

動画像からの移動物体の抽出と速度推定

Image-Extraction and Speed Estimation of a Moving Object from Photographed Digital Pictures

佐久間 湖[†], 堀田 厚生^{††}

Hiroshi SAKUMA[†], Atsuo HOTTA^{††}

Abstract This study aims at speed estimation of a moving object from pictures photographed with a digital camera. The background subtraction method was used to recognize a moving object. In addition, the recognition rate is improved with a median filter and noise elimination by pixel and line matching. Speed estimation was performed with comparing positions of the gravity of the object in two different pictures. Speed estimation is achieved with the accuracy of $\pm 1\%$.

1. はじめに

1・1 画像認識

近年、コンピュータの演算速度の向上や記憶媒体の容量増加などが進み、画像処理ソフトウェアの性能が飛躍的に向上している。これに伴い、これまでは困難であった動画画像処理も容易に行えるようになり、これを用いた動画画像認識技術への期待が急速に高まっている。このような流れの中、3次元空間上を移動する物体を動画像中において抽出する技術について、さまざまな研究が行われている。動画中の移動物体を抽出する従来の方法として、主に次の2つが挙げられる。

1、背景差分などの差分処理を用いる方法

2、物体の移動方向を示すオプティカルフローを用いる方法

まず、1の方法では、一般に背景画像が比較的容易に得られる場合には背景差分法が用いられ、背景が安定して得られる状況にない場合にはフレーム間差分法が用いられ

る。しかし、フレーム間差分法のみでは物体が一時的に静止している場合などには対応できない。一方、2の方法は剛体の動きを求めるのには有用であり、オプティカルフロー自体を求める手法についてもさまざまな方法が提案されている。しかし、その計算量は膨大であり、リアルタイム処理には向いていない。

1・2 スピード測定装置

現在、道路などで使用されるスピード測定装置は、測定区間を通過する時間を測定して車速を測定する。赤外線と光電管を使うワイヤー式およびレーダー波と呼ばれる周波数の高い電波を使い、反射して変化した周波数で測定するレーダー式がある。道路際に固定してあるものを固定式、パトカーなどに搭載するものを移動式という。

本研究では、画像認識とスピード測定の2つをあわせ、2次元画像でスピード測定をすることができるものを提案する。その際、前述の画像処理の説明のうち、背景差分方式が最も簡単かつ実用的な方法であると考えられるので、本検討ではこの方法を用いて移動物体の抽出を行い、2次元画像からの物体の移動量、速度を測ることを行う。

[†] 愛知工業大学 大学院工学研究科(豊田市)

^{††} 愛知工業大学 工学部 電子工学専攻(豊田市)

1・3 本研究の方法

本研究は、動画像認識と速度測定を目標とする。2次元空間上の動画像認識のため背景差分法を使用した。さらにメディアンフィルタなどにより認識率を向上し、また、画像同士から物体の重心を求め、差分を取ることで、認識した移動物体の速度を求める。

まず、カメラの原画像フォーマット(JPEG)から BMP ファイルに変換した。これは、作成した画像を保存するときに、非可逆圧縮である JPEG では、画像が劣化し、結果に誤差を引き起こす可能性があるためである。

2. 準備

2・1 使用用具と詳細

表 2-1 カメラの機能

有効画素数	約 320 万画素
記録画像ファイルフォーマット	JPEG, Motion JPEG(MOV) 動画から静止画の切り出し可能
記録画素数	静止画:2048×1536, 1600×1200,1280×960, 640×480 動画:320×240,160×120
動画フレームレート	15 フレーム/秒, 30 フレーム/秒

研究を始める際に、移動対象である車を含めた動画像を撮影する。使用したカメラはデジタルカメラ Dimage Xg である。

今回は、以下の条件での画像を使用する。

1. 晴天時で午前 10 時から 11 時までの 1 時間の間に撮影された画像。
2. 撮影時に使用するカメラを撮影中は完全に固定
3. 移動対象の物体が切れていない画像

この条件で撮影された動画像に背景および移動物体を含む画像があることを確認する。

2・2 使用ソフト

Microsoft Visual C++ V6.0 を用いてプログラムを行った。

2・3 画像ファイルの形式と構造

2・3・1 BMP

BMP(Microsoft Windows Bitmap Image) または DIB(Device Independent Bitmap) は、マイクロソフトと IBM が Windows と OS/2 にわかれる前の OS を共同で開発していたころに作られた画像ファイル形式である。圧縮の方法についても定義されているが、Windows が、標準では無圧縮のファイルを生成するため、他のアプリケーションにおいても、無指定時は、圧縮はされていない場合が多い。

