

成果発表

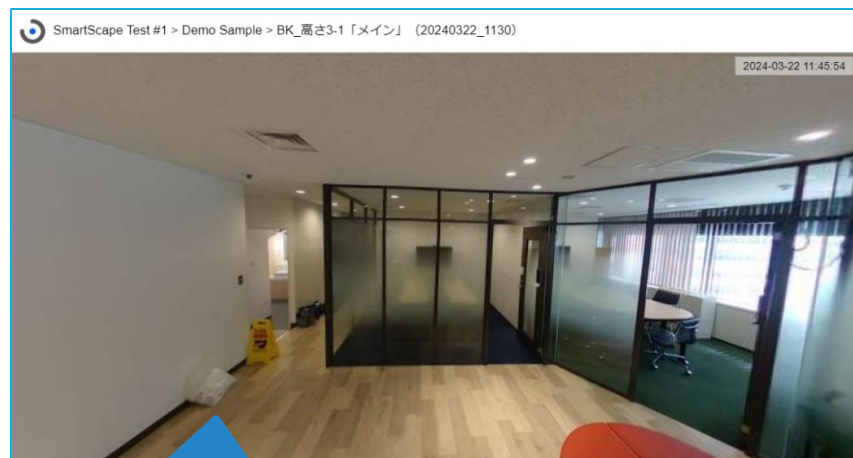
日付: 2024/8/9

目次

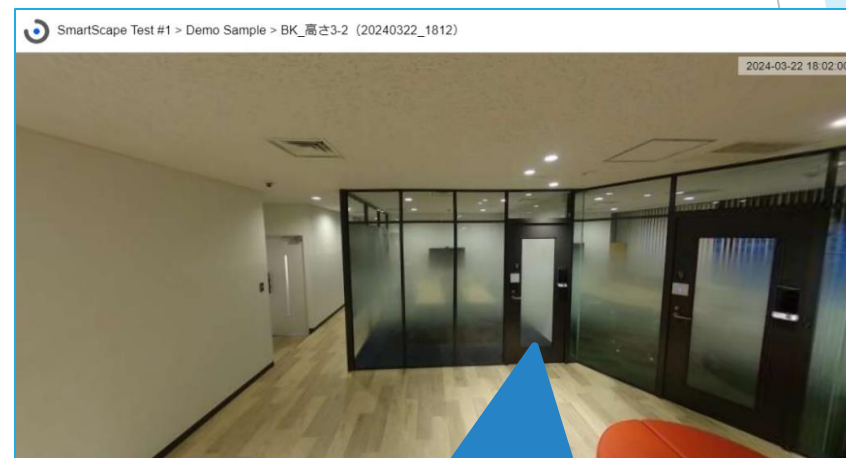
- ▶ 課題内容
- ▶ 実装の流れ
- ▶ 画像解析、画像処理で使⽤した⼿法
- ▶ その⼿法を選⽢した理由
- ▶ ライブラリ等、開発で使⽤したもの
- ▶ 成果報告
- ▶ 所感

課題内容

- ▶ 時間帯の異なる同じ場所の画像から変化した場所を見つけ出し、ユーザへわかりやすく伝える



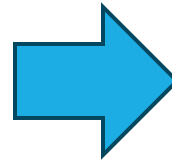
- 廊下に障害物がある



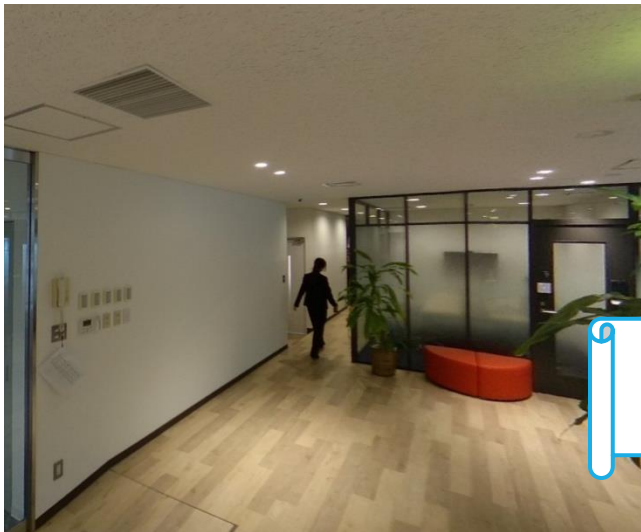
- ドアが閉まっている

- ▶ 検討事項
 - ▶ 差異検出口ジック
 - ▶ 差異検出の精度
 - ▶ ユーザへの提示方法

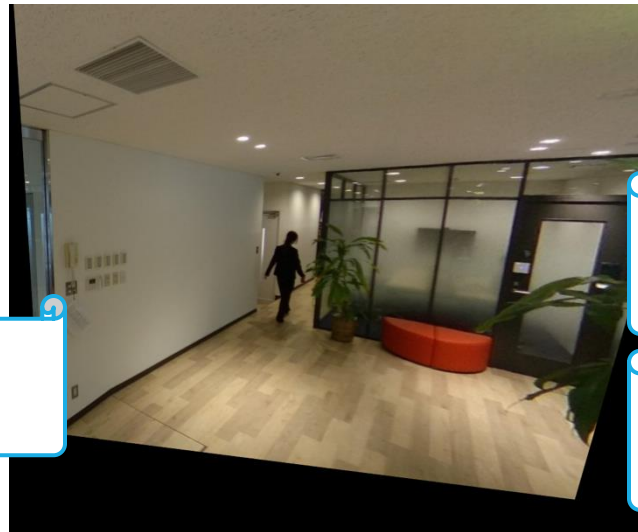
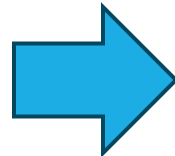
成果報告



処理の流れ 1/3

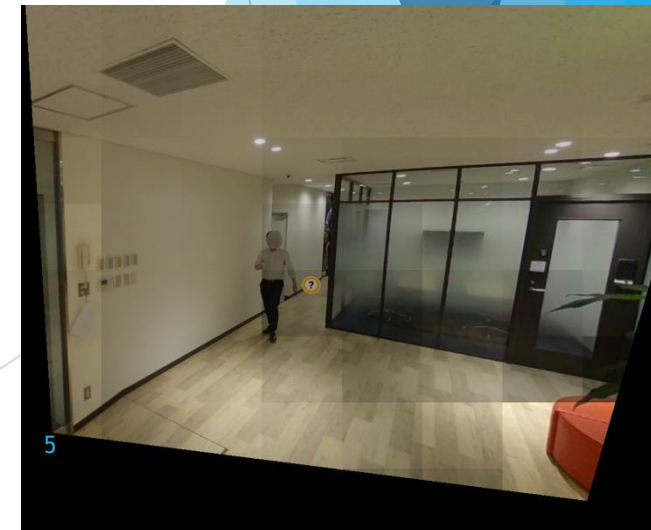
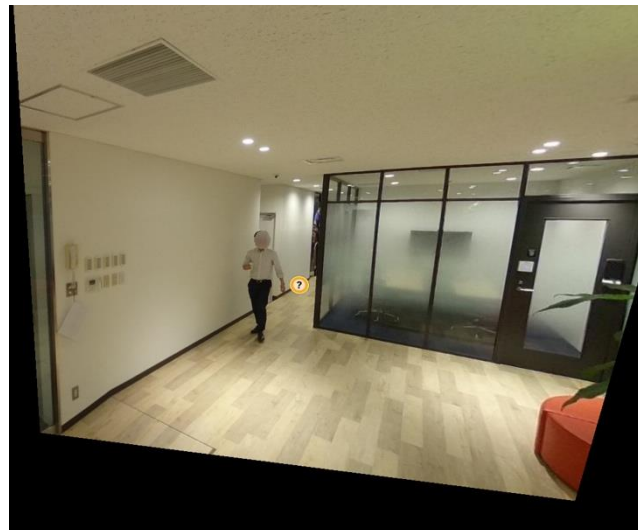
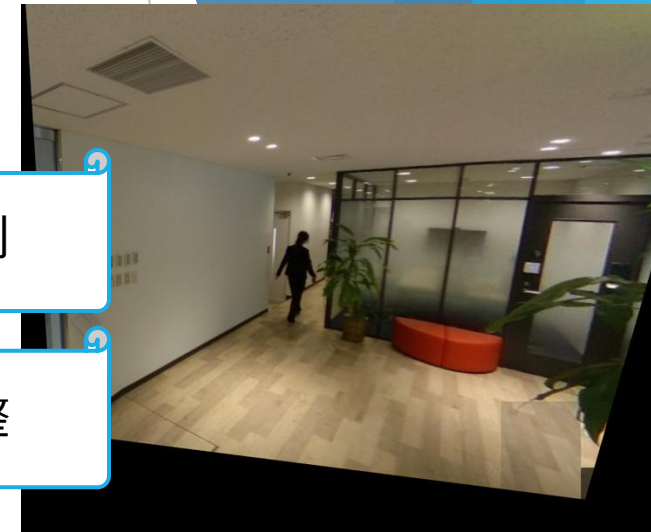
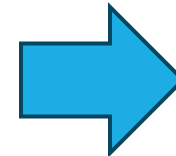


