Vol.2018-CVIM-212 No.18
2018/5/10

マルチドメイン学習による複数物体追跡

齋藤 恭兵[1,a)]　川本 一彦[2,b)]

概要：単一物体追跡のための深層学習モデル MDNet（Multi-Domain Convolutional Neural Network）を拡張し，複数物体追跡のための深層学習モデルを提案する．MDNet は，さまざまなドメインの動画での追跡タスクを同時に学習し，複数ドメインに共通する画像特徴を獲得することで汎化性能を向上させている．提案モデルでは，MDNet の共通層に加えて，複数クラスの出力層をもつドメイン固有層を導入し複数物体追跡を実現している．ドメイン固有層は，シーンに固有の画像特徴をオンライン学習で獲得するために用いられる．ドメイン共通層を用いることで，オンライン学習で更新するパラメータを少なくでき，効率的に学習できる．評価実験では，MOT Benchmark データセットに対して，MOTA と MOTP を用いて提案モデルを評価し有効性を示している．

キーワード：物体物体追跡，深層学習，畳み込みニューラルネットワーク

1. はじめに

物体を追跡し，人物の挙動を予測することは，防犯やマーケティングにおいて重要なタスクとなっている．このような複数物体追跡の研究は，セキュリティ分野においては，防犯カメラ映像から不審な人な動きや，暴行事件の発生現場における，監視の負担を軽減できる．マーケティング分野では，店舗内での人の流れから購買者の行動パターンを分析することなどが出来る．オンライン型の追跡であることから，データマイニングに限らず，様々な場面で応用が期待出来る．現在までに，単一物体追跡関する研究は多く進められ，精度も良いが，複数物体追跡に関してはまだまだ研究の余地がある．複数物体追跡の手法も多岐にわたり，カルマンフィルタ [1] やパーティクルフィルタ [2] を用いたものが初期には多く用いられてきた．

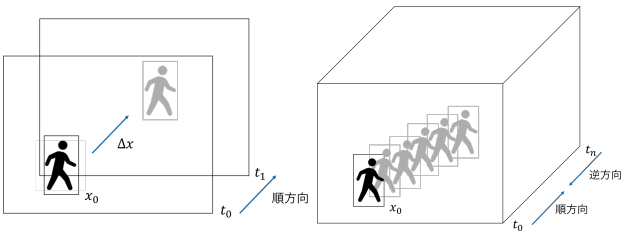
現在では深層学習を用いた手法 [3]，[4] も多く用いられる．深層学習は，画像分類，セマンティックセグメンテーション，物体検出など様々なタスクで用いられている．これにより，特徴量を自動で学習し，追跡を行う事が可能となる．しかし，深層学習を用いた複数物体追跡の研究は，オフライン型のもので end-to-end のモデルではないことが多い．

[1] 千葉大学大学院 融合理工学府
Graduate School of Science and Engineering, Chiba University
[2] 千葉大学大学院 工学研究院
Graduate School of Engineering, Chiba University
[a)] k-saito@chiba-u.jp
[b)] kawa@faculty.chiba-u.jp

本研究では，リアルタイム性を考慮したオンライン型の複数物体追跡を目的とする．そのために，深層学習を用いた end-to-end の複数物体追跡モデルを構築する．Nam ら [5] の Multi-Domain Network（以下，MDNet）は，モデルを小さくし，マルチドメイン学習することで，深層学習でオンライン型の物体追跡を可能にした．しかし，MDNet は単一物体追跡を対象としている．提案手法では MDNet をベースに，複数物体の追跡に対応するため，オンライントラッキングで用いるモデルの最終層の出力を複数クラスにする．そして，提案手法を用いて，データセットによる評価する．

