基礎輪講2週目

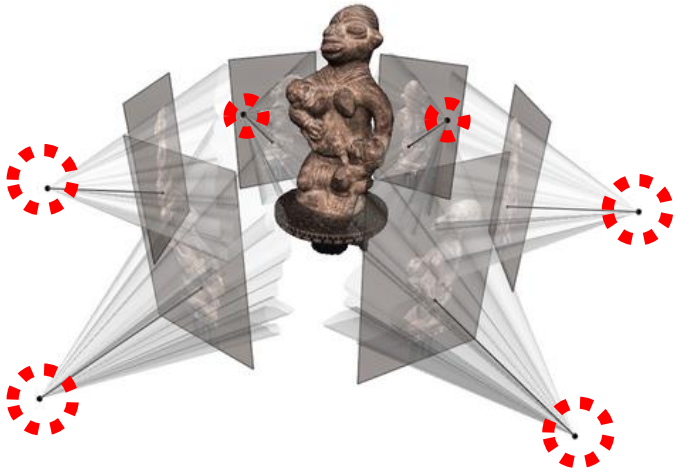
Kinectの話

3D Computer Vision

カメラから3次元の情報を取得

3次元再構築：2次元の画像から3次元形状の復元

多視点画像群†



3次元再構築結果†

3次元の情報を持つ画像ってないの？

距離画像

Depth（距離）の情報を持った画像

→各画素の部分に距離の値が入る.

距離画像

距離画像を取得する機器が必要

距離画像センサ

実世界の距離情報を取得できる.

