

実世界型電子付箋のための ARKit を用いた 自己位置認識による紙付箋認識について

伊藤 栄俊^{†1,a)} 大園 忠親^{†1,b)} 新谷 虎松^{†1,c)}

概要: 我々は、実世界型電子付箋システム Mahoca を開発している。Mahoca では、物理付箋と電子付箋の双方を用いた会議の支援を行っている。本稿では、Mahoca における物理付箋の電子化の際に、物理付箋の位置に電子付箋を自動で配置するシステムの開発を行った。本研究の特色は、付箋の位置合わせの際に、ARKit を利用した現実空間にもとづき行った。これにより、簡便な方法で現実空間と仮想空間の位置合わせを実現した。タッチパネル等のセンサーを含む特殊なディスプレイ上に投影された画面に対して行うわけではない。水平な机の上にプロジェクターで投影した画面の枠内に存在する物理付箋に対して、物理付箋の置かれた位置に変換された電子付箋を配置する。位置合わせの評価を行ったところ、水平方向に対して平均 0.9cm 、垂直方向に平均 2.0cm ほどの誤差が生じる結果となった。よって、本手法により、物理付箋の置かれた位置に電子付箋を配置することが可能であることが示された。

On Sticky Note Extraction using World Tracking by ARKit for a Reality-based e-Sticky Notes System

ITO EISHUN^{†1,a)} OZONO TADACHIKA^{†1,b)} SHINTANI TORAMATSU^{†1,c)}

Abstract: We are developing a reality-based e-sticky notes system, named Mahoca. Mahoca supports conferences using both physical notes and e-sticky notes. In this paper, we developed a system that automatically recognizes e-sticky notes at the position of physical sticky notes when importing physical sticky notes into Mahoca. The feature of this research was based on the real space using augmented reality technology at the time of aligning the notes. As a result, the real space and the virtual space were aligned in a simple way. We did not use a special purpose display based on specialized sensors. Our system can extract e-sticky notes with absolute positions in the real world by using limited sights on a horizontal desk. When the alignment was evaluated, an error of 0.9cm on the average in the horizontal direction and 2.0cm on the average in the vertical direction resulted. Therefore, our method can arrange e-sticky notes at the position where the physical sticky notes are placed.

1. はじめに

会議を行う際に、付箋の紙を用いられることがある。アイデアを付箋に書き出し、ホワイトボードなどに貼り付ける。付箋は自由に記述、配置ができ、チーム内でアイデアを共有しやすいという利点がある。過去に行われた会議における、紙の付箋の内容だけでなく、配置方法や付

箋のグループといった情報は、現在の会議の参考にするときに有益である。しかし、再利用を考えた場合、電子化された付箋の方が、過去の状態を保存、管理できる点で優れている。我々は実世界型電子付箋システム Mahoca を開発した。Mahoca は、物理付箋を取り込み、電子付箋に変換したり、直接電子付箋を作成したりするシステムである。電子化された付箋情報をもとに関連情報の推薦などを実現している [2] [3]。

本研究で課題としているのが、物理付箋と電子付箋の境界をできるだけ小さくし、電子付箋も物理付箋と同じように扱えることである。物理付箋を電子化する際、物理付箋

^{†1} 現在、名古屋工業大学
Presently with Nagoya Institute of Technology
a) ashun@toralabj.org
b) ozono@toralabj.org
c) tora@toralabj.org

をカメラで撮影し、付箋の情報を抽出する機構はこれまでに開発している。しかし、現実空間の座標系とシステム上の座標系が異なるため、物理付箋の置かれた位置に電子付箋を配置することができていなかった。そのため、電子化された付箋を取り込まれた後、再配置する必要がある。そこで、本研究では、ARKit による現実空間と仮想空間の位置合わせを行うことで、実世界型電子付箋システムにおける付箋抽出の際の位置合わせの問題の解決を行う。本研究で使用する付箋の大きさは、縦横ともに 7.5cm のものを使用する。故に、物理付箋を電子化した際に、電子付箋が物理付箋を重ねる程度、つまり、水平方向および垂直方向の位置合わせの誤差が 7.5cm 以下になることを目的とする。また、本研究において特筆すべき点として、提案手法では、タッチパネル等のセンサーを含む特殊なディスプレイ上に投影された画面に対応した位置に付箋を置くのではなく、水平な机の上にプロジェクターで投影した画面の枠内に存在する物理付箋に対して、物理付箋の置かれた位置に変換された電子付箋を配置する。

