

クラスに依存しない潜在表現の獲得による継続学習における安定性と可塑性のジレンマの解消

村田 健悟¹ 伊東 聖矢² 大原 剛三²

概要：近年、深層学習モデルを対象とし、Class Incremental Learning (CIL) と呼ばれる、分類対象クラスが増加する継続学習のシナリオについて広く研究されている。CIL では、モデルが既学習クラスへの識別性能を保持可能なほどに安定、かつ、新規クラスについて十分学習できるほどに可塑である必要がある。この相反する要件は、安定性と可塑性のジレンマと呼ばれ、このジレンマの解消が CIL における課題の一つである。そこで本研究では、新規クラスの識別にも有用な潜在表現を事前に獲得することが学習に必要な可塑性を軽減し、このジレンマの解消につながり得ることに着目し、クラスに依存しない潜在表現の獲得を促す学習フレームワークを提案する。提案フレームワークでは、その表現を分類に適した表現へ変換するメカニズムを導入し、各クラスに対する分類精度の低下を抑制する。ベンチマークデータを利用した実験では、本フレームワークを既存手法に適用することで、最終的な全クラスに対する平均精度が向上することを示す。

Breaking Down Stability-Plasticity Dilemma in Continual Learning by Acquiring Class-Agnostic Features

1. はじめに

深層学習モデルは、画像分類タスクにおいて非常に高い性能を示すことが知られている [1, 2]。しかし、実世界におけるクラス分類問題では、従来のクラス分類問題と異なり、分類対象クラスが増加していくことが考えられる。そこで近年、深層学習モデルを対象とし、Class Incremental Learning (CIL) と呼ばれる、分類対象クラスが増加する継続学習のシナリオについて広く研究されている [3]。このシナリオでは、新たな学習対象クラスにより構成されるクラス識別タスクが逐次的にモデルに与えられ、モデルはこれまでに学習したクラスへの認識性能を低下させることなく、与えられたタスクに含まれるクラスの識別が可能になるようにパラメータを更新する必要がある。そのため、モデルがすでに学習したクラスに対する認識性能を保持可能なほどに安定であり、同時に、新たなクラスについて十分

学習可能であるほどに可塑である必要がある。このような相反する要件は、安定性と可塑性のジレンマ [4] と呼ばれ、このジレンマを解消することが CIL における課題の一つである。特に、深層学習モデルの過度な可塑性は、破滅的忘却と呼ばれる、新たなクラスについて学習する際に生じるパラメータの更新により、既学習クラスへの認識性能が大幅に低下する現象を引き起こすことが知られている [5]。

多くの先行研究では、破滅的忘却を回避し安定性と可塑性とのバランスを保つために、深層学習モデルの安定性を向上させる手法が提案されている。たとえば、Knowledge Distillation を用いる手法 [6, 7] では、クラス分類問題における損失関数として一般に用いられる交差エントロピー誤差に加えて、モデルの入出力関係の変動を阻害する損失についてもパラメータを最適化することで、モデルの安定性を向上させている。また、正則化による手法 [8] では、既学習クラスの識別に重要なパラメータ群の変動を妨げる損失が用いられている。しかしこれらの手法では、安定性の向上を可塑性の阻害により達成しているため、新規クラスへの識別性能を向上させるために高い可塑性が必要な場合、その識別性能を犠牲にしなければ既学習クラスへの識別性能を保持できない。加えて、必要な可塑性を本質的に軽減

¹ 青山学院大学大学院理工学研究科
Graduate School of Science and Engineering, Aoyama Gakuin University, Kanagawa 252-5258, Japan

² 青山学院大学理工学部
College of Science and Engineering, Aoyama Gakuin University, Kanagawa 252-5258, Japan

するようなメカニズムはこれらの手法には組み込まれていない。実際、クラス分類に関する学習が交差エントロピー誤差の最適化のみに依存しているため、将来的に分類対象となり得るクラスの識別に有用な特徴であっても、現在の分類対象クラスの識別に有用でない限りモデルはその特徴を表した潜在表現を獲得することができない。

