視覚障害者歩行支援のための歩行者検出システム

A Pedestrian Detection System for the Visually Impaired using a Single-camera

平成 24 年度

修士論文

横浜市立大学大学院

生命ナノシステム 科学研究科

ナノシステム科学専攻

115208 岸野嵩久

主指導教官 Ruggero Micheletto

副指導教官 大月俊也

野嶋俊司

目次

第 1 章

序論

1.1 研究背景

近年、国内の視覚障害者は在宅者だけでも 31 万人を上回ると言われ[1]、視覚障害者の歩行を支援して行動範囲の拡大を促進することは福祉的に大きな意義がある。歩行支援の一環として、視覚障害者が単独歩行をする際には白杖や盲導犬などを利用することが一般的である。また、公共のインフラ設備によるバリアフリー化や点字ブロックなどの誘導サインも視覚障害者の歩行支援の役割を担う場合もある。しかし、白杖は知覚できるものが近接したものに限られる、盲導犬は育成に掛かる時間から絶対数が需要に対して足りないといった問題点がある。また、点字ブロックには線状ブロックと点状ブロックの 2 種類しかなく、情報提示能力に限界があると考えられている。これらの問題点を解決するべく、視覚障害者の歩行を支援するためのシステムの研究開発は現在も盛んに行われている。

視覚障害者の歩行支援には、経路のナビゲーションと移動時の安全確保の 2 つの要素が必要である。これまで提案または開発されてきた視覚障害者支援のシステムの多くは、GPS を利用した座標測位や特定のランドマークを認識する位置推定による経路ナビゲーションシステムなどであった [2]。その一方で、これらのシステムは歩行者や車などの移動物体のことは考慮されていないことが多く、衝突回避の喚起など歩行中の安全確保の課題が存在する。また、厚生労働省の調査によって、視覚障害者が外出の際に最も不安を感じることの 1 つは、人や車といった道を移動する存在と歩行中に衝突することであることが明らかにされている[1]。しかし、移動物体を検出して移動中の安全を確保するようなシステムの研究は、経路ナビゲーションシステムの研究に比べて報告例が少ない。そこで、本研究では視覚障害者の歩行中の安全確保のために、前方の歩行者を検出して衝突の危険性があれば通知して事故を未然に防ぐシステムの開発を目指す。

歩行者などの障害物を検出して視覚障害者の歩行を支援するシステムは IR センサーや超音波センサーなどの距離測定に特化した多数のセンサーを使用し、専用のウェアラブルデバイスを作成する必要があるシステムが多く提案されてきた[3]。しかし、これら専用のデバイスは生産のコストが大きいことや希少性

から一般に普及しにくいといった問題が懸念される。これらの理由から、視覚障害者を支援するシステムは既に普及されているデバイスに搭載されているセンサーを用いて実現することが望ましい。また、視覚障害者が利用する際に携帯しやすい重さや大きさのデバイスであることも重要である。これらの理由から、我々は携帯電話やスマートフォンなどのモバイルデバイスに注目した。これらのデバイスは一般への普及率が高く入手が容易であることに加えてカメラとスピーカーが搭載され、システムに必要なセンサーを備えている。本研究はモバイルデバイスでの実装を視野に入れ、カメラ映像から歩行者を検出して衝突の危険を察知するシステムの開発を目指す。

1.2 研究目的

本研究では前方の歩行者の検出を行うことで、視覚障害者が他の歩行者との衝突の危険を未然に防ぐことができるようなシステムの開発を目指す。モバイルデバイスでのシステム実装を想定するため、従来の研究とは異なりデータの入力は光学センサーである CCD カメラからのみとする。また、本研究の歩行者検出はリアルタイムで行われることが求められる。しかし、従来の人物検出は検出の際に、画像上を複数回走査するために処理に膨大な時間を要するという問題がある。そこで、本研究で行う検出処理では画像の走査範囲を効率的に制限することで、処理に必要な時間を削減してリアルタイムでの処理の実現を目指す。そして、衝突の危険性を判断するために、検出結果から前方の歩行者までの距離の推定を行う。

第 2 章

歩行者検出

2.1 人物検出に用いられる特徴量

カメラ映像から人物を検出するシステムの研究は盛んに行われているが、これらの多くは監視カメラをはじめとした固定カメラを用いたシステムである。これらのシステムは検出の精度を向上させるために背景差分法を用いることが一般的である[4]。背景差分法では検出処理をする画像の他に、予め検出対象が写っていない背景画像を基地情報として与える。この背景画像と入力画像の差分を計算した差分画像を生成し、得られた差分画像を閾値で分別することで検出対象の領域を特定することが出来る。背景差分法は実装が容易であることに加えて、処理負荷が高くないという利点がある。しかし、カメラが移動してしまい背景画像が一定でない場合には使用することが出来ず、また検出対象以外の物体移動や照明の変化に対応することが出来ないという問題がある。一方で、背景情報に依らず、検出対象事態の特徴を利用した検出法がある。

検出に使用される特徴の 1 つに、検出対象全体を表現する特徴量があり、これらの検出方法は 1970 年頃から盛んに研究されてきた[5]。しかし、対象全体を表す大局的な特徴では、複雑なモデルを作成することが困難であることや、歩行者のような向きや姿勢などにより見え方の変化が大きいために対応しきれないといった課題があった。そこで、近年は機械学習法の発展に伴って、検出対象の局所的な特徴を統計的学習と組み合わせた手法が注目されている。局所的な特徴としては、輝度の情報や検出対象の形状(エッジ)情報に注目したものに大きく分類できる。

2.1.1 Haar-like 特徴量

Haar-like 特徴量は画像の輝度値に着目した特徴量である。輝度値は照明条件の変動やノイズの影響を受けやすいため、近接した領域の輝度差を計算することで影響の軽減をする。Fig.2.1 に示すような、白領域と黒領域の輝度差を以下の式より算出したものが Haar-like 特徴量とされる。

$$H(r_1, r_2) = S(r_1) - S(r_2)$$

ここで、$S(r)$は領域rの輝度の和を算出する関数とする。また、白領域をr_1、黒

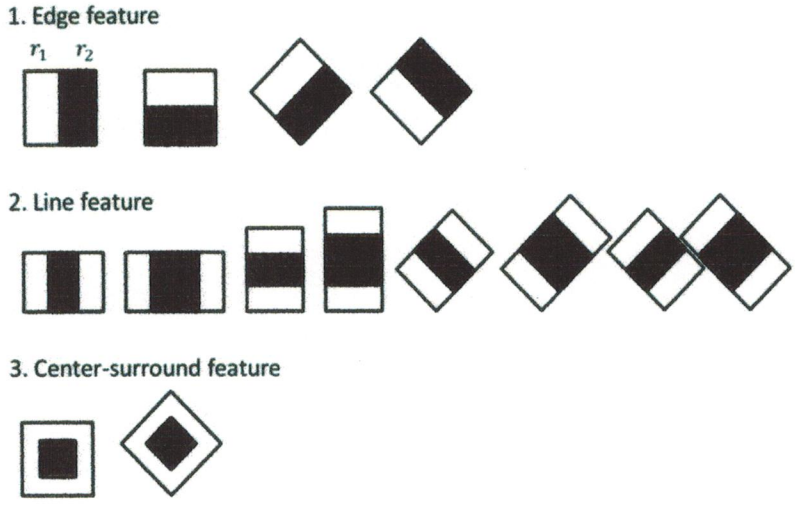


Fig.2.1 Haar-like 特徴量

領域をとr_1とした。輝度差を求める 2 つの領域のパターンは複数存在し、Fig.2.1 に示すようなパターンが一般的に用いられている。これらのパターンを組み合わせることによって、縦方向や横方向、斜め方向の明暗の差を捉えることが可能になっている。これらの領域パターンを基にして位置やスケールを網羅的に変化させることにより、検出対象をモデル化した特徴量を生成する。生成した特徴量から、対象の検出に有効な特徴量を Boosting によって選出する手法が Viola らによって提案された[6]。

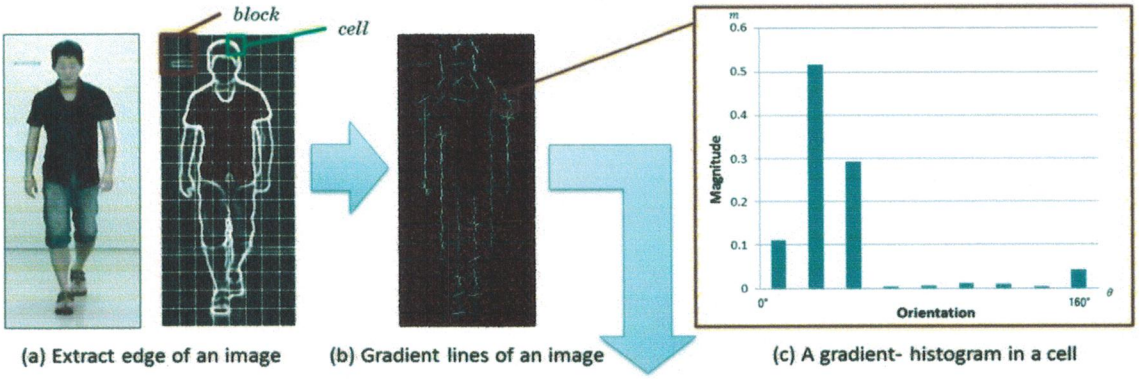
2.1.2 Histograms of Oriented Gradients (HOG)特徴量

HOG 特徴量は、検出対象の形状に着目した特徴量である[7]。Fig.1.2 に HOG 特徴量の算出過程を示す。Fig.1.2(c)のように、局所領域におけるエッジの勾配方向とエッジ強度をヒストグラム化した特徴量で形状を表現する。ヒストグラム化する際に局所領域ごとに正規化を行うので、照明などによる明暗のバラつきなどの影響を受けにくいという特徴がある。勾配の方向θとその強度mは以下の式によって算出される。

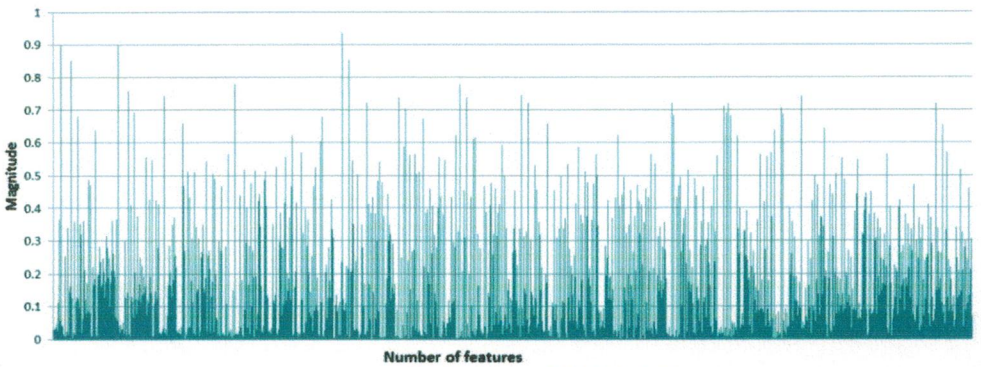
$$m(x,y) = \sqrt{f_x(x,y)^2 + f_y(x,y)^2}$$

$$\theta(x,y) = \tan^{-1}\frac{f_y(x,y)}{f_x(x,y)}$$

ここで、$f_x(x,y), f_y(x,y)$は各画素における輝度値Lの差分であり、以下の式で与えられる。



(a) Extract edge of an image (b) Gradient lines of an image (c) A gradient- histogram in a cell