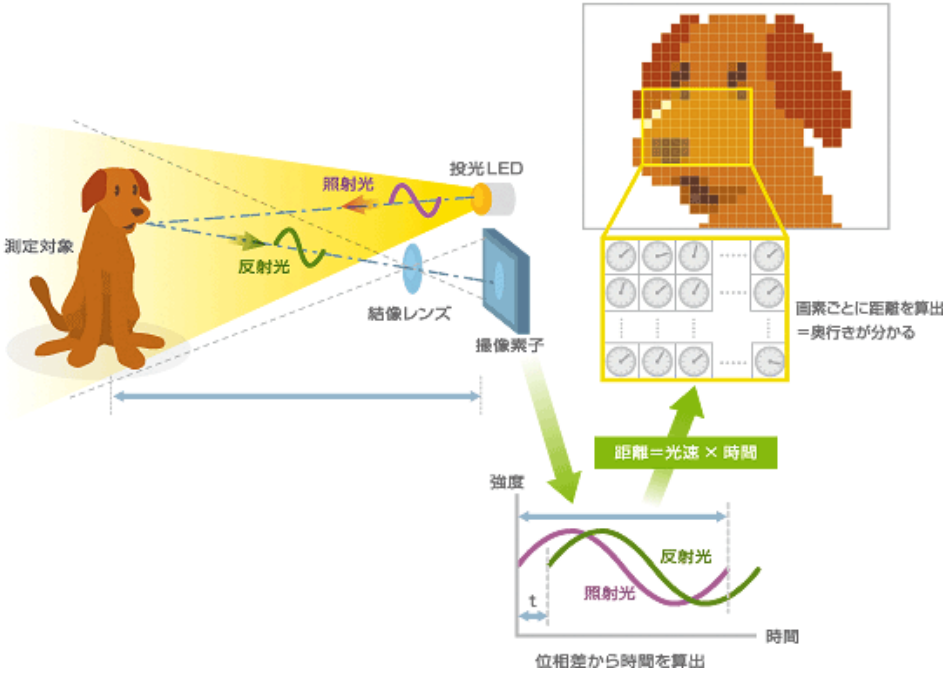
主に2つの方式

①TOF(Time of Flight)方式

②パターン照射(Projector-Camera)方式

TOF(Time of Flight)方式

レーザーが対象まで往復するのにかかる時間(Time of Flight)から距離を計測

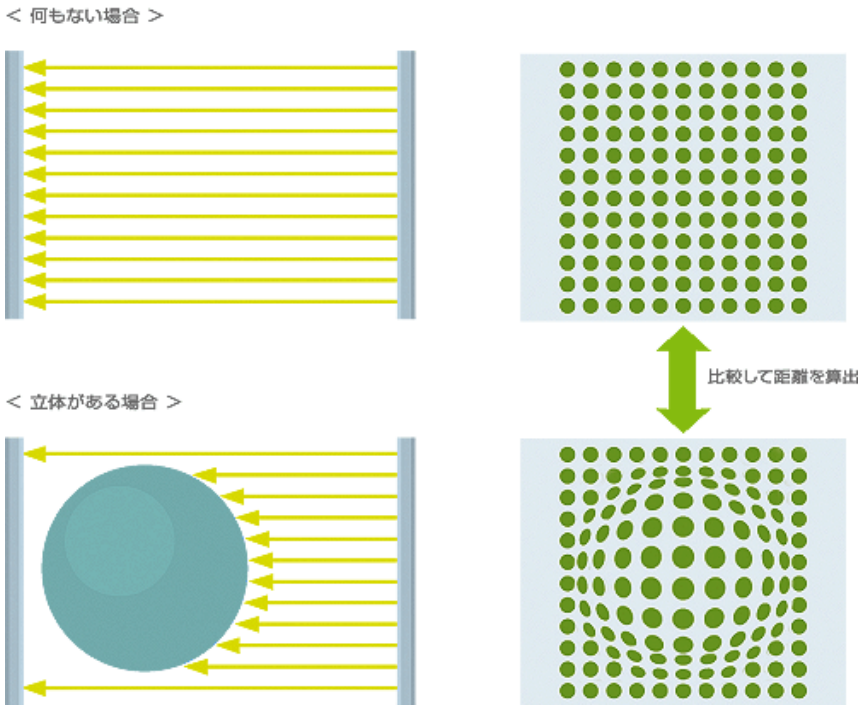


TOF方式による距離計測†

†http://www.tdk.co.jp/techmag/knowledge/201102u/

パターン照射(Projector-Camera)方式

あらゆるパターンのレーザー光線を当て，反射する光のパターンのひずみで距離を測定



パターン照射方式による距離計測 †

†http://www.tdk.co.jp/techmag/knowledge/201102u/

パターン照射

Kinect for Windows v1はパターン照射方式を採用

Kinectの投影光†

†https://www.youtube.com/watch?v=nvvQJxgykcU

距離画像センサの問題点

技術自体は古くからあるが非常に高価なものが多い.

距離画像センサ D-IMager

ご注文品番：EKL3104

販売価格：207,900円 (税込)

納期：受注後2週間～4週間

数量: 1 カートに入れる

お問い合わせ お気に入りに登録

※実際の商品は写真とは異なる場合がございます。

距離画像センサの例

→もう少し安いセンサが欲しい！

コンシューマデプスセンサ

距離情報を取得する低価格なセンサ

→各製品2万～5万程度

→低価格なため広く普及し研究される.

→中でもKinectは数多くの研究に用いられている.