このように、新規クラスへの識別性能を向上させるために必要な可塑性を軽減させる仕組みの導入は、既学習クラスへの識別性能の保持と新規クラスへの識別性能の向上を両立するために不可欠な要素であると考えられる。そこで本研究では、新規クラスの識別にも有用な潜在表現を事前に獲得することが学習に必要な可塑性の軽減に寄与することに着目し、クラスに依存しない潜在表現の獲得を促す学習フレームワークを提案する。本フレームワークでは、自己教師あり学習 [9, 10] を利用することでクラスに依存しない表現をモデルに学習させ、注意機構 [11, 12] によりその表現ベクトルを分類に最適な情報を含む表現ベクトルに変換することで、汎用的かつクラス分類に最適な潜在表現の獲得を可能とする。加えて、注意機構による変換メカニズムの安定性を向上させるために、Knowledge Distillation に基づいた新たな損失を導入する。さらに、ベンチマークデータを利用した実験を通して、本フレームワークを既存手法に適用することで、最終的な全クラスに対する平均精度が向上することを示す。

以下、2 節で関連研究について紹介し、3 節で CIL の問題設定と本フレームワークにおけるベースライン手法として用いる UCIR と呼ばれる CIL 手法の概要を説明する。4 節で提案フレームワークの詳細について述べ、5 節で評価実験の報告および提案手法の有効性を議論する。6 節で本稿のまとめを述べる。

2. 関連研究

CIL における忘却は、新規クラスについて学習する際に、新規クラスのサンプルに加えて、既学習クラスに対する学習時に用いたすべてのサンプルについても損失を最適化することで容易に回避できる。しかしこの方法は、学習に利用したすべてのサンプルを保持し、それらについて最適化が必要があるため、時間計算量、空間計算量（メモリ消費量）の面から現実的ではない。そこで、学習に利用したサンプルの一部のみを保持し、それらサンプルを最適化に利用する Experience Replay [13] と呼ばれる手法が提案されている。この手法は、その単純さと忘却抑制効果の高さから、CIL におけるベースライン手法として広く利用されている。加えて、モデルの安定性を更に向上させるために、Knowledge Distillation に基づいた手法が多く提案されている [7, 14, 15]。これらの手法は、新規クラスについて学習する際に、それらのクラスについて学習する前のモデルにより定義される入出力関係を保持するようにパラメータを

最適化することで忘却を抑制する。

また、CIL 手法の中でも特に表現学習に注目した手法として、対照学習を用いた手法 [16]、メタ学習を用いた手法 [17]、および自己教師あり学習を用いた手法 [18–21] が挙げられる。対照学習を用いた手法では、教師あり対照損失 [22] を利用することで、異なるクラスに属するサンプルの表現ベクトル間の距離が大きく、同一クラスに属するサンプルの表現ベクトル間の距離が小さくなるように、パラメータが最適化される。これに対して、メタ学習を用いた手法では、メタ学習のフレームワークを導入することで、新規クラスに対する学習に有用な潜在表現をモデルに事前に獲得させる。しかし、メタ学習のフレームワークでは、推論対象サンプルがどのタスクに所属しているかの情報が必要であるため、与えられたサンプルの所属タスクを推論するようなシステムを導入し、かつそのシステムに生じる忘却も緩和する仕組みが必要である [17]。自己教師あり学習を用いた手法では、回転させた訓練画像の角度を自己教師ラベルとして用いる手法 [19, 20] などが提案されている。

