

## 壁面を利用したピン型入出力デバイスのためのカメラを用いた位置検出手法

岸野泰恵<sup>†</sup> 寺田 努<sup>†</sup> Nicolas Villar<sup>‡</sup> Hans Gellersen<sup>‡</sup> 西尾章治郎<sup>†</sup>

<sup>†</sup> 大阪大学大学院情報科学研究科

<sup>‡</sup> Computing Department, Lancaster University

あらまし：筆者らの研究グループでは、壁面をネットワークとして使い、そこにさまざまなデバイスを刺し込むことでユビキタスコンピューティングを実現する Pin&Play をこれまでに提案した。すでにネットワーク化された掲示板と画鋸を用いたプロトタイプシステムを実現している。実際のアプリケーションで本システムを用いる場合、刺した画鋸の位置を知ることが重要になるが、これまで画鋸が刺さっているかどうかを識別することしかできなかった。そこで本研究では、掲示板の前に設置したカメラを用いてピンの位置を検出する手法を提案する。提案手法では、設置された周囲の照明の条件が変化しても正しくピンを検出できるように、状況に応じて動的に画像処理パラメータを変更する。本研究ではさらに、提案手法の有効性を確認するために評価実験を行った。

### A Position Detection Mechanism using Camera Images for Pin-Shaped Input/Output Devices

Yasue Kishino<sup>†</sup>, Tsutomu Terada<sup>†</sup>, Nicolas Villar<sup>‡</sup>, Hans Gellersen<sup>‡</sup>, and Shojiro Nishio<sup>†</sup>

<sup>†</sup> Graduate School of Information Science and Technology, Osaka University

<sup>‡</sup> Computing Department, Lancaster University

**Abstract :** We have proposed Pin&Play system which uses the wall surface as a network infrastructure and controls inserted devices to realize ubiquitous computing. We also implemented a prototype system using a board and pushpins. Though the positions of the inserted pins are important for most applications, we could know only information that these pins are alive or not. Thus, we propose a position detection mechanism for the pins using a camera which is placed in front of the board. The proposed method has a dynamic adjustment method of image processing parameters, to get correct position even when the surrounding light condition changes. Moreover, we also clarify the effectivity of our proposed method by performance evaluation.

## 1 はじめに

近年いつでもどこでもコンピュータを利用できるユビキタスコンピューティング環境に対する注目が高まっている。ユビキタスコンピューティング環境においては、ユーザの身近なあらゆる場所にさまざまなデバイスが存在し、それらが連携してユーザにサービスを提供することが想定されている。このようにあらゆる場所にデバイスが存在する環境を実現するには、それらのデバイスへの電源の供給や通信手段が課題となる。

通信手段は、無線のネットワークと有線のネットワークの2種類に大きく分類できる。無線のネットワークを用いる場合には、セキュリティやデバイスへの電源供給のバッテリーが問題となり、有線のネットワークを用いる場合には、ネットワークを構築し、電源を確保するためのケーブルが問題となる。これらの問題を解決するために、筆者らは壁を通信と電源供給の手段として利用することに注目し、これまでに Pin&Play と呼

ぶ枠組みを提案した [4]。壁はあらゆる場所にあり、壁を利用した電源の確保や通信が可能になれば、有線のネットワークを敷設する必要もなく、無線の場合のセキュリティや電源の問題も解決する。

Pin&Play では、壁面などをネットワークと電源の供給源とし、そこにさまざまなデバイスを刺し込むことでユビキタスコンピューティングを実現する。これまでに、掲示板(ボード)と画鋸(ピン)を用いたプロトタイプシステムを実装した。掲示板に紙を留める際には、重要なものを中央に配置したり、時系列に沿って並べるなど、その位置が重要な意味をもつ。さらに、さまざまなデバイスを壁面に設置する際にも、デバイスの位置によって低い場所ならば子供向けの動作をし、高い場所ならば大人向けの動作をするなど、デバイスの位置が重要となる。しかし、これまでどのピンがボードに刺されているかは検出できるが、ピンが刺された位置を検出することはできなかった。そこで、本研究

