

## RGB-D カメラを用いた KJ 法支援ソフトウェアの開発

熊谷 隼人<sup>†</sup> 田村 仁<sup>†</sup>日本工業大学工学部創造システム工学科<sup>†</sup>

## 1. はじめに

会社での会議やミーティングなどでグループウェアを行うことが一般的である。しかし、パソコンを用いたグループウェアはプロジェクトリーダーが画面を操作するため、他のメンバーを含めた複数人で同時に画面を操作することはまだ不可能であり、不便だと考える。そこで、グループウェアで用いられる KJ 法に着目し、複数人で同時に画面操作が可能なシステムを提案する。実現には、Kinect<sup>1)</sup> の RGB カメラと赤外線深度センサを組み合わせる。顔と指の 3 次元座標から画面内を指差ししてポインティングし、各種操作が可能なシステムを開発し、非接触での操作方法を検討する。

昨年の研究<sup>2)</sup>において、図 1 に示したようなシステムにて、複数人での操作には成功しているが、実装されている機能はポインティングのみであり、システムとして実用段階には至っていない。

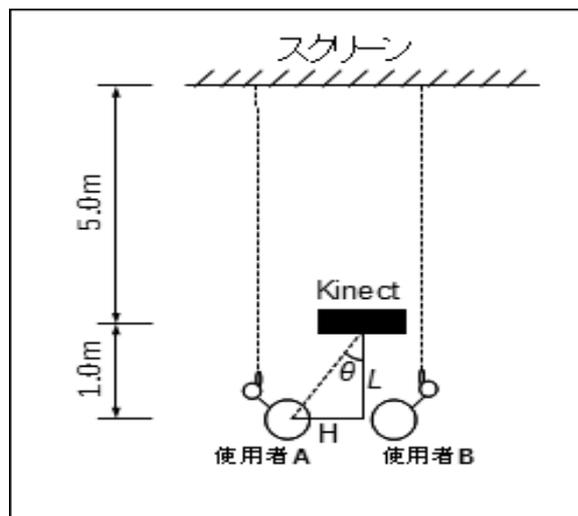


図 1 大型スクリーンの操作インターフェースのイメージ図

## 2. 研究概要

今回提案するものは、プロジェクタスクリーンを指差しすることで、画面上のカードを操作するインターフェースである。

このシステムは、複数のユーザが同時に指で指し示すことでスクリーン上をポインティングし、表 1 に示したジェスチャを用いて、画面に表示されているカードを任意に操作することを可能としている。赤外線深度センサで取得した距離情報より、赤外線カメラから見てユーザごとに最も距離が近い点を指先座標とし、さらに顔の座標を検出し、顔座標と指先座標の延長線上とスクリーンの交点を計算する。

また、グルーピングを行う上で、ラベル名やグループ名などの文字入力を必要とするため、書記を 1 名用意し、キーボードにて文字入力を行ってもらうこととする。

しかし、2) は Linux 環境でのみ動作するプログラムであり、また、既に提供が終了した OpenNI を使用しているため、これを Windows 環境のみで動作する Kinect SDK<sup>3)</sup> に変更する。

システムを使用する環境として、スクリーンのサイズは 4:3 の 300 インチ、Kinect の高さは地面から 1.8m、スクリーンから Kinect までの距離は 5.0m、Kinect から使用者までの距離を約 1.0m と想定する。

表 1 ジェスチャー一覧

操作	ジェスチャー動作
ドラッグ	2.0 秒間、対象のカードをポインティングする
ドロップ	2.0 秒間、ドラッグしたカードを指定範囲内にとどめる

## 3. Kinect について

Kinect とは、複数のセンサとプロセッサ、USB2.0 インタフェースによって構成されるデバイスである。本研究で使用する Kinect のセンサは 8bitVGARGB カメラ、クラス 1 赤外線レーザ式パターン投光器と赤外線カメラで構成される深度センサの 2 つである。

Development of the KJ Method Support Software Using a RGB-D Camera

<sup>†</sup> Hayato Kumagai, Hitoshi Tamura · Innovative Systems Engineering Nippon Institute of Technology

#### 4. 実験方法

使用機材は、Windows8(64bit)、Microsoft Kinect(Xbox360用)、プロジェクタを用いる。また、使用言語はC++であり、画像処理ライブラリはOpenCV2を用いる。

