

# 小型距離画像センサを用いた ヒューマノイド歩行中における障害物認識 Obstacle recognition in a humanoid walking by using a small range image sensor

学 渡辺 圭(中央大) 学 彦坂 直孝(中央大) 正 梅田 和昇(中央大)

Kei Watanabe, Chuo University, kwatana@sensor.mech.chuo-u.ac.jp  
Naotaka HIKOSAKA, Chuo University, hikosaka@sensor.mech.chuo-u.ac.jp  
Kazunori UMEDA, Chuo University, umeda@mech.chuo-u.ac.jp

**Abstract**—This paper deals with a target that a humanoid with a small range image sensor recognizes obstacles and avoid them autonomously while walking. Planar regions, on which the humanoid can walk, and obstacles are distinguished by using a relative disparity image obtained by a small range image sensor. In addition, an integrated system that can feed back obtained obstacle information to the humanoid is constructed. Experiments show that the humanoid can recognize obstacles and stop walking autonomously by the proposed system.

**Key Words:** Small range image sensor, Relative disparity image, Humanoid, Obstacle recognition

## 1. 序論

近年、一般家庭へロボットが普及するようになってきた。人間の生活する環境など多くの障害物が存在する環境で移動する場合、ロボットは平面領域を検出する必要がある [1]。我々は、簡易に平面領域を検出するために、基準平面からの相対的な視差のみを用いた手法を提案し、さらに、ヒューマノイド歩行中でも安定して平面領域を検出できる手法を構築してきた [2] [3]。本研究では、小型距離画像センサから得た障害物情報をヒューマノイドにフィードバック可能な統合的なシステムを構築し、ヒューマノイドが歩行中に床面上の障害物を認識し自律的に回避することを目指す。

## 2. 小型距離画像センサ

Fig.1 にヒューマノイドに搭載する小型距離画像センサを示す。スポット光投影部はレーザープロジェクト (MORITEX SNF-519X) を使用している。波長 670[mm], 19 × 19 点のドットマトリクス状のスポット光を投影する。画像入力部は CCD カメラ (TOSHIBA IK-M41MR) を使用し、レンズの焦点距離は 15[mm] である。外乱光を除去するため、受光部に R64 フィルタ (HOYA) を取り付けている。CCD カメラとレーザープロジェクトとの基線長は 23[mm] となっている。画像処理装置は、画像処理ボード (HITACHI IP-5000) を使用している。レーザープロジェクトを用いて対象物にマルチスポット光を投影し、そのシーンを CCD カメラで取得し、各スポット光の結像点の検出を行う。得られた画像から、各スポット光の重心を算出し、その点のズレ量を計測することによって、平面、あるいは障害物を認識する。

## 3. 小型距離画像センサのヒューマノイドへの 応用

### 3.1 平面領域検出における精度評価

#### 3.1.1 平面領域検出手法

平坦な基準平面を取得した際と対象を取得した際のスポット光の画像上における結像位置の差を相対視差と定義する。この相対視差を用いることで平面領域を検出する [4]。Fig.2 に床面とセンサの位置姿勢の関係を示す。基準平面を取得した時のセンサの位置姿勢を  $\theta_0, \phi_0, h_0$  とする。センサの位置姿勢が  $\theta, \phi, h$  に変化したとき、画像上のスポット光の相

