrak{s}_t$ の Segment Embedding をそれぞれ加算して，TE へ入力特徴量する．

TE は Self Attention モジュールによって，これらの特徴量を相互に考慮して変換した新たな特徴量 $\tilde{\mathcal{O}}, \tilde{\mathcal{B}}, \tilde{\mathcal{T}}$

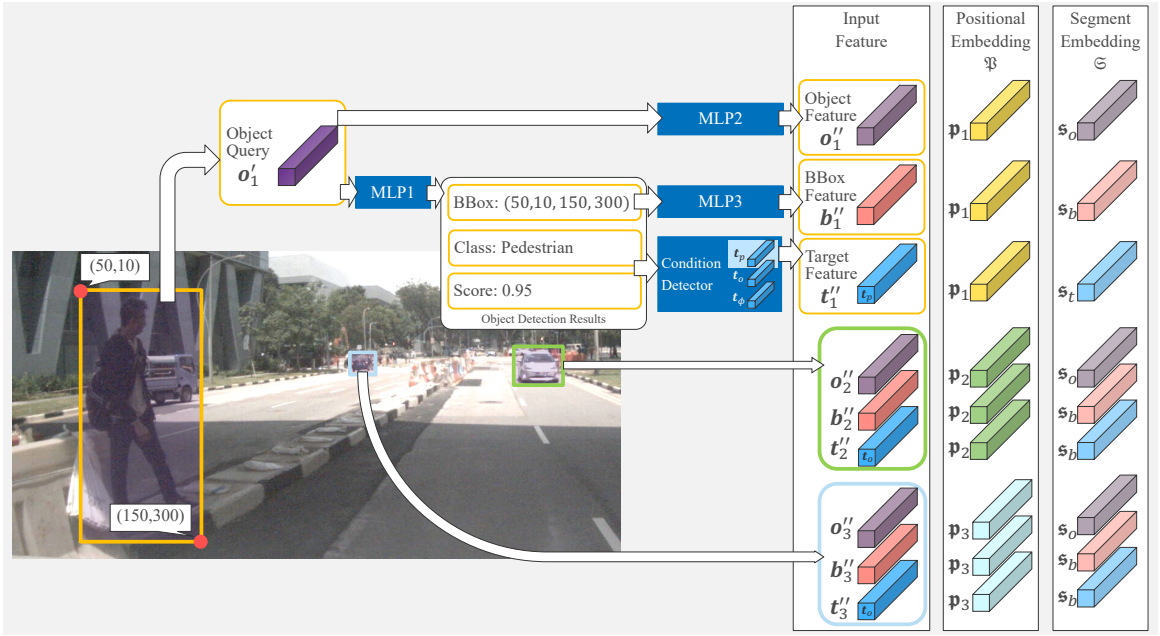

図 4　Transformer Encoder の入力例

を出力する．

$$\begin{pmatrix} \widetilde{\mathcal{O}} \\ \widetilde{\mathcal{B}} \\ \widetilde{\mathcal{T}} \end{pmatrix} = \mathrm{TE}\left(\begin{pmatrix} \mathcal{O}'' \\ \mathcal{B}'' \\ \mathcal{T}'' \end{pmatrix} + \mathfrak{P} + \mathfrak{S} \right) \tag{8}$$

そして，$\widetilde{\boldsymbol{o}}_i \in \widetilde{\mathcal{O}}$ を MLP_4 へ入力し，各物体に対応する PEGO スコア $\widehat{\boldsymbol{y}} = \{\widehat{y}_1, \widehat{y}_2, \ldots, \widehat{y}_N | \widehat{y} \in \mathbb{R}\}$ を次式のようにして得る．

$$\widehat{y}_i = \mathrm{MLP}_4(\widetilde{\boldsymbol{o}}_i) \tag{9}$$

Label Generator　Label Generator は Projection Layer の学習に用いる教師信号の PEGO ラベル $y \in \mathbb{R}$ を生成する．PEGO ラベル y には注視対象物のインデックス i を次式の条件で割り当てる．

$$\begin{aligned} &(sx'_i \leq lx \leq ex'_i) \wedge (sy'_i \leq ly \leq ey'_i) \wedge (s' > \delta) \\ :=&\{\boldsymbol{l} = (lx, ly), \boldsymbol{b}'_i = (sx'_i, sy'_i, ex'_i, ey'_i), \boldsymbol{b}'_i \in \mathcal{B}'\} \end{aligned} \tag{10}$$