2. 実世界型電子付箋システム Mahoca

実世界型電子付箋システム Mahoca は、アイデアの再利用のために電子付箋にアイデアを入力し、会議参加者間で共有するためのシステムである。Mahoca では、会議参加者が「スクリーン」と「クライアント」の二つのアプリケーションを利用することで、電子付箋を利用した会議を行う。図 1 に Mahoca を使用している様子を示す。スクリーンは、公開された電子付箋を画面に表示し、会議参加者に共有するアプリケーションである。スクリーンでは、会議に対応する「ベースボード」と呼ばれる画面を全面に表示する。ベースボード上には会議参加者が公開した電子付箋が表示される。スクリーンは、水平な面に対してプロジェクターで投影する状況を想定している。図 1 におけるスクリーンでは、物理付箋と電子付箋の両方が置かれている。物理付箋と電子付箋の境界をできるだけ小さくし、同じように扱えるようにすること目的としている。

クライアントは、電子付箋を作成、または編集してベースボードに公開するために利用されるアプリケーションである。クライアントは、各会議参加者が持つスマートフォンやタブレット端末上で動作する。電子付箋には、付箋に入力したテキストや手書きメモの他に、メタ情報としてテキストのメモと任意の URL を入力して付加できる。また、物理付箋を電子化の際は、クライアントのカメラから物理付箋の撮影を行い、撮影画像から付箋情報を抽出し、付箋の電子化を行う。

Mahoca では、会議参加者は、クライアントを利用して自分の意見を書いた電子付箋を作成する。作成した電子付箋をスクリーンで表示したベースボード上に公開することで、他の会議参加者に意見を共有する。ベースボード上の



図 1 Mahoca を使用している様子
Fig. 1 Usecase of Mahoca.

電子付箋で他者の意見を確認し、議論をすることで会議を進行できる。

クライアントでは、物理付箋を取り込むための付箋取り込み機構が存在する。付箋取り込み機構では、スマートフォンやタブレットで撮影された画像を元に付箋情報を抽出し、スクリーン上に投影することができる。付箋情報を取り込む際は、内容を抽出するために取り込みたい付箋のみが画面に写ることが望ましい。そのため、画像中に取り込みたい付箋の周囲の状況が映らない。物理付箋がスクリーン上のどの位置に置かれたのか把握するためには、取り込んだ付箋がある基準点に対してどの位置に存在しているのかを知らせなければならない。

電子付箋を物理付箋の位置に自動で配置するためには、物理付箋がスクリーン上のどの位置に置かれたのか推定する必要がある。しかし、付箋の内容が読み取れる程度にカメラを近づけた場合、周囲の位置情報がカメラで捉えられないため、スクリーン上のどこに物理付箋が置かれたのか分からなくなってしまう。物理付箋の取り込みの際の課題をまとめると、一つ目は現実空間に存在する付箋と投影されたスクリーンの認識することである。二つ目は、スクリーンが投影された平面を認識し、スクリーンを基準とした平面上の座標系を取得することである。三つ目は、現実空間の座標系と Mahoca のスクリーン上の座標系、つまりベースボードにおける位置とのキャリブレーションである。これらの課題を解決するために、本研究では ARKit を採用した。拡張現実とは、現実空間の情報をもとにした仮想空間に、三次元モデルなどの情報を付加することで現実空間の情報を拡張する技術である。本研究では、拡張現実の実現のために使われる平面検出をもとにした仮想三次元空