(d) A histogram of HOG feature

Fig.2.2 HOG 特徴量算出過程

$$f_x(x,y) = L(x+1,y) - L(x-1,y)$$

$$f_y(x,y) = L(x,y+1) - L(x,y-1)$$

各ピクセルで算出した勾配の方向θとその強度mから、セルと呼ばれる領域ごとに量子化した勾配方向を要素とするヒストグラムを作成する。なお、各セルは$n \times n$[pixel]から構成され、セル同士の構成ピクセルは重なり合わない。勾配の方向θは0°から360°までの値であるが、ヒストグラムを作成する際には0°から180°までの値に変換する。これにより、色情報に左右されない勾配方向を扱うことができる。変換した勾配の方向θを量子化して、量子化勾配方向θ'の勾配強度の和を次式より求める。

$$v(\theta') = \sum_x \sum_y m(x,y)\delta_{\theta',\theta}$$

ここで、$\delta_{\theta',\theta}$は Kronecker のデルタ関数であるので、量子化勾配方向θ'と量子化した勾配方向θが同じヒストグラムの要素である場合 1 となる。量子化した勾配方向がK方向である時、勾配方向のヒストグラム$V = \{v(0), v(1), \cdots, v(K)\}$となる。

照明などによる明暗のバラつきなどの影響に強固になるために、ブロックと呼ばれる領域ごとに正規化を行う。ブロックは$l \times l$[pixel]から構成される領域で

あり、1 セルごとに正規化の対象となる領域が重なるように移動しながら正規化を行う。ブロックごとの正規化は以下の式によって行われる。

$$v'(\theta') = \frac{v(\theta')}{\sqrt{\left(\sum_{k=1}^{n\times n\times K} v(k)^2\right) + \epsilon}}$$

ここで、ϵは計算の際に出力不可になることを回避するための数値であり、$\epsilon = 1$とした。1 ブロックに含まれるセルの数をBとすると、正規化後の勾配方向ヒストグラムV'は、$V' = \{v'(0), v'(1), \cdots, v'(B \times K)\}$となる。ブロック領域を 1 セルずつ移動させていき作成された各ヒストグラムを順々に並べていくことで、Fig.1.2(d)のような 1 つのヒストグラムとして対象を表現することが出来る。

人物検出の場合、色情報は服装や肌の色の違いだけでなく場所や時間による照明の変化にも大きく影響を受けることため、歩行者の検出には適した特徴ではないことが考えられる。そこで、本研究での歩行者検出は HOG 特徴量を用いることとした。

2.2 機械学習

前節で挙げたような特徴量は対象の局所領域の特徴を抽出したものであるので、対象全体を表現するためにはヒストグラムの総数が膨大になる。そこで、統計的学習手法によって特徴量を学習することで、識別に有効な特徴量のみを抽出して使用する方法が用いられる[8]。統計的学習法として、Boosting を用いる手法が近年一般的になっている。Boosting とは、精度は高くないものの単純なアルゴリズムから成る弱識別器を複数組み合わせることで、高精度かつ高速な強識別器を構築する手法である。Boosting には、体表的なものに AdaBoost[8]、Real AdaBoost[9]などがある。ここでは、代表的なこの 2 つの手法について述べる。

2.2.1 AdaBoost

AdaBoost は Freund と Schapire によって 1996 年に提案された手法で、Boosting の中でも最もよく知られているアルゴリズムの 1 つである[8]。AdaBoost のアルゴリズムは、学習サンプルに対して個々に重みを設定する。学習サンプルは識別対象(ポジティブクラス)とそれ以外の背景(ネガティブクラス)の 2 クラスから構成される。学習サンプル数をN個、特徴量の総数をM個

とするとき、以下の式で与えられるエラー率$e_{t,m}$が最小となる弱識別器を一定の学習回数だけ選出する。

$$e_{t,m} = \sum_{n}^{N} w_{n,t} |h_t(x_n) - y_n|$$

ここで、tは学習のラウンド数、mは識別器の番号、$w_{n,t}$はn個目のサンプルのtラウンド目での重み、y_nはn個目のサンプルのクラスラベル(ポジティブ:1、ネガティブ:-1)である。$h_t(x_n)$は学習サンプルx_nに対する弱識別器の出力であり、Violaと Jonesの顔検出法では以下の式のような、ある特徴量の1次元ヒストグラムを閾値により判別するような弱識別器が使用された[8]。

$$h_t(x_n) = \begin{cases} 1 & pv(x_n) > p\theta \\ 0 & otherwise \end{cases}$$

ここで、$v(x_n)$はサンプルx_nを入力した際に得られる特徴量であり、θは識別を判断するための閾値、pは不等号の向きを決定するための符号で-1または+1の値をとる。

最終的に対象物か非対象物を識別するための強識別器は、弱識別器に対する重みα_tをつけて足し合わせた値が閾値λを超えるかどうかで判断する。強識別器の値は以下の式によって算出される。

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - e_t}{e_t}\right)$$

$$H(x_n) = sign\left[\sum_{t=1}^{T} \alpha_t h_t(x_n) - \lambda\right]$$

2.2.2 Real AdaBoost

Real AdaBoostは AdaBoostを拡張した手法である[9]。AdaBoostとは異なる特徴として、弱識別器の値が特徴量の分布に応じた実数値になる。また、AdaBoost に比べて学習の収束が早いために、少ない弱識別器で高精度の検出をすることが出来る。この理由から、本研究では歩行者の特徴量学習にこのReal AdaBoostを用いる。

学習サンプルの2つのクラス(ポジティブクラス：対象物、ネガティブクラス：非対象物)ごとの特徴量の分布を確率密度分布W_+、W_-によって表すことにより、2つの分布の差に応じて数値が実数化され、必要な弱識別器の数が削減された。各クラスの確率密度分布は以下の式で求められる。

$$W_{+,j} = \sum_{n:j\in J \wedge y_n=+1} D_t(n)$$

$$W_{-,j} = \sum_{n:j\in J \wedge y_n=-1} D_t(n)$$

ここで、jは量子化した特徴量の区間番号を示し、BIN 数はJとなる。y_nは学習サンプルnのクラスラベルを表し、$y_n\{+1,-1\}$となる。また、$D_t(n)$はn個目の学習サンプルのt回目の学習における重みを示す。

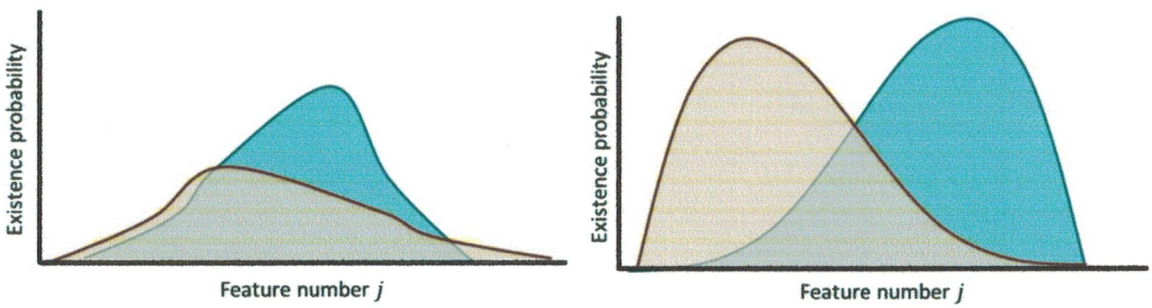
弱識別器$h(x_n)$は 2 つの確率密度分布の差によって求められるので、

$$h(x_n) = \frac{1}{2}\ln\left(\frac{W_{+,j}+\epsilon}{W_{-,j}+\epsilon}\right)$$

で与えられる。ϵは計算不可になることを防ぐための数値であり、$\epsilon = 0.0000001$とした。M個の弱識別器候補の中から識別に有効な弱識別器を選択するには、2 つの確率密度分布の Bhattacharyya 距離から類似度を利用する。識別に対する有効性の判断は、以下の式で与えられる評価値z_mの値によって判別される。

$$z_m = \sum_j \sqrt{W_{+,j}W_{-,j}}$$

Fig.2.3(a)に示すように、2 つの確率密度分布の類似度は高い時はz_mの値が大きく、クラス間で特徴量の傾向に違いが生じにくい。つまり、分布の分離がしづらいために識別には適さない特徴量となる。反対に、Fig.2.3(b)のような場合はz_mの値が小さくなり分布が分離しやすく、識別に適した特徴量であると言える。よって、多数の弱識別器候補でtラウンド目の学習において最も識別に有効な弱識別器は、確率密度分布の分離度を表す評価値z_mの値が最小値となる確率密度分布を持つ弱識別器であり、以下の式で与えられる。



(a) 分離しにくい分布例　　(b) 分離しやすい分布例

Fig.2.3 評価値z_mの確率密度分布例

$$h_t = \arg\min z_{t,m}$$

弱識別器を選択した後、その学習ラウンドで誤識別された学習サンプルが次の学習ラウンドでは正しく識別されるように、学習サンプルの重みを更新・正規化をする。重みの更新は弱識別器の出力に基づいて行われ、以下の式によって新しい重みが求められる。

$$D_{t+1}(n) = D_t(n)exp(-y_n h_t(x_n))$$

$$D_{t+1}(n) = \frac{D_{t+1}(n)}{\sum_{i=1}^{N} D_{t+1}(i)}$$

確率密度分布の作成から学習サンプルの重みの更新及び正規化までの処理を1回の学習ラウンドとし、これを一定の回数繰り返すことにより強識別器$H(x_n)$を得る。

$$H(x_n) = \text{sign}\left[\sum_{t=1}^{T} h_t(x_n)\right]$$

Real AdaBoostの強識別器は弱識別が実数を出力するために重みが必要ないので、AdaBoostの場合とは異なり上式のように表される。

2.3 検出手法の実装

画像から対象物体の検出を行うためには、画像中を網羅的に探索する必要がある。一般的に、これは検出ウィンドウが画像をラスタスキャンする形式で実装される。検出ウィンドウ内の領域のHOG特徴量を算出し、前節で構築した識別器を適用することで検出ウィンドウ内が対象物体であるか識別することが出来る。また、ラスタスキャンをする検出ウィンドウの大きさを変化させることで、様々な大きさの対象を検出することが出来る。Fig.2.4に、Real AdaBoostを用いて構築した歩行者の識別器を用いて実際に検出ウィンドウをラスタスキャンした例を示す。強識別器の出力が定めた閾値以上であった場合の検出ウィンドウを細線で表した。Fig.2.4に示されているように、1人の歩行者に対して複数の検出結果を得ていることが分かる。これは、画像を走査するラスタスキ

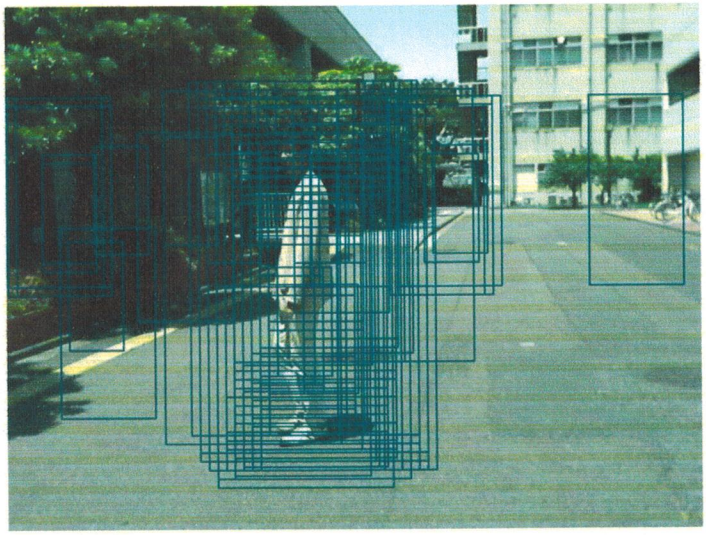


Fig.2.4 識別器のラスタスキャンによる検出結果

ャンは同じ歩行者でも区別なく何度も横切るために生じる。そこで、同じ歩行者から得られた検出結果を 1 つの検出結果として統合し、最終的な検出結果を出力することが必要となる。検出結果の統合処理には、一般的には Mean Shift クラスタリング[10]と Nearest Neighbor Rule(NN 法)が用いられる。

