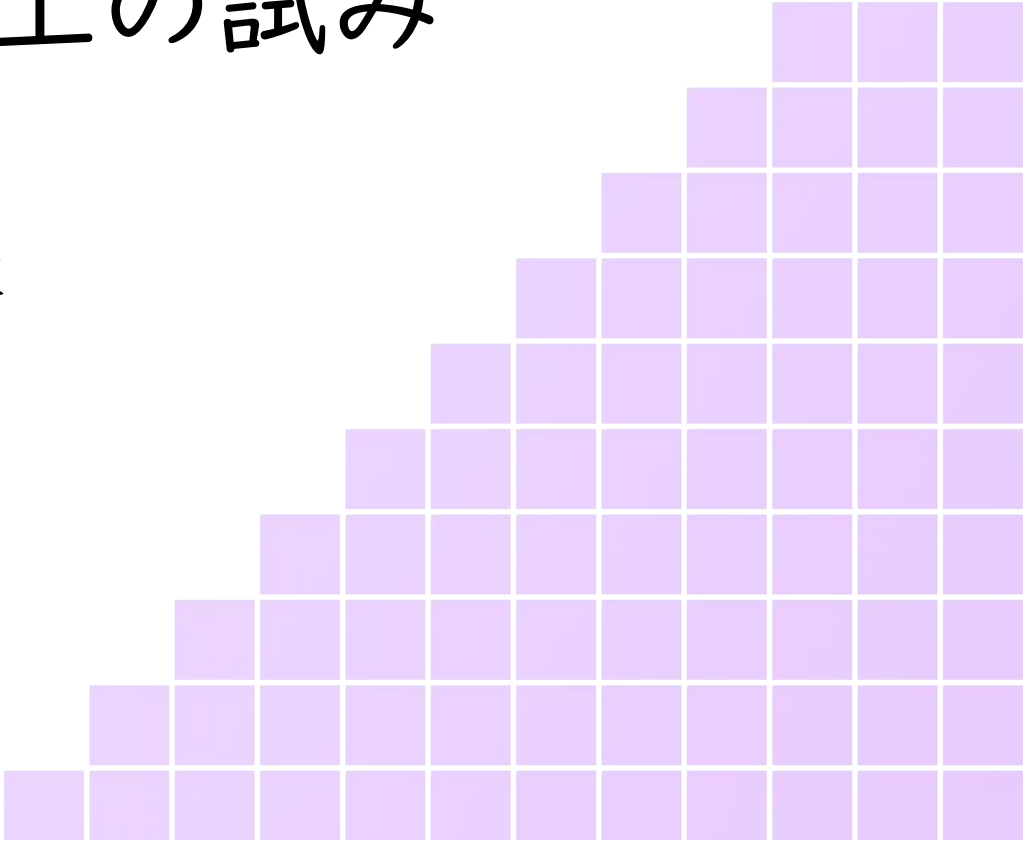
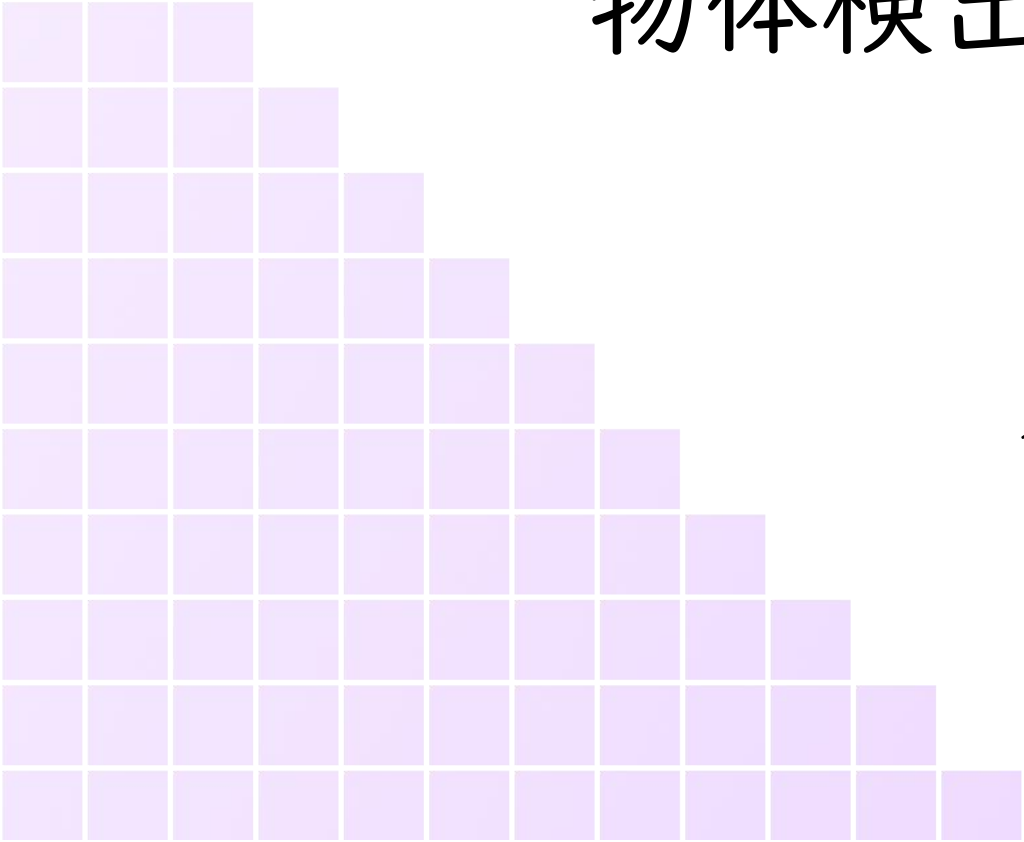