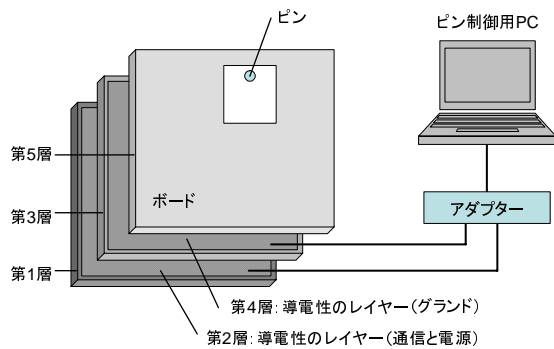


図 1: Pin&Play のシステム構成図

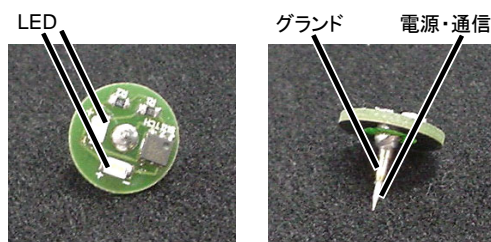


図 2: Pin&Play のピン

ではボードをカメラで撮影し、画像処理によってピンの位置検出を行う手法を提案する。

以下、2章で Pin&Play の概要について述べ、3章でピンを検出するアルゴリズムの詳細を説明し、4章でその実装について述べる。5章でピンの検出精度の評価実験について報告し、6章で考察を行い、7章でまとめる。

## 2 Pin&Play

Pin&Play のシステム構成図を図 1 に示す。システムは通信および電源供給を行なうボードとそこに刺さるピン(図 2)からなる [6]。ボードは 5 層からなり、第 2 層と第 4 層には導電性繊維の布地からなり、第 1 層、第 3 層、第 5 層は絶縁性の素材である。第 2 層は通信と電源供給のためのレイヤーであり、第 4 層はグラウンドのレイヤーである。ボードの作成途中の様子を図 3 に示す。このボードにピンが刺し込まれるとピンは図 2 右のように 2 つの端子をもち、それぞれがボードの導電性の各レイヤーと接続される。また、ピンは ID をもち、2 色の LED (図 2 右) を備えている。ID に対応して掲示物の内容や有効期限などを記録することで、その情報をもとに重要な掲示物の存在や有効期限切れの掲示物をとめるピンの LED を点灯させてユーザに知らせることが可能である。

Pin&Play のボードは、定期的にボード上に存在する

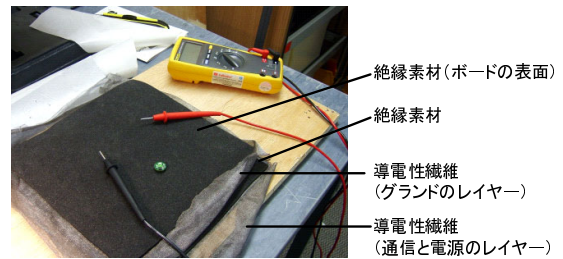


図 3: 作成中の Pin&Play のボード

ピンの ID をチェックすることで新たなピンが刺し込まれたり、ピンが抜き取られたことを検出する。しかし 1 枚のボードの全面が 1 つの接点となるため、ピンの差し込まれた位置は取得できない。

## 3 位置検出手法

位置の検出は、画像処理を用いて行う。画像処理ではなく、ボードをいくつかの領域に分割し、どのボードに刺さったのかを検出することでピンの大まかな位置を取得する手法はすでに実現されている [3]。しかし、大まかな位置しか求められず、ボードの分割数が増えるとシステムの構成が複雑になるという問題があった。その他の手法としてはセンサを用いて位置を取得する方法も考えられる。既存のパブリックディスプレイなどに設置するために、赤外線センサを画面の隅に設置して画面上を触れたものの位置を検出する手法は実用化されているが、複数のものを同時に検出できない場合が多い。本研究では、ピンやボード自体に拡張が不要であり、さらに撮影した画像をアプリケーションでも利用できるという利点からカメラを用いた画像処理を採用する。ピン上の LED を 1 つずつ一定間隔で点滅させ、カメラでこの点滅を撮影した画像の差分画像を作成し、その画像を処理することでピンの位置を算出する。システムの構成図を図 4 に示す。

