DEIM Forum 2021 E24-5

室内動作認識のためのドメイン適応による合成データ活用の検討

礒井　葉那† 　竹房あつ子†† 　中田　秀基††† 　小口　正人†

† お茶の水女子大学　〒 112–8610 東京都文京区大塚 2–1–1
†† 国立情報学研究所　〒 101–8430 東京都千代田区一ツ橋 2–1–2
††† 産業技術総合研究所　〒 305–8560 茨城県つくば市梅園 1–1–1
E-mail: †{hana,oguchi}@ogl.is.ocha.ac.jp, ††takefusa@nii.ac.jp, †††hide-nakada@aist.go.jp

あらまし　ディープニューラルネットワークの進歩に伴う学習データ不足の問題について様々な議論が行われており, その解決策の 1 つに合成データを利用した学習がある. 合成データには生成が比較的容易であるという利点があるが, 合成データを用いて学習したモデルには, 実データ解析時にデータの分布の違いから解析精度が低下するドメインシフトが起こるという課題がある. 本研究では, 合成動画像データを活用した高精度な実動画像データ解析の実現を目的とし, 写実的な合成動画像データを作成して学習し, その解析精度を調査した. 実験では, 作成したラベル付き合成データのみを用いて学習したネットワークと, ラベル付き合成データとラベルなし実データの両方を用いてドメイン適応して学習したネットワークを用いて実データの解析を行った. 実験の結果, 合成データのみでの学習でもドメイン適応を用いた学習でも現時点では十分な解析精度が得られず, 改善の余地があることがわかった.

キーワード　ドメイン適応, 合成データ, 動画像認識, 深層学習, コンピュータビジョン

1　はじめに

ディープニューラルネットワーク (DNN) の発展により、コンピュータービジョン分野において様々な技術が進歩している. 最近の研究では動画像から人間の行動を解析することができるようになってきており [1] [2] [3]、その技術を家庭内の子供や高齢者の見守りなどへの応用が期待されている。

DNN による画像解析の学習精度は、ラベル付き学習データセットのサイズとバリエーションに大きく依存していることが知られている [4] が, 十分なデータの収集とラベル付けには大変な時間と費用がかかる. そうした学習データ不足の問題に対する解決策として, 合成データを活用することが注目されている [5] [6] [7]. 合成データとはコンピュータを用いて生成したデータのことであり, 実データと比較して大量かつ多様なデータを容易に生成することができるという利点がある. 特に, 静止画像よりも作成が困難である動画像データにおいて, 合成データの活用が期待されている [8]. しかしながら, ドメインシフトといって, 性質が異なるデータでの学習による解析では精度が低下してしまうため, 多くの場合にドメイン適応による対応が必要であることが知られている [9] [10] [11] [12]. 特に, 動画像データのドメイン適応については十分に研究されておらず, 合成動画像データによるドメイン適応では, 高精度な実データ解析は実現されていない [13].

我々は, 合成データを用いたドメイン適応による高精度な実動画像データ解析を実現することを目的として, 写実的な合成データを作成して学習し, その解析精度を調査した. 人間の動作を収録する実動画像データセット **Ochahouse-Real** と, 同様な状況を写実的にシミュレートした合成動画像データセット **Ochahouse-Syn** を作成し, これらの **Ochahouse-Dataset** を用いて合成データを活用した実データ解析実験を行った. 実験ではまず, ドメイン適応を行わずにラベル付き Ochahouse-Syn のみで学習したモデルで Ochahouse-Real の解析を行い, 十分な精度での解析ができずドメインシフトの問題が起きていることを確認した. このとき, 学習データへのノイズ付与は解析精度を下げるが, ぼかし・カラージッターのデータ拡張により解析精度が向上することがわかった. 次に, ドメイン適応手法である DANN を適用して学習したモデルで Ochahouse-Real の解析を行ったが, 十分な解析精度が得られず, さらなる改善の必要性を確認した.

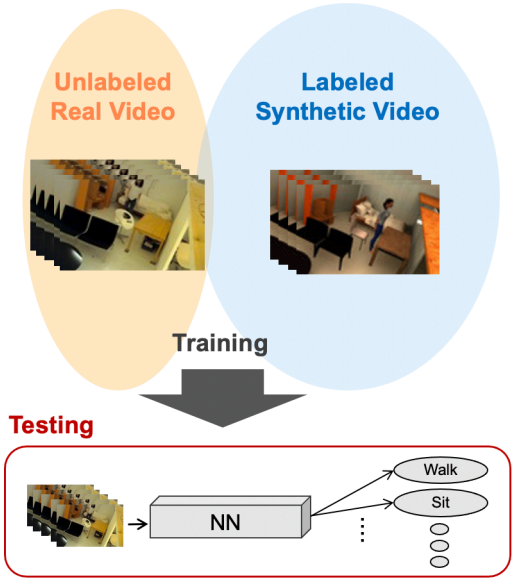


図 1　概　　要

表 1　Ochahouse Dataset の動作クラスと各データ数

クラス	walking	sitting down	sitting	standing up	lying down	lying	getting up
合成データ Ochahouse-Syn	997	747	1118	780	250	250	250
実データ Ochahouse-Real	96	44	56	51	32	39	32

2　関 連 研 究

2.1　合成データ

合成データとは, コンピュータにより自動的に生成されるデータであり, 実データと比較して大量かつ多様な生成が容易であるという利点がある. 合成データは, 実データに加えて学習データの多様化・増量や, ドメイン適応を含む転移学習に活用される.

