	テーマ番号	2EP99					
	プロジェクト テーマ	和文 機材	機械学習用いた電車の車両タイプの判別システムの開発				
		英文 A (Construction Method of Optimum Integer-to-integer Transform ed on an Error Propagation Model	指導教員	鷹合 大輔 准教授		
	プロジェク	١	4EP1 68 野崎 悠渡 (NOZAKI Yuto) 4EP5-11 松永 久秀 (MATSUNAGA Hisahide)				
	メンバー		4EP5-29 筒井 順慶 (TSUTSUI Junkei) 4EP5-100 百地 丹波 (MOMOCHI Tanba)				

Abstract Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. (何が問題で、それをどんな手法で取り組んで、どういう結果であったかなどを英語で要約して下さい) Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here. Describe about 5 lines of abstract in English here.

Keywords Qwerty1, Qwerty2, Qwerty3, Qwerty4, Qwerty5

1. はじめに

電車の車両タイプは JR の在来線だけでも 100 種類近く存在している. 多くの人は電車を見て電車だと認識することは可能だが、その電車の車両タイプまでを判断できる人は少ない. 電車についての知識がある人は一目見るだけでその電車の車両タイプを判断できるが、大多数の人は似ている電車の車両タイプを判断することが難しい. 簡単に画像や動画に写っている車両が何なのかを判別できるようになりたい

2. システム概要

本プロジェクトで開発するシステム概要を図1に 示す.

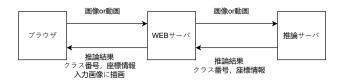


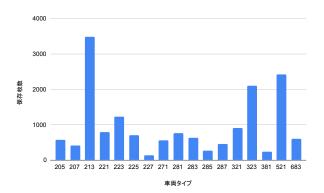
図 1 ああああ Fig.1 aaaaa

3. 判別モデルの開発の流れ

電車が写っている画像を集めて,データセットを作成し,学習をするという流れで判別モデルを開発する.

3.1 データ収集

YouTube で特定の電車のみが映っている $1 \sim 3$ 種類の動画を保存して、指定した枚数のランダムなフレームを保存する.保存した画像を識別して、電車が映っているものだけ保存する.動画ごとに電車が写っている時間が異なるため、集めた画像は車両タイプごとに異なる.保存した画像の枚数を表 3.1 に示す.



3.2 データセットの作成

本プロジェクトで作成するモデルは識別モデル,分類 モデルの二種類である.分類モデルのデータセット内の 画像をアノテーションして識別モデル用のデータセット を作成した.

3.2.1 分類とは

分類はデータやオブジェクトを異なるクラスやカテゴリに分けるプロセスを指す. 画像の分類とは, 画像が特定のカテゴリやクラスに属するかどうかを判別する作業である. 例えば, 画像に写っているのが猫か犬か, 車か飛行機かなどのクラスに分類することがある.

3.2.2 識別とは

識別とは画像のどこに何が写っているのかを判断する プロセスを指す. 1枚の画像に複数の物体が写っていて も識別はできる.

3.2.3 アノテーションとは

アノテーションとは、機械学習の分類の一つである教師あり学習において、分析対象データにラベルを付与するプロセスである。画像にバウンディングボックスと呼ばれる四角形を描画しクラス番号を指定する。バウンディングボックスを描画することでその画像に写っている物体の座標情報を取得することができる。クラス番号とは、判別したいものリストを作成し、画像に写っている物体に対応した、リストのインデックスのことである。アノテーションをした結果は、識別モデルの学習時に使用する。

3.3 モデルの学習

YOLOv8 を用いてモデルを作成した

3.3.1 YOLOv8 の概要

YOLO とは You Only Look Once の略で,人間のように一目見るだけで物体検出ができることを指している.データセットを作成し学習させることで,任意の物体のみ検出させることが可能である.YOLOv8 は YOLO シリーズの最新バージョンであり,ディープラーニングとコンピュータビジョンの最先端の進歩に基づいており,速度と精度の麺で比類のない性能を提供している.