位置合わせ



画像分割

明度調整



処理の流れ 2/3

グレースケール変換

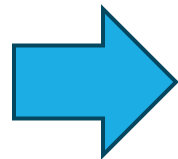
ノイズ付与

差分抽出

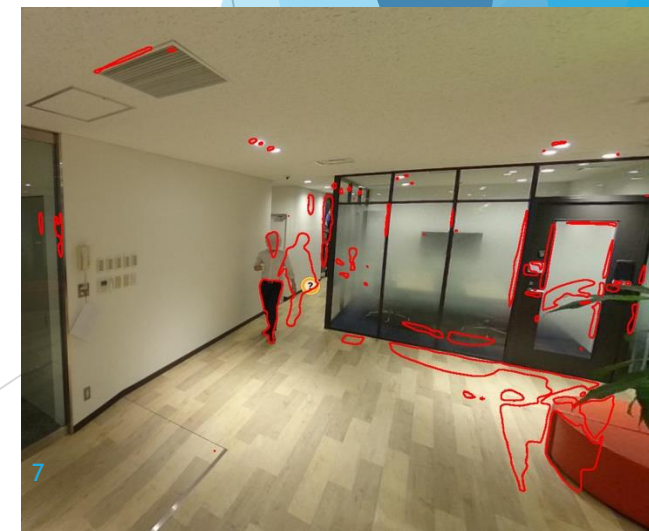
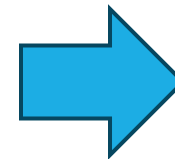
処理の流れ 3/3



二値化



差分出力



画像解析、画像処理で使した手法

- ▶ 画像の位置合わせ
 - ▶ ホモグラフィ変換
- ▶ 明度変更
 - ▶ ヒストグラム平坦化
 - ▶ 明度調整
- ▶ 画像分割
- ▶ フィルタリング
 - ▶ ガウシアンフィルタ
- ▶ 差分取得処理

処理の流れ 1/3



画像の位置合わせ: ホモグラフィ変換

目的

- ▶ 差分を検出するために、二つの画像を同じ視点から比較したい

メリット

- ▶ 画像間の視点の違いを補正できる
- ▶ 柔軟性 (平行移動、回転、スケーリング、透視変換)

デメリット

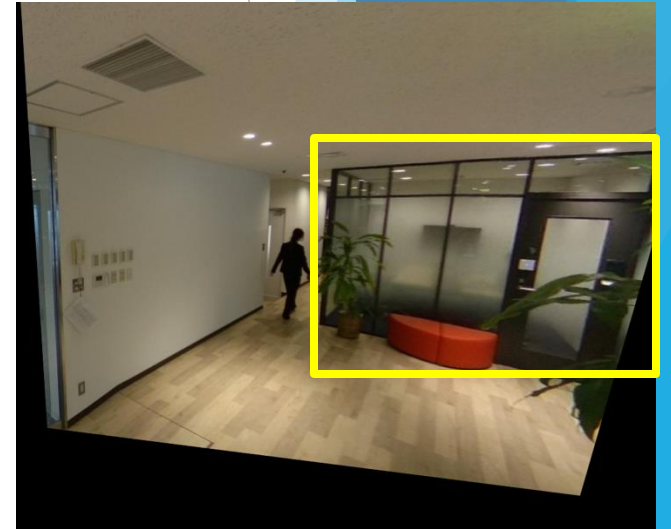
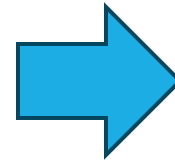
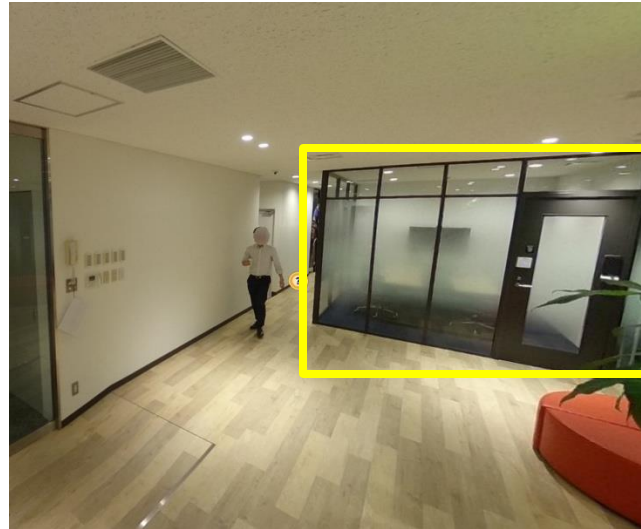
- ▶ 特徴点の検出が必要
- ▶ 非線形な変換ができない (魚眼レンズ、画像間で物体が変形、etc.)
- ▶ 3D変換に対応できない

ホモグラフィ変換: 成功例

変換元画像



変換先画像



ホモグラフィ変換: 失敗例

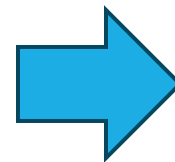
変換元画像



変換先画像



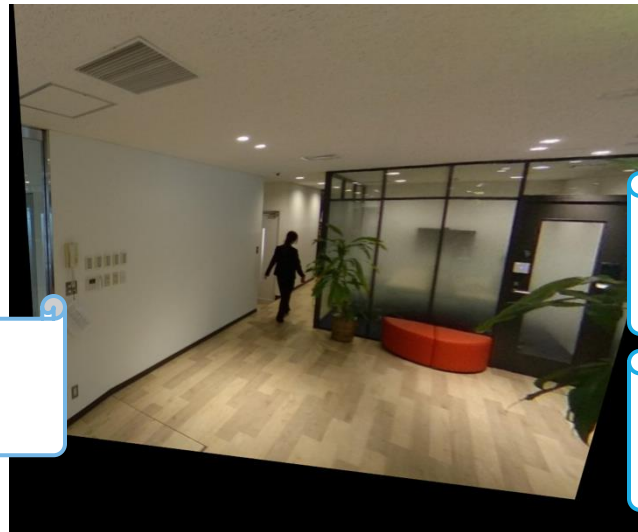
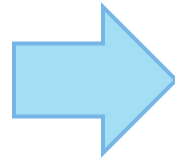
マスの形や大きさに
特徴がない