Kinect

もともとは，イスラエルのPrimesense社が製造しているデプスセンサ（Primesensor）

→Microsoft社に技術提供

→MicrosoftがXbox 360のゲームコントローラ(Kinect)として販売

→安価なデプスセンサとして利用可能なため広く普及

（USBでPCと接続可能→ハッキングしやすい！）

Kinectの用途

①距離情報の取得

→3次元点群処理

→Computer Vision的な使い方

②人物姿勢のトラッキング

→NUI(Natural User Interface)等に利用

→Interaction的な使い方

距離情報の取得

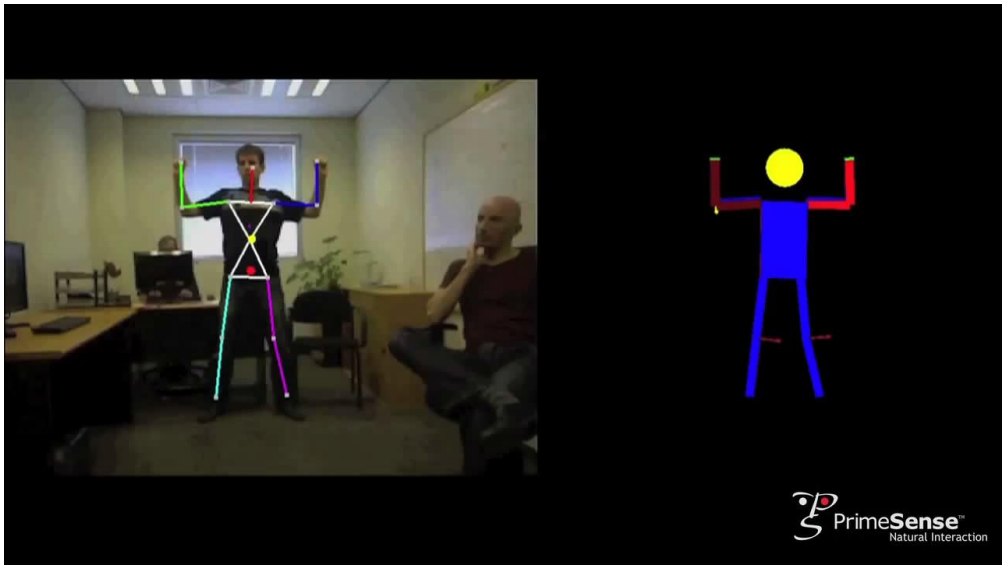
Kinectにより3次元点群を取得

3次元点群の表示

→PCL(3次元点群処理ライブラリ)を使おう！

人物姿勢のトラッキング

体の各部位の推定→スケルトン（ボーン）のトラッキング



スケルトントラッキング

→様々なパターンの人物姿勢を機械学習し人を識別

→開発キットを使おう！

Kinectの種類

研究室には3つのKinectがあります！

 Kinect for XBOX360	・本来はXbox 360用，商用利用はNG ・開発時や個人用途に限りKinect for Windows SDK（Kinect用の開発キット）で利用可能 ・Nearモード使用不可
 Kinect for Windows v1	・Windows PCに対応，商用利用OK ・Nearモード使用可
 Kinect for Windows v2	・Windows 8以降のみに対応 ・Visual Studio 2012以降で動作 ・USB3.0が必須 ・使いこなせる人がまだほとんどいない

Kinectの主な仕様

	Kinect for Windows v2	Kinect for Windows v1
カラーカメラ解像度	1920×1080	640×480
距離カメラ解像度	512×424	320×240
距離センサ有効距離	0.5m～8.0m	0.8m～4.0m (Nearモード：0.4m～, Extendモード：～10.0m)
距離取得方式	TOF	パターン照射
最大fps	30	30
認識人数	6人	6人
認識関節数	25	20
屋外利用	○	×

Kinectの開発キット

①OpenNI：PrimeSense社らが開発しているAPI群

- →OpenNI(API)，NITE（ジェスチャー認識ライブラリ），Sensor（デバイスドライバ）から構成されている.

②Kinect for Windows SDK：Microsoft社が開発したSDK

- →こっちが公式

※OpenNIとSDKは競合します！！

2014年4月23日をもってOpenNIはサポート終了
Windows SDKを使ってください！

Kinectプログラミングのために

PCLとWindows SDKをインストール

1. All-in-one Installerを利用してPCL 1.7.2をインストール
   - (参考URL：http://unanancyowen.com/?p=712)
2. Kinect for Windows SDK v1.8をインストール
   - (http://www.microsoft.com/en-us/download/details.aspx?id=40278)
3. システム環境変数の変数Pathの値にPCL 1.7.2のdllファイルが含まれるフォルダのパスを追加
   (dllファイルは主に¥PCL 1.7.2¥3rdPartyより下のフォルダにあります)
4. (PC再起動)

動作確認

Developer Toolkit Browser v1.8.0(Kinect for Windows)内のサンプルプログラムが動けばOK

適当に選んでRunボタンを押すと起動します.

いろいろ試してみましょう.

サンプルコードと課題

MyKinect.zipをダウンロード→付属のプロパティシートを追加

課題：次の2つの関数を作成して下さい.

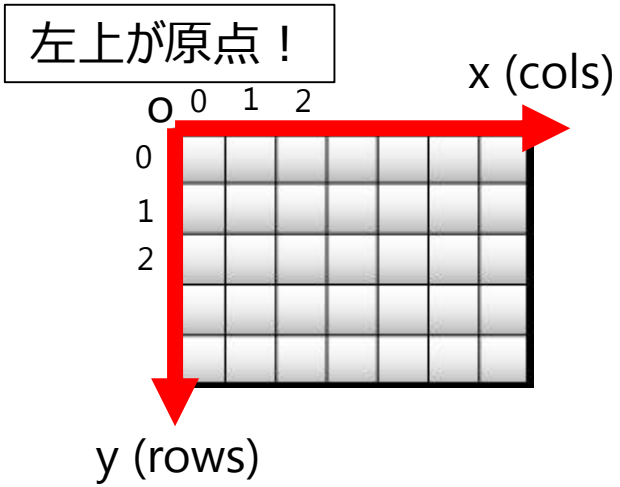
- setDepthImage関数の一部（空白部分）
  - 距離画像(rawDepthImage, depthImage)の作成
  - PCLの点群データ(cloud)に色つきの3次元点を保存
- saveData関数
  - RGB画像・距離画像の出力，Mat型距離データをxml，点群をpcdで出力
  - ファイル名固定は避ける.
  - 出力したpcdファイルはPCL 1.7.2¥bin¥pcl_viewer_release.exeで開けます.