3.3.2 学習の実行

学習を進めると徐々に性能が上がっていき性能が向上 しなくなると学習が途中で中断される. 学習は中断され るまで続けたため, 学習回数はモデルによって差がある.

3.4 モデルの使い方

識別モデルと分類モデルをどのように使えば結果が出力されるのかを説明する. python のコードを貼り付ける? 田村のパートに記載してもいいかもしれない

4. 作成したモデルの評価

4.1 分類モデルの性能評価

作成した分類モデルを用いてテストデータセットの分類を行った結果を図2に示す.

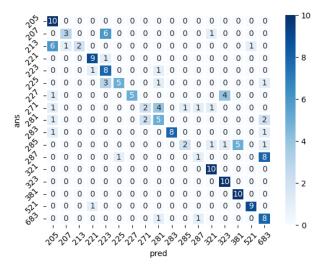


図 2 性能評価 Fig.2 seinouhyouka

4.2 識別モデルの性能評価

作成した識別モデルを用いてテストデータセットの識別を行った結果を表 1 に示す

表 1 表その 1 Table 1 Write a table-caption here in English

Table 1	Write a table-caption here in English				
車両タイプ	識別成功数	識別失敗数	識別不能数		
205	8	1	1		
207	4	2	4		
213	2	7	1		
221	10	0	0		
223	8	1	1		
225	1	7	2		
227	4	4	2		
271	5	4	1		
281	2	8	0		
283	7	0	3		
285	0	8	2		
287	5	5	0		
321	8	1	1		
323	9	0	1		
381	8	1	1		
521	9	1	0		
683	7	3	0		

5. 考察

正解率は電車によって異なることがわかる.特に外見が似ている電車だと、誤判別していることが多かった.

各車両タイプの画像の枚数に差があったことが判別結果に影響を与えていると考えられる. 画像の枚数が少ない車両タイプの判別結果が必ず悪くはならなかった. 画像の枚数が判別結果に影響を与えるのではなく, 車体の

特徴が鮮明に写っている画像の枚数が判別結果に影響を与えると考えられる。データセットとして質の悪い画像を大量に集めるのではなく、車体の特徴が鮮明に写っている画像を車両タイプごとに集める必要があったと考えられる。

6. システムについて

6.1 システムの動作要件

本システムでは、ノートパソコンに Linux サーバのセットアップを行い、サーバとして使用した. ネットワークは、学内の固定 IP を割り当ててもらい、学内ネットワークを使用した. サーバのスペックは以下の通りである.

6.1.1 サーバのスペック

- OS: Ubuntu 22.04.3 LTS
- メモリ: 4GB
- CPU: Intel® Core(TM) i7
- GPU: NVIDIA GeForce GTX 470M
- HDD: 80GB

6.2 システムの開発環境

本システムの開発には、以下の言語やツールを使用した.

6.2.1 使用言語

- フロントエンド:
 - html
 - JavaScript
 - css
- バックエンド:
 - JavaScript (Node.js)
 - Python

6.2.2 使用ツール

- エディタ: VSCode
- ブラウザ: Google Chrome 119

6.3 システムの機能詳細

本システムは、以下の3つの機能を提供する.

6.3.1 電車の画像の分類

この機能では、ユーザがブラウザからアップロードした画像に含まれる電車の種類を分類する.分類の結果、最も可能性が高いものを HTML ページに出力する.

6.3.2 電車の画像の識別

この機能では、ユーザがブラウザからアップロードした画像に含まれる電車の位置と種類を識別する. 識別の結果、バウンディングボックスが追加された画像をHTMLページに表示する.

6.3.3 電車の動画の識別

この機能では、ユーザがブラウザからアップロードした動画に含まれる電車の位置と種類を識別する. 識別の結果、バウンディングボックスが追加された動画をHTMLページに表示する.

7. システムの考察

8. まとめ