IPA

2022 年度 未踏 IT 人材発掘・育成事業 採択案件評価書

1. 担当 PM

田中 邦裕（さくらインターネット株式会社　代表取締役社長）

2. クリエータ氏名

長沢 瑛史（静岡大学大学院総合科学技術研究科 工学専攻）

3. 委託金支払額

2,736,000 円

4. テーマ名

リアルタイムな動画内物体認識技術を用いた物探しシステム

5. 関連 Web サイト

FineTop：https://fine-top.sakura.ne.jp/

6. テーマ概要

スーパーなどの店舗で買い物をする際、買いたい商品がなかなか見つからないことがある。既にその商品が視界に入っていてもその存在に気付けないことも多く、時には広い店内を探し続けて時間を浪費してしまうこともある。

そこで本プロジェクトでは、ユーザの探したい商品をリアルタイムに探し出すスマートフォンアプリケーション（以降、「アプリ」）を開発する。ユーザはまず、探したい商品の画像をインターネットから取得してアプリに登録する。すると、アプリはカメラを通じて周囲を観察し始める。ユーザが店内を移動し、登録しておいた商品がカメラに映ると、アプリは即座にそのことを認識し、商品の存在をユーザに通知する。本アプリを利用することで、ユーザは探したい商品を見落とすことなく素早く発見できるようになる。また予め複数の商品を登録しておき、本アプリに購入の有無を管理させることで、買い忘れを防止するといった使い方もできる。

本アプリは商品に限らず、様々な物体の探索に応用可能である。例えば、道路標識を登録しておけば運転者が標識を見落とすのを防ぐことができ、自動車の安全運転支援として応用できる。このようにユーザがより豊かな生活を送ることができるようなアプリを開発することが本プロジェクトの目的である。

本プロジェクト実現に向けての技術的なチャレンジは、登録した商品がスマートフォンのカメラに映った時に、そのことを正しく認識する技術を開発することである。置かれる場所、角度、照明などによって物体の見え方は様々に変化する。そのような変化に対して頑健な物体認識技術を深層学習によって構築する。

7. 採択理由

本プロジェクトでは、カメラをかざすだけで、探し物が画面上でハイライトされるという、探し物が苦手な人間にとって画期的なプロダクトを作ることを目標としている。

ディープラーニングの登場によって、画像認識の精度は飛躍的に向上し、様々な産業分野で利用されるようになったが、人間を補完する機能がアプリとして提供されることも増えており、人の目を拡張し QoL を向上させるプロダクトとして、とても有用性の高いものの一つになると考える。

手法としては、探している風景の画像と、探し物の画像を、ResNet50 に入力させ、それぞれ 2048 次元の特徴ベクトルに変換し、それを 10 層の結合層を通して、0 か 1 が出力されるモデルを構築し、風景画像に探し物の画像があれば 1 と出力するような学習を繰り返す。

これにより、新しい探し物の画像があったとしても、その画像自体を事前学習させる事なく、探したいものをすぐに探せるという UI/UX を実現することが特徴的である。

もちろん、汎用的なデータセットをもとに、事前学習していない画像を探し出すことが果たして精度良くできるのかが課題ではあるが、それが未踏性のある部分でもあり、既にある程度の実装を通じて動くシステムも作り始めており、実現可能性は充分にあると判断して採択することとした。

8. 開発目標

本プロジェクトは、ユーザが素早く目的の物体を見つけることができる、使いやすく汎用的な Web アプリケーションを開発することを開発目標とした。もともとクリエータの研究により、2 つの画像の中に同じ物体が存在することを検出するモデルを完成させており、探したい商品画像をもとにして物体検出をすることは可能であったが、スマートフォンなどを使って外出先で利用するにはアプリケーション開発が必須であった。

本プロジェクトでは商品探しに焦点を置き、まずアプリ内に実装したインターネット上の画像検索機能において、探したい商品の画像を検索し、次にスマートフォンを周囲に向けて撮影したカメラ映像からリアルタイムに商品を認識し、ユーザに通知するという機能を提供することを目指した。

