Vol.2019-CVIM-217 No.30
2019/5/30

インスタンスセグメンテーションのための半教師あり学習を用いた画像合成

大羽　剛瑠[1,a)]　浮田　宗伯[1,b)]

概要：インスタンスセグメンテーションを行うほとんどの手法では，大量のラベル付きデータが必要である．しかし，大量のラベル付きデータはアノテーションコストが高いため，入手が困難である．この問題を解決する方法として，画像合成や半教師あり学習によるラベル付き学習画像の自動生成が挙げられる．画像合成では，3D モデルなどを使用することにより，手作業によるアノテーションなしにラベル付きデータを作成することができるが，様々な種類の検出対象物体の様々な模様や見え方に対応するためには，膨大な数の 3D モデルを用意する必要がある．また，半教師あり学習を用いたインスタンスセグメンテーションは，画像分類のみの半教師あり学習とは異なり，クラスだけでなく，物体の位置や形状まで自動でアノテーションしなくてはならないため難しい．本研究では，画像合成と半教師あり学習を組み合わせることで，少量の 3D モデルとインターネット上の大量の画像を用いて，人手によるアノテーションなしで学習を行う．一般的に，半教師あり学習では，追加学習データの候補に対して自動的に付与するラベルの正誤判定が困難で最重要課題である．提案手法では，半教師あり学習によりインターネット上の画像から大量の模様の情報を集めつつ，3D モデルを用いることで，正しい形状のデータのみを選択し学習することで，精度を向上させる．実験では，提案手法が従来手法よりもインスタンスセグメンテーションの精度が高いことを示した．

キーワード：インスタンスセグメンテーション，画像合成，半教師あり学習

1. はじめに

深層学習を利用することで，インスタンスセグメンテーションの精度は劇的に向上している．しかし，現在の深層学習の手法の多くは，ラベル付きデータが必要な教師あり学習であり，インスタンスセグメンテーションも同様である．インスタンスセグメンテーションは，ピクセル単位でのアノテーションが必要であるために，画像認識技術の中でも特にアノテーションコストが高く，大量のデータを用意するのが難しい．そのため，正解ラベルを自動的に収集することができる画像合成 [1] や，ラベルがないデータを利用できる半教師あり学習が注目されている [2].

画像合成を用いた方法では，物体データ（3D モデルや物体の領域のみ切り抜きされた画像）と背景画像を組み合わせることで，画像を作成する．この方法では，多様な背景で画像を作成できるのに加えて，画像中のどこにどの物体があるかは既知であるため，ラベルデータも自動的に作成することができる．しかし，物体データの収集は困難であるために，多様な物体の形状や模様を網羅した画像を作成することは難しい．

一方，半教師あり学習は，少量のラベル付きデータと大量のラベル無しデータを用いて学習を行う手法である．半教師あり学習にも様々な手法があるが，物体検出やインスタンスセグメンテーションへの応用が容易な手法として Self-Training [2] が挙げられる．Self-Training はラベル付きデータのみで教師付き学習を行った後，その学習モデルを参照することでラベル無しデータの答えを予測し，予測の確信度が高いデータのみを「予測したクラスのラベルが付いた学習データ」として，学習モデルを更新（再学習）する．この手順を繰り返すことで徐々に正しく推定できるデータを増やす．インターネット上にある大量の画像データを用いて半教師あり学習を用いれば，物体の多様な模様や形状を学習できる可能性があるが，誤った答えを推定してしまった場合，精度が低下する問題がある．クラスラベルに加えて，インスタンスセグメンテーションにおいては，形状も正しい必要がある．しかし，従来のインスタンスセグメンテーションの多くは形状の正しさに対する確信度を

[1] 豊田工業大学
Toyota Technological Institute
[a)] sd19410@toyota-ti.ac.jp
[b)] ukita@toyota-ti.ac.jp

明示的に出力しないため，上述したように「正しい予測結果のみを再学習」するためには，形状の正しさも検証する必要がある

本研究では，画像合成に不足している物体のテクスチャデータを半教師あり学習で補いつつ，3D モデルの形状を利用して，予測結果の形状の正しさを考慮することで，半教師あり学習の精度の向上させる手法を提案する．

提案手法の性能を確認する実験においては，手法の長所・短所や限界を確認するために十分な難易度の目標が必要になる．本研究では，種類も多く，見た目が類似した物体も多く（クラス間分散が小さい），形状変化も多様（クラス内分散が大きい）である「食材」を利用した．

2. 関連研究

本研究に関する研究を説明する．まず，教師あり学習によるインスタンスセグメンテーションの説明を行った後，画像合成を用いた機械学習の手法の説明を行う．その後，半教師あり学習の説明と，ラベルの入手が困難であるという問題に対処するその他の手法として弱教師あり学習の説明を行う．

