**人物検出RTCマニュアル**

マニュアル目次

[１．本コンポーネントの概要](#概要)

[２．開発環境](#開発環境)

[３．本コンポーネントを使用するまでの手順](#手順)

[(1) Kinect SDKのインストール](#KinectSDK)

[(2) コンポーネントの準備](#準備)

[４．本コンポーネントの使用方法](#使用方法)

[(1) Kinect v2の仕様](#仕様)

[(2) 起動時の注意点](#注意点)

[(3) 出力ポート・出力データについて](#出力)

[(4) コンフィグレーションについて](#こんふ)

[５．問い合わせ先](#問い合わせ先)

１．本コンポーネントの概要

　PeopleDetectionRTCは、Kinect v2を用いて人物を検出するRTCです。RTCの具体的な使用方法などは『４．コンポーネントの使用方法』に記述してあります。

Kinect v2を使用するにはUSB3.0のポートが必要となります。

２．開発環境

　本コンポーネントの開発環境は以下の通りです。

|  |  |
| --- | --- |
| OS | Windows 10 Pro (64bit) |
| コンパイラ | Microsoft Visual Studio Community 2013 |
| RTミドルウェア（C++） | OpenRTM-aist-1.1.2-RELEASE |
| Eclipse | Eclipse SDK-4.4.2 |
| CMake | CMake-3.5.2 |
| Kinect SDK | Kinect for Windows SDK v2.0 |

<次ページへ続く>

３．本コンポーネントを使用するまでの手順

(1) Kinect SDKのインストール

[1] 【<https://developer.microsoft.com/en-us/windows/kinect/develop>】のウェブページにアクセスし、『Get the Kinect for Windows SDK』をクリックします。



[2] 『Continue』をクリックします。



[3] ①については登録してもしなくてもよいので適当な方を選んで、②の『Next』をクリックすると、インストーラのダウンロードが始まります。



[4]ダウンロードしたインストーラを実行し、Kinect SDKをインストールします。

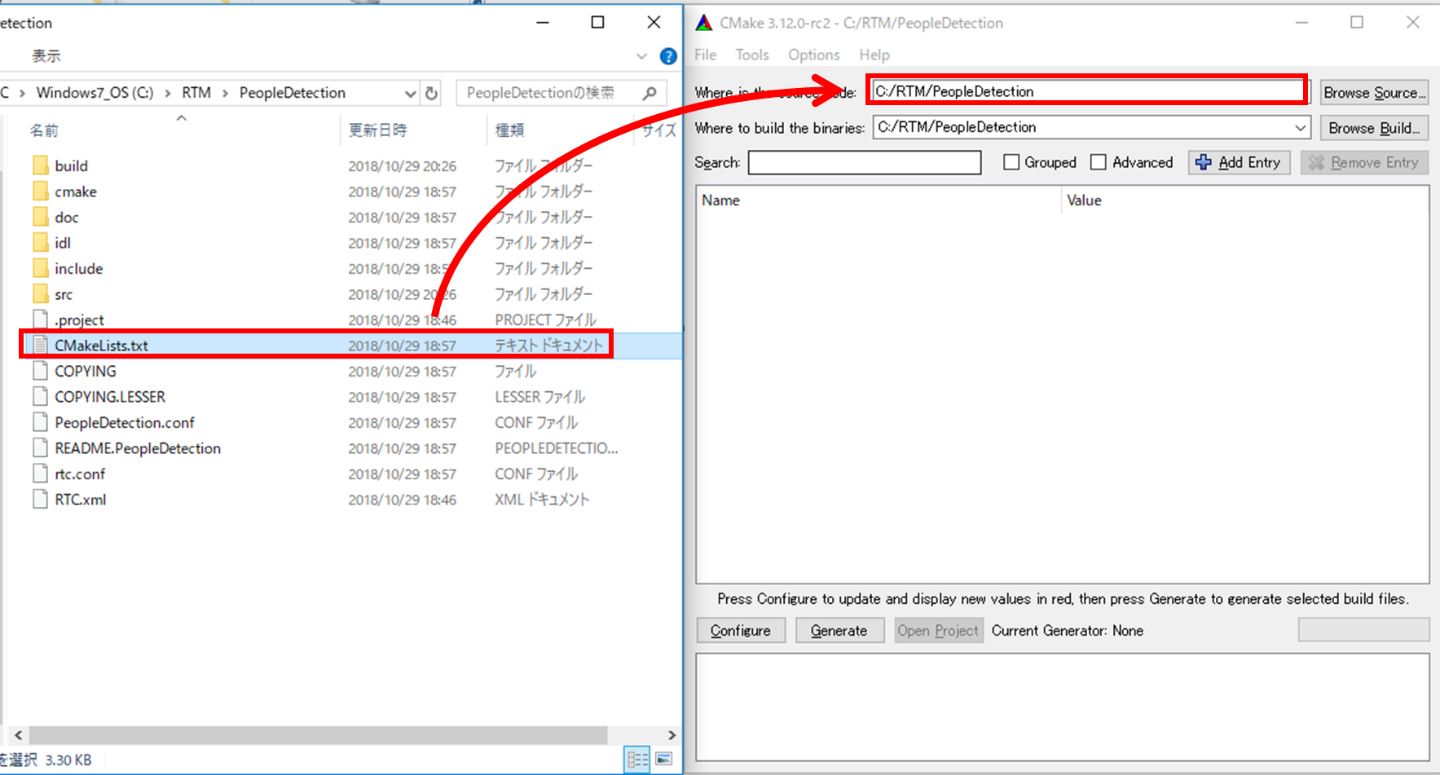
(注意：インストーラを実行する前にKinect v2はパソコンから外しておくこと)