RGB-D画像と画像修復を活用した オクルージョン時における 物体検出の精度向上の試み

5419030 石川浩太

令和4年度 卒業演習



目次

1 はじめに

1.1 物体検出

1.2 現状の物体検出

1.3 先行研究

1.4 3次元点群

1.5 問題解決への手法提案

1.6 演習概要

2 準備

2.1 RGB-D画像

2.2 Azure Kinect

2.3 Open3D

2.4 YOLOv5

2.5 LaMa

3 実験方法

3.1 画像の取得

3.2 点群化

3.3 ③視点変更と画像保存

3.4 ④画像修復

3.5 ⑤物体検出

4 実験結果

5 考察

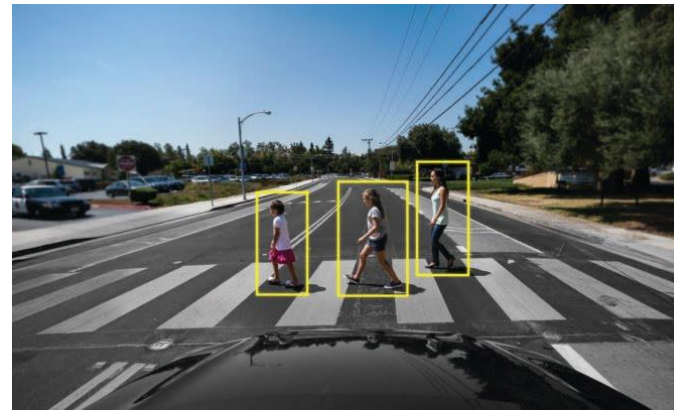
6 今後の課題

7 参考文献

ある共通する技術が使われている



Googleレンズ



画像引用:
自律走行車はどのように周囲を見ているのか
<https://blogs.nvidia.co.jp/2019/05/15/how-does-a-self-driving-car-see/>