2.1 インスタンスセグメンテーション

インスタンスセグメンテーションはセマンティックセグメンテーションとは異なり，同じクラスの異なる物体を識別する必要があるため，FCN [3] や U-Net [4] などのセマンティックセグメンテーションに用いられるモデルをそのまま利用することはできない．そのため，多くの手法では，Faster-RCNN [5] や SSD [6] などの物体検出の手法とセグメンテーションの技術を組み合わせることで，異なる物体を認識しつつ，セグメンテーションを行うことでインスタンスセグメンテーションを実現している．Li ら [7] の FCIS では物体の候補領域を計算した後，その領域内において前景と背景を分離するセグメンテーションと矩形領域内の物体のクラスを同時に推定し，最適化するネットワークを提案した．このようなネットワーク構造は He ら [8] の Mask-RCNN でも用いられている．Mask-RCNN では，物体の候補領域から特徴量を取り出す RoI Pooling をサブピクセルを用いることで改良し，より正確に特徴量を取り出す RoI Align を提案した．

2.2 画像合成を用いた機械学習

合成画像を用いた機械学習における問題は，合成画像と実際の画像の差である．この問題にはドメインアダプテーションやドメインランダマイゼーションによって対処されている．

画像合成におけるドメインアダプテーションの例として，Bousamlis ら [9] は，Pix2Pix [10] や cycle GAN [11] などの画像変換技術を用いることで，合成画像を実際の画像に近くなるように変換することで，画像の差に対処した．また，REN ら [12] は合成画像から得られる特徴量の分布と，実際の画像から得られる特徴量の分布が近づくように学習することで，ドメインアダプテーションを行った．これらのドメインアダプテーションは，実際の画像を用意するのが比較的簡単であるが，アノテーションが高コストである場合に有用な手法である．しかし，実際の画像を用意するのが困難であるような場合には利用できない．

ドメインランダマイゼーションは，実際の画像を用意するのが困難な場合に有用な手法である．ドメインランダマイゼーションはシミュレーター内において，物体の反射率やテクスチャなど様々な要素をランダムにし，特定の環境への過学習を防ぎ，実際の環境で精度が低下することを防ぐ手法であり，Tobin ら [13] によって提案された．また，Tremblay ら [1] はドメインランダマイゼーションを物体検出に適応し，物体検出におけるドメインランダマイゼーションの有用性を示した．

CG シミュレータを用いない画像合成手法として Cut, Paste and Learn [14] がある．Cut, Paste and Learn では，切り抜き画像を様々な背景に角度や大きさを変えつつ貼り付けることでデータ作成を行う．この手法は CG シミュレータに必要な 3D モデルが得られない場合に有用な手法である．

2.3 半教師学習と弱教師あり学習

半教師あり学習の手法の例としては上述した Self-Training [15] がある．Self-Training はシンプルであるが，予測の誤ったデータが学習データに追加されると精度が低下する．そこで，答えの推定を複数の機械学習モデルを用いて様々な観点から行うことで，精度を向上させる Co-Training [15] がある．その他の半教師あり学習の手法としては，入力画像に近い画像の答えは急激に変化することはないと仮定し，データ空間において教師なしデータ近傍でも答えが急激に変化しないような制約を加えることで，学習を行う Virtual Adversarial Training [16] や生成モデルを活用し多様体の学習を行い，その上で半教師あり学習を行う Kingma ら [17] の手法などがある．これらの手法は，Self-Training よりも高精度で半教師あり学習が行うことができるが，物体の位置や形状まで考慮しなくてはならないインスタンスセグメンテーションへの適応は難しい．半教師あり学習を物体検出へ応用した例としては，Rosenberg らの [2] がある．Rosenberg らは物体検出において，Self-Training を利用する方法を提案し，既存の物体検出手法と組み合わした時の精度の変化を調査した．

ラベル作成のコストを軽減させるためのもう一つの手法として，弱教師あり学習がある．弱教師あり学習では，実際に学習したい問題があったとき，その答えを直接与えて学習するのではなく，より簡易的な答え（弱ラベル）を与

