機械知覚&ロボティクスグループ
中部大学 工学研究科 ロボット理工学専攻

MPRG
MACHINE PERCEPTION AND ROBOTICS GROUP

2020年度　藤吉研究室　修士論文発表　アブストラクト

Style transfer　Test-time Augmentation　Domain Adaptation

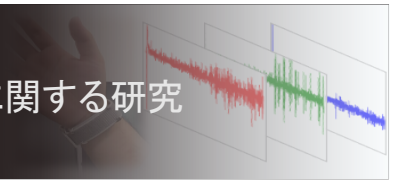
クラス情報を考慮したスタイル変換によるテストタイム拡張の高精度化に関する研究

今枝 航

Deep Learning　Neural Signal　Operation Identific

1D Self-Attention Networkによる神経信号からの動作識別に関する研究

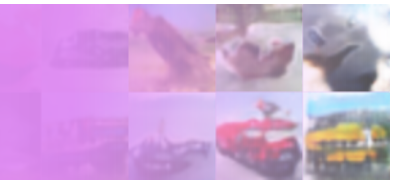
田邊 稜



Deep Learning　Generative Model　Collaborative Learning

Generative Adversarial Networksの共同学習に関する研究

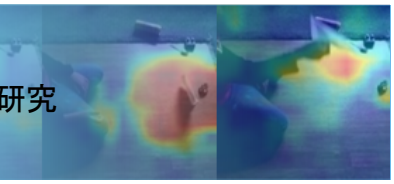
塚原 拓也



Deep Learning　Visual Explanation　Attention Mechanism

動画像における時空間情報を考慮したDNNによる視覚的説明に関する研究

三津原 将弘



クラス情報を考慮したスタイル変換によるテストタイム拡張の高精度化に関する研究

TR190002 今枝 航　　指導教授：藤吉 弘亘

1. はじめに

深層学習によるスタイル変換は，入力画像が持つスタイルを異なるスタイルへ変換するアプローチである．スタイル変換を用いて異なるドメインの画像に変換することで，物体認識のドメイン適応が可能となる．スタイル変換を用いたドメイン適応の代表的な手法である CyCADA[3] は，画像のクラスが一貫した変換は可能であるが，入力画像に対して生成できる変換画像は 1 枚である．

本研究では，スタイル変換の際にクラス情報を考慮して複数の変換画像を生成できるスタイル変換手法を提案する．さらに，画像の認識時にデータ拡張を行うテストタイム拡張に提案手法を用いることで高精度化が期待できる．提案手法は，クラス情報を考慮しながら様々なスタイル変換が可能となり，高いアンサンブル効果を実現できる．

2. 従来研究

2.1 スタイル変換

スタイル変換は，異なるスタイルを持つ画像間において，一方向もしくは双方向に画像変換するアプローチである．スタイル変換の代表的な手法として，CycleGAN[1] がある．CycleGAN は，変換先のペア画像を必要としない変換手法であり，2 つのスタイル間で双方向に画像変換できる．また，3 つ以上のスタイルを変換する手法として，StarGAN v2[2] がある．StarGAN v2 は入力画像とは異なる画像，もしくはガウス分布に基づく潜在変数からスタイルコードを生成し，スタイルコードに埋め込まれたスタイルになるように入力画像を変換する．

2.2 ドメイン適応

ドメイン適応は，教師ラベルを持たないターゲットドメインと，教師ラベルを持つソースドメインの異なるドメイン間のギャップを軽減するように学習することで，ターゲットドメインでの識別精度を向上させる技術である．スタイル変換によるドメイン適応として Cycle-Consistent Adversarial Domain Adaptation(CyCADA)[3] が提案されている．CyCADA は，CycleGAN にクラス情報を考慮する Semantic consistency を導入している．これにより，ソースドメインで事前学習した識別器を用いて，変換前後の画像におけるクラス情報が一致するように学習できる．

3. 提案手法

本研究では，StarGAN v2 での画像変換時にクラス識別を行う補助タスクを追加することにより，スタイル変換時にクラス情報を考慮する．補助タスクには，CyCADA の Semantic consistency を用いる．これにより，本手法はクラス情報を考慮しながら複数の画像へ変換ができる．本手法を評価時にデータ拡張するテストタイム拡張に適用することで，高精度な識別が可能となる．

3.1 クラス情報を考慮した StarGAN v2

提案手法の構造を図 1 に示す．提案手法は，ソース画像 $\boldsymbol{x}_s$ とターゲット画像 $\boldsymbol{x}_t$ に対して以下の処理を順に行う．

Step1 潜在変数 $\boldsymbol{z}$ より生成されるスタイルコード $\tilde{s}_t$ を用いたスタイル変換．

Step2 実画像 x_s と変換画像 $\tilde{x}_t$ の真贋判定．

Step3 実画像のスタイルコード $\hat{s}_s$ を用いた画像 $\tilde{x}_t$ の再構成．

Step4 実画像 x_s と変換画像 $\tilde{x}_t$ のクラス識別．

ここで，Generator を Encoder-Decoder モデルとしたとき，入力画像が持つコンテキストを Encoder で抽出，Decoder でスタイル情報の付与を行うため，Generator の中間特徴はドメイン不変の特徴量となる．そのため，Generator の中間にクラス識別器を追加することにより，両方のドメインにおいて十分な識別性能が期待できる．また，追加したクラス分類器は画像変換時には用いない．

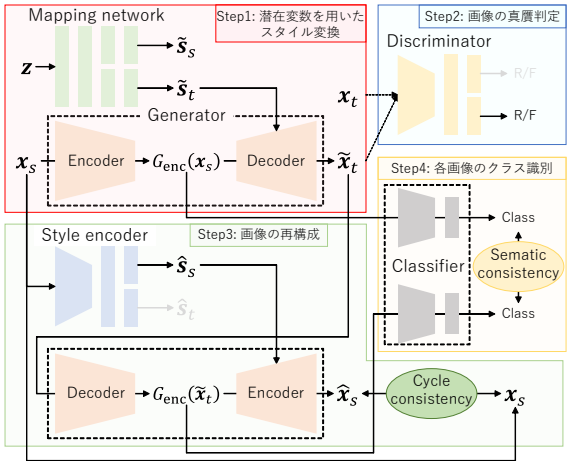


図 1：提案手法の構造

3.2 誤差関数

誤差関数は，クラス識別器を学習する $\mathcal{L}_{task}$ と，スタイル変換におけるクラス情報の一貫性を考慮する $\mathcal{L}_{sem}$ を用いる．各誤差関数は式 (1)，式 (2) と定義する．

$$\mathcal{L}_{task} = \mathrm{CE}(C(G_{\mathrm{enc}}(\boldsymbol{x}_s)), t_s) \tag{1}$$

$$\begin{aligned}\mathcal{L}_{sem} = {} & \mathrm{CE}(C(G_{\mathrm{enc}}(G(\boldsymbol{x}_s, \boldsymbol{s}_t)), C(G_{\mathrm{enc}}(\boldsymbol{x}_s)) \\ & + \mathrm{CE}(C(G_{\mathrm{enc}}(G(\boldsymbol{x}_t, \boldsymbol{s}_s)), C(G_{\mathrm{enc}}(\boldsymbol{x}_t))\end{aligned} \tag{2}$$