合成データのみでの学習は, ロボットのシミュレーション実験などを主な目的として研究されてきた. 文献 [5] では合成画像のみを用いて実画像の学習を行うことを目指し, 前段階として ImageNet で事前学習し, ランダム化されたレンダリングピクセルでファインチューニングしたニューラルネットワークでロボット制御が行えることを示している. Tobin らは 2017 年, シミュレートするテクスチャ, オクルージョンレベル, シーンの照明, カメラの視野, レンダリングエンジン内の均一なノイズに対してドメインランダム化を行うことで, 単純な環境でシミュレートされた画像のみで学習した DNN でドメイン適応を行わずに実画像での高精度な物体検出に初めて成功した [6].

実データに加えて学習データを増強する目的での利用として, 実際の都市での運転シーンにおける物体検出のための合成動画像データセット ”Virtual KITTI” [7] がある. 彼らはカメラの視点, 光源, オブジェクトのプロパティをランダム化した写実的な画像をレンダリングによって生成し, 合成データが物体検出, 特にマルチオブジェクトの追跡において実世界の解析に有用であることを示した.

動画像における合成データに関する先行研究には [8] [14] [13] がある. 文献 [8] では, 多様で写実的な人間行動動画像のデータセット PHAV(Procedural Human Action Videos) を作成し, 実データに加えて学習すると HMDB-51 [15], UCF-101 [16] における解析に有効であることを示した. 文献 [14] では, 動画像のテキストや背景は物体の動きを表現するオプティカルフローにほとんど影響を与えないことに着目し, 背景を簡略化した人間行動合成動画像データセットを作成した. このデータセットから抽出したオプティカルフローで, RGB 画像とオプティカルフローそれぞれに畳み込みを施し統合する動画像解析ネットワークの 1 つ TSN(Temporal Segment Networks) [17] を追加で学習することにより, UDF-101 および HMDB-51 における精度向上に有効であることを示した.

実データと併用して用いられたこれらのデータセットと異なり, ゲームプレイ動画像から収集した 50 クラスの行動データセット Kinetics-Gameplay は, ドメイン適応による動作分類のために作成された [13].

2.2　ドメイン適応

ドメイン適応とは, ドメインシフトに対応するための手法であり, 合成データで学習された分類器を実データに用いる場合にドメイン適応が必要とされることが知られている [9] [10] [11] [12]. ドメイン適応の代表的な手法には, 解析したいデータであるターゲットデータと正解ラベルなどの多くの情報を持つソースデータとを同時にネットワークに入力してデータ間に共通する特徴を学習させる DANN(Domain-Adversarial Neural Networks) [18] の他に, ソースデータで学習させた特徴抽出器をターゲットデータ用特徴抽出の学習に用いる ADDA(Adversarial Discriminative Domain Adaptation) [19], 2 つのクラス分類器を用いて特徴抽出後の分布を近づける MCD(Maximum Classifier Discrepancy) [20] などがある.

図 2　合成動画像 Ochahouse-Syn の 1 フレーム

図 3　実動画像 Ochahouse-Real の 1 フレーム

動画像におけるドメイン適応では, Chen らが TA^{3}N(Temporal Attentive Adversarial Adaptation Network) というドメイン適応ネットワークを提案した [13]. [13] では, 学習と同時に時間的ダイナミクスのアラインメントを行い, またドメインの不一致を利用して時間的ダイナミクスを明示的に考慮することで高精度なドメイン適応を実現した. Pan らはクロスドメイン共同アテンション機構を提案し, ドメイン間の時間的なずれの問題に対処する方法を提案した [21]. また Chio ら [22] は, より識別性の高いクリップに焦点を当て, ビデオレベルのアラインメントを直接最適化する注意メカニズムを提案した. さらに, 補助タスクとしてクリップ順序予測を使用し, これらにより行動に大きく関与している人物や物体に焦点を当てた表現を学習することに成功した.

文献 [13] では合成データで動画像ドメイン適応を行っている. Kinetics-Gameplay というゲームプレイ動画から作成したデータをソースデータに利用して, ターゲットデータ (Kinetics の 30 のサブクラス) の分類に 17.22% から 27.50% の精度向上を達成した. しかしながら, この精度はラベルを使用してターゲットデータで学習した場合の 64.49%には遠く及ばない.

