

# 人型ロボットの視覚情報認識のための画像処理機能の性能比較\*

—画像処理ハードウェアと画像処理ソフトウェアの性能比較—

長 井 達 一 郎\*\*

森 野 友 哉\*\*

近 藤 禎 敏\*\*

## Performance Comparison of Image Processing for Robot between Hardware and Software

Tatsuichiro NAGAI\*\*, Tomoya MORINO\*\* and Tadatoshi KONDO\*\*

Recently, it is hoped for the robot to advance to person's life space and to help the person. The visual sensation of the robot is important to recognize the environment. The image processing is necessary to recognize the thing by the vision sensation. There are two methods of image processing: the method of doing with hardware and the method of doing with software. In this paper, to examine the method that is appropriate for the robot, we compared the performances of the image processing by hardware and the image processing by software. The hardware is IP7000BD and the software is OpenCV.

*Key Words* : Humanoid Robot, Image Processing, Object Detection, IP7000BD, OpenCV

### 1. はじめに

ロボットは工場等の現場から人間の生活空間へ進出し始め、生活の中で人間を助けるなど、徐々に人との共生が始まっていくと期待されている。ロボット自身が周りの環境を認識するためには、人間同様、視覚情報が重要である。現在、ロボットにはカメラが搭載されているものが多い。当研究室においても、人型ロボット MES-HR の頭部に 4 つのカメラを搭載し、研究に使用している。

ロボットが周りの環境を認識するためには、カメラからの映像を画像として取得し、画像処理を行い、画像の中にある物体の色、輪郭、位置などを検出する必要がある。これら画像処理を行うには、画像処理モジュールや画像処理ボードなど専用のハードウェアを用いる方法

と、計算機上で実行する画像処理プログラムなどソフトウェアを用いる方法の大きく 2 種類がある。画像処理専用のハードウェアは処理が高速で、ロボットなどリアルタイム性が必要な用途に適している。しかし、ハードウェア上に実装されているため、画像処理のアルゴリズムを変更するなどの汎用性は乏しい。一方、画像処理のソフトウェアは汎用性が高く、処理の方法に自由度がある。しかし、処理速度は計算機性能に依存し、リアルタイム性が求められる用途にはあまり適さなかった。さらに、画像処理の専門知識がないとプログラムの作成ができない問題もある。

当研究室では、長年、画像処理専用のハードウェアを研究に用いてきた。しかし、近年、計算機の処理速度が急速に向上し、また画像処理のプログラムもオープンソースでライブラリ化され、非常に使いやすくなってきた。そこで、本論文では当研究室で使用している画像処理ボード IP7000BD(株式会社 日立情報制御ソリューションズ)と画像処理ライブラリ OpenCV を比較し、