間を用いて問題の解決を図る．本研究では，平面検出を行い，現実の三次元空間を認識し，スクリーンや付箋の三次元情報を取得し，投影されたスクリーン上のどこに物理付箋が置かれたのかを検出する．3節で詳細を述べる．

3. 紙付箋認識

紙付箋認識を行うために，本研究では，平面検出，衝突判定，平面領域の生成，電子付箋の作成，キャリブレーションの手順で処理を行うことにした．それぞれの処理について述べる．現実空間の認識のために，ARKitの一つである平面検出を利用する．カメラ映像をもとに平面検出を行い，鉛直方向を z 軸とした，右手座標系の三次元空間にもとづく，スクリーンの位置と付箋の位置の認識を行う．センサーデバイスには，スマートフォンやタブレット付属のカメラを利用する．水平な平面上にスクリーンが投影されている状況を想定する．プロジェクターを用いて，平面上に投影されたスクリーン画面の位置と大きさを取得する．平面上に存在する領域を定義するために，ユーザの指定した平面との衝突判定により，取得した World 座標系における3点の三次元座標 $\{X_0, X_1, X_2\}$ を用いる．ユーザはスマートフォンもしくはタブレットのカメラ画像に対してタッチされた地点をもとに衝突判定を行う． X_0 は基準点として用いる．スクリーンの縦方向の成分 e_x と横方向の成分 e_y は次の式の通りに定める．

$$e_x = \frac{X_1 - X_0}{|X_1 - X_0|}$$

$$e_y = \frac{X_2 - X_0}{|X_2 - X_0|}$$

現実空間上における，スクリーン上の付箋の位置の取得について述べる．スクリーンが投影されている領域が定まっているならば，付箋の抽出と配置を行う．まず，付箋の位置が定義した領域内であるか衝突判定を行う．付箋が領域内である時，スクリーン上の指定位置に電子化した付箋を配置する処理を行う．付箋の位置は，スクリーン領域の各成分 e_x と e_y に対する値から決める． e_x と e_y はそれぞれ単位ベクトルであるため，付箋のスクリーン領域の基準点に対する相対的な位置ベクトル $X'_c = (x'_{cx}, x'_{cy})$ は次の式の通りとなる．ただし，World 座標系における付箋の位置ベクトルを X_c とする．

$$x'_{cx} = X_c \cdot e_x$$

$$x'_{cy} = X_c \cdot e_y$$

キャリブレーションについて述べる．現実空間の座標系と，スクリーン上での座標系は異なるため，キャリブレーションを行う必要がある．本研究では，以下の環境に限定した条件のもとにキャリブレーションを行う．ここで，スクリーンの領域を矩形と仮定する．付箋が正面見える方向に対して，左上の頂点を V_0 ，右上の頂点を V_1 ，左上の頂

Algorithm 1 A region extraction of an e-sticky note

```

1: plane = detectPlane()
2: result = hitTest(plane, touch.position)
3: point = result.position
4: if  $X_0 \neq null$  then
5:   if  $X_1 \neq null$  then
6:     if  $X_2 \neq null$  then
7:       if screen.isCollision(point) then
8:         position = projection(screen, point)
9:         image = captureImage(camera)
10:        post(image, position, screen)
11:       end if
12:     else
13:        $X_2 = result.position$ 
14:       screen = createScreen( $X_0, X_1, X_2$ )
15:     end if
16:   else
17:      $X_1 = result.position$ 
18:   end if
19: else
20:    $X_0 = result.position$ 
21: end if