2. 深層学習を用いた物体追跡

物体追跡アルゴリズムは，オフライン型とオンライン型の 2 つに分けられる．オフライン型（図 1b）は，動画の全フレームを用いて，動画全体で一括処理する手法で，順方向のみならず，逆方向からも最適化ができる．オフライン型の利点は，精度は良いということがあげられる．これは，逐次処理よりもノイズ，オクルージョンに対し頑健になるためである．しかし，使用用途がデータマイニングや 2 次利用に限られる．オンライン型（図 1a）は，各フレームが入力されるたびに処理する手法で，順方向でのみ逐次処理する．そのため，未来に入力されるフレームがわからなくても，処理ができる．精度はオフライン型と比較して悪い傾向にある．しかし，各フレームごとの処理が高速であれば，リアルタイムにターゲットの追跡が可能になる．そのため，汎用性がある．



(a) オンライン型　　(b) オフライン型

図 1: 物体追跡の種類

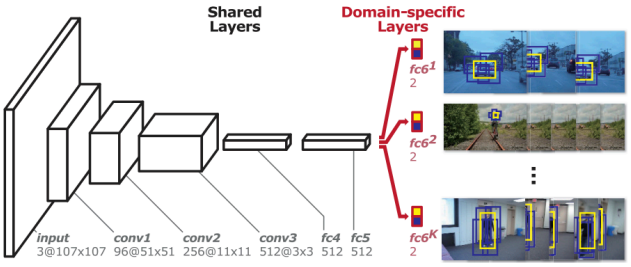


図 2: MDNet モデル構造（[5] より引用）

[3] は SiameseCNN を用いた手法である．こちらは CNN の出力に対し，Softassign アルゴリズム [6] を用いる．[4] は LSTM を用いた手法で，複数に分かれたモデルの出力を最後に結合し，結果を求める． これらの手法は end-to-end モデルではない．end-to-end のモデルであることは，ドメインに依存する，パラメータの設定を少なく出来る．さらに，モデル構造が簡単になり，利用者は動画とラベルを与えることで結果が得られる利点がある．単一物体追跡である MDNet はオンライン型の end-to-end 物体追跡モデルである．MDNet はマルチドメイン学習を特徴とする CNN で，単一物体追跡では多くの派生したモデル [7]，[8] が存在する．これは，MDNet のモデル構造が簡単なことと，オンライン追跡のため，初期フレームの情報のみで利用でき，ドメインに依存しないという点で拡張性があるためである．以下では，MDNet ついて説明する．

2.1 MDNet

MDNet[5] のモデル構造を図 2 に示す．3 つの畳み込み層と 2 つの全結合層の共有レイヤー，各ドメインに対応する K 個の全結合層の，ドメイン固有レイヤーからなるモデルである．AlexNet[9] や VGG-Nets[10] と異なり，モデルが小さい理由として，3 つある．物体追跡はターゲットと背景の 2 クラス分類であること，深い CNN は，空間情報が希釈されてしまう傾向にあること，モデルの入力であるターゲットが通常小さいサイズであることである．このモデルでの追跡では，プレトレーニングとオンライン追跡の 2 つのフェーズに別れる．

2.1.1 プレトレーニング

プレトレーニングではマルチドメイン学習する．ground truth をもつ動画のデータセットを用いて，各フレームで，ポジティブデータとネガティブデータを生成する．生成したデータを 107×107 の RGB 画像としてモデルの入力とする．出力である K 個のドメイン固有レイヤーはバイナリ分類のための 2 出力をもつ．つまり，各ドメイン内のターゲットとバックグラウンドの 2 クラスに分類する．モデル全層を学習の更新対象とし，ドメインに共通の特徴を学習する．ドメイン固有レイヤーを持つことによって，ドメインに依存しない情報とドメイン固有の上を分離し，統一的な学習ができる．これにより，テストデータと異なるドメインをトレーニングデータに用いることが可能になる．