\* 平成 22 年 2 月 3 日 受付

\*\* 電子情報工学科

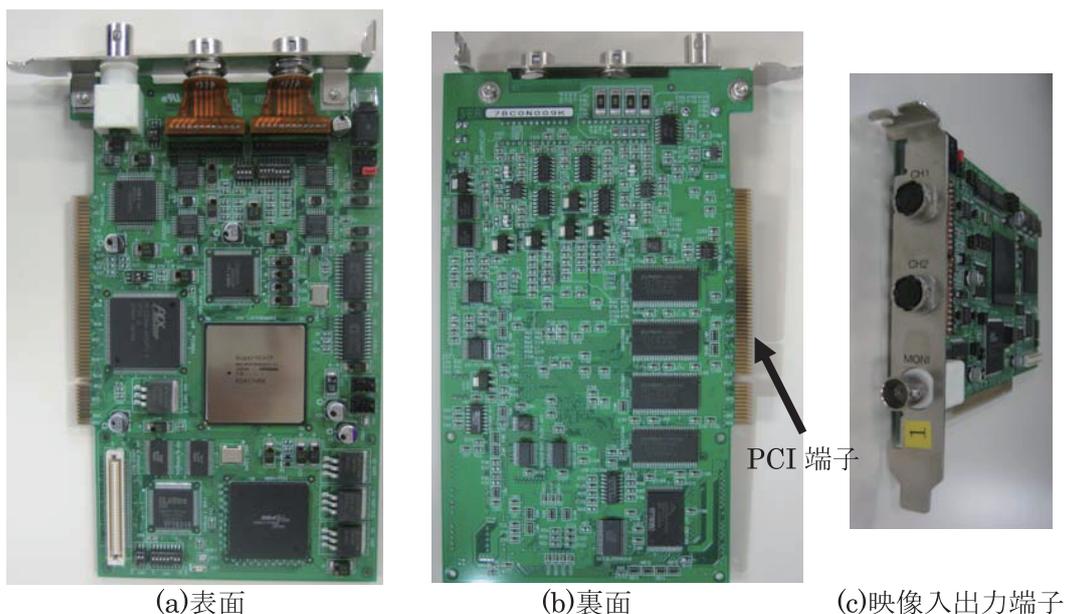


図 1 IP7000BD の外観

現時点でどちらがよりロボットの視覚情報認識のための画像処理に向いているか検証を行った。

## 2. 画像処理ボード IP7000BD

画像処理ボード IP7000BD(株式会社 日立情報制御ソリューションズ)は計算機の PCI スロットに接続する画像処理ボードである。この画像処理ボードは約 10 年前から販売されている製品だが、非常に高性能で現在でも多く使用され続けている。IP7000BD の外観を図 1 に、

仕様を表 1 にそれぞれ示す。

IP7000BD はカメラからの映像を映像入力端子から受取り、命令された画像処理の結果を映像出力や PCI を通して出力する。命令は C/C++ のプログラムで行う。開発環境は Windows の場合は Microsoft Visual C++, Linux の場合は gcc(GNU) である。画像処理の機能としては、エッジ強調、正規化相関、色抽出などがある。

## 3. 画像処理ライブラリ OpenCV

Intel 社で開発された Intel Open Source Computer Vision Library(以下、OpenCV)は、画像処理の C 言語ライブラリであり、またオープンソースであるためロボットの研究等で多く使用されている。画像処理の機能としては、モーション解析、物体追跡、パターン認識、物体検出などがある。また、画像処理の詳しい知識がなくてもある程度は簡単に使用できる仕様になっている。プログラムであるため、画像処理内容の変更や改良も自由である。処理速度は実行する計算機の能力に左右される。開発環境は Windows, Linux, Mac OS X 等多くの OS をサポートしている。今回、実験に使用するバージョンは OpenCV 1.1 pre である。

## 4. 人型ロボット MES-HR の視覚部

当研究室で使用している人型ロボット MES-HR(三井造船)の頭部には、カメラが 4 台装備されている。うち 2 台はメインで視覚情報を取得するズームレンズ内蔵のカラーカメラブロック、他 2 台はロボットの周辺を広く見るための広角レンズ搭載のカメラである。今回、画像処理の実験のために用いるのはメインのカメラである。MES-HR の全体とカメラが搭載されている頭部の

表 1 IP7000BD の主な仕様

項目	内容
CPU	日立 SuperRISC CPU SH-4(166MHz)
画像プロセッサ	画像認識プロセッサ SuperVchip(166MHz)
画像メモリ	モノクロ(512×512×8×120 面)または モノクロ高精細(1,024×1,024×8×30 面)
映像入力	NTSC モノクロ/カラー、モノクロ高精細兼用 標準 2ch 同時入力(2ch:オプション:最大 4ch) CVBS カラー2ch 切替入力(オプション 4ch 切替)
映像分解能	8ビット 256 階調
映像出力	NTSC カラー(1ch)
画像処理機能	画像間マトリックス演算 Min/Max、平滑化/エッジ強調、他 正規化相関、色抽出、他 8 段 2 値、画像演算・2 値・ヒストグラムバイブライン、 画像処理コマンド約 260 種類
PCI インタフェース	PCI バス仕様 Rev2.1 準拠、5V、32ビット PCI ハーフサイズ
周囲温度	10~35°C