```

点を V_2 ，とする． $\{X_1 - X_0, X_2 - X_0\}$ からなる矩形と， $\{V_1 - V_0, V_2 - V_0\}$ からなる矩形が相似になるとすると，電子付箋の位置ベクトル $V_c = (v_{cx}, v_{cy})$ は次のようになる．

$$v_{cx} = \frac{|V_1 - V_0|}{|X_1 - X_0|} x'_{cx}$$

$$v_{cy} = \frac{|V_2 - V_0|}{|X_2 - X_0|} x'_{cy}$$

物理付箋の横幅と縦幅をそれぞれ w_r, h_r とすると，電子付箋の横幅と縦幅 w_v, h_v は次の式のとおりに定まる．

$$width_v = \frac{|V_1 - V_0|}{|X_1 - X_0|} width_r$$

$$height_v = \frac{|V_2 - V_0|}{|X_2 - X_0|} height_r$$

Algorithm 1 に紙付箋認識のアルゴリズムを示す．まず，*detectPlane* 関数により平面を検出する．検出した平面をもとに，ユーザの指定した位置との *hitTest* で衝突判定を行い，現実空間上における座標を取得する．取得した座標を順に， X_0, X_1, X_2 に代入していく． X_2 が代入された段階でスクリーンを示す領域を生成する．領域が生成されたら，衝突判定を行う．領域内に付箋が置かれ選択された時，付箋の画像を撮影し，サーバに位置情報と，位置情報を現実空間のスクリーンのデータとともに *post* にて送信する．

4. システム構成

図2にシステム構成図を示す．本システムは，クライアント，サーバ，スクリーンから構成される．クライアントでは，スマートフォンやタブレットを用いて，付箋の位置情報と画像を取得する．本研究で想定しているクライアントの端末は iPad である．紙付箋認識，及び付箋を含む画

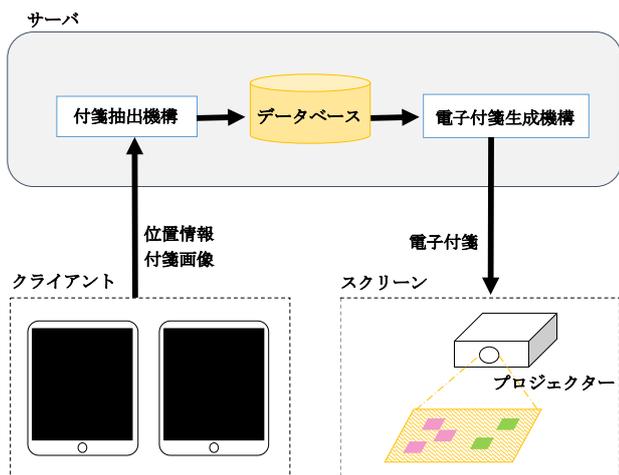


図 2 システム構成図

Fig. 2 System configuration.

像の撮影はクライアントで行う。取得した情報は、サーバに送られ、画像情報から矩形を検出し付箋部分のみを抽出する。付箋の位置情報と抽出された画像データは、付箋データとしてデータベースに送られ記録される。付箋データをもとに、電子付箋作成機構で電子化を行う。電子化を行う際に、現実空間に投影されているスクリーンの大きさとシステム上の大きさをもとに、電子付箋のキャリブレーションを行う。キャリブレーションされた電子付箋は、スクリーンに送られ、プロジェクターから投影される。

図 3 に紙付箋認識の実行例を示す。左上の図がスクリーン上の物理付箋をタブレットで撮影している様子である。右下の図が、タブレットの画面の拡大図。左下の図が、物理付箋の置かれた位置を定規で測定した様子を示している。左上の図に示されているように、実行例では、基準点から物理付箋までの距離を推定している。図 3 では、本手法により紙付箋認識を行い、付箋の位置推定を行ったところ、右下の図に示されているように、基準点から水平方向に 26cm 、垂直方向に 12cm の位置に配置されていることが示された。左下の図に示されているように、基準点から定規で測定したところ、基準点から水平方向に 26cm の位置に物理付箋があることが確認された。

5. 評価と考察

提案手法の妥当性を示すために、紙付箋認識に関する評価実験を行った。評価実験では、物理付箋を置いた位置と、作成された電子付箋の位置との誤差を測定した。60 回付箋の取り込み操作を行い、その平均値と分散値を評価した。単位は cm である。実行環境は、クライアントでは iPad Pro, iOS 11.2 を使用している。プロジェクターは RICOH PJ WX4152, スクリーンは Google Chrome Version 63.0.3239.84 を使用した。表 1 に結果をまとめる。 $horizontal$ はスクリーンに対して水平方向、横方向を表し、