2.3.1 Mean Shift クラスタリング

Mean Shift はカーネル密度推定[11]を用いるロバストなデータ解析手法で、1975 年に提唱された[12]。各検出ウィンドウの中心座標を入力値として、中心点の最頻値点を算出する。検出ウィンドウの中心座標の集合を$\{x_i\}_{i=1,\dots,n}$、注目する検出ウィンドウの中心座標を始点とする注目点を$\{y_j\}_{j=1,2,\dots}$とするとき、以下の式によって注目点は最頻値に収束していく。

$$y_{j+1} = \frac{\sum_{i=1}^{n} x_i K\left(\left\|\frac{y_j - x_i}{h}\right\|\right)}{\sum_{i=1}^{n} K\left(\left\|\frac{y_j - x_i}{h}\right\|\right)} - y_j$$

ここで、hはバンド幅であり、この幅以内の集合の最頻値を計算する。本研究ではこのバンド幅を$h = 20[pixel]$とした。また、Kはカーネル関数であり、ここではガウス関数による正規分布のカーネルを用いるので以下のような関数となる。

$$K\left(\left\|\frac{y_j - x_i}{h}\right\|\right) = \exp\left(-\frac{1}{2}\left\|\frac{y_j - x_i}{h}\right\|^2\right)$$

注目点y_{j+1}の移動量が閾値以下に収束するまで繰り返し、移動量が収束した時の

y_{j+1}をクラスタリング後の検出ウィンドウの中心座標とする。以上の手順によって多数の検出ウィンドウの座標から極値を算出し、クラスタリングをすることができる。

本研究では、検出ウィンドウの座標xとyだけでなく、同様にしてウィンドウのスケールサイズに関しても考慮して3次元でクラスタリングを行うこととした。

2.3.2 Nearest Neighbor Rule による結合

Mean Shift によってクラスタリングされた検出ウィンドウの中心座標をさらに1つの点に結合させる。そこで、本研究ではNN法による結合を用いる。

NN 法はクラスタリングされた検出ウィンドウの座標上の距離を観測し、最も近い中心点を探索する。検出ウィンドウの座標の集合から中心となる任意のウィンドウに注目し、このウィンドウに最も近いウィンドウとの相対的な距離dを計算する。座標上での距離はユークリッド距離を考え、以下の式で求められる。

$$d = \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

ここで、中心座標を(x, y)、最も近い検出ウィンドウの中心座標を(x_i, y_i)とする。そして、距離dが閾値以下であった場合、2つのウィンドウを結合させる。結合する際、2つのウィンドウの中間点を新たなウィンドウの中心座標とし、結合できるウィンドウがなくなるまで処理を繰り返す。このとき、ウィンドウのサイズも 2 つのウィンドウの平均値となる。NN 法によって結合された結果を

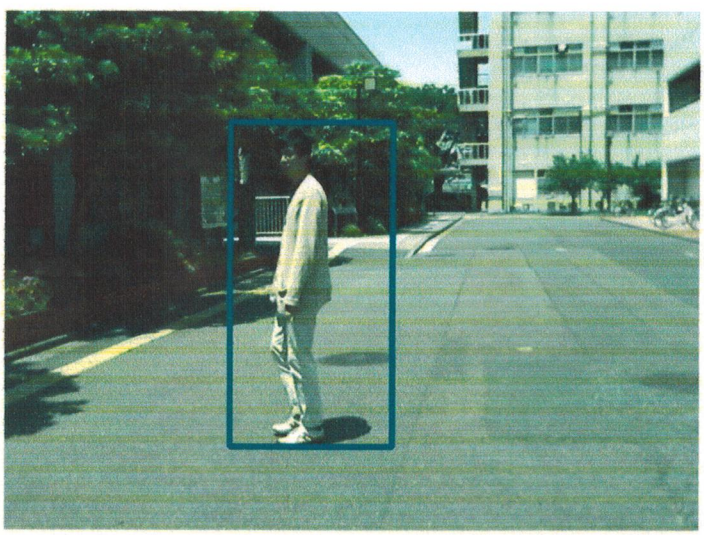


Fig.2.5 検出ウィンドウの統合結果

Fig.2.5 に記す。これらの処理により、1 人の歩行者に対して 1 つの検出結果を得ることが出来る。

第 3 章

Multiple Parts HOG Detector による歩行者検出

前章では歩行者検出のためのモデリングに用いる特徴量として、輝度情報に基づく Haar-like 特徴量と、形状情報に基づく HOG 特徴量を紹介した。一般的に、人物検出における輝度情報は服装や肌の色の違いだけでなく、場所や時間による照明の変化による影響を大きく受ける。その一方で、形状特徴は個体差や照明の変化による影響が輝度情報に比べて少ないため、HOG 特徴量に基づいた検出法は服装や周辺環境の照明に対して強固な検出をすることが出来る。このことから、本研究では HOG 特徴量に基づいた歩行者検出を行う。また、識別器の構築には AdaBoost よりも高精度の識別を行える Real AdaBoost を用いることとした。

3.1 HOG 特徴量による歩行者検出の惰弱性

本研究の目的である歩行者の検出のような、人物の全身検出は顔検出などと比較して難しいと言われている。その理由として、歩行者は様々な動作を行っていることが予想され、その動作によって歩行者の姿勢は変化することが考えられる。Fig.3.1 に示すように同一人物であっても姿勢の変化に伴い、歩行者の形状や見え方も変化をしてしまう。そのため、HOG 特徴量でもその変化に対応しきれないことが考えられる。また、実環境のように背景が複雑である場合では、人物ではない物体を人物と誤認識してしまい正しく識別することが困難になることが従来の HOG 特徴量の抱える問題であった。Fig.3.2 は実際に屋外で

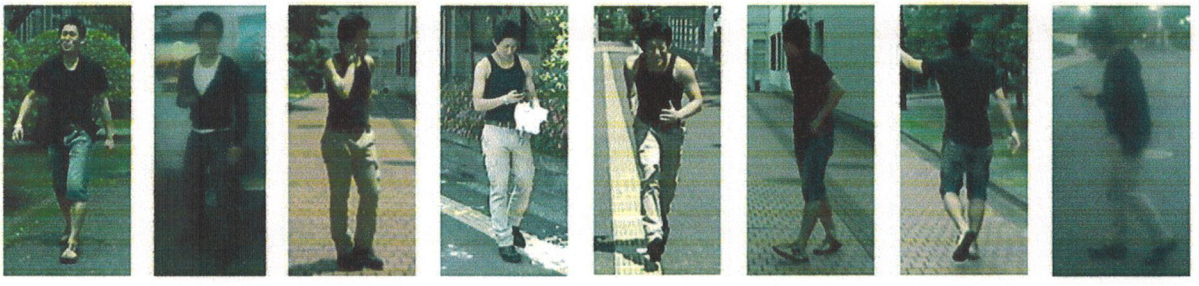


Fig.3.1 同一人物の姿勢変化による見え方の変化例

Fig.3.2 従来の HOG 特徴量に基づく検出方法による誤検出例

撮影された映像を用いて、従来の HOG 特徴量に基づく手法で歩行者の検出を行った画像を示したものである。歩行者事態の検出は行えているものの、車や木など人物以外のものも誤って検出されていることが分かる。接近する歩行者を検出して視覚障害者の歩行中の安全を確保するためのシステにおいては、誤検出による歩行者接近の誤通知は更なる事故や危険を招く可能性がある。そのためシステム実現のためには、このような誤検出を抑えることが必要とされる。

3.2 従来の HOG 特徴量による検出法からの拡張

複雑な背景環境である場合や姿勢の変化が大きい場合、従来の HOG 特徴量に基づく単純なモデリングでは人物を正確に表現するのは難しい。そこで近年、HOG 特徴量に基づく検出手法を拡張し、誤検出を抑えて検出精度を向上させる研究が盛んに行われている。

一般に、人物検出の精度を向上させるためのアプローチの手段は大きく分けて 2 種類ある。1 つは特徴量に基づく検出を行う前に背景領域と前景領域を区別し、前景領域のみを抽出して検出処理を行うという手法である[13][14]。これらの手法は、背景差分やステレオカメラを用いて事前に背景領域を区別することが出来る。同様のアプローチで、ステレオカメラの代わりに赤外線 Time of Flight 方式のカメラを使用して距離画像から人物を検出する手法も報告されている[15]。しかし、これらの方法は使用するカメラが固定カメラであったりステレオカメラであったりと、実装の際に様々な制限が存在する。一方で、もう 1 つのアプローチは特徴量に基づく人物のモデル化の方法を拡張し、複数の部分の特徴を組み合わせて精度を向上させる手法である[16][17]。複数の部分の特徴を組み合わせることでより詳細な特徴を捉えることができ、背景と人物の区別をすることが可能になる。

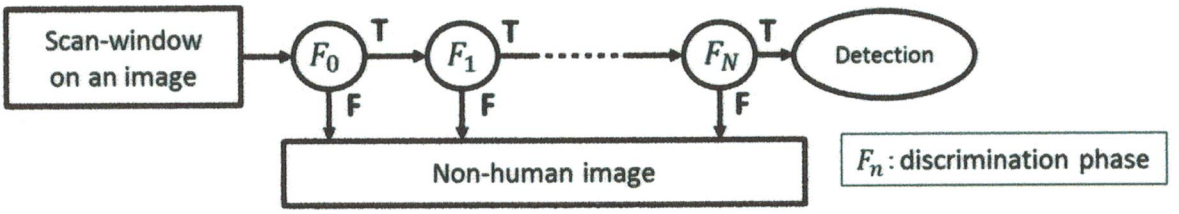


Fig.3.3 Cascade 構造の検出器

本研究は既存のモバイルデバイスでの実装を想定しているため、ステレオカメラの実装は難しい。また、カメラ自体が移動するので背景差分の使用も不可能である。以上の理由から、本研究では複数の部分を組み合わせた人物のモデル化手法により、従来の検出法の課題であった複雑な背景下での誤検出を抑制する。

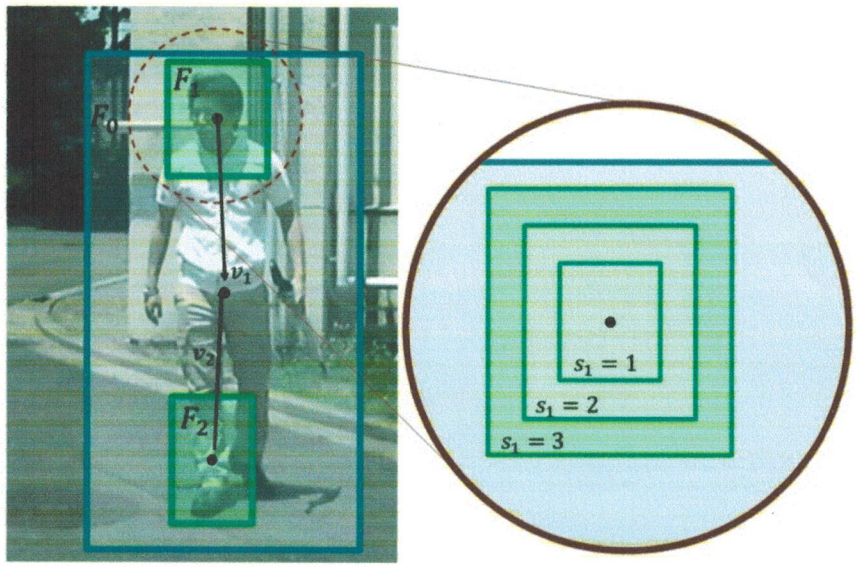
3.3 Multiple Parts HOG Detector

本研究では、複雑な背景環境における誤検出を削減する手法として **Multiple Parts HOG Detector** を提案する。**Multiple Parts HOG Detector** は人物の全身に加えて、体の部位の検出を行う。各部位の識別器が **Fig.3.3** のような **Cascade** 構造を形成し、人物を検出する 1 つの検出器となる。**Cascade** 型検出器はいずれかのステージで要素を識別する識別器が真値とならなかった時点で、以降の識別処理をやめて検出器が非人物という結果を返却する。したがって、人物の形状と類似していない領域の検出処理は早い段階で偽値となるため、効率的に処理時間を短縮することが出来る。