(2) コンポーネントの準備

〇CMakeについて

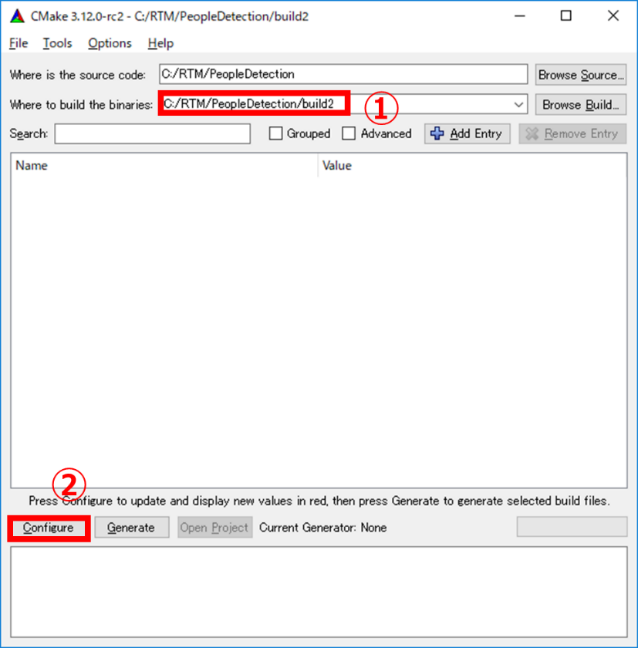
[1] CMakeとダウンロードしたPeopleDetectionRTCのフォルダを開きます。

