

# 画像合成による物体検出モデルの効率的な環境適応技術

山本 健生, 平井 佐和, 濱走 秀人, 岡本 大和

第三次 AI ブームにおける技術の飛躍的進化に伴い、従来は人が担っていた業務を AI によって自動化し、労働力不足の課題を解決する期待が高まっている。例えば社会領域では安心安全のために防犯カメラ等の映像を人手で確認するモニタリング業務の自動化が期待され、これは機械学習手法によって検出したい対象をラベル付けした大量の画像データをモデルに学習させることで実現できる。

しかしながら、機械学習モデルには学習データと異なる環境に対して性能低下するという問題がある。環境ごとに学習データを準備して追加学習させることで性能を改善できるが、学習用の画像収集と人手で正解ラベルを付与する作業には膨大なコストがかかる。特に社会領域では撮影地点やカメラ設置条件など多種多様な環境相違が想定されるため、環境ごとに追加学習を行うのは現実的ではない。

本稿では、機械学習モデルを様々な環境へ効率的に導入するため、学習用の画像と正解ラベルを画像合成により自動生成する手法を提案する。実験は高速道路や交差点における車両検出を対象に行った。自動生成した画像では従来の学習データを準備して追加学習した場合の約 8 割の性能改善効果を、5.5%の作業コストで得られた。これによりモニタリングの自動化のさらなる推進が期待できる。

## Efficient Scene Adaptation Technology for Object Detection Using Auto Image Synthesis

YAMAMOTO Yoshiki, HIRAI Sawa, HAMABASHIRI Hideto and OKAMOTO Yamato

Recently, because of the growing need for automation of monitoring, machine learning is widely used in the social systems domain. However, the problem of machine learning is that the pre-learned model will work in poor performance when the environment changes. Especially in social systems domain, there are various environmental changes such as place, camera angle, etc., and additional learning through scene adaptation is essential to achieve the required accuracy.

Therefore, we tried to develop a method to make more efficient scene adaptation. Until now, it took a lot of cost to collect images of the new scene and enter the ground truth. In this paper, we investigate a method to automatically generate images and ground truth labels for training by image synthesis. The experiment was conducted for vehicle detection at highways and intersections. As a result, we were able to generate the training dataset with less cost while achieving accuracy close to that of manual input. This method can be expected to further automation.

### 1. まえがき

社会領域では労働力不足解決のため、モニタリングの自動化のニーズが高まっている。特に、白杖検知・車いす検知・線路転落検知や、車番認証・交通量調査・逆走検知などの自動化ニーズが顕在化している<sup>1)</sup>。モニタリングには様々なシーン・用途が考えられ、求められる機能も物体検

出、行動追跡、イベント検知など様々である。本稿では主流である固定カメラを想定して、モニタリングの最も主要な機能である、物体検出の実現をターゲットとする。

物体検出は検出したい対象をラベル付けした大量のデータをモデルに学習させることで実現が期待できるが、一般的に、機械学習により構築したモデルは、学習データと異なる環境で取得したデータに対して性能低下する問題が知られている。これは、環境の相違によってデータ分布が異

Contact : YAMAMOTO Yoshiki yoshiki.yamamoto@omron.com

なることに起因するとされている<sup>2)</sup>。例えば、本稿が扱う固定カメラによる物体検出モデルでは、撮影地点やカメラ設置条件が学習データと異なる場合に検出精度が落ちる問題がある。このとき、モデルを導入する環境ごとに学習データを準備し、追加学習させることで性能を改善できる。しかし、物体検出モデルにおける正解ラベル入力、人手で画像中の物体領域を1つずつ確認して座標入力するため膨大なコストがかかる。特に撮影地点やカメラ設置条件など多種多様な環境相違が想定される社会領域では環境ごとに追加学習を行うのは現実的ではない。よって、効率的に追加学習をしてモデルを環境に適応させる手法が求められる。