3. 問題設定と既存手法の概要

本節では、CIL の問題設定について述べたのちに、本フレームワークにおけるベースライン手法として用いる UCIR [14] と呼ばれる CIL 手法について説明する。まず、 N 個のクラス識別タスクから構成されるタスク列を $\{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_N\}$ とする。また、 t 番目のタスク \mathcal{T}_t に付随する学習データセットを \mathcal{D}_t と表す。このとき、 $\mathcal{D}_t = \{(x_t^i, y_t^i)\}_{i=1}^{n_t}$ は画像 $x_t^i \in \mathcal{X}_t$ とその画像に対応するラベル $y_t^i \in \mathcal{Y}_t$ の対により構成される。なお、任意の 2 タスク $\mathcal{T}_i, \mathcal{T}_j (i \neq j)$ に対し、ラベル集合 $\mathcal{Y}_i, \mathcal{Y}_j$ は互いに素であると仮定する。加えて、深層学習モデルを特徴抽出器 f とクラス分類器 g の合成関数 $g \circ f$ により表す。また、特徴抽出器 f は L 個の畳み込みブロックにより構成され、クラス分類器 g は 1 層の全結合層により構成されるとする。なお、畳み込みブロックは残差ブロック [1] といった、畳み込み層やバッチ正規化層の組み合わせにより構成されるとする。ここで、 t 番目のタスク \mathcal{T}_t が与えられたとき、モデルは付随する学習データセット \mathcal{D}_t に基づいてパラメータを更新することで、任意の画像 $x \in \bigcup_{j=1}^t \mathcal{X}_j$ について正しいラベルを予測できるようになることが要求される。以下では、 t 番目のタスクについて学習中のモデルの特徴抽出器を現行の特徴抽出器と呼び、 f_t と表す。また、 $t-1$ 番目のタスクに対する学習が完了した時点のモデルの特徴抽出器を過去の特徴抽出器と呼び、 f_{t-1} と表す。加えて、 t 番目のタスクに対する学習時に利用する M 個の保持サンプル集合を $\mathcal{M}_t = \{(x_{\text{mem}}^i, y_{\text{mem}}^i)\}_{i=1}^M \subset \bigcup_{j=1}^{t-1} \mathcal{D}_j$ と表す。

UCIR では、モデルの安定性と可塑性とのバランスを保つために、コサイン正規化、Knowledge Distillation の利用、新規クラスと既学習クラスの特徴ベクトル間の分離性

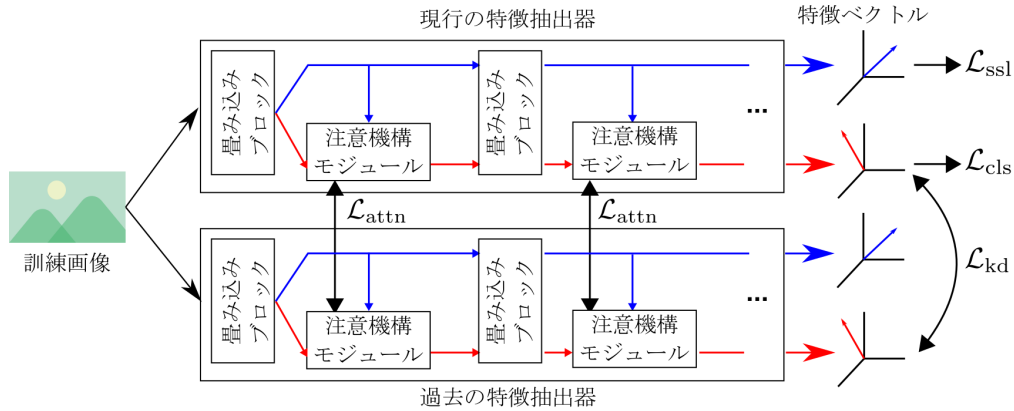


図 1 提案フレームワークを UCIR に組み込んだ場合の概要図

Fig. 1 Illustration of the proposed framework combined with UCIR

を向上させる損失の利用という 3 つの要素を導入している。まず、コサイン正規化は、特微抽出器の出力ベクトルとクラス分類器の各重みベクトルとの内積演算時に適用される。具体的には、各ベクトルを l_2 正規化したのちに内積演算を行うことで、新規クラスへの偏った予測を抑制する。次に、Knowledge Distillation については、現行モデルの特微抽出器の出力ベクトルと、過去モデルの特微抽出器の出力ベクトル間のコサイン距離を損失関数として利用することで、モデルに生じる忘却を緩和する。ここで、その損失関数 \mathcal{L}_{kd} を式 (1) に示す。

$$\mathcal{L}_{kd} = 1.0 - \langle \overline{f_{t-1}(x)}, \overline{f_t(x)} \rangle \quad (1)$$