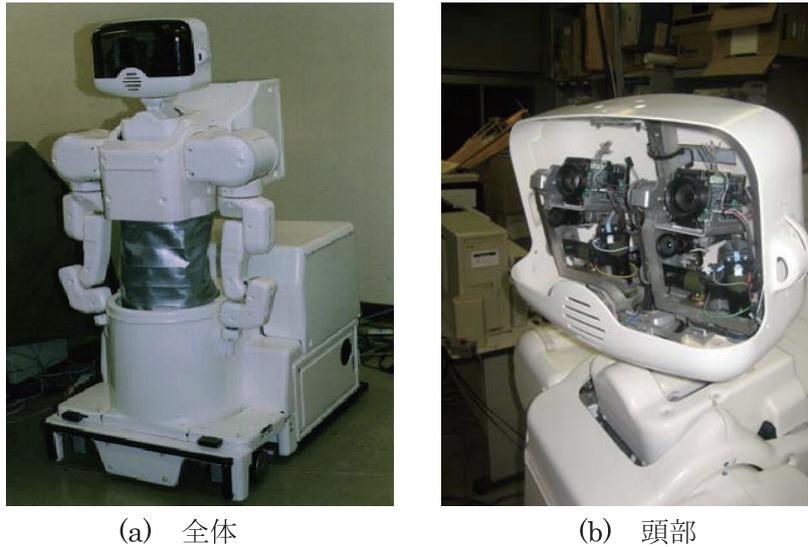


図2 人型ロボット MES-HR

写真を図2に示す。

搭載されているメインのカメラは、FCB-IX47A(ソニー)である。このカメラは、有効画素数38万画素(768×494ピクセル)、光学18倍ズームレンズ内蔵の多機能カラーカメラブロックである。

現在は、このカメラからの映像ケーブルを計算機に装備してあるIP7000BDの映像入力端子に接続して、画像を取得し画像処理を行い、その結果を計算機で取得している。主に使用している画像処理は、色認識、テンプレートマッチング、移動体の追跡などである。現在、当研究室で使用しているロボットの視覚のシステム構成を図3に示す。比較実験は図中の視覚処理用計算機上で、IP7000BDの制御プログラムとOpenCVの画像処理プログラムを実行する。

今回使用する視覚処理用計算機の仕様と開発環境は次の通りである。

- CPU : Intel Core 2 Extreme Processor QX9650 [3.0GHz]

- GPU : NVIDIA GeForce 9600 GT
- MEM : 3.25GB [DDR2]
- HDD : ST3500320AS [SATA]
- OS : Microsoft Windows XP Professional ServicePack3
- 開発環境 : Microsoft Visual C++ 6.0 Enterprise
- 画像処理ボード : IP7000BD (株式会社 日立情報制御システム★)
- 画像処理ライブラリ : OpenCV 1.1 pre

実験において計算機の性能がボトルネックにならないよう、できる限りハイスペックのものを用意した。

### 5. 画像処理機能性能比較

当研究室でよく使用する色認識、移動体の追跡の画像処理を行いIP7000BDとOpenCVの性能を比較する。色認識は、ロボットに作業を行わせる対象物を色で指定する場合に用いる。移動体の追跡は、人や物の移動をロボットに認識させる場合に用いる。今回は3種類の性