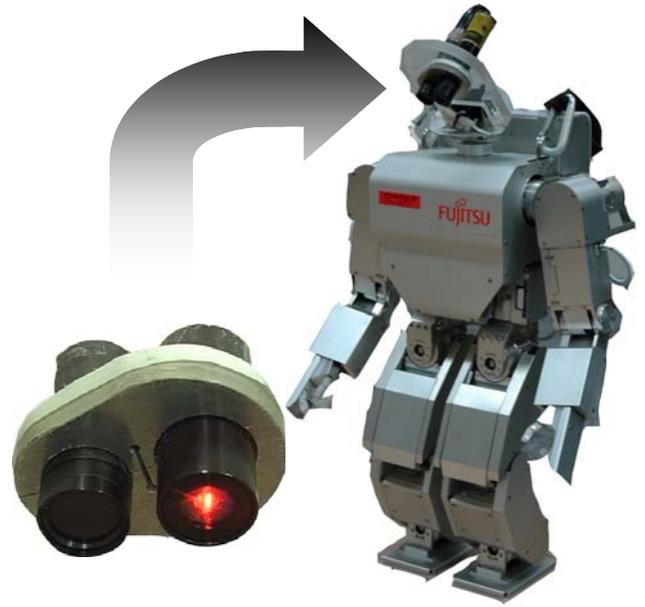


Fig.1 Small range image sensor and humanoid

対視差  $\Delta k$  は式 (1) のように表すことができる。

$$\begin{aligned} \epsilon \frac{\Delta k}{b \cdot f} = & \left( \frac{\cos \theta \sin \phi}{h} - \frac{\cos \theta_0 \sin \phi_0}{h_0} \right) s \\ & + \left( \frac{\cos \theta \cos \phi}{h} - \frac{\cos \theta_0 \cos \phi_0}{h_0} \right) t \\ & + \left( \frac{\sin \theta}{h} - \frac{\sin \theta_0}{h_0} \right) \end{aligned} \quad (1)$$

$b, f$  はそれぞれ基線長, 焦点距離であり既知である。式 (1) より、相対視差がスポット光の  $s-t$  座標系の座標値  $s, t$  に対する平面となるのがわかる。また、未知となるパラメータは、 $\theta, \phi, h$  の三つであるので、相対視差が構成する平面の三つのパラメータからこれらを求めることができる。

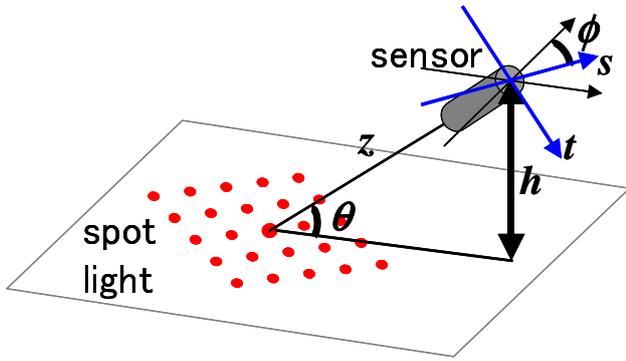


Fig.2 Position and orientation of a small range image sensor for the floor

Table 1 Experimental conditions

condition	$\theta$ [deg]	$\phi$ [deg]	$h$ [mm]
reference	45	0	470
1	60	0	451
2	55	0	457
3	50	0	464
4	45	0	470
5	40	0	476
6	35	0	482
7	30	0	488

### 3.1.2 精度評価実験

式 (1) を用いた平面領域検出における精度評価を行う。ヒューマノイドの頭部に搭載されたセンサの位置姿勢を Table 1 に示すように変化させ、相対視差を取得した。Fig.3 に相対視差画像を示す。(a), (b) はそれぞれ条件 1, 7 での結果を、(c) は平面上に障害物 (約  $25 \times 50 \times 25$ [mm] の箱) がある時の結果を示している。Fig.3(a), (b) より、相対視差画像は平面に近いことがわかる。これらに対して平面あてはめを行い、残差二乗和により平面度の評価を行った。その結果を Fig.4 に示す。Fig.4 から、残差二乗和により平面のみであるか障害物があるかを判断することが可能であると言える。

Table 2 に、センサの位置姿勢検出における精度を示す。結果は十回の計測における平均、標準偏差である。 $\theta$ ,  $h$  の誤差はそれぞれ  $\pm 3.2$ [deg],  $\pm 13$ [mm] 以内におさまっているものの、 $\phi$  の誤差は、 $\pm 13$ [deg] もの大きな値を示している。これは、CCD カメラとレーザプロジェクタの光軸が傾いているためと考えられる。

