

疑似ラベルを用いた自動運転のための遠赤外線画像からの物体検出

B4 加藤 達也

1 研究背景および目的

- 背景： 完全自動運転の実用化に向けて技術の開発が進められており、その為に車載カメラ画像からの物体検出は重要な要素技術である。可視光画像からの物体検出は天候や時間帯によって精度が低下するので、その解決策として遠赤外線からの物体検出手法を考える。
- 課題:遠赤外線画像のデータセットは可視光画像のデータセットと比較して数が少ない。
- 目的： 遠赤外線画像を入力として低照度下でも安定的に動作する検出モデルを構築する。また、RGB 画像に適応して得た検出領域を教師とするドメイン適応を用いて、遠赤外線領域における検出モデルを構築する。

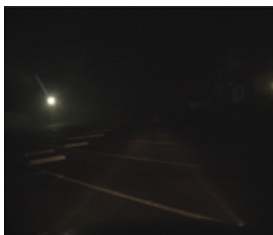


図 1: RGB 画像



図 2: FIR 画像

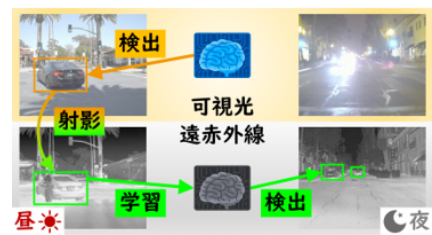


図 3: ドメイン適応の流れ

2 これまでの研究のまとめ

- 谷本先輩の最終的な提案手法は以下の通りである。流れとしては図 4(a) の通りである。
 1. RGB 画像に対して高精度な可視光モデルを用いて物体検出を行い、その結果を FIR 画像に変換して疑似ラベルを生成する。
 2. 可視光モデルの出力信頼度に基づき、高信頼の検出結果のみを疑似ラベルとして使用し、誤学習を防止する。
 3. FIR 画像と疑似ラベルを用いて、可視光モデルをファインチューニングし、FIR 画像専用の物体検出モデルを構築する。

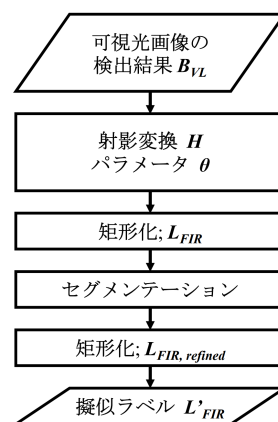


図 4: 疑似ラベル作成までの流れ

- 損失関数のカスタマイズを行った。採用した損失関数は FocalLoss と L1Loss。
- クラス分類とオブジェクト性に FocalLoss、バウンディングボックス回帰に SmoothL1Loss を使用した。

- 現在、サイズが小さいオブジェクトに対しての精度が低いため、難しいサンプルに着目し、分類性能を向上させる FocalLoss を採用した。
- SmoothL1Loss に関しては YOLOX で中心座標と幅、高さの回帰に対して L1 距離を使うことが多いため、採用した。

Category	mAP	mAP_50	mAP_75	mAP_s	mAP_m	mAP_l
person	0.039	0.194	0.001	0.039	0.079	0.087
car	0.163	0.490	0.055	0.039	0.322	0.644

表 1: 各カテゴリの検出精度 (mAP)

3 前回の GM からの進捗

3.1 損失関数の実装について

- config ファイルは base ファイルに上書きや追加をすることによってカスタマイズをする構造になっているが、改めて base ファイルの方を参照して見たところ、既に損失関数は実装されていた。
- bbox には IoULoss、クラス分類とオブジェクト性に CrossEntropyLoss が使用されていた。
- この 2 つとも、Yolox に最初から実装されている損失関数であったため、デフォルトの損失関数から他のカスタマイズされた損失関数に変更することによってどのように検出精度が向上するかを考えるべきであると思った。

3.2 アノテーションファイルの改善検討

項目	値
base_lr	1.0000×10^{-2}
lr	1.0000×10^{-2}
eta	2:10:43
time	0.2130
data.time	0.0158
memory	11174
loss	0.0021
loss_cls	0.0000
loss_bbox	0.0021
loss_obj	0.0000

表 2: mmengine の学習ログ (Epoch 169, iteration 250)

- 表 2 の loss_cls、loss_obj の値が 0 となっている。
- 原因不明なので、損失関数のパラメータを変更して変化するかを確かめたい。

3.3 クラス数の訂正

- いままでコード内でクラス数が 80 と記述されていた部分があったが、2 に統一したところ、検出精度の改善が見られた。(実験の最初の部分のみの切り取り)

表 3: 各クラスにおける性能比較 (旧モデル vs 新モデル)

クラス	指標	以前 (旧)	今回 (新)
person	mAP	0.034	0.017
	mAP_50	0.126	0.063
car	mAP	0.000	0.066
	mAP_50	0.000	0.221

3.4 ITS ミーティングについて

- 愛知工科大学の久徳先生との毎週月曜日に行われる ITS ミーティングにおいて、アドバイスをいただいた。
- クラス数が 80 であった部分は学習時点では 80 で、最後のファインチューニング時にクラス数を 2 に絞ることを意図として谷本先輩が設定していた。
- 損失関数については既存の損失関数はそれぞれ想定された使い方が存在するので、それに従って適した損失関数を各所の実装する。
- 既存の損失関数でも良いが、谷本先輩は独自の計算での損失関数の作成を目指していた。

4 今後の課題&スケジュール

- FLIR のバージョンアップをする。
- 損失関数に関する理解を深め、より良い検出精度を目指す。できれば独自の計算で損失関数を作成したい。