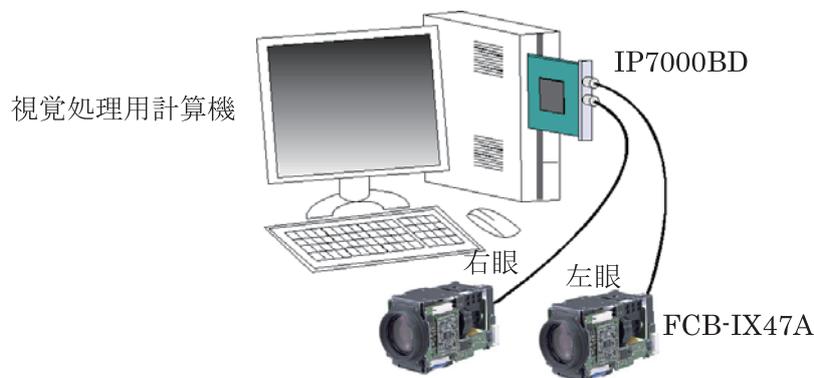


図3 現行のシステム構成

★現 : 株式会社 日立情報制御ソリューションズ

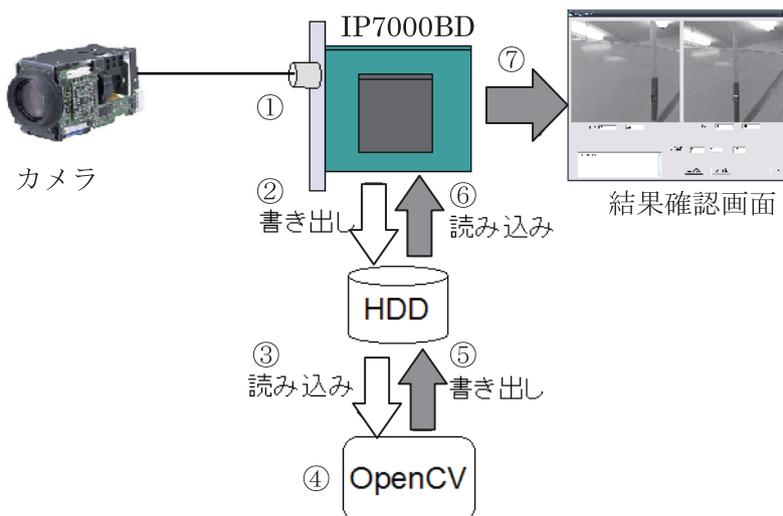


図4 実験システムのデータの流れ

能比較の実験を行った。以下実験内容と結果について述べる。

### 5.1. 実験システム

ロボット MES-HR のメインのカメラ FCB-IX47A からの映像ケーブルを IP7000BD に接続し、IP7000BD を視覚処理用計算機に接続し、計算機上でプログラムにより IP7000BD と OpenCV に画像処理を行わせ、その性能を比較する。今回の実験ではカメラを1台使用する。実験システムのデータの流れを図4に示す。IP7000BD の性能測定では、まず、カメラからの映像を IP7000BD の映像入力端子から受取り①、IP7000BD 上の画像プロセッサを使用して画像処理を行い、その結果を計算機の画面に表示させる⑦。OpenCV の性能測定では、カメラからの映像を IP7000BD の映像入力端子から受取り①、その映像をビットマップファイルとしてハードディスクに保存し②、保存されたビットマップファイルを OpenCV のプログラムが開き③、OpenCV が画像処理を行い④、結果の映像をビットマップファイルとしてハードディスクに保存し⑤、IP7000BD が OpenCV の結果のファイルを開いて⑥、計算機の画面に表示させる⑦。

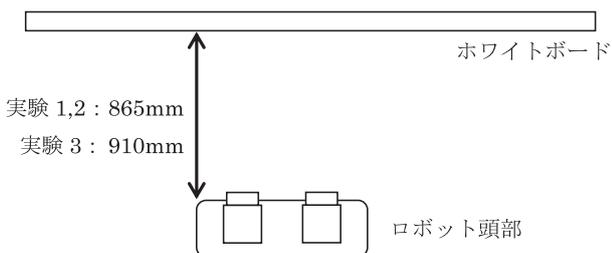


図5 実験環境

### 5.2. 実験環境

実験では、ホワイトボードにプロジェクターで映した画像を対象物としてロボットのカメラで捕らえる。その配置関係を図5に示す。対象物が映るホワイトボードとロボットの頭部の間は実験1と実験2では86.5cm、実験3では91cm離れている。プロジェクターのリフレッシュレートは60Hzである。