[2] ファイル内のCMakeLists.txtをCMakeの『Where is the source code』のテキストボックスにドラック＆ドロップします。

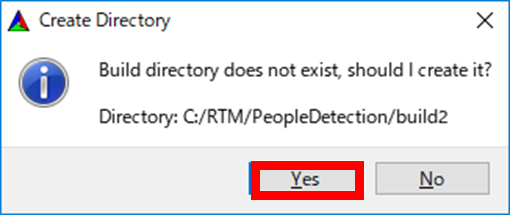


[3] ①『Where to build the binaries』のテキストボックス内の最後に【/build2】を追加します。

②『Configure』をクリックします。

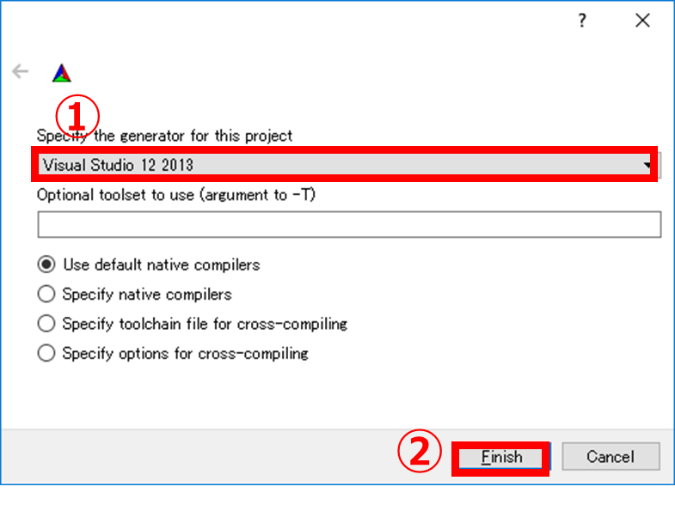


[4] Create Directoryのウィンドウが出たら、『Yes』をクリックします。

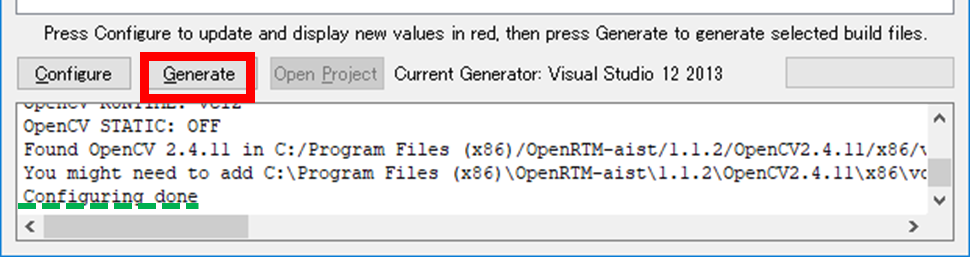


[5] ①『Specify the generator for this project』のうち、自分が使用しているVisual Studioのバージョンを選択します。

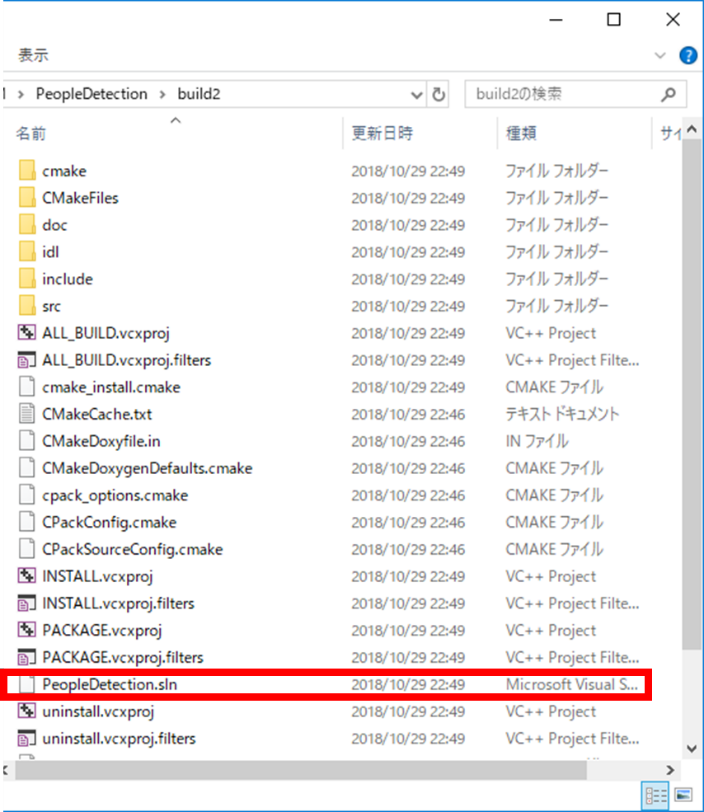
②『Finish』をクリックします。



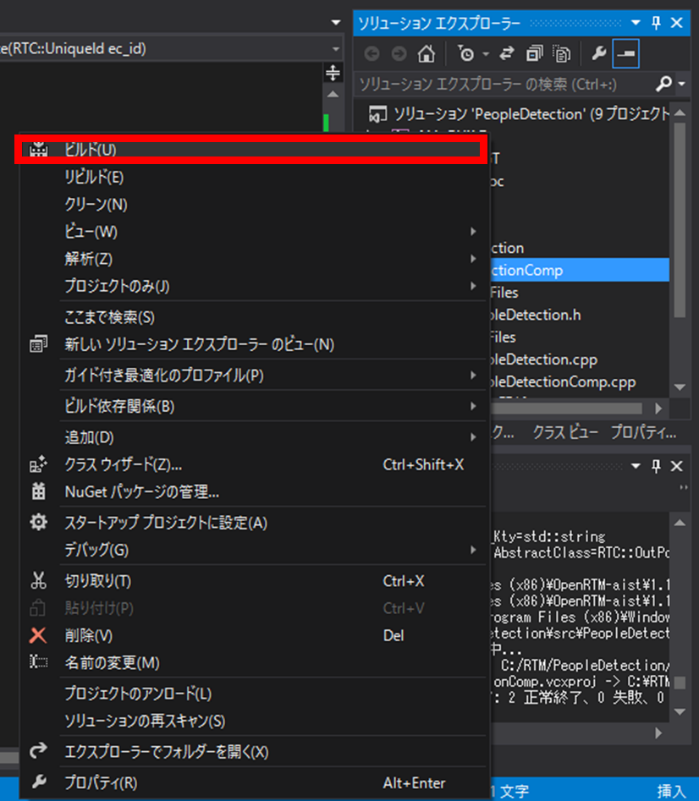
[6] 『Configure』の下のテキストボックスに、【Configuring done】と出てきたら、『Generate』をクリックします。