最後に，Projection Layer が出力する各物体の PEGO スコア $\widehat{\boldsymbol{y}}$ と Label Generator が生成する PEGO ラベル y の間の Cross Entropy を計算した損失を用いる．ここで，1 つのシーン画像内に複数の歩行者が存在する場合に，本来は注視対象物が異なる歩行者であっても同じ注視対象物を推定する問題が生じる可能性がある．なぜなら同じシーン画像の場合，歩行者ごとに異なる入力は Target Feature $\mathcal{T}''$ のみであり，その他の入力が同一なためである．そこで 式 (11) に示すように，シーン画像内の対象歩行者数 n を Cross Entropy の計算結果に乗じることでこの問題の解決を図る．複数の歩行者が含まれるシーンは各歩行者ごとに正しく注視対象物を推定しないと，通常よりも大きな損失が生じる．これにより，シーン画像単位で注視されやすい物体を推定するのではなく，各歩行者単位で注視対象物の推定を目指す．ここで $\mathbb{1}_{(i=y)}$ は，$i = y$ のときに 1，それ以外で 0 を出力する One-hot ベクトルである．

$$\mathcal{L} = -n \cdot \sum_{i}^{N} \mathbb{1}_{(i=y)} \log \widehat{y}_i \tag{11}$$

4　実験

訓練済みの PEGO Transformer V2 を用いて，歩行者注視対象物推定を行なった．また PEGO Transformer V2 の有効性を示すため，歩行者注視対象物推定タスク用に調整した既存手法と比較した．以下で実験設定について詳しく述べる．

我々が構築した PEGO データセット [8] を用いて各手法を訓練した．1,157 人の歩行者の 8 割を訓練データ，2 割を検証データとして用いた．訓練データに対しては入力画像の左右反転とランダムクロップによるデータ拡張をした．なお，PEGO データセットは 3 人のアノテータによる注視点が存在し，複数の注視対象物候補

が存在する場合がある．学習データセットでは各注視対象物ごとにデータを分割し，一つのシーン画像につき注視対象物を一つに限定する．同じシーンに存在していたその他の注視対象物は，注視対象物推定の候補から除外している．ただし PEGO Transformer の場合は，入力サイズが同一でないと訓練ができないため，対象外の注視対象物に対応する Object Feature $\boldsymbol{o}''$，BBox Feature $\boldsymbol{b}''$ には 式 (5)， 式 (6) の処理を行なわずにランダムな値を設定し，Target Feature $\boldsymbol{t}'' = \boldsymbol{t}_\phi$ とした．検証時には全ての注視対象物を含み，どの注視対象物を選択しても正解と判定している．

PEGO Transformer V2 の学習時には，Projection Layer のパラメータのみを更新対象とした．Feature Extractor と Deformable Transformer は，Deformable DETR [12] で訓練済みの重みを用いて初期化し，使用するオブジェクトクエリの数は $N = 40$ とした．検出尤度スコア s' のしきい値 δ は 0.3 に設定した．

比較手法には GazeFollow [2] を参考にした視線ベースの手法と，Wang らの GaTector [5] を用いた．視線ベースの手法は，まず事前学習済みの骨格検出器 [1] を用いて，歩行者の頭部に対応する特徴点座標 (右目，左目，鼻) を得る．次にシーン画像，切り出した頭部画像，顔の中心座標（鼻の座標）を CNN と MLP で構成されたモデルに入力し，歩行者の視線の方向を推定する．視線方向の教師信号は，PEGO データセットで歩行者の目の位置と注視点を結ぶ直線の角度を用いる．最後に，推定した視線上で歩行者の目の位置に近い物体から順に注視対象物とする．

GaTector は，シーン画像，学習済みの骨格検出器の結果を用いて切り出した頭部画像，顔の中心座標（鼻の座標）を入力に用いる．実験では，GOO データセット [6] で事前学習済みの GaTector を，PEGO データセットでファインチューニングしたモデルを用いた．

5　実験結果および考察

図 5 に歩行者注視対象物推定の精度を示す．各注視対象物候補の PEGO スコア $\hat{\boldsymbol{y}}$ の最大値に対応するインデックスが，PEGO ラベル y と一致するかを Top1 正解率として評価に用いた．また推定した PEGO スコアの高い順に上位 2 位内，3 位内，4 位内，5 位内までに正解が含まれるかどうかを評価した結果を Top2，Top3，Top4，Top5 正解率とし，あわせて示している．さらに文献 [8] に対する改善を示すため，BBox Feature $\mathcal{B}''$ と Target Feature $\mathcal{T}''$ を用いずに推定した結果を PEGO Transformer V1 の名前で示す．また， 図 6 に PEGO Transformer V2 による歩行者注視対象物推定結果を示す．