## 2. 関連研究

モデルを効率的に環境適応させるために、前景と背景をそれぞれ準備して、組み合わせて学習データを生成する手法が提案されている。T. Hodan らは実画像データの様々な背景に対して、CGで作成した3Dモデルオブジェクトを光の当たり方・重なり方を考慮して貼りつける手法<sup>3)</sup>を提案した。G. Georgakis らは自律走行するロボットに搭載したカメラを対象に、様々な場所で撮影した物体を前景、モデルを適応させたい環境で撮影した画像を背景として、背景の奥行き情報を考慮しながら貼り付ける手法<sup>4)</sup>を提案した。

これらの手法は、モデルを適応させたい環境とは異なる環境からの前景取得を暗に前提としている。そして、前景と背景を異なる環境から取得する想定のため、貼りつけ境界や陰影など違和感なく貼りつけるための大量の計算処理を必要とする。例えば、T. Hodan らの手法は400ノードのCPUクラスタを利用し、640×480画素の画像と正解ラベルの生成に1枚平均120秒を要する。

そこでわれわれは背景と前景の両方を同一の環境、すなわち、モデルを適応したい環境から取得する前提で、前景を貼り付ける座標とサイズを指定するだけの簡単な処理で高速に画像と正解ラベルを生成する手法を提案する。これにより人手による正解ラベル入力を行うことなく、かつ大量の計算処理も必要もせずに学習データを生成し、環境ごとに適応したモデルを追加学習させることができる。

## 3. 提案手法について

提案手法は大きく分けて3つの処理で構成される。最初に、モデルを適応させたい現場の画像から背景と前景をそれぞれ抽出し、データベースに保存する。次に、そのデータベースの背景と前景を様々なパターンで組み合わせて大量の合成画像と正解ラベルを生成する。最後に、生成した合成画像と正解ラベルを用いて物体検出モデルを追加学習させる。(処理フローについて図1参照)

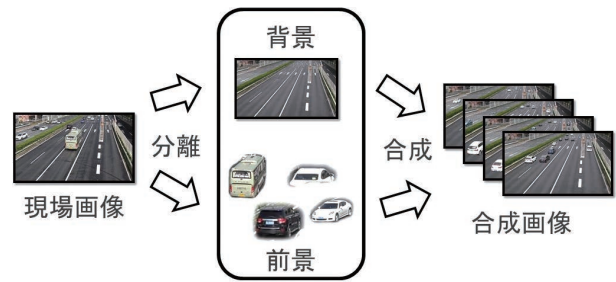


図1 提案手法の処理フロー

### 3.1 物体検出のアルゴリズム

物体検出のアルゴリズムには、Faster R-CNN<sup>5)</sup> や SSD<sup>6)</sup>、YOLO<sup>7)</sup> など様々な手法が提案されている。この中でも、SSDは他の手法に比べて処理が軽量かつ高速である特徴が挙げられる。社会領域では移動速度が速い自動車や不規則に進行方向が変化する人を検出するため、高いフレームレートでの処理が求められる。また、通信環境が整備されていない状況やロボットなど即時性が求められるアプリでは、サーバーと通信せずに現場のエッジ端末など限られた計算リソースで処理できるように、モデルの軽量化が求められている。以上の観点から、社会領域の利用に適したSSDを物体検出アルゴリズムに採用して、提案手法を検証する。

### 3.2 背景・前景の抽出方法

背景と前景の抽出方法には人手で切り出す、背景差分法によって抽出する、学習済みの機械学習モデルで抽出するなど、様々な方法が考えられる。

本稿では、撮影画像の平均画像を生成して、これを背景画像とする。前景は効率化と正確性の観点から、インスタンスセグメンテーションに広く用いられるMask-RCNN<sup>8)</sup>を用いて抽出する。Mask-RCNNは、どの環境にも特化させていない一般公開されている標準モデルを用いて、前景領域と判定された部分を抽出する。なお、前景を貼りつける際に背景との境界部分の親和性を考え、境界部分の透過処理とスムージング処理を施す。

## 4. 前景の貼り付け方の検証

### 4.1. 実験データ・評価方法

既設の防犯カメラを想定して、高速道路の上方から固定カメラで撮影した画像に対し、物体検出モデルによって車両を検出する実験を行う。なお、実験で利用する画像は道路管理者に許可を得たうえで撮影し、社内規定に則り利用・管理を行った。(画像管理番号: G190035-000)