表 2　実験の条件

フレームワーク	PyTorch [24]
最適化手法	Adam [25]
学習率	one-cycle learning [26] でスケジュール (最大 1e-4)
エポック数	100
バッチサイズ	16 または 32(ドメイン適応)
計算資源	ABCI (産業技術総合研究所の AI 橋渡しクラウド)

3　Ochahouse Dataset

我々は, 合成動画像におけるドメイン適応のための **Ochahouse Dataset** を作成した. これは部屋の中を 1 人の人が自由に動きまわり 7 種類の動作をする様子を 1 台の固定されたカメラで収録した合成動画像 **Ochahouse-Syn** と, 実動画像 **Ochahouse-Real** で構成される. Ochahouse-Syn の作成には Unity®を使用した. Ochahouse-Syn では [8] と同様に, Unity Asset Store から入手した既成人物モデルや行動アニメーションを利用した. Ochahouse-Real は, お茶の水女子大学の実験住宅 *OchaHouse* [23] 内で筆者が動作を行い収録した. いずれもフレームレートは 5fps である. Ochahouse Dataset では, walking, sitting down, sitting, standing up, lying down, lying, getting up の 7 種類の動作クラスを作成した. 各動作クラスのデータ数は表 1 の通りであり, 各動画像は約 3 秒から 7 秒程度の長さとなっている. 作成した動画像データの 1 フレームを図 2, 図 3 に示す.

4　実　　験

我々は作成した Ochahouse Dataset を用いて, 実データ Ochahouse-Real の解析のために合成データ Ochahouse-Syn を活用する学習を行い, Ochahouse-Real の解析精度を調査する. まず, ドメイン適応を行わずにラベル付き Ochahouse-Syn のみで学習したモデルで Ochahouse-Real の解析を行い, ドメインシフトの問題が起こることを確認する. このとき, Ochahouse-Syn にデータ拡張を行うことによる効果も調査する. 次に, ドメイン適応手法である DANN を適用して学習したモデルで Ochahouse-Real の解析を行い, ドメインシフトに対応し高精度な解析が行えるか調査する.

4.1　実 験 設 定

実験の条件を表 2 に示す. 実験のコードは全て PyTorch [24] で実装され, 学習は一般的な最適化手法である Adam [25] によって最適化される. また, より汎化能力を高めドメインシフトを小さくするために, 学習率は最大学習率を 1e-4 とする one-cycle learning [26] に基づきミニバッチごとに更新される. 学習時, それぞれのドメインにつきバッチサイズは 16 であり, ドメイン適応時は両ドメインを同時に入力するためバッチサイズは合計 32 となる. 各画像データは切り抜かずに 112*112 ピクセルにリサイズされ, データ拡張された後に正規化されモデルに入力される.

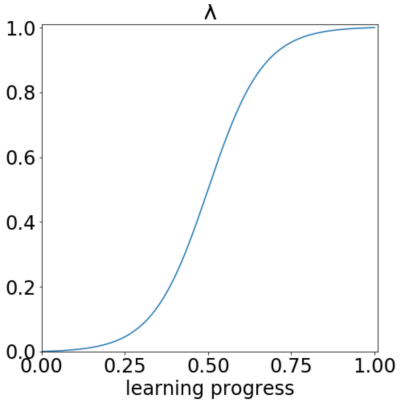


図 4　学習の進行に伴うドメイン適応のパラメータ λ が増加する様子

データ拡張では [6] [27] に基づきランダムな強さのぼかし, ノイズの付与と, 明るさ, 明度, 彩度のランダムな変更 (カラージッターと呼ぶ) を行う. ぼかしは次の式 (1) のようにモデル化されるガウスぼかしを適用した.

$$I_{blur}(x,y) = \sum_m \sum_n I(x+m, y+n) K(m,n) \qquad (1)$$

$I_{blur}(x,y)$ は処理後の位置 (x,y) の画像の値, $K(m,n)$ は二次元ガウス分布に基づくカーネルである. ノイズは次の式 (2) のようにモデル化されるガウスノイズを適用した.

$$I_{noise}(x,y) = max(min(I(x,y) + \eta_{gauss}, 0), 255) \qquad (2)$$

ここで $I_{noise}(x,y)$ は処理後の位置 (x,y) における画像の値, $I(x,y)$ は元の画像の位置 (x,y) の値, η_{gauss} はガウス分布に基づく値である. また, カラージッターには PyTorch の実装である torchvision.transforms.ColorJitter を用いた.

また, ドメイン適応を行うときの最適化の目的関数 $L_y + \lambda L_d$ のパラメータ λ について, 本実験では以下のように設定した.

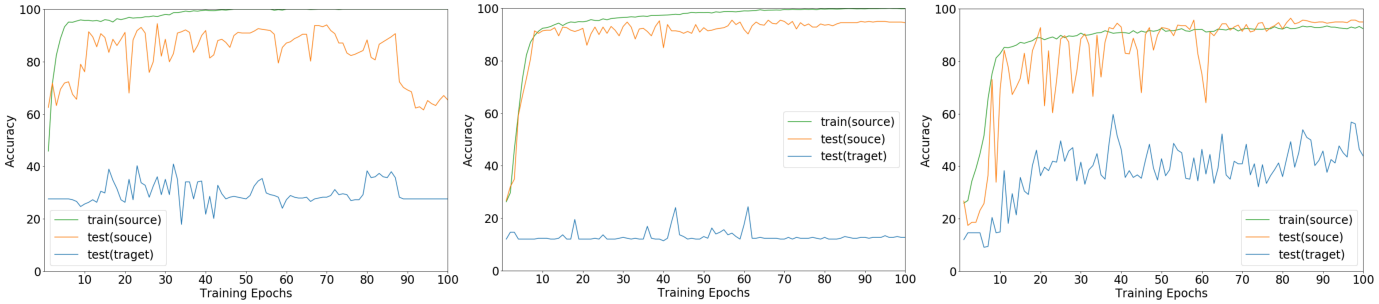
$$\lambda = 0.5 \cdot \left(\frac{1}{1 - 2 \cdot \gamma \cdot (p - 0.5)} - 0.5 \right) / \left(\frac{1}{1 - 2 \cdot \gamma \cdot 0.5} - 0.5 \right) + 0.5$$

