

画像処理技術の組み込み分野への応用に関する研究（第 1 報）

若生 進一* 富長 博* 大高 理秀*

1. はじめに

近年、施設のメンテナンスや防犯・セキュリティ等様々な分野で画像処理が注目されている。

一方で、開発に要求される仕様は、処理の高速化、機器の小型化等、これまでの汎用コンピュータによるソフトウェア処理のみでは、実現が困難になってきている。

少量多品種の製品開発に於いて増大する要求仕様を満たすデバイスとして内部回路の変更が可能であり、並列処理による高速化が見込める PLD に着目した。

PLD とは、内部回路を設計者が自由に変更することができる LSI で、これまでは初期コストの高い専用 LSI の仕様を決定する際の試作に用いられることが多かった。

現在は価格も下がり、内部回路を自由に変更できるメリットを生かしてそのまま製品に組み込まれることも少なくない。

PLD を用いることで、ソフトウェア処理で画像処理を扱う製品に付加価値を付与する可能性について研究を行った。

本研究は平成 19 年度～20 年度の 2 年間で特別電源所在県科学技術振興事業の補助金を受けている。

2. 目的

これまで汎用コンピュータや専用ボードを用いる事が多かった画像処理をより安価なデバイスで実現し、小型化するためのハードウェア実装技術及び画像処理アルゴリズムについて検討する。

3. 実験方法

3.1 ハードウェア実装実験

実験装置として、写真 1 のような CMOS カメラ、液晶モニター及び PLD 実装基板を各々接続した構成を用いて PLD によるハードウェア実装実験を行った。

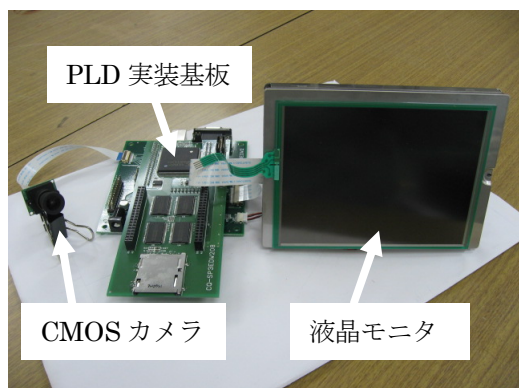


写真 1 実験装置

今回実験に使用した CMOS カメラは画像サイズ VGA (640×480 画素) で、画像フォーマットが YUV 形式で出力されるため、液晶モニタのアナログ RGB 出力にフォーマット形式を合わせる必要がある。

画像フォーマットには様々な形式があり、一般的に RGB の様に色の明るさを示す輝度情報で表す形式と、YUV の様に輝度情報と輝度情報に関連した色情報に分離して表す形式とがある。

RGB と YUV の関係式は以下のようになる。

$$\begin{aligned} R &= 1.164(Y-16) + 1.596(V-128) \\ G &= 1.164(Y-16) - 0.392(V-128) - 0.813(U-128) \\ B &= 1.164(Y-16) + 2.017(U-128) \end{aligned}$$

これらの演算を PLD 内部回路で行うためにシフト演算を用いて固定小数点演算になるように式を変換し、回路に組込んだ。

その他カメラインタフェース、表示インタフェース、メモリアンタフェースの信号処理回路を構築した。

全ての PLD 内部回路構築には、C 言語に類似した HDL と呼ばれるハードウェア記述言語を用いている。これまでの PLD 開発は、回路図で設計する構築方法が主だったが、HDL を用いる事で C のプログラムの様に抽象的に回路を構築することが可能となった。

写真 2 は実装した PLD を用いて CMOS カメラからの入力画像を液晶モニターに表示した結果である。

カメラの出力周波数に対して PLD のクロック周波数は 2 倍であるため、時分割処理を用いて画像情報の取得とモニタの表示をほぼ同時に行い、リアルタイム表示される事が確認できた。



写真 2 表示結果

次に、画像処理回路としてフレーム間差分処理回路を PLD 内部に構築した。

フレーム間差分処理とは、時間差のある静止画像の画像情報を差分することで、画像情報に変化のあった画素を抜き出す手法である。複数のフレーム間差分を行うことで画像から動きのある部分だけを抜き出す等ができる。

今回は、一定時間毎に静止画像をメモリに取得し、カメラからの取得画像とフレーム間差分を行い、差分結果が閾値を超えている画素を赤 (R255:G0:B0) に書き換えて表示する回路とした。

