物体再配置タスクのための Co-Scale Cross-Attentional Transformer

Co-Scale Cross-Attentional Transformer for Object Rearrangement Task

松尾 榛夏 *1 Haruka Matsuo　石川 慎太朗 *1 Shintaro Ishikawa　杉浦 孔明 *1 Komei Sugiura

*1 慶應義塾大学 Keio University

In this paper, we propose Co-Scale Cross-Attentional Transformer for Rearrangement Target Detection, where the model generates a change mask for objects that should be rearranged. We intorduce the Serial Encoder which consists of CoaT serial blocks and the Cross-Attentional Encoder which models the relationship between the goal and current states. We validated our method on the new dataset, and the results demonstrated that our method outperformed a baseline method on mean IoU and F_1-score.

1. はじめに

高齢化社会では，介助従事者の不足が社会問題になっており，この人手不足の解決策の一つとして，家庭用の生活支援ロボットが有望視されている [Yamamoto 19]．生活支援ロボットにとって，家庭環境内で位置に変化が生じた物体を認識し元の状態に片付ける rearrangement タスクを扱うことができれば便利である [Batra 20]．rearrangement タスクには元の状態と現在の状態の比較により再配置すべき物体の検出が重要であるが，その性能は現状十分ではない．

本研究では，事前に観察した部屋の状態を復元するため，元の状態および現在の状態から再配置すべき物体を検出する Rearrangement Target Detection (RTD) を扱う．例えば，目標状態では机の端にあった物体が，現在の状態では机の中央に移動している場合，それぞれの状態の画像から変化があった物体を検出し，その物体をマスクした画像を出力することが望ましい．また，目標状態では閉じていた扉が，現在の状態では開いている場合，それぞれの状態の画像から変化があった扉を検出し，その扉をマスクした画像を出力することが望ましい．

2 枚の入力画像の画素値を比較する手法では，再配置すべき物体をセグメンテーションできないことが多いため，RTD は困難である．このような手法を用いる場合，物体の移動によって変化した光および影の影響や，扉や引き出しの開閉により値が大きく変化する画素の少なさが原因で，セグメンテーション誤りが発生する可能性がある．RTD と関連の深い Scene Change Detection には広く既存研究が存在する（CSCDNet [Sakurada 20], DR-TANet [Chen 21]）．しかし，その多くは複雑な形状の物体やドアの角度の変化した領域のセグメンテーションに失敗することが多い．

そこで，本研究では Co-Scale Cross-Attentional Transformer を提案する．提案手法は，複数の CoaT serial block [Xu 21] を組み合わせたもので構成されている Serial Encoder を導入しているため，視覚情報の強化が可能になると考える．さらに，cross-attention 構造を用いることにより，目標状態と現在の状態の関係性をモデル化する Cross-Attentional Encoder を導入しているため，適切なマスク画像の取得が可能になると考える．既存手法と異なる点は，CoaT serial block [Xu 21] を用いた Serial Encoder を 2 個並列に導入している点，目標状態と現在の状態の関係性をモデル化する Cross-Attentional Encoder を導入している点である．

本論文の主要な貢献は以下である．

- RTD のための Co-Scale Cross-Attentional Transformer を提案する．

連絡先: 松尾榛夏，慶應義塾大学，神奈川県横浜市港北区日吉 3-14-1，haruka.matsuo-25@keio.jp

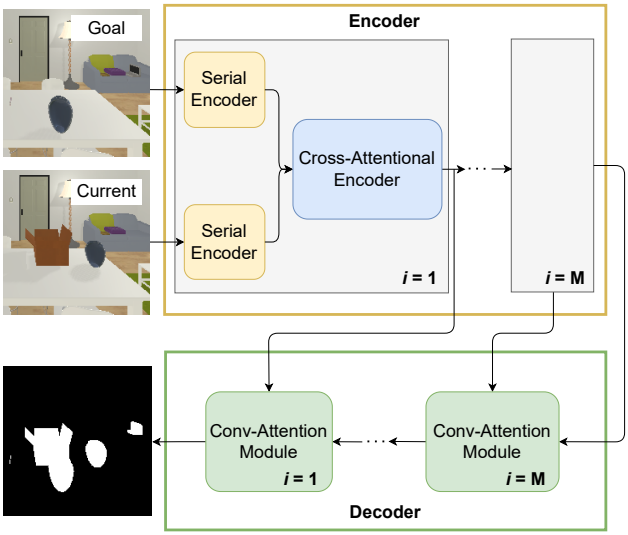

図 1: 提案手法の概要．

- 目標状態および現在の状態のそれぞれに対して CoaT serial block [Xu 21] を用いて特徴量を抽出する Serial Encoder を導入する．
- 目標状態と現在の状態の画像特徴量の関係性をモデル化する Cross-Attentional Encoder を導入する．