ここで，$\boldsymbol{x}$ が入力画像，t_s がソースドメインの教師ラベル，C がクラス分類器，G が Generator，$\boldsymbol{s}$ がスタイルコード，CE がクロスエントロピー誤差である．また，添え字は各ドメインを表しており，s がソースドメイン，t がターゲットドメインを表している．$\mathcal{L}_{task}$ は，教師ラベルを持つソースドメインの画像のみで誤差を計算する．$\mathcal{L}_{sem}$ は，クラス分類器より出力される変換前のクラス確率を仮ラベルとし，変換後のクラス確率とのクロスエントロピー誤差を計算する．これにより，画像の変換前後でクラス情報の一貫性を考慮した学習が可能となる．

3.3 テストタイム拡張

テストタイム拡張は，識別時テストサンプルに対してデータ拡張を行い，各画像に対するクラス確率を平均して最終的な予測結果を決定する．本手法では，ソースドメインで事前学習した識別器へターゲットドメインの画像を入力する際に，スタイル変換によるドメイン適応を行う．同時に，潜在変数を複数回サンプリングし，様々なスタイルの画像へ変換する．そして，従来と同様に式 (3) で定義するアンサンブル推論にて最終的な予測結果を決定する．

$$p_i = \frac{1}{N}\sum_{n=1}^{N} \sigma_i(f(G(\boldsymbol{x}, F(\boldsymbol{z}_n)))) \tag{3}$$

ここで，i がクラス番号，p が予測結果，N が増幅数，σ が Softmax 関数，f が事前学習された識別器，F が潜在変数よりスタイルコードを生成する Mapping network，$\boldsymbol{z}$ が潜在変数である．

4. 評価実験

提案手法の有効性を示すために，変換画像の傾向調査とテストタイム拡張における識別性能の比較，および変換画像の分布調査を行う．評価実験に使用するデータセットは，数字認識用のデータセットである SVHN と SynthDigits を用いる．ここで，ソースドメインには，CG 画像である SynthDigits，ターゲットドメインは実画像である SVHN とし，クラス数は 10 である．また，テストタイム拡張における増幅数は 10 とする．

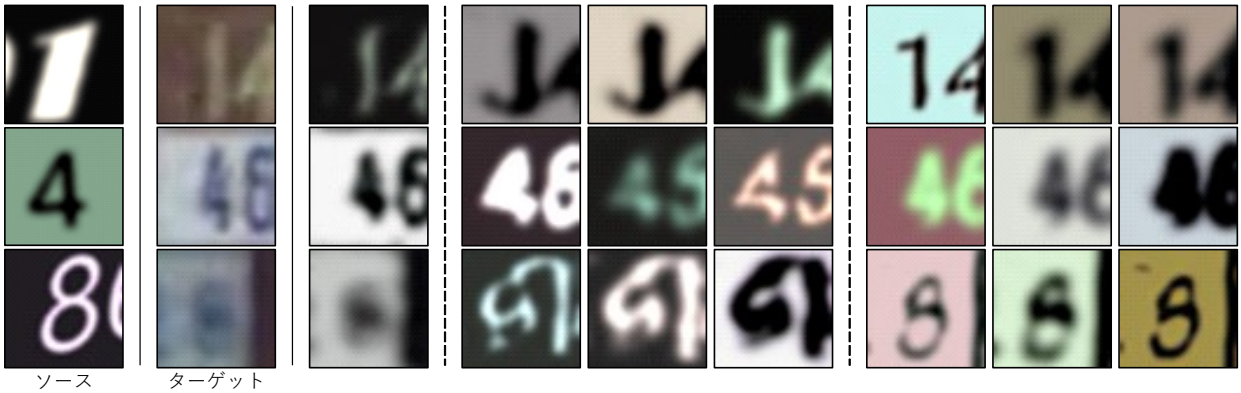
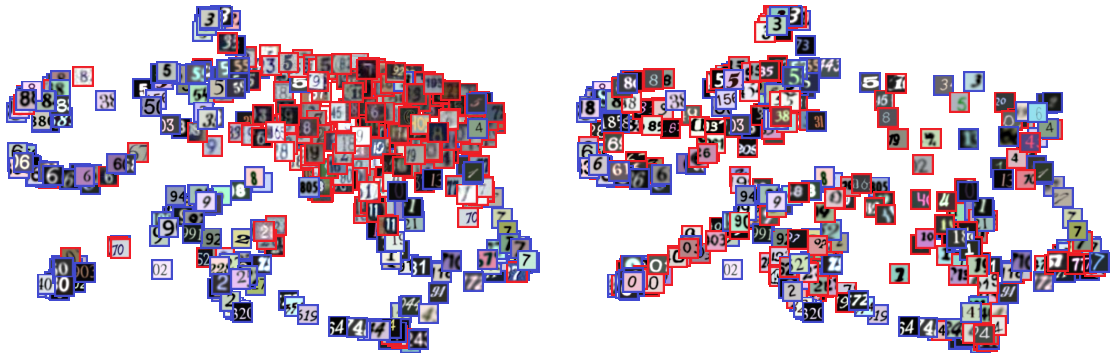


図 2：各手法による変換結果



(a) スタイル変換前の分布　　(b) スタイル変換後の分布

図 3：UMAP による分布の可視化

表 1：テストタイム拡張における識別精度

変換手法	増幅数	識別率 [%]	
		ResNet-20 (701.4k)	CNN (24.3k)
変換無し	-	74.88	66.54
ベースライン (StarGAN v2)	1	77.48	76.44
	10	79.85	79.50
提案手法	1	87.73	85.77
	10	**89.21**	**87.91**

4.1 変換画像の定性的評価

提案手法および従来手法 (CycleGAN，StarGAN v2) を用いて変換した結果を図 2 に示す．これより，従来手法ではクラス情報の維持が困難であり，変換元であるターゲットドメインの画像が低画質であった場合はその傾向が顕著である．一方，提案手法による変換は，クラス情報を維持した状態で異なるスタイルの画像へ変換できていることが分かる．

4.2 テストタイム拡張を用いた定量的評価

テストタイム拡張に使用する事前学習済み識別器には，ResNet-20 と，ドメイン間の差異を吸収することが困難な 5 層の CNN を用いて評価実験を行う．テストタイム拡張による識別精度を表 1 に示す．ここで，各ネットワークのパラメータ数はモデル名の下に示す．表 1 より，ResNet-20 の結果において，提案手法は変換無しと比較して識別精度の向上が確認できる．また，増幅数 1 では，提案手法による識別精度はベースラインの StarGAN v2 と比較し，10.25pt の精度向上を確認した．さらに，増幅数 10 では，増幅数 1 と比較して 1.48pt の精度向上を確認した．この傾向は CNN の識別結果においても同様である．

4.3 UMAP による変換画像の分布の調査

各ドメインの画像と提案手法で変換した画像を UMAP で次元圧縮し，可視化した結果を図 3 に示す．ここで，青枠がソースドメイン，赤枠がターゲットドメインを表している．次元圧縮に使用する画像枚数は，各クラスからランダムに 25 枚選択し，各ドメイン 250 枚とする．また，圧縮する特徴には定量的評価で使用した CNN の中間特徴を用いる．図 3(a) はスタイル変換を行っていないため，ドメインごとの分布が異なっている．一方，スタイル変換を行った図 3(b) は各クラスの分布がほぼ一致している．

5. おわりに

本研究では，クラス情報を考慮したスタイル変換手法および，テストタイム拡張におけるスタイル変換を用いたデータ拡張を提案した．評価実験において，提案手法によるスタイル変換は，クラス情報を考慮した変換ができていることを確認した．また，テストタイム拡張による定量的評価では，変換無しと比較して，14.33pt の精度向上を確認した．今後は，セマンティックセグメンテーションなどの異なるタスクへの適応を検討する．

参考文献

[1] J.Y.Zhu, *et al.*, “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”In ICCV, 2017.