### 3.2 小型距離画像センサを搭載した

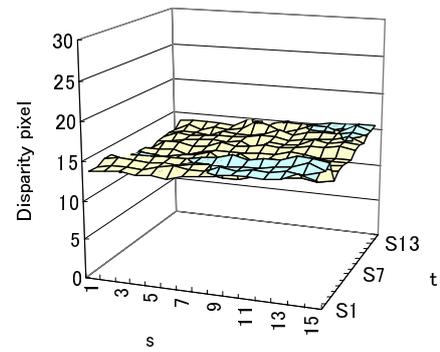
#### ヒューマノイド制御システム

Fig.5 に、小型距離画像センサを搭載したヒューマノイド制御システムを示す。ヒューマノイドは FUJITSU HOAP-2 を使用し、PC は DELL Dimension1100 (Pentium4 2.8GHz), OS は RT-Linux (Kernel2.4.18-pre1) を採用している。小型距離画像センサは、ヒューマノイドと同一の PC で制御し、RT-Linux 上のユーザ空間で動作している。

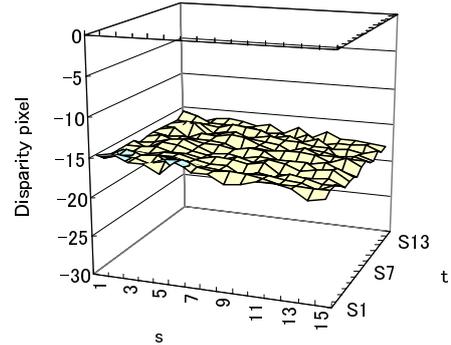
### 3.3 歩行時における障害物認識実験

#### 3.3.1 処理の流れ

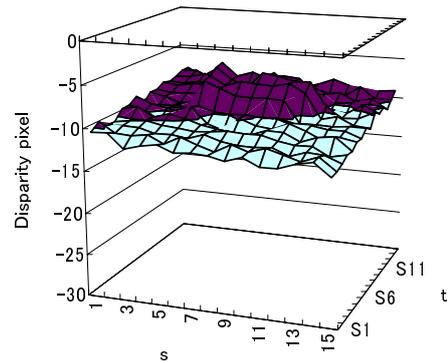
小型距離画像センサを搭載したヒューマノイドが歩行中に障害物を認識して停止する実験を行う。歩行中の処理の流れを Fig.6(a) に、歩行停止までの処理の流れを (b) に示す。ここでは、POSIXthread [5] を用いて画像処理プロセス (ImageProcess thread) とヒューマノイド指令プロセス (Socket thread) を並列化している。Socket thread とヒューマノイドを制御するためのソフトウェア (NueROMA OptionLibrary) とソケット通信を行い制御を行っている。Im-



(a) Condition1



(b) Condition7



(c) With an obstacle

Fig.3 Relative disparity map

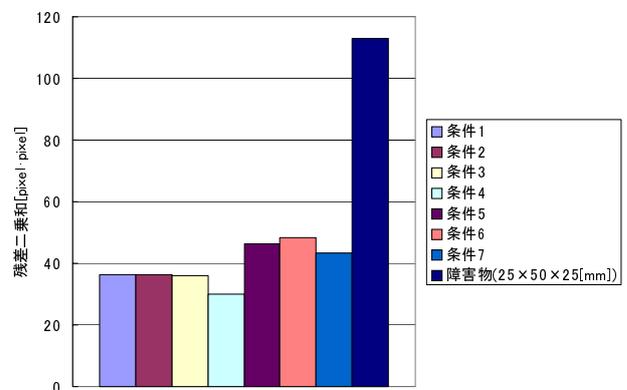


Fig.4 Residuals of plane fitting

Table 2 Measurement results of plane parameters

condition	$\theta$ [deg]	$\phi$ [deg]	$h$ [mm]
1 real value	60	0	451
experiment	56.86	12.30	448.98
standard deviation(10 times)	0.304	0.363	1.541
2 real value	55	0	457
experiment	53.48	8.08	456.70
standard deviation(10 times)	0.289	0.205	1.722
3 real value	50	0	464
experiment	49.29	4.09	464.85
standard deviation(10 times)	0.285	0.146	1.999
4 real value	45	0	470
experiment	45.78	-0.03	470.13
standard deviation(10 times)	0.260	0.144	2.131
5 real value	40	0	476
experiment	41.00	-3.21	478.47
standard deviation(10 times)	0.241	0.152	2.416
6 real value	35	0	482
experiment	36.75	-6.90	488.86
standard deviation(10 times)	0.271	0.083	3.094
7 real value	30	0	488
experiment	33.17	-10.14	500.80
standard deviation(10 times)	0.274	0.203	3.663