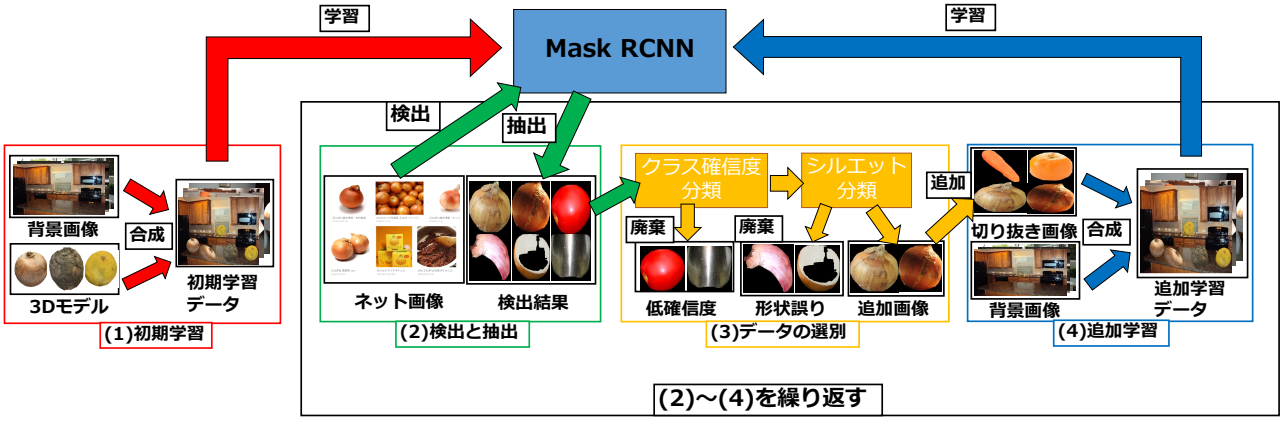

図 1 提案手法の概略図．（1）では少量の 3D モデルと背景画像を CG シミュレータを用いて組み合わせることで，半教師あり学習に必要な最初の教師ありデータを作成する．その後，（2）でインターネット上の画像の検出と抽出を行う．抽出後は，（3）にあるように，抽出したデータの中で，形状（シルエット）と種類の両方の観点で，正しいデータのみを選別する．（4）では選別したデータと背景を組み合わせることで，追加の学習データを作成し，追加の学習を行う．（2）～（4）の手順を繰り返すことで，徐々に学習データを増やしていく．

えて学習を行う手法である．インスタンスセグメンテーションでは二種類の弱教師あり学習があり，一つは物体のクラスと矩形領域が与えられる弱教師あり学習である．Zhao ら [18] はグラフベースの手法である Graph Cut を利用することで，矩形領域内で，ピクセル単位の位置予測を行いインスタンスセグメンテーションを行った．もう一つの弱教師あり学習では，物体のクラスのみを与えて学習を行う．Zhou ら [19] や Zhang ら [20] は，まずクラスラベルを用いてクラス分類を行う機械学習モデルの学習を行う．その後，クラス分類の際に，機械学習モデルが画像中のどこに注目しているかを求め，その注目領域を利用することで，物体のピクセル単位での位置を予測を行い，それを答えとすることで，物体の位置とクラスの学習を行う．これらの弱教師あり学習の手法は，完全な状態のラベルなしに学習が行えるが，今回は画像合成により，完全なラベルがついたデータを利用できるため半教師あり学習を用いる．

3. 半教師あり学習と画像合成を組み合わせた提案手法の流れ

提案手法の手順は以下であり，概略図を図 1 に示す．

(1) CG シミュレータを用いた最初の学習画像の作成と学習（初期学習）
(2) インターネット画像から食材の検出および抽出（検出と抽出）
(3) 抽出した画像の選別（データの選別）
(4) 画像合成を用いた追加学習画像の作成と追加学習（追加学習）
(5) 2～4 を繰り返す

以下では，それぞれの手順について述べる．

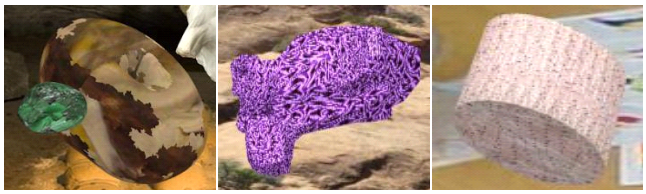

図 2 偽物体の例．左は玉ねぎの形状データにランダムなテクスチャを貼り付けたデータで，中，右は blender で作成できる形状にランダムなテクスチャを貼り付けたデータである．

3.1 CG シミュレータを用いた最初の学習画像の作成と学習

まず，半教師あり学習に必要なラベル付き画像データを，ドメインランダマイゼーションを用いた画像合成により作成する．ドメインランダマイゼーションは Tobin ら [13] や Tremblay ら [1] を参考にし，CG シミュレータ内の以下の要素をランダムにする．

- 光源の個数と位置
- 検出対象物体の個数と位置
- 背景画像
- カメラの位置と向き
- 偽物体の位置，個数，テクスチャ，形状

偽物体の例を図 2 に示す．偽物体は [1] で利用されており，偽物体を検出しないように学習させることで，誤検出を減らす．偽物体の形状は今回使用する CG シミュレータである blender で簡単に作成できる形状（球，円柱，円錐，ドーナッツ型，長方体など）に加えて，検出対象物体の形状からランダムに選択される．

この条件で画像を作成した後，作成した画像で最初の学習を行う（図 1 の（1）初期学習）．インスタンスセグメンテーションを行うアルゴリズムとしては Mask-RCNN [8]

図 3 追加候補画像の抽出の仕方．検索キーワードを記憶しておき，Mask-RCNN の予測結果とキーワードが一致したときのみ，追加候補画像とする．

を用いた．学習時には COCO データセット [21] で学習したモデルを初期値とした．

3.2 インターネット画像から食材の検出および抽出

半教師あり学習に用いるための，教師なしデータの収集と，学習済みモデルを用いた検出と抽出を行う（図 1 の（2）検出と抽出）．まずインターネットを用いて画像のキーワード検索を行い大量の教師なし画像データを収集する．収集された各教師なし画像データにおいて，図 3 にあるようにキーワードごとに学習済みモデルで物体を検出する．検出されたクラスと検索キーワードが一致した場合，検出された領域を抽出し，追加候補画像とする．