2. 関連研究

変化検出分野では多くの研究が行われている [Shi 20]．変化検出分野は Remote Sensing Change Detection（RS）と Scene Change Detection（SCD）の 2 種類に分けることができる．RS は，衛星画像や航空画像を用いて，建物，道路，農地などの変化を検出するタスクである [Lyu 18, Bandara 22]．SCD は，ストリートビュー画像や室内環境で得られた画像を用いて，街並みや物体の変化を検出するタスクである．CSCDNet [Sakurada 20] は，カメラ視点誤差による推定誤差を解決するために correlation layer を導入した新しいシャムネットワーク構造をもつ手法である．

CSCDNet [Sakurada 20] などの多くの SCD 手法とは異なり，ドアや引き出しの角度の変化を検出することができる．また，ResNet [He 16] で抽出した画像特徴量を連結する CSCDNet [Sakurada 20] の変化検出性能は，2 枚の画像の関係性をモデル化するには不十分である．一方，本手法では，複数の CoaT serial block [Xu 21] を並列で用いた Serial Encoder，および

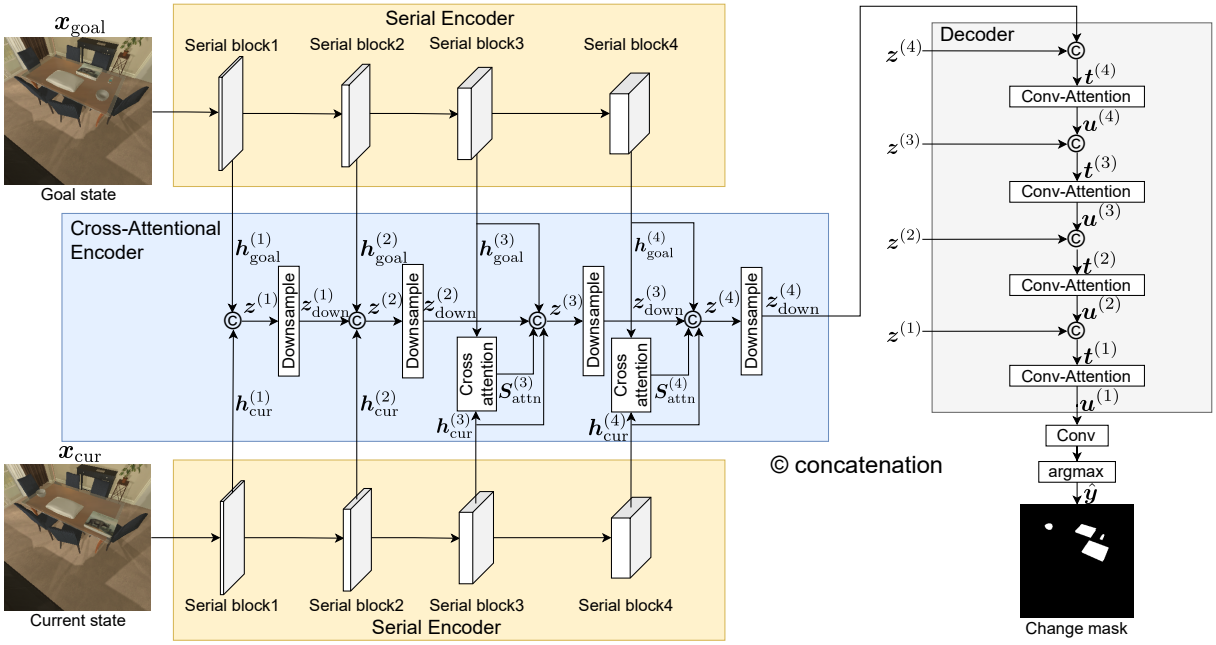

図 2: 提案手法のネットワーク構造.

cross-attention を用いた Cross-Attentional Encoder を導入している.

