

J-07 Kinect を使ったペイントツールの製作

伊東 秀輔

指導教員 石舘 勝好

1. はじめに

私は、Java を通じて学んだオブジェクト指向について興味を持ち、就職先での使用が予定されている C# を使って深く学びたいと考えた。また、NUI (Natural User Interface) を実現するデバイスとして話題となっている Kinect を活用して、面白いアプリケーションを作りたいと考え、このテーマに取り組むことにした。

2. 開発環境

- ・ハードウェア：Kinect, PC, プロジェクタ
- ・ライブラリ：OpenNI, NITE
- ・OS：Windows 7
- ・開発言語：Microsoft Visual Studio 2010 C#

3. システムの概要

- ・Windows のペイントツールを参考に、体の動きで絵を描くアプリケーションを作成する(図 1)。
- ・腕の動きで自由曲線を描ける。その他にも、四角形や丸、直線はジェスチャーの認識により簡単に描けるものにする。
- ・ペンやブラシ、色の変更を可能とする。

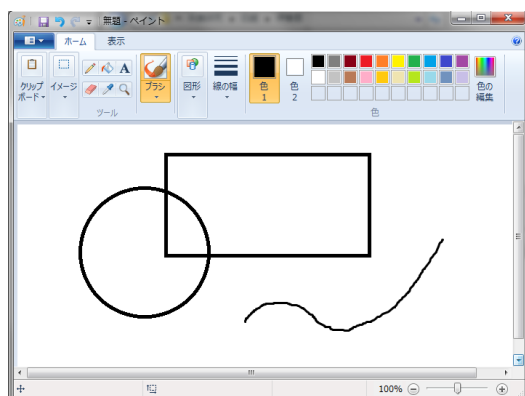


図 1. ペイントツール

4. OpenNI の概要

4.1 OpenNI のライブラリ構成

OpenNI は、Kinect の環境設定、RGB カメラ (画像を取得)、3D センサー (距離を測る)、IR カメラ(3D センサーのための赤外線 (infra-red) 出力)、オーディオデバイス(マイク)といった、Kinect で利用可能なカメラ、センサーを使用するためのインターフェースである。

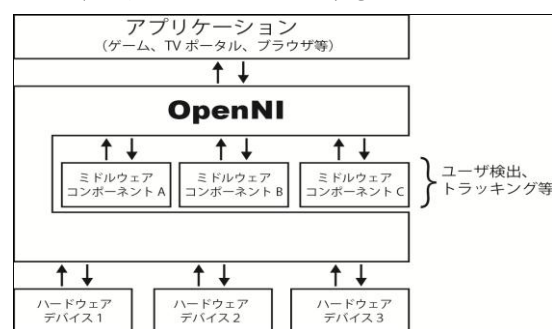


図 2. OpenNI のライブラリ構成

4.2 使用する OpenNI 及び NITE ライブラリ

OpenNI, NITE のクラスのうち本研究で使用するクラスを表 1 に示す。

表 1. OpenNI, NITE の主なクラス

分類	クラス	概要
Context	Context	Kinect初期化
Generator	ImageGenerator	カメラ画像生成
	DepthDenerator	距離検出
	UserGenerator	ユーザー検出
Capability	SkeletonCapabilirty	骨格検出機能
	PoseDetectionCapability	ポーズ検出機能
	MirrorCapability	鏡面機能
NITE	SessionManager	セッションの検出
	~Detector (wave, Push, Swipe)	ジェスチャの検出