アプリケーションの開発にあたっては、スマートフォンにおいて稼働可能な

ようにモデルの軽量化が必要となり、かつブラウザで動作させられるように、JavaScript をベースとした実装が必要となる。

モデルの軽量化にあたっては、GHostNet を利用して特徴マップの生成方法を変更して計算量の低減を行ったほか、モデルの分割を行って推論時間の改善を進めた。ブラウザで動作させるための実装としては、TensorFlow.js を使用することで端末での推論を実現し、誰でも利用可能な Web アプリケーションを実装し、加えてユーザビリティとプラットフォームの互換性を確保するために、洗練されたデザインと使いやすい UI/UX の開発を行った。

9. 進捗概要

本プロジェクトにおいては、スマートフォンで動作可能な Web アプリケーションの開発と、スマートフォン上に搭載できる軽量なモデルの作成の 2 点を中心に行なった。

まず Web アプリケーションであるが、Python で書かれた、TensorFlow または CNTK、Theano 上で実行可能な高水準のニューラルネットワークライブラリである Keras を用いて実装したモデルを、TensorFlow.js 上で動作させることで、JavaScript をベースにブラウザ上で稼働するものとした。

動作時には、予めスマートフォンのメモリ上にモデルを読み込んでおき、スマートフォン上の画像フォルダから探したい画像を選択して読み込み、スマートフォンのカメラで撮影している画像との比較をリアルタイムに行う。アプリケーションは商品を認識するとページの背景色をオレンジ色にすることでユーザに視覚的にフィードバックを返し、また iPhone の Safari など一部環境を除きバイブレーションによるフィードバックも行う機能を有している。このアプリケーションにより、決してハイエンドと呼ばれるような最上位の性能を有していないスマートフォンでも推論を行い、物体を認識できることが確認できた。

ただ、課題として浮かびあがったのはその動作の重さである。JavaScript はシングルスレッドで動作する言語であり、これはすなわち推論を行っている間端末の画面を更新できないことを意味している。そのため、画面は 1 秒に 2-3 回程度しか更新されず、画像の選択もままならないためユーザ体験があまりよくないものとなっていた。更に、多くの人が利用していると考えられる Android 版 Chrome においてはモデルの読み込みにエラーが生じることが確認され、この問題の早期の解決に乗り出した。

アプリケーションの動作軽量化の為に考えられる解決方法の 1 つはマルチスレッド化である。JavaScript がシングルスレッドであるために画面の更新が追い付かなかったので、画面の描画と推論を別のスレッドで行えば重さを軽減できるのではないかと考えた（図 4）。JavaScript はシングルスレッドで動作するが、Thread.js という JavaScript を疑似的にマルチスレッド化してくれるライブラリが存在し、これを利用することでアプリケーションの動作を軽くできるの