3.3.1 人物モデル

検出対象である人物のモデルは、人物の全身を表すベースモデルと体の部位を表すパートモデルから構成され、各モデルの特徴はベースフィルタ(全身)と、パートフィルタ(部位)によって表現される。2 種類のフィルタはいずれも **Real AdaBoost** によって構築された識別器である。本研究では全身に加えて、比較的動きの少ない頭部と距離の算出に必要となる脚部の 2 つを検出する部位とした。したがって、人物モデルは全身をモデル化したベースモデル B_0 と部位をモデル化した 2 つのパートモデル (P_1, P_2) から構成され、**Fig.3.4(a)** のような配置になる。

一般的に、特徴量の解像度を高くすることで、局所的に精密な特徴を捉えら



(a) Constitution model of human's parts　　(b) Size level of a part-model

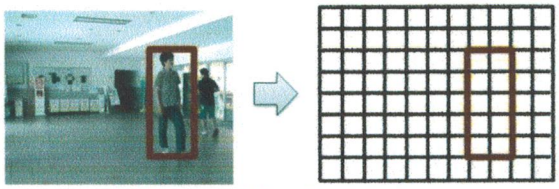
Fig.3.4 Multiple Parts HOG Detector の人物モデル

れるので高い識別性能を得ることが出来る。そのため、誤検出を減らすためには細かい解像度での検出を行う場合が多い。しかし、細かい解像度での検出はより詳細な検出が行える利点の反面、検出対象の汎用性がなくなることや、処理時間が膨大になってしまうデメリットがある。そのため、リアルタイム処理が困難になることや、誤検出が削減される分に応じて未検出が増加してしまうことが懸念される。そこで、Multiple Parts HOG Detector は大小 2 つのスケールの HOG 特徴量を用いて人物モデルを表現する。小さなスケールの HOG 特徴はベースフィルタが全身の検出を行う際に使用する。スケールが小さくなる場合、Fig.3.5 のように HOG 特徴量の解像度は粗くなる。反対に、スケールが大きくなると HOG 特徴量の解像度は細かくなり、より詳細な特徴を捉えることが可能になる。全身の特徴は姿勢などによるばらつきが予想されるために大まかな特徴を捉え、細かい解像度の HOG 特徴量をパートフィルタで使用することで体の部位の精密な特徴を効率的に捉えられるようにする。これにより、検出対象の汎用性を保ったまま局所的な部位について詳細な検出を行うことが出来る。本研究では、ベースフィルタの解像度を $cell = 8 \times 8$[pixel]、パートフィルタの解像度を $cell = 5 \times 5$[pixel]となるようにした。

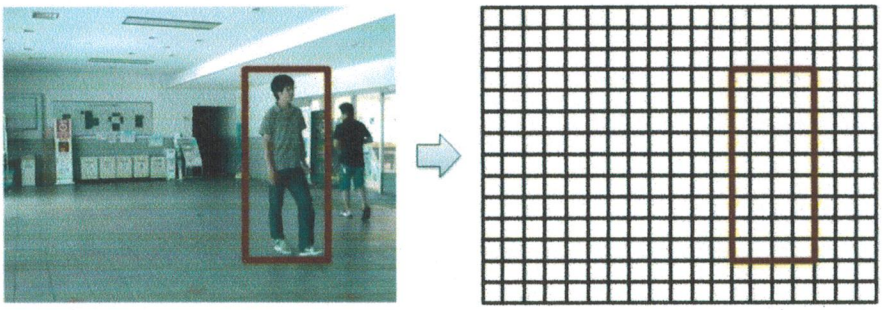
3.3.2　人物モデルの可変性

ベースモデルB_0と各パートモデルは以下のように与えられる。

$$B_0 = (F_0, x_0, s_0)$$



(a) Low resolution HOG Feature



(b) High resolution HOG Feature

Fig.3.5 スケールによる HOG 特徴量の解像度の変化

$$P_i = (F_i, v_i, s_i)$$

ここで、F_0は全身の検出を行うベースフィルタであり、x_0,はベースフィルタの中心座標、s_0はベースフィルタのサイズを表す。また、iは部位のナンバリング(頭=1,脚=2)であり、F_iは部位の識別器となるパートフィルタである。v_iはベースフィルタと部位の相対位置を表すベクトル、s_iはパートフィルタのサイズを表す。

各パートモデルは可動式であり、パートフィルタの面積の 50%以上がベースフィルタの範囲に被るように位置を変えられる。これにより、姿勢の変化にも対応することが出来る。そして、部位の検出の際には、パートフィルタは Fig3.4(b)に示されるような 3 段階のサイズを使用する。このような、パートモデルの位置とサイズの可変性によって、歩行者の姿勢の変化などによって生じる体の部位の位置や大きさのばらつきに対しても、ロバストにその特徴を捉えることが期待できる。

3.4 評価実験

Multiple Parts HOG Detector の検出性能評価のために、従来法との検出精度の比較を行った。

3.4.1 データベース作成

評価実験にあたり、学習用画像データベースを作成した。このデータベース

は特徴量を学習し、識別器を構築するのに使用される。Multiple Parts HOG Detector と従来法は同じ学習用データベースを用いて識別器の構築を行った。今回、学習用に使用した画像は様々な場所で撮影しているため、背景や照明が異なったデータになっている。学習用の画像は撮影画像から一部分を切り出したものを使用した。全身の特徴学習に使用した画像の枚数は、人物の全身が写った画像(ポジティブ画像)を 4000 枚、背景のみの画像(ネガティブ画像)を 6258 枚であった。また、Multiple Parts HOG Detector は、体の部位の検出のための識別器の構築も行う。頭部の学習のためにはポジティブ画像 1760 枚とネガティブ画像枚 3040 を用い、脚部の学習のためにはポジティブ画像 2750 枚とネガティブ画像 4500 枚を使用した。

3.4.2 実験概要

評価用データベースを用いて、従来法と Multiple Parts HOG Detector による検出を行い、検出率と誤検出率を比較する。検出を行う評価用の画像として、Fig.3.6 に示すような人物もしくは背景のみを切り出した画像それぞれ 1000 枚から構成される評価用データベースを作成した。以下、人物画像をポジティブサンプル、背景画像をネガティブサンプルとする。なお、評価用データベースと学習用データベースは全て異なる画像を使用している。

検出の有無は検出器の出力 H と閾値 λ との比較によって判断する。本実験では、従来法による予備実験において検出率と誤検出率の差が最も大きかった時の閾値、$\lambda=$ として両手法とも検出の有無の判断を行う。

n 番目のポジティブサンプルに対する識別結果に応じた出力値 z_n^+ を以下のように与え、検出率 DR を以下の式で定義した。

Fig.3.6 評価用データベース一例

$$z_n^+ = \begin{cases} 1 & H > \lambda \\ 0 & otherwise \end{cases}$$

$$DR = \frac{\sum_{n=1}^{N} z_n^+}{N} \times 100 \quad (N = 1000)$$

また、検出率DRから、ポジティブサンプルに対する未検出率MRは以下のように求められる。

$$MR = 100 - DR$$

検出率DRと同様にして、n番目のネガティブサンプルに対する識別結果に応じた出力値z_n^-を以下のように与え、誤検出率FRを以下の式で定義した。

$$z_n^- = \begin{cases} 1 & H > \lambda \\ 0 & otherwise \end{cases}$$

$$FR = \frac{\sum_{n=1}^{N} z_n^-}{N} \times 100 \quad (N = 1000)$$

3.4.3 実験結果

評価用データベースに対して本手法と従来法の両手法による歩行者検出を行い、検出率と未検出率、及び誤検出率を測定した。両手法の検出率の測定結果より、横軸を未検出率、縦軸を誤検出率としてプロットしたグラフを Fig.3.7

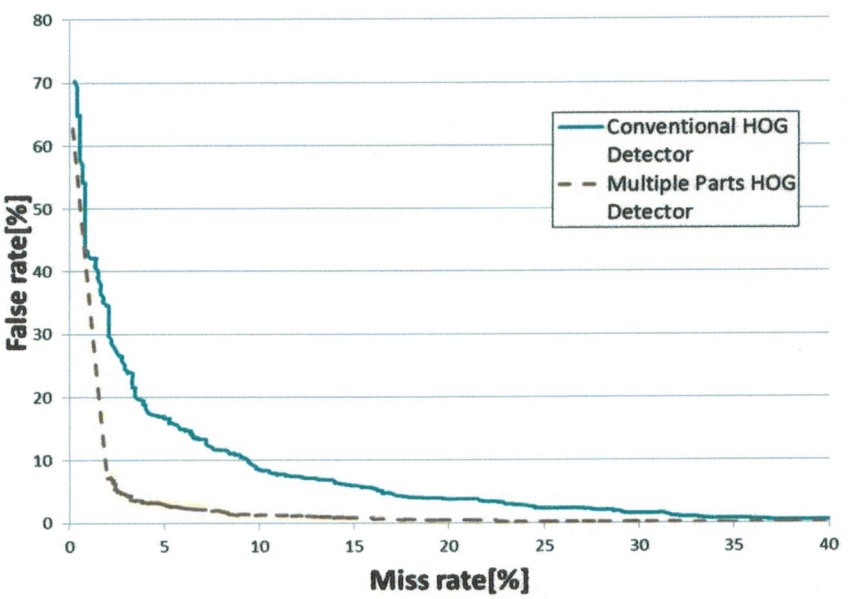
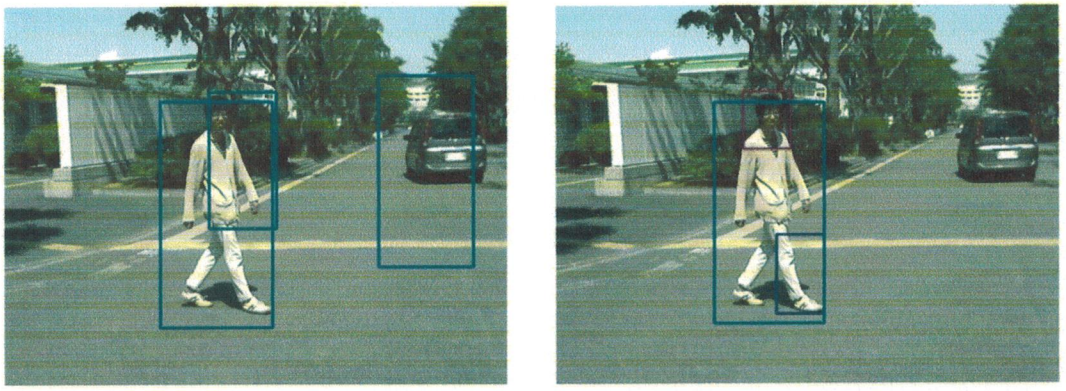


Fig.3.7 評価データベースに対する検出結果

Table.3.1 検出精度(%)

	Precision	Miss	False
Conventional HOG Detector	94.60	5.40	18.79
Multiple Parts HOG Detector	95.40	4.60	2.90



(a) Scene1

(b) Scene2

Fig.3.8 歩行者検出例

に示す。一般的に、未検出率が低くなると誤検出率が高くなる。本手法の結果は、従来法の結果と比べて左下にプロットされていることから、同程度の未検出の時でもより低い誤検出率であることが分かる。また、実験結果より、検出率が 95%付近だった時の両手法の精度を Table.3.1 に示す。従来法では、検出率が 94.60%の時に誤検出率は 18.79%となった。これに対して本手法は、検出率が 95.4%の時の誤検出率は 2.9%となり、15.89%低いという結果が得られた。以上の結果から、本手法は従来法と同程度の検出率においても、誤検出は$\frac{1}{6}$程度に抑えられていることが分かる。

また、Fig.3.8 に実際に撮影された画像から歩行者を検出した一例を示す。各シーンの左側は従来法による歩行者検出結果、右側は本手法による検出結果を

表している。従来法による検出結果には、車などの部分に誤検出を確認することが出来るが、本手法による検出結果ではそれらの誤検出が解消されていることが見て取れる。これらの結果から、本手法ではより細かい解像度でパーツの検出を行うために、従来法では誤検出された背景領域も正しく識別できていることが確認できた。

3.5 まとめ

本章では、従来の HOG 特徴量に基づく歩行者検出の課題であった背景の誤検出を抑制する手法として、人物モデルを異なる解像度の HOG 特徴量で複数のパーツから構成する Multiple Parts HOG Detector を提案した。