5. 設計と作成

5.1 ペイントツールの試作

Windows フォームを使って、基本的なペイント機能を作成し、グラフィック処理、イベント処理の実装方法を学んだ。

5.2 関節の検出

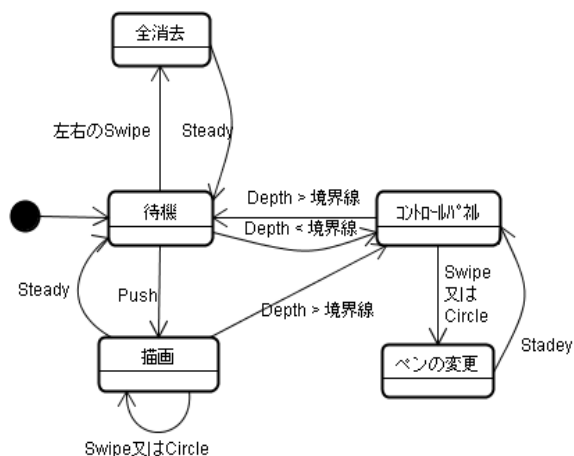
SkeletonCapability クラスを利用すれば、人の関節の座標を取得する事が出来る。

Kinect と OpenNI で、頭、首、右肩、右ひじ、右手首など代表的な関節全20箇所の3次元座標を取得できる。

5.3 関節の3次元座標を使った描画

右手をマウスに見立てて、そのXY座標値を取得して描画用のポインタを動かす処理を作成した。

SessionManager と Detector のイベントを取得し、右手を前に突出、固定することにより描画モードに移行する。描画モード中に動きを止めることにより描画モードを抜ける。発生するイベントと状態遷移を図3に示す。



Push …手を前に出す。
Steady…動作停止。
Swipe…上下左右手を動かす。
Circle…手の円を検出。

図3. 状態遷移図

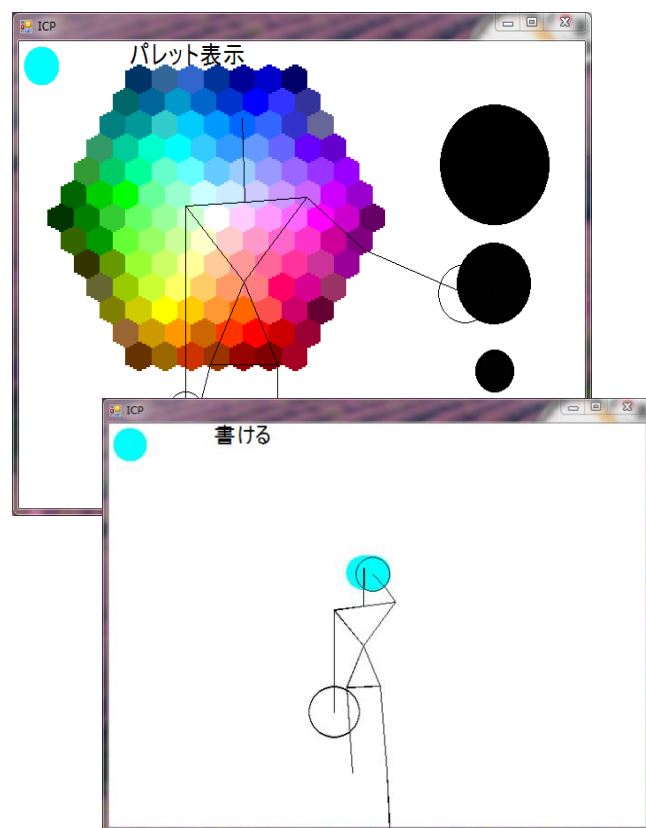


図4. コントロールパネルと描画モード

6. 検証

- ・色の変更やペンの太さを変更できる。
- ・SessionManager を使用して描く事が出来る。
- ・SessionManager の感度が不安定である。
- ・2人以上入った場合, Detector が誤作動する。

7. おわりに

C#を通して、オブジェクト指向に対してより深く理解することが出来た。ジェスチャによる丸や四角形、直線の変更は出来なかったが、色の変更やペンのサイズ変更を作る事が出来た。

参考文献

- (1) 中村 薫：KINECT センサープログラム、秀和システム、2011
- (2) かおるんダイアリー：
<http://d.hatena.ne.jp/kaorun55/>
- (3) OpenNI 公式サイト：<http://openni.org/>