ファイル形式の細部の変更が何度か行われており、(特に改善されているわけではない)、その結果として Windows と OS/2 で多少ファイル形式が異なることがある。

機械独立のファイル形式として設計されたため、実際に存在する画像表示装置や、印刷装置が、画像を上方から処理するものがほぼ全てであるにもかかわらず、幾何学的な X 軸、Y 軸方向に座標を指定する形式となっている。その結果、画像を下から上に向かって記録している (Bottom up) のが特徴であるが、高さに負の値を指定することでその他大多数の画像ファイル形式と同じように画像を上から下へ向かって記録する (Top down) こともできる。

BMP データの構造と変化を以下に示す。

・ OS/2 1.1 の場合

BITMAPFILEHEADER 構造体
BITMAPCOREHEADER 構造体
RGBTRIPLE 構造体 (インデックスカラー時)
画像データ

・ Windows 3.0 以降の場合

BITMAPFILEHEADER 構造体
BITMAPINFOHEADER 構造体
RGBQUAD 構造体 (インデックスカラー時)
画像データ

- Windows 95 以降の場合

BITMAPFILEHEADER 構造体
BITMAPV4HEADER 構造体
RGBQUAD 構造体 (インデックスカラー時)
画像データ

Windows 98 以降採用

BITMAPFILEHEADER 構造体
BITMAPV5HEADER 構造体
RGBQUAD 構造体 (インデックスカラー時)
画像データ

2・3・2 JPEG

JPEG 形式は Joint Photographic Experts Group の頭文字を取ったもので、自然画の圧縮に威力を発揮する。一般に JPEG 形式と呼ばれるものは JFIF (JPEG File Interchange Format) である。

- ヘッダーの構造

2 バイト	\$FF,\$D8 の 2 バイト
2 バイト	\$FF,\$E0 の 2 バイト
2 バイト	APPO フィールド (セグメント) 長
4 バイト	"JFIF" の文字
1 バイト	ゼロ
2 バイト	フォーマットバージョン
1 バイト	単位
2 バイト	水平解像度
2 バイト	垂直解像度
1 バイト	サムネール画像のピクセル数 (水平)
1 バイト	サムネール画像のピクセル数 (垂直)
不定	フルカラーの場合 (R×G×B) × 縦×横
256	カラーの場合 768 バイト (RGB 3 バイトのパレットデータ)
	縦×横バイト

2・3・3 Exif

Exif 画像の構造は基本的には通常の JPEG 画像形式そのものであり、その中にサムネール画像や撮影情報等のデータを JPEG の規約に準拠した形で埋め込んだものである。よって JPEG 形式をサポートしているインターネッ

トブラウザ、画像ビューア、フォトタッチソフト等を使えば、Exif 画像ファイルは通常の JPEG 画像として見ることができる。

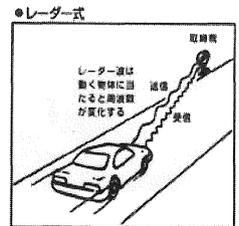
デジタルカメラの Exif には、使用カメラ、撮影時間などが記録されており、その後は、JPEG 形式と同様の構造になっている。

2・4 現在の主な車速の測定方法

車が一定区間を通過する時間を測定してその車の速度を算出する方法である。測定区間の始めと終わりに設置するセンサーには、赤外線や光電管、ワイヤ



ー (ループ式) などがある。また、レーダー波(マイクロ波)を使用して速度を算出する方法がある。



レーダー式ではレーダー 図 2-1 速度測定装置

一波 (あるいはマイクロ波) と呼ばれる周波数の高い電波が使われている。電波は周波数が高くなるほど直進性が増し、指向性が高くなる。この指向性の高いレーダー波は、動く物体に反射するとその周波数を変化させるという特性を持っており、それはドップラー効果と呼ばれている。そのドップラー効果を利用して、走行車両の速度を測定するのがレーダー式取締機である。取締機の送受信部から発信されたレーダー波は、走行車両に反射し、周波数が増加して、もどって来る。この発信されたレーダー波の周波数と、反射波の周波数の差を基に車両のスピードを測定する。レーダー式には、測定装置を道路際に固定して測定する「固定式」と、パトカーに搭載して移動しながら測定する「移動式」がある。