写真 3 に処理結果を示す。



写真 3 フレーム間差分処理結果

処理が増えても、写真 2 の結果同様にリアルタイム表示される事が確認できた。

3.2 アルゴリズム検討実験

組込み機器向け画像処理として、監視カメラを対象としたアルゴリズム検討及び実験について汎用 PC を用いて行った。

(1) オプティカルフロー

オプティカルフローとは、画面上のある点における速度ベクトルを表し、物体の移動方向や速度等の動きを検出するアルゴリズムとしてよく用いられる手法である。

オプティカルフローを求める手法には大きく分けて勾配法とブロックマッチング法の 2 種類が有る。今回は、画像情報の取得先として屋外の駐車場を想定し、輝度値が急激に変化する箇所では誤差が大きくなる勾配法ではなくブロックマッチング法を実験に用いることとした。

ブロックマッチング法はある時間に取得した画像をいくつかの領域に分割し、ある領域が次の時間に取得した画像においてどの位置に対応するかを検索する手法である。

写真 4 にオプティカルフローの処理結果を示す。

原画像、処理画像ともに左から右に向かって時間が流れている。画像から左から右に向かって人が、手前に向かって車がそれぞれ移動しているのが分かる。

ソフトウェア処理として、駐車場における人や車の動きの動作量から色を選択してブロック毎に表示している。検出の誤差はみられるが動きのある物に対する検出が確認できた。



写真 4 オプティカルフロー実験結果

(2) ステレオ画像処理

子供の安全監視として、小型の画像処理装置を携帯する場面を想定し、ステレオ画像処理による人物接近検知アルゴリズムの検討及び実験を行った。

携帯可能とするために、WindowsOS が搭載可能なボードコンピュータを用いてバッテリー駆動の小型画像処理装置を試作した。

写真 5 に試作した画像処理装置を示す。

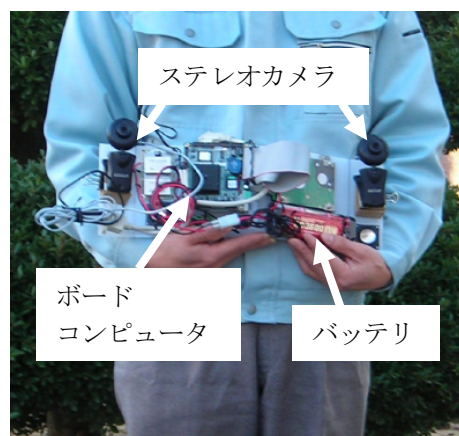


写真 5 試作した画像処理装置

2 台のカメラ画像を同時に取り込み、ステレオ視差画像を用いて対象物の距離を推定するプログラムを、ボードコンピュータに実装し、実験を行った。

写真 6 に実験結果を示す。



写真 6 ステレオ画像処理実験結果

左画像の下部に実験装置からの距離に比例した濃度変化を表示し、人物の接近具合によって、濃度が変化していることが確認できた。

4. 結果及び考察

ハードウェア実装実験およびアルゴリズム検討実験から、以下の結果が得られた。

- HDL を用いて、PLD に画像フォーマット変換回路とフレーム間差分処理回路を実装することができた。
- フレーム間差分処理回路の実装に於いて、結果表示がほぼリアルタイムで実現できることが確認できた。
- オプティカルフローによる動き検出アルゴリズムを確認することができた。
- ステレオ画像処理による人物接近検出アルゴリズムを確認することができた。

ハードウェア実装実験を通して、PLD を用いた画像処理回路は処理の高速化、機器の小型化に有効であることが確認できたが、HDL を用いた PLD 回路の開発はハードウェアを意識した記述が必要であり、実際の画像処理回路の実装においては、PLD の実装に適したアルゴリズム検討が必要であると思われる。

アルゴリズム検討実験を通して、監視向けのアルゴリズムとしてオプティカルフローやステレオ画像処理のような相関演算や画像間差分処理が重要であることが確認できたが、ソフトウェア処理だけでは処理時間が長く、ハードウェア処理を含めたアルゴリズム開発の検討が必要であると思われる。

5. まとめ

今年度の研究を通して PLD を用いた画像処理回路の有効性確認及び監視アルゴリズムで重要となる演算処理の検討と課題抽出を行うことができた。

次年度では PLD 向けの監視アルゴリズムを検討し、実際に PLD に実装して、PLD を用いた小型画像処理装置の試作検討を行う。