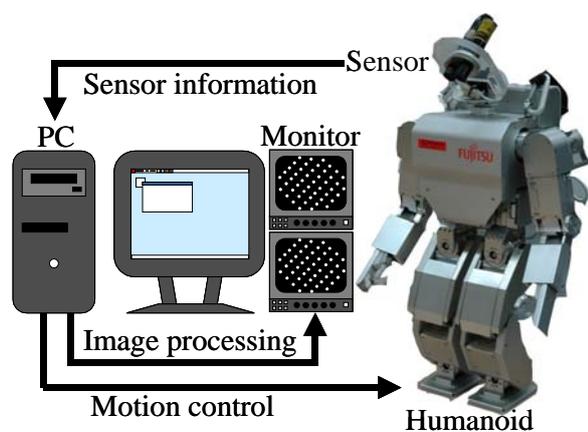


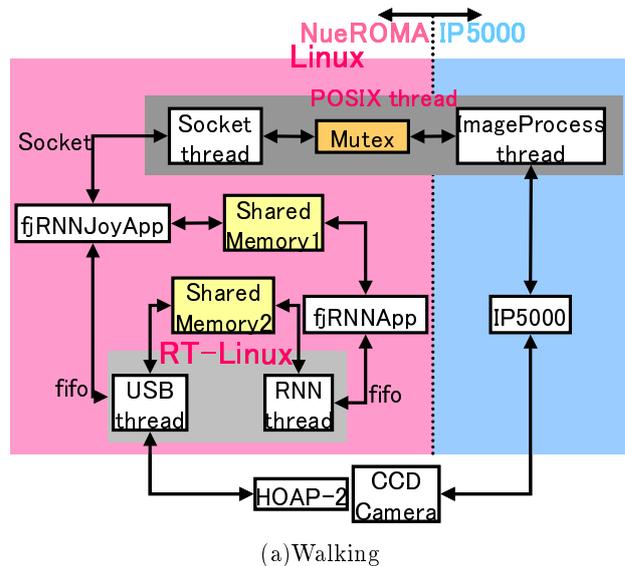
Fig.5 Integrated system

ageProcess thread と Socket thread は Mutex を介して同期や通信を行う。

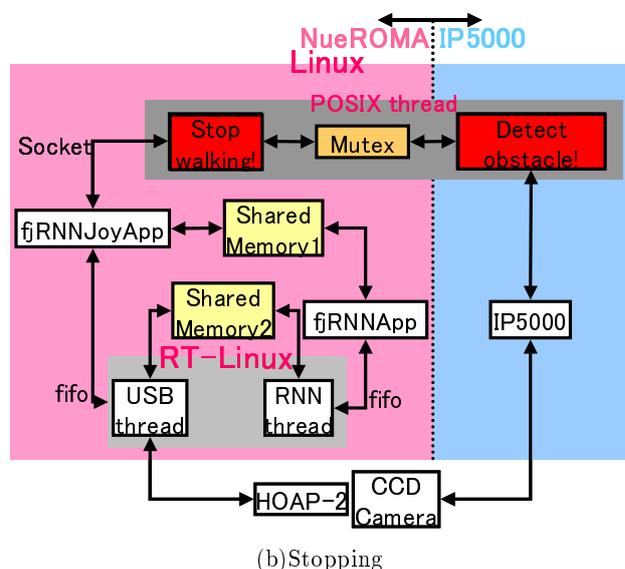
センサにより基準平面を取得し、二つのプロセス間 (ImageProcess thread- Socket thread) で同期を取ってから同時に処理を開始する。障害物を検出すると、ImageProcess thread は Mutex [5] を介して Socket thread に歩行停止の指令を送り、Socket thread は fjRNNJoyApp に対して歩行停止の指令をソケット通信にて送信する。歩行停止の指令を受けた fjRNNJoyApp は共有メモリ 1 に歩行停止の指令を書き込み、また、RT-Linux のカーネル空間に対して通信も行う。共有メモリ 1 に書き込まれた歩行停止の指令を読み込んだ fjRNNApp は、RT-Linux のカーネル空間における RNN thread と通信し、共有メモリ 2 に歩行停止コマンドを書き込む。RT-Linux のカーネル空間における USB thread は歩行停止コマンドを読み込んでヒューマノイドの歩行を停止させる。