現在のスピード違反の取締りは、おもにマイクロ波(電波)を利用した「レーダー方式」が採用されている。

3. 処理の手順

3・1 背景差分法による物体の抽出

背景差分法により画像から目的の物体画像を抽出する。この動作を複数枚分行う。

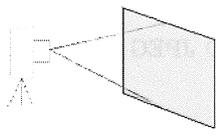


図 3-1 背景画像

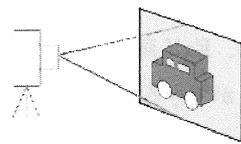


図 3-2 観測画像

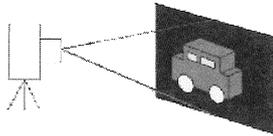


図 3-3 差分画像

3・2 メディアンフィルタによる残部除去

メディアンフィルタより、点状に残った部分の除去を行う。

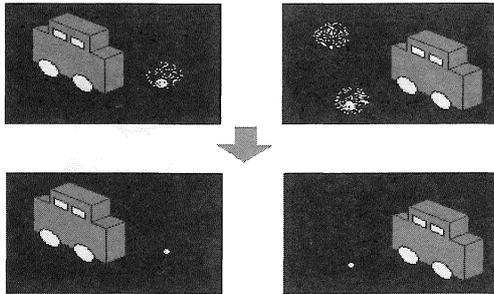


図 3-4 メディアンフィルタによる点状の除去

3・3 ピクセル検索法・ライン検索法による移動物体の再抽出

複数の抽出画像同士で形状の確認をして余分な部分を切り出す。形状の確認の際、ピクセル単位とライン単位の検索を行う。

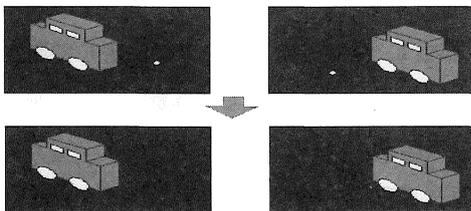


図 3-5 同形状の物体を認識

3・4 重心検索法による重心位置の測定、重心の差分による速度測定

差分抽出画像に残った物体の重心を調べて互いの重心位置と 1 フレーム間の時間を調べて速度を測定する。(ピクセル単位)

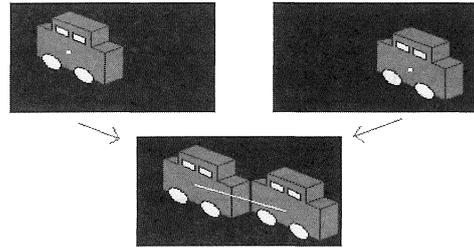


図 3-6 重心と速度の測定

4 処理の詳細

4・1 背景差分法

移動物体が存在しない状態の画像（背景画像）を事前に取得しておき、移動物体が含まれている入力画像と背景画像の各画素の画素値を比較し、その値が一致しなければその画素を活動観測領域とする方法である。しかし、背景差分法では画像上に現れる明度の変化をすべて対象として検出するため、画像上の背景領域で明度変化が起こりやすい環境では誤検出が生じる。そのため、実際には画素値の測定には誤差が伴うので、画素値の差が一定以上のときにはじめて活動観測領域とみなす。この問題に対し、本研究では、雲や時間による明度変化に対応するために 1 つの動画像に対し、きわめて近い時間のもの同士で比較を取り、RGB の明るさの合計値が 50 以上のものを観測領域とし、明度変化の小さいものを切り捨てることで移動物体を検出することにした。

また、はじめの部分でも述べたように、移動物体の存在しない背景画像を取得するのが難しい状況や背景自体が変化する状況においては背景画像を画像処理によって生成する必要がある。この背景画像を精度良く生成できれば背景差分の精度も向上すると考えられる。背景差分法のアルゴリズムを次に示す。

$$d_i(x,y;t) = f_i(x,y;t) - b_i(x,y;t) \quad (1)$$

$$g_i(x,y;t) = \begin{cases} 1 & \text{If } d_i(x,y;t) \geq D \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