- 終わった人：点群の平面領域を検出（PCLに関数があります）
  - 平面の色を変える，平面を除去する，etc…　　自分で調べてみましょう！

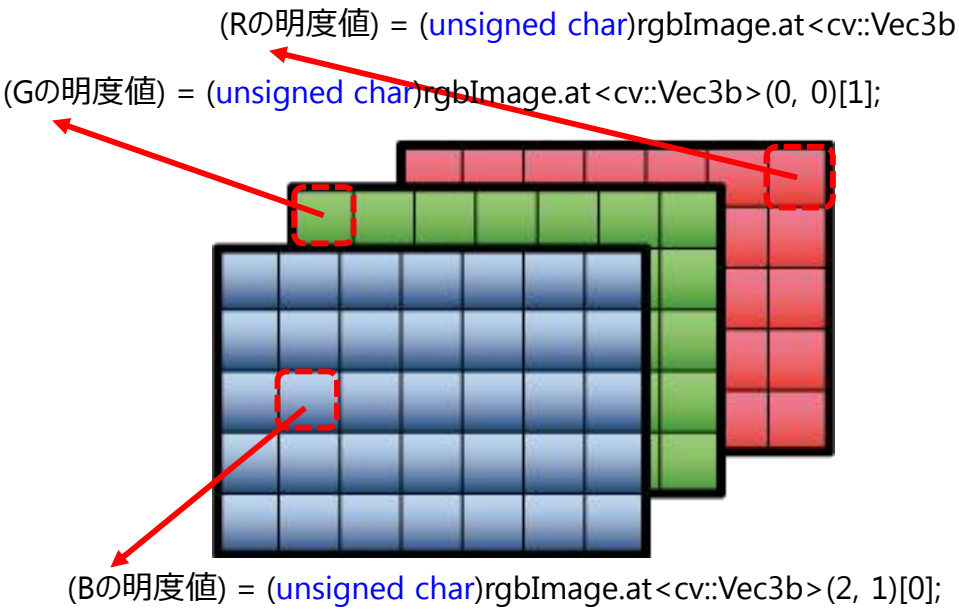
Kinectのカラー画像

NUI_LOCKED_RECT colorData (KinectからのRGB情報)

→cv::Mat_<cv::Vec3b>　rgbImage　に登録



Mat型の概要

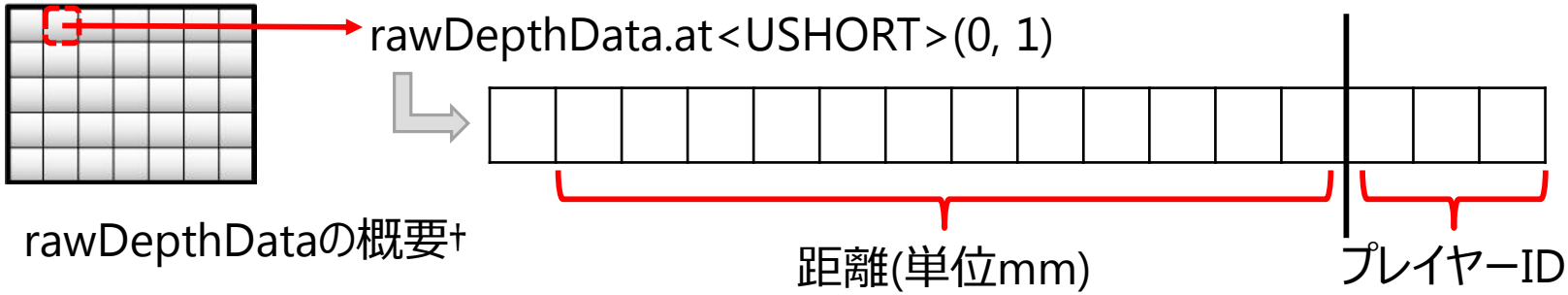


rgbImageの概要

Kinectの距離データ

NUI_LOCKED_RECT depthData (Kinectからの距離情報)

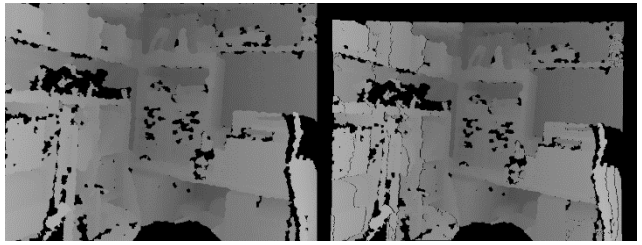
→cv::Mat_<USHORT>　rawDepthData　に登録



※カラー画像と距離画像の座標は異なる→単純に重ねてはダメ！

→NuiImageGetColorPixelCoordinatesFromDepthPixelAtResolution関数

カラー画像



距離画像（左：距離画像座標　右：カラー画像座標）

†http://opencv.jp/cookbook/opencv_mat.html#id1

Kinectの座標系変換

rawDepthDataの値から3次元（カラーカメラ座標系）に変換

```
rawDepthData(距離画像座標系)
col         = (画像のx座標);
row         = (画像のy座標);
rawDepthData.at<USHORT>(row, col) = (距離の値[mm] + プレイヤーID);
```

NuiTransformDepthImageToSkeleton関数

```
real (カラーカメラ座標系)　（realはVector4型　real.wは無視）
real.x = (x座標[m]);
real.y = (y座標[m]);
real.z = (z座標[m]);
```

関数が想定する座標系とカメラ座標系が異なる.
正方向に注意してPointCloudに格納