提案手法は精度評価実験において、従来法と同程度の検出率の場合でも誤検出率が 15.89%低く、誤検出を$\frac{1}{6}$程度に抑えられることが確認された。また、実際に撮影された画像に本手法を適用した場合も、従来法では見られた誤検出を解消して歩行者を検出できることが示された。以上のことから、提案手法である Multiple Parts HOG Detector は、従来法の課題であった背景の誤検出を抑えることが出来るため、本研究のシステムのような前処理を用いない歩行者検出に有効であると言える。

第 4 章

走査領域特定による処理時間短縮

前章で紹介した Multiple Parts HOG Detector は、複雑な背景環境においても背景の誤検出抑制効果が高い検出を行うことが出来るが、1 フレームに要する処理時間が長いことが問題となる。視覚障害者の歩行中の安全確保のためのアプリケーションとしての人物検出には、リアルタイム検出を行うことが必要不可欠である。そこで、本研究では Multiple Parts HOG Detector が画像を走査する領域を削減することで、処理時間の短縮を図る。画像の走査範囲を効率的に削減するには、検出対象である歩行者の次のフレームでの位置を予測・特定することが有効である。そこで、本研究では時系列フィルタリングによる歩行者の移動軌跡の追跡を行い、次フレームでの位置を予測して検出器の走査範囲の特定を行うこととした。

4.1 時系列フィルタリング

本研究は、歩行者の追跡と移動位置予測に時系列フィルタリングを用いる。時系列フィルタリングは、過去に観測された信号から未来の値を予測する場合や、ノイズを含む不完全な観測データから対象の状態を推定する場合に有効であることが知られている。

時刻 t における追跡対象の状態量を χ_t、観測値を z_t とし、また、これまでの観測データを $Z_t = \{z_0, z_1, \ldots, z_t\}$ とするとき、時系列フィルタリングによる状態推定は事後分布 $p(\chi_t|Z_t)$ を推定する問題として定式化される。事後分布 $p_t(x_t|Z_t)$ の状態推定は、次のベイズの定理により推定することが出来る。

$$p(\chi_t|Z_t) = \frac{p(z_t|\chi_t)p(\chi_t|Z_{t-1})}{p(z_t|Z_{t-1})}$$

ここで、$p(x_t|Z_{t-1})$ は時刻 t における事前分布を表す。また、$p(z_t|\chi_t)$ は尤度であり、観測データから推定される。推定対象の状態がマルコフ過程に従うと仮定したとき、事前分布 $p(\chi_t|Z_{t-1})$ はチャップマン・コルモゴロフ(Chapman-Kolmogorov)方程式により以下のように計算される。

$$p(\chi_t|Z_{t-1}) = \int p(\chi_t|\chi_{t-1})p(\chi_{t-1}|Z_{t-1})d\chi_{t-1}$$

これにより逐次的な状態推量が可能となり、対象の追跡や予測に利用することが出来る。

時系列フィルタリングには、観測モデルやシステムモデル、ノイズ分布モデルなどによっていくつかの種類が存在する。ここでは、時系列フィルタリングの一種として代表的なカルマンフィルタ [18]とパーティクルフィルタ[19]について述べる。

4.1.1 カルマンフィルタ

カルマンフィルタは 1960 年に Kalman が提案したアルゴリズムであり[18]、システムモデルと観測モデルが時間発展に対して線形、ノイズの分布モデルがガウス分布であるという過程の下で、推定誤差を最小とする最適な解を求めることが出来る。カルマンフィルタのアルゴリズムは、状態の推定と予測の 2 つの部分から構成される。

システムモデルと観測モデルの時間発展過程が線形である場合、状態方程式と観測方程式は以下の式で与えられる。

$$\chi_{t+1} = F_t\chi_t + G_t w_t$$

$$y_{t+1} = H_t\chi_t + v_t$$

ここで、F_tはシステムモデルを表す状態遷移行列、H_tは観測モデルを表す観測行列である。また、w_tとv_tはシステムノイズと観測ノイズである。この時事前分布と事後分布はガウス分布となるため、時刻tにおける推定値$\hat{\chi}_{t|t}$は以下の式で線形に計算できる。

$$\hat{\chi}_{t|t} = \chi_{t|t-1} + K_t\left(y_t - H_t\hat{\chi}_{t|t-1}\right)$$

$$\hat{\chi}_{t|t-1} = F_{t-1}\hat{\chi}_{t-1|t-1}$$

このとき、K_tはカルマンゲインと呼ばれる行列式であり、観測の信頼度に応じた重みである。カルマンゲインはw_tとv_tの共分散行列Q_tとR_tを用いて以下の式で与えられる。

$$K_t = P_{t|t-1}H_t^T\left(H_tP_{t|t-1}H_t^T + R_t\right)^{-1}$$

ただし、$P_{t|t-1}$は解析誤差の共分散行列、H_t^TはH_tの随伴行列である。

4.1.2 パーティクルフィルタ

パーティクルフィルタ事前分布と事後分布を、多数の標本サンプルである重み付きの粒子サンプル$s^{(n)}$を用いて離散的な仮説群$S = \{s^{(n)}, \pi^{(n)}; n = 1, \ldots, N\}$として近似表現する。ここで、$\pi^{(n)}$は粒子サンプルの重みを表す。これにより、観測モデルが非線形である、もしくはノイズが非ガウス型である確率分布に対して強固な推定が可能になる。

パーティクルフィルタの処理は Fig.4.1 に示すような以下の手順によって行われる。

(ⅰ)粒子サンプル選択(Selection)

パーティクルフィルタでは粒子の抽出を行う。時刻$t-1$における事後分布$p(\chi_{t-1}|Z_{t-1})$をN個の粒子サンプル群$\{s_{t-1}^{(1)}, \ldots, s_{t-1}^{(N)}\}$の各粒子サンプルの重み$\{\pi_{t-1}^{(1)}, \ldots, \pi_{t-1}^{(N)}\}$に従い、重みの大きな粒子のみを抽出し、重みの小さな粒子を消滅させる。ここで、χ_{t-1}追跡対象の状態量を、$Z_{t-1} = \{z_0, z_1, \ldots, z_{t-1}\}$はこれまでの観測データを表す。そして、抽出した粒子の重みの比に基づいてその粒子を複製し、新たな粒子群$\{s'^{(1)}_{t-1}, \ldots, s'^{(N)}_{t-1}\}$を選択する。

(ⅱ)状態遷移モデルに基づく予測(Prediction)

選択ステップで抽出された粒子群$\{s'^{(1)}_{t-1}, \ldots, s'^{(N)}_{t-1}\}$を、事前に定義した状態遷移モデルに基づいてそれぞれ移動させることで、時刻tにおける各粒子の位置を予測した粒子$\{s_t^{(1)}, \ldots, s_t^{(N)}\}$、事前分布$p(\chi_t|Z_{t-1})$を近似表現する。一般的に、状態遷移モデルは対象の時間発展による変化を運動モデルによる移動にランダムノイズを含めるもので構成される。

(ⅱ)観測による重み付け(Weighting)

粒子の重み付けのために時刻tにおける尤度を求める。尤度の算出は観測モデルに従って行われ、以下の式により粒子$s_t^{(n)}$の重み$\pi_t^{(n)}$を算出する。

$$\pi_t^{(n)} = \frac{p(z_t|\chi_t = s_t^{(n)})}{\sum_{n=1}^{N} p(z_t|\chi_t = s_t^{(n)})}$$

以上のステップを毎時刻繰り返すことで、対象の追跡を行う。

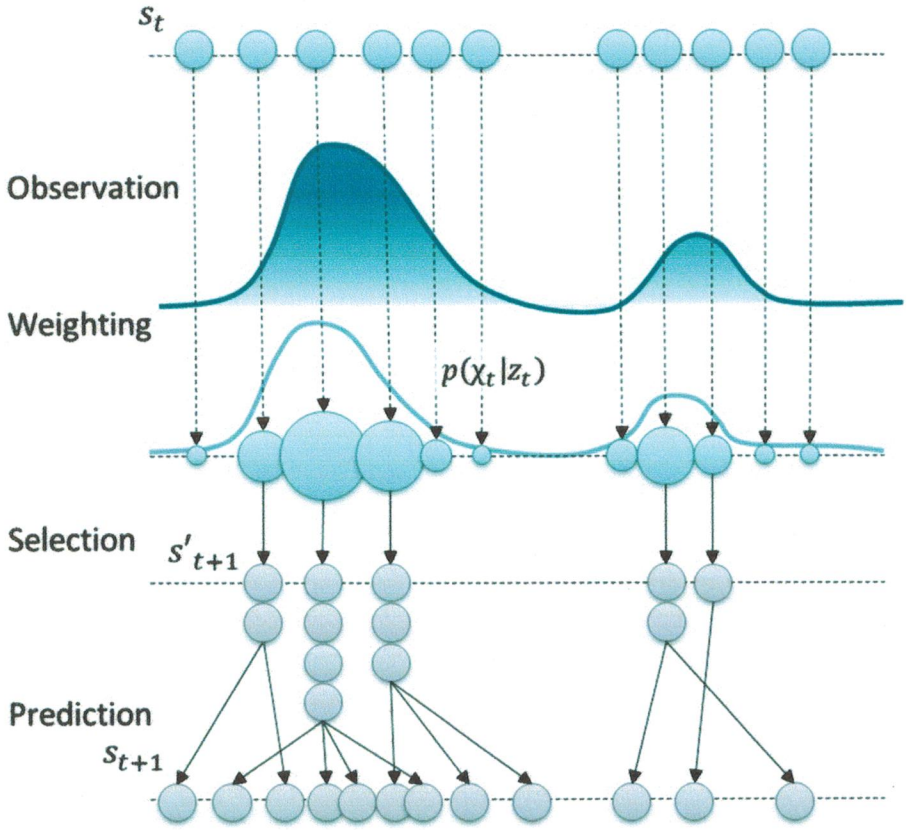


Fig.4.1 パーティクルフィルタの計算フロー

4.2 パーティクルフィルタを利用した歩行者追跡と移動予測

撮影された映像内の歩行者の移動軌跡は、歩行者の動きやカメラを持ったユーザーの動きなど様々な影響を受けるために、ノイズを完全な線形モデルでは表現できないことが予想される。そのため、本研究では非線形のシステムモデルに対しても状態推定を行うことが出来るパーティクルフィルタを用いて歩行者の追跡と移動予測を行う。

人物追跡を行う際の観測モデルには人物の姿勢や照明条件、複雑な背景などの環境条件に対してロバストな特徴量であることがもとめられる。これまで、観測モデルとして多く利用されてきたものとしては、色分布情報、エッジ情報、背景差分による前景領域情報などが挙げられる。しかし、これらの特徴量では前述した環境条件に対してロバストな値を得ることが難しい。そこで、本研究では前章で紹介した複雑な環境下でもロバストな結果を得ることが出来る **Multiple Parts HOG Detector** に着目した。人物検出結果に基づく検出器の出力を観測モデルとすることで人物の姿勢や照明条件に対しても、頑強な追跡を行えることが期待できる。なお、ここでいう検出器の出力とは、検出時に統合

