



# SUBARU 画像認識チャレンジ 画像説明資料

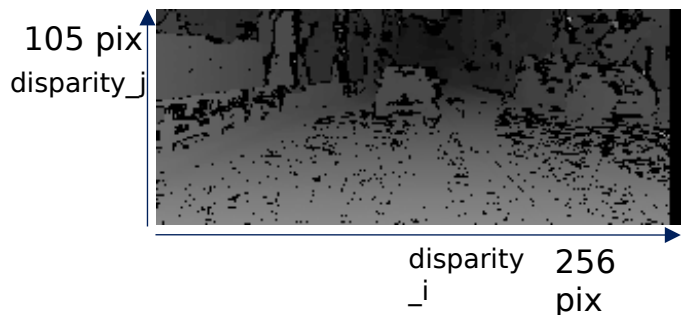
SUBARU CORPORATION

SUBARU Lab

Nov 19, 2021

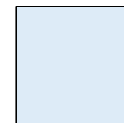
# 各画像説明

視差画像

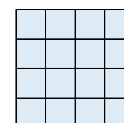


※見やすさのため明る  
さ調整しています。

※視差情報は  
250pix(1000pix/4)ま  
で。残り6pixは0を格  
納。



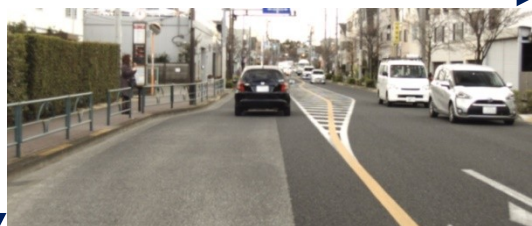
4x4pix毎に  
1視差を格納



右画像

right\_i 1000pix

right\_j  
420pix



左画像

left\_i 1128pix

left\_j  
420pix



## 注意

- ・ 視差画像は左下  
右/左画像は左上が座標原点になっています。
- ・ disparity\_PNGフォルダの画像はモニタ用で  
整数視差のみのものです。disparityフォルダの  
rawファイルは小数視差まで含まれています。

# 視差画像について

右画像



左画像



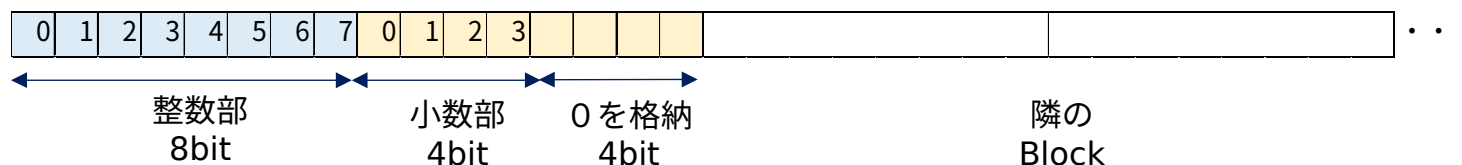
右と左のブロックがマッチングした座標を視差として格納しています。

※距離情報にするためには以下の変換が必要です

$$\text{距離(m)} = 560 / (\text{視差} - \text{inf\_DP})$$

- inf\_DPはカメラの調整パラメータのため、Frame毎のSequenceデータに含まれます。
- 視差 = inf\_DPの場合、距離は $\infty$ という意味になります。
- 視差 = 0の場合、該当ピクセル/ブロックには距離情報がないという意味になります。

視差フォーマット disparityフォルダのrawファイル  
2byte(12bit)



# 視差⇒距離のサンプルコード

右画像（左上原点）の(i,j)座標の距離（単位m）を求める処理

- ・視差情報は、disparityフォルダの●●.rawを使用

```
-----
disparity_image_width # 視差画像横サイズ 256
right_image_height    # 右画像縦サイズ 420
inf_DP                # 補正パラメータ Frame毎のSequenceデータから読み込み
```

```
right_i                # 右画像の横(i)座標
right_j                # 右画像の縦(j)座標
disparity_i            # 視差画像の横(i)座標
disparity_j            # 視差画像の縦(j)座標
disparity              # 視差情報
distance               # 距離情報
-----
```

```
# rawファイルを開く
with open(rawfile_path, 'rb') as f:
    disparity_image = f.read()
```

```
# 右画像座標位置に対応する視差画像座標を求める
```

```
disparity_j = int((right_image_height - right_j - 1) / 4) # 縦座標
# 視差画像と右画像は原点が左下と左上で違うため上下反転
```

```
disparity_i = int(right_i / 4) # 横座標
```

```
# 視差を読み込む
```

```
disparity = disparity_image[(disparity_j * disparity_image_width + disparity_i) * 2] # 整数視差読み込み
disparity += disparity_image[(disparity_j * disparity_image_width + disparity_i) * 2 + 1] / 256 # 小数視差読み込み
```

```
# 視差を距離へ変換
```

```
if disparity > 0: # disparity = 0 は距離情報がない
    distance = 560 / (disparity - inf_DP)
```