[2] Y.Choi, *et al.*, “StarGAN v2: Diverse Image Synthesis for Multiple Domains”, In CVPR, 2020.

[3] J.Hoffman, *et al.*, “CyCADA: Cycle Consistent Adversarial Domain Adaptation”, In ICML, 2018.

研究業績

[1] 今枝航 等, “Attention 機構を導入した CycleGAN による識別に有効なスタイル変換”, 画像の認識・理解シンポジウム, 2019.

[2] 今枝航 等, “Generative Adversarial Networks を用いたからあげピッキングにおける前処理としての画像変換”, ビジョン技術の実利用ワークショップ, 2020.

1D Self-Attention Network による神経信号からの動作識別に関する研究

TR19005 田邉　稜　　指導教授：藤吉弘亘

1. はじめに

人物の動作識別は，画像ベースの識別法 [1] と信号ベースの識別法 [2] に大別できる．画像ベースの手法は，RGB カメラから取得した画像を用いるため，装着型計測デバイスを必要とせず，非接触で動作の識別が可能である．しかし，物を掴むなどの物体に対して手先のみで行う動作は，オクルージョンが発生すると識別が困難となる．一方，信号ベースの手法は，人体に装着したデバイスから筋電信号を用いるため，オクルージョンが発生しないというメリットがある．手先の動作識別を行うためには，手先の動作に深く関わる信号を計測する必要がある．そこで本研究では，人体の手首に位置し，手の筋肉を支配する手根管神経から計測可能な神経信号に着目する．神経信号の時系列データから動作固有の特徴を捉えるために，1D-Self-Attention Block を提案する．

2. 従来研究

信号ベースの従来手法として，Morbidoni らは表面筋電図 (sEMG) を階層型ニューラルネットワークに入力して，人物のスタンスフェーズとスイングフェーズの動作識別を行う手法を提案している [3]．また，Ozal らは人間の心電図からフラグメントと呼ばれる小領域をサンプリングし，CNN を適応することで 17 クラスの心不全を検出する手法を提案している [2]．これらの従来手法は，クラス識別のための特徴抽出に多層パーセプトロンや畳み込みニューラルネットワークを用いている．しかし，畳み込み層は，近傍の時刻に着目するため，離れた時刻の重要な変化を捉えることに適していない．

3. 提案手法

人物の手先の動作は，その種類により動作時間が異なる．そのため，動作の短期的および長期的な変化を捉える事が重要となる．本研究では，時系列データにおいて，周辺時刻を考慮した特徴抽出が可能な 1D-Self-Attention Block(1D-SAB) により構成される 1D-Self-Attention Networks(1D-SAN) を用いた動作識別法を提案する．

3.1. 手根管神経

本研究では，人物の手先の動作を対象とするため，人体の手首に位置し，手の筋肉を支配する手根管神経から計測可能な神経信号を用いる．手根管神経には，橈骨神経 (Radial narve)，正中神経 (Median narve)，尺骨神経 (Ulnar narve) がある．これらの神経信号を Mudra wearable device により計測する．サンプリング間隔は 1ms とする．

3.2. 1D Self-Attention Networks(1D-SAN)

本研究では，Self-Attention Networks(SAN)[4] で用いられる Self-Attention Block(SAB) を 1 次元データに対応させた，1D-SAB を提案し，1D-SAN を構成する．提案する 1D-SAB の構造を図 1 に示す．1D-SAB には，手根管神経から取得した時系列信号を入力する．入力した信号は，1 時刻ごとに処理し，対応する時刻の Self-Attention を算出する．図 1 における青色の値を処理の注目時刻とした時，赤色を近傍時刻 1 , 緑色を近傍時刻 2 とする，各近傍時刻に対して，注目時刻 (青) との Pointwise conv 処理を行う．また，注目時刻と近傍時刻を，ϕ および ψ の学習可能な関数 (1D-CNN) へ入力し，関係関数 δ を求める．関係関数 δ の出力は，関数 ϕ の出力から関数 ψ の出力を減算したものであり，マッピング関数 γ によって 1 つ目の処理の出力とチャンネル数を合わせる．その後，2 つの出力の要素積を算出する．この処理を近傍時刻分行い総和する．生成した Self-Attention Map(SAM) は Pointwise conv 処理により入力チャネル数と同じチャネルにする．この出力に，ステップ機構として入力信号を加算し，最終出力とする．1D-SAB を用いることで，神経信号内の重要な位置に大きな重みを与えることができ，離れた信号値との関係性を考慮できる．

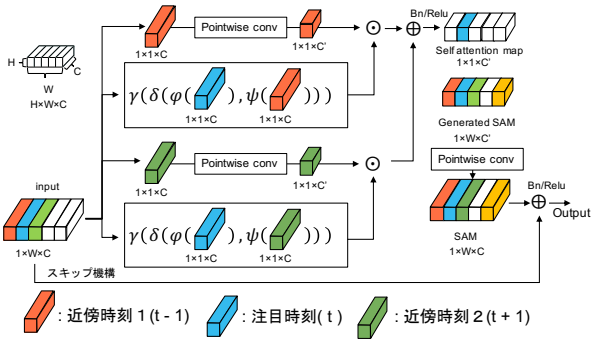


図 1：1D Self-Attention Block の構造

3.3. ネットワークの構造

本研究では，3 層の 1D-SAB からなる 1D-SAN を用いる．一層目の 1D-SAB における近傍時刻の間隔を 3 時刻，2，3 層目の 1D-SAB における近傍時刻の間隔を 5 時刻とする．これにより，離れた時刻の信号との変化を考慮できる．1D-SAN の構造と各 1D-SAB の近傍設定を図 2 に示す．

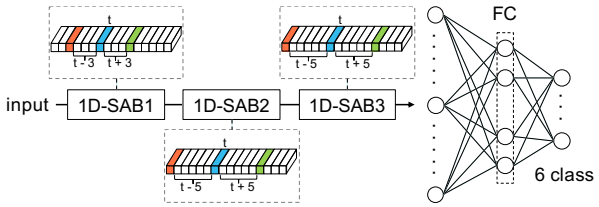


図 2：ネットワーク構造と近傍

3.4. 動作識別の流れ

本研究での動作識別は，3 つの Step となる．Step 1 は，Mudra wearable device を用いて取得した神経信号を一定区間ごとに固定長でサンプリングし，1D-SAN で推論ができるデータへ整形する．Step 2 は，学習済み 1D-SAN に，識別対象のデータの 0 時刻を起点として，入力するデータの領域を 10 時刻ずつ繰り返し入力し，動作識別する．Step 3 は，各区間の動作識別結果を統合して最終的な動作識別結果を出力する．