処理の流れ 1/3

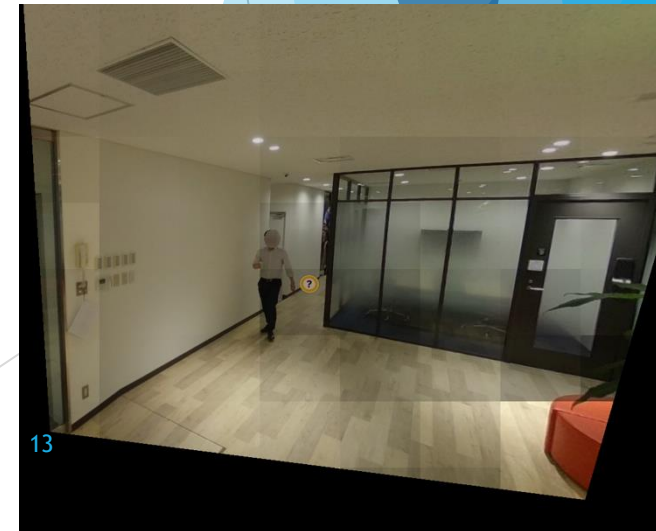
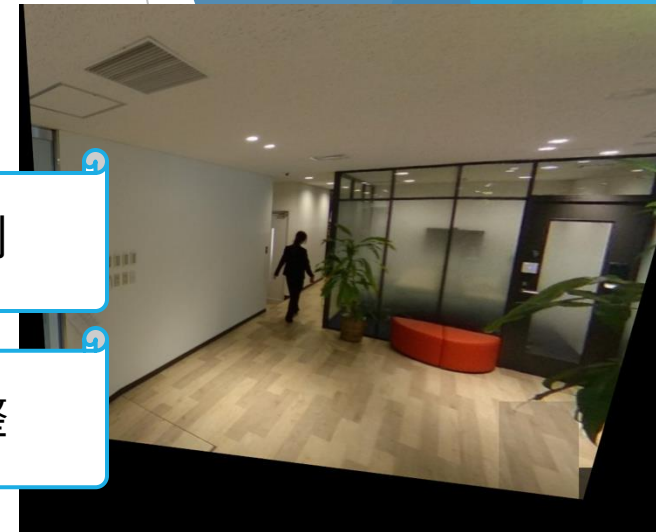
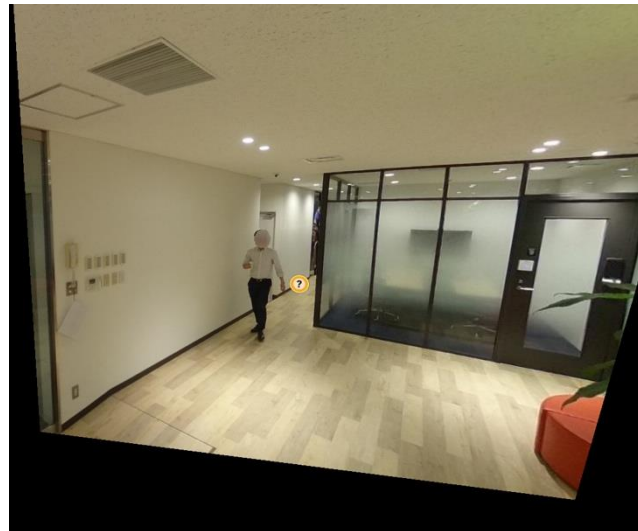
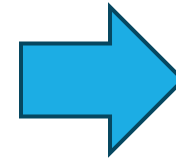


位置合わせ



画像分割

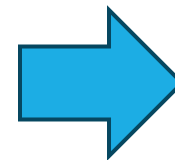
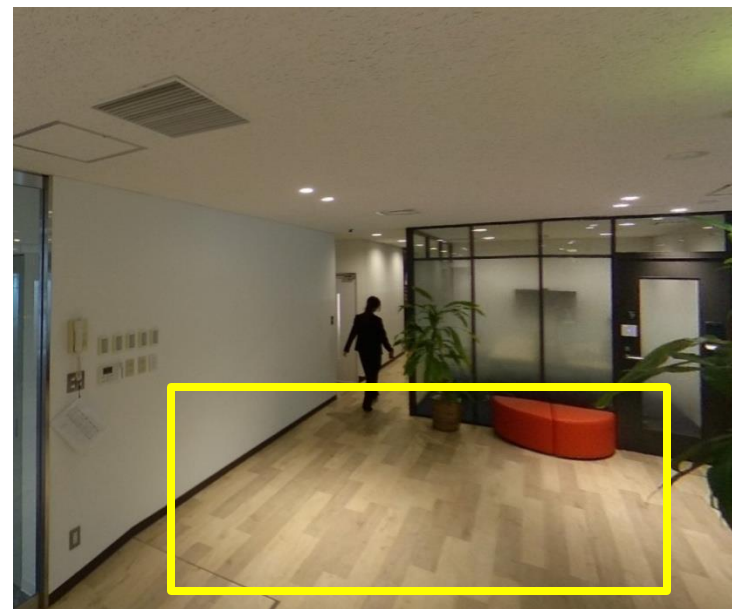
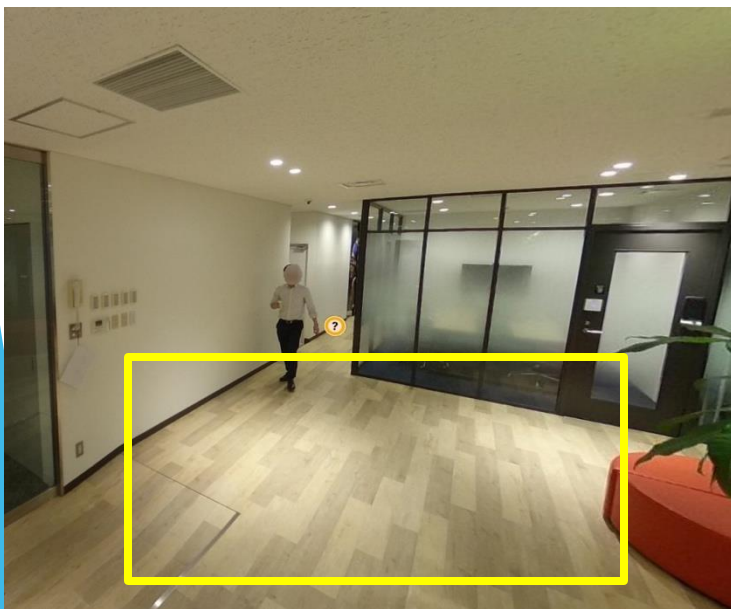
明度調整



明度調整

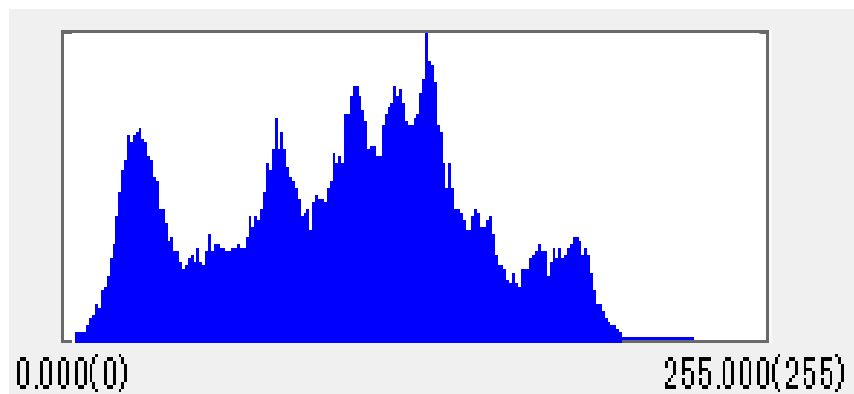
問題点

- ▶ 配置は同じなのに明るさの違いを差分として抽出してしまう

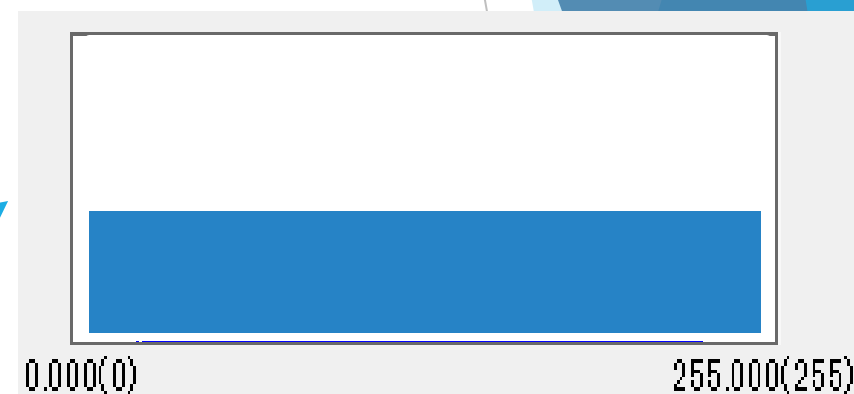


明度調整の手法

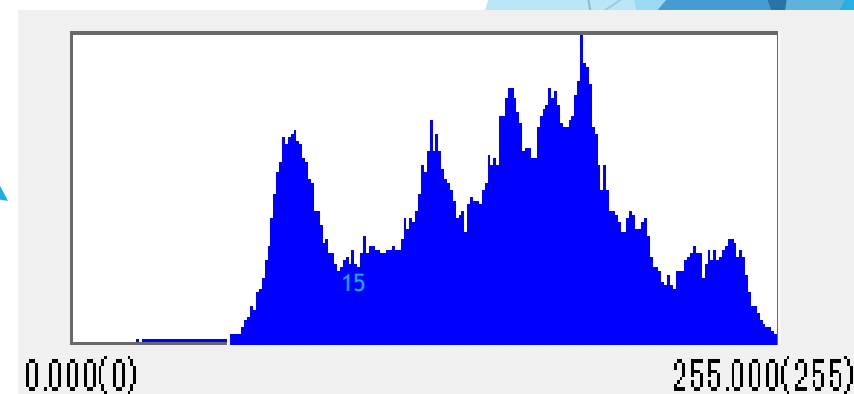
- ▶ ヒストグラム平坦化
- ▶ 片方の画像に明度をあわせる (オリジナル手法)
 - ▶ ヒストグラムの形状は変えずに、明度の平均値をあわせる