2.1.2 オンライン追跡

マルチドメイン学習によるプレトレーニングを終えると，K 個のブランチは，テストシーケンスのための 1 つのブランチに置き換え，新しいモデルを構築する．オンライン追跡ではまず，1 フレーム目で ground truth をもとにトレーニングする．2 フレーム目以降も 2 つの間隔で再トレーニングする．ランダムで生成した，N 個の候補 $x^1, ..., x^N$ はモデルによって評価され，ポジティブスコア $f^+(x^i)$ とネガティブスコア $f^-(x^i)$ が得られる．推定されるターゲットは最も大きいポジティブスコアに対応する候補 x^* である．つまり，式 (1) によって得られる x^* である．

$$x^* = \arg\max_{x^i} f^+(x^i) \tag{1}$$

オンライン追跡でのトレーニングでは，4 層目以降を学習の更新対象とし，転移学習する．これにより，テストデータに固有の特徴を学習する．

2.1.3 ハードネガティブマイニング

ポジティブサンプルと比較し，ネガティブサンプルの方が冗長なデータが多く，利用可能なデータ数が多くなる．トレーニングサンプルが学習に均等に寄与する SGD (stochastic gradient descent) を用いると，ポジティブサンプルとネガティブサンプルのトレーニングの間に重大な不均衡をもたらす．この問題の解決方法として，ハードネガティブマイニング [11] がある．すべてのネガティブデータを利用する代わりに，各ネガティブサンプルを予め，モデルに入力し，高いネガティブスコアに対応する入力を選択し，データの選別する．これにより，早い最適化，安定した訓練ができる．

2.1.4 バウンディングボックス回帰

本手法は前フレームで推定されたバウンディングボックスを中心に，ランダムにデータを生成する．そのため，対象を囲むバウンディングボックスが見つからないことがある．物体検出で一般的な，バウンディングボックス回帰 [12]，[13] を適用し，検出位置精度を改善する．テストシーケンスの 1 フレーム目が与えられると，ターゲットの付近のサンプルを入力としたときの 3 番目の畳み込み層の特徴マップを用いて，線形回帰モデルを訓練する．2 フ

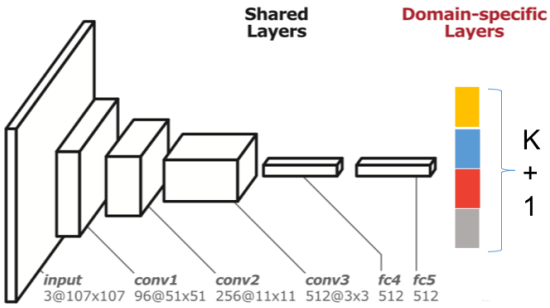


図 3: 提案モデル

レーム目以降では，式 (1) で推定されたターゲットの位置を調整する．オンライン追跡において，毎フレームおきに回帰モデルの学習すると，時間的コストが大きくなる．そのため，1 フレーム目でのみトレーニングされる．

3. 提案手法

MDNet は単一物体追跡のためのモデルである．複数物体追跡の問題に適応する方法を考える．MDNet はプレトレーニング，追跡ともに出力層は 2 クラスの出力をもつ．これはシングル追跡では対象が 1 つのため，対象である・ないの 2 クラス分類となるからである．2 人以上の複数物体を追跡するため，クラス数を変更する．プレトレーニングでは，人物の特徴のみを学習すればよいため，MDNet と同じく，クラス数は 2 つとした．複数物体追跡では，各フレームごとで人数が変化することがある．今回は問題を簡単にするため，動画内の人数が K 人で，人数固定の場合を考える．

3.1 プレトレーニング

MDNet と同様に，2 クラス分類のモデルをマルチドメイン学習によって，トレーニングする．トレーニング動画に登場するすべての人物と同数の出力層を定義する．各動画フレーム I_1，I_2，I_3，...，I_N と，対応するフレーム内での人物の位置（x 座標，y 座標，$width$，$height$）を表す ground truth データを用意する．モデルの入力には，ground truth をもとにランダムに生成した，クロップ画像を与える．ランダムに生成するとき，人物の特徴を学習するために生成するポジティブデータと，背景部分の特徴を学習するために生成するネガティブデータがある．ポジティブデータはガウス分布，ネガティブデータは一様分布で生成する．このとき，真の人物位置とのオーバラップ率 IoU（intersection over union）（式（2））を計算し，生成する．

$$\mathrm{IoU} = \frac{r_t \cap r_a}{r_t \cup r_a} \tag{2}$$

r_t は真の矩形領域，r_a は予測された矩形領域を表す．また，ネガティブデータを発生させるとき，各バウンディングボックスに他の対象が含まれないようにする．他クラスとの IoU が 0.3 以下となるように生成する．プレトレーニングでは，VOT dataset[14] で学習されたモデルを用いて，転移学習により，本実験で使用するデータセットを学習する．