作成したプログラムを用いて、スクリーン上をポインティングし、任意のカードを指定した位置までドラッグ&ドロップした場合の精度を計る実験を行う。図2のように、Kinect正面からポインティングした場合(プレイヤーA)、右に1.0m移動してポインティングした場合(プレイヤーB)、左に1.0m移動してポインティングした場合(プレイヤーC)の3通りの条件で、図3に示した縦40pixel、横113pixelのカードAをポインティングし、ドラッグ判定が行われた後、カードAを指定範囲まで移動させ、ドロップ判定が行われるまでの時間を計測する。指定範囲は、図3に示した擬似画面の左上座標を(0, 0)、右下座標を(640, 480)としたとき、座標(300, 50)から座標(413, 90)の矩形領域とする。

実験で使用する環境は、スクリーンのサイズを4:3の300インチ、Kinectの高さを地面から1.8m、スクリーンからKinectまでの距離を1.67mとする。

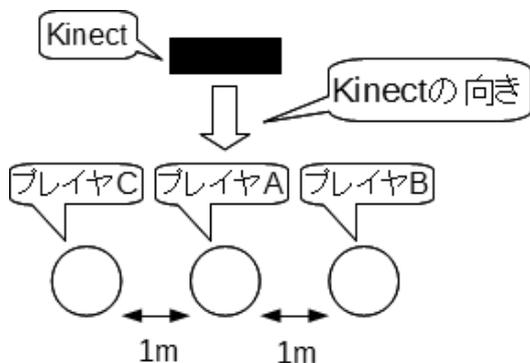


図2 Kinectから見た各プレイヤーの配置

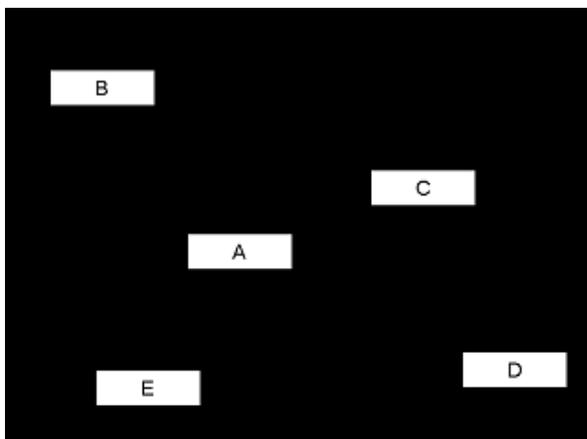


図3 実験で使った擬似画面

#### 5. 実験結果

実験結果を以下の表2に示す。

今回の実験で、プレイヤーAの位置からのポインティングは正確であった。しかし、プレイヤーBの場合は左寄りに、プレイヤーCの場合は右寄りにカーソルが指した位置に対してズレが生じた。

また、ズレが発生する方がポインティングに時間を要することが判明した。

複数人での操作については、現状2人での同時操作に成功している。

表2 実験結果

	平均所要時間[s]	ズレの有無	ズレの方向
プレイヤーA	12.46	無	無
プレイヤーB	13.28	有	左
プレイヤーC	13.17	有	右

#### 6. 考察

Kinect正面以外の位置からのポインティングの結果、画面上の指差点とのズレが生じる原因として考えられるのは、深度センサで取得した顔座標が想定していた顔座標と異なっていることだと考えられる。深度センサの基点が固定されているために、正面以外の顔を検出できず、処理内での顔の中心座標と実際の顔の中心座標が異なっているからだと考えられる。

#### 7. まとめ

本研究では、大型スクリーンに投影されたカードを複数人で同時操作を可能とする処理手法を考案する。実験の結果、Kinect正面から左右にずれた位置からのポインティングで精度が下がることが判明した。今後は、深度センサ範囲内のどこからでも高精度なポインティングを可能とする処理手法を考案するとともに、ジェスチャによる操作機能を実装することを目指す。

#### 参考文献

- 1) 中村薫, KINECT センサープログラミング, (株)秀和システム, 2011
- 2) 萩島弦, "複数人でプロジェクタ・スクリーンの遠隔操作を行うポインティングシステムの開発", 情報処理学会第76回全国大会講演論文集(分冊4)(情報処理学会) pp. 143-144, 2014. 03
- 3) 中村薫, KINECT for Windows SDK プログラミング C++編, (株)秀和システム, 2012