3.3 抽出した画像の選別

確実に正しい追加候補画像のみを学習データに追加するために，追加候補画像のクラスによる分類とシルエットによる分類を行う（図 1 の（3）データの選別）．

クラスによる分類では，半教師あり学習の手法である Self-Training [15] に基づき，クラス確信度を用いる（図 1 の（3）左のクラス確信度分類）．クラス確信度とは，Mask-RCNN が出力するどのクラスに属するかの確率である．クラス確信度が閾値以下のデータは追加候補画像から除去する．

形状の判断は，用意した 3D モデルのシルエットと抽出された画像のシルエットを比較することで行う（図 1 の（3）右のシルエット分類）．ここでは複数の物体を一つの物体と誤って識別しているデータ（図 4 左列）や誤検出の危険性を増やすデータ（図 4 右列）など形状的に正しくないデータを除去する．その方法として，Auto Encoder（AE）を利用した異常検知を利用する．AE による異常検知は以下の手順で行う．

まず，AE の学習に使うデータを作成する．様々な角度から見た 3D モデルのシルエットを CG シミュレータ上で作成する．作成した画像を教師データとして，入力画像と AE による復元画像の平均二乗誤差を最小にするように AE を学習させる．これをクラスごとに行う．ここまでが AE の学習の手順である．

次に学習済み AE を用いて，追加候補画像の形状が正しいかの判定を行う．Mask-RCNN によって出力された物体のマスク画像を追加候補画像のシルエットとし，対応するクラスの学習済み AE に入力する．AE により復元された画像と元のシルエット画像の差分を取ることで，形状の正しさを測る．差分の値が小さいほど，その形状が正しいことを示しているので，差分値がある閾値を超えたら異常な形状と判断し，追加候補画像から除去する．

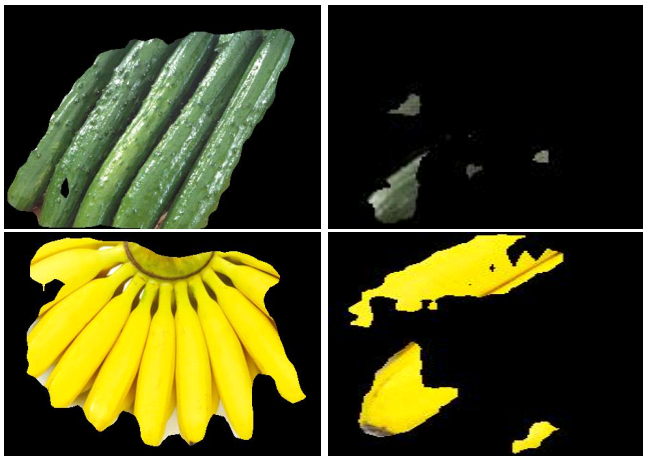

図 4 誤った形状の例．左の列は複数の物体をまとめて一つと検出している例で，右列は一部しかセグメンテーションに成功しておらず，学習データに加えると誤検出が増える危険性があるデータの例である．

閾値は以下の手順で計算する．正常な形状における復元誤差の平均と分散を計算するために，CG シルエット画像における復元誤差の平均 μ と分散 σ をクラスごとに計算する．この復元誤差の分布がガウス分布に従うと仮定したとき，$\mu + 3\sigma$ を閾値とすると，CG シルエット画像の約 99%が正常と識別されるため，$\mu + 3\sigma$ を閾値とした．

また，AE を利用するためには画像のサイズが一定である必要がある．画像のサイズを揃えるためにリサイズを行うと形状が変化してしまうので，前処理として画像サイズが長辺×長辺になるようにパディングを行う．

3.4 画像合成を用いた追加学習画像作成

クラスと形状の両方で正しいと判断された追加候補画像を追加画像として，追加の教師データを作成する（図 1 の（4）追加学習）．追加画像は 3D モデルではなく切り抜き画像のため，追加の画像作成は以下の手順で行う．

（1）追加画像からランダムに複数枚選択する
（2）選択した画像をランダムにリサイズ，回転する
（3）背景をランダムに選択する
（4）追加画像が一定以上重ならないように貼り付ける位置をランダムに選択する
（5）追加画像を背景画像に貼り付ける

追加画像の枚数はクラスによって異なる．そのため，追加画像を完全にランダムに選択するとクラス間での画像枚数に差が生じ，追加画像の少ないクラスの学習が困難にな

る．この問題を防ぐために，クラス間のサンプリング回数が均一になるように追加画像を選択した．また追加画像同士の重なりが大きい場合，その物体を識別するのは困難で誤検出の増加に繋がる．そのため，本研究では追加画像の面積の 50%以上の重なりが生じた場合，再度貼り付ける位置をランダムに選択する．この条件で作成したデータを用いて，Mask-RCNN の再学習（ファインチューニング）を行う．その後，再学習した Mask-RCNN を用いて，再び（2）～（4）の処理を繰り返すことで徐々に正しく検出できるデータを増やす．バリデーションデータかテストデータで mean Average Precision（mAP）を測り，mAP が低下したところで反復を終了する．