物体再配置タスクの分野でも，多くの研究が行われている [Batra 20]. [Trabucco 22] は再配置すべき物体を決めるために，ボクセルベースのセマンティックマップを使用する手法を提案している.

生活支援ロボットが家庭環境から取得した画像を用いて行うタスクとして，RTD 以外に，[Matsubara 22, Magassouba 21, Iocchi 15] が挙げられる. [Matsubara 22] は容器の 3D モデルを推定する際に最適なフレームを選択するために，マスクベースの幾何学的アルゴリズムを提案している. [Magassouba 21] では，視覚情報の入力のみから衝突可能性を推定する手法である PonNet を提案している. RoboCup@Home は片付けタスクを行うロボットのベンチマークとして実施されており，目標状態は画像では与えられていない [Iocchi 15].

3. 問題設定

本論文は，様々な室内環境において，目標状態および現在の状態から再配置すべき物体を検出する RTD を対象とする. 本対象タスクでは，目標状態と現在の状態の画像が与えられたとき，モデルは位置が変化した物体を検出し，その物体をマスクした画像を出力する.

入力は目標状態および現在の状態の RGB 画像である. 出力は変化マスク画像である. 本論文で使用する用語を以下のように定義する.

- **再配置対象**：位置と向きが変更された物体. 目標状態および現在の状態の間で角度が変化した引き出しや扉.

本論文は再配置対象の検出が目的である為，物体のカテゴリ認識および再配置は扱わない. また，2022 AI2-THOR Rearrangement Challenge [Weihs 21] の設定と合わせるため，1 画像中の再配置対象の個数は 5 個以下であることを前提とする. 評価尺度として mean Intersection over Union (mIoU) と F_1-score を使用する.

4. 提案手法

図 2 に提案手法のネットワーク構造を示す. 本モデルの主要モジュールは Serial Encoder, Cross-Attentional Encoder, Decoder の 3 個である.

ネットワークの入力は $\boldsymbol{x} = f\boldsymbol{x}_{\text{goal}}, \boldsymbol{x}_{\text{cur}}g$ と定義する. ここに，$\boldsymbol{x}_{\text{goal}}$ 2 $\mathsf{R}^{3\ \ 256\ \ 256}$ および $\boldsymbol{x}_{\text{cur}}$ 2 $\mathsf{R}^{3\ \ 256\ \ 256}$ はそれぞれ目標状態の RGB 画像および現在の状態の RGB 画像を表す.

4.1 Serial Encoder

Serial Encoder は M 個の serial block [Xu 21] を用いて画像特徴量を抽出する. i 番目の serial block での処理は以下である. まず，入力 $\boldsymbol{x}_{\text{goal}}$ にパッチ埋め込み層を用いてダウンサンプリングを行い，$\boldsymbol{o}^{(i)}_{\text{goal}}$ 2 $\mathsf{R}^{H_i W_i\ \ C_i}$ を得る. ここで，H_i, W_i, C_i はそれぞれ $\boldsymbol{o}^{(i)}_{\text{goal}}$ の高さ，幅，チャネル数を表す.

次に，$\boldsymbol{o}^{(i)}_{\text{goal}}$ を平坦化して画像トークンとし，画像トークンに CLS トークン $\boldsymbol{v}^{(i)}_{\text{goal}}$ 2 R^{C_i} を結合する. そして，Conv-Attention Module を通して $\boldsymbol{v}^{(i)}_{\text{token}}$ に Depthwise Convolution および Factorized Attention [Xu 21] を適用する.

その後，画像トークンと CLS トークンを分離し，画像トークンを変形して $\boldsymbol{h}^{(i)}_{\text{goal}}$ 2 $\mathsf{R}^{C_i\ \ H_i\ \ W_i}$ を得る. $f\boldsymbol{h}^{(i)}_{\text{goal}}\ j\ i = 1, \ldots, Mg$ がこのモジュールの出力である. 同様に，入力 $\boldsymbol{x}_{\text{cur}}$ から $f\boldsymbol{h}^{(i)}_{\text{cur}}\ j\ i = 1, \ldots, Mg$ を得る.