3.2 オンライン追跡

マルチドメインによるプレトレーニング後，ドメイン固有レイヤーである複数の最終層を単一のものに置き換える．オンライン追跡では，最終層を $K+1$ クラスに変更した，提案モデル（図 3）を使用する．追跡では各ターゲットを区別する必要がある．これは分類問題にあたる．分類問題では出力層は分類したいクラスと同じユニットで構成する必要がある．そのとき，用いられる活性化関数として式（3）に表すソフトマックス関数がある．

$$y_k = \frac{\exp(a_k)}{\sum_{i=0}^{K} \exp(a_i)} \tag{3}$$

K はクラス数，k は k 番目のユニットの出力である．$0 \leq y_k \leq 1$，$\sum_{i=0}^{K} y_i = 1$ となり，確率として表現することができる．出力層にソフトマックス関数を用いる場合，損失関数として式（4）に表す交差エントロピーを用いる．

$$loss(x_i, y_i) = -\sum_{i=1}^{K} y_i \log(x_i) \tag{4}$$

x_i はソフトマックス関数の出力，y_i は教師データである．

トレーニングでのポジティブデータとネガティブデータデータの生成方法はプレトレーニングと同様である．1 フレーム目の ground truth をもとに，ポジティブデータ，ネガティブデータを生成し，モデルのトレーニングする．この時，バウンディングボックス回帰モデルもトレーニングする．トレーニング後，2 フレーム目でのターゲット位置を推定する．2 フレーム目以降も，前フレームで推定された位置をもとにデータを生成し，同様に処理する．追跡中は一定フレームごとにトレーニングする．そのとき，ポジティブデータは過去 100 フレーム，ネガティブデータは過去 20 フレームを使用しトレーニングする．また，追跡が失敗した場合にもトレーニングする．追跡対象となるクラスの出力が閾値を下回った場合を失敗とする．オンライン追跡のアルゴリズムを Algorithm 1 に示す．

4. データセットによる評価実験

提案手法によって，$K = 2, 3$ の人数固定のシーンで実験する．

4.1 評価用データセット

使用したデータは，MOT Benchmark[15] である．MOT Benchmark は複数物体追跡のためのベンチマークである．用意されたトレーニングデータによってトレーニングを行い，テストデータに対する性能を評価する．各歩行者に対

Algorithm 1 オンライン追跡アルゴリズム

Input: プレトレーニングされた重み $\{w_1, ..., w_5\}$, 初期ターゲット位置 $x_{1,2,..,k}$
Output: 推定されたターゲット位置 $x^*_{1\in t}, x^*_{2\in t}, .., x^*_{k\in t}$

ランダムに w_6 を初期化
バウンディングボックス回帰をトレーニング
ポジティブデータ $S^+_{1\in t}, S^+_{2\in t}, .., S^+_{k\in t}$, ポジティブデータ $S^-_{1\in t}, S^-_{2\in t}, .., S^-_{k\in t}$ を生成
$\{w_4, w_5, w_6\}$ を $S^+_{1\in t}, S^+_{2\in t}, .., S^+_{k\in t}$, $S^-_{1\in t}, S^-_{2\in t}, .., S^-_{k\in t}$ を使用し初期化
T_l はポジティブデータの集合，T_s はネガティブデータの集合
repeat
    サンプル $x^i_{1\in t}, x^i_{2\in t}, .., x^i_{k\in t}$ を生成
    式 (1) にて $x^*_{1\in t}, x^*_{2\in t}, .., x^*_{k\in t}$ を決定
    if $f^+(x^*_{1\in t}, x^*_{2\in t}, .., x^*_{k\in t}) > \frac{1}{K+1}$ **then**
        $S^+_{1\in t}, S^+_{2\in t}, .., S^+_{k\in t}, S^-_{1\in t}, S^-_{2\in t}, .., S^-_{k\in t}$ を生成
        $T_s \leftarrow T_s \cup \{t\}$, $T_l \leftarrow T_l \cup \{t\}$
        if $|T_s| > 20$ **then** $T_s \leftarrow$ 最も古いデータを除いた T_s
        if $|T_l| > 100$ **then** $T_l \leftarrow$ 最も古いデータを除いた T_l
    end if
    if $t \bmod N = 0$ **then**
        $S^-_{v\in T_s}$, $S^+_{v\in T_l}$ を使用し，$\{w_4, w_5, w_6\}$ を更新
    end if
until end of sequence