ではないかと考えた。しかしこのライブラリは制約も多く、画面の描画と推論を完全に別のスレッドとして分離することは難しいということがわかったため、利用を断念した。

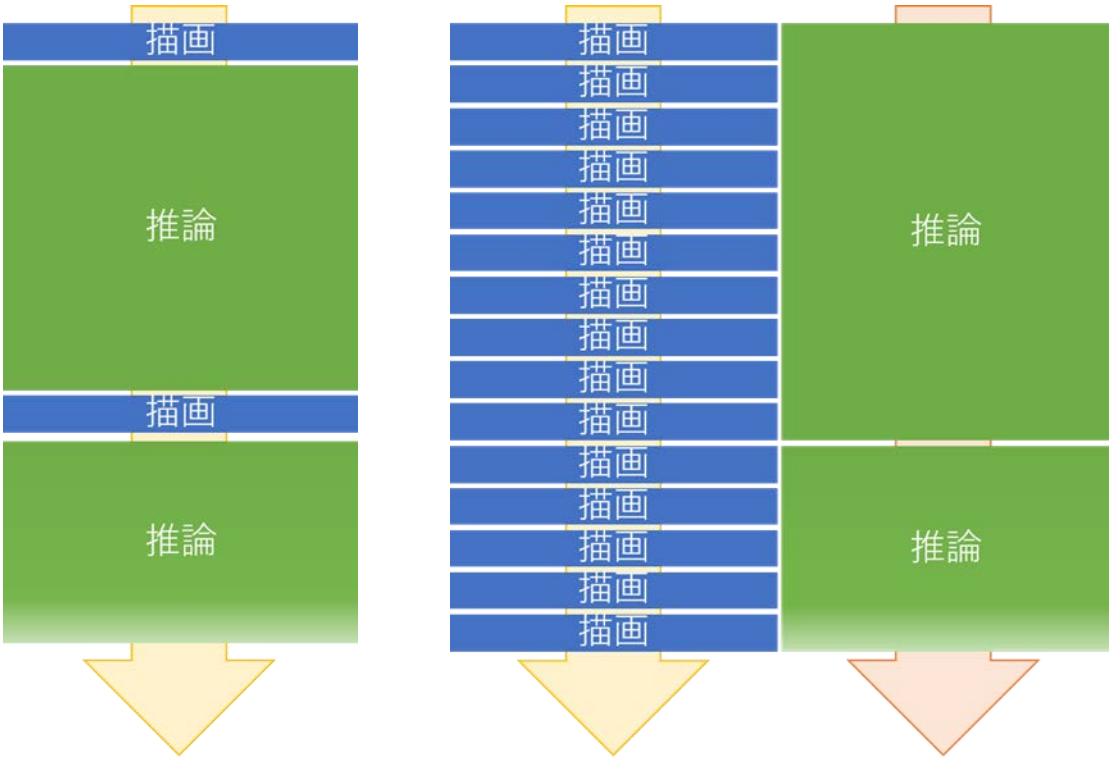

図 1：マルチスレッド化の概略図

続いて、モデルの軽量化によってアプリケーションの動作を軽量化することに挑戦した。推論の所要時間は基本的にはモデルが大きくなることに伴って長くなっていくことが考えられる。このモデルのレイヤの内、2 つの特徴量を掛け合わせる層と全結合層（計 10 層）のパラメータ数は全体のパラメータ数の 1％にも満たず、モデルの殆どのパラメータは特徴抽出器が占めていることがわかる。従って、2 つの特徴抽出器のパラメータ数を如何に減らすかがモデル軽量化の鍵となる（図 2）。