図2 実験現場の撮影映像（高速道路）

図2の地点で30分間の映像を撮影し、これを学習データ生成用の素材とする。そして、一定時間経過後に再び30分撮影し、1秒ごとに切り出した1800枚を評価用データとする。

合成用の背景を取得するために、30分の映像を1秒ごとに切り出した1800枚から平均画像を50枚生成して、データベースに保存する。これは、1枚のみ生成すると、生成した1枚に前景が含まれてしまうなどのリスクを避けるためである。また、前景は1秒ごとに抽出すると同じ車両が何回も登場するため、映像から4秒ごとに切り出した450枚からMask-RCNNで抽出された前景すべてをデータベースに保存する。その後、データベースに含まれている背景と前景を組み合わせて、1800枚の画像と正解ラベルを生成し、これを約350回ずつモデルに学習させて性能評価を行う。

性能評価にはAP (Average Precision) を用いる。これは、物体検出で広く用いられるPASCAL VOC<sup>9)</sup> やMS COCO<sup>10)</sup> データセットでも用いられる指標で、0から1の間の値を取り、1に近いほど誤検出や見逃しが少ないことを意味する。IoUと呼ばれる正解判定の閾値を本実験では0.5に固定し、101pointの補間APで計算されるAP<sub>0.5</sub>を使用した。これは再現率 (Recall) が0, 0.01, ..., 0.99, 1.0のときの適合率 (Precision) の平均である。

$$AP_{0.5} = \frac{1}{101} (Pre.(0) + Pre.(0.01) + \dots + Pre.(1.00))$$

なお、本実験で扱う道路画像は、物体検出したい領域を捉えられるよう撮影したが、目的外の領域も映り込んでいる。例えば、今回は図2中央の高速道路エリアで性能評価を行う予定であったが、両脇の一般道が映りこんでいる。本実験では、このような目的外領域は物体検出の対象外として、領域を制限してAPを算出する。(これを、AP (Mask) と表記する)

#### 4.2 前景を貼りつけるときの制約

機械学習における一般論として、学習データの分布はモデルを導入する現場で実際に観測されるデータに近いことが望ましい。現場で観測されるデータに比べて学習データ

のバリエーションが少ない場合は、未学習のパターンが出現したときに正しく検出できない。一方で、実際に現場で観測しえないデータ、例えば空を飛ぶ車両や実在しない巨大サイズの車両が学習データに含まれていると、誤検出が増え性能劣化を引き起こす。以上を踏まえて、提案手法では生成した合成画像が実際に現場で観測されるデータの分布へと近づくよう、現場知見に基づく3つの制約を画像生成時に設けた。

制約1. 車両のサイズと貼りつける領域を制限する

制約2. 画像1枚の前景の平均数を、合成画像と現場のデータで一致させる

制約3. 重なった車両を一定確率で発生させる

表1にそれぞれの制約の有効性を網羅的に検証した結果をまとめる。なお、実験1-4,2-3,3-1は同じ条件である。✓は適用していること、×は制約を適用していないことを示す。

表1 貼りつけ時の制約と、合成画像学習モデルの精度

実験番号	サイズの制約	位置の制約	貼りつけ平均台数(範囲)	重なり平均台数	AP (Mask) %
1-1	×	×	10 (1 ~ 19)	0	65.82
1-2	×	✓	10 (1 ~ 19)	0	66.37
1-3	✓	×	10 (1 ~ 19)	0	67.86
1-4	✓	✓	10 (1 ~ 19)	0	<u>69.60</u>
2-1	✓	✓	1 (1 ~ 1)	0	58.90
2-2	✓	✓	5 (1 ~ 9)	0	65.87
2-3	✓	✓	10 (1 ~ 19)	0	<u>69.60</u>
2-4	✓	✓	20 (11 ~ 29)	0	68.58
3-1	✓	✓	10 (1 ~ 19)	0%	69.60
3-2	✓	✓	10 (1 ~ 19)	20%	85.40
3-3	✓	✓	10 (1 ~ 19)	40%	87.52
3-4	✓	✓	10 (1 ~ 19)	60%	<u>89.24</u>