ここで p は学習の進行率で, 0 から 1 まで線形に増加する. λ が 0 から 1 まで次第に増加することで, 学習の初期はドメイン分類損失よりクラス分類損失の方をより優先してモデルを更新させる. 実験により γ は 6 に決定した. 学習の進行に伴う λ の変化を図 4 に示す.

4.2　合成データのみでの学習

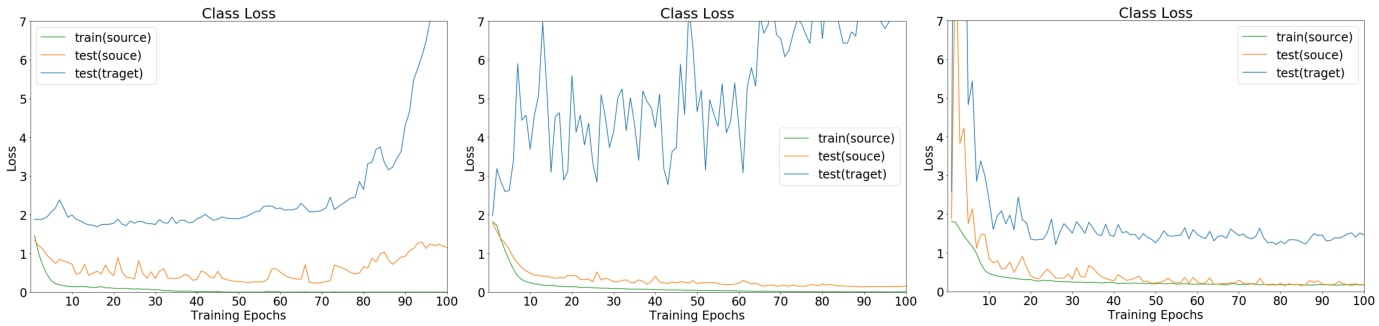
作成した Ochahouse-Dataset について, ドメイン適応なしで Ochahouse-Syn のみで学習したモデルでの Ochahouse-Real の解析性能を評価する. モデルには, 3D 畳み込みネットワークの 1 つである 3D ResNet-18 [28] を用いた. また, 合成データにノイズ, ぼかしの付与とカラージッターを行うデータ拡張を行い, 実データの解析精度向上への効果を確認する.

Ochahouse-Real で学習したモデルで Ochahouse-Real の動作分類を行った結果, 81.14 % で分類ができた. Ochahouse-Syn で学習したモデルで Ochahouse-Real の動作分類を行った精度を図 5 に示す. 図 5 では, none ではデータ拡張を行わず



(a) データ拡張なし　(b) ノイズ、カラージッター付与　(c) ぼかし、ノイズ、カラージッター付与

図 6　3D ResNet-18 での学習の様子 (精度)



(a) データ拡張なし　(b) ノイズ、カラージッター付与　(c) ぼかし、ノイズ、カラージッター付与

図 7　3D ResNet-18 での学習の様子 (ロス)

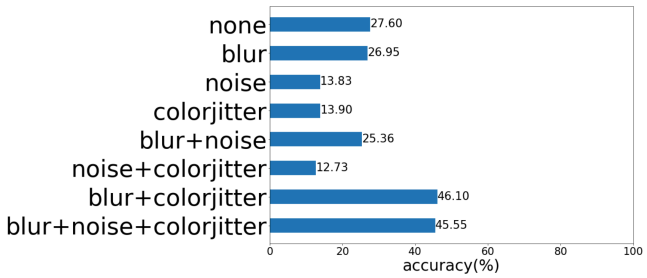


図 5　さまざまなデータ拡張を行った Ochahouse-Syn で学習したモデルでの Ochahouse-Real 解析精度．none はデータ拡張なし, blur はぼかし付与, noise はノイズ付与, colorjitter はカラージッターの付与を表す.