して，フレーム番号，ID，x 座標，y 座標，$width$，$height$ が与えられている．本実験では MOT Benchmark のトレーニングデータのうち，9 個をトレーニングデータ，6 個をテストデータとする．6 個のテストデータは，選択した 2 人または 3 人の対象が写ってるフレームのみを選択する．そして，各テストデータでオクルージョン有り無し，各 3 種類，計 36 個作成する．トレーニングデータを表 1，テストデータを表 2 と 3 に示す．

4.2 実験設定

4.2.1 パラメータ設定

パラメータの設定は MDNet と同じとした．追跡では，各フレームの人物ごとに，初期フレームを除き，ポジティブデータ $S^+_t = 50$，ネガティブデータ $S^-_t = 200$ によってトレーニングする．初期フレームは $S^+_t = 500$ と $S^-_t = 5000$ である．バウンディングボックス回帰のトレーニングは [13] と同じパラメータで，人物ごとに 1000 のサンプルを生成し使用する．$IoU \geq 0.7$ を S^+_t，$IoU \leq 0.3$ を S^-_t とする．オンライン追跡中のトレーニングは 10 フレームごとにする．その他パラメータについては表 4 と 5 に示す．

4.2.2 追跡難易度の変更

追跡において，オクルージョンが発生する状況は追跡タスクとして難しい．本実験では，オクルージョンが発生しない状況と発生する状況の 2 つに分け実験した．オクルージョンが発生する難しい状況では追跡対象は 2 人（K=2）で，発生しない簡単な状況では追跡対象は 3 人（K=3）である．

4.3 評価指標

MOT Challenge で用いられる評価指標である，MOTA，MOTP で結果の客観評価した．

4.3.1 MOTA（multiple object tracking accuracy）

MOTA[16] は式 (5) のように表される．

$$\mathrm{MOTA} = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{5}$$

- FN：未検出数 (false negative)
- FP：誤検出数 (false positive)
- $IDSW$：ID 変化回数
- GT：真の追跡対象数 (ground truth)

FN，FP，$IDSW$ 及び GT はそれぞれ時刻 t における未検出数，誤検出数，ID 変化数，及び真の追跡対象数である．3 つのエラーを含む指標であることから，物体追跡の全体的な性能を考慮した指標であるといえる．

4.3.2 MOTP（multiple object tracking precision）

MOTP[16] は式 (6) のように表される．

$$\mathrm{MOTP} = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{6}$$

- c：一致した追跡対象数
- d：真の追跡対象とのオーバラップ率

c，d はそれぞれ時刻 t における，一致した追跡対象数，真の追跡対象とのオーバラップ率である．追跡に成功したボックスの精度を表す指標であるといえる．推定位置と ground truth の IoU が 20%を超えた場合，追跡成功とする．

4.4 実験結果

図 4 のグラフに客観評価の結果を示す．MOTA，MOTP は 3 つのデータの平均値である．図 4 にオクルージョン有り無しそれぞれの MOTA，MOTP のグラフを示す．追跡成功例を図 6，失敗例を図 7 に示す．