次節から3つの制約について仮説を述べ、検証結果を示す。

#### 4.3 位置・サイズの制約【実験1】

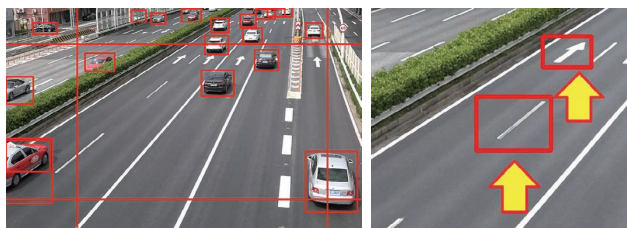
前景を背景に貼りつけるときの位置とサイズの制約を考える。前景を抽出時と同じ位置に貼りつけるだけでは、合成画像のバリエーションが少なくなってしまう。そこで、前景を抽出時と異なる位置にも貼りつけ、様々なパターンでの合成画像を生成する。このとき、実際の現場では、壁や分離帯上には車両が出現しないことから、車両の出現位置

に環境特有の制約がある。SSDは原理上、前景が出現しやすい位置を学習する傾向があるため、道路上のみに貼りつけるような位置の制約は有効に働くと期待できる。

また、カメラ画像では遠近法により手前の車両は大きく、奥の車両は小さく映るため、抽出時と異なる座標に前景を貼りつける際にはサイズ比率を計算して、手前に貼りつける場合は拡大し、奥に貼りつける場合は縮小する。これにより、いっそう合成画像が実際に観測される画像に近づくと期待できる。ただし、極端に拡大縮小すると前景画像が歪むため、拡大率と縮小率に上限と下限を設け、その範囲内で収まる位置にのみ前景を貼りつける。

これらの制約の有効性を検証するため、サイズの制約と位置の制約がある場合とない場合を4つのパターンで比較検証した。座標の制約がない場合は前景をランダムに貼り付け、制約がある場合は事前指定した道路領域内の座標にだけ貼り付ける。サイズの制約がない場合は異なる位置に貼りつけてもサイズを変更せず、制約ありの場合は75%~133%の範囲内で貼りつけられる範囲にのみ貼りつけ、その際サイズを変更した。

実験結果より、両方の制約を採用した場合が最も学習後の検出精度が高かった。サイズの制約がない実験1-1の結果(図3(a))では画面全体に検出枠を出してしまうことがある。これは、貼りつける際に大きさの変化を考慮せず貼りつけることで、その結果、極端に大きい車が発生してしまい、それを学習するためである。逆に実験1-4(図3(b))では白線を誤検出している傾向にある。



(a) 実験1-1 (b) 実験1-4

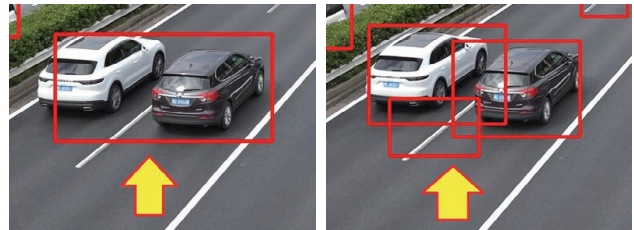
図3 実験1の結果

#### 4.4 貼りつける台数【実験2】

画像には車両が1台だけのときもあれば複数台同時に映り込む場合もある。合成画像の前景数が極端に少ない画像や多い画像ばかりに偏ると、現場のデータと乖離する可能性がある。そこで、1枚の画像に貼りつける前景数の平均を現場画像の前景数の平均にすることで、合成画像を実際に観測される画像に類似すると期待できる。前景数を調整することの有効性を確認するために、実験データの台数平均である10台/枚の場合、極端に少ない場合として1台/枚の場合、5台/枚の場合、そして極端に多い例として20台/枚の場合の計4パターンで学習後の性能を比較した。なお4つのパターンとも、実験1で効果が認められた「サイズ