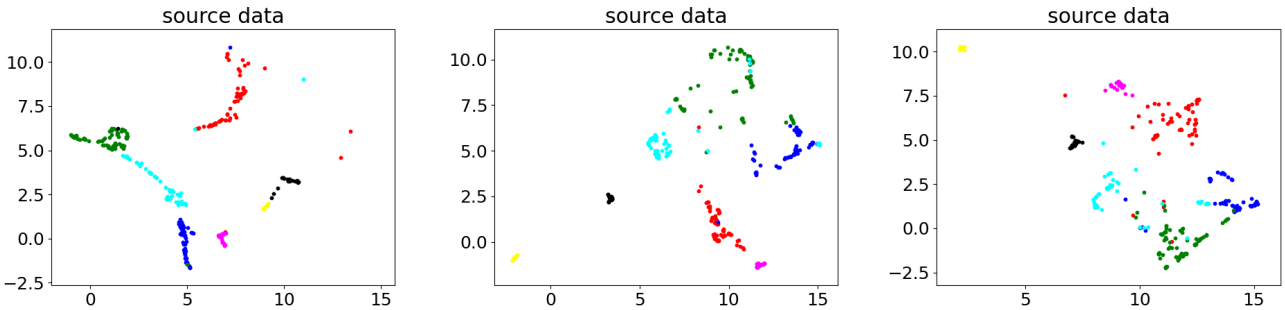
に Ochahouse-Syn で学習した場合, blur ではぼかしの付与を, noise ではノイズの付与を, colorjitter ではカラージッターの付与を行った Ochahouse-Syn で学習した場合の結果を示している．図 5 より, Ochahouse-Syn のみを用いた学習では 12.73% から 46.10% と, Ochahouse-Real で学習した時の 81.14%に対し解析精度が低いこと, ノイズを加えると実データ解析精度が低下するが, ぼかしとカラージッターの両方を施すと精度が改善することがわかった.

図 6, 図 7 では, (a) データ拡張なし, (b) ノイズ, カラージッター付与, (c) ぼかし, ノイズ, カラージッター付与のときの学習の様子を示す．横軸は学習エポックを, 縦軸はそれぞれ精度と損失を示している．緑線は学習時, オレンジ色の線は Ochahouse-Syn での検証時, 青線は Ochahouse-Real での検証時の精度または損失を表している．図 6,7 より, Ochahouse-Real の解析精度が低いノイズとカラージッターのデータ拡張を行った時も, 比較的精度が高いぼかし、ノイズ、カラージッターのデータ拡張を行った時も, Ochahouse-Syn の解析精度は十分に高く, Ochahouse-Syn の学習自体は問題なくできていることがわかる.

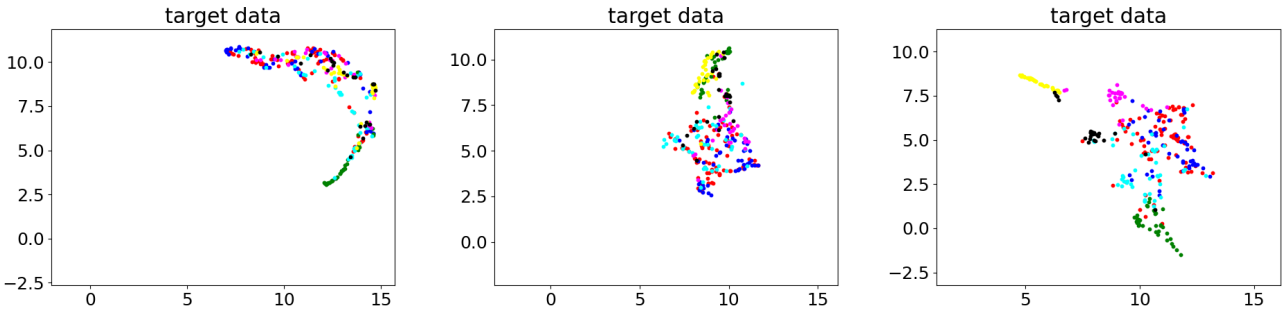
さらに, 3D ResNet-18 の pooling 層の前までを特徴抽出器とみなし, 学習された特徴抽出器で抽出した Ochahouse-Syn, Ochahouse-Real の特徴を UMAP [29] でプロットしたものを図 8 に示す．各色は赤が walking, 青が sitting down, 緑が sitting, シアンが standing up, マゼンタが lying down, 黄が lying, 黒が waking up の動作クラスを表している．Ochahouse-Syn と Ochahouse-Real を入力とした場合の結果を比較すると, いずれのデータ拡張で学習した場合も抽出された特徴の分布がほとんど一致していないことがわかる．ただし, ぼかし, ノイズ, カラージッターのの 3 つ全てを施すデータ拡張により形状が少し近くなったことが読み取れる．これらの結果から, Ochahouse-Real の解析精度が低い原因はドメインシフトであり, データ拡張によって改善がみられたが不十分なこと, ドメイン適応によってさらに改善する可能性があることがわかった.

4.3　ドメイン適応を行う学習

ドメイン適応を用いた手法で学習を行い, その効果を調査する．学習には 3 次元畳み込みネットワークである 3D ResNet-18 と DANN を組み合わせた図 9 のようなネットワーク 3D ResNet-18 based DANN を作成した．このネットワークでは, まず 3 次元畳み込みネットワークに基づいた特徴抽出器で空間方向・時間軸方向の 3 次元の次元削減を同時に行い動画像から



(a) データ拡張なし (Ochahouse-Syn) (b) ノイズ・カラージッター付与 (Ochahouse-Syn) (c) ぼかし・ノイズ・カラージッター付与 (Ochahouse-Syn)



(d) データ拡張なし (Ochahouse-Real) (e) ノイズ・カラージッター付与 (Ochahouse-Real))(f) ぼかし・ノイズ・カラージッター付与 (Ochahouse-Real)