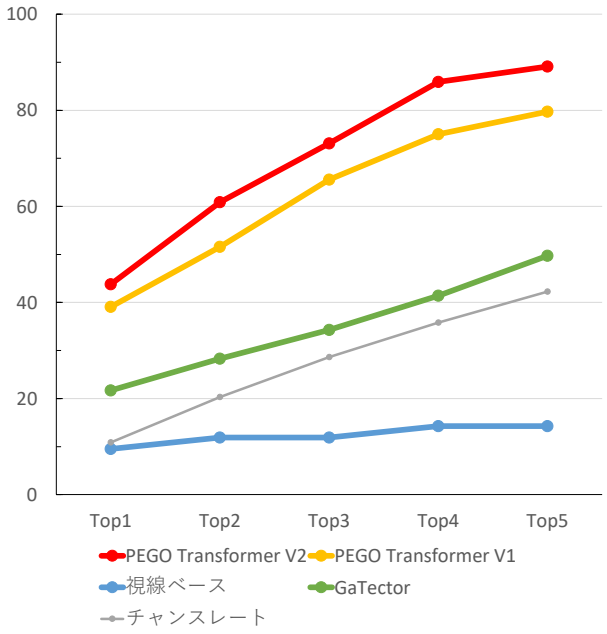

図 5　PEGO 推定の精度

PEGO Transformer V2 は比較手法や改善前の PEGO Transformer V1 と比べて高い性能を発揮した．まず視線に基づく方法では，正しく視線を推定することが困難であった．これは視線の推定に用いる頭部画像の解像度が低く，推定に必要な情報が十分に獲得できなかったためだと考えられる．また視線を正しく推定できた場合でも，視線の近くに複数の物体が存在する場合は注視対象物を推定することが困難であった．特に 図 6 (b) に示すように，正解の注視対象物と歩行者を結ぶ視線上に別の物体（図の例では別の歩行者）が存在する場合，その物体が注視対象物に選ばれるため，正しく推定ができなかった．一方，提案する PEGO Transformer V2 は歩行者と注視対象物が離れた位置に存在し，その間に別の物体が存在しても正しく推定ができた．次に，GaTector は注視対象物を推定するために高解像度の頭部画像を必要とするが，交通シーンでは対象歩行者の距離が遠くなるため，そのような画像を得ることは困難である．そのため，GaTector の性能は PEGO

[1] https://mmpose.readthedocs.io/en/latest/model_zoo/body_2d_keypoint.html#topdown-heatmap-swin-on-coco

(a) 検出成功：右側の歩行者は中央の車両を注視

(b) 検出成功：右側の歩行者は対向車両を注視

(c) 検出成功：手前の歩行者は奥の車両を注視
((d) と同じシーン画像)

(d) 検出成功：奥の歩行者は手前の車両を注視
((c) と同じシーン画像)

図 6　PEGO Transformer V2 の PEGO 推定結果：黄枠が対象歩行者，赤枠が推定結果の PEGO，赤点がアノテータがクリックした注視点を示す

Transformer V2 より低い結果となった．また注視対象物の推定に用いる視野ヒートマップの推定が不十分であることも確認した．これについては，視野ヒートマップの推定精度が注視対象物推定の性能のボトルネックになるという Wang らの考察とも一致する．さらに，PEGO Transformer V1 は一つのシーン画像に複数の歩行者が存在する場合，各歩行者についてそれぞれ異なる注視対象物を推定することが困難である．一方提案する PEGO Transformer は 図 6 (c) と (d) に示すように，同じシーン画像であっても歩行者毎に異なる注視対象物推定が可能である．これは Target Feature $\mathcal{T}''$ で明示的に対象歩行者を示すとともに，シーン画像内の歩行者数に応じて損失を調整するように変更したことが貢献していると考えられる．