された複数の強識別器の出力のことを指す。

検出器の出力を観測モデルに利用したパーティクルフィルタによる歩行者追跡を以下のように適用する。

4.2.1 状態遷移モデルの設定

歩行者の動きを追跡するためには、早い動きにも対応する必要がある。そこで、本研究では線形な運動モデルによるシステムモデルとして、状態遷移モデルに等速直線運動を使用する。線形予測モデルは、対象の動きが予測モデルにあっていれば効率的なサンプリングが可能である。本研究の場合、画像上の二次元位置の追跡を行うので、時刻tにおける追跡対象の状態量はχ_t以下のように与えられる。

$$\chi_t = [x_t \; y_t \; \dot{x}_t \; \dot{y}_t]$$

ここで、x_tとy_tは歩行者の位置座標、$\dot{x}_t$と$\dot{y}_t$は歩行者の各成分の速度とする。これにより、等速直線運動による状態遷移モデルは以下の式で定義できる。

$$\chi_t^{(n)} = F\chi_{t-1}^{(n)} + w_t^{(n)}$$

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

ただし、この時システムノイズ$w_t^{(n)}$はランダムに生成される。

4.2.2 尤度の算出

尤度の算出は、粒子サンプル$s_t^{(n)}$を状態パラメータとしたときに観測モデルが入力画像に適合しているかを評価することで行う。バウンディングボックスとして画像上を走査する検出器は、走査位置ごとに学習した人物特徴との適合度に応じた値を出力する。様々なサイズの歩行者を検出するために、検出器は様々なサイズのバウンディングボックスで何度も画像上を走査する。そのため、各サイズの検出器の出力をマッピングし、平滑化を行うことで、歩行者の存在する付近に高いピークを持つ出力分布を得ることが出来る。つまり、この出力分布を観測モデルとすることで、歩行者の存在する位置に高い尤度を持つモデルを得ることが出来る。

4.3 パーティクルフィルタによる走査領域の特定

1フレームの処理時間削減のために、本研究はパーティクルフィルタによって散布された粒子の結果に基づいて検出器の画像走査範囲の特定を行う。次フレームの走査範囲の特定を行うには、次のフレームでの歩行者の位置と見た目の大きさを予測し、検出器が走査する範囲とその位置を決定する必要がある。本研究では、パーティクルフィルタの予測処理における粒子の散布状態から走査範囲の大きさと位置を推定する。

検出した歩行者の見た目の大きさが大きい場合、検出器の出力する範囲も広くなるため、尤度は広い範囲に亘って分布する。これに伴い、粒子サンプル選択の過程では広い範囲に亘って粒子が複製される。そのため、状態遷移モデルに基づいて散布された粒子サンプル群は歩行者の見た目の大きさが大きい場合には広い範囲に分散する。以上の検出した歩行者の見た目の大きさと粒子サンプルの分散の関係から、本研究では状態遷移モデルに基づいて散布された粒子サンプル群の分散度によって、次のフレームでの検出器の画像を走査する領域の大きさを決定する。また、散布された粒子サンプル群の重心点を求めることで、算出された走査範囲の画像上での位置を決定する。

状態遷移モデルに基づいて散布されたN個の粒子サンプルの分散度PDを、Fig.4.2に示すようなモデルとして以下の式で定義する。

$$PD = 1 - \frac{\sum_{n=1}^{N} |x_n - x_c|}{NR}$$

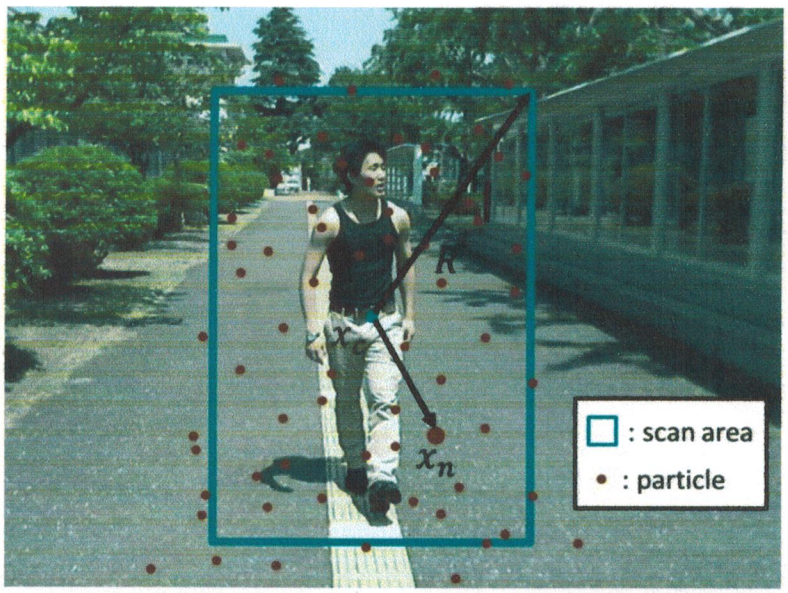


Fig.4.2 追跡結果に対する粒子の分散度定義モデル図

ここで、x_nはn個目の粒子サンプルの座標、x_cは粒子サンプル群の重心座標を表す。また、Rは現在のフレームで走査した領域(矩形)の中心点から頂点までの長さを表す。そして、算出した分散度PDにより決定される次フレームの走査領域の大きさL_{t+1}は、予備実験によって以下のように定義した。

$$L_{t+1} = \begin{cases} \frac{1}{2}L_t & PD < 0.3 \\ \frac{3}{4}L_t & 0.3 \leq PD < 0.5 \\ \frac{5}{4}L_t & 0.5 \leq PD < 0.7 \\ \frac{3}{2}L_t & 0.7 \leq PD \end{cases}$$

時刻$t = 0$の場合は、散布された粒子サンプルの座標からx_{min}、x_{max}となる座標を求め、全ての粒子が内部に含まれるような走査範囲とする。

また、上記の式で決定した走査範囲を画像上のどの位置に適用するかを、散布された粒子サンプル群の重心x_cによって決定する。今回、粒子サンプル群の重心を各粒子の重みによる平均座標とし、以下の式で定義した。

$$x_c = \sum_{n=1}^{N} \pi_n x_n$$

ここで、π_nは 0.0 から 1.0 の範囲に正規化された粒子の重みを表す。算出された重心x_cを走査範囲の中心点として、次フレームの走査領域の位置を決定する。

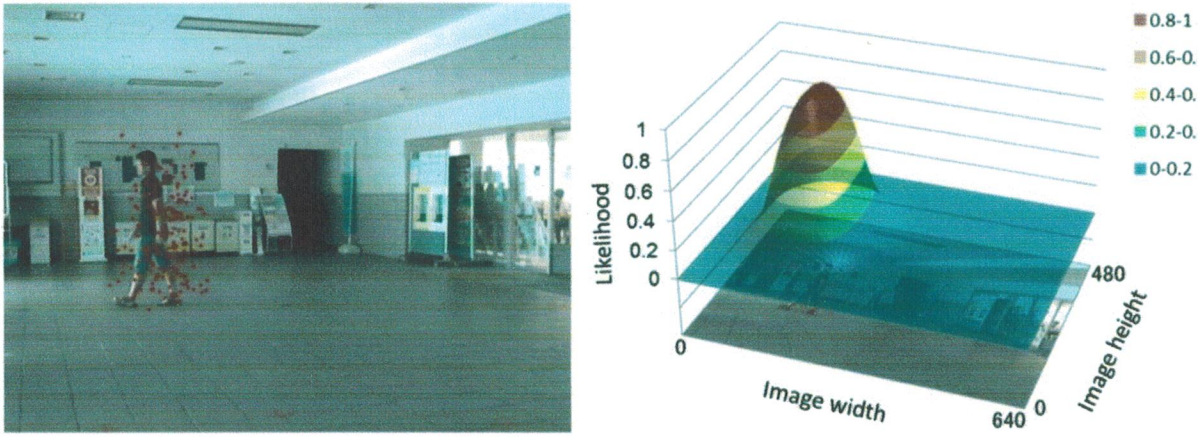
4.4 評価実験

パーティクルフィルタによる歩行者の追跡と走査領域特定による処理時間短縮効果を検証するための実験を行った。今回は簡単のために、画像内で追跡する対象となる歩行者の人数は 1 名とした。また、粒子サンプルの総数は$N = 500$とした。

4.4.1 追跡評価

パーティクルフィルタによる歩行者追跡の精度評価を行った。撮影された画像から歩行者の検出を行い、その結果を基に尤度分布を算出して粒子サンプルの散布を行う。本実験における人物検出は、前章で紹介した Multiple Parts HOG Detector による検出結果を使用した。

Fig.4.3 に画像から歩行者を検出した時の検出器の出力から求めた尤度分布



(a) The original image　　(b) Likelihood map

Fig.4.3 検出結果に基づいた尤度分布

Fig.4.4 歩行者追跡結果例

の一例を示す。求められた尤度分布は、検出結果画像上の歩行者が検出された部分にピークを持つように分布していることが見て取れる。このことから、これらの尤度分布に基づいて粒子サンプルの散布を行うことで、粒子サンプル選択において複製された粒子が検出された歩行者の付近に分布し、歩行者の追跡が可能であることが窺える。また、Fig.4.4 に実際にパーティクルフィルタを適用した結果画像を 4 フレーム分示す。図中の小さな赤い点が粒子選択過程にお

Table.4.1 検出結果とパーティクルフィルタによる追跡結果の平均誤差

	Prediction		Tracking	
	pixels	proportion	pixels	Proportion
Error	18.78	0.13	11.18	0.08

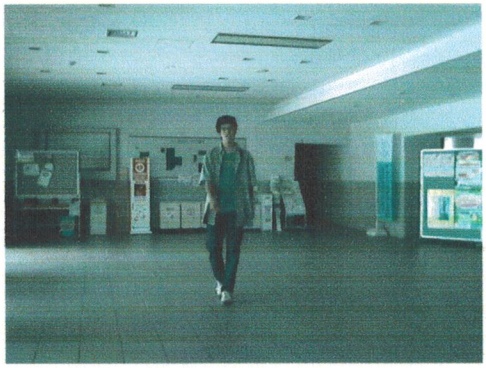
いて抽出・複製された粒子、大きな青い点がこの粒子群の重心を表す。結果画像の各フレームにおいて、粒子は歩行者の周辺に散布されていることが見て取れる。これらの結果から、パーティクルフィルタによって検出した歩行者を追跡できていることが確認できる。

また、検出した歩行者を囲む枠の中心と粒子群の重心との平均誤差と標準偏差を求めた結果を Table.4.1 に示す。今回、枠の中心と重心との距離として、ピクセルを単位としたユークリッド距離を計算した。また、誤差の比率として、歩行者を検出した枠の中心から頂点までの長さに対する誤差の割合を算出した。検出結果に基づく尤度分布によって実質的に追跡を行う Selection 過程で散布された粒子サンプルは、歩行者との平均誤差は 11.18pixel であり、歩行者領域に対する比率の平均は 0.08 になる。このことから、粒子サンプル群は歩行者の追跡を高い精度で行うことが出来ているのが分かる。また、Prediction 過程によって予測散布された粒子群と、実際に歩行者が検出された位置の誤差は 18.78pixel であった。これは、歩行者領域に対して 0.13 の平均比率になり、粒子サンプルが次フレームの歩行者の検出位置を中心として散布していることが分かる。このことから、予測処理によって歩行者の検出位置を絞り込むことが出来ているため、予測結果から走査領域の特定を行うことの有効性が示唆される。

以上の結果から、検出器の出力を尤度に割り当てたパーティクルフィルタの適用によって、検出した歩行者を追跡することが確認できた。また、予測処理において、歩行者の次のフレームでの検出位置を高い精度で特定できていることも確認することができた。

4.4.2 走査域特定による処理時間短縮効果評価

散布された粒子の状態から次フレームの走査領域を特定の行うことによる、処理時間の短縮の効果を評価する実験を行った。実際に路上において異なる Fig.4.5 に示すような 2 つのシーンを撮影し、シーン中から歩行者 1 名が写っている連続した 50 フレームを抜粋した。抜粋したシーンに対して、通常の人



(a) Scene1

(b) Scene2

Fig.4.5 実験に使用した2つのシーン例

Table.4.2 平均処理時間(msec)

		Time
Multiple Parts HOG Detector	Scene1	1689.92
	Scene2	1924.52
	Average	1807.22
Multiple Parts HOG Detector + Particle Filter	Scene1	680.51
	Scene2	545.38
	Average	612.95

物検出とパーティクルフィルタによる走査領域特定を用いた検出の両手法を適用し、1フレームに対する平均処理時間を計測した。人物検出には Multiple Parts HOG Detector を用いる。また、撮影に使用する画像の解像度は640×480pixel とした。