図 8　3D ResNet-18 の特徴抽出器で抽出した特徴を UMAP で可視化した結果

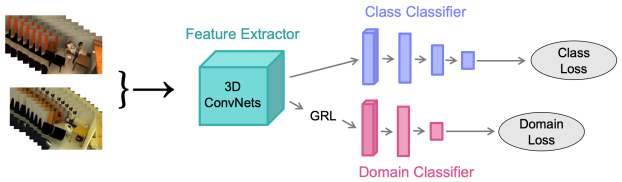


図 9　3D ResNet based DANN

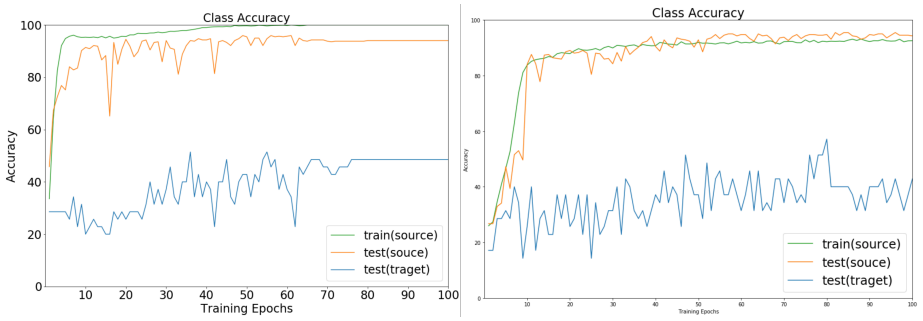
特徴抽出を行う．DANN と同様に，抽出された特徴は，クラス分類器と，勾配反転層 (GRL, Gradient Reverse Layer) を経てドメイン分類器とにそれぞれ提供され，クロスエントロピー誤差関数でクラス分類損失 L_y とドメイン分類損失 L_d が算出される．これらの加重和 $L_y + \lambda L_d$ を最適化することで，クラス間の違いは識別できるように，ドメイン間の違いは混合するようになり，ネットワークにドメイン間に共通する動作の特徴を学習させる．

4.1 節と同様に表 2 の条件でこのネットワークを使用して 7 クラス動作分類を行った結果を表 3 に示す．また，3D ResNet based DANN での学習の様子を図 10, 図 11 に示す．表 3 では (i) はラベル付き Ochahouse-Real で学習を，(ii)(iii) はラベル付き Ochahouse-Syn のみで (ドメイン適応せずに) 学習を，(iv)(v) はラベル付き Ochahouse-Syn とラベルなし Ochahouse-Real を用いてドメイン適応を適用して学習を行った時の Ochahouse-Real の解析精度を表している．表 3(ii)(iv) から，データ拡張を行わない場合はドメイン適応により解析精度が 21.49% 向上したが，(iii)(v) からデータ拡張を行う場合は解析精度が 6.98% 低下したことがわかる．図 10(a) から，ドメイン適応により精度が向上したデータ拡張なしでは Ochahouse-Syn の学習に伴い OChahouse-Real の解析精度も徐々に向上していること，(b) から精度が低下したデータ拡張ありでは終盤に精度が低下していることがわかる．

また，図 12 に特徴抽出器で抽出された特徴を UMAP で可視化したものを示す．図 12 から，いずれの場合でも Ochahouse-Syn と Ochahouse-Real の特徴の分布の形状が大きく異なっており Ochahouse-Syn と Ochahouse-Real のドメインは十分に混合できていないことがわかる．ただし，右下の青い点のように，一部においてはドメイン間共通の特徴の抽出に成功していた．ドメイン適応が十分でない原因について，図 11(a) のデータ拡張なしでの学習時の source loss が学習の後半に増えていることから，学習率の設定が適切でない可能性がある．また，クラス分類器とドメイン分類器の学習のスケジュールや損失関数の加重和の取り方など，各パラメータ調整が不十分であることなども考えられる．

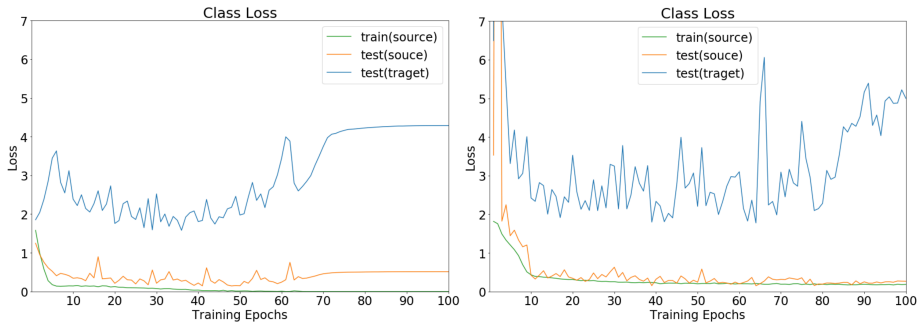
5　まとめと今後の課題

本研究では，ラベルなし実動画像データの解析に向けた合成動画像データ活用方法について検討した．まず，同様なシーンで人間の動作を収録した実動画像データ Ochahouse-Real と合成動画像データ Ochahouse-Syn を作成した．それらを用いた実験の結果，我々人間の目で見て大きな違いがないこれらのデータ