Pin&Play のシステムは、明るい部屋の中や暗い廊下などさまざまな照明条件下での使用が予想され、さらに設置された場所によっては昼と夜で明るさが変化することも予想される。また、ボードの中でも一部に照明が当たっている場合など、1 台のカメラで撮影した画像中でも条件が変化することが予想される。また、個々のピンの LED の明るさは少しずつ違い、さらに LED には指向性があるためピンが刺し込まれたときの傾きもカメラで取得するピンの画像に影響を及ぼす。位置検出システムはこれらの変化に動的に対応する必要がある。本研究では、画像処理のパラメータを動的に変

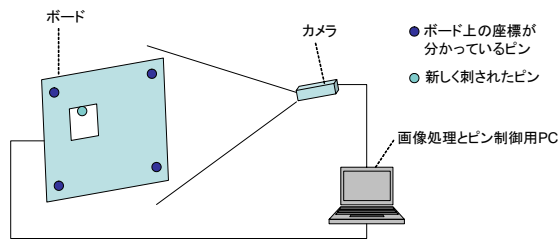


図 4: 位置検出のシステム構成図

えることで、周囲の環境の変化にも対応する。これによって、ボードをさまざまな場所に設置できるようになる。

位置検出では、まずカメラ画像からピンが撮影されている位置を特定し、次に画像上の位置とボード上のあらかじめ測定されているピンの座標を用いてカメラ画像と実際のボード上の座標系の変換式を求める。さらに画像処理に用いるパラメータのいくつかは、取得した画像を利用して動的に決定する。

### 3.1 ピンの検出

画像処理は差分画像を利用して行なう。まず、前フレームとの差分画像を作成し、グレースケールイメージに変換し、雑音の除去のためにぼかしフィルタと最大値フィルタをかける。処理した画像を二値化し、白色の連結領域を探す。領域の数が点滅しているピンの数と同じであれば、画面内のピンが検出できていると仮定し、その座標を記録する。領域の数が点滅しているピンの数よりも多い場合や、領域の大きさがピンだとみなせないほど大きい場合には、カメラとボードの間に何らかの物体があり、ピンの検出に失敗している可能性が高いため、そのフレームからはピンの検出を行わない。この処理を繰り返し、4回同じ座標を検出すると、その場所にピンがあるとみなす。

### 3.2 画像処理パラメータの決定

新たなピンを発見した際には、その際の画像を用いてその後のピンの発見に最適な画像処理パラメータを求める。本研究では、差分画像の二値化の閾値とピンだとみなす白色連結領域の最大の大きさを画像処理によって動的に決定する。

二値化の際の閾値は、ピンを発見できた画像を閾値2から2ずつ閾値を増やして画像処理を繰り返し、3回連続ピンを検出できた際の2回目の値を最適な閾値とした。50回この処理を繰り返しても閾値が求められなかった場合は初期値を用いる。また、閾値は1枚のボード上でも照明の条件が変化することを考慮し、1枚の

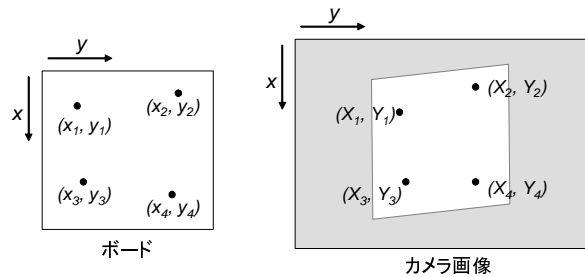


図 5: ボード上とカメラ画像上の座標

画像でも場所によって異なる値を使用する。各画素を二値化する際には、最も近くに存在するピンが検出された際に最適だとされた値を用いる。1本もピンがない場合は、初期値を用いる。

白色連結領域の最大の大きさは、そのピンを発見した際の領域の大きさの2倍とした。ピンと最大の大きさを記録し、同じIDのピンが再度刺されたときには記録された値を用いて画像処理を行う。記録されていないピンが刺されたときには、固定値1000を用いる。なお、最小のピンの大きさは4に固定した。