4.2 Cross-Attentional Encoder

本モジュールでは，ダウンサンプリングと cross-attention を層ごとに適用することにより，目標状態と現在の状態の間の潜在的な特徴を抽出する. 入力として $f\boldsymbol{h}^{(i)}_{\text{goal}}, \boldsymbol{h}^{(i)}_{\text{cur}}\ j\ i = 1, \ldots, Mg$ を用いる.

まず，$\boldsymbol{h}^{(1)}_{\text{goal}}$ と $\boldsymbol{h}^{(1)}_{\text{cur}}$ を連結して $\boldsymbol{z}^{(1)}$ 2 $\mathsf{R}^{(C_1\ \ 2)\ \ H_1\ \ W_1}$ を得る. 次に，$\boldsymbol{z}^{(1)}$ をダウンサンプリングして $\boldsymbol{z}^{(1)}_{\text{down}}$ 2 $\mathsf{R}^{C_1\ \ H_2\ \ W_2}$ を得る. その後，$\boldsymbol{z}^{(1)}_{\text{down}}$, $\boldsymbol{h}^{(2)}_{\text{goal}}$, $\boldsymbol{h}^{(2)}_{\text{cur}}$ を連結して $\boldsymbol{z}^{(2)}$ 2 $\mathsf{R}^{(C_1+C_2\ \ 2)\ \ H_2\ \ W_2}$ を得る. そして，$\boldsymbol{z}^{(2)}$ をダウンサンプリングして $\boldsymbol{z}^{(2)}_{\text{down}}$ 2 $\mathsf{R}^{C_2\ \ H_3\ \ W_3}$ を得る.

任意の行列 $\boldsymbol{X}_{\text{A}}$ および $\boldsymbol{X}_{\text{B}}$ を用いた cross-attention を以下のように定義する.

$$\boldsymbol{f}^{(j)}_{\text{attn}}(\boldsymbol{X}_{\text{A}}, \boldsymbol{X}_{\text{B}}) = \text{softmax}(\frac{(W^{(j)}_q \boldsymbol{X}_{\text{A}})(W^{(j)}_k \boldsymbol{X}_{\text{B}})^{\top}}{\sqrt{d}})(W^{(j)}_v \boldsymbol{X}_{\text{B}})$$

ここで，W_q, W_k, W_v は学習可能な重みであり，d はスケーリングファクターである.

表 1: 定量的結果および Ablation study

Method		Feature Extractor	Cross attention		mIoU [%]	F_1-score [%]
			$\boldsymbol{h}^{(3)}_{\text{goal}}, \boldsymbol{h}^{(3)}_{\text{cur}}$	$\boldsymbol{h}^{(4)}_{\text{goal}}, \boldsymbol{h}^{(4)}_{\text{cur}}$		
CSCDNet [Sakurada 20]		-	-	-	52.5 4.1	81.7 1.9
Ours	(i-a)	CoaT [Xu 21]		✓	54.7 0.1	83.1 0.1
	(i-b)	CoaT [Xu 21]	✓		53.6 0.1	82.5 0.0
	(full)	CoaT [Xu 21]	✓	✓	**55:0±0:4**	**83:3±0:1**

$i = 3, \ldots, M$ 1 のとき，$\boldsymbol{h}^{(i)}_{\text{goal}}$ および $\boldsymbol{h}^{(i)}_{\text{cur}}$ に cross-attention を適用する．それらに multi-head attention を適用して，以下のように attention スコア $\boldsymbol{S}^{(i)}_{\text{attn}}$ を求める．

$$\boldsymbol{S}^{(i)}_{\text{attn}} = f\boldsymbol{f}^{(j)}_{\text{attn}}(\boldsymbol{h}^{(i;j)}_{\text{goal}}, \boldsymbol{h}^{(i;j)}_{\text{cur}}) \, j \, j = 1, \ldots, Ag$$

ここで，A は attention ヘッドの数を表す．

以上より，$\boldsymbol{z}^{(i \ 1)}_{\text{down}}$, $\boldsymbol{S}^{(i)}_{\text{attn}}$, $\boldsymbol{h}^{(i)}_{\text{goal}}$ および $\boldsymbol{h}^{(i)}_{\text{cur}}$ を連結して $\boldsymbol{z}^{(i)}$ 2 $\mathsf{R}^{(C_{i-1}+C_i \ \ 3) \ \ H_i \ \ W_i}$ を得る．その後，$\boldsymbol{z}^{(i)}$ に対してダウンサンプリングを行い，$\boldsymbol{z}^{(i)}_{\text{down}}$ 2 $\mathsf{R}^{C_i \ \ H_{i+1} \ \ W_{i+1}}$ を得る．