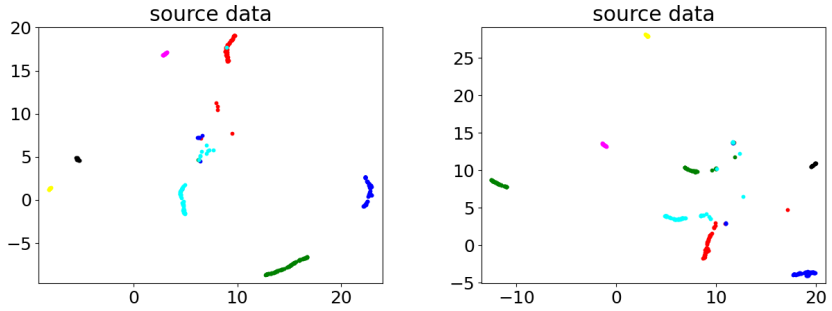
(a) データ拡張なし　　(b) ぼかし、ノイズ、カラージッター付与

図 10　3D ResNet-18 based DANN でドメイン適応を適用した学習の様子 (クラス分類精度)

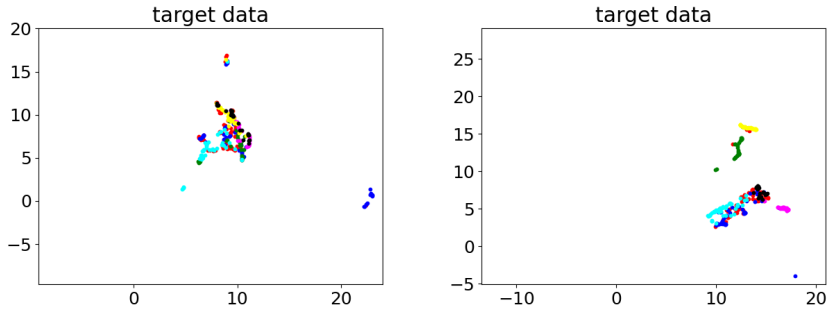


(a) データ拡張なし　　(b) ぼかし、ノイズ、カラージッター付与

図 11　3D ResNet-18 based DANN でドメイン適応を適用した学習の様子 (クラス分類ロス)



(a) データ拡張なし (Ochahouse-Syn)　　(b) ぼかし・ノイズ・カラージッター付与 (Ochahouse-Syn)



(c) データ拡張なし (Ochahouse-Real)　　(d) ぼかし・ノイズ・カラージッター付与 (Ochahouse-Real)

図 12　3D ResNet-18 based DANN の特徴抽出器で抽出した特徴を UMAP で可視化した結果

間の違いは今回採用した 3 次元畳み込みニューラルネットワークにとっては大きく, 合成データのみでの学習では十分な精度で実データの解析ができないことがわかった. また, データ拡張およびドメイン適応を用いた手法によって, Ochahouse-Real の解析精度が向上し, さらに改善できる可能性があることがわかった.

表 3　ドメイン適応を用いて学習したモデルでの実データ解析精度

	精度
(i) 3D ResNet-18 trained to Ochahouse-Real	81.14%
(ii) 3D ResNet-18	27.08%
(iii) 3D ResNet-18 + データ拡張	45.55%
(iv) 3D ResNet-18 based DANN	48.57%
(v) 3D ResNet-18 based DANN + データ拡張	38.57%

今後はドメイン適応を用いた手法でのパラメータチューニングを行い, 効果を確認する. また, 3 次元畳み込みだけでなく 2 ストリーム CNN や TSN を用いたり, DANN だけでなく様々なドメイン適応手法を用いて実験し, 効果的な合成データの活用方法について調査および検討を行う.

謝　　辞

この成果の一部は，JSPS 科研費 JP19H04089，JP19K11994 及び，2020 年度国立情報学研究所公募型共同研究 (20S0501) の助成を受けたものです.

文　　献

[1] Cheng, G., Wan, Y., Saudagar, A. N., Namuduri, K. and Buckles, B. P.: Advances in Human Action Recognition: A Survey, *ArXiv*, Vol. abs/1501.05964 (2015).

[2] Wu, D., Sharma, N. and Blumenstein, M.: Recent advances in video-based human action recognition using deep learning: A review, *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2865–2872 (online), 10.1109/IJCNN.2017.7966210 (2017).

[3] Takasaki, C., Takefusa, A., Nakada, H. and Oguchi, M.: A Study of Action Recognition Using Pose Data Toward Distributed Processing Over Edge and Cloud, *2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 111–118 (online), 10.1109/CloudCom.2019.00027 (2019).

[4] Sun, C., Shrivastava, A., Singh, S. and Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era, *Proceedings of the IEEE international conference on computer vision*, pp. 843–852 (2017).

[5] Sadeghi, F. and Levine, S.: CAD^{2}RL: Real Single-Image Flight without a Single Real Image, *ArXiv*, Vol. abs/1611.04201 (2016).

[6] Tobin, J., Fong, R. H., 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 (IROS)*, pp. 23–30 (2017).