Step 1：前処理

本研究で扱う神経信号波形は，可変長のデータである．1D-SAN で推論を行うために，64 時刻分の 1 次元波形をサンプリングする．

Step 2：サンプリング時刻での動作識別

動作識別は，10 時刻ずつサンプリング区間をずらしながら 1D-SAN に入力する．

Step 3：動作識別結果の統合

各時刻におけるネットワークの出力をクラス毎に累積する．累積値が最大のクラスを動作識別結果とする．識別結果累積処理を図 3 に示す．

4. 評価実験

提案手法の有効性を示すために，1 次元波形に対応した畳み込みニューラルネットワーク (1D-CNN) と認識精度の比較を行う．また，フーリエ変換し，周波数空間を入力とした際の精度とも比較する．

4.1. データセット

本研究では，6 クラスの動作に対する神経信号を 100 回計測した 600 サンプルを用いる，これらを 6:4 で学習および評価データに分割して用いる．クラスのデータ取得時の

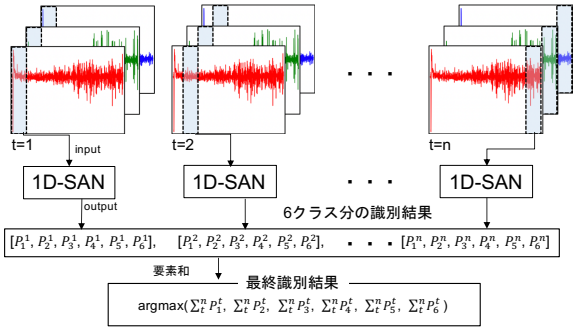


図 3：最終処理判定

画像を図 4 に示す．データは，Mudra wearable device を被験者の右手首に装着し，動作中の 1 次元波形を収集した．図 5 に動作 Push に対する神経信号を示す．

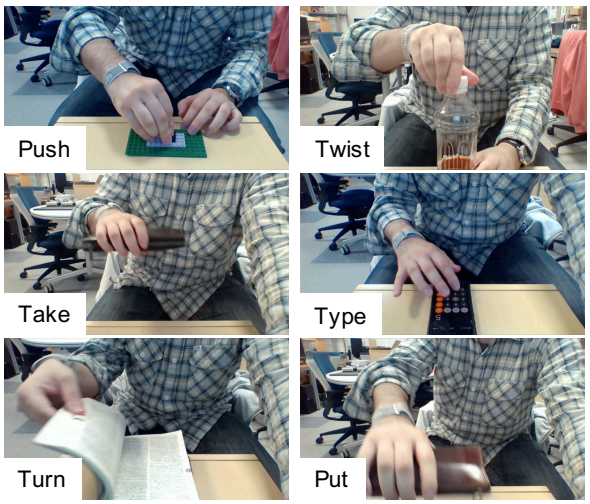


図 4：対象動作クラス

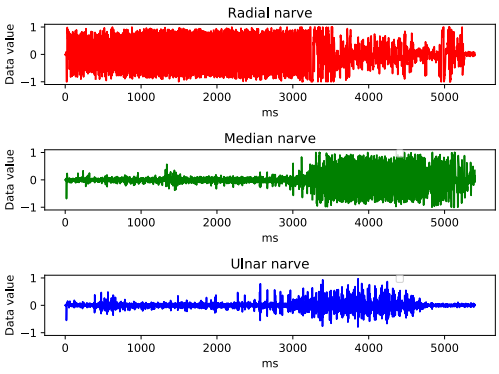


図 5：信号データサンプル (Push)

4.2. 実験概要

1D-CNN を用いたモデルと 1D-SAB を用いたモデルの精度比較を行う．1D-CNN は，2 層の畳み込み層と出力層の計 3 層の構造である．学習は，Epoch 数を 20000，BatchSize を 128，サンプリング幅を 64 とする．Optimizer は Momentum SGD とし，学習率を 0.001 ，Loss 関数に CrossEntropy Loss を用いる．

4.3. 実験結果

表 1 に各手法の Confusion Matrix を示す．比較実験の結果，1D-CNN の動作識別精度は 90.4 %，提案手法は 91.2 %となり，提案手法が動作識別で有効であることがわかった．提案手法は，Turn，Twist，Type の精度が向上し，Take，Push の精度が低下した．1D-CNN では，Twist を Turn と誤識別したが，提案手法では，Turn への誤識別は減少した．一方で提案手法は，Push を Twist と誤識別することが多い．Push に対して提案手法の精度が低下した要因は，物を押し込む動作をする際に，力を込めて強く押し込むケースがあり，Push でありながら Twist のように振動する信号が計測されたことが原因と考えられる．このようなケースを学習データにも追加する必要がある．また，Turn や Twist など，時系列的な変化が大きなクラスに対する精度が向上していることから，提案手法は信号値が時系列的に大きく変化するクラスに有効である．

表 1：Confusion Matrix による識別結果の比較

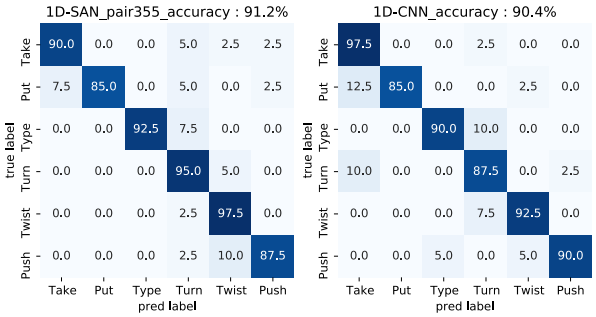


表 2 に，周波数領域における各手法の精度を示す．提案手法と 1D-CNN で比較を行った結果，周波数領域では，1D-SAN の動作識別精度が 80.0 %，1D-CNN の動作識別精度が 90.4 %となり，1D-SAN の精度が下回った．一方でフーリエ変換なしでは，1D-SAN の識別精度は 91.2 %と向上した．1D-SAN は，相対的な位置関係とその重要性を Self-Attention により表現できるためフーリエ変換なしで動作識別が可能である．

表 2：周波数領域における精度比較

ネットワーク	フーリエ変換あり (%)	フーリエ変換なし (%)
1D-SAN	80.0	**91.2**
1D-CNN	90.4	90.4

5. おわりに

本研究では，手先の筋肉を支配する手根管神経から取得した神経信号からの動作識別を行うために，1 次元波形に対応した 1D Self-Attention Network を提案した．精度比較実験では，従来手法の 1D-CNN に対して，提案手法が約 0.8 ポイント上回り，動作識別において，提案手法が有効であることを示した．また，Push のような神経信号値が連続的に大きくなるクラスでは，精度が低下したが，Turn や Twist など，時系列的な変化が捉えやすいクラスに関して精度が向上した．

参考文献

[1] T. Xiao,*et al.*, “Reasoning About Human-Object Interactions Through Dual Attention Networks” ,ICCV, 2019.

[2] O. Yildirim, *et. al.* , “Arrhythmia detection using deep convolutional neural network with long duration ECG signals”, Computers in Biology and Medicine, vol102, 2018, p411 420.

[3] C.Moribidoni,*et al.*, “A Deep Learning Approach to EMG-Based Classification of Gait Phases during Level Ground Walking” ,MDPI, 2019.