### 3.3 座標の変換

カメラで撮影した画像上の座標からボード上への座標の変換には射影変換を用いる。射影変換では、画像上の座標を  $(x, y)$ 、ボード上の座標を  $(X, Y)$  として、以下の式を用いて、座標系の変換を行う。

$$X = \frac{a_1x + b_1x + c_1}{a_0x + b_0y + 1}$$

$$Y = \frac{a_2x + b_2x + c_2}{a_0x + b_0y + 1}$$

この式を利用するためには係数  $(a_0, a_1, a_2, b_0, b_1, b_2, c_1, c_2)$  を求める必要があり、図5に示すボード上の座標と画像上の座標が分かっている4点  $(P_1 : (x_1, y_1), (X_1, Y_1), P_2 : (x_2, y_2), (X_2, Y_2), P_3 : (x_3, y_3), (X_3, Y_3), P_4 : (x_4, y_4), (X_4, Y_4))$  を用い、以下の行列式を解いて算出する。

$$P \begin{bmatrix} a_0 \\ b_0 \\ a_1 \\ b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ X_3 \\ Y_3 \\ X_4 \\ Y_4 \end{bmatrix}$$

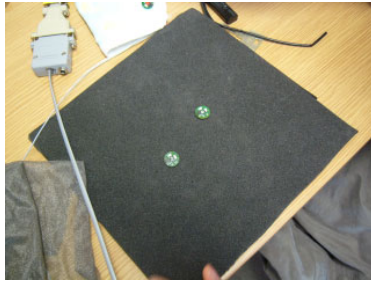


図 6: 実装したピンとボード

ただし、

$$P = \begin{bmatrix} -x_1 X_1 & -y_1 X_1 & x_1 & y_1 & 1 & 0 & 0 & 0 \\ -x_1 Y_1 & -y_1 Y_1 & 0 & 0 & 0 & x_1 & y_1 & 1 \\ -x_2 X_2 & -y_2 X_2 & x_2 & y_2 & 1 & 0 & 0 & 0 \\ -x_2 Y_2 & -y_2 Y_2 & 0 & 0 & 0 & x_2 & y_2 & 1 \\ -x_3 X_3 & -y_3 X_3 & x_3 & y_3 & 1 & 0 & 0 & 0 \\ -x_3 Y_3 & -y_3 Y_3 & 0 & 0 & 0 & x_3 & y_3 & 1 \\ -x_4 X_4 & -y_4 X_4 & x_4 & y_4 & 1 & 0 & 0 & 0 \\ -x_4 Y_4 & -y_4 Y_4 & 0 & 0 & 0 & x_4 & y_4 & 1 \end{bmatrix}$$

である。

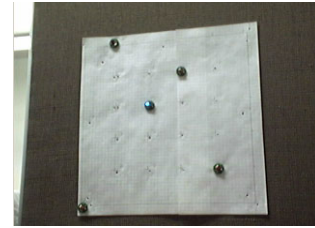
## 4 実装

### 4.1 Pin&Play の実装

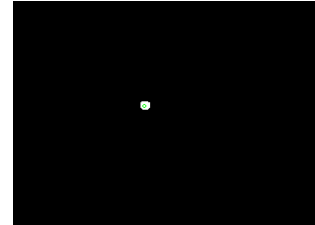
Pin&Play のプロトタイプシステムはダラスセミコンダクター社の 1-Wire ネットワークを利用して実装した。このネットワークでは、個別の ID をもった IC を 2 本の線のみからなるネットワークに接続することで、ON/OFF の制御、センシングなどが行える。ピンの実装にはスイッチの機能をもつ DS2406 を使用している。このスイッチ機能を利用して LED の ON/OFF を制御している。LED には高輝度の青色と緑色のものを用い、LED に流れる電流を調節するため 6.8K オームの抵抗を実装した。ネットワークはシリアルポートから制御する。シリアル通信と 1-Wire ネットワークのアダプタにはフィリップス社の DS9097U-E25 を用いた。ネットワークへの電源は計算機からシリアルポートを通じて供給される。図 6 に実際に実装したピンとボードを示す。図 6 の左上にあるのは使用したアダプタである。

### 4.2 提案手法の実装

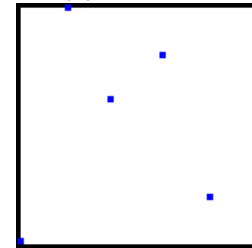
以上で説明したアルゴリズムを Microsoft Visual C++ を用いてノートパソコン (NEC Lavie LL750/4, CPU:1.5GHz, メモリ:752M, OS:Windows XP Home edition) 上に実装した。カメラは USB カメラ (Logicool