[7] Gaidon, A., Wang, Q., Cabon, Y. and Vig, E.: VirtualWorlds as Proxy for Multi-object Tracking Analysis, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4340–4349 (2016).

[8] De Souza, C. R., Gaidon, A., Cabon, Y. and López, A. M.: Procedural Generation of Videos to Train Deep Action Recognition Networks, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2594–2604 (online), 10.1109/CVPR.2017.278 (2017).

[9] Vázquez, D., López, A. M., Marín, J., Ponsa, D. and Gerónimo, D.: Virtual and Real World Adaptation for Pedestrian Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 4, pp. 797–809 (online), 10.1109/TPAMI.2013.163 (2014).

[10] Xu, J., Ramos, S., Vázquez, D. and López, A. M.: Domain Adaptation of Deformable Part-Based Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 12, pp. 2367–2380 (online), 10.1109/TPAMI.2014.2327973 (2014).

[11] Sun, B. and Saenko, K.: From Virtual to Reality: Fast Adaptation of Virtual Object Detectors to Real Domains, *Proceedings of the British Machine Vision Conference*, BMVA Press (2014).

[12] Busto, P. P., Liebelt, J. and Gall, J.: Adaptation of Synthetic Data for Coarse-to-Fine Viewpoint Refinement, *Proceedings of the British Machine Vision Conference (BMVC)* (Xianghua Xie, M. W. J. and Tam, G. K. L., eds.), BMVA Press, pp. 14.1–14.12 (online), 10.5244/C.29.14 (2015).

[13] Chen, M.-H., Kira, Z., AlRegib, G., Yoo, J., Chen, R. and Zheng, J.: Temporal Attentive Alignment for Large-Scale Video Domain Adaptation, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019).

[14] Ballout, M., Tuqan, M., Asmar, D., Shammas, E. and Sakr, G.: The benefits of synthetic data for action categorization, *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8 (online), 10.1109/IJCNN48605.2020.9207337 (2020).

[15] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T. and Serre, T.: HMDB51: A Large Video Database for Human Motion Recognition, pp. 2556–2563 (online), 10.1109/ICCV.2011.6126543 (2011).

[16] Soomro, K., Zamir, A. R. and Shah, M.: UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild, *CoRR*, Vol. abs/1212.0402 (online), http://dblp.uni-trier.de/db/journals/corr/corr1212.htmlabs-1212-0402 (2012).

[17] Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X. and Van Gool, L.: Temporal Segment Networks: Towards Good Practices for Deep Action Recognition, *Computer Vision – ECCV 2016* (Leibe, B., Matas, J., Sebe, N. and Welling, M., eds.), Cham, Springer International Publishing, pp. 20–36 (2016).

[18] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V.: Domain-Adversarial Training of Neural Networks, *J. Mach. Learn. Res.*, Vol. 17, No. 1, p. 2096–2030 (2016).

[19] Tzeng, E., Hoffman, J., Saenko, K. and Darrell, T.: Adversarial Discriminative Domain Adaptation, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2962–2971 (online), 10.1109/CVPR.2017.316 (2017).

[20] Saito, K., Watanabe, K., Ushiku, Y. and Harada, T.: Maximum Classifier Discrepancy for Unsupervised Domain Adaptation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018).

[21] Pan, B., Cao, Z., Adeli, E. and Niebles, J. C.: Adversarial Cross-Domain Action Recognition with Co-Attention, *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34, No. 07, pp. 11815–11822 (online), 10.1609/aaai.v34i07.6854 (2020).

[22] Choi, J., Sharma, G., Schulter, S. and Huang, J.-B.: Shuffle and Attend: Video Domain Adaptation, *Computer Vision – ECCV 2020* (Vedaldi, A., Bischof, H., Brox, T. and Frahm, J.-M., eds.), Cham, Springer International Publishing, pp. 678–695 (2020).

[23] : OchaHouse Project Page, `http://is.ocha.ac.jp/~siio/index.php?OchaHouse`.

[24] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang,

L., Bai, J. and Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library, *Advances in Neural Information Processing Systems 32* (Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E. and Garnett, R., eds.), Curran Associates, Inc., pp. 8024–8035 (online), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf (2019).

[25] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Bengio, Y. and LeCun, Y., eds.), (online), http://arxiv.org/abs/1412.6980 (2015).

[26] Smith, L. N. and Topin, N.: Super-convergence: Very fast training of neural networks using large learning rates, *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, Vol. 11006, International Society for Optics and Photonics, p. 1100612 (2019).

[27] Carlson, A., Skinner, K. A., Vasudevan, R. and Johnson-Roberson, M.: Modeling Camera Effects to Improve Visual Learning from Synthetic Data, *ECCV Workshops* (2018).

[28] Tran, D., xiu Wang, H., Torresani, L., Ray, J., LeCun, Y. and Paluri, M.: A Closer Look at Spatiotemporal Convolutions for Action Recognition, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6450–6459 (2017).

[29] McInnes, L., Healy, J., Saul, N. and Grossberger, L.: UMAP: Uniform Manifold Approximation and Projection, *The Journal of Open Source Software*, Vol. 3, No. 29, p. 861 (2018).