ヒストグラム平坦化



オリジナル手法



明度調整: オリジナル手法

① 画像を分割

明度調整前



② 明度の平均値を計算

明度の平均値: 100

③ 画面全体の明度を変更 (暗いほうに合わせる)

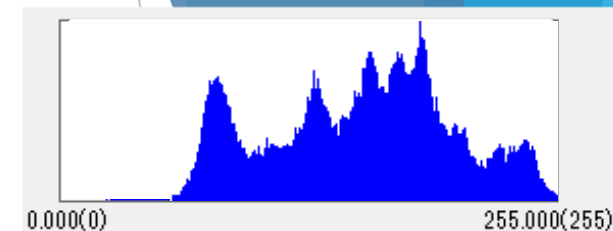


明度の平均値: 120

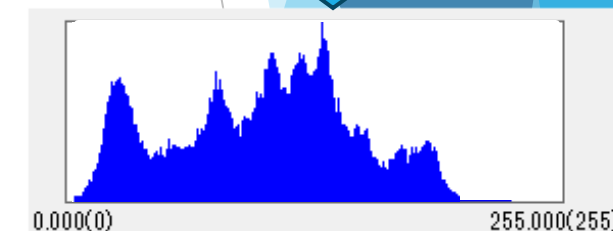
明度調整後



明度の平均値: 100



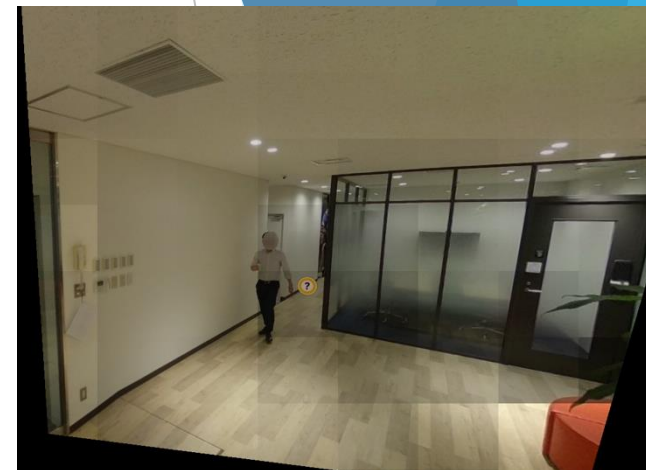
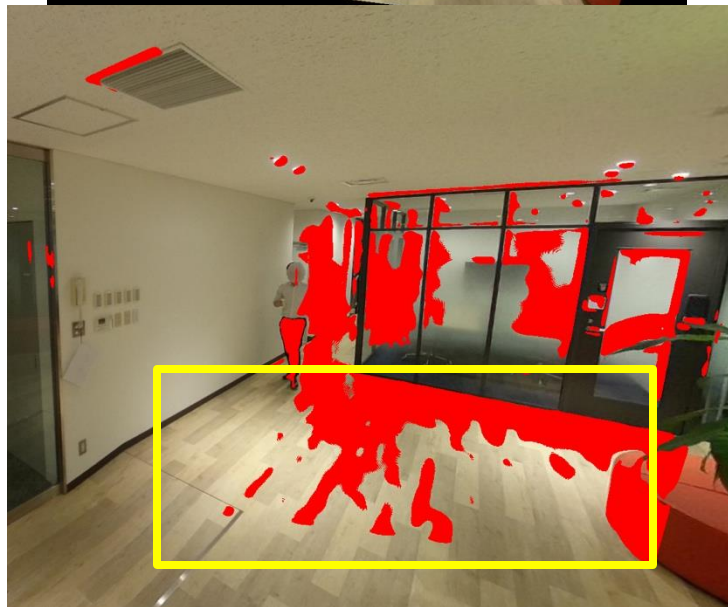
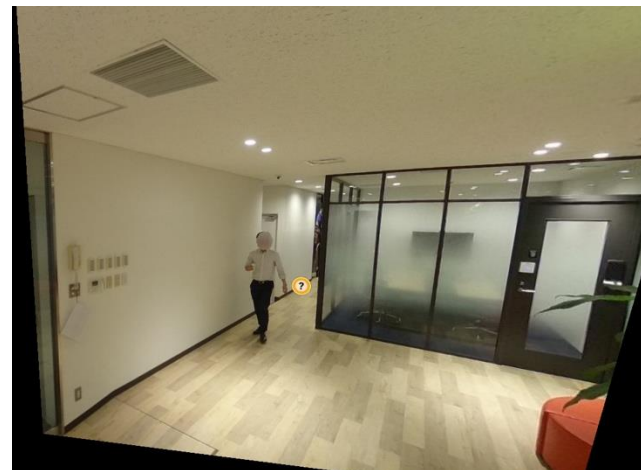
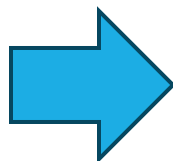
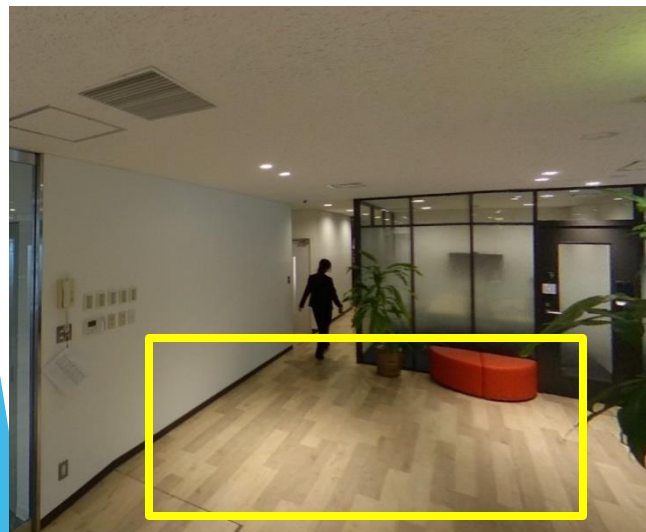
形状は変えない