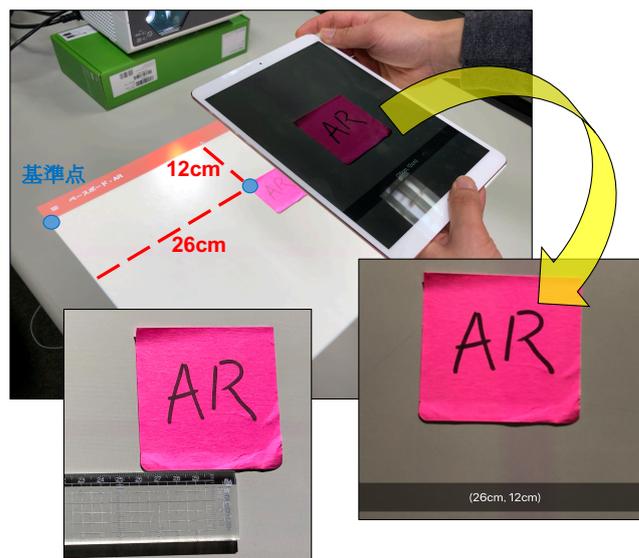


図 3 紙付箋認識の様子。左上の図がスクリーン上の物理付箋をタブレットで撮影している様子。右下の図が、タブレットの画面の拡大図。左下の図が、物理付箋の置かれた位置を定規で測定した様子。

Fig. 3 A region extraction of an e-sticky note: The figure on the upper left shows a tablet shooting a physical sticky note on the screen. The figure on the bottom right is an enlarged view of the screen of the tablet. The figure in the lower left shows the position where the physical sticky notes is placed measured with a ruler.

$vertical$ はスクリーンに対して垂直方向、縦方向を表す。表 1 より、 $horizontal$ 方向のズレの平均値は 0.9cm 、 $vertical$ 方向のズレの平均値は、 2.0cm であった。スクリーンの横幅が 67.0cm 、縦幅が 41.5cm であって、誤差は、水平方向に 1.4% 、垂直方向に 4.8% という結果になった。これは、本手法により、物理付箋の置かれた位置に電子付箋を配置することが可能であることが示された。また、基準点に近い物理付箋はほど正しい位置を推定することができた。

評価実験より、 $horizontal$ 方向、 $vertical$ 方向ともにズレが生じることが確認された。これは、基準点や付箋の位置を選択する際に、ユーザの指定した位置に誤差があることが原因であると考えられる。基準点をユーザに指定してもらう際に、指定位置をタッチすることによって座標点の指定を行っている。タッチの際に $\pm 0.5\text{cm}$ 程度の誤差が生じていることが推測される。水平方向の指定、垂直方向の指定、それぞれも同様にタッチした座標点で指定しているため、同様に誤差が生じることが推測される。これにより、最大で 1.0cm 程度の誤差が生じる。また、付箋の位置の推定の際、撮影画面上の物理付箋の左上をユーザにタッチしてもらう仕様となっている。その時にも、タッチによる誤差が生じている可能性がある。改善策として、画像処理によるイメージ画像からの付箋領域の位置情報の推定があげられる。現在、イメージ画像から付箋の領域を抽出する処理はサーバで行っているが、これを付箋の位置情報が決ま

表 1 物理付箋を電子化した際の位置ズレの平均値と分散

Table 1 Average value and variance of positional shift when digitizing physical sticky notes

	<i>horizontal</i>	<i>vertical</i>
<i>average</i> [cm]	0.9(1.4%)	2.0(4.8%)
<i>variance</i> [cm]	0.9	2.7