の変更」及び「貼りつけ位置の制約」を採用している。

実験結果としては、1台の場合が極端に追加学習後の性能が低く、最も性能が高いのは10台の場合であった。1台の場合は近接している複数の車両をまとめて1台として検出する傾向にある。一方で、20台の場合は背景の誤検出や、1台の車両に対して複数の検出枠がつくなど、過剰に検出をする傾向がある。



(a) 実験2-1 (b) 実験2-4

図4 実験2の結果

#### 4.5 重なった車両の生成【実験3】

物体検出では物体の一部が隠れて観測できない場合に、うまく検出できないという課題がある。例えば、物体同士が重なった場合は、手前の物体は検出できていても、奥の物体は欠けて見えるため検出できなかったり、まとめて1つの物体として検出したりする。そこで、この隠れの課題に対処するために、合成する際に重なった車両をあえて生成し、学習させる。これにより、隠れた車両へのロバスト性が向上できると期待できる。重なった車両の生成方針は、4.3章の方針に従って、拡大縮小しながら座標指定して車両を貼り付けるときに、一定数の車両を他の車両と重なる座標に貼り付けることで実現する。このとき、カメラから手前の車両が奥の車両を隠すようにする(図5)。

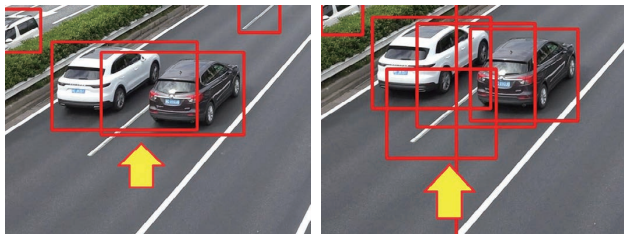


図5 拡大縮小・移動して重ねた車両

重なった車両を生成する有効性を確認するために、車両を重ねて貼りつける割合を0%、20%、40%、60%の4パターン設定し、それぞれの学習後の性能を比較した。このとき、1枚の画像に貼りつける車両の台数は4.4章の結果より最も高性能だった平均10台とする。

実験の結果、60%の車両を重ねたときが最も学習後の性能が高く、その次に性能が高いのは40%に設定したときだった。検出結果を確認したところ、重なる台数が少ないと重なっていない車両は検出できるが、重なった車両の

検出精度が低下する傾向にあった。一方で、重ねる台数が多すぎると重なっている車両は検出できるが、重なっていない車両に対して複数の検出枠を出してしまい、重なっていない車両の精度が低下する傾向にあった。



(a) 実験 3-1

(b) 実験 3-4

図 6 実験 3 の結果

また、最も性能が高かったのは 60% の場合だが、重ねて貼りつけられる前景ペアを探索するのに時間がかかり、前景を重ねる数が増えるほど探索に時間がかかる。重なりなしのとき、1 枚の画像を生成するのに平均 2.4 秒かかるが、40% の時は平均 6.4 秒、60% の時は平均 14.7 秒と大きな差がある。特に、40% の時と 60% の時では性能はあまり差がない一方、生成時間には 2.3 倍の差がある。

## 5. 性能評価

### 5.1 性能評価の条件設定

合成画像による学習の有効性を確認するために、「人手で正解ラベルを入力した追加学習」、「人が素材を確認した後、画像合成で画像・正解ラベルを生成した追加学習（提案手法：素材確認あり）」、「人が素材を確認せず、画像合成で画像・正解ラベルを生成した追加学習（提案手法：素材確認なし）」、「追加学習を行わなかった場合」の 4 つを比較する。

実験には、2 つのシーン（1. 高速道路、2. 交差点）で撮影されたデータを用いる。1 の高速道路は 4 章と同一の地点である。2 の交差点を図 7 に示す。



図 7 実験現場の撮影映像（交差点）

物体検出モデルや学習手順は 4 章と同様とする。提案手法

で前景を貼り付けるときには、4 章の実験結果から最も有効だった方針、つまり、座標の制約を課し、貼りつけ時にサイズの変更を行い、前景の同時出現数の平均を現場と合わせて、重なった車両を生成する方針とする。ただし、重なった車両を生成する割合は、生成時間も考慮して 40% とする。