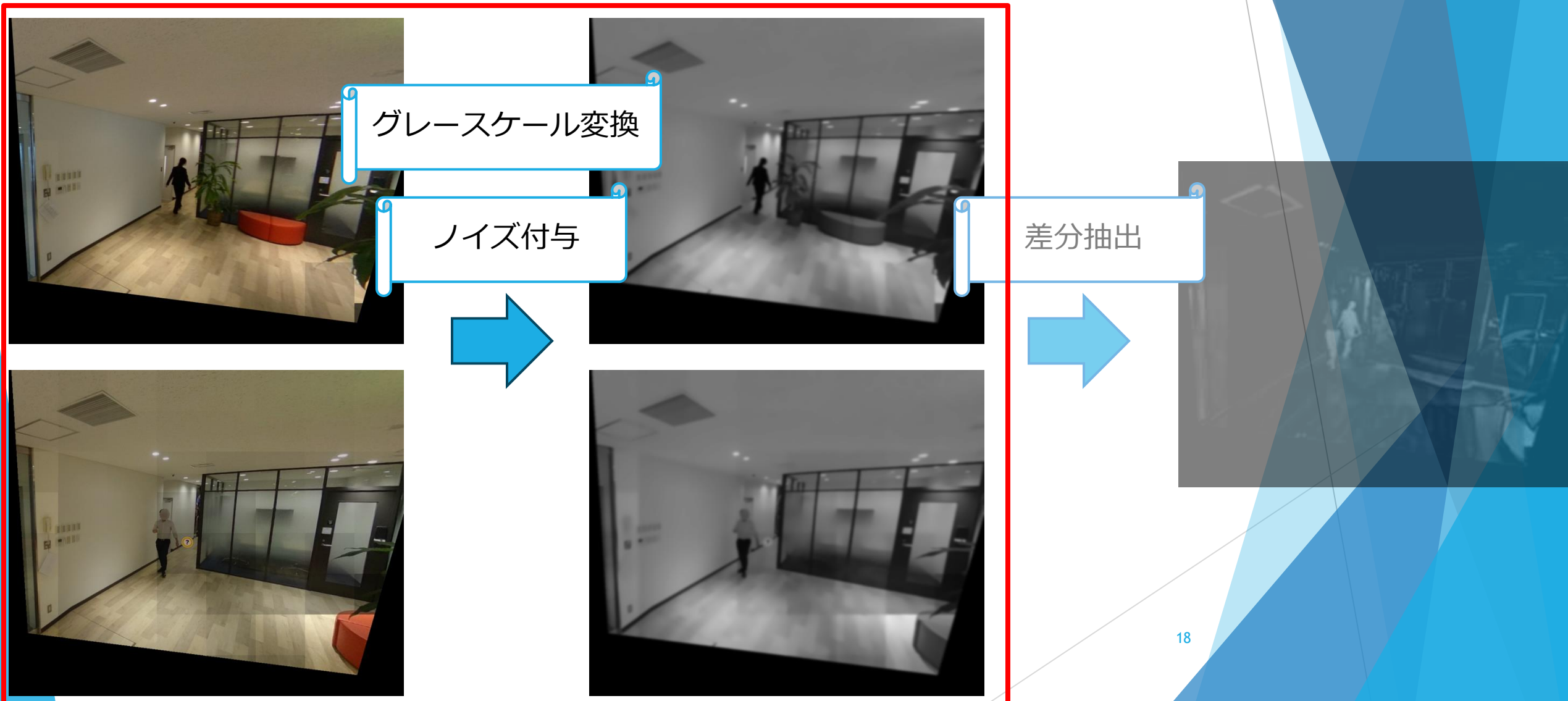
手法の比較

ヒストグラム平坦化

オリジナル手法

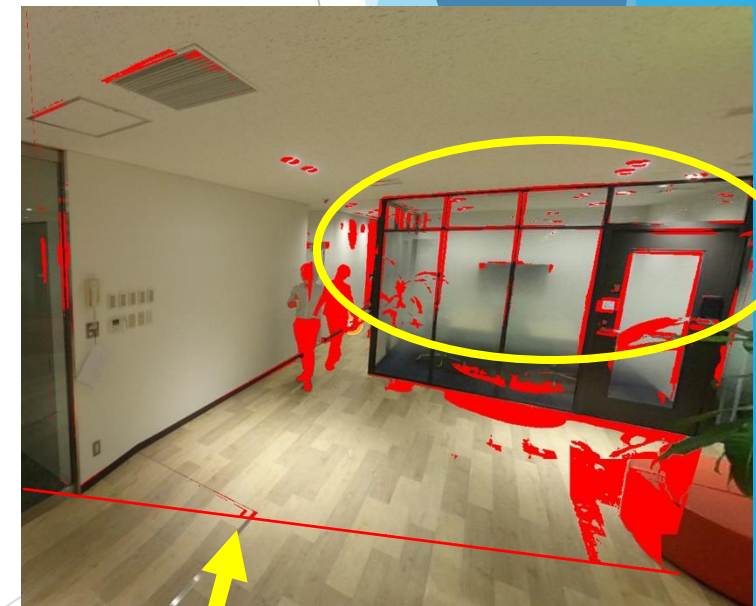
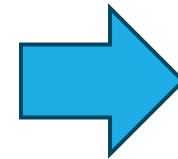


処理の流れ 2/3



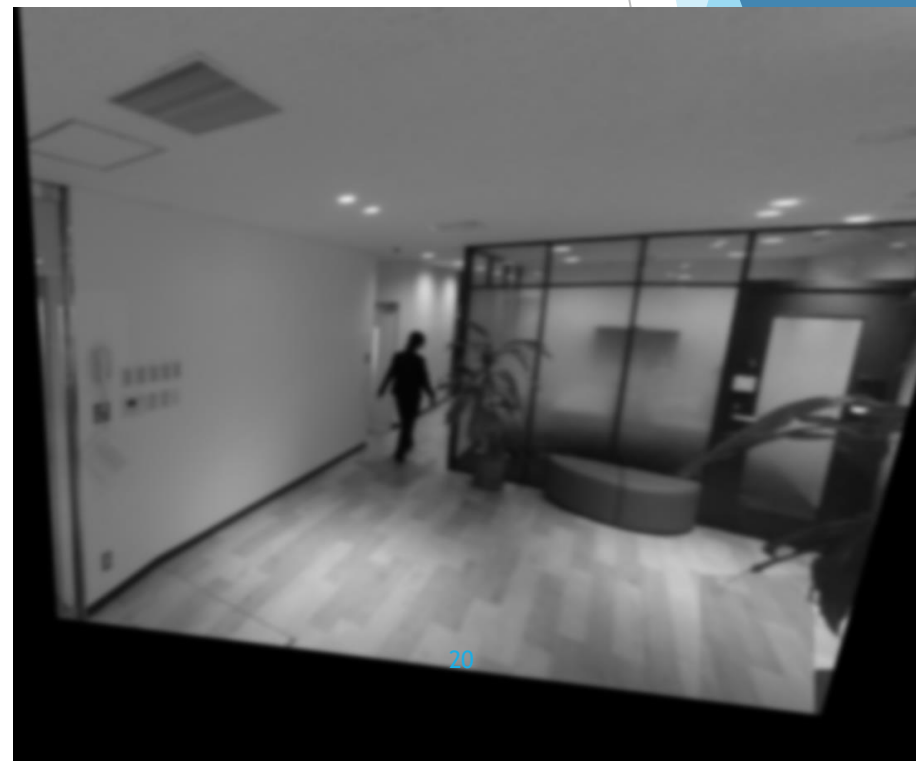
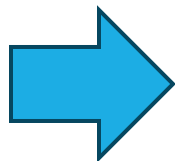
フィルタリングなし