(a): カメラ画像



(b): 差分画像



(c): 算出したピンの位置

図 7: 画像処理の手順

Qcam for Notebooks Pro) を用い、 $320 \times 240$  のサイズの RGB 各色が 8bit の画像を使用する。画像の取得には DirectShow を利用し、画像のグレースケール化、ぼかしフィルタ、最大値フィルタにはインテルの IPL ライブラリを利用した。1 フレームの画像処理には 30 ミリ秒程度必要であるため、ピンの LED の点滅間隔は 200 ミリ秒として実装した。また、4 回同じ座標でピンを検出した際に、ピンがその位置にあるとみなすが、撮影した画像の微妙な変化によって座標がずれることが多いため、 $\pm 2$  の誤差を許容する。二値化の際の閾値の初期値は 10 とした。

実際の画像処理の様子を図 7 に示す。図 7(a) のような画像を取得し、これから差分画像を作成すると図 7 のような画像が得られる。ここから射影変換を用いて座標の位置を計算した結果を図 7(c) に示す。図 7(c) は図 7(a) 中の白い紙が貼られた部分のピンの配置を示している。このようにほぼ正確にピンの位置を取得できることを確認した。

さらに、ピンの抜き差しを管理し、新たなピンが刺されたことを検出すると画像処理を行い、ピンの位置

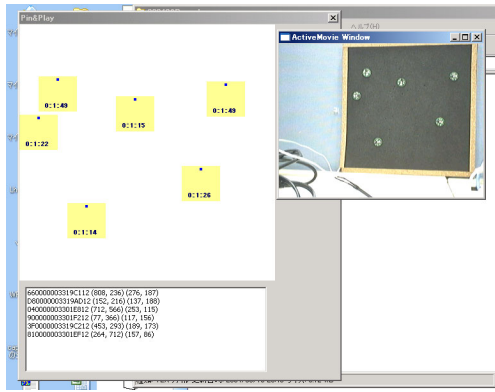


図 8: ピン管理アプリケーションのスクリーンショット

を決定する簡単なアプリケーションを実装した。図 8 にスクリーンショットを示す。左側のウィンドウにピンの位置と差し込まれてからの経過時間を表示し、右側のウィンドウにはボードを撮影した画像を表示している。

## 5 実験

実現した手法の精度を確認するため評価実験を行った。実験には約 25cm 四方のボードと 6 個のピンを用いた。ボード上の座標は整数型で管理し、左上を座標 (0, 0)、右下を (1000, 1000) とした。4 個のピンのボード上の座標をあらかじめ入力し、残りの 2 個のピンの座標の位置を求めることで実験を行った。また、ピンを発見できない場合もあるため、1 本のピンにつき 2 秒のタイムアウトを設け、これを超えても発見できなかった場合は、そのピンの検出をあきらめ、他のピンの検出を先に試す。

実験では、検出したボード上の位置の誤差と、ピンを検出するまでに必要なピン上の LED の ON/OFF 切り替え回数を計測した。ピンの検出した位置がボード外であった場合は、その検出結果は無効であったとみなし、位置の誤差の計算には含めない。あるピンの検出を終えると、同時に次のピンを点灯させ点滅を開始する。ピンは最短で、最初に他のピンが点灯していなければ 4 回、点灯していれば 5 回 ON/OFF を切り替えることで発見できる。画像処理パラメータの算出を含めてすべての操作を何度か繰り返した。

まずどの程度の距離離れた場所にカメラを設置してもピンを発見できるかを実験した後、斜めにカメラを設置した場合、真っ暗な部屋に設置した場合、ボード上に濃い色の紙を貼った場合など、さまざまな条件下でも提案手法の有効性を調べた。



(a): 350cm で撮影した画像  
中央にボードを設置。



(b): (a) の画像を撮影した地点での閾値の分布  
(色が薄い順に閾値 10, 閾値 8, 閾値 6)