実際の実験の様子を図6に示す。プロジェクターがロボットの視覚部からみて、少し低い位置に設置してあるため、映し出される四角は正方形にはならない。

### 5.3. 実験1 色認識

まず、色認識の算出時間の測定を行った。画像処理では赤色を認識させ、その重心を算出し重心部分に印を表示する処理を行わせた。実験では赤の四角が表示されてから重心を出力するまでの時間を IP7000BD と OpenCV で比較した。実験で使用した画面を図7に示す。同図(a)はプロジェクターで表示する画面である。

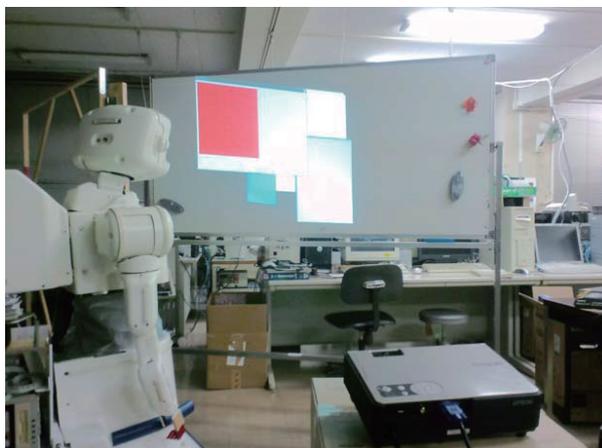
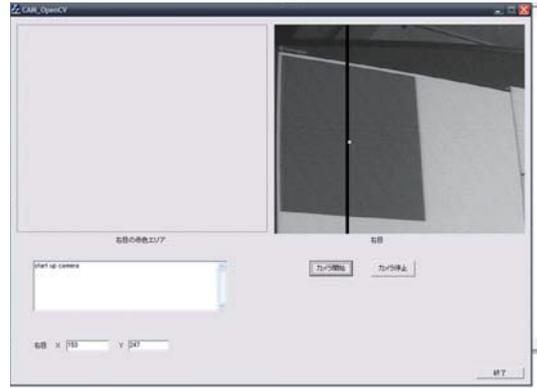


図6 実験の様子



(a) 認識テスト画像



(b) 確認用システム画面

図 7 色認識の実験画面

最初、画面は灰色で、しばらくすると赤の四角が表示される。同図 (b) は画像処理後の赤の四角の重心を表示している。同図 (a) の赤い四角は 500×500 ピクセルの四角で、ホワイトボード上では 37.5×33[cm] の四角で映った。

赤色認識の実験結果を表 2 に示す。同表はプロジェクターでホワイトボード上に赤色が表示されて、画像処理後の重心が確認画面に表示されるまでの時間を示している。10 回計測を行った。ばらつきはあるものの平均すると OpenCV の方が約 25msec ほど反応速度が速いことが分かった。但し、画像データを IP7000BD から OpenCV に渡すためにハードディスクを経由する時間は含まれていない。表 2 をグラフにしたものを図 8 に示す。

グラフでは、帯が低いほど反応時間が速かったことを

示す。当研究室の実験環境においては、グラフからも分かるように、OpenCV の方が色認識においてはリアルタイム性に優れていることが確認できた。

#### 5.4. 実験 2 最小の大きさの認識限界

次に認識できる最小の大きさを測定した。実験 1 で使用した赤の四角を図 9 に示すように徐々に小さくしていき画像処理で赤の四角の重心を算出できる限界を測定した。

赤の四角をプロジェクターで表示し、カメラの画像を処理した結果、赤の四角の重心座標が算出できたか判断した。測定結果を表 3 に示す。表中、大きさは四角の一辺のピクセル数、○は認識に成功、×は認識不可、その他条件付成功を示している。背景が黒なら成功の場合には対象物以外は作業空間に置かれていない状態であるため、人の日常生活ではほとんどない状態である。したがって、IP7000BD は一辺がおよそ 40 ピクセル (約 3cm) の四角が限界であり、OpenCV は一辺がおよそ 30 ピクセル (約 2cm) の四角が限界であった。

表 2 色認識の反応時間 (単位 : msec)