[4] Hengshuang.Zhao,*et al.*, “Exploring Self attention for Image Recognition” ,CVPR, 2020.

研究業績

[1] 田邉稜 等, “Self-Attention Networks による神経信号からの動作識別”, ビジョン技術の実利用ワークショップ, 2020.

Generative Adversarial Networks の共同学習に関する研究

TR19006 塚原拓也　　指導教授：藤吉弘亘

1. はじめに

Generative Adversarial Networks（GANs）は Generator および Discriminator を敵対的に学習することで，実在しない画像を生成する手法である．Generator は，潜在変数を用いて画像を生成し，生成画像が Discriminator により実画像と識別されるように学習する．Discriminator は，実画像と生成画像を識別できるように学習する．GANs の学習時，Generator の生成画像に偏りが生じると，Discriminator は実画像と生成画像を簡単に識別でき，モード崩壊を起こすことがある．この問題を解決するために，複数の Generator を用いることで，モード崩壊の発生を抑制する手法が提案されている [2]．しかし，この手法は Generator が独立して画像を生成するため，生成画像に偏りが生じる問題は解決していない．そこで，本研究では生成画像の偏りを抑制するために，複数の Generator が知識を転移しながら共同学習して画像を生成する手法を提案する．

2. 関連研究

従来手法である Deep Convolutional GANs および Multi-Agent Diverse GANs について述べる．

Deep Convolutional GANs（DCGANs）[1] は，畳み込み層を用いることで生成画像の質を向上させた GANs の手法である．実画像として正面の顔画像のみのデータ等の類似したデータのみ扱う場合においては，実画像と区別ができない鮮明な画像を生成できる．学習する際にモード崩壊が発生する場合が多く，安定した学習が困難である．

Multi-Agent Diverse GANs（MAD-GANs）[2] は，複数の Generator および 1 つの Discriminator で構成された GANs の手法である．Discriminator は実画像と生成画像を識別するのみならず，どの Generator から生成された生成画像であるかを識別するように学習する．MAD-GANs は，1 つ Generator が生成する画像に偏りが生じた場合，他の Generator は偏った画像とは異なる画像を生成しており，Discriminator は実画像と生成画像を簡単には識別できない．これにより，MAD-GANs を用いることでモード崩壊の発生を抑制することが可能となる．しかし，生成画像の偏りが生じること自体を防ぐことはできない．

3. 提案手法

本研究では複数の Generator を用いて学習を行い，Generator が互いの知識を転移しながら共同学習して画像生成を行う GANs の手法を提案する．

3.1 GANs の共同学習

提案手法のネットワーク構造を図 1 に示す．学習に使用する Generator の数を k とすると，各 Generator は $G_1, \ldots, G_k$ で表される．各 Generator は，潜在変数 $\mathbf{z} \sim P_{\mathbf{z}}$ を用いて生成画像を生成する．そのため，k 枚の生成画像が得られる．また，各 Generator の特徴マップまたは生成画像を知識転移グラフを用いて転移させることで，Generator 同士での知識の伝達を可能にしている．Discriminator は D で表され，実画像と生成画像を識別するのみならず，どの Generator から生成された生成画像であるかを識別する．そのため，最終層の出力値のユニットの数は $k+1$ 個となっており，ソフトマックス関数を用いてクラススコア $d_1, \ldots, d_{k+1}$ を出力する．スコア $d_1, \ldots, d_k$ は各 Generator から生成された生成画像である確率を表し，スコア d_{k+1} は実画像である確率を表す．ネットワーク内部には，学習の安定化を図るため，Batch Normalization が用いられる場合が多い．しかし，GANs における学習の安定化には Batch Normalization は不十分である．そこで，本手法では Discriminator に Spectral Normalization（SN）を導入する．これにより，損失関数に変更を加えることなくモード崩壊を抑制し，GANs における学習の安定化を図ることが可能となる．

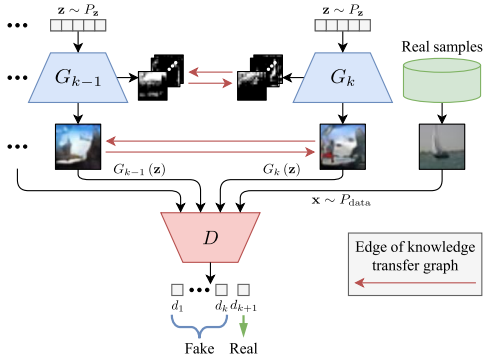


図 1: 提案手法のネットワーク構造

3.2 知識転移グラフによる最適化

提案手法は，知識転移グラフを導入することで，各 Generator が互いに知識転移しながら学習する．3 つの Generator を用いた場合における知識転移グラフを導入した提案手法を図 2 に示す．学習に使用する Generator の数を k とすると，各 Generator は $G_1, \ldots, G_k$，Discriminator は D となり，ノードとして表す．各 Generator のノード間には 2 つのエッジを定義し，それぞれ異なる向きの有向グラフで表す．各エッジの向きは，学習時に勾配の情報が伝わる方向を表している．各 Generator のノード間のエッジは Generator 同士の知識の伝達を表し，i 番目の Generator のエッジ $L_{D,i}$ は i 番目の Generator が生成した生成画像が Discriminator に実画像と判断されたかどうかの情報の伝達を表す．また，エッジ $L_{\text{adversarial}}$ は，Discriminator の識別結果が正しいかどうかの情報の伝達を表す．各エッジには個別の損失関数を定義し，各 Generator の方向を向いたエッジの損失関数にはゲート関数を導入することで，伝播される損失を制御する．本手法では，ゲート関数として Through Gate，Cutoff Gate，Negative linear Gate および Positive linear Gate を用いる．Through Gate は，入力されたサンプルごとの損失をそのまま通す関数である．Cutoff Gate は損失を 0 とし，損失の計算を行わない関数であり，任意のエッジを切断することができる．Negative linear Gate は学習が進むにつれて損失を減少させる関数であり，学習の序盤は Through Gate，終盤は Cutoff Gate と同じ処理となる．Positive linear Gate は学習が進むにつれて損失を増加させる関数であり，学習の序盤は Cutoff Gate，終盤は Through Gate と同じ処理となる．

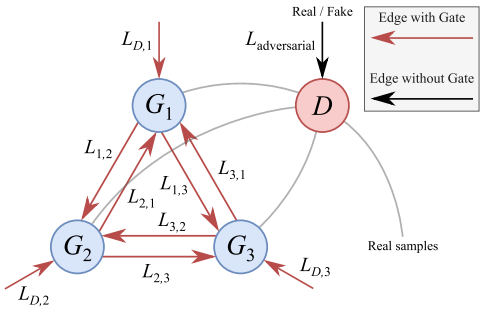


図 2: 複数の Generator における知識転移グラフ（$k = 3$）

3.3 学習方法

提案手法では，Discriminator の学習および Generator の学習を交互に繰り返すことで GANs の学習を行う．また，Discriminator の学習は MAD-GANs と同様である．

Generator の学習は，各 Generator の生成画像が Discriminator により実画像と識別されるように行う．また，提案手法では知識転移グラフに応じて，各 Generator が互いの知識を転移しながら学習する．各 Generator のノード間のエッジは Generator 同士の知識の伝達を表し，i 番目の Generator のエッジ $L_{D,i}$ は i 番目の Generator が生成