自動運転

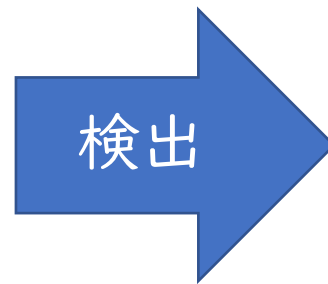


顔認識

1.1 物体検出

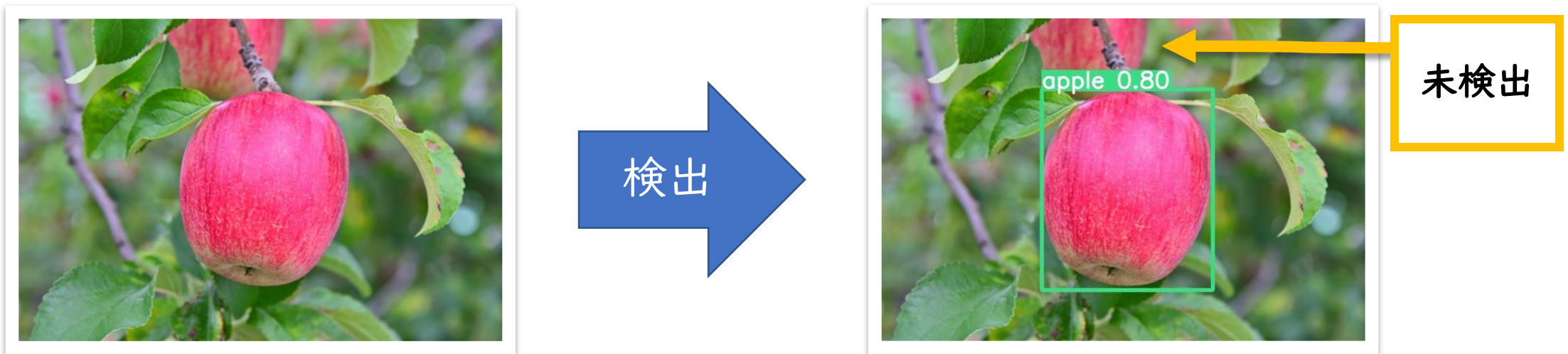
画像や動画に映っている物体の位置や種類, 個数などを
検出する技術

例) 画像から人間と犬を検出



1.2 現状の物体検出

物体が遮られているオクルージョンという状態で
未検出や検出領域の結合, 誤検出などが起こってしまい
検出精度が下がってしまうことが課題の一つ

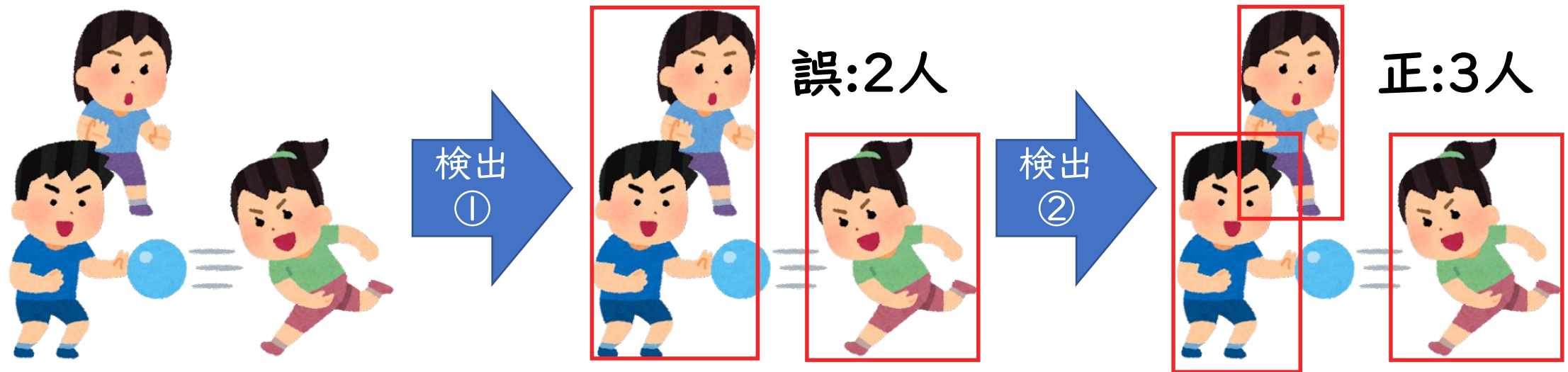


枝や画像サイズによって一部分しか見えないリンゴが検出できていない例

1.3 先行研究

オクルージョン状態で検出領域が結合してしまう問題への改善策として**3次元点群**を活用する手法が提案

(高橋正裕 池勇勲 梅田和昇, ステレオカメラによる YOLOと3次元点群を用いた人物検出, 2018)