4.5 考察

MOTA はオクルージョン無しの場合，ほとんどの場合で 100%となり，動画全体で正確に追跡出来た．オクルージョン有りの場合，27.4〜100%となり，テストデータによって差があった．これは，オクルージョンが発生しているフレームの長さ，オクルージョンによって対象がどの程度が覆われているか，フレームレートによる移動速度の違いなどの影響があると考えられる．MOTP はオクルージョン有り無し，どちらの場合でも 70%前後で，わずかにオクルージョン無しのほうが精度が良い．よって，追跡対象の人数固定で，オクルージョン無しの時，トラッキングがほとんどの場合でき，オクルージョン有りの時はテストデータに依存することが分かった．

表 1: トレーニング動画

動画名	フレーム数	fps	トラッカー	画像サイズ
MOT17-10（MOT17）	654	30	57	1920 × 1080
MOT17-11（MOT17）	900	30	75	1920 × 1080
MOT17-13（MOT17）	750	25	750	1920 × 1080
MOT17-05（MOT17）	837	14	837	640 × 480
ETH-Pedcross（MOT15）	1000	14	171	640 × 480
ETH-Sunnyday（MOT15）	354	14	30	640 × 480
KITTI-13（MOT15）	340	10	42	1242 × 375
KITTI-17（MOT15）	145	10	9	1242 × 370
ARENA-N1-01_02_TRK_RGB_1（PETS2017）	513	30	5	1280 × 960

表 2: テスト動画:オクルージョン無し

動画名	フレーム数（データ 1 / 2 / 3）	fps	画像サイズ
MOT17-02（MOT17）	131 / 36 / 51	30	1920 × 1080
MOT17-04（MOT17）	271 / 60 / 176	30	1920 × 1080
MOT17-09（MOT17）	71 / 59 / 62	30	1920 × 1080
PETS09-S2L1（MOT15）	111 / 53 / 37	7	768 × 576
TUD-Campus（MOT15）	21 / 22 / 25	25	640 × 480
TUD-Stadtmitte（MOT15）	30 / 21 / 30	25	640 × 480

表 3: テスト動画:オクルージョン有り

動画名	フレーム数（データ 1 / 2 / 3）	fps	画像サイズ
MOT17-02（MOT17）	214 / 47 / 55	30	1920 × 1080
MOT17-04（MOT17）	345 / 150 / 530	30	1920 × 1080
MOT17-09（MOT17）	71 / 121 / 84	30	1920 × 1080
PETS09-S2L1（MOT15）	130 / 42 / 24	7	768 × 576
TUD-Campus（MOT15）	43 / 21 / 30	25	640 × 480
TUD-Stadtmitte（MOT15）	30 / 51 / 63	25	640 × 480

表 4: プレトレーニング概要

学習回数	50 iter [a]
バッチサイズ	128（ポジティブデータ：32，ネガティブデータ：96）
最適化手法	SGD （lr[b]= 0.0001 or 0.001, $momentum = 0.9, weightdecay = 0.0005$）
損失関数	Binary Cross Entropy

[a] 1iter は各動画 8 フレーム，各フレームで全ての人物に対してバッチデータを生成し，トレーニング

[b] 畳み込み層では 0.0001，全結合層では 0.001

表 5: 追跡概要

学習回数	初期フレーム：30 iter，その他：15 iter
バッチサイズ	128（ポジティブデータ：32，ネガティブデータ：96 [a]）
最適化手法	SGD 初期フレーム（lr[b]= 0.0001 or 0.001, $momentum = 0.9, weightdecay = 0.0005$） 2 フレーム目以降（$lr = 0.0002, momentum = 0.9, weightdecay = 0.0005$）
損失関数	Cross Entropy

[a] ミニバッチのネガティブデータはハードネガティブマイニングによって，1024 個の中から選択した 96 個

[b] 畳み込み層では 0.0001，全結合層では 0.001

4.6 サンプル生成パラメータの調整

オクルージョン有りの場合，MOTA にばらつきがあり，テストデータに依存することが分かった．本実験では，オンライン追跡時，正規分布によってサンプルを生成している．そのとき，分散は全て同一のものを使用した．しかし，各テストデータにより，フレームレートやオクルージョンが発生しているフレーム数が異なる．そこで，テストデータごとに分散を変更し追加実験した．