また、Mask-RCNN で車両を抽出したときに、誤って白線や矢印を前景素材として抽出してしまうことがある。例えば図 8 のように車両の端や舗装を抽出してしまい、これを車両として学習してしまうと性能低下の原因となる。誤った前景を学習したときの影響を調査するため、Mask-RCNN で取り出した前景を人手でチェックし、不良な素材を排除した場合（素材確認あり）と、しない場合（素材確認なし）の 2 つを比較する実験を行う。



図 8 車両以外を抽出してしまった例

### 5.2 結果と考察

2 つのシーンでの実験結果を表 2, 3 にそれぞれ記す。いずれのシーンでも、合成画像によって学習した場合の性能は追加学習を行わなかった場合の性能を大きく上回り、合成画像による学習の有効性を確認できた。具体的には、従来の 35 point の改善に比べ 27 point の改善ができた。すなわち、素材確認ありの高速道路では従来の 5.5% の作業コストで、従来の 77% の性能改善を図ることができた。

表 2 実験の結果（高速道路）

手法	AP (Mask) %	必要工数 (時間)
人手入力	97.52	144
画像合成 (素材確認あり)	89.72	8
画像合成 (素材確認なし)	89.56	4
追加学習なし	62.46	0

表 3 実験の結果（交差点）

手法	AP (Mask) %	必要工数 (時間)
人手入力	96.49	144
画像合成 (素材確認あり)	85.50	8
画像合成 (素材確認なし)	84.95	4
追加学習なし	80.90	0

一方、交差点では 16 point に対して 5 point の改善に留まった。

検出結果を確認すると、追加学習を行わなかった場合は検出できなかったパターンが、合成画像を学習した場合には検出できていることがわかる。例えば図9の場合、追加学習なしでは中央分離帯に隠れている車が検出できていないが、合成画像で追加学習すると検出できるようになっている。(黄矢印) また、素材確認なしでは矢印舗装を車両として誤検出しているが、素材確認を行い除外すると検出しなくなっている。(青矢印)

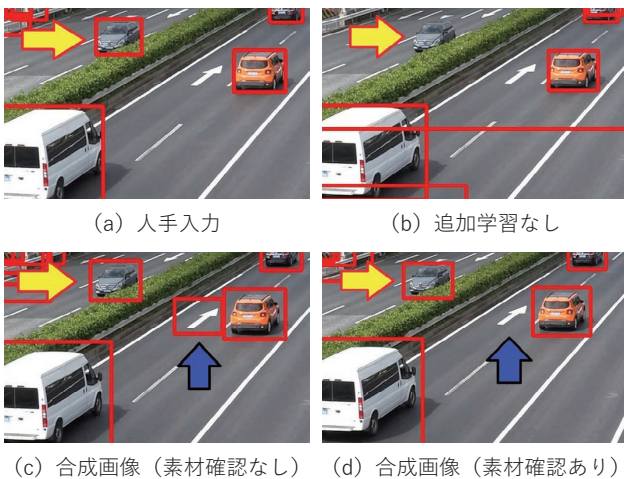


図9 検出結果の比較

合成画像による追加学習では、人手で真値入力したデータを学習した場合の性能には到達できなかったが、コスト面で優位性を示した。例えば、本実験で使用した 1800 枚の画像に人手で正解ラベルを入力するには延べ 144 時間を要した。一方、画像合成の場合は生成画像を確認しながら、パラメーターを調整するのみで済むため、素材を確認しない場合は約 4 人時、確認する場合は約 8 人時で画像と正解ラベルが生成できた。

さらに、合成画像を扱う従来研究と比べて計算処理時間を大幅に削減できた。T. Hodan らの手法では、 $640 \times 480$  画素の画像の生成に 1 枚平均 120 秒を要したが、提案手法では  $1600 \times 900$  画素の画像生成を 1 枚平均 6.4 秒で達成した。

残課題としてはこの作業コストを維持しつつ、さらに実際の映像に近づくための貼り付け方の検討する必要がある。欠けた前景や車両以外を除外しても、人手で入力した際の精度には及ばなかった。これは、自動で貼りつけた際の画像の分布が実際の画像の分布と異なっているからと考えられる。貼りつけた際の陰影処理や、重なった車の生成方法の再検討などが必要と考えられる。