図 9: 実験の様子

### 5.1 距離と精度に関する実験

ボードとカメラの間の距離と位置の誤差の関係について実験を行った。50cm の距離から実験を開始し、50cm ずつ距離を離していった。実験は明るい照明の屋内で行い、カメラはボードのほぼ正面に設置した。実験の様子を図 9(a) に示す。これは 350cm は離れた地点で撮影された画像である。この際の閾値の分布の一例を図 9(b) に示す。各点で 2 個のピンを発見する操作を 20 回ずつ繰り返し、5 回ごとにカメラの位置を少し変更して設置しなおした。しかし、350cm を超えるとピンを発見できなくなる場合があったため、その際は途中で発見を中断し、中断した回数も記録した。

距離と誤差に関する実験結果を図 10 に示す。横軸はボードとカメラの間の距離、縦軸は算出した座標と本来の座標の間の距離であり、ボード外の座標が算出された場合は除いた誤差の平均である。距離とピンの ON/OFF の切り替え回数との関係を図 11 に示す。横軸はボードとカメラの間の距離、縦軸はピン発見までに ON/OFF を切り替えた回数であり、タイムアウトになった場合を除いて結果を集計している。また、ピン

表 1: 各点におけるタイムアウト, 実験中断, ボード外を検出した割合

距離 (cm)	パラメータ決定後のタイムアウト	パラメータ決定前のタイムアウト	実験中断	ボード外を検出
50	0.08	0	0	0
100	0	0	0	0
150	0	0	0	0
200	0	0	0	0
250	0.02	0	0	0
300	0.019	0.25	0	0
350	0.27	0.55	0.15	0.03
400	0.34	0.53	0.7	0.4

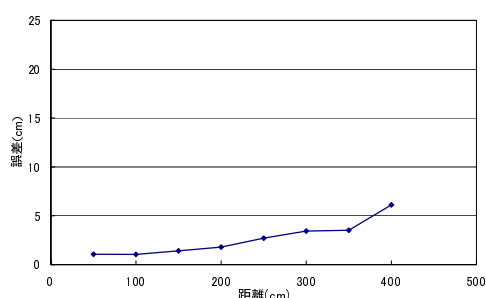


図 10: 距離と誤差の関係

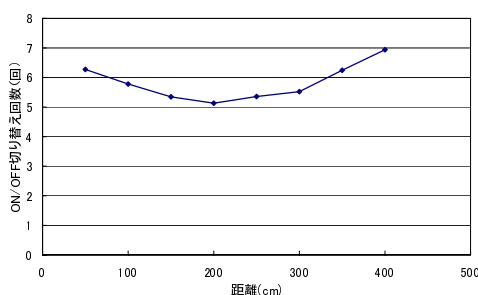


図 11: 距離と点滅回数の関係

の検出中にタイムアウトになった回数, 実験を中断した回数, ボード外だとみなされる座標を算出した割合を表 1 に示す. 最初の 1 個のピンを見つけるまでは画像処理パラメータに初期値を使用するため, 1 個目が見つかるまでのタイムアウトをパラメータ決定前のタイムアウトとし, それ以降のタイムアウトをパラメータ決定後のタイムアウトとしている.

結果より, 距離 250cm 以下ならば 1 割以下の誤差でピンの位置を算出でき, 検出の際のタイムアウトもほとんど発生しないことがわかった. また, 距離が 350cm, 400cm となるとタイムアウトや発見をあきらめる回数が増え実用に適さないことがわかった. また, 画像処理パラメータを設定することで遠い距離でのタイムア

ウト回数が減少することが分かった.

## 5.2 さまざまな環境下での実験

次に, 安定してピンを検出できることがわかった 100cm の距離にカメラを設置し, 以下の 3 つの条件下と明るい照明の部屋で提案手法の動作を確認した. また, 比較のために二値化の閾値を提案手法を用いて動的に決定した場合と, 閾値を 4, 10, 16 に固定した場合を実験した.

- 濃い色の紙が貼られた場合: 原色の赤, 黄, 青の 3 枚の紙をボードに貼った.
- 真っ暗な部屋
- カメラがボードの正面ではなく斜め下に設置された場合: ボードをカメラより 35cm 高い台の上に設置した.