最後に，PEGO Transformer V2 が歩行者の真の注視対象物を推定できているかを確認するシナリオ実験を行なった．PEGO データセットに収録されている注視対象物は，第三者のアノテータが判断した主観的な指標である．そこで，シナリオ実験では歩行者が物体を注視するシーンを撮影し，歩行者自らが注視対象物を回答することでラベル付けをした．さらに，人間の推定精度を評価するため，27 枚のシーン画像を対象に 3 人のアノテータが注視対象物をラベル付けした．そして，PEGO データセットで事前学習済みの PEGO Transformer V2 で注視対象物の推定を行ない，人間の推定精度と比較した．PEGO Transformer V2 の結果例を 図 7 に示す．図中の黄枠が対象歩行者，赤枠が推定結果の PEGO，黄点が実験歩行者がクリックした真の注視点，赤点がアノテータがクリックした注視点を示す．人間の Top1 正解率が 55.6 %に対し，PEGO Transformer V2 も 55.6 %となることを確認し，人間と同等の推定精度が得られることを確認した．

6　むすび

本発表では，交通シーン中の各歩行者の注視対象物を推定する PEGO Transformer V2 を提案した．我々の提案する PEGO Transformer V2 は，従来の注視対象物推定手法で用いられている高解像度の頭部画像や視線ヒートマップなしに歩行者の注視対象物を推定することができる．今後の課題として，シーン画像の時

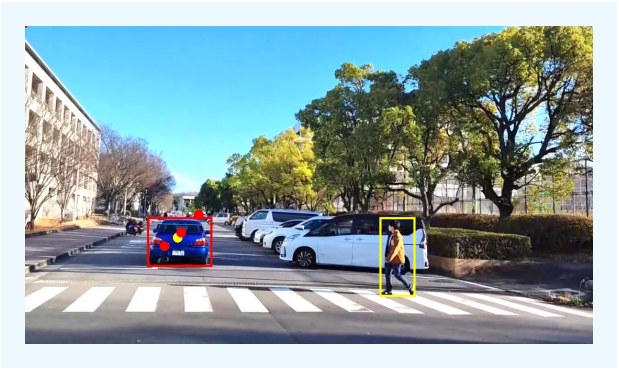

図 7　シナリオ実験における PEGO Transformer V2 の結果例

系列情報の活用や訓練データに存在しない物体クラスでも推定可能なオープンワールド手法の開発が挙げられる．

謝辞　本研究の一部は JSPS 科研費 23H03474 による．本研究の一部は名古屋大学のスーパーコンピュータ「不老」の一般利用にて実施した．

参考文献

[1] 畑隆聖，出口大輔，平山高嗣，川西康友，村瀬洋：“Eye-contact transformer: シーンコンテキストを考慮した遠方歩行者のアイコンタクト検出”，電子情報通信学会論文誌，Vol.J107-D，No.04，2024，(印刷中)．

[2] A. Recasens, A. Khosla, C. Vondrick, A. Torralba: “Where are they looking?”, in *NIPS*, Vol.28, pp.199–207, 2015.

[3] D. Lian, Z. Yu, S. Gao: “Believe it or not, we know what you are looking at!”, in *ACCV*, pp.35–50, 2018.

[4] E. Chong, N. Ruiz, Y. Wang, Y. Zhang, A. Rozga, J. M. Rehg: “Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency”, in *ECCV*, pp.383–398, 2018.

[5] B. Wang, T. Hu, B. Li, X. Chen, Z. Zhang: “GaTector: A unified framework for gaze object prediction”, in *CVPR*, pp.19 588–19 597, 2022.

[6] H. Tomas, M. Reyes, R. Dionido, M. Ty, J. Mirando, J. Casimiro, R. Atienza, R. Guinto: “GOO: A dataset for gaze object prediction in retail environments”, in *CVPR Workshop*, pp.3119–3127, 2021.

[7] A. Rasouli, I. Kotseruba, T. Kunic, J. Tsotsos: “PIE: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction”, in *ICCV*, pp.6261–6270, 2019.

[8] H. Murakami, D. Deguchi, T. Hirayama, Y. Kawanishi, H. Murase: “Pedestrian’s gaze object detection in traffic scene”, in *VISAPP*, 2024, (in press).

[9] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, O. Beijbom: “nuScenes: A multimodal dataset for autonomous driving”, in *CVPR*, pp.11 618–11 628, 2020.

[10] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, D. Anguelov: “Scalability in perception for autonomous driving: Waymo Open Dataset”, in *CVPR*, pp.2443–2451, 2020.

[11] K. He, X. Zhang, S. Ren, J. Sun: “Deep residual learning for image recognition”, in *CVPR*, pp.770–778, 2016.

[12] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai: “Deformable DETR: Deformable transformers for end-to-end object detection”, in *ICLR*, 2021.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova: “Bert: Pre-training of deep bidirectional transformers for language understanding”, in *NAACL*, pp.4171–4186, 2019.