した生成画像が Discriminator に実画像と判断されたかどうかの情報の伝達を表す．Generator のノード間のエッジで，t 番目の Generator の知識を s 番目の Generator に転移する際の損失関数を $L_{t,s}$ とし，i 番目の Generator が生成した生成画像が Discriminator に実画像と判断されたかどうかを表す損失関数を $L_{D,i}$ とする．また，各エッジにおけるゲート関数は $Gate(\cdot)$ とする．Generator の学習では，各 Generator に入力される潜在変数 $\mathbf{z} \sim P_{\mathbf{z}}$ は同一とする．最終的な各 Generator の損失関数はエッジから得られた $L_{t,s}$ および $L_{D,i}$ を総和して算出する．使用する Generator の数を k とすると，Discriminator に対して i 番目の Generator が生成した生成画像を入力した際の識別結果は $\mathbf{d}^{(i)} = d_1^{(i)}, \ldots, d_{k+1}^{(i)}$ で表される．また，正解ラベルは $\mathbf{t}^{\mathrm{g}} = t_1^{\mathrm{g}}, \ldots, t_{k+1}^{\mathrm{g}}$ で表され，t_{k+1}^{g} が 1 でその他が 0 のベクトルとなる．このとき，損失関数 $L_{D,i}$ は式 (1) となる．

$$L_{D,i} = Gate\left(-\sum_{j=1}^{k+1} t_j^{\mathrm{g}} \log_e d_j^{(i)}\right) \tag{1}$$

提案手法では，特徴マップを転移することにより，知識転移を行う．特徴マップのチャネル番号を $c = 1, \ldots, C$，特徴マップの層の位置を $m = 1, \ldots, M$，特徴マップのベクトル番号を $n = 1 \ldots, N$ とすると，i 番目の Generator の特徴マップは $F_{i,m,c,n}$ で表される．このとき，ハイパーパラメータを α とすると，損失関数 $L_{t,s}$ は式 (2) および式 (3) となる．

$$Q_i^{(m,n)} = \frac{1}{C}\sum_{c=1}^{C} F_{i,m,c,n}^2 \tag{2}$$

$$L_{t,s} = Gate\left(\frac{\alpha}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(\frac{Q_t^{(m,n)}}{\left\|Q_t^{(m,n)}\right\|_2} - \frac{Q_s^{(m,n)}}{\left\|Q_s^{(m,n)}\right\|_2}\right)^2\right) \tag{3}$$

4. 評価実験

従来手法との比較により，提案手法の有効性を調査し，最適な知識の転移方法を探索する．実験には，10 クラスの一般物体のカラー画像で構成されている CIFAR-10 データセットを用い，学習用のサンプルである 50,000 枚を 64×64 にリサイズして使用する．本実験では，ベースの GANs の手法として DCGANs を用いる．また，知識転移グラフにおけるゲート関数の探索回数を 1,500 とし，各 Generator における生成画像の Inception Score（IS）を最大化するゲート関数の組み合わせを探索する．学習時のバッチサイズを 256 とし，学習回数を 50，各 Generator に入力する潜在変数の次元数を 100，ハイパーパラメータ α を 1,000 とする．生成画像の質を評価する指標として IS および Fréchet Inception Distance（FID）を用いる．また，IS は高い値，FID は低い値になるほど生成画像の質が高いことを表している．

4.1 提案手法の有効性の調査

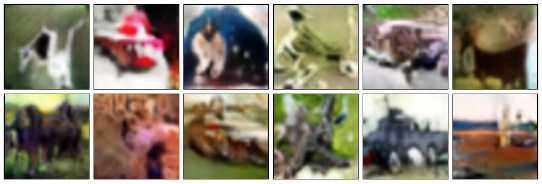
本実験では，DCGANs および MAD-GANs，提案手法における生成画像の質を比較する．DCGANs は SN の導入の有無による 2 パターンを比較対象とする．MAD-GANs および提案手法は SN を導入し，Generator の数を 3 とする．学習回数を 200 Epoch とし，学習終了時のモデルパラメータを用いて評価を行う．各手法の IS および FID を表 1 に示す．表 1 より，提案手法は IS が最も高い値となり，最も生成画像の質が高いことが確認できる．FID は SN を導入した DCGANs が最も低く，提案手法も同様に低い値である．GAN は学習が進むと，IS と FID がトレードオフの関係にあることが知られている．本実験における提案手法は，各 Generator における生成画像の IS を最適化対象とし，最大化するゲート関数の組み合わせを探索した．提案手法は最適化対象を FID とし，最小化するゲート関数の組み合わせの探索により，FID に特化した手法も実現できる．

4.2 生成画像の可視化

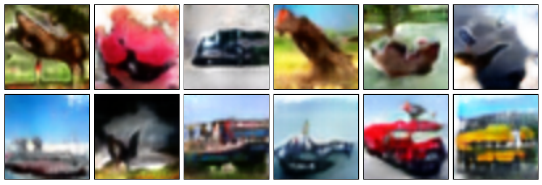
MAD-GANs および提案手法に同一の潜在変数を入力し，生成した画像を比較する．生成画像の可視化結果を図 3 に示す．図 3 の結果から，提案手法は実画像に近い画像が生成できていることが確認できる．

表 1: IS および FID の比較結果

	SN	IS ↑	FID ↓
DCGANs		4.57 ± 0.03	25.79
	✓	4.77 ± 0.05	**21.17**
MAD-GANs	✓	4.77 ± 0.03	28.76
提案手法	✓	$\mathbf{5.37 \pm 0.09}$	23.77
実画像	–	8.70 ± 0.14	–



(a) MAD-GANs



(b) 提案手法

図 3: 生成画像の可視化結果

4.3 知識転移グラフの最適化

提案手法において，知識転移グラフを用いたゲート関数の探索を 1,500 回行い，知識の伝達方法を最適化した．最適化対象である各 Generator における生成画像の IS が最大の値となったグラフを図 4 に示す．赤色のエッジが Positive linear Gate，灰色のエッジが Through Gate を表している．G_1 は，他の Generator から知識を転移されない G_2 と比較して IS の値が高く，知識の転移は生成画像の質の向上に有効であるといえる．また，Generator 間の知識の転移は学習初期は不要であり，学習が進むにつれて転移すると良いことや，G_3 は G_1 のバッファのような役割を果たすことで，G_1 の学習の安定化に寄与していることが確認できる．

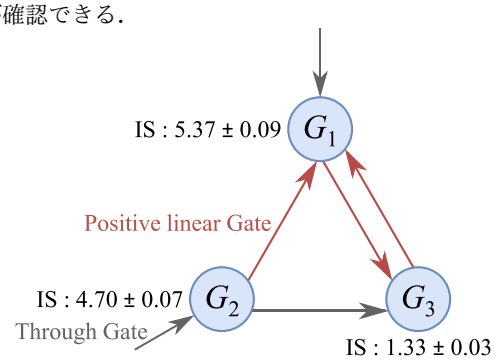


図 4: 知識転移グラフの最適化結果

5. おわりに

本研究では複数の Generator および 1 つの Discriminator を用いて，それぞれの Generator で知識を転移しながら共同学習を行う手法を提案した．評価実験により提案手法の有効性を確認した．今後は複数の Discriminator を用いることで，アンサンブル効果により生成画像のさらなる質の向上を目指す．

参考文献

[1] A. Radford, *et al.*, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, ICLR, 2016.