4. 実験内容および結果

4.1 実験条件

実験時に使用したデータやモデルについて述べる．3.1 節や 3.3 節で用いる 3D モデルは AutoDesk 社の ReCap [22] を用いて作成した．作成した 3D モデルは全 15 種の食材であり，表 1 に種類を示す．各食材の 3D モデルはそれぞれのクラスに 1 つのみ作成した．また，3.1 節における背景画像と偽物体のテクスチャはそれぞれ，Place365 データセット [23]，dtd データセット [24] を用いた．これらのデータを用いて，初期学習用のデータを作成し，Mask-RCNN[8] の初期学習を行った．半教師あり学習における教師なしデータであるインターネット画像データは Google のキーワード検索で収集した．キーワード検索は日本語と英語の両方で行った．各物体ごとの収集した画像枚数を表 1 に示す．

表 1 google からキーワード検索を用いて収集したデータの数

apple	avocado	banana	broccoli	carrot
834	1564	1591	1377	1497
cucumber	garlic	kiwi	mushroom	onion
1489	1593	1517	1651	1592
orange	potato	sweet corn	sweet potato	taro
1211	1318	1594	1546	1453

3.4 節の追加学習画像作成における背景画像には，同様に Place365 データセットを用いた．また，テストデータは 200 枚の画像を実際に撮影し作成した．形状の異常検知（3.3 節）における AE には [25] の CAE と同じモデルを用いた．

4.2 提案手法と従来法のインスタンスセグメンテーション精度の比較

テストデータを用いた実験結果を図 5 に示す．評価指標には IoU の値が 50%以上を正解とする mAP@.50（mean Average Precision）を用いた．手法は画像合成のみの手法（図 5 の画像合成），形状を考慮しない Self-Training を用いた手法（図 5 の Self-Training），提案手法の 3 つで比較を

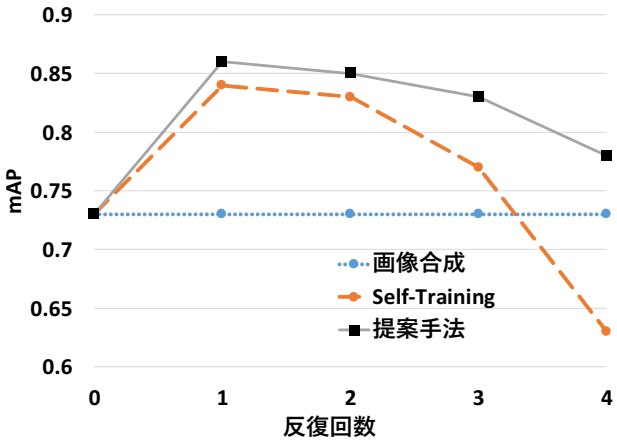

図 5 提案手法とその他の手法の反復回数による mAP の変化．

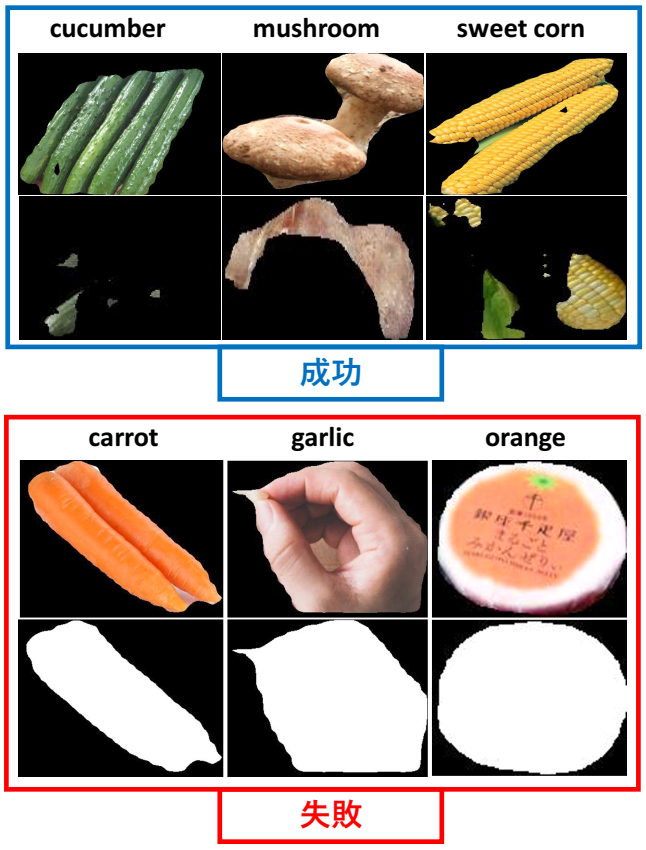

図 6 AE を用いた異常検知の結果．上の青枠は成功した例であり，下の赤枠は失敗した例である．赤枠の上段は画像で下段はそのマスクである．

を行っている．横軸は反復回数であり，縦軸が mAP である．ただし画像合成手法は反復を行っていないため変化しない一定値である．Self-Trainig と提案手法では一回の反復ごとに 3.4 節の手法を用いて合成画像を 5000 枚作成し，学習を行っている．Self-Trainig と提案手法の反復回数は実際に使用する際には，mAP が下がった時点で反復を終了するが，今回は実験的に mAP が下がっても反復を続けた．図 5 は提案手法がクラスのみを考慮した Self-Training よりも mAP が 2%高く，さらに反復を続けたときの mAP