ここに、 x,y は画像座標位置、 t は時刻、 $f_i(x,y;t)$ は入力画像、 $b_i(x,y;t)$ は背景画像、 $g_i(x,y;t)$ は背景差分画像、 D は閾値である。また、 $i=r;g;b$ は各 RGB 成分を表す。背景差分の閾値 D は検出漏れを防ぐためには

小さいほどよいといえる。しかしながら、閾値を小さくするとそれだけ検出誤りも増えてくる。したがって、背景画像生成の精度がこの閾値をどれだけ小さくすることができるかに大きく影響を与えられ、サイズ5×5 閾値5とすると次のようになる。

表 4-1 背景画像

10	11	10	9	8
11	12	13	10	7
12	13	10	8	6
13	14	12	10	5
14	15	12	8	4

表 4-2 観測画像

10	11	10	9	8
11	15	10	26	10
12	16	15	20	30
13	15	20	30	31
14	15	30	31	31

表 4-3 差分画像

0	0	0	0	0
0	3	3	16	3
0	3	5	12	24
0	1	8	20	26
0	0	18	23	27

表 4-4 閾値調整後

0	0	0	0	0
0	0	0	16	0
0	0	5	12	24
0	0	8	20	26
0	0	18	23	27

4・2 メディアンフィルタ

非線形演算要素を使ったフィルタでその名のとおりに、メディアン(中央値)をとるフィルタで、図に示すようにある点の近傍8点を入れた3×3のエリアで大きい順に並べたときの真ん中の値を、その中心の位置に置き換えるものである。

メディアンフィルタは、パルスノイズのように突出した値を持つノイズを、画像をあまりぼけさせることなく除去できる。

今回使用した理由はノイズを除去するのではなくノイズのような点状に残った部分を除去し、一定以上の大きさを持つものを物体と判断するために使用した。また、この手法を取り入れるとき隅の1点は除去されることがある。

次の表の2行目を見た場合、2つの値に変化がある。

表 4-5 フィルタ前の状態

0	0	0	0	0	0
0	1*	0	1**	1***	1****
0	0	0	1	1	1

表 4-6 フィルタ後の状態

0	0	0	0	0	0
0	0*	0	0**	1***	1****
0	0	0	1	1	1

* 並べ替えると 0,0,0,0,0,0,0,1 となり、0 となる。

** 並べ替えると 0,0,0,0,0,1,1,1,1 となり、0 となる。

*** 並べ替えると 0,0,0,0,1,1,1,1,1 となり、1 となる。

**** この手法では、隅を調べることはできないので変化なし

4・3 ピクセル検索法

背景差分法によって検出された物体に対し、同形状の物体を発見するために考案した方法の1つで、複数撮影した画像同士を比較することで同形状の物体を発見し検出する。それと同時に背景差分法で取りきれなかった背景部分をなくす目的もある。

検出方法として、明度がある部分を1ピクセルずつ調べ、発見した位置から縦横±2の範囲がもう一方の画像と同等の形である部分があるか調べ、同形があるならば検出する手法である。しかし、背景差分法による抽出画像は、ばらつきがあるため、少しの違いを考慮した抽出を行う。

表 4-7 検索範囲と表 4-8 との比較

0	0	0	0	0	0
0	0	0	0	0	0
0	0	1	1	1	0
0	0	1	0	1	1
0	0	0	1	1	1
0	1	1	0	1	1

表 4-8 検索範囲と表 4-7 との比較

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	1	1	1
0	0	0	1	0	1
0	0	1	0	1	1
0	0	1	0	0	1

表 4-7、表 4-8 では横方向 1 の移動と検索ミスが 1 つある。

4・4 ライン検索法

4-3 で示した検索法と別で考案した手法で、1 ピクセルごとの縦または横のラインの明度を互いに調べれば同等の明暗変化がある部分を検出する方法である。移動後の物体が画面外に出た場合、隅の画像の列(行)の明るさの数を検出し一致した列と行の交差位置より前までの数と一致するか調べ、判断する。また、ピクセル検索法と同様に背景差分法による抽出画像は、ばらつきがあるため、少しの違いを考慮した抽出を行う。今回は、横のラインを使用した。