[2] A. Ghosh, *et al.*, “Multi-Agent Diverse Generative Adversarial Networks”, CVPR, 2018.

研究業績

[1] T. Tsukahara, *et al.*, “Collaborative learning of generative adversarial networks”, VISAPP, 2021.

(他 3 件)

動画像における時空間情報を考慮した DNN による視覚的説明に関する研究

TR19010 三津原将弘　　指導教授：藤吉弘亘

1. はじめに

画像認識分野において，深層学習の判断根拠を解析する手法の一つに視覚的説明がある．視覚的説明では，深層学習が認識する際に注視した領域をヒートマップで表現した Attention map を解析する．一方，動画像認識では静止画像とは異なり，空間情報だけでなく時間情報も考慮する必要があることから，判断根拠の解析が困難とされてきた．本研究では，視覚的説明を動画像認識に拡張し，動画像における新たな視覚的説明の手法である Spatio-Temporal Attention Branch Network (ST-ABN) を提案する．ST-ABN は，推論時の空間情報と時間情報に対する重要度を獲得し，認識処理に応用することで認識性能の向上と視覚的説明の獲得を実現する．評価実験では，Something-Something データセットを用いて実験を行い，提案手法により認識性能の向上と空間情報と，時間情報を同時に考慮した視覚的説明が可能となることを示す．

2. 関連研究

従来手法である 3D CNN 及び視覚的説明を用いた手法について述べる．

3D Convolutional Networks　3D CNN の代表的な手法である 3D Convolutional Networks （C3D） [1] は，空間方向に対する 2D の畳み込み処理を時間方向に拡張し，3D 空間に対する畳み込み処理を行うことで時空間の特徴を獲得する．3D 空間に対して畳み込み処理を行うことで，従来の空間方向に対する畳み込み処理を行う 2D CNN では困難であった時系列情報を捉えることができる．

Attention Branch Network　視覚的説明を用いた手法に Attention Branch Network (ABN) [2] がある．ABN は，Attention map を Attention 機構により特徴マップに重み付けすることで認識性能と視覚的な説明性の向上を実現している．

3. 提案手法

本研究では，重要な空間情報と時間情報を同時に考慮した視覚的説明が可能な ST-ABN を提案する．

3.1 ST-ABN の構造

ST-ABN は，図 1 に示すように Feature extractor，Spatio-Temporal (ST) Attention branch，Perception branch の 3 つのモジュールで構成する．Feature extractor は，複数の畳み込み層で構成されており，入力から特徴マップを獲得する．

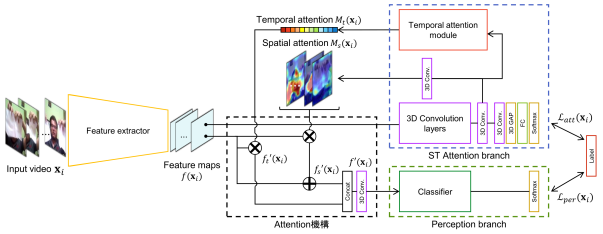


図 1：ST-ABN のネットワーク構造

ST Attention branch は，空間情報に対する重要度を示す Spatial attention と時間情報に対する重要度を示す Temporal attention を獲得する．Perception branch は，Attention 機構により Spatial attention と Temporal attention を重み付けした特徴マップを入力し，各クラスの確率を出力する．ST-ABN は，式 (1) のように ST Attention branch の学習誤差 $\mathcal{L}_{att}$ と Perception branch の学習誤差 $\mathcal{L}_{per}$ を用いて学習する．クラス識別誤差は，Softmax 関数とクロスエントロピー誤差を用いて算出する．

$$\mathcal{L}(\mathbf{x}_i) = \mathcal{L}_{att}(\mathbf{x}_i) + \mathcal{L}_{per}(\mathbf{x}_i) \tag{1}$$

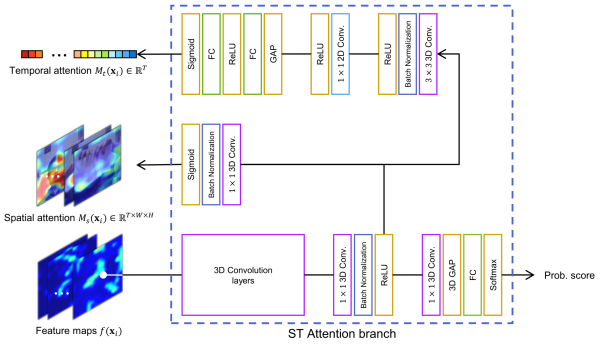


図 2：ST Attention branch の構造

3.2 Spatio-Temporal Attention branch

ST Attention branch では，図 2 に示すように Feature extractor から出力された特徴マップを用いて，空間情報と各フレームに対する重要度を獲得し，Global Average Pooling (GAP) を介してクラス識別を行う．Feature extractor から出力された特徴マップは，複数の畳み込み層を経て，クラス数分のチャネルを持つ 1×1 の畳み込み層によりクラス数分の特徴マップを獲得する．

Spatial attention は，このクラス数分の特徴マップを用いて生成する．クラス数分の特徴マップは，1×1 の畳み込み層により 1 枚の特徴マップに集約する．その後，フレームごとに Sigmoid 関数で 0 から 1 の範囲に正規化した Attention map を獲得する．

Temporal attention も Spatial attention と同様にクラス数分の特徴マップを用いて生成する．はじめに，クラス数分の特徴マップを 1×1 の畳み込み層によりチャネル方向に対して次元を圧縮する．その後，フレーム数分のチャネル数を持つ畳み込み層と GAP を介して，各特徴マップの空間方向に対する平均値を求める．最後に，全結合層，ReLU 及び Sigmoid 関数を介して，各フレームの重要度を獲得する．

3.3 Attention 機構

Spatial attention $M_s(\mathbf{x}_i)$ は，式 (2) より特徴マップ $f(\mathbf{x}_i)$ に重み付けし，重み付け前の特徴マップを加算する．これにより，特徴マップの消失を抑制し，Attention map を効率的に認識に反映させることができる．

$$f'_s(\mathbf{x}_i) = (1 + M_s(\mathbf{x}_i) \cdot f(\mathbf{x}_i)) \tag{2}$$

Temporal attention $M_t(\mathbf{x}_i)$ は，式 (3) より特徴マップに乗算することで重み付けを行う．

$$f'_t(\mathbf{x}_i) = M_t(\mathbf{x}_i) \cdot f(\mathbf{x}_i) \tag{3}$$

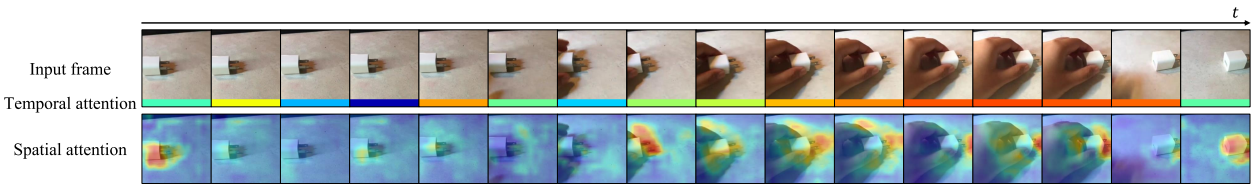
Spatial attention と Temporal attention でそれぞれ重み付けした特徴マップは，式 (4) よりチャネル方向に結合する．その後，結合した特徴マップを畳み込み層に入力して統合する．

$$f'(\mathbf{x}_i) = \mathrm{conv}\left(f'_s(\mathbf{x}_i) \odot f'_t(\mathbf{x}_i)\right) \tag{4}$$