なお、ベクトル v_1, v_2 に対して、 $\overline{v_1} = v_1 / \|v_1\|_2$ は l_2 正規化ベクトル、 $\langle \overline{v_1}, \overline{v_2} \rangle$ はコサイン類似度の計算処理を表す。さらに、新規クラスと既学習クラスの表現ベクトル間の分離性を向上させるため、式 (2) により定義される損失 \mathcal{L}_{mr} を導入している。

$$\mathcal{L}_{mr} = \sum_{k=1}^K \max(m - \langle \overline{w^{y_{mem}}}, \overline{f(x_{mem})} \rangle + \langle \overline{w^k}, \overline{f(x_{mem})} \rangle, 0) \quad (2)$$

なお、 m はマージンを調整するハイパーパラメータ、 w^k は新規クラス $y \in \mathcal{Y}_t$ に対応するクラス分類器の重みベクトルのうち $\overline{f(x)}$ とのコサイン類似度が k 番目に高いベクトルを表す。このとき、全体の損失関数 \mathcal{L}_{ucir} は式 (3) により定義される。

$$\mathcal{L}_{ucir} = \mathcal{L}_{ce} + \beta_t \cdot \lambda_{kd} \mathcal{L}_{kd} + \lambda_{mr} \mathcal{L}_{mr} \quad (3)$$

なお、 \mathcal{L}_{ce} は交差エントロピー誤差、 λ_{kd} 、 λ_{mr} は各損失項の影響を調整するハイパーパラメータを表す。加えて、 $\beta_t = \sqrt{|\bigcup_{j=1}^t \mathcal{Y}_j| / |\mathcal{Y}_t|}$ は \mathcal{L}_{kd} の影響を調整するスカラー値を表す。

4. 提案フレームワーク

本節では、汎用的かつクラス分類に最適な潜在表現の獲得をモデルに促す提案フレームワークについて説明する。ここで、図 1 に、提案フレームワークを UCIR に組み込んだ場合の概要図を表す。本フレームワークでは、特微抽出器が 1 つの画像から 2 種の特微ベクトルを出力する。まず、畳み込みブロックのみを通り出力されるベクトル（青線）は、自己教師損失 \mathcal{L}_{ssl} の計算に利用される。この損失の最適化により、畳み込みブロックのパラメータはクラスに依存しない情報を抽出するように更新される。一方、注意機構モジュールを通り出力されるベクトル（赤線）は、クラス分類に関する損失 \mathcal{L}_{cls} の計算に用いられる。この損失 \mathcal{L}_{cls} の最適化を通して、畳み込みブロックの抽出したクラスに依存しない特徴を分類に最適な特徴に変換するように各注意機構モジュールのパラメータが更新される。さらに、Knowledge Distillation に基づいた損失 \mathcal{L}_{attn} を導入することで、注意機構による変換メカニズムの安定性を向上させる。