Intel Core i7 CPU 2.93GHz を計算機として用いた場合の実験結果を Table.4.2 に示す。走査域特定を用いない場合、2 シーンでの平均処理時間は1807.22msec となり、約 0.55fps の処理速度になった。この結果に対して、走査域特定を用いた場合は 2 シーンでの平均処理時間は 612.95msec を記録し、最大で約 1.83fps の処理速度となった。この結果から、パーティクルフィルタを用いた走査領域の特定処理によって、約 67%の処理時間の短縮が可能であることが確認できた。

4.5 まとめ

本章では歩行者の追跡と移動予測による、検出器の画像走査領域を削減する

手法を取り入れた。検出器による歩行者検出結果を基にしたパーティクルフィルタの適用によって、検出した歩行者の追跡と移動予測が可能であることを確認した。さらに、パーティクルフィルタの粒子サンプルの散布状態から、次のフレームでの歩行者の位置を予測し検出器が画像を走査する範囲を算出した。取り入れた手法は評価実験において、走査領域特定を用いない場合に比べて処理時間を約 67%削減することが出来た。これらの結果から、検出器の出力を尤度としたパーティクルフィルタを用いた検出器の走査領域の特定は、効率的に画像の走査範囲を削減し歩行者検出の処理時間を短縮できるため、リアルタイム動作のために時間短縮に有効であると考えられる。

第 5 章

歩行者の接近判断

前章までは撮影したカメラ映像から歩行者を検出する手法について触れてきた。本研究が目的をしている視覚障害者支援システムは、前方の歩行者との衝突の危険判断を行う。そのため、検出結果から歩行者までの距離を推定することで、歩行者がどの程度接近しているか感知する必要がある。

そこで、本章では歩行者の検出結果を用いた距離推定を行う手法、及び本研究の距離算出システムの手法として、検出した歩行者の足の位置を基にして距離を算出する手法を提案する。本研究における歩行者検出システムは単眼カメラで撮影された画像を使用しているため、単眼画像を使用した歩行者までの距離を推定と歩行者の接近の判断を行う。

5.1 画像情報を用いた距離算出法

一般的に、対象までの距離を測定するためには赤外線や超音波などを利用した距離センサーを用いることが多い[20][21]。その一方で、近年は画像処理技術の発展を受けて画像から距離を算出する手法も注目され、盛んに研究が行われている[22][23]。

人間は視覚的な情報から距離・奥行きを知覚する際に様々なアルゴリズムを使用する。それにならって、画像から距離を算出するアルゴリズムはこれまでにいくつか提案されている。ここでは、その中でも代表的である、視差を利用した手法と線遠近を利用した手法について述べる。

5.1.1 ステレオ画像

ステレオ画像とは複数のカメラを用いて異なる視点から撮影された画像を指す。異なる視点から同一の対象を見たとき、カメラの視点によって画像上の位置が異なる。三角測量の原理に基づいて、この位置の違いから 3 次元の距離を逆算する。ステレオ画像による距離推定には最低でも 2 台のカメラが必要であり、Fig.5.1 のようにカメラを 2 台使用した手法が一般的である[22]。

5.1.2 透視法

透視法は 3 次元空間を 2 次元平面に変換する手法として、コンピュータグラ

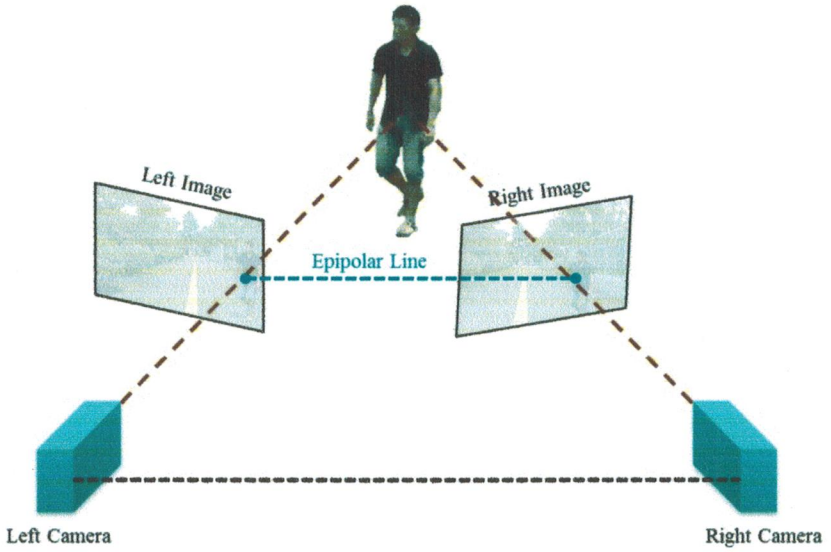
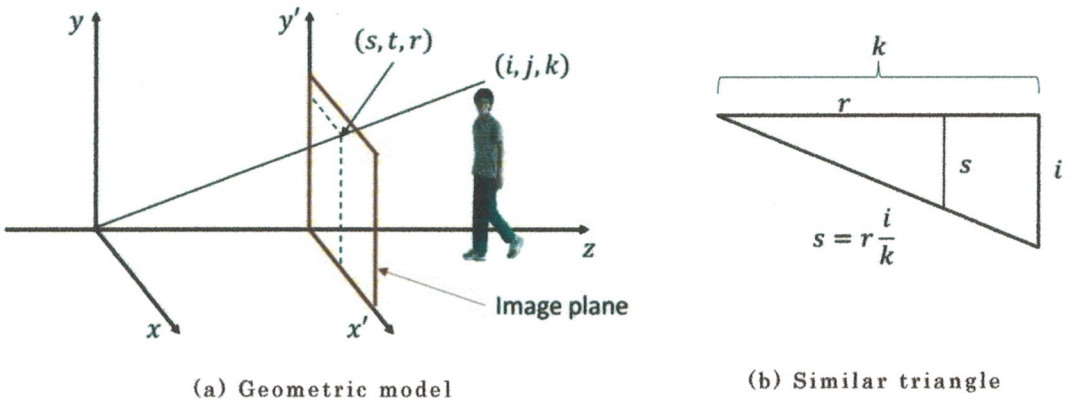


Fig.5.1 ステレオカメラシステム



(a) Geometric model　　(b) Similar triangle

Fig.5.2 透視投影モデル図

フィックスなどの分野で多く使用される。Fig.5.2 に示すような透視投影モデルでは、対象の 3 次元座標を三角形の相似関係の性質を利用して、2 次元平面上に投影した場合の座標を算出する。これにより、対象までの距離を算出することが出来る。透視法による距離の算出に必要なカメラは 1 台のみでよく、複数のカメラを用意する必要がない。

5.2 ピンホールカメラモデルによる距離推定

本研究の目的である視覚障害者支援システムは、歩行者検出に使用するセンサーを単眼カメラのみとしているためステレオ画像を用いることが出来ない。そこで、本研究では単眼カメラ画像からでも距離の推定を行える透視法に基づいて、ピンホールカメラモデルによる距離の推定を行う。

ピンホールカメラモデルによる距離の算出には大きく、対象の大きさから距

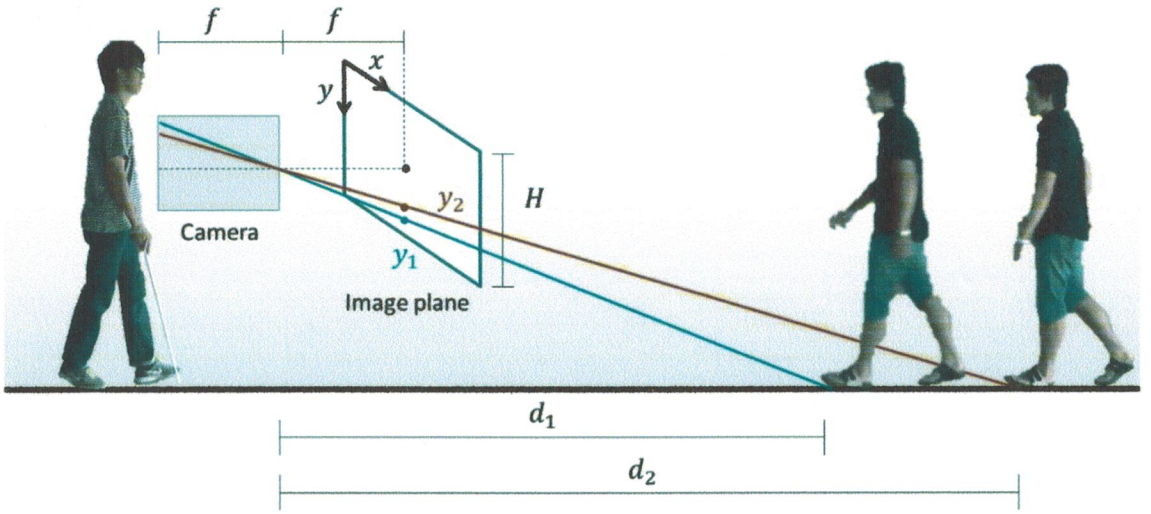


Fig.5.3 ピンホールカメラモデルによる画像と距離の幾何関係

離を逆算する手法と対象の座標から距離を逆算する手法の 2 つがある。画像上での歩行者の見た目の大きさは、歩行者の身長による個人差や姿勢による誤差が大きいことが考えられる。そのため、距離のみに依存した安定な値を得られず、算出された距離には多くの誤差が含まれることが懸念される。これに対して、対象の座標から距離を逆算する手法は歩行者の身長等の個体差が生じにくい。特に、歩行者と地面との接地点は身長や姿勢に左右されることのない、距離のみに依存した安定な値である。そこで、本研究では地面との接地点である足の座標を基に、歩行者の存在する距離の推定を行う。歩行者の足の座標は、**Multiple Part HOG Detector** によって検出された脚部の下辺座標を使用する。**Fig.5.2** にピンホールカメラモデルによる歩行者の足の座標と距離の幾何学モデルを示す。今回は簡単のために、カメラの高さは固定され、地面と平行であると仮定した。

カメラの高さをh、歩行者までの距離dとすると、画像上での歩行者の足の位置yは三角形の相似の関係から、以下の式で求められる。

$$y = \frac{fh}{d} + \frac{H}{2}$$

ここで、fはピクセル単位でのカメラの焦点距離、Hは画像の縦幅を表す。上式を変形することで、足の座標yから歩行者までの距離dを算出する以下の式が得られる。

$$d = \frac{2fh}{2y - H}$$

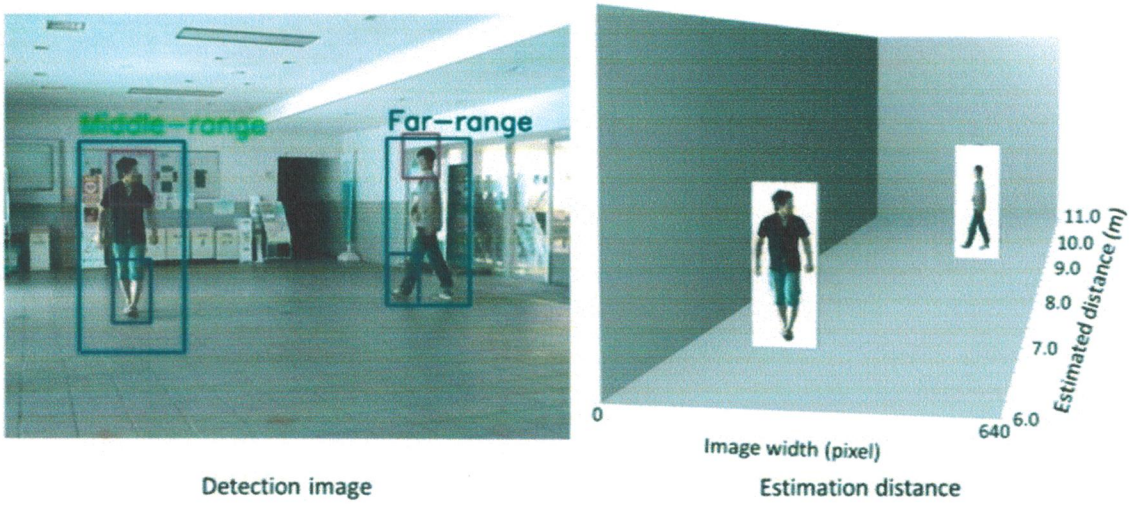
Fig.5.4 画像と距離の関係と 3 段階の距離レベル

Fig.5.3 から、距離d_1に存在する歩行者の足は画像上では高さy_1の位置に投影される。対して、それよりも遠い距離d_2に位置する歩行者の足は、画像上の高さy_2の位置に投影される。$y_1 > y_2$となることから、歩行者までの距離が離れるほどに画像上での歩行者の足の位置は高い位置に投影されることが示された。この距離と座標を基にして、本研究では Fig.5.4.に示すように、歩行者の存在する位置を 3 段階のレベルに区別する。4m 以下の距離を Near range、8m 以上の距離を Far range、その中間の位置を Middle range とした。この位置レベルを、歩行者がユーザーに接近している程度を判断する目安として使用する。