### 3.4 障害物認識実験

Fig.7 に画像処理プロセスのみを動作させた時の画像一枚にかかる処理時間、およびヒューマノイド制御プロセスと並列に動作させた時の画像一枚にかかる処理時間を示す。Fig.8 に障害物認識前後の平面あてはめにおける残差二乗和を示す。Fig.9 にヒューマノイド歩行中における障害物認識実験の様子を示す。(a)~(p) は歩行開始点から障害物 (長方形の箱: 約  $200 \times 200 \times 100$ [mm]) を認識して自律的に停止するまでの様子を 1[s] 毎に示したものである。Fig.9 より、歩行中上体が不安定なヒューマノイドに小型距離画像センサを搭載して



(a) Walking



(b) Stopping

Fig.6 Flowchart of processing

も平面上の障害物を認識し、自律的に歩行を停止するのを確認することができた。Fig.7 より、画像一枚にかかる処理時間は、画像処理プロセスのみを動作させた時とヒューマノイド制御プロセスと並列に動作させた時の差が最大で約 250[ms] となり、並列化したときの画像一枚の処理速度が著しく低下することが分かった。これは、ヒューマノイドを制御するソフトウェアである NueROMAOptionLibrary が、GUI による操作であるため計算コストが大きくなっているためと考えられる。Fig.8 より、ヒューマノイド歩行中における残差二乗和は  $50$ [pixel<sup>2</sup>] 以下と低い値を示しているが、障害物を検出すると、 $1000$ [pixel<sup>2</sup>] 以上の大きな値を示していることから、歩行中でも平面のみが障害物があるかを容易に判断することができる。本システムではシャープペンシルの芯のケース (約  $20 \times 60 \times 5$ [mm]) のように比較的小さなものでも障害物として認識可能であることを確認している。

## 4. 結論

小型距離画像センサをヒューマノイドに搭載するための統合的なシステムを構築した。さらに、小型距離画像センサを用いて簡略的に平面領域検出が可能であることを示し、平面領域検出における精度評価を行った。小型距離画像センサから得た情報をヒューマノイドにフィードバックし、ヒューマノイドが障害物を認識して自律的に歩行を停止するのを確認