追加実験結果のグラフを図 5 に示す．サンプル生成時の分散をテストデータに合わせて変更することで，MOTA がほとんどの場合で上昇し，精度が向上した．サンプル生成時に人物の移動方向を考慮するなど，サンプル生成方法について今後検討する必要がある．

5. おわりに

本研究では，オンライン型の複数物体追跡を目的とし，深層学習を用いた end-to-end モデルを提案した．単一物体追跡である MDNet を拡張し，人数固定でオクルージョンが発生しない状況において複数物体追跡にも応用が可能であることを確認した．しかし，オクルージョンが発生する状況では，精度が低下することが課題として残った．オンライン追跡中のデータの生成方法を工夫することで，改善が期待できる．

今回は 2 人または 3 人の状況で，人数固定の場合に限定し実験した．今後の課題として，3 人より多い状況で，シーンでの人数可変の場合に対応する必要がある．人数が減少する場合は対象となる出力を無効にすることが必要である．人数が増加する場合には，モデルを部分的にトレーニングする必要である．今回は出力層のみ変更した．人数が増えることで，必要とする中間層のユニット数，レイヤー数も変更する必要がある．

謝辞

本研究は JSPS 科研費 JP16K00231 の助成を受けたものです．

参考文献

[1] J. Black, T. Ellis, and P. Rosin. Multi view image surveillance and tracking. *In Proceedings of the Workshop on Motion and Video Computing*, 2002.

[2] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. *In ECCV*, 2004.

[3] B. Wang, L. Wang, B. Shuai, Z. Zuo, T. Liu, K. Luk Chan, and G. Wang. Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association. *In CVPRW*, 2016.

[4] Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. *arxiv*, 2017.

[5] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. *CoRR*, abs/1510.07945, 2015.

[6] S. Gold and A. Rangarajan. Softmax to softassign: Neural network algorithms for combinatorial optimization. *Artificial Neural Nets*, 2(4):381–399, 1995.

[7] Hyeonseob Nam, Mooyeol Baek, and Bohyung Han. Modeling and propagating cnns in a tree structure for visual tracking, 2016.

[8] Sangdoo Yun, Jongwon Choi, Youngjoon Yoo, Kimin Yun, and Jin Young Choi. Action-decision networks for visual tracking with deep reinforcement learning. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[10] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[11] Kah Kay Sung and Tomaso A. Poggio. Example-based learning for view-based human face detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(1):39–51, 1998.

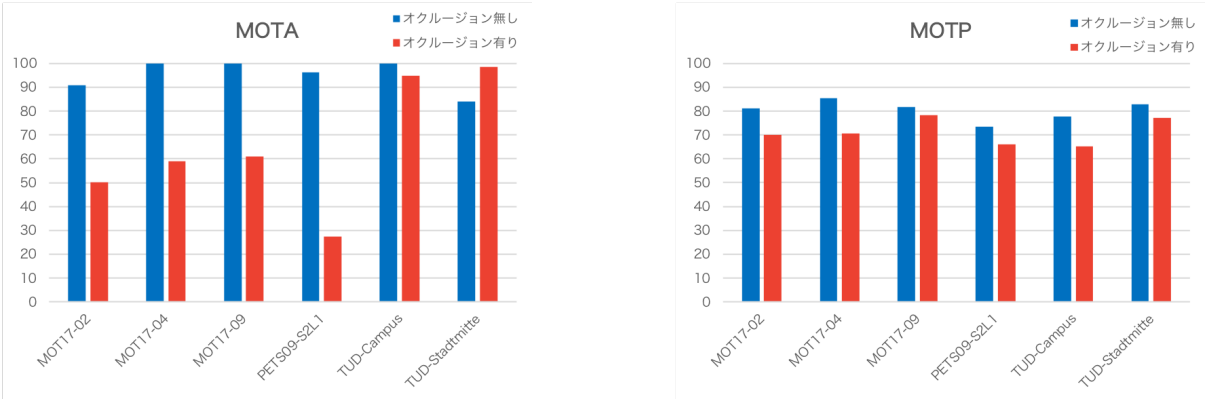
[12] Pedro F. Felzenszwalb, Ross B. Girshick, David A. McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010.

