

Python を用いた信号認識処理の組み込み実装と評価

1220009 池内 建太 (集積システム研究室)

(指導教員 密山 幸男 教授)

1. はじめに

本研究では、赤信号の認識処理を対象として、プロセッサと FPGA を統合したデバイスであるプログラマブル SoC と組み込み開発向け GPU について、処理性能と消費電力の観点で比較評価を行うことを目的とする。信号機認識アルゴリズムには HOG 特徴量計算を用いたランダムフォレストを用いた。主要な処理の実装に Python を用いることを条件として、3 種類の組み込みボードに実装した。

2. HOG 特徴量計算を用いたランダムフォレスト

HOG 特徴量とは局所画像特徴量であり、これを用いる事で物体の大まかな形状を表現できる。この HOG 特徴量を用いてランダムフォレスト法により信号認識処理(推論)を行う。ランダムフォレスト[1]は複数の決定木による推論結果の多数決を相関性の低い決定木から行う。決定木はノードで次々に質問することで分類する。

3. 信号認識処理フロー

図 1 に赤信号の認識処理の処理フローを示す。信号認識処理は、カメラから取得した画像の特定領域内を対象として赤画素数をカウントし、赤画素数が条件を満たせば HOG 特徴量計算とランダムフォレストによる信号機認識処理を行う。

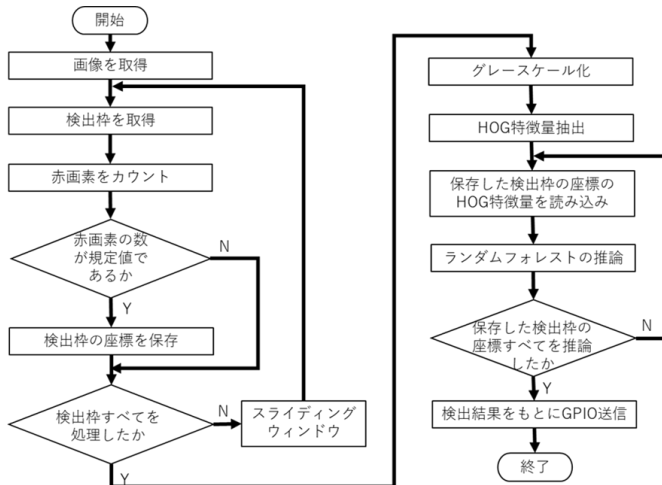


図 1: 信号認識処理フロー

4. ボードへの実装

4.1 開発環境

評価対象のボードは、プログラマブル SoC を搭載する Zybo Z7-20 と PYNQ-Z1、組み込み開発向け GPU を搭載する Jetson Nano Developer Kit を用いた。信号認識処理は Windows PC で開発し、コードは Python で記述した。

4.2 実装用 Python コード

実装する Python コードは USB ウェブカメラによる学習用画像の撮影、学習用画像から HOG 特徴量の計算、計算した HOG 特徴量からランダムフォレスト学習用データの生成、ランダムフォレストによる信号認識処理の 4 つで構成される。

4.3 各ボードへの実装

開発したコードは特に書き換える必要なく各ボードで動作可能である。カメラには USB ウェブカメラを使用するが、Jetson Nano については CMOS カメラを用いた場合の処理性能も計測し比較材料とするため、CMOS カメラを使用するようにコードを書き換える。これにはマルチメディアプレー

ムワークである Gstreamer を使用する。今回、信号認識処理は各ボードの CPU に実装する。表 1 に各ボードが搭載している CPU の情報を示す。

表 1: ボード別搭載 CPU 情報

ボード	動作周波数	コア数	プロセッサ
Zybo Z7-20	667MHz	2	Cortex-A9
PYNQ-Z1	650MHz	2	Cortex-A9
Jetson Nano	1,430MHz	4	Cortex-A57

5. 評価結果

USB ウェブカメラ(BUFFALO 社製 BSW50KM02BK)を 3 種類のボードで用いた場合と Raspberry Pi Camera V2 を Jetson Nano で用いた場合の計 4 条件で実験を行った。

5.1 信号認識処理性能

表 2 に実験条件毎の処理性能を示す。Zybo Z7-20 の処理性能は PYNQ-Z1 より優れている。最も処理性能に優れるのは Jetson Nano であった。Zybo Z7-20 の処理性能に対して USB カメラを用いた Jetson Nano の処理性能はおよそ 2.6 倍になっており、動作周波数とコア数の差が結果に出ていると言える。処理時間の内訳については赤画素カウント、特徴量計算では処理時間が 1/4 になっているのに対しランダムフォレストでは 1/2 になった。違いが生じた理由については検討が必要である。

表 2: 条件毎の処理性能

実験条件	赤画素カウント (ms)	HOG 特徴量計算 (ms)	ランダムフォレスト推論 (ms)	フレームレート (fps)
Zybo Z7-20	0.934 / 回	19.133	12.327 / 回	4.983
PYNQ-Z1	0.972 / 回	19.338	15.042 / 回	4.258
JetsonNano (USB camera)	0.212 / 回	4.588	5.427 / 回	12.929
JetsonNano (RasPi camera)	0.262 / 回	4.952	6.742 / 回	10.495

5.2 性能電力効率

表 3 に各条件での消費電力と性能電力効率を示す。Zybo Z7-20 と PYNQ-Z1 は同等のスペックを持つため同程度の消費電力を予想していたが、PYNQ-Z1 の方が約 1.12W 高い。消費電力は Zybo Z7-20 が最も小さいという結果となった。また、性能電力効率について評価すると、USB カメラを用いた Jetson Nano が最も優れており、次いで Zybo Z7-20 が優れるという結果となった。

表 3: 条件毎の消費電力

使用ボード	使用カメラ	消費電力	性能電力効率
Zybo Z7-20	USB ウェブカメラ	2,006 mW	2.48
PYNQ-Z1	USB ウェブカメラ	3,128 mW	1.36
Jetson Nano	USB ウェブカメラ	4,684 mW	2.76
Jetson Nano	Raspberry Pi Camera V2	4,515 mW	2.32

6. まとめ

主要な処理を Python で記述した信号認識処理を 3 種類のボードに実装し、処理性能と消費電力を測定した。実験結果から性能電力効率を評価した結果、Jetson Nano が最も性能電力効率が優れており、組み込み開発に適していると言える。

Zynq のプログラマブルロジックや Jetson Nano の GPU を用いて処理を実装すれば性能電力効率の向上が見込まれるが、今回は時間の都合で CPU への実装にとどまった。プログラマブルデバイスや GPU コアへの実装と評価は今後の課題とする。

参考文献

- [1] M. PAL, "Random forest classifier for remote sensing classification" International Journal of Remote Sensing Vol. 26, No. 1, 217-222, 2005