

FPGA を用いた動きセンサの開発とひと検出への応用

大村 功, 堤 大祐

Development of Motion Detection Sensor using an FPGA Device

Isao OHMURA, Daisuke TSUTSUMI

抄 録

自動化機器の安全確保や高齢者向け緊急通報装置等に利用可能な精度の高いひと検出センサが求められている。本開発ではFPGAとマイコンを組み合わせ、動きからひとを検出するセンサを開発した。動き検出にはフレーム間の差分を、ひと検出にはひとのシルエットを利用したテンプレートマッチングを用いた。FPGA (Field Programmable Gate Array)を利用することで処理の高速化を図る一方、マイコンにより拡張性を備えたセンサとした。

キーワード：プログラマブル・デバイス, FPGA, 画像処理, 動き検出, テンプレート・マッチング

Abstract

The motion detection sensor is required which is used to avoid dangerous situations around machines and to detect critical accidents of old persons in their house. We developed the Motion Detection Sensor with tracking object image that is applied the flame differential method and the template matching method. The Sensor is consisted of an FPGA (Field Programmable Gate Array) device to achieve high speed processing and a microcomputer to get flexible interface.

KEY-WORDS : Programmable Device, FPGA, Image Processing, Motion Detection, Template Matching

1. はじめに

近年の高齢化に伴って高齢者のみの世帯が増加している。このような世帯では住宅内での単独の事故や急病の際の対応の遅れが問題となっている。特に独居世帯においては事故から発見までに数日を要する場合もあり重大な問題である。

このような問題に対応するために、生活センサと呼ばれるセンサが利用されている。生活センサは従来の赤外線センサなどにタイマー機能を付加し、一時点での応答状況だけでなく、一定時間の応答回数を計数して行動の状況を出力するものである。一定リズムで生活する高齢者の場合は毎日ほぼ同じパターンの出力となるため、このような特性を利用することで緊急時に通報するセンサシステムとして利用することができる。

しかし、このようなセンサは、ひとだけでなく、ペットや

カーテンの揺らぎなどにも反応するなど、誤動作の問題が挙げられている。また一般的に利用されている自動ドアでもこのようなセンサを利用しているが、大型のタイプでは近年になってセンサの誤動作による重大事故が発生している。さらに緊急通報装置では、従来のひとの検出だけでなく、ひとの振る舞いから事故などの緊急事態を判断する機能が求められている。このようにこれらの課題を解決するためには従来のセンサでは不十分であり、画像を用いた高度なセンサ機能が必要である。

このような機能を実現するためにはひとを検出して追跡する機能の開発が重要である。本開発ではこのようなセンサのベースとして、動画像を利用したひと検出センサを開発した。開発したセンサは動きのある対象物を検出し、ひとの識別を行って追跡する機能を有する。FPGAとマイコンを用

いることでリアルタイムでの処理を実現した。

2. 動き検出とひと検出

2.1 処理の概要

本開発では、緊急通報装置向けセンサとして住宅内への適用を目指している。この用途では小型化、低コスト化が重要なポイントとなる。このため、機能をひとの検出と追尾に絞り、FPGA およびマイコンを利用した開発を行った。

処理内容は、動きの検出とひとの検出の2つの処理により構成されている。

動きの検出処理は動画像から動きのあった部分を検出して抽出する処理である。開発したセンサは動きのあった部分に着目してひとの検出処理を行う方法を採用しており、入力画像は1つ前の画像と比較して明度変化の大きさを示す画像へと変換される。この処理は画像の各画素について繰り返し行うため、処理回数が膨大となる。本開発では、動画をリアルタイムで処理するためにFPGAによるハードウェア処理を行い、高速化を図った。

後半の処理はひとを検出して追尾する処理である。動きの検出結果に対して、ひとの上半身のシルエットに相当したテンプレートを当てはめることにより、ひとの検出を行っている。テンプレートのマッチング処理は、確からしい位置を求めるため、テンプレートの移動、変形に相当する繰り返し演算を含んでいる。しかし、フレーム・レート（動画像の取込速度）でのひとの移動量は限られているため、適用範囲を限定した処理の大幅な低減を行うことが可能である。