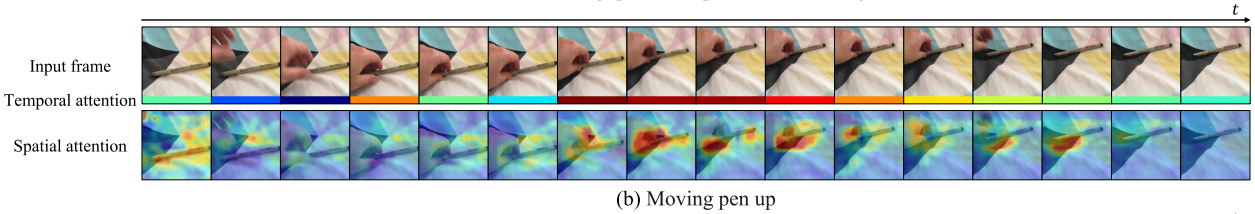
この統合した特徴マップを Perception branch に入力し，最終的な認識結果を出力する．これにより，重要な時空間特徴に着目した学習が可能となる．

4. 評価実験

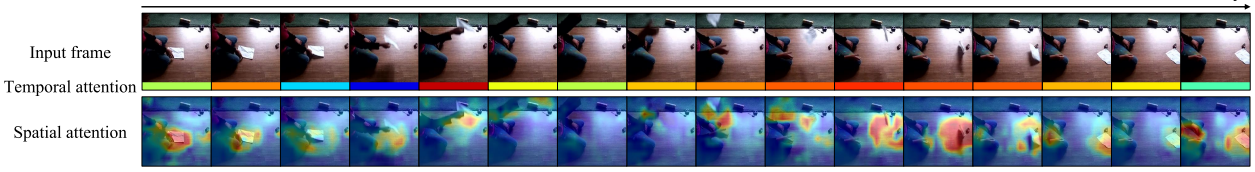
評価実験では，Something-Something データセット [3] の V1 と V2 を用いてベースラインと提案手法との認識精度を比較する．Something-Something データセットは，大規模な動作認識用のデータセットであり，人が日用品を扱う 174 種類の基本的な動作を認識する．提案手法は，ベースラインである 3D ResNet-50 をベースに構築する．入力



(a) Pushing iphone adapter from left to right



(b) Moving pen up



(c) A piece of paper falling like a feather or paper

図 3：Spatial attention と Temporal attention の可視化結果

表 1：ベースラインと提案手法との認識精度の比較結果 [%]

Model	Frames	Something-Something V1		Something-Something V2	
		Top-1	Top-5	Top-1	Top-5
3D ResNet-50 (ベースライン)	32	45.2	74.5	55.6	81.6
3D ResNet-50 + 提案手法	32	**45.9**	**75.4**	**56.6**	**82.0**
3D ResNet-50 (ベースライン)	32+32	52.9	81.5	63.8	89.2
3D ResNet-50 + 提案手法	32+32	**53.3**	**82.0**	**64.1**	**89.6**

するフレーム数は，32 フレームと 64 フレームを入力した場合で認識精度を比較する．64 フレームを入力する場合は，32 フレームの動画を 2 つ入力し，2 つの動画に対するスコアを平均することで最終的な認識精度を算出する．

4.1 ベースラインとの認識精度の比較

表 1 にベースラインと提案手法との認識精度の比較結果を示す．表 1 に示すように提案手法がベースラインよりも高い認識精度を獲得した．さらにフレーム数を増やした場合においても提案手法がベースラインよりも高い認識精度を獲得した．

4.2 Attention の定性的な評価

図 3 に Something-Something データセット V2 における Spatial attention と Temporal attention の可視化結果を示す．図 3 の上から下に向かって入力フレーム，Temporal attention，Spatial attention，入力動画のクラスを示している．Temporal attention では，各フレームに対して出力された重要度をヒートマップの色に変換することで可視化しており，重要度が高いほど赤く，重要度が低いほど青くなる．図 3 に示すように，動きを含むフレームの重要度が高くなることが確認できる．

Spatial attention では，各フレームに対して出力された Attention map をヒートマップとして可視化する．Spatial attention は，動作特有の特徴的な空間領域を捉えつつ，動的なフレームに対して強く注視していることが確認できる．これらの結果から，提案手法により空間情報と時間情報を同時に考慮した視覚的説明が可能である．

4.3 Attention の有効性の評価

Spatial attention と Temporal attention の有効性を定量的に評価する．評価方法として Spatial attention と Temporal attention を反転させて推論を行う．そして，反転する場合と反転しない場合で認識精度がどれだけ低下するかを調査し，認識に有益な空間情報と時間情報に対する重要度が得られていることを確認する．

表 2 に Something-Something データセットにおける Spatial attention と Temporal attention の反転による認識精度の比較結果を示す．表 2 に示すように，Spatial attention のみの反転と Temporal attention のみの反転により認識精度が低下することが確認できる．さらに，Spatial attention と Temporal attention を反転させることで大幅に認識精度が低下することから認識に有益な重要度が得られていることが確認できる．

表 2：Attention の反転による認識精度の比較結果 [%]

Attention		Something. V1		Something. V2	
Spatial	Temporal	Top-1	Top-5	Top-1	Top-5
		45.9	75.4	56.6	82.0
✓		36.2	66.8	19.1	39.7
	✓	28.3	57.4	45.5	75.6
✓	✓	**22.7**	**50.0**	**12.4**	**29.0**

5. おわりに

本研究では，ST-ABN を提案し，動画像認識における空間情報と時間情報を同時に考慮した視覚的説明を実現した．ST-ABN は，3D CNN がベースであるモデルの推論時における空間情報と時間情報に対する重要度を獲得し，Attention 機構に応用することで視覚的な説明性と認識性能の向上が可能となる．今後は，提案手法を分類以外の他の動画像認識タスクへ応用することを検討する．

参考文献

[1] D. Tran, *et al.*, “Learning Spatiotemporal Features with 3D Convolutional Networks”, ICCV, 2015.

[2] H. Fukui, *et al.*, “Attention Branch Network: Learning of Attention Mechanism for Visual Explanation”, CVPR, 2019.

[3] R. Goyal, *et al.*, “The“Something Something” Video Database for Learning and Evaluating Visual Common Sense”, ICCV, 2017.

研究業績

[1] 三津原将弘 等, “Attention map を介した人の知見の組み込み”, 第 22 回 画像の認識・理解シンポジウム, 2019.

[2] M. Mitsuhara, *et al.*, “Embedding Human Knowledge into Deep Neural Network via Attention Map”, VISAPP, 2021.