[13] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587. IEEE Computer Society, 2014.

[14] Matej Kristan, Jiri Matas, Aleš Leonardis, Tomas Vojir, Roman Pflugfelder, Gustavo Fernandez, Georg Nebehay, Fatih Porikli, and Luka Čehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2137–2155, Nov 2016.

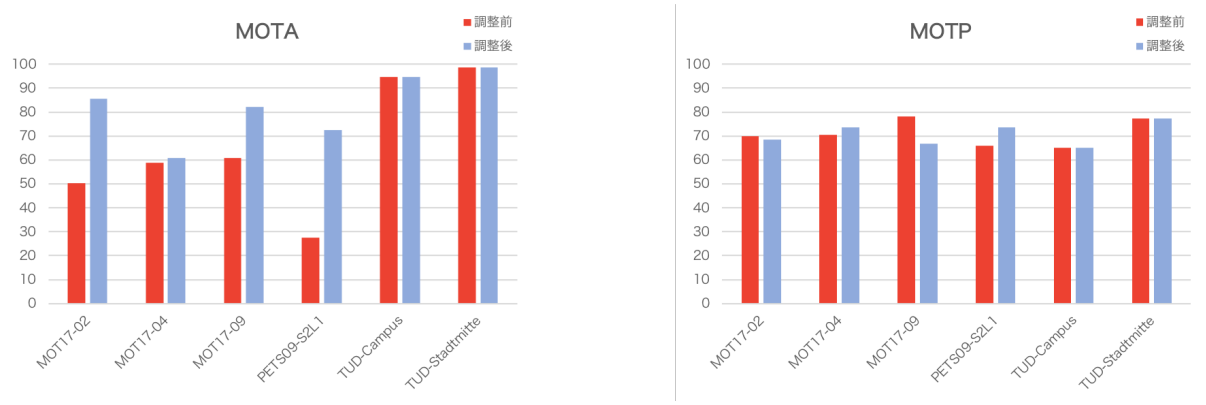
[15] Anton Milan, Laura Leal-Taixe, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. abs/1603.00831, 2016.

[16] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP J. Image and Video Processing*, 2008, 2008.



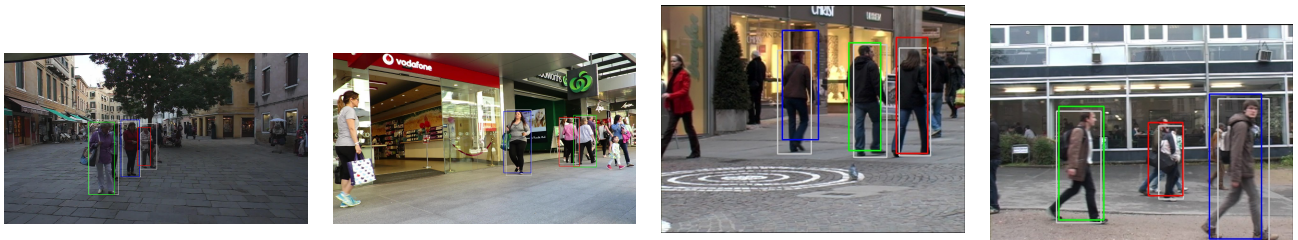
(a) MOTA (b) MOTP

図 4: オクルージョン有り無しでの MOTA，MOTP の比較



(a) MOTA (b) MOTP

図 5: サンプル生成確率変更前後の MOTA の比較



(a) MOT17-02（frame：66） (b) MOT17-09（frame：36） (c) TUD-Stadtmitte（frame：11） (d) TUD-Campus（frame：17）

図 6: 追跡結果（成功例）：ground truth（白），予測されたバウンディングボックス（赤，青，緑）

(a) MOT17-02（frame：82） (b) MOT17-09（frame：61） (c) PETS09-S2L1（frame：101） (d) TUD-Campus（frame：29）

図 7: 追跡結果（失敗例）：ground truth（白），予測されたバウンディングボックス（青，緑）