[7] 先ほどのテキストボックスに【Generating done】とでてきたらCMakeを閉じて、CMakeLists.txtと同じ階層にあるbuild2フォルダ内のSoundDirection.slnを開きます。



[9] ソリューションエクスプローラー内の『PeopleDetectionComp』を右クリックし、『ビルド』をクリックします。



[10] build2フォルダ→srcフォルダ→Releaseフォルダ（Debug）フォルダ内に、PeopleDetectionComp.exeができます。

４．本コンポーネントの使用方法

(1) Kinect v2の仕様

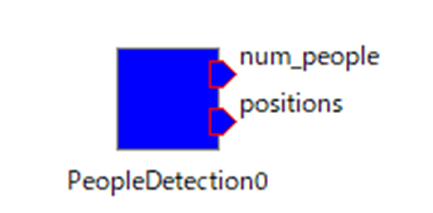
　下記の表は、本コンポーネントに関わるKinect v2の動作仕様になります。

|  |  |
| --- | --- |
| Color画像 | 1920x1080 |
| Colorフレームレート | 30 fps |
| 水平視野角 | 70 deg |
| 垂直視野角 | 60 deg |
| 人の検出 | 0～6人 |
| 人の検出距離範囲 | 0.5～4.5 m |
| 検出骨格数 | 25点 / 1人 |

(2) 起動時の注意点

PeopleDetectionRTCはKinect v2をパソコンに接続した後に起動してください。

(3) 出力ポート・出力データについて



|  |  |  |  |
| --- | --- | --- | --- |
|  | 名前 | データ型 | 概要 |
| 出力ポート | num\_people | TimedLong | 検出した人数を出力 |
| positions | TimedDoubleSeq | 検出した人の指定関節\*1のxyz座標を出力\*2 |