結果を表 2 に示す. 閾値を固定すると, 今回のシステムでの最適な閾値に近い 10 を用いた場合は, 提案手法とほぼ同等の性能を示すが, 閾値に 4 を用いると真っ暗な場合にはピンを検出できず, 閾値に 16 を設定すると LED の光が弱まる斜めからの場合にタイムアウトの回数が増えピンの検出に時間がかかることが分かった. これに対して提案手法で動的に閾値を決定すると, 固定の閾値に性能が劣る場合もあるが, いずれの場合でもピンの座標の取得に成功している. なお, 閾値は斜め下からの場合は 6 程度の場合が多く, 部屋を真っ暗にした状況では 10 から 14 程度になった. これより, 閾値を動的に変化させることで, さまざまな環境でも精度よくピンの座標を取得できることが分かった. 新たな環境にシステムを設置する際や, 新たなカメラを用いる場合など最適な閾値が分からない場合でも, 提案手法を用いれば細かく閾値を設定する必要なくシステムを設置することが可能となる.

表 2: さまざまな環境下での性能

通常			
閾値	誤差 (cm)	切り替え回数	タイムアウト率
動的	0.417	5.06	0
4	0.337	7.2	0.16
10	0.772	5.9	0.04
16	0.225	5.54	0.12

斜めからの場合			
閾値	誤差 (cm)	切り替え回数	タイムアウト率
動的	0.46	6.46	0.22
4	2.60	6.33	0.14
10	2.82	5.96	0.04
16	0.97	6.77	0.46

真っ暗の場合			
閾値	誤差 (cm)	切り替え回数	タイムアウト率
動的	1.15	7.06	0.08
4	-	-	-(実験不可能)
10	1.92	6.5	0.02
16	1.48	5.76	0.08

濃い色の紙が貼られた場合			
閾値	誤差 (cm)	切り替え回数	タイムアウト率
動的	0.75	5.36	0
4	0.69	6.73	0.04
10	0.48	5.18	0
16	0.08	5.42	0.02

## 6 考察

### 6.1 位置精度

実験の結果、ボードとカメラの間の距離が 250cm 以下ならば 1 割以下の誤差でピンの位置を算出できることが分かった。実験の様子を観察したところ、カメラで撮影したピンの位置はほぼ正確に検出できていることが確認できたため、距離が遠くなるにつれて誤差が増えているのは、射影変換の計算の際の誤差が大きいためであると考えられる。今回の実験では小さいボードを使ったため、距離が遠いところでは、画面上の座標の少しのずれが大きく結果に影響を及ぼし、誤差が大きくなったと考えられる。実際の掲示板でピンの位置を検出する場合には、ボードは実験に使ったものよりもかなり大きなものを使うため、その上で十分に間隔をあけてボード上の座標が分かっているピンを配置すればこのような誤差はかなり小さくなると予想できる。掲示板などのアプリケーションではこの程度の精度でも十分だと考えられるが、ボード上の座標の射影変換の係数の計算での精度を上げることでさらに精度が上がるのではないかと予想している。

また、長期間にわたってピンの位置検出プログラム

を動作させると、最初に座標を設定されたピンが他の用途のために別の場所に刺され、位置が確かに分かっているピンの数が減る。そのような状況でカメラのずれなどのために射影変換のパラメータを再計算すると、誤差が大きくなる。このような誤差を減らすため、今後は各ピンの位置の信頼性を記録することを考えている。位置が確かなピンには高い信頼度を設定し、信頼度の高いピンの座標から位置を求めたピンに次に高い信頼度を設定し、高い信頼度のピンを優先的に使用して射影変換のパラメータを算出すれば、誤差の蓄積を防げる。

### 6.2 ピン検出性能

実験では、距離が 400cm になると、ピンを発見する際にタイムアウトが発生することが多くなり、ピンを見つけにくくなることが分かった。これは距離が離れすぎたため、ピンが暗く、画像上に小さく映るためだと考えられる。ピンはシリアルポートから電源の供給を受けているため、できるだけ消費する電力量を抑える必要があるが、ピンの LED の明るさが強くなれば距離が伸びると予想し、ピンの LED のための抵抗を 6.8K オームから 2.8K オームに変更したところ 500cm の距離での検出が可能となった。これより、Pin&Play を設置する場所の状況に応じて、カメラを遠く設置する必要があるときには、明るさの強いピンを用意するとさらに性能が向上することが分かった。