- ▶ フィルタリングをしないと差分だらけ
- ▶ ちょっとした位置ずれに敏感
 - ▶ ホモグラフィ変換をしても、位置ずれを完璧に修正することはできない

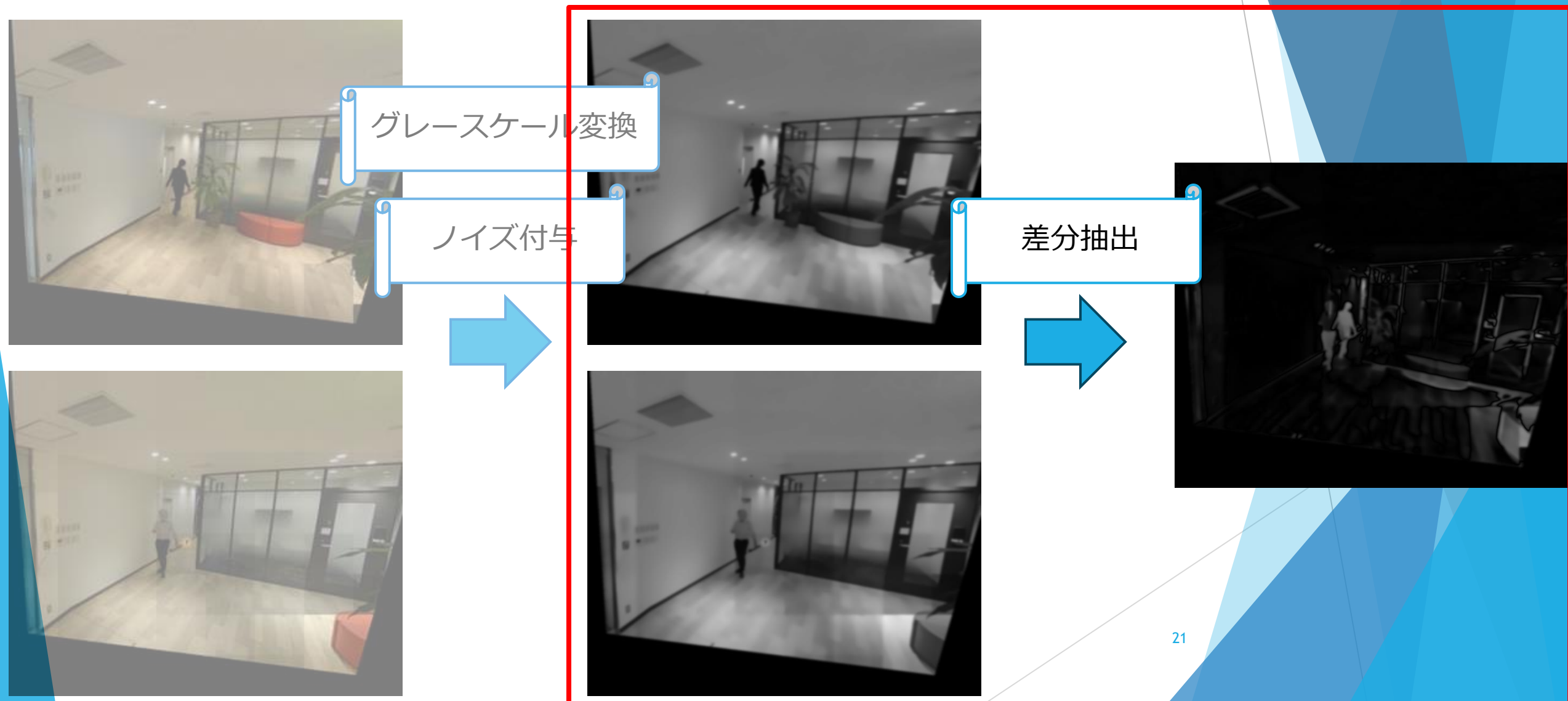


フィルタリング: ノイズ付与

- ▶ ノイズを付与して画像を荒くすることで、位置ずれなどの細かい差分を検出しないようにする
- ▶ ガウシアンフィルタを使用
- ▶ 汎用的なパラメータを検証



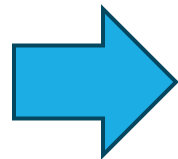
処理の流れ 2/3



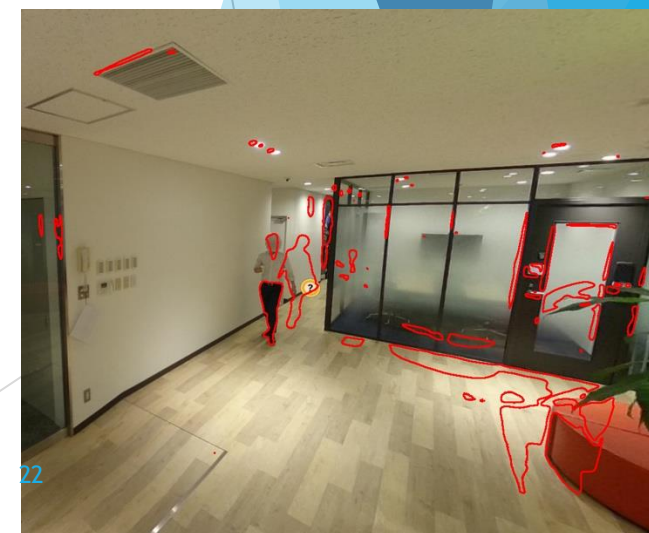
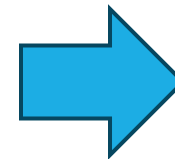
処理の流れ 3/3



二値化



差分出力

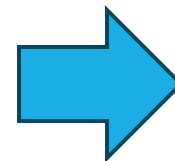
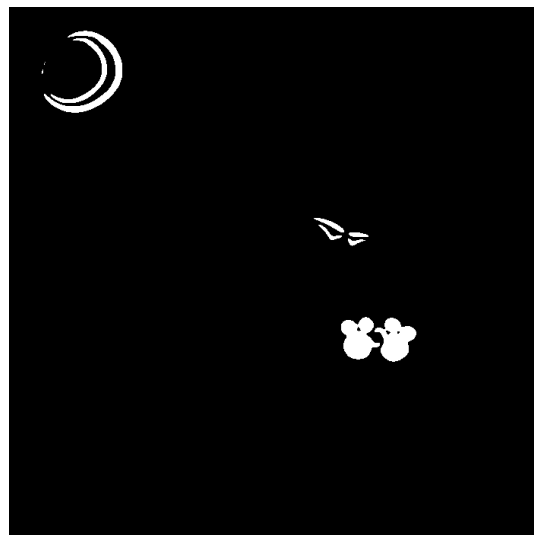
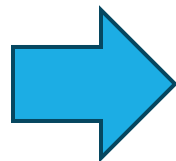


ライブラリ等、開発で使ったもの

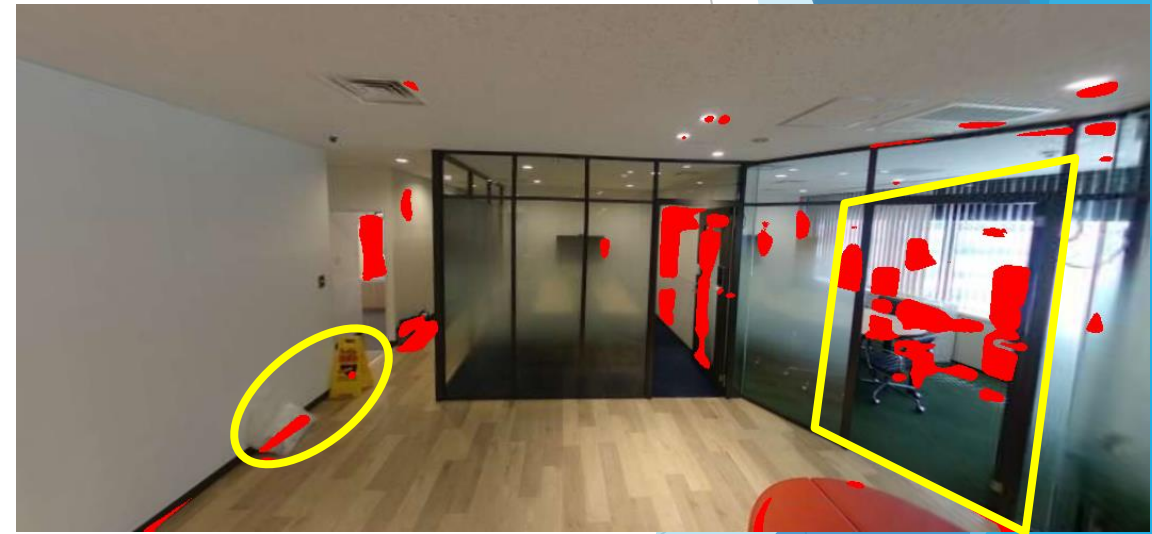
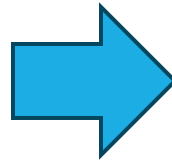
ライブラリ名	バージョン	ライセンス	概要
OpenCV-Python	4.10.0.84	Apache 2	画像処理
NumPy	1.26.4	修正BSD	数値計算
Matplotlib	3.9.0	BSD	画像表示
yolo	8	AGPL-3.0	物体検出