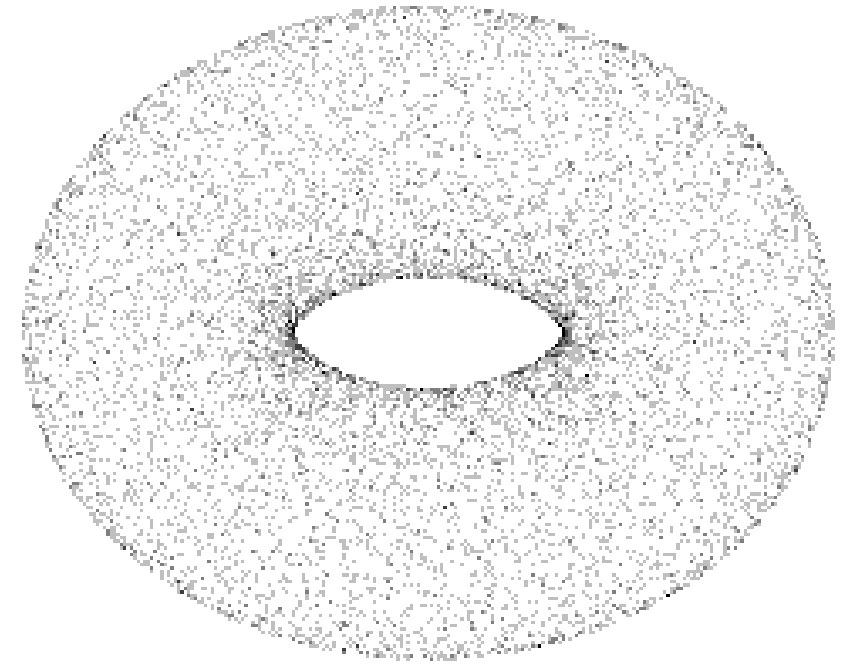
1.4 3次元点群

点の集まりのことを点群といい、3次元点群は点群のうち3次元座標(X, Y, Z)の情報を持ったものを指す

座標情報に加え色情報を含む場合もある

点に大きさを持たせ可視化でき、
様々な視点・大きさでの観察が可能

点情報の無い領域は表示されず欠損となる



1.5 問題解決への手法提案

オクルージョン時の物体検出の精度低下が問題



3次元点群を用いて別の角度から物体を見る



オクルージョンの状態が変化



オクルージョン時の物体検出の精度が向上するのでは？

1.6 演習概要

3次元の情報を持った写真を用意し点群化



点群の撮影時とは異なる視点で表示しキャプチャ



撮影時に死角であるため点の欠損した部分を画像修復技術で補完



加工前の画像と、疑似的に別角度となった
加工後の画像とで物体検出した際の結果を比較

2.1 RGB-D画像

RGB(赤緑青)の色情報を持つ
一般的なカラー画像(RGB画像)に
深度(Depth)情報を加えたもの

付加する深度情報はレーザー光
を利用して深度を測るLiDARと
略される装置や複数個の
カメラの視差を利用して得られる



RGB画像



深度画像

2.2 Azure Kinect

Microsoft社の発売するデバイス

RGBカメラやLiDAR, マイク,
方向センサーなどが搭載され
本演習ではRGB-D映像の取得に使用



RGBカメラのLiDARの視点に差があるため
取得したカラー画像・深度画像間で位置合わせが必要

画像引用:

Azure Kinect 開発者キットを購入する - Microsoft
<https://www.microsoft.com/ja-jp/d/azure-kinect-dk/8pp5vxmd9nhq>

2.3 Open3D



3Dデータ関連のソフトウェア開発を補助するための
オープンソースライブラリ

データ構造とアルゴリズムをC++とPythonの両方で提供

本演習ではRGB-D画像の3次元点群化とキャプチャで使用

画像引用:

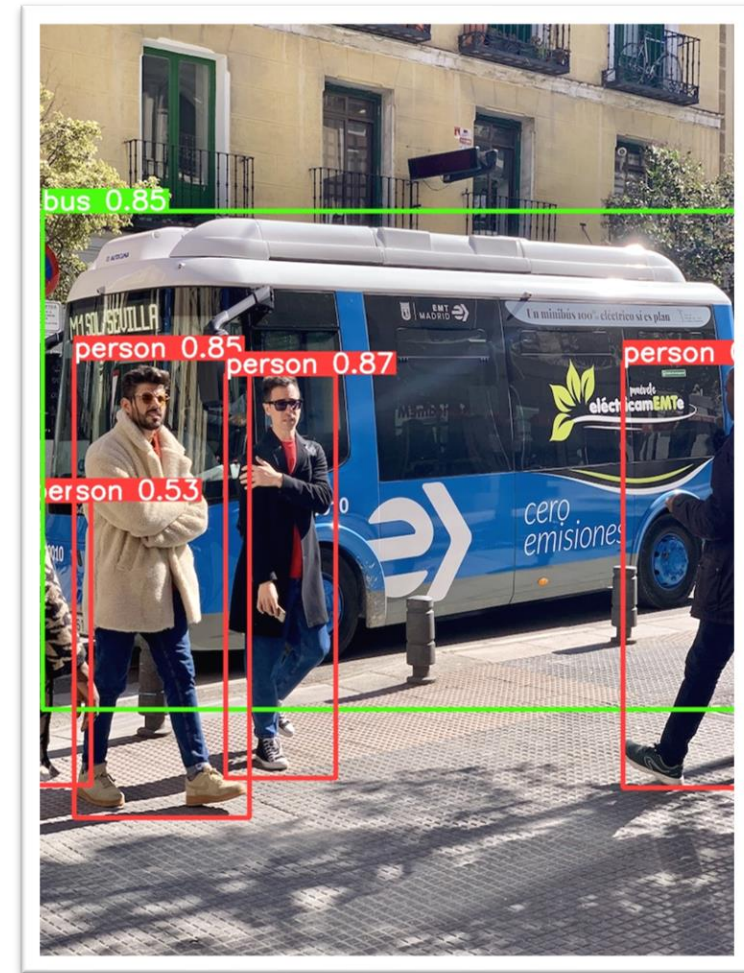
Open3D: A Modern Library for 3D Data
Processing