回数	OpenCV	IP7000BD
1	125	62
2	31	63
3	157	125
4	93	140
5	94	156
6	93	141
7	109	141
8	47	78
9	109	125
10	47	125
平均	90.5	115.6

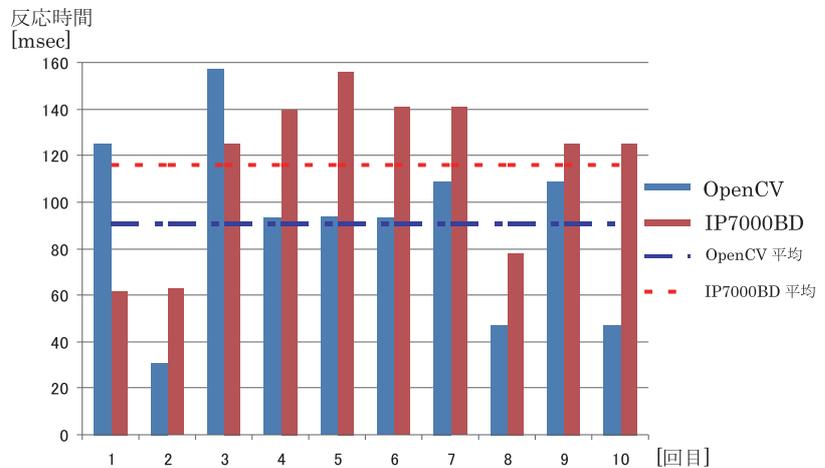


図 8 色認識の反応時間のグラフ

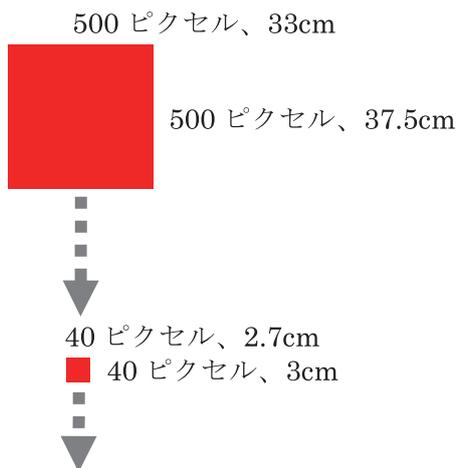


図9 最小の認識限界の測定方法

実験2の結果から、今回の実験環境ではOpenCVの最小限界はIP7000BDの最小限界の3分の2の大きさまで認識可能なことが分かった。

5.5. 実験3 移動体の追跡

最後に、移動体の追跡の実験を行った。ロボットの研究では視覚に映った移動体を追跡し、対象物がどの方向に行ったかの認識が必要なことがある。実験3では、移動体の速度を徐々に速めて、追跡の限界を測定した。移動体は図7(a)の画面上に80×80ピクセル(4.5×5.8[cm])の赤の四角を表示し一定速度で移動させる。また、移動方向は横と縦向きとの2つの方向で測定を行った。図10

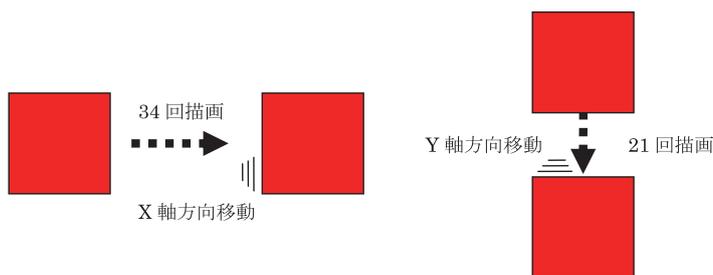


図10 移動体の追跡実験

に移動体の動きを示す。横向きであるX軸方向は34回画面を描き直すことで四角を移動させ、縦向きであるY軸方向は21回画面を描き直すことで四角を移動させた。実験は、移動体が表示され端まで移動し、画面から消えるまでの移動体の検出回数を測定した。