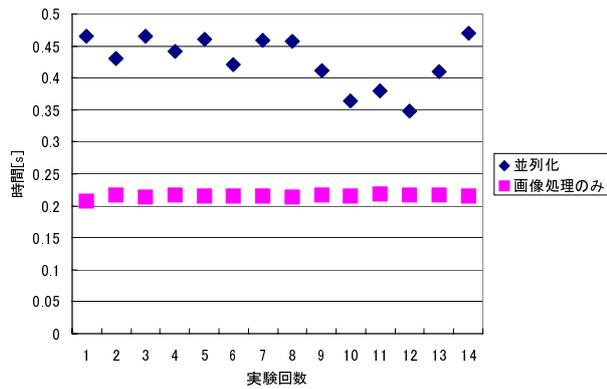


Fig.7 Comparison of processing time

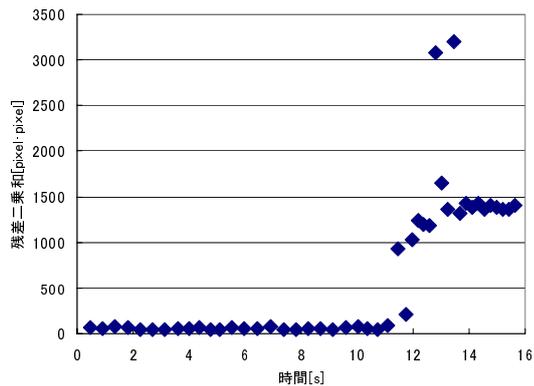


Fig.8 Change of residuals

することができた．ヒューマノイド制御プロセスと画像処理プロセスを並列化したことによる画像の処理時間への影響を確認した．

本論文では、ヒューマノイドが床面上で障害物を認識して歩行を停止するまでにとどまったが、今後は、検出した障害物を自律的に回避させるべく、より自由度の高い動作生成を行う必要がある．各動作を使い分けて障害物を回避させることが課題となってくる．また、現在ではヒューマノイドの指令プロセスと並列化したときの小型距離画像センサの画像一枚あたりの処理時間は約 450[ms] と比較的処理速度が遅いため、今後は処理速度の向上を目指す．

#### 参考文献

- [1] 加賀美聡他, プレーンセグメントファインダ, 第 5 回ロボティクスシンポジウム予稿集, (2000), pp. 381-386.
- [2] 梅田和昇, 平面領域検出における相対視差画像の利用の検討, 第 20 回日本ロボット学会学術講演会予稿集, (2002), 2A16.
- [3] 渡辺圭, 小型距離画像センサを用いた二足歩行ロボットの障害物認識, 第 23 回日本ロボット学会学術講演会予稿集, (2005), 1B18.
- [4] Kazunori Umeda, A Compact Range Image Sensor Suitable for Robots, Proc. 2004 Int. Conf. on Robotics and Automation, (2004), pp. 3167-3172.
- [5] David R. Butenhof, Programming with POSIX Threads, アジソン・ウェスレイ・パブリッシャーズ・ジャパン, 東京 (1998) pp. 21-31.

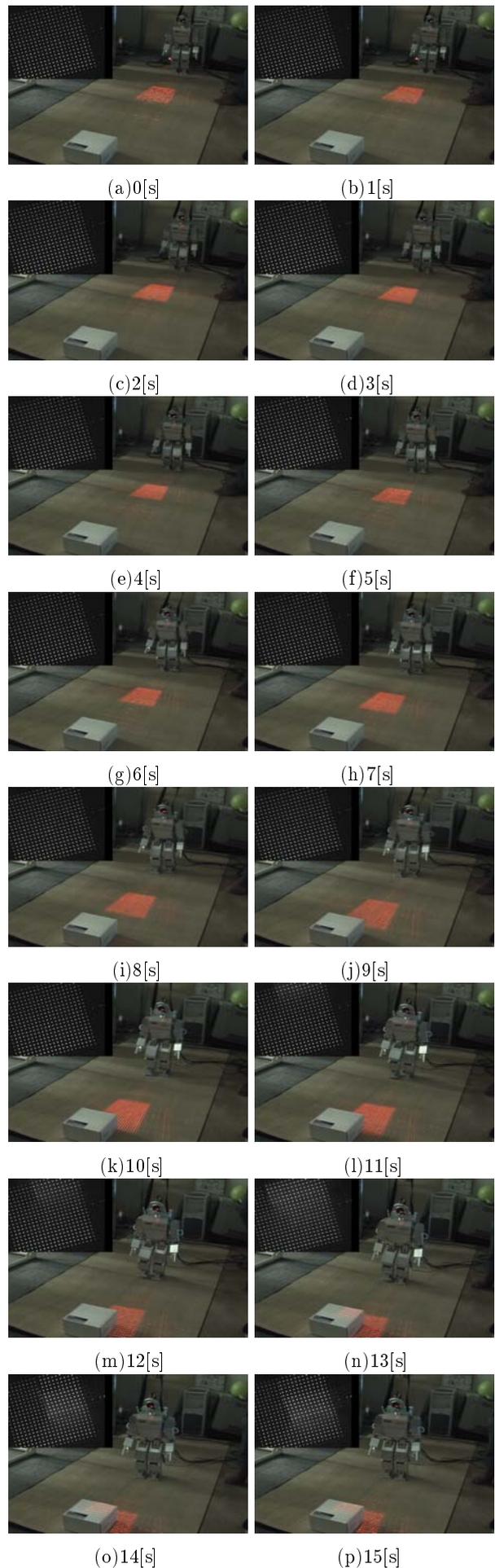


Fig.9 Experimental scene