<http://www.open3d.org/docs/release/>

2.4 YOLOv5

YOLOはYou Only Look Onceを
コンセプトに研究されている
物体検出技術でその系列の一つ

数あるYOLOシリーズの中でも
GitHubでのフォーク数やスター数が
最も多く人気なバージョン



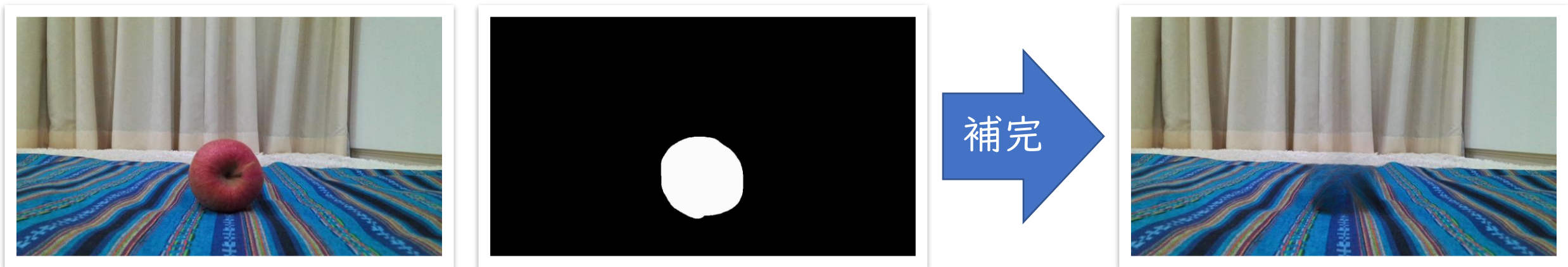
サンプル画像で検出を試みた例

2.5 LaMa

2021年にサムスンの研究機関が発表した画像修復技術の一つ

素材となる画像と欠損箇所を示すマスク画像を基に補完が行える

本演習では派生ソフトの『Lama Cleaner』で使用



3 実験方法

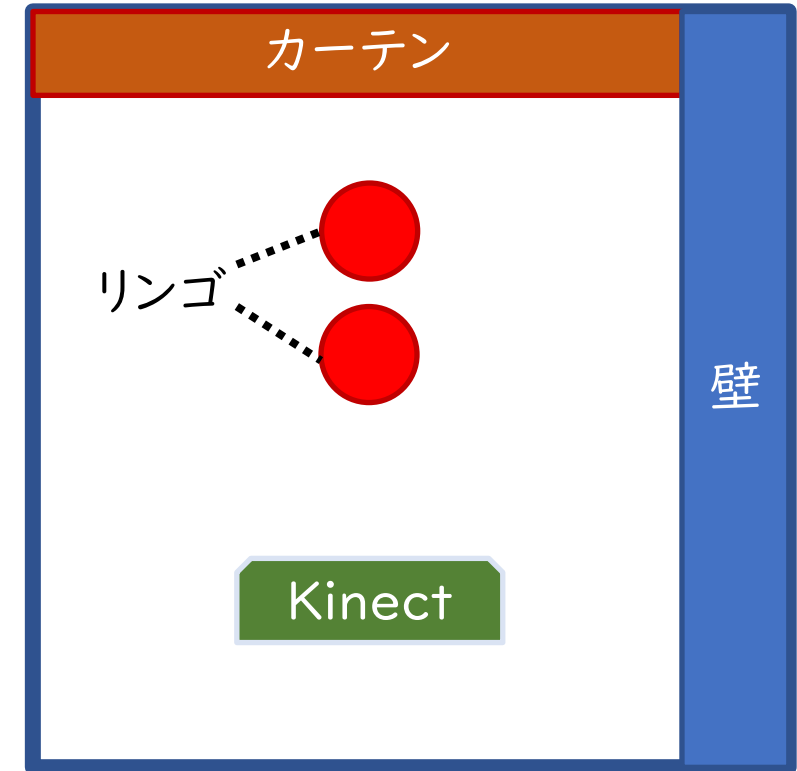
ソフトウェア環境は以下の通り

ソフトウェア	バージョン
Windows 10 Home	21H2
Anaconda3	22.11.1
Python	3.9.15
Open3D	0.16.0
Azure Kinect SDK	v1.4.1
Lama Cleaner	0.34.0
YOLOv5	v7.0

3.1 ①画像の取得

PCにAzure Kinectを接続した状態で
Open3Dのサンプルコード
『azure_kinect_recorder.py』を
実行しMKV形式の動画ファイルを取得

本演習ではリンゴを前後に並べて置き、
Azure Kinectを手に持って動かし
様々な角度から撮影



撮影状況を上から見た図

3.1 ①画像の取得

Open3Dのサンプルコード
『azure_kinect_mkv_reader.py』
でフレームごとに分割しカラー画像と
深度画像のペア432組とカメラの歪み
情報を記録した『intrinsic.json』を取得

画像のペアを15フレーム毎に抽出して
29組58枚にダウンサンプリング



RGB画像(1280x720px 24bit)