際に利用することで、タッチによる誤差の問題は解決される。

もう1つ誤差が生じる原因として、現実空間を想定した三次元仮想空間の精度の問題がある。例えば、実験を行う中で、基準点の位置ずれが生じた。基準点や付箋の位置は、現実空間を想定した仮想空間上で把握されている。現実空間と仮想空間と位置合わせは、スマートフォン、タブレットのセンサー情報に依存している。そのため、端末の急激な移動や向けの変更、カメラを付箋に近づけすぎたことによる周囲の位置情報の欠落により、座標系のズレが生じてしまう。付箋の抽出を行う際は、物理付箋にカメラを近づけるため、座標系のズレが生じやすい。そのため、長時間に渡り連続して付箋抽出を続けると、位置ずれの問題が発生しやすくなる。クライアント上で想定している三次元空間が、現実空間と生じている誤差がある場合、紙付箋認識の際に位置情報のズレとして現れることは十分にあり得る。

horizontal 方向に比べ、*vertical* 方向の方が位置合わせの際の誤差が大きい結果となった。これは、*vertical* 方向の縮尺が最適ではないということを示している。原因として、スクリーンを投影しているプロジェクターにあると考えられる。プロジェクターが垂直な平面に対して投影しているため、台形補正を加えて画面を投影している。そのため、垂直方向の長さが、スクリーンの画面設定と異なっている可能背が高い。プロジェクターの他に、平面に対して水平なディスプレイで実験できる環境がなく、また、壁などの鉛直方向に対する検出は行えなかったため、確認することが出来なかった。今後の課題として、鉛直方向に対する平面検出が可能となった時に、液晶ディスプレイなどに投影したスクリーンで同様に位置合わせを行った場合の結果と比較したい。

6. おわりに

本稿では、ARKit による現実空間と仮想空間の位置合わせを行うことで、実世界型電子付箋システムにおける付箋抽出の際の位置合わせの問題の解決を行なった。平面検出を行い、現実空間にもとづく座標系を生成し、電子付箋が投影されているスクリーンの位置と物理付箋の位置を取得し、検出を行った。取得した付箋情報とスクリーンの座標系のキャリブレーションを行い、物理付箋の置かれた位置に電子付箋を自動で配置することに成功した。位置合わせの評価を行ったところ、水平方向に対して平均 0.9cm 、

垂直方向に平均 2.0cm ほどの誤差が生じる結果となった。よって、本手法により、物理付箋の置かれた位置に電子付箋を配置することが可能であることが示された。また特筆すべき点として、タッチパネル等のセンサーを含む特殊なディスプレイ上に投影された画面に対応した位置に付箋を置くのではなく、水平な机の上にプロジェクターで投影した画面の枠内に存在する物理付箋に対して、物理付箋の置かれた位置に変換された電子付箋を配置した。今後の課題として、鉛直方向に対する平面検出が可能となった時に、液晶ディスプレイなどに投影したスクリーンで同様に位置合わせを行った場合の結果と比較したい。

謝辞 本研究の一部は JSPS 科研費 JP15K00422, JP16K00420 の助成を受けたものです。

参考文献

- [1] 大園忠親, 丹羽佑輔, 藤江翔太郎, 渡邊正人, 鈴木智也, 伊藤栄俊, 岩田 知, 新谷虎松: Mahoca: 議論事例の再利用を可能にする実世界型電子付箋システム, 平成 27 年度電気情報通信学会 人工知能と知識処理研究会, Vol.116, No.350, pp.33-38 (2016).
- [2] 鈴木智也, 大園忠親, 新谷虎松: 実世界型電子付箋システムにおける連想支援のための関連情報提示機構について, インタラクシオン 2017, pp.388-392, (2017).
- [3] 伊藤栄俊, 鈴木智也, 丹羽佑輔, 大園忠親, 新谷虎松: ブレインストーミング事例の再利用のための関連情報推薦機構の試作, 2017 年度人工知能学会全国大会 (第 31 回) 講演論文集, Vol.31, pp.1-4, (2017).