表 4-9 検索範囲と表 4-10 との比較

0	0	0	0	0	0
0	0	0	0	0	0
0	0	1	1	1	0
0	0	1	0	1	1
0	0	0	1	1	1
0	1	1	0	1	1

表 4-10 検索範囲と表 4-9 との比較

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	1	1	1
0	0	0	1	0	1
0	0	1	0	1	1
0	0	1	0	0	1

表 4-10 と表 4-11 では横方向 1 ずれた位置で検索される。

4・5 重心検索法

手順 4 までにできた画像の全体から、移動物体の重心を求める手法として考案した方法を示す。手順 4 までにできた画像は、移動物体の部分のみ明度があると考え、全体から明度のある部分の位置、数を調べ、総和÷総数を縦、横ともに行うことで導き出された値が重心位置とした。この手法は、背景の残り、誤検索が少ないほどより正確な重心位置を与える。

表 4-11 のような状態のとき(縦・横ともに 0 から数えるものとする)

縦成分 $2 \times 4 + 3 \times 3 + 4 \times 5 + 5 \times 6 + 6 \times 8 + 7 \times 4 + 8 \times 4 = 175$

横成分 $1 \times 2 + 2 \times 5 + 3 \times 7 + 4 \times 4 + 5 \times 4 + 6 \times 6 + 7 \times 3 + 8 \times 3 = 154$

縦平均 $175 \div 34 = 5.14 \div 5$

横平均 $154 \div 34 = 4.52 \div 5$

よって赤色の位置(5,5)が重心となる。

表 4-11 画像の状態と重心位置

	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
2	0	0	0	1	1	1	1	0	0
3	0	0	0	1	1	0	1	0	0
4	0	0	1	1	1	1	1	0	0
5	0	1	1	1	0	1	0	1	1
6	0	1	1	1	1	1	1	1	1
7	0	0	1	1	0	0	1	0	1
8	0	0	1	1	0	0	1	1	0

4・6 速度測定

4 の手法により導き出された重心位置を別の画像同士でみることで物体の移動距離 (ピクセル単位) を導き出すことができる。この距離とカメラの動画フレーム数を調べることで秒単位の速度(pixel/s)が導き出せる。今回は、1 秒 15 フレーム、30 フレームの二つの動画画像から測定した。

1 フレームにかかる時間

15 フレーム/秒時

$1 \div 15 = 0.0666 \dots \div 0.067$

30 フレーム/秒時

$$1 \div 30 = 0.0333 \dots \approx 0.033$$

速度

$$\text{距離(ピクセル)} \div \text{時間}(0.067 \text{ or } 0.033)$$

4・7 研究中に作成した抽出法

4・7・1 フレーム検索法

背景画像と同位置で撮影された連続画像の読み込みをし、特定の列(行)に異物が侵入してからもう一つの特定の列(行)に到達した際の物体の頂点(方向によって位置は変化)を見ることで物体の進入速度を測る。異物の検索は、背景差分法を使い、列(行)内に色のある部分が一定以上の連続しているものがあると判断したとき、それを進入物と判断する。距離と速度の設定の仕方は、動画像のフレームによって変化する。

問題点

背景差分法を使用する際、閾値を高く設定しておかねば、残った背景部分を異物と誤認してしまう。

閾値が高いため、物体の形状がおかしくなり、正確な計測を行いにくい。

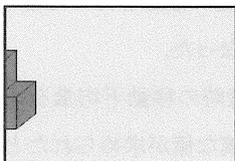


図 4-1 進入画像

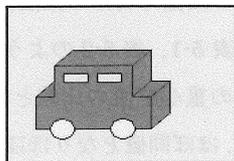


図 4-2 フレーム内画像

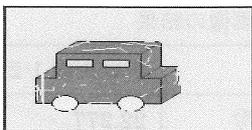


図 4-3 極端な問題画像(先が消失)

4・7・2 AND 型検索法

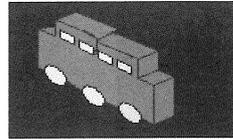
動画像の隣り合った2つの画像同士で背景差分法をとり、さらに、背景差分法の画像の隣り合った2つの画像でAND計算を行い、残った部分を物体とみなす。さらに隣り合った画像同士(1フレームのずれ)で移動量等を測り、速度を求める。