全体の処理フローを図1に示す。なお、処理対象としている画像は320×240画素、毎秒30フレームのモノクロ画像である。

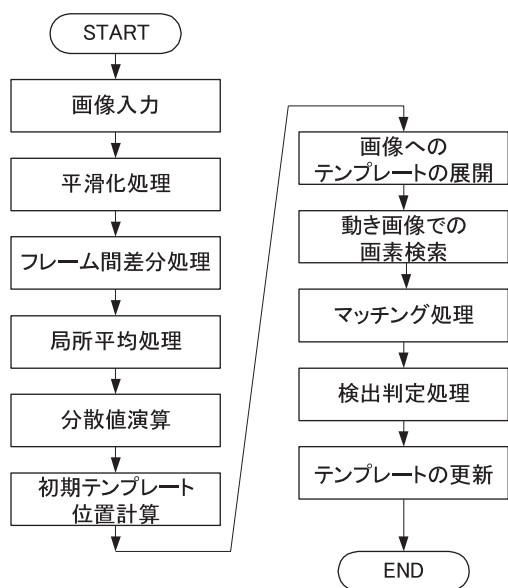


図1 全体の処理のフロー

2.2 動きの検出

動きのある対象物のシルエットを抽出する手法としては、背景差分法やフレーム間差分法などが知られている。背景差分法は基準となる背景画像との撮像環境の変化が小さな場合に有効であるが、24時間ひとの動きをチェックするセンサとしては、背景画像の保持方法や日中と夜間の撮像環境差などの問題点を解決しなければならないため適用が難しい。一方、フレーム間差分では明度変化が少ない場合に完全なシルエットが抽出できないという問題があるが、撮像環境の影響が少なく、動きの検出方法としてはコンパクトな実装が可能である。このため本開発ではフレーム間差分による方法を採用し、動画像における前後の画像の明度変化を動きとして抽出している。

動きの検出処理では以下の処理を行う。

- (a) 平滑化
- (b) フレーム間差分演算
- (c) 局所平均処理
- (d) 分散値演算
- (e) 初期テンプレート位置算出

平滑化は画像中のノイズ成分の影響を低減するための処理である。また、フレーム間の差分は各画素値の比として算出しており、処理 (b) ~ (d) は以下の式 (1), (2) で表される。

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n \left(\frac{I'}{I} - \mu \right)^2 \quad \dots (1)$$

ここで、 I および I' は現在のフレームおよび1つ前のフレームにおける画像の各画素値、 n は処理ブロックサイズの画素数、 μ はブロック内における平均値で式 (2) で表される。

$$\mu = \frac{1}{n} \sum_{i=1}^n \frac{I'}{I} \quad \dots (2)$$

通常のフレーム間差分との違いは、ブロック毎に画素値の比の平均を求め、それを用いて分散 σ^2 を求めている点である。分散値を採用したことで、特に蛍光灯等のちらつきによる画像全体の輝度変化の影響を低減することができた。

実際の画像による処理の例を図2に示す。画像は通常の歩行速度で奥行き方向へ移動するもので、図2-1、図2-2が画像の前後するフレーム、図2-3が処理結果である。

ひとの検出処理ではここで求めた分散値 σ^2 により構成される画像に対して、テンプレート・マッチング処理を行う。なお、処理 (e) ではひとが未検出の状態における初期テンプレート位置として、動き検出処理画像の重心位置を算出する。

2.3 ひとの検出

前述した動き検出処理で得られる画像は、動きのあった対象物の曖昧な輪郭を示す画像となる。これは対象物の完全な形状を示していないため、不完全な部分を推定した識別を行



(2-1) 1つ前のフレーム



(2-2) 現在のフレーム



(2-3) 処理結果

図2 動きの検出処理

う必要がある。ひとの検出処理では、このような処理方法として、ひとのシルエット形状を利用したテンプレート・マッチング処理¹⁾を採用し、ひとの検出、追跡を行った。

本センサの利用環境においては、カメラ視野内へのひとの出現および消失は歩行状態であると仮定できる。このことから採用したテンプレートはひとの上半身のシルエットに相当する形状とした。このような形状のテンプレートでは対象者の向きによって検出率が低下する可能性があるが、出現時に比較的高い検出率が期待でき、同時に処理量の低減を図ることができる。小型化が重要なポイントとなるセンサでは有利な検出方法である。また、住宅内という限定された環境においては対象者が限られるため、追跡時に検出条件をゆるめるなど対象者の向きの変化に対応した処理内容とすることが考えられる。