表 2 各手法によるクラス確信度が 98%以上のデータの数

Class	画像合成	Self-Training 1	Self-Training 2	Self-Training 3	提案手法 1	提案手法 2	提案手法 3
apple	1058	959	1038	694	996	**1154**	978
avocado	169	156	145	159	148	222	**271**
banana	835	733	778	756	**1051**	785	758
broccoli	1280	1008	1127	1042	1251	**1293**	1191
carrot	1364	1364	821	683	**1600**	1010	846
cucumber	245	254	348	428	524	**546**	518
garlic	434	396	370	267	**475**	421	347
kiwi	223	352	351	252	404	**509**	445
mushroom	241	191	644	448	377	**774**	603
onion	49	32	**250**	154	27	146	210
orange	976	1305	1432	999	1651	**1765**	1400
potato	129	285	397	252	304	**501**	347
sweet corn	543	580	**906**	763	725	891	766
sweet potato	43	81	201	168	65	220	**250**
taro	48	126	168	183	113	264	**292**

の低下が小さいことを示している．これはクラスによるSelf-Training だけでは誤ったデータの混入を防ぐことができないためだと考えられる．

テストデータを用いた検出結果の例を図 7 に示す．図 7 の 1 行目と 2 行目の画像が示すように，Self-Training と提案手法では新たなテクスチャを獲得することで，画像合成のみでは検出できなかった物体の検出に成功している．また，図 7 の 3 行目の画像に示すように Self-Training では検出できなかった物体を提案手法では検出できている例もある．これは形状を考慮しなかったことで追加された誤ったクラスの画像が，正しいデータのクラス確信度を下げてしまい，結果的に一部の正しいデータが追加されなかった可能性や，図 4 の右列に示すような画像が学習を難化させ正しい学習ができなかったなどの可能性が考えれる．また，図 7 の 4 行目の画像に示すように，提案手法は Self-Training に比べて誤検出の低下に成功している．これは，上述したように形状を考慮することで誤ったテクスチャが学習データに追加されることを防ぐことができたからだと考えられる．しかし，図 7 の 5 行目の画像に示すように提案手法では防ぐことができなかった誤検出の例もある．

4.3 反復によるデータの増加数と異常検知の結果

半教師あり学習によりどれだけテクスチャを増やすことができたかを示す．インターネット上の画像には答えがないため，どれだけクラスと形状の双方が正しいデータを学習データに追加できたのかを測ることはできない．そのため，ここでは正しさを考慮せず，どれだけデータが増えたかのみを評価する．まず，提案手法と Self-Trainig の比較を行うために，クラス確信度が閾値を超えたデータの数を比較する．閾値は実験時と同じく 98%とした．その結果を表 2 に示す．表 2 は収集したインターネット画像で，それぞれの手法及び反復回数で学習されたモデルを用いて検出されたデータの内，クラス確信度が 98%以上のデータの数である．表 2 の Self-Training 1 は手法が Self-Training で反復回数が 1 回であることを示している．また太字は，各クラスで最も数が多かったものを示している．表 2 が示すように，提案手法のほうが，Self-Training よりも高確信度のデータが多い．これは，Self-Training は形状やクラスの誤ったデータにより学習が難化し，正しくデータでさえ低い確信度で検出してしまうためだと考えられる．Self-Training と画像合成の結果を比較してみると，反復を繰り返しても画像合成よりも高確信度のデータ数が少ないクラスが 5 クラスある．これらのクラスは半教師あり学習に失敗している．一方，画像合成と提案手法を比較すると反復を繰り返すことで，すべてのクラスで一度は高確信度のデータ数が増加している．これは，誤ったデータが学習データに追加されることを防ぐことで学習が容易になり，高確信度で検出できるデータが増えたと考えられる．

次に，形状とクラスの双方が正しいと判定されたデータの数（提案手法で追加されたデータの数）を表 3 に示す．表 3 は収集したインターネット画像で，それぞれの手法及び反復回数で学習されたモデルを用いて検出されたデータの内，クラス確信度が 98%以上で，形状も正しいと判断されたデータの数である．表 3 からは，1 つのクラスを除き全てのクラスでデータが一度は増加していることが分かる．この結果から，提案手法では目的通りテクスチャの増加を行えたことが分かる．しかし，反復を重ねることでデータが単調に増加するのではなく，減少してしまう例があることも表 3 から分かる．これは，形状的には正しいがクラスが誤っている物体をクラス確信度で除去することができずに学習データに追加されてしまったことや，形状の異常検知に失敗してしまっているなどの理由が考えられる．