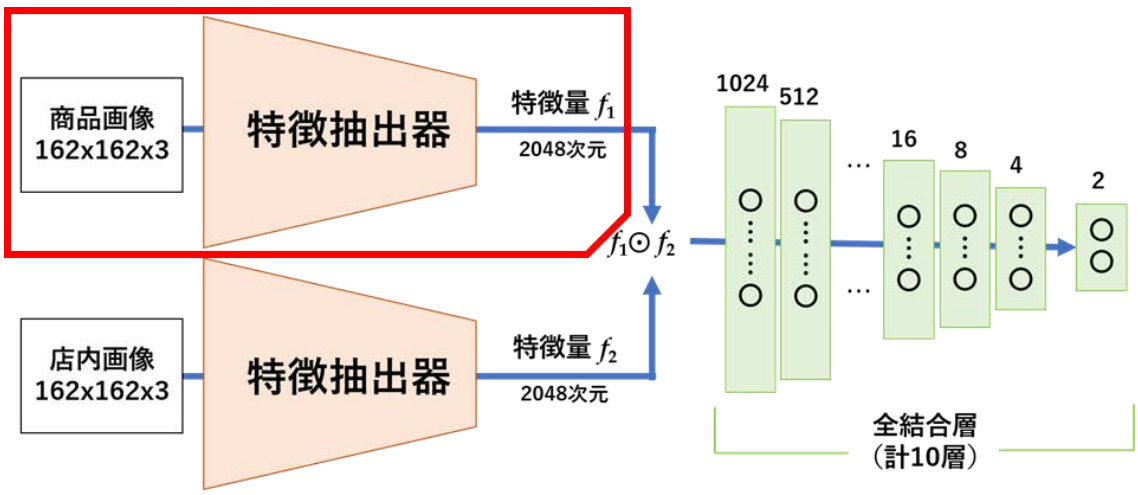

図 2：モデルの概略図

特徴抽出器の軽量化手法として GhostNet を採用することとした。当初特徴抽出器として用いていた ResNet の畳み込み操作を置き換えるだけで良いことが最大のメリットである。GhostNet に置き換えたモデルを学習させたところ、殆ど精度は変わらずに計算時間を半分にすることができた。またモデルを軽量化した後には Android 版 Chrome でのアプリケーションの動作も確認できた。GhostNet 以外にも Xception と MobileNetV2 の 2 つのネットワークをそれぞれ特徴抽出器として用いることができないかも検討した。GhostNet は独自に実装し学習を行ったが、Xception と MobileNetV2 は Keras の予め事前学習済みモデルとして用意されていたため、そちらを用いることとした。また事前学習済みモデルを用いて学習を行うことで学習時間の短縮と精度が向上する効果も見込むことができる。他にも多くの特徴抽出器が利用できる中 Xception と MobileNetV2 を試した理由は、どちらもスマートフォンのようなリソースが限られた端末で推論を行うことを目指して設計されたモデルであり、モデルサイズが小さくありながら高精度に分類できるモデルであることが示されており、また入力画像の解像度が自身の実装した GhostNet よりも大きくできたためである。しかし、実際に学習させてみたところ、学習自体は上手くいき、学習データに対しては高精度に認識を行うモデルを作ることができたものの、未知の画像データに対しては全く対応できないものとなってしまった。汎化性能を引き出すことができなかったため、これらを用いることは断念した。

ここまでの実装ではモデルにはフレーム毎に「商品画像」と「店内映像」の 2 つの特徴量を計算させ、これらを掛け合わせることで結果を得ていた。しかし商品画像の特徴量は商品画像が変わった時に一度計算すればよく、毎フレームで計算し直す必要はない。そこでモデルを「商品画像の特徴量を抽出するモデル」と「店内映像の特徴量を抽出すると共に（既に計算済みの）商品画像の特徴量を受領し店内映像の特徴量と掛け合わせて結果を得るモデル」の 2 つに分割し、

商品画像を選んだ時以外は前者の計算をせずに済むようにした。これにより推論時間は GhostNet と合わせて当初の 1/4 となった。

本プロジェクトでは明治の「きのこの山」「たけのこの里」を識別するタスクにも挑戦し、画像を用意して検証を行った。その結果、本システムでもこれらの商品を識別することができることが確認できた。

10. プロジェクト評価

このプロジェクトでは、スマートフォン上で動作する軽量な物体認識システムを実現することに成功した。GhostNet を導入し、量子化による軽量化も行い、スマートフォンでも高速に物体認識が行えるようになった。また、Web アプリケーションとして一般に公開することで、誰でも容易に利用可能となった。簡単な手順で使用できるため、事前準備や前提知識が必要ないため、幅広い層が使うことができる。

ただ、物体を認識するというベースの部分については、自らの研究で本プロジェクトの採択前には実現しており、実際に開発する部分は Web アプリケーションだけであって、難易度が高いと言えるものではなかった。

採択時には、探したいものをすぐに探せるという UI/UX を実現し、人の目を拡張し QOL を向上させるプロダクトとして世の中に出せるということまで期待していたが、結果としては Web アプリケーションの試作にとどまってしまったことは勿体なかったと感じる。

とはいえ、このプロジェクトの成果は、スマートフォン上で高速に物体認識ができる軽量なシステムの実現であり、幅広い用途に応用可能であるという点において高く評価される。今回は、Web アプリケーションとして広く一般に公開することはできなかったが、今後も改良を続けることで、より多くの人々に利用されるシステムとなることが期待される。

11. 今後の課題

今回のプロジェクトを通じて、自らの研究である物体認識を Web アプリケーション化でき、クリエータ自身のウェブ開発能力も高まったのではないかと思う。

それを生かして、実際に広く利用されるアプリケーションを作ることができれば、世の中に対して大きな価値を提供できるのではないかと考えられる。