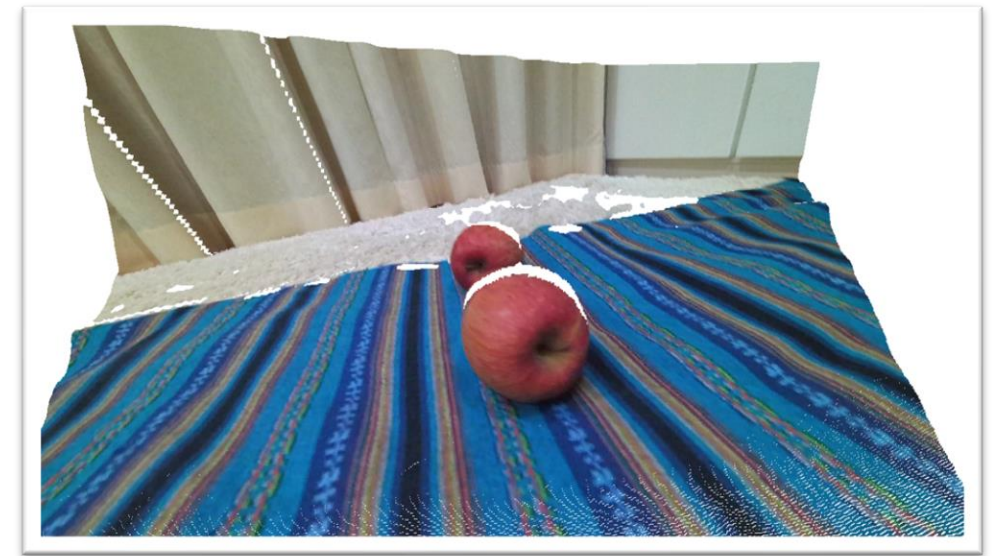


深度画像(1280x720px 16bit)

3.2 ②点群化

取得したカラー画像と深度画像のペアを『intrinsic.json』をOpen3Dの『geometry.PointCloud.create_from_rgbd_image』に使用して3次元点群化

『visualization.Visualizer』を使って可視化し、1フレーム目で目立った色と深度の位置ズレを画像編集ソフトGIMP2で修正



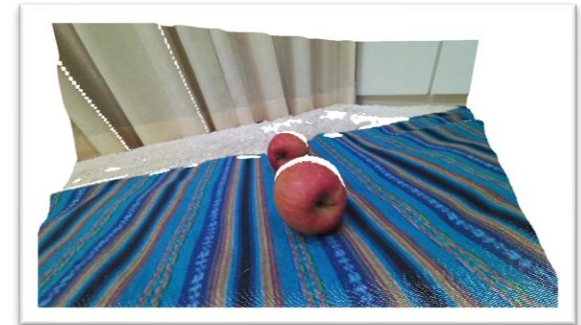
RGB-D画像を点群化したもの

3.3 ③視点変更と画像保存

Open3Dの『visualization.Visualizer』を使い点群を再度可視化

点群の大きさを二番目に小さいものに設定

左寄り・右寄り・上寄りに視点を変えた後
『capture_screen_image』関数で
表示をキャプチャ



視点上寄り



視点左寄り

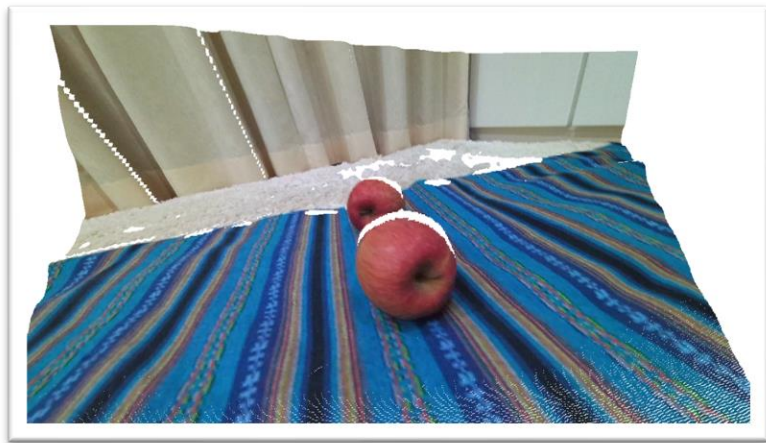


視点右寄り

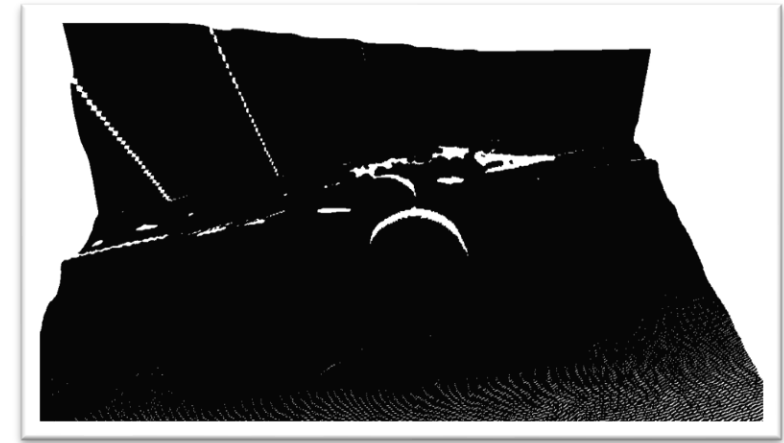
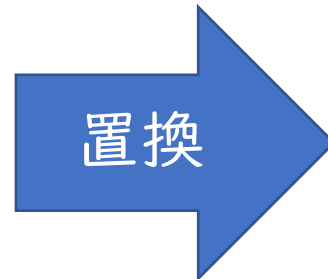
3.4 ④画像修復