また、実際に運用する際には、照明の環境は昼と夜などで変化するため、定期的に、あるいは、照明の大きな変化を捉えて自動的に画像処理パラメータを動的に変更する機能が必要となる。

さらに実際にシステムを設置すると、カメラがボードに紙を貼る人の動きを撮影し、ピンが隠されたり、人の動きが画像処理アルゴリズムの雑音となりピンを検出できない場合が多く発生すると考えられる。一般的なカメラを用いた位置検出手法では、これらの問題を解決するために複雑な処理を必要としたり、誤検出が増加するなどの問題があった。これに対し提案手法では、画像処理プログラムがピンの点滅タイミングを制御するため、画面上に撮影されるべきピンの状態を得ることができ、より確実な画像処理が可能となる。

### 6.3 関連研究

Pin&Play と同じように画紙をインタフェースとするコンピューティングとしては、PushPin Computing がある。PushPin Computing でもピンの位置検出手法

が研究されている [2] . この研究では光と超音波を送信する pinger と呼ぶデバイスを設置して , 両者がピンに到達するまでの時間の差からピンの位置を求めている . 誤差が 4cm 程度と精度がよいが , ピン自体にも拡張が必要であり , ピン以外にも位置検出専用のデバイスが必要である .

画像処理を用いてデバイスの位置を取得する方法は , これまでにも盛んに研究されている . 赤外線タグを点滅させるシステム [1] や , 高速に点滅するビーコンを特殊なカメラで撮影するシステム [5] などがある . これらの研究は , カメラを制御する計算機とデバイスが接続されていない状況で , デバイスの位置を検出する手法についての研究であり , いつデバイスが画面内に出現するか分からないため , 外乱の少ない赤外線領域を用いたり , LED を高速に点滅させている . 本研究では , 点滅するピンとカメラがネットワークで接続されているため , 画像処理プログラムはピンの LED の状態を画像処理で利用できる . さらに , ピンの位置に大きく誤差があると予想される場合に再度ピンを点滅させて画像処理を繰り返すことで , より正確な位置が得られる .

## 7 まとめ

本稿では , カメラを用いた Pin&Play のピンのための位置検出手法を提案した . また , 評価実験によって 250cm の距離までピンの位置を取得できることを示し , 提案手法の有効性を明らかにした . 提案手法によってピンの位置を利用したアプリケーションの構築が可能となる .

今後の課題としては , 複数のボードと複数のカメラが存在する状況への提案手法の拡張 , 遠隔地からボードやカメラを利用したアプリケーション構築のための枠組みの実現や , 提案手法を用いたアプリケーションの構築が挙げられる .

## 謝辞

本研究の一部は , 文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」によるものである . ここに記して謝意を表す .

## 参考文献

- [1] 青木恒: カメラで読み取る赤外線タグとその応用, インタラクティブシステムとソフトウェア VIII, 日本ソフトウェア科学会 WISS 2000, 近代科学社, pp. 131-136 (Dec. 2000).

- [2] M. Broxton: “Localization and Sensing Applications in the Pushpin Computing Network,” master of engineering thesis, MIT (Jan. 2005).
- [3] F. Helin, T. Hoglund, R. Zackaroff, M. Hakansson, S. Ljungblad, and L. E. Holmquist: “Supporting Collaborative Scheduling with Interactive Pushpins and Networking Surfaces,” in Proc. of UbiComp 2004 Adjunct Proceedings (Spet. 2004).
- [4] K. Van Laerhoven, N. Villar, A. Schmidt, H.-W. Gellersen, M. Hakansson, and L. E. Holmquist: “Pin&Play: The Surface as Network Medium,” IEEE Communications Magazine, Vol.41 No.4., ISSN: 0163-6804, IEEE Press, pp. 90-96 (Apr. 2003).
- [5] 松下伸行, 日原大輔, 後輝行, 暦本純一, 吉村真一: ID Cam: シーンと ID を同時に取得可能なイメージセンサ, 情報処理学会シンポジウムシリーズ (インタラクシオン 2002), Vol. 2002, No. 7, pp. 6-13 (Mar. 2002).
- [6] Pin&Play: <http://ubicomp.lancs.ac.uk/pin&play/>.