4.1 クラスに依存しない特徴の学習

本フレームワークでは、SimSiam による表現学習手法 [10] を利用することで、クラスに依存しない特徴の学習を実現する。具体的には、まず、ある訓練画像 x に対してデータ拡張を適用することで 2 つの画像 \hat{x}, \tilde{x} を生成する。次に、各画像に対して特微抽出器 f_t 、射影器 $proj_t$ 、予測器 $pred_t$ を通すことで 2 種のベクトルを生成する。なお、射影器と予測器はそれぞれ多層パーセプトロンにより構成される。ここで、生成されるベクトルをそれぞれ $\hat{p} = (pred_t \circ proj_t \circ f_t)(\hat{x})$ 、 $\tilde{q} = (proj_t \circ f_t)(\tilde{x})$ と表す。最後に、式 (4) により定義される損失 \mathcal{L}_{ssl} を計算する。

$$\mathcal{L}_{ssl} = -\frac{1}{2} (\langle \hat{p}, stopgrad(\tilde{q}) \rangle + \langle \tilde{p}, stopgrad(\hat{q}) \rangle) \quad (4)$$

なお、 $stopgrad$ は勾配グラフからの除外処理を表す。本フ

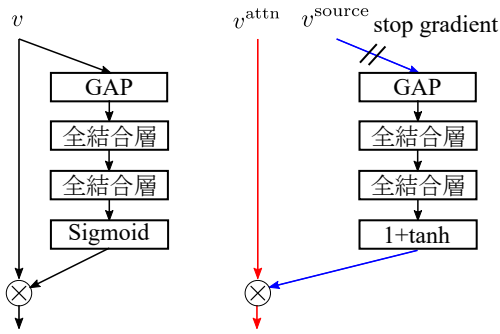


図 2 SE block (左図) と提案注意機構モジュール (右図) の概要図
Fig. 2 The schemas of the original SE block (left) and our proposed attention module (right).

フレームワークでは、この損失 \mathcal{L}_{ssl} について畳み込みブロックを最適化することで、畳み込みブロックがクラスに依存しない特徴を抽出するように学習することを保証する。

4.2 注意機構による変換メカニズム

損失 \mathcal{L}_{ssl} の最小化により、畳み込みブロックはクラスに依存しない特徴の抽出が可能となるが、この特徴は現行の分類対象クラスの識別に無関係な情報を含んでいることから、クラス分類に最適とは言えない。そこで、このような無関係な情報を弱め、クラス分類に有用な情報のみを強調するような、注意機構による変換メカニズムを導入する。ここで、図 2 に本フレームワークで用いる注意機構モジュールの概要図を表す。図 2 に示すように、本注意機構モジュールは畳み込みニューラルネットワークにおける注意機構モジュールとして一般に用いられている SE block [12] について、その入力ベクトルと活性化関数を変更したものとなっている。

SE block では、中間特徴ベクトルから注意マップを生成し、その注意マップを同じ中間特徴ベクトルに適用することで特徴を変換している。しかし、このような変換メカニズムは、中間特徴ベクトルに生じる変動を雪だるま式に増大させてしまうため、モデルの安定性を著しく低下させてしまう。そこで、このような変動の増大を避けるために、本注意機構モジュールでは畳み込みブロックのみを通して出力される中間特徴ベクトルを用い注意マップを生成する。これは、クラスに依存しない情報を含む中間特徴ベクトルから注意マップを生成していることを意味する。この中間特徴ベクトルは、自己教師有学習について最適化された畳み込みブロックから生成されるため、安定かつ注意マップを生成するために必要な情報を十分に含んでいると考えられる。加えて、この注意マップ生成に関わる勾配が自己教師あり学習による最適化を妨げ得ることから、勾配グラフからの除外処理を注意機構にベクトルを与える際に導入している。

活性化関数については、SE block で用いられるシグモ

イド関数ではなく、 $1 + \tanh(\cdot)$ を用いる。この活性化関数は、値域が $[0, 2]$ であるため、本注意機構により中間特徴ベクトルの各値を強めることと弱めることの双方が可能となる。加えて、ある注意機構モジュールにおいて、パラメータがすべて 0 であった場合、そのモジュールは恒等写像となる。この性質は、本注意機構を用いた変換メカニズムによる不必要な特徴の変換を避けることが可能であることを示している。そのため、自己教師あり学習により獲得したクラスに依存しない特徴が破壊的に変換されることを避けることが可能となる。