ひとの検出では以下の処理を行っている。

- (a) テンプレートの展開
- (b) テンプレート法線方向における微分と最大値をもつ画素の検索
- (c) 新たなテンプレート状態の算出
- (d) ひとの検出の判定
- (e) テンプレート状態の更新

テンプレートは25個の点から構成されており、処理(a)ではテンプレート中心位置から各点の位置を算出している。テンプレートの中心位置は前フレームでのひとの検出位置である。ひとが検出されていない状態では動き検出画像の重心位置にテンプレートを展開する。

前述したように対象がひとの場合ではフレーム間で大きな動きはないと仮定できるため、新たなひとのシルエットは展開されたテンプレートの近傍に位置すると考えられる。このことから処理(b)では、展開されたテンプレートの各点の位置付近で新たな輪郭の画素を検索する。検索する画素は法線方向の前後とし、10画素に限定した。また、微分値が一定値以下の点では輪郭を検出できなかったとして、処理から除外した。

処理(b)により得られた画素を対象として、処理(c)では新たなテンプレートの状態を求める演算を行っている。

処理(a)から処理(c)までがテンプレート・マッチングに相当する処理である。これらの処理はテンプレートの状態が収束するまで繰り返す。各パラメータが予め決められた値以下に収束しない場合は検出不能として処理を終了する。

本開発では、テンプレートの状態を示すベクトル \mathbf{q} を、2次元の画像における中心座標 (X, Y) とテンプレートのスケール (S) 、さらに傾き (θ) の4つのパラメータで構成した。すなわち、これらのパラメータの変更できる範囲でマッチング処理を行っている。

テンプレートを構成する点の数を n 個として、各点の座標を示すベクトルを $\mathbf{u} = (x_1, y_1, x_2, y_2, \dots, x_n, y_n)$ のように書き表すと、ベクトル \mathbf{u} は \mathbf{q} を用いた関数として式(3)のように表される。

$$\mathbf{u} = \mathbf{F}(\mathbf{q}) \quad \dots\dots (3)$$

\mathbf{u}_0 を処理(2)によって得た各画素の座標から構成されるベクトル、すでに適用したテンプレートの状態を \mathbf{q}_c 、 \mathbf{q}_c におけるテンプレートの各座標を \mathbf{u}_c として、 \mathbf{u}_c の周りに式(3)をTaylor展開して1次の項までを取ると式(4)のようになる。

$$\mathbf{u}_0 = \mathbf{u}_c + \left. \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \right|_{\mathbf{u}_c} \Delta \mathbf{q} \quad \dots\dots (4)$$

したがって、 \mathbf{u}_0 に適合するテンプレートの状態 \mathbf{q} と \mathbf{q}_c との差 $\Delta \mathbf{q}$ は式(5)で表される。

$$\Delta \mathbf{q} = \left[\left. \frac{\partial \mathbf{F}}{\partial \mathbf{q}} \right|_{\mathbf{u}_c} \right]^* (\mathbf{u}_0 - \mathbf{u}_c) \quad \dots\dots (5)$$

ここで、 $[A]^*$ は A の疑似逆行列である。 $\Delta \mathbf{q}$ は1次近似で得られた式から算出しているため、誤差を含んだ状態ベクトルとなっている。

この式で得られる $\Delta \mathbf{q}$ から式(6)により、新たなテンプレートの状態 \mathbf{q}_c を繰り返し求めることで、誤差を低減させ、より確からしいテンプレートの状態に収束させることができる。

$$\begin{aligned} \mathbf{q}_c^{t+1} &= \mathbf{q}_c^t + \Delta \mathbf{q} \\ \mathbf{u}_c^{t+1} &= \mathbf{F}(\mathbf{q}_c^{t+1}) \end{aligned} \quad \dots\dots (6)$$