キャプチャした画像を点群が全て入るような形でトリミング

トリミングした画像の背景色を白, それ以外を黒に置換し保存
これを画像補完時のマスク画像とする



トリミング後の画像



マスク画像

3.4 ④画像修復

Lama Cleanerで使用モデルを『lama』に選択した上で
トリミング後の画像とこれに対応するマスク画像を用い補完を実行

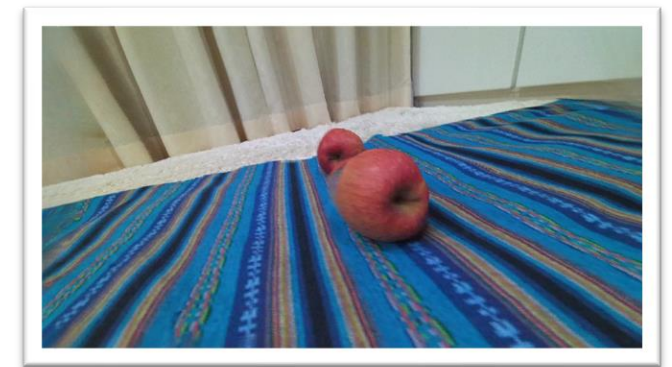
出力された補完後の画像を保存



トリミング後の画像



マスク画像



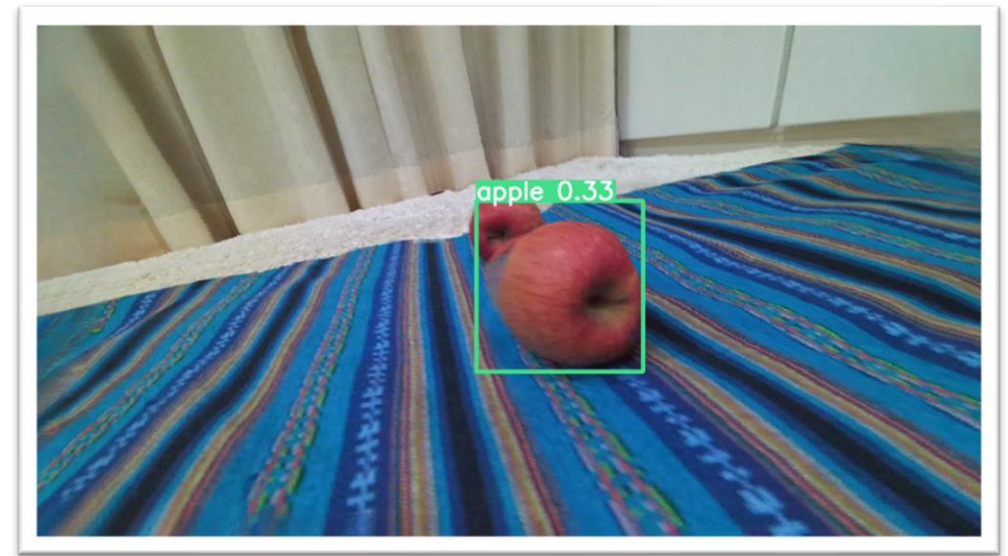
補完後の画像

3.5 ⑤物体検出

Azure Kinectで取得したカラー画像と, 補完後の画像全てに YOLOv5を用いた物体検出を実行

検出結果に関するオプションは全てデフォルトのまま使用

- 判定閾値0.25
- モデル『yolov5s』



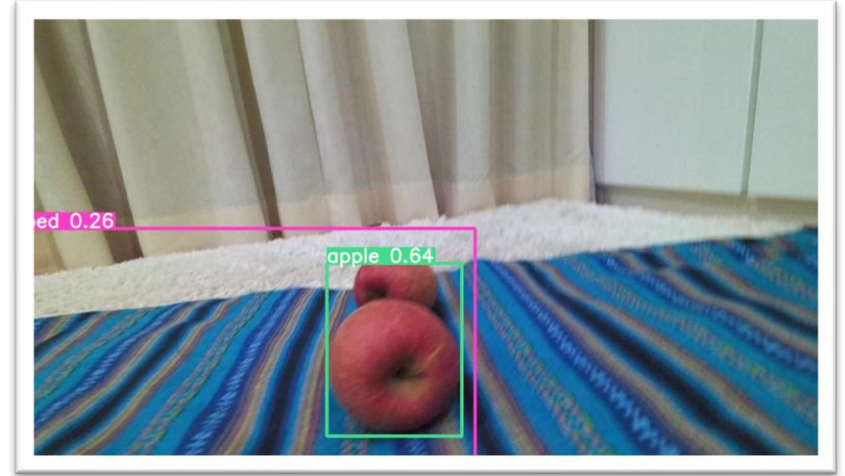
リンゴが1つ検知された例

4 実験結果

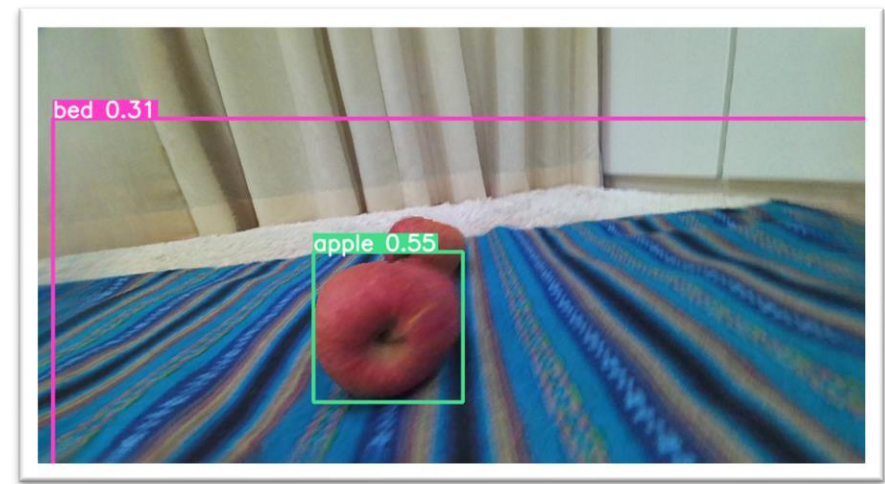
以下は目視で計測したリンゴの個数を
正解としたときの各検出結果の
正解数と正解率

画像データ	正解数	正解率