同様に，$\boldsymbol{S}^{(M)}_{\text{attn}}$ および $\boldsymbol{z}^{(M)}$ 2 $\mathsf{R}^{(C_{M-1}+C_M \ \ 3) \ \ H_M \ \ W_M}$ が得られる．また，$\boldsymbol{z}^{(M)}$ に畳み込み層を適用し，$\boldsymbol{z}^{(M)}_{\text{down}}$ 2 $\mathsf{R}^{C_M \ \ H_M \ \ W_M}$ を得る．したがって，本モジュールの出力は $f\boldsymbol{z}^{(i)} \, j \, i = 1, \ldots, Mg$ および $\boldsymbol{z}^{(M)}_{\text{down}}$ である．

4.3 Decoder

Decoder では，M 個の Conv-Attention Modules を用い、与えられた $f\boldsymbol{z}^{(i)} \ j \ i = 1, \ldots, Mg$ および $\boldsymbol{z}^{(M)}_{\text{down}}$ から予測マスク画像を得る．まず，パッチ埋め込み層を $f\boldsymbol{z}^{(M)}_{\text{down}}, \boldsymbol{z}^{(M)}g$ に適用して $\boldsymbol{t}^{(M)}$ 2 $\mathsf{R}^{H_M W_M \ \ C_{M-1}}$ を得る．次に，Conv-Attention Module およびアップサンプリング層を $\boldsymbol{t}^{(M)}$ に適用して $\boldsymbol{u}^{(M)}$ 2 $\mathsf{R}^{C_{M-1} \ \ H_{M-1} \ \ W_{M-1}}$ を得る．

さらに，$\boldsymbol{t}^{(i)}$ 2 $\mathsf{R}^{H_i W_i \ \ C_{i-1}}$ (i 2 fM 1, ..., 2g) を得るために，$f\boldsymbol{u}^{(i+1)}, \boldsymbol{z}^{(i)}g$ にパッチ埋め込み層を適用する．その後，Conv-Attention Module およびアップサンプリング層を $\boldsymbol{t}^{(i)}$ に適用して，$\boldsymbol{u}^{(i)}$ 2 $\mathsf{R}^{C_{i-1} \ \ H_{i-1} \ \ W_{i-1}}$ を得る．最後に，$f\boldsymbol{u}^{(2)}, \boldsymbol{z}^{(1)}g$ から $\boldsymbol{u}^{(1)}$ 2 $\mathsf{R}^{(C_1=2) \ \ 256 \ \ 256}$ を生成する．カーネルサイズ 1 の畳み込み層を $\boldsymbol{u}^{(1)}$ に適用した後，2 値化し，モデルの最終出力であるサイズ 256 256 の変化マスク $\hat{\boldsymbol{y}}$ を得る．

4.4 Soft Dice Loss

損失関数 L は以下のように定義される．

$$L = \lambda_{\text{ce}} L_{\text{ce}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) + \lambda_{\text{sDice}} L_{\text{sDice}}(\boldsymbol{y}, p(\hat{\boldsymbol{y}}))$$

ここで，λ_{ce} および λ_{sDice} は重みである．L_{ce} および L_{sDice} はそれぞれ交差エントロピー誤差および soft Dice loss を表す．

マスクされていない領域を改善するために，soft Dice loss を新たに導入する．この損失は IoU loss [Rahman 16] および Dice loss [Milletari 16] と関連が深い．Dice loss [Milletari 16] とは異なり，soft Dice loss はハードアサインではなくマスク画像の予測値のソフトアサインを取り入れている．L_{sDice} は以下のように定義される．

$$L_{\text{sDice}} = 1 \quad \frac{1}{K} \sum_{i=1}^{K} \frac{2 \sum_{j=1}^{N} \boldsymbol{y}_{i;j} p(\hat{\boldsymbol{y}}_{i;j})}{\sum_{j=1}^{N} \boldsymbol{y}_{i;j} + \sum_{j=1}^{N} p(\hat{\boldsymbol{y}}_{i;j}) + \epsilon}$$