問題点

隣り合った画像間で大きな変化が起きるとまったく違

う結果ができる。

背景差分法の際に重なり合った部分の処理法に難あり(そのまま行くと、物体が重なった部分が消しあってしまう)。



う)。

図 4-4 期待画像

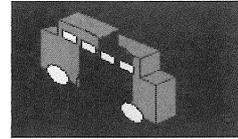


図 4-5 実際になりうる画像

4・7・3 データ照合法

動画像でとった画像と、事前に用意した車のデータと照合し、一致したものを抽出する。

これらで抽出された画像同士で差分をとり、移動量、速度の測定を行う。

問題点

事前に用意する画像の量が膨大になり、それぞれと照合するため、処理量が多く、また用意した画像と違うと物体が照合できない(用意できる画像に対してのみ有効)。

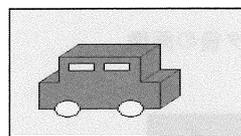


図 4-7 照合できる車

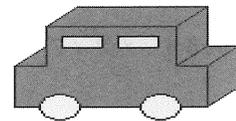


図 4-6 照合前の画像

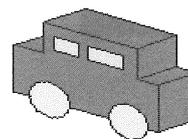


図 4-8 うまく照合できない車

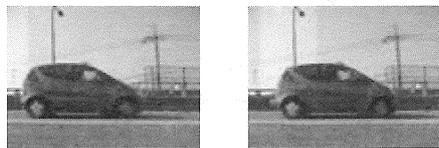
5 実験結果

5・1 背景差分法

この画像は、今回使用した背景画像と観測画像である、背景差分法により移動物体を抽出したものが画像 5-3 である。しかし、差分のみでは背景が少し抽出される問題があるので、閾値 RGB 合計値 50 として大きな差分のみを抽出されるようにした。

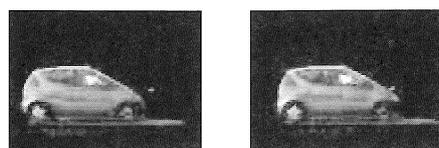


図 5-1 背景画像



画像 1 画像 2

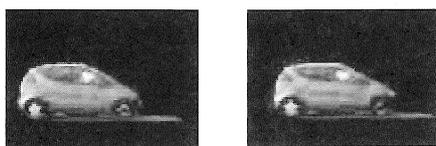
図 5-2 入力画像



画像 1 画像 2

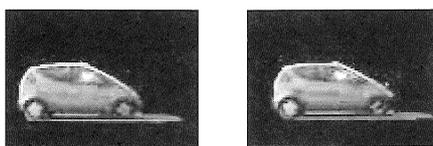
図 5-3 背景差分後の画像

5・2 メディアンフィルタと画像修正



画像 1 画像 2

図 5-4 メディアンフィルタ後の画像



画像 1 画像 2

図 5-5 修正後の画像

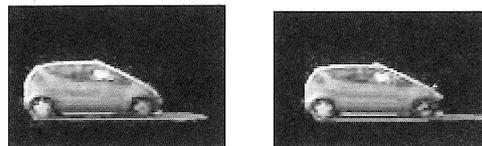
メディアンフィルタは、画像のノイズを処理するものが一般的であるが、背景差分法の残った部分が雑音のように残っているため、小さな物体を対象にしていなければ背景の残りを処理する方法として活用できる。しかし、物体の輪郭部分が少しけずれ、少しぼやける傾向があるため、画像位置の縦横±1の部分を元の画像から抜き出すことで、物体の形状を戻した。

5・3 ピクセル検索、ライン検索

画像の状態が悪いと視覚判断したときに再修正できるように作成した検索プログラムであるが、今回準備、使

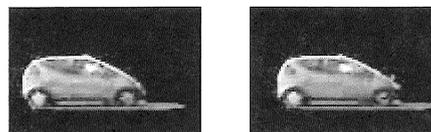
用した画像に対して、使用する必要がなかった。しかし、他の画像に対して必ずしも不要であると限らないために作成した。

今回作成した検索は2つあり、個別の横成分の移動距離に大きなばらつきがあるこれは、検索ミスによる誤差を±10%と設定した結果、認識可能な位置が増えたため、目的の位置と違う部分を抽出したと考えられる。しかし、平均では、ほぼ正確な移動距離を与えるという結果となった。



画像 1 画像 2

図 5-6 ピクセル検索の結果



画像 1 画像 2

図 5-7 ライン検索の結果

5・4 重心、速度測定結果

今回の画像で物体の抽出、速度の推定を行った結果、次ページ表 5-1、表 5-2 のようになった。

物体の重心位置の比較と検索時の移動平均量を比較した結果、ほぼ同値となりほぼ正確な値が求められたと考えられる。

表 5-1 ライン検索した画像の結果

	実験画像 1-2
ライン検索横移動平均	22.277
最大	32
最小	11
ライン検索縦移動平均	-0.6
最大	0
最小	-2
横移動最多移動距離	22
縦移動最多移動距離	0
重心比較 横	22.306
縦	-0.418
移動速度(pixel/s)	651.300...