次に形状異常検知の結果の例を図 6 に示す．図 6 の上段青枠は正しく異常と判断されたデータであり，下段赤枠は

誤って正常と判断されたデータである．図 6 の青枠に示すような明らかに形状が異常なデータに関しては，除去することに成功している．しかし，赤枠にあるような画像はマスク画像からでは異常と判断するのは難しく，除去に失敗している．このようなデータを除去するためには，クラス確信度によるデータ除去の精度を上げる必要がある．

表 3 クラスと形状が共に正しいと判断されたデータの数

Class	画像合成	提案手法 1	提案手法 2	提案手法 3
apple	418	380	**455**	422
avocado	63	48	83	**101**
banana	176	245	247	**258**
broccoli	739	705	**817**	799
carrot	**215**	180	126	99
cucumber	25	58	**77**	72
garlic	233	**264**	238	203
kiwi	102	152	164	**167**
mushroom	85	117	**299**	275
onion	19	11	53	**68**
orange	436	647	**679**	509
potato	108	230	**403**	262
sweet corn	87	85	**100**	83
sweet potato	7	15	75	**81**
taro	19	52	**143**	142

5. まとめ

本研究では，画像合成と半教師あり学習を組み合わせ，少量の 3D モデルから人手によるアノテーションなしでインスタンスセグメンテーションの学習を行った．提案手法においては，半教師あり学習の手法である Self-Training において，クラスだけでなく，形状も考慮することで精度を向上させた．

実験結果は提案手法が従来手法の Self-Training や画像合成のみの手法に比べて，精度が高いことを示しており，反復回数の増加による精度の低下が少ないことも示した．しかし，反復を重ねるにつれて誤検出が増加することや，追加される画像が減るなどの問題がある．また 3D モデル形状との比較では，形状のばらつきが大きいデータへの適用は難しい．特に食材は調理によって形状の変化が発生するため形状のばらつきが大きいため，改善する必要がある．

今後はこれらの問題を解決するために，形状とテクスチャの両方を考慮した異常検知への拡張や，動画を用いることにより，食材の形状の変化を追跡し新たな形状データ獲得することで，形状変化へ対処するなどを検討していく．将来的には，この食材のインスタンスセグメンテーションを，料理の認識・解析や調理の自動化システムなどの研究に発展させていきたい．

参考文献

[1] Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S. and Birchfield, S.: Training Deep Networks With Synthetic Data: Bridging the Reality Gap by Domain Randomization, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 969–977 (2018).

[2] Rosenberg, C., Hebert, M. and Schneiderman, H.: Semi-supervised self-training of object detection models.

[3] Long, J., Shelhamer, E. and Darrell, T.: Fully convolutional networks for semantic segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440 (2015).

[4] Ronneberger, O., Fischer, P. and Brox, T.: U-net: Convolutional networks for biomedical image segmentation, *International Conference on Medical image computing and computer-assisted intervention*, Springer, pp. 234–241 (2015).

[5] Ren, S., He, K., Girshick, R. and Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems*, pp. 91–99 (2015).

[6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A. C.: Ssd: Single shot multibox detector, *European conference on computer vision*, Springer, pp. 21–37 (2016).

[7] Li, Y., Qi, H., Dai, J., Ji, X. and Wei, Y.: Fully convolutional instance-aware semantic segmentation, *arXiv preprint arXiv:1611.07709* (2016).

[8] He, K., Gkioxari, G., Dollár, P. and Girshick, R.: Mask r-cnn, *2017 IEEE International Conference on Computer Vision*, IEEE, pp. 2980–2988 (2017).

[9] Bousmalis, K., Silberman, N., Dohan, D., Erhan, D. and Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks, *The IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, No. 2, p. 7 (2017).

[10] Isola, P., Zhu, J.-Y., Zhou, T. and Efros, A. A.: Image-to-image translation with conditional adversarial networks, *arXiv preprint* (2017).

[11] Zhu, J.-Y., Park, T., Isola, P. and Efros, A. A.: Unpaired image-to-image translation using cycle-consistent adversarial networks, *arXiv preprint* (2017).

[12] Ren, Z. and Jae Lee, Y.: Cross-domain self-supervised multi-task feature learning using synthetic imagery, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 762–771 (2018).

[13] Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W. and Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world, *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, pp. 23–30 (2017).

[14] Dwibedi, D., Misra, I. and Hebert, M.: Cut, paste and learn: Surprisingly easy synthesis for instance detection, *The IEEE international conference on computer vision* (2017).

[15] Zhu, X.: Semi-supervised learning literature survey, *Computer Science, University of Wisconsin-Madison*, Vol. 2, No. 3, p. 4 (2006).

[16] Miyato, T., Maeda, S.-i., Ishii, S. and Koyama, M.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning, *IEEE transactions on pattern analysis and machine intelligence* (2018).