ここで，K, N, $\boldsymbol{y}_{i;j}$, $\hat{\boldsymbol{y}}_{i;j}$ はそれぞれクラス数，全画素数，正解マスク画像の i 番目のクラスの j 番目の画素値，予測マスク画像の i 番目のクラスの j 番目の画素値を示す．ϵ はゼロ除算を避けるための小さな正の値で，1 10 ^{7} に設定されている．

5. 実験

5.1 データセット

本論文では，シミュレーション環境における新しい RTD データセットを構築した．既存のデータセットは物体再配置タスク向けではなく，室内環境でもないため，対象タスクには適さない．そのため，今回は AI2-THOR [Kolve 17] を用いてこのデータセットを以下の手順で構築した．

まず，ロボットの一人称視点画像を保存した．次に，ランダムに物体を移動させ，引き出しやドアの角度を変えた．さらに，再びロボットの一人称視点画像を保存した．RGB 画像を収集する際，目標状態および現在の状態のそれぞれについて，再配置された物体の ID および位置を保存した．これらを用いて，再配置対象の画素をマスクした画像を得た．再配置対象は，50cm 以上移動させた物体と定義する．

RTD データセットには 12,000 サンプル含まれ，各サンプルは目標状態および現在の状態の RGB 画像，変化マスク画像で構成される．RTD データセットの訓練集合，検証集合，テスト集合のサンプル数はそれぞれ 10,000，1,000，1,000 である．

5.2 実験設定

最適化関数は Adam ($\beta_1 = 0.5$, $\beta_2 = 0.999$) を使用し，学習率は 0.001 に，バッチサイズは 16 に，損失関数の重みは $\lambda_{\text{ce}} = 1, \lambda_{\text{sDice}} = 1$ にした．また，Serial Encoder の serial block および Decoder における Conv-Attention Module の層数はそれぞれ [2, 2, 2, 2], [1, 1, 1, 1] にした．ただし，左から i 番目の値が，i 番目の serial block または Conv-Attention Module の層数を示す．Serial Encoder, Cross-Attentional Encoder, Decoder のそれぞれにおける Attention の Head 数はすべて 8 とした．さらに，$H_1 = W_1 = C_1 = 64$ および $(H_{i+1}, W_{i+1}, C_{i+1}) = (H_i/2, W_i/2, C_i \ \ 2)$ とする．ただし，$C_3 = 320$ および $C_4 = 512$ である．

提案手法における訓練可能パラメータ数は約 2500 万である．積和演算数は 990M である．学習は GeForce RTX 3090（メモリ 24GB）および Intel Corei9 10900K を搭載した計算機上で行った．学習は 2 時間半程度で完了した．また，1 サンプルあたりの推論に要した時間は 2ms 程度であった．各エポックで検証集合による評価を行い，損失関数の値が 5 エポック連続で改善しなかった場合に学習を停止した．

5.3 定量的結果

定量的結果を表 1 に示す．なお実験は 5 回行い，その平均値および標準偏差を示す．表 1 より，ベースラインおよび提案手法の mIoU はそれぞれ 52.5%および 55.0%であり，提案手法がベースラインを 2.5 ポイント上回った．同様に，F_1-score はそれぞれ 81.7%および 83.3%であり，提案手法がベースラインを 1.6 ポイント上回った．したがって，ベースラインと比較して提案手法の方が優れるという結果が得られた．CSCDNet [Sakurada 20] は SCD タスクへの適用に成功しているため，ベースライン手法として選択した．

評価尺度は mIoU と F_1-score を用いた．mIoU が主要尺度である．mIoU および F_1-score は RTD と密接に関連するタスクである SCD の標準的な指標であるため，これらを選択した．

5.4 定性的結果

図 3 に提案手法の成功例の定性的結果を示す．左列から順番に $\boldsymbol{x}_{\text{goal}}$, $\boldsymbol{x}_{\text{cur}}$, $\boldsymbol{y}$, ベースライン手法によって得られた $\hat{\boldsymbol{y}}$, 提案手法によって得られた $\hat{\boldsymbol{y}}$ である．

図 3(a) に関して，目標状態から現在の状態において，机上でのダンボールおよび像の移動がある成功例である．ベースライン手法では，段ボールの領域全体をマスクすることができず，さらに別の物体の後ろにある再配置対象を検出することが