重心位置 画像 1(122.5 178.8),画像 2(144.8 147.4)

表 5-2 ピクセル検索した画像の結果

	実験画像 1-2
Pixel 検索横移動平均	21.829
最大	199
最小	-180
Pixel 検索縦移動平均	-0.596
最大	1
最小	-1
横移動最多移動距離*	-59
縦移動最多移動距離*	-1
重心比較 横	22.000
縦	0
移動速度(pixel/s)	660.000...

重心位置 画像 1 (122. 147) 画像 2 (144. 147)

5・5 途中結果と最終結果の比較

メディアンフィルタ後の状態と補正後の状態の重心、移動量、速度を表 5-3 に示し、今回使用した画像のピクセル・ライン検索前と検索後の結果の比較を行なった。

表 5-3 ピクセル、ライン検索前の状態

		画像 1
メディアンフィルタ後	重心 横	22.240
	縦	-0.420
	移動量	22.236
	移動速度	667.218
補正後	重心 横	22.306
	縦	-0.418
	移動量	22.310
	移動速度	669.314

今回比較した画像に関して、ピクセル・ライン検索による同物体の確認を行うほどの乱れた結果にはならず、メディアンフィルタ、補正後の画像を使用してもほぼ正確な結果となった。これは、時間による明度変化の少ない状態であったことや、物体と背景との色相が大きく違っていたためと思われる。

今回使用したカメラでは、撮影した画像は 320×240 とかなり小さめの画像となったため、処理時間は 1 秒もかからなかったが、通常写真の 2048×1024 となると、処理時

間 1 分以上とかなりかかるため、画像のサイズを考慮した処理方法が必要であると考えられる。

6 結言

動画画像からの移動物体の速度を測定することを目的として、次の研究を行った。

- 1) デジタルカメラで移動物体の撮影を行う。
その中の一部画像を用いて以下の処理を行う。
- 2) 2つの画像から背景差分法により移動物体を切り出し。さらに、メディアンフィルタの適用により認識率の向上を行った。
- 3) 切り出した移動物体像の重心を認識された物体のピクセルの位置成分の総和の平均で求めた
- 4) 以上のデータより、移動物体の速度を求め、誤差±1%程度で測定できることを確認した。

7 今後の課題

- 1) pixel 検索法の改良
- 2) 背景差分法の改良
- 3) 背景差分法以外の抽出法の検討
- 4) 重心検索の改良
- 5) 斜め走行の移動物体の速度推定
- 6) カメラとの連動によるダイレクト処理

8 参考文献

[1] [PDF] 動画画像からの背景画像生成を用いた移動物体抽出方法に関する一検討

—
<http://www.tom.comm.waseda.ac.jp/~hayasaka/Study/vm29-1.pdf>

[2] [PDF] 道路画像の中の自動車認識と交通自動測定

—
<http://www.ei.it-hiroshima.ac.jp/home/nakamura/kenkyugaiyo/takeuchi.pdf>

[3]RADAR DETECTOR

<http://www.k-electro.co.jp/radar/radaare.html>

[4]レーダー探知機機能一覧

<http://www.warp.or.jp/~mr-shop/rader/radar04.htm>

[5]Sample

<http://www.acilom.co.jp/rader.html>

[6]我楽多頓陳館 各種解説コーナー

<http://www3.starcat.ne.jp/~snap/room03/c02/c02.htm>

[7]C 言語で学ぶ実践画像処理 P57 オーム社

井上誠喜・八木伸行・林 正樹・中須英輔・三谷公二・
奥井誠人 共著

[8]C 言語で学ぶ実践デジタル映像処理 P144 オーム社

井上誠喜・八木伸行・林 正樹・中須英輔・三谷公二・
奥井誠人 共著

(受理 平成 17 年 3 月 17 日)