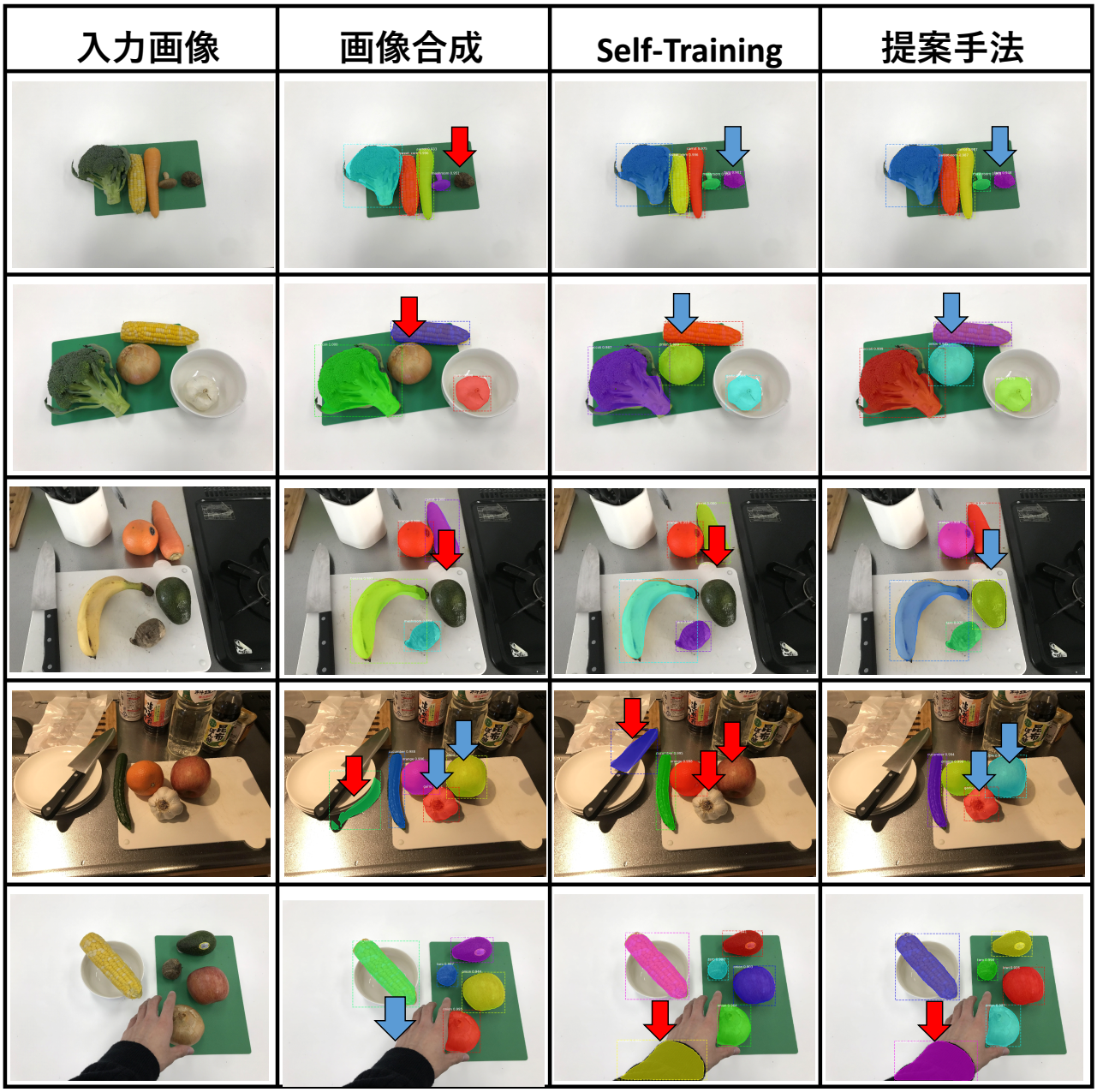

図 7 各種法のテストデータにおけるインスタンスセグメンテーション結果の例．青の矢印は成功している例であり，赤の矢印は失敗している例である．

[17] Kingma, D. P., Mohamed, S., Rezende, D. J. and Welling, M.: Semi-supervised learning with deep generative models, *Advances in neural information processing systems*, pp. 3581–3589 (2014).
[18] Zhao, X., Liang, S. and Wei, Y.: Pseudo Mask Augmented Object Detection.
[19] Zhou, Y., Zhu, Y., Ye, Q., Qiu, Q. and Jiao, J.: Weakly Supervised Instance Segmentation using Class Peak Response, *arXiv preprint arXiv:1804.00880* (2018).
[20] Zhang, X., Wei, Y., Kang, G., Yang, Y. and Huang, T.: Self-produced guidance for weakly-supervised object localization, *Proceedings of the European Conference on Computer Vision*, pp. 597–613 (2018).
[21] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L.: Microsoft coco: Common objects in context, *European conference on computer vision*, Springer, pp. 740–755 (2014).
[22] AUTODESK: AUTODESK ReCap, https://www.autodesk.co.jp/products/recap/overview.
[23] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A. and Torralba, A.: Places: A 10 million Image Database for Scene Recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).
[24] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A.: Describing Textures in the Wild, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014).
[25] Guo, X., Liu, X., Zhu, E. and Yin, J.: Deep clustering with convolutional autoencoders, *International Conference on Neural Information Processing*, Springer, pp. 373–382 (2017).