加工無し	14	約48.27%
視点左寄り加工	6	約20.69%
視点右寄り加工	5	約17.24%
視点上寄り加工	7	約24.14%

Kinectで取得したままの画像に比べて
加工後の画像の正解率は半分以下



加工前での検出結果

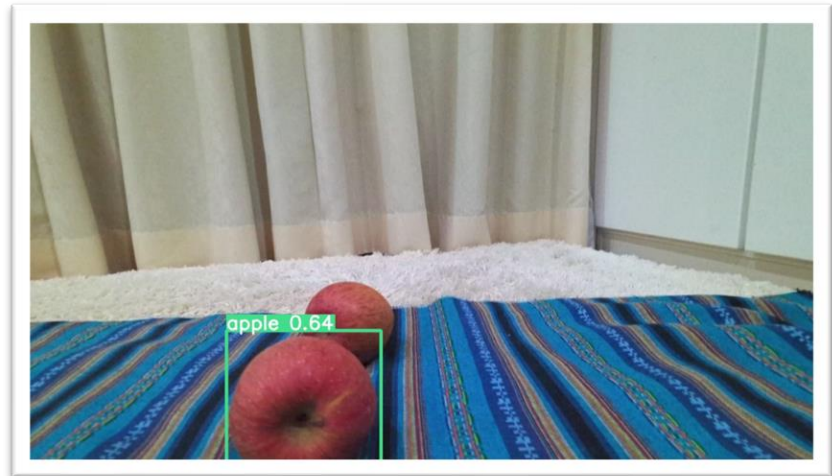


視点左寄り加工後の検出結果

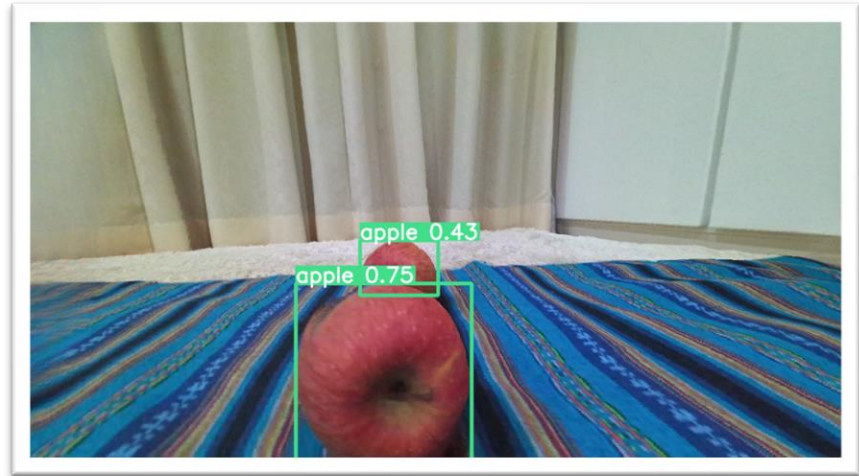
4 実験結果

加工前の画像では正しく検出出来ず
不正解となったデータで
加工後では正しい検出がされ結果が
好転した例があった

画像データ	悪化数	好転数
加工無し	-	-
視点左寄り加工	9	1
視点右寄り加工	11	1
視点上寄り加工	8	0



加工前での検出結果



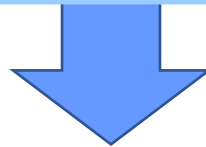
視点右寄り加工後の検出結果

5 考察

オクルージョン時での物体検出の精度低下が問題



僅かな例だが加工前の画像で未検出の物体が検出



既存技術と本演習の提案手法の検出結果を合わせ
検知数の精度向上ができる可能性があると推察

6 今後の課題

- 物体領域の検出
- 点群化に伴う画質劣化の改善
- 死角で隠されるはずの点が遮られず見えてしまうことの解消
- 適切な視点移動方向や距離の特定