成果報告

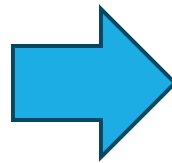
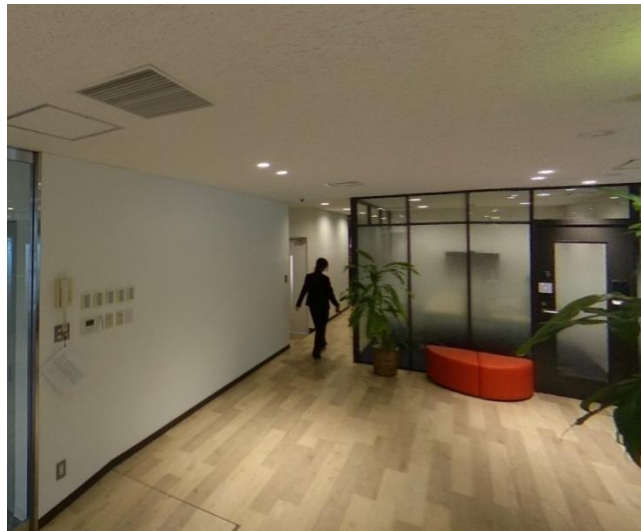
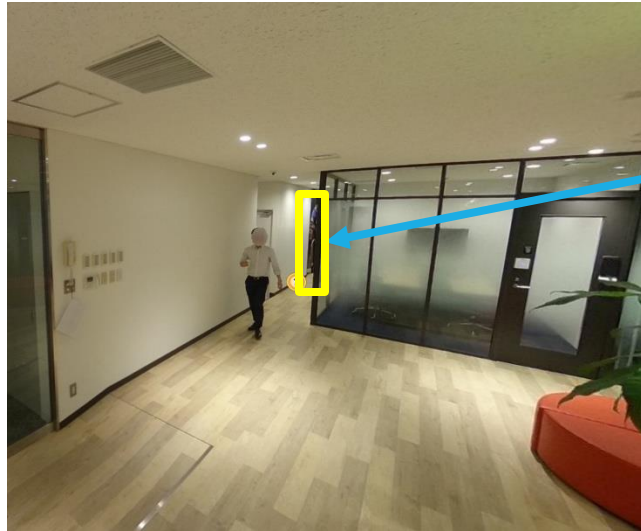
成果報告: 基礎問題



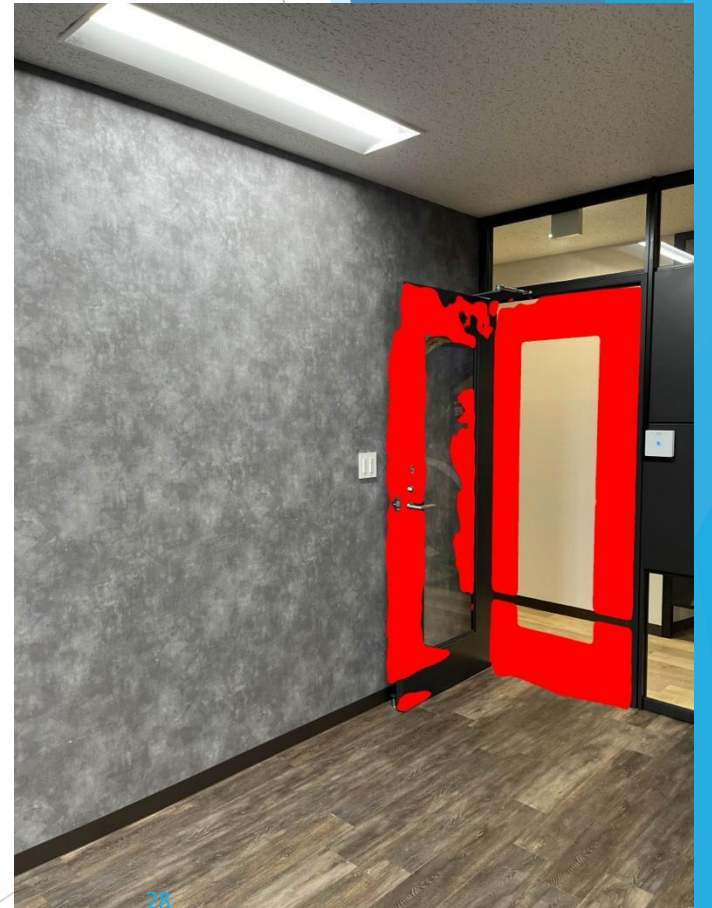
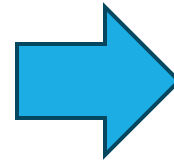
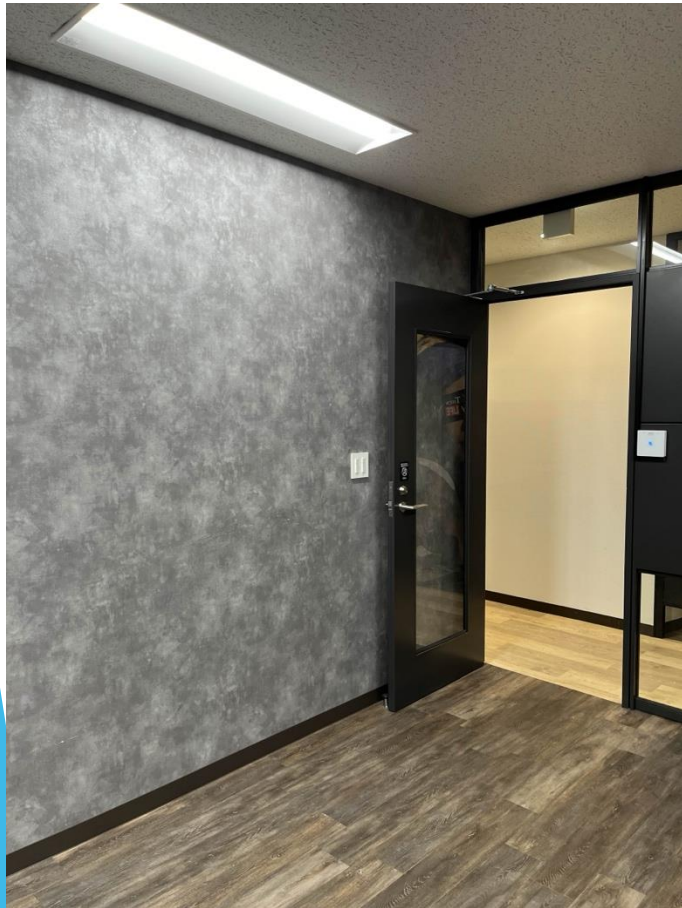
成果報告: 応用問題