\*1　指定関節については(4)コンフィグレーションにて説明します。

\*2　配列でのこれらのデータの出力方法について

positions.dataの配列の個数は18個となっています。単位はm（メートル）です。

データは

positions.data[0]：検出された１人目の指定関節のx座標

positions.data[1]：検出された１人目の指定関節のy座標

positions.data[2]：検出された１人目の指定関節のz座標

positions.data[3]：検出された２人目の指定関節のx座標

positions.data[4]：検出された２人目の指定関節のy座標

positions.data[5]：検出された２人目の指定関節のz座標

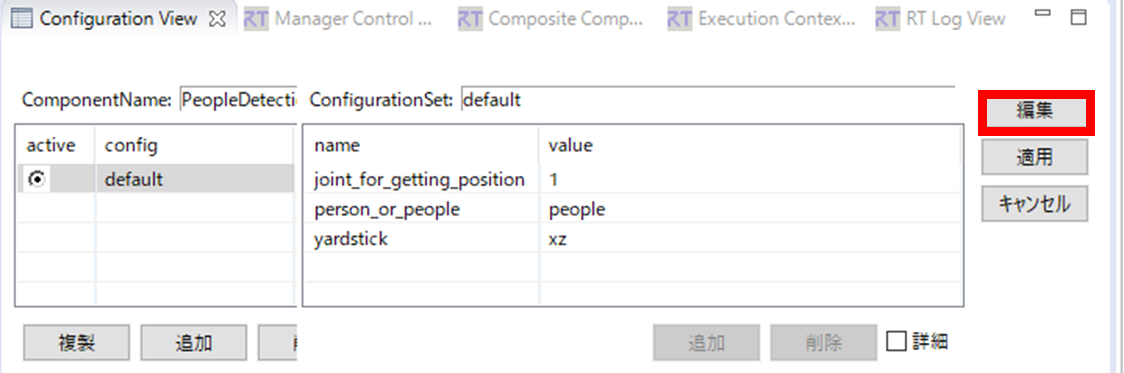
positions.data[6]：検出された３人目の指定関節のx座標

　　　… …

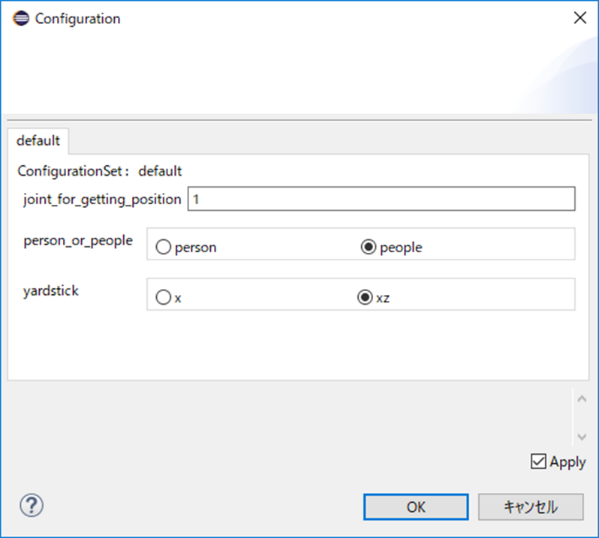
のように入っています。

また、検出人数が6人より少ない場合には、各配列には0が入った状態で出力されます。

(4) コンフィグレーション

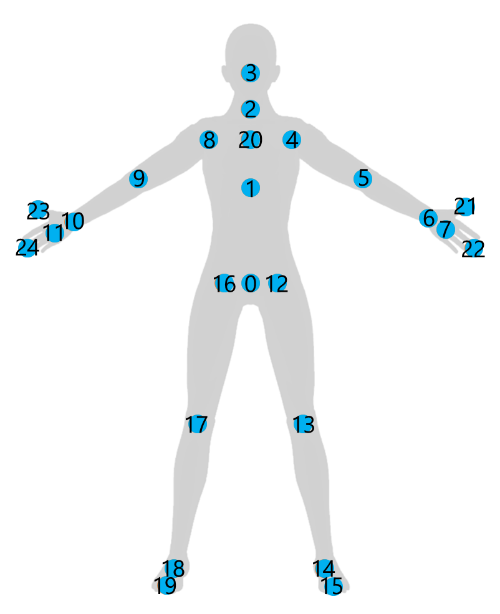


これらのコンフィグレーションは『編集』をクリックすることで変更することができます。『編集』ボタンをクリックすると次のようなウィンドウが表示されます。



これらを変更し『OK』をクリックすることで変更が適用されます。

|  |  |  |
| --- | --- | --- |
| 名前 | データ型 | 概要 |
| joint\_for\_getting  \_position | int | 検出した人物の関節のうち、どこの関節の位置を出力するか\*3 |
| person\_or\_people | String | 1人のみ検出するか複数人検出するか\*4 |
| yardstick | string | 1人のみ検出する場合、Kinectから人までの距離として、z座標のみの距離を使用するか、x座標z座標の距離を使用するか\*4 |

\*3 関節について

関節の番号は右図のようになっています。

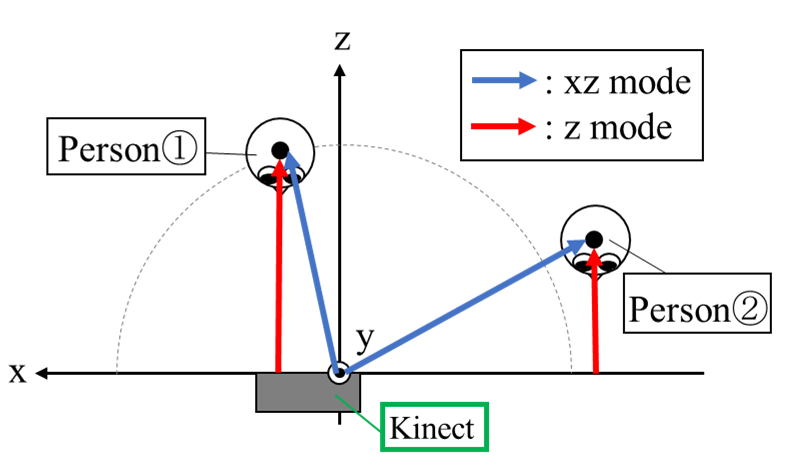
デフォルトでは1の位置の関節位置を取得する

よう設定されています。

\*4 １人検出モードについて

本コンポーネントはKinect v2の視野内にいるすべての人を検出するのではなく、Kinect v2から最も距離の近い位置にいる人のみを検出するモードを使用することができます。

この際、距離の取り方として①zモード②xzモードがあります。それぞれについて下図に示します。



①zモードの場合の距離は図の赤矢印で示したものとなります。この場合、Kinect v2から最も距離が近いのはPerson②となります。

②xzモードの場合の距離は図の青矢印で示したものとなります。この場合、Kinect v2から最も距離が近いのはPerson①となります。

５．問い合わせ先

本コンポーネントについての質問がございましたら、以下のメールアドレスまでご連絡ください。

--------------------------------------------

東京理科大学理工学研究科

機械工学専攻2年

佐古　奈津希

mail:7517624@ed.tus.ac.jp

--------------------------------------------