X軸方向の測定結果を表4に示す。同表は各速度の時の移動体の検出回数を示している。ほとんどの速度でOpenCVの方が検出回数が多い。

Y軸方向の測定結果を表5に示す。同表は各速度の時の移動体の検出回数を示している。ほとんどの速度でOpenCVの方が検出回数が多い。

表4, 表5の結果をグラフにしたものを図11に示す。同グラフは、帯が高いほど検出回数が多いことを示す。同図(a)は表4のX軸方向移動のグラフであり、同図(b)は表5のY軸方向のグラフである。X軸方向の移動、Y軸方向の移動とも、OpenCVの方が検出回数が多かった。これにより、ロボットのカメラに映る移動

表3 最小の認識限界の測定結果

大きさ	OpenCV	IP7000BD
500	○	○
100	○	○
50	○	○
40	○	○
35	○	背景が黒なら○
34	○	背景が黒なら○
33	○	背景が黒なら○
32	○	背景が黒なら○
30	○	×
25	端以外なら○	×
20	背景が黒なら○	×
10	×	×

表4 移動体の追跡実験の結果(X軸方向)

Speed[cm/sec]	OpenCV	IP7000BD
20.5	45	35
22.5	41	31
25	34	26
28.1	35	27
32.1	30	22
37.5	25	17
45	22	18
56.3	18	13
75	10	9
112.5	8	8
225	5	4

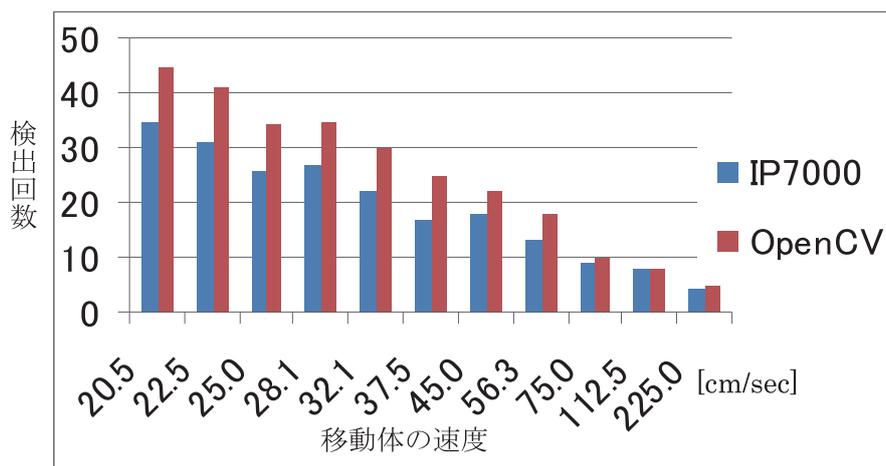
表 5 移動体の追跡実験の結果 (Y 軸方向)

Speed[cm/sec]	OpenCV	IP7000BD
20.5	33	22
22.5	28	20
25	25	16
28.1	25	17
32.1	21	14
37.5	16	11
45	15	11
56.3	13	8
75	8	5
112.5	8	6
225	3	3