## 6. むすび

本研究では車両検出のタスクに対して、自動化されたツールで高速に画像と正解ラベルを生成し、追加学習を行

う手法を研究・開発し、その有効性を確認できた。

具体的には、現場で撮影した画像から車両と背景を分離し、背景に対して車両を様々に貼りつけることで自動的に画像と正解ラベルを生成した。また、貼り付け方を工夫することにより、自動生成された画像と実画像の分布をできるだけ近づけ、見逃しや誤検出を減らすことができた。その結果、人手で入力した正解ラベルに比べて 5.5% の工数で精度を向上させ、環境適応することができた。

今後の課題としては、この作業コストの少なさを維持しつつ、人手で正解ラベルを入力した場合との性能の差をどのようにして埋めていくか、という点が挙げられる。この差は貼りつけ方がまだ実際の画像と異なっていると考えられる。今後、この課題に対して自然な貼り付け方を模索して改善を進めていき、固定カメラの車両のみならず、人・動物の検出や、移動カメラなど様々なフィールドへ技術展開していきたいと考えている。

## 参考文献

- 1) 公益社団法人日本防犯設備協会. “画像解析に関する調査研究報告書”. [https://www.ssaj.or.jp/jssa/pdf/gazou\\_kaiseki.pdf](https://www.ssaj.or.jp/jssa/pdf/gazou_kaiseki.pdf), (参照 2020-5-13).
- 2) A. Torralba; A. A. Efros. “Unbiased look at dataset bias”. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2011, p.1521-1528.
- 3) T. Hodan; V. Vineet; R. Gal; E. Shalev; J. Hanzelka; T. Connell; P. Urbina; S. N. Sinha; B. K. Guenter. Photorealistic image synthesis for object instance detection. arXiv preprint arXiv:1902.03334, 2019, (参照 2020-5-13).
- 4) G. Georgakis; A. Mousavian; A. C. Berg; J. Kosecka. Synthesizing training data for object detection in indoor scenes. arXiv preprint arXiv:1702.07836, 2017, (参照 2020-5-13).
- 5) S. Ren; K. He; R. Girshick; J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017, Vol.39, No.6, p.1137-1149.
- 6) Liu. Wei; D. Anguelov; D. Erhan; C. Szegedy; S. E. Reed; C. Y. Fu, et al. “SSD: Single Shot MultiBox Detector”. Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol.9905, Springer, Cham. 2016, p.21-37.
- 7) J. Redmon; S. Divvala; R. Girshick; A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, p.779-788.
- 8) K. He; G. Gkioxari; P. Dollár; R. Girshick. “Mask R-CNN”. IEEE International Conference on Computer Vision (ICCV). 2017, p.2980-2988.
- 9) M. Everingham; L. V. Gool; C.K. I. Williams; J. Winn; A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. International Journal of Computer Vision. 2010, Vol.88, No.2, p.303-338.

- 10) Lin, T.Y. et al. "Microsoft COCO: Common Objects in Context"  
Computer Vision –ECCV 2014. Lecture Notes in Computer Science. Vol.8693, Springer, Cham. 2015, p.740-755.

## 執筆者紹介



**山本 健生 YAMAMOTO Yoshiki**

オムロン ソーシャルソリューションズ  
株式会社  
技術創造センタ 先端技術研究所  
専門：数学（微分幾何学）



**平井 佐和 HIRAI Sawa**

オムロン ソーシャルソリューションズ  
株式会社  
技術創造センタ 先端技術研究所  
専門：情報工学



**濱走 秀人 HAMABASHIRI Hideto**

オムロン ソーシャルソリューションズ  
株式会社  
技術創造センタ 先端技術研究所  
専門：画像処理  
所属学会：電子情報通信学会



**岡本 大和 OKAMOTO Yamato**

オムロン ソーシャルソリューションズ  
株式会社  
技術創造センタ 先端技術研究所  
専門：知能情報学  
所属学会：人工知能学会

---

本文に掲載の商品の名称は、各社が商標としている場合があります。