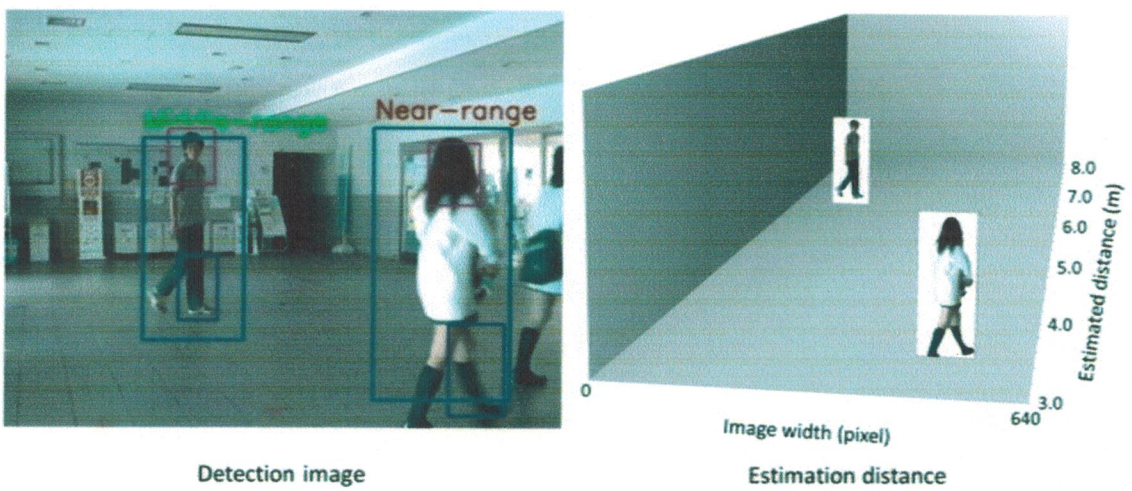
5.3 評価実験

ピンホールカメラモデルによる、歩行者検出結果を用いた距離の推定と接近の判断の精度評価のための実験を行った。撮影した画像から歩行者の検出を行い、その結果から距離の推定を行う。なお、今回はカメラを$h = 1.4$m の高さに地面と平行となるように固定して撮影を行った。また、予備実験により使用したカメラの焦点距離は$f = 540$pixel であること定めた。

Fig.5.5 に撮影した画像から歩行者の距離の推定と距離レベルの区分を行った結果の一例を示す。両図とも画像内には 2 名の歩行者が存在するが、検出結果から得た各歩行者の足の位置から距離を推定している。そして、画像内の 2 名の歩行者の距離の遠近関係を正しく推定することが出来ていることが示されている。また、推定した距離の結果から、各歩行者を位置レベルによって分類することが出来た。



(a) Scene A



(b) Scene B

Fig.5.5 検出結果に基づく距離推定と距離レベル判別例

Table.5.1 各距離レベルでの推定距離の誤差(m)

	Error
Near range	0.30
Middle range	0.20
Far range	0.70
Average	0.38

また、撮影した画像から 88 枚を抜粋して、推定した距離と実距離との誤差を測定した。3 段階の各距離レベルにおける平均誤差を Table.5.1 に示した。

近傍の 2 レベルでの平均誤差はそれぞれ 0.30m、0.20m であったのに対して、遠方の Far range での平均誤差は 0.70m となり、約 2.8 倍の誤差となった。画像上の座標と歩行者の距離は反比例の関係であるため、検出時のピクセルのずれが大きな距離誤差につながることが要因であると考えられる。3 レベル通しての平均誤差は 0.38m となり、およそ人間の歩幅 1 歩分程度であるため、歩行者との衝突を防ぐための接近の程度を判断するのには十分な精度であると考えられる。

5.4 まとめ

本章では単眼画像での歩行者検出結果から接近の程度を判断するシステム実現のために、ピンホールカメラモデルによる検出した歩行者の足の座標を利用した距離の算出法を提案した。提案手法によって、Multiple Parts HOG Detector による歩行者の検出結果から歩行者までの距離を推定することができ、その距離に応じて接近の程度の目安となる距離レベルに分類することが出来た。また、各距離レベルにおける推定距離と実距離との平均誤差は、全体で 0.38m でありおよそ人間の歩幅 1 歩分程度であるため、歩行者との衝突を防ぐための接近の程度を判断するのには十分な精度であると言える。

第 6 章

むすび

本論文では、視覚障害者の自立歩行支援のためのシステム実現のために、単眼カメラ画像からの前方の歩行者の検出と歩行者接近の程度を判断するための検出結果からの距離の推定を行う手法について述べた。

歩行者の検出方法として、本研究では人物モデルを異なる解像度の HOG 特徴量で複数のパーツから構成する Multiple Parts HOG Detector を作成した。提案手法は、従来法の課題であった背景の誤検出を抑える効果があり、評価実験において誤検出を従来の 0.17 倍程度に抑えられることが示された。

しかし、提案手法は高精度を持つ一方で処理時間が多くなる問題があった。そこで、歩行者の追跡と移動予測による、検出器の画像走査領域を削減する手法を取り入れた。検出器による歩行者検出結果を基にしたパーティクルフィルタの適用によって検出した歩行者の追跡と移動予測を行い、次フレームでの検出器の走査領域を特定した。本手法は評価実験において Multiple Parts HOG Detector の処理時間を約 67%削減することが出来た。

そして、歩行者の検出結果から接近の程度を判断するために、ピンホールカメラモデルによる検出した歩行者の足の座標を利用した距離の算出法を提案した。提案手法を用いて、検出結果から歩行者までの距離を推定し、その距離に応じて接近の程度を距離レベル別に分類することが出来た。また、精度評価実験において、推定距離と実距離との平均誤差は 0.38m と、接近の程度を判断するには十分な精度を記録した。

今後の課題として、歩行者がカメラを携帯する場合には、カメラ自体が傾くことが予想されるため、カメラ自体の傾きを考慮した距離の推定を検討する必要がある。また、実環境下で本システムを使用する場合には、複数の歩行者の検出と追跡を同時に行う必要がある。そのため、パーティクルフィルタによる複数の対象追跡を取り入れることが求められる。そして、スマートフォンのようなモバイルデバイスでのリアルタイム検出を実現するためには、メモリの削減が必要になる。メモリ削減の手法として、HOG 特徴量の量子化や 2 値化が挙げられる。

参考文献

[1] 厚生労働省社会・援護局 障害保健福祉部 企画課統計調査係,"平成 18 年身体障害児・者実態調査結果", 2008

[2] A.Helal, S.Moore, and B.Ramachandran. Dridhti, "An integrated navigation system for the visually impaired and disabled," Fifth International Symposium on Wearable Computers (ISWC01), pp. 149-156, 2001.

[3] S.Krishna, D.Colbry, J.Black, V.Balasubramanian, and S.Panchanthan, "A Systematic Requirements Analysis and Development of an Assistive Device to Enhance the Social Interaction of People Who are Blind or Visually Impaired," *Workshop on Computer Vision Applications for the Viually Impaired(CVAVI 08), ECCV 2008.*

[4] 杉村大輔, "行動履歴を反映させた適応的環境属性を伴う三次元人物追跡", 東京大学 平成 19 年度 修士論文

[5] 金森証, 小谷一孔, "Wavelet 変換を用いた顔距離画像の特徴解析に関する研究：顔距離画像の Wavelet 係数による個人性の抽出", 映像情報メディア学会技術報告書 25(84), 41-46, 2001.

[6] P.Viola and M.Jones , "Robust real-time face detection," International Journal of Computer Vision, 57, 2, pp. 137-154, 2004.

[7] N.Dalal and B. Triggs, "Histograms of oriented gradients for human detection," CVPR, vol.1, pp.886-893, 2005.

[8] Y.Freund. and R.E.Schapier. "Experiments with a new boosting algorithm," In Proceedings of the Thirteeuth International Conference on Machine Learning. 1996.

[9] R.E.Schapire and Y.Singer. "Improved Boosting Algorithms Using Confidence-rated Predictions," Machine Learning. No. 37, pp. 297-336, 1999.

[10] D.Comaniciu, P.Meer, "Mean shift analysis and applications", IEEE International Comference on Computer Vision, pp. 1197-1203, 1999.

[11] B.W.Silverman, "Density Estimation for Statistics and Data Analysis," Chapman & Hall, 1986.

[12] K.Fukunaga and L.Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," IEEE Trans.

Information Theory, 21(1): 32-40, 1975.

[13] "Fast and Stable Human Detection Using Multiple Classifiers Based on Subtraction Stereo with HOG Features"

[14] Katsunori Onishi, Tetsuya Takiguch, Yasuo Ariki, "3D Human Posture Estimation Based on Linear Regression of HOG Features from Monocular Images", Advances in Computer Science and Engineering, ISSN: 0973-6999, Volume 3, Issue 3, pp. 175-186, (November 2009).

[15] 池村翔, 藤吉弘亘, "距離情報に基づく局所特徴量によるリアルタイム人検出", 電子情報通信学会論文誌, Vol.J93-D, No.3, pp.355-364, 2010.

[16] P.Felzenszwalb and D.Huttenlocher, "Pictorial structures for object recognitnion," IJCV, 61(1), 2005.

[17] P.Felzenszwalb, R.Girschick, and D.McAllester: "Cascade object detection with deformable part models", In CVPR, pp. 1-8, 2010.

[18] E.Cuevas, D.Zaldiver, and R.Rojas: "Kalman filter for vision tracking", Technical Report B, Fachbereich Mathematik und Informatic, Freie Universitat Berlin, 2005.

[19] P.Brasnett, L.Mihaylova, D.Bull and N.Canagarajah: "Sequential Mnte Carlo Tracking by Fusing Multiple Cues in Video Sequences", Image Vision Comput., 25, 8, pp. 1217-1227, 2007.

[20] 竹内栄二郎, 大野和則, 田所論, "1A1-D13 移動ロボットによる障害物検出のための3次元観測計画", ロボティクス・メカトロニクス講演会講演概要集 2009, "1A1-D13(1)"-"1A1-D13(4)", 2009.

[21] 羽多野裕之, 山里敬也, 片山正昭, "超音波アレイエミッタを用いた自動車用近距離障害物検出システムの検討", 電子情報通信学会技術研究報告書. ITS 107(161), 21-26, 2007.

[22] 佐竹純二, 三浦純, "ステレオカメラを用いた移動ロボットのための人物追跡", 電離情報通信学会技術報告書.PRMU, パターン認識・メディア理解 108(263), 37-42, 2008.

[23] G. Stein and O. Mano and A. Shashua, "Vision-based ACC with a Single Camera: Bounds on Range and. Range Rate Accuracy", *IEEE Intelligent Vehicles Symposium (IV2003)*, 2003.

研究発表実績

学会発表

1. 第 38 回感覚代行シンポジウム
   感覚代行研究会　2012 年 12 月
   発表題目：視覚障害者歩行支援のための単眼カメラを用いた歩行者検出システム

謝辞

本研究を進めるにあたって、最後まで心を折ることがなかった自分自身に心から感謝しています。また、様々なご助言、ご指導をしてくださったナノシステム科学専攻の先生方にも心からお礼申し上げます。

そして、励ましのエールなどにより、心の支えとなってくださった榊原ゆい様には、心からの感謝を示します。ありがとうございました。