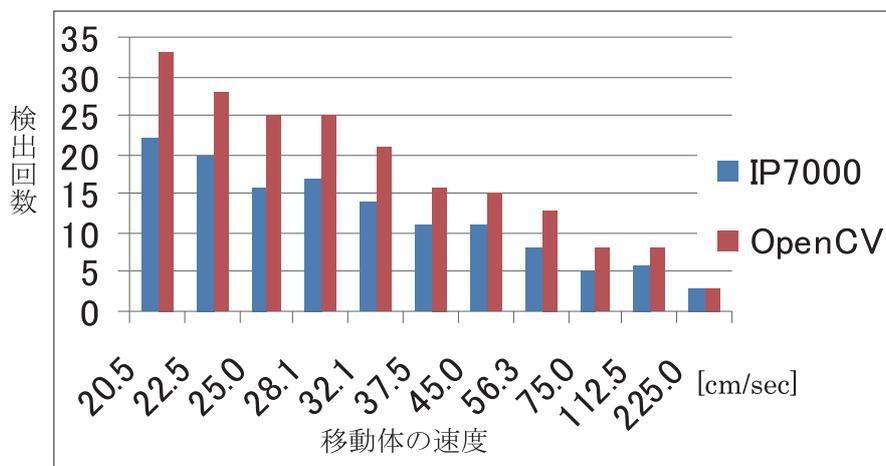
体の検出には OpenCV の方が適していると考えられる。人の歩く速度はおよそ時速 4km(111.1[cm/sec]) と言われている。ロボットのカメラ前方 1m の辺りを人が歩いて横切った場合、実験 3 の 75 ~ 112.5[cm/sec] に該当する。この距離と速度では IP7000BD と OpenCV の差はほとんどない。しかし、ロボットのカメラ前方 1m より距離が離れているところを人が歩いて横切った場合は、OpenCV の方が検出率が高いことが分かる。よって IP7000BD と OpenCV を比較した場合、当研究室の実験環境においては、OpenCV の方がロボットの視覚情報処理に適していると言える。

6. まとめ

本論文では、ロボットの視覚で使用している画像処理ボード IP7000BD と画像処理ライブラリ OpenCV の色認識と移動体の追跡の性能比較実験を行った。ロボットの視覚の処理はリアルタイム性が求められる。IP7000BD はハードウェアのため処理速度が速い。しか



(a) X 軸方向移動



(b) Y 軸方向移動

図 11 移動体の追跡実験の結果のグラフ

し、近年の計算機の処理速度は非常に速くなり、画像処理をソフトウェアでリアルタイムに処理することも可能になりつつある。さらに、画像処理の共有化、ライブラリ化が進みプログラミングにおいても利用しやすくなった。また、OpenCV はフリーのソフトウェアであるため入手も容易である。

実験は、実際にロボットの研究環境を用いて測定を行った。今回の実験結果は、全ての実験で OpenCV の優位性が示された。しかし、その差はまだそれほど大きくない。IP7000BD は、およそ 10 年前からある画像処理ボードであるが、この IP7000BD に比べ処理速度が約 20 倍の現在の高性能計算機上でのソフトウェアによる画像処理の性能と速度に近い結果であった。これは、ハードウェアに実装された画像処理の回路がいかに速いかを示している。逆に、ハードウェアによる画像処理が速度的に有利と一般的に言われていたが、徐々にソフトウェアによる画像処理でも十分な性能が出る時代になりつつあることが分かった。

一方、画像処理のプログラミングについては、IP7000BD に画像処理を行わせるための制御プログラムのステップ数と、OpenCV で同じ処理を行わせるためのステップ数の比は IP7000BD:OpenCV = 5 : 1 であった。つまり、OpenCV の方が少ないプログラムのステップ数で画像処理を行わせることができる。そのため、画

像処理のプログラムを作成しやすかった。

今後は、ロボットの視覚情報認識でよく使用される形状認識のためのパターンマッチングやエッジ検出などの性能比較を行い、さらに詳しく IP7000BD と OpenCV の適性を調べる必要がある。

#### 参 考 文 献

- [1] “人間型ロボット MES-HR 仕様・取扱説明 Vol.1, Vol.2”, 三井造船株式会社, 1998.
- [2] “画像処理コマンドドライバ IP7000 プログラミングマニュアル”, 株式会社日立情報制御システム, 2002.
- [3] “画像処理ボード IP7000BD”,  
URL < <http://www.hitachi-ics.co.jp/product/ganin/7000/7000.html> >, 株式会社日立情報制御ソリューションズ
- [4] “OpenCV”, URL < <http://opencv.jp/> >
- [5] 奈良先端科学技術大学院大学 OpenCV プログラミングブック制作チーム: “OpenCV プログラミングブック第2版 OpenCV 1.1 対応”, 毎日コミュニケーションズ, 2009.
- [6] David A. Forsyth, Jean Ponce: “コンピュータビジョン”, 共立出版, 2007.