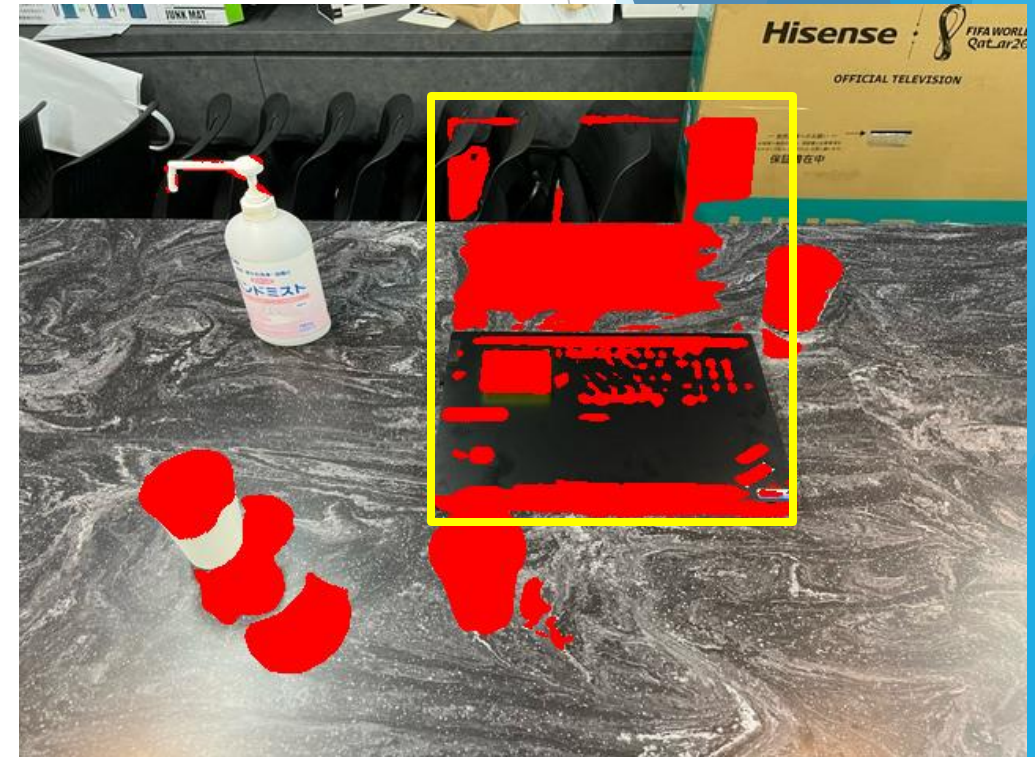
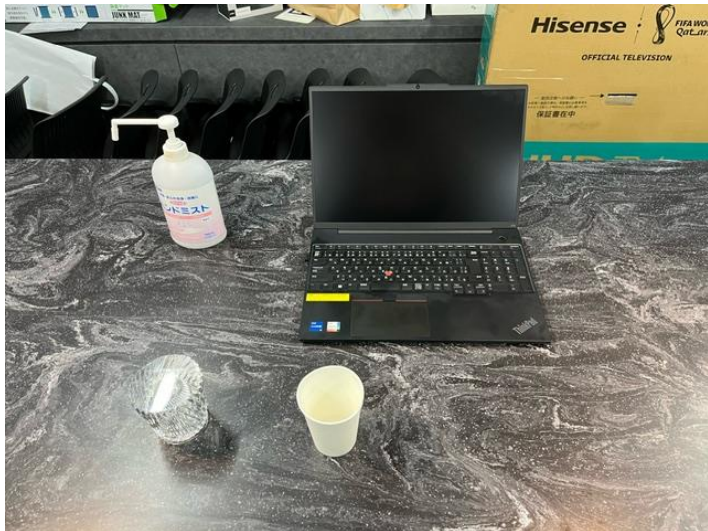
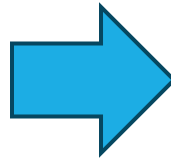
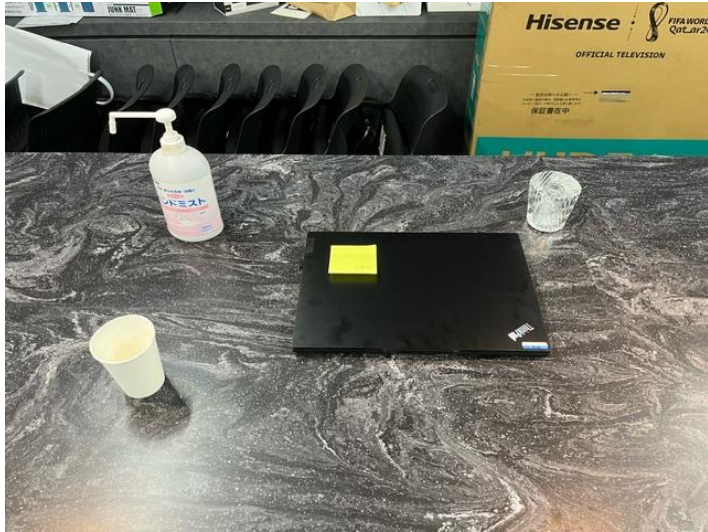
成果報告: Beamo



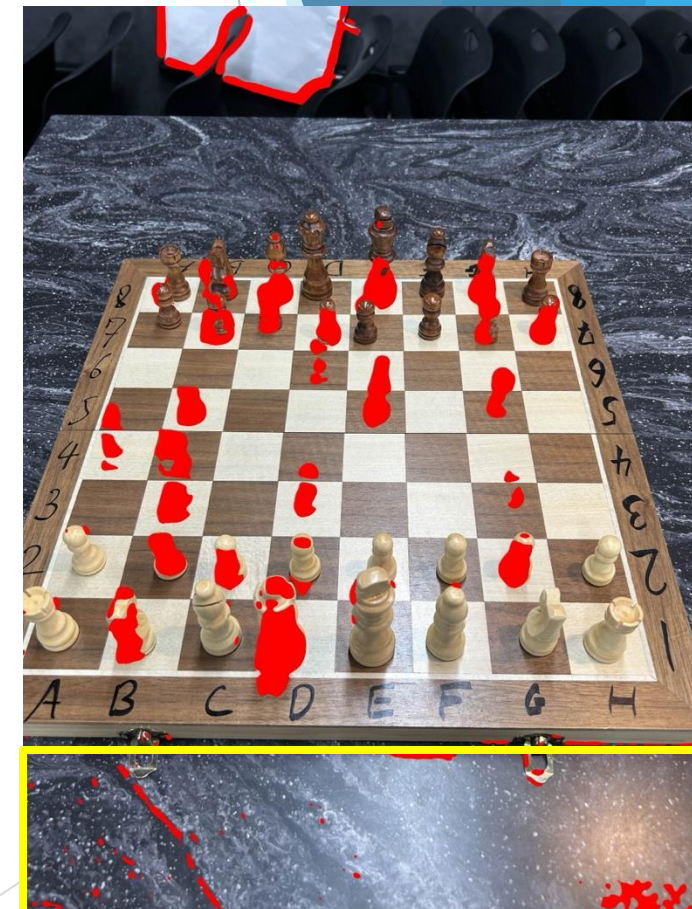
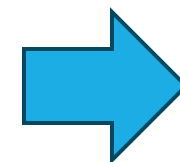
成果報告: 扉の開閉



成果報告: 複数物体

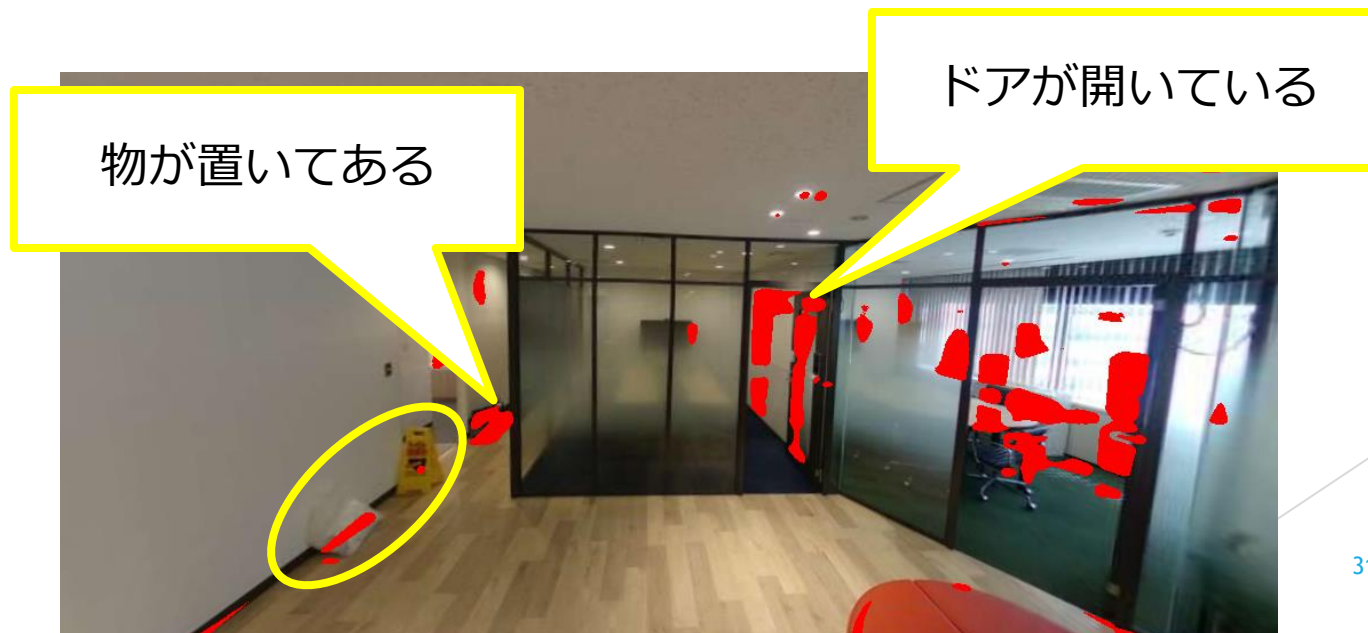


成果報告: チェスの駒



今後の展望

- ▶ 差分を言語化したい
 - ▶ 物体検出と組み合わせる
- ▶ 色が似ているもの(掃除中看板やごみ袋)を検出できるようにしたい
 - ▶ 色のコントラストを強調
 - ▶ 明度調整と逆のことをするので誤検出が増えてしまう可能性



物体検出