4.3 注意機構による特徴変換メカニズムの安定性向上方法

本フレームワークで導入する注意機構モジュールは、 $l \in [1, L]$ 番目の畳み込みブロックが生成するプーリング後中間特徴ベクトル z_t^l について、全結合層による演算処理 h_t^l を適用することで、注意マップ $a_t^l = 1 + \tanh(h_t^l(z_t^l))$ を生成する。この注意マップの生成メカニズムの安定性を向上させるためには、注意機構により定義される入出力関係を保持することに加えて、中間特徴ベクトル z_t に生じる変動を妨げるような損失を導入する必要がある。そこで、Knowledge Distillation の考え方を踏襲し、入出力関係の保持を促進する損失 $\mathcal{L}_{attn}^{map,l}$ と、中間特徴ベクトルに生じる変動を妨げる損失 $\mathcal{L}_{attn}^{source,l}$ を導入する。

まず、入出力関係の保持は、過去の特徴抽出器に含まれる l 番目の畳み込みブロックが生成するプーリング後中間特徴ベクトル z_{t-1}^l と、対応する注意機構の全結合層 h_{t-1}^l を用いることで達成する。具体的には、各中間特徴ベクトル z_t^l , z_{t-1}^l について、現行の注意機構モジュールの出力する注意マップと過去の注意機構モジュールの出力する注意マップとの距離を最小化することで、入出力関係の保持を達成する。ここで、これら 2 種の距離により定義される損失 $\mathcal{L}_{attn}^{map,l}$ を式 (5) に示す。

$$\mathcal{L}_{attn}^{map,l} = \lambda_{attn}^{map,new} \mathcal{L}_{attn}^{map,new,l} + \lambda_{attn}^{map,old} \mathcal{L}_{attn}^{map,old,l} \quad (5)$$

$$\mathcal{L}_{attn}^{map,new,l} = \|\tanh(h_t^l(z_t^l)) - \tanh(h_{t-1}^l(z_t^l))\|_2^2 \quad (6)$$

$$\mathcal{L}_{attn}^{map,old,l} = \|\tanh(h_t^l(z_{t-1}^l)) - \tanh(h_{t-1}^l(z_{t-1}^l))\|_2^2 \quad (7)$$

なお、 $\lambda_{attn}^{map,new}$ と $\lambda_{attn}^{map,old}$ はそれぞれ各距離関数の影響を調整するハイパーパラメータを表す。

次に、中間特徴ベクトルに生じる変動を妨げる損失 $\mathcal{L}_{attn}^{source,l}$ の詳細について説明する。本フレームワークでは、現行特徴抽出器内で抽出される中間特徴ベクトル z_t^l と過去特徴抽出器内で抽出されるベクトル z_{t-1}^l 間の距離を最小化することで、中間特徴ベクトルに生じる変動の阻害を達成する。具体的には、式 (8) により定義される重み付き距離を利用する。

$$\mathcal{L}_{\text{attn}}^{\text{source},l} = \lambda_{\text{attn}}^{\text{source}} \hat{\omega}_l \langle \bar{z}_t^l, \bar{z}_{t-1}^l \rangle \quad (8)$$

$$\hat{\omega}_l = \frac{\omega_l}{\sum_{j=1}^L \omega_j} \quad (9)$$

なお、 $\lambda_{\text{attn}}^{\text{source}}$ はこの損失の影響を調整するハイパーパラメータ、 ω_l は重み係数を表す。この重み係数は、注意マップの各値を中間ベクトルの各値について微分した値により導出される。具体的には、各タスク $\mathcal{T}_{t'}$ に対する学習が終了した時点で、この重み係数を式 (10) に従って更新する。

$$\omega_l \leftarrow \frac{1}{2} \left(\omega_l + \mathbb{E}_{\mathcal{D}_{t'}} \left[\frac{1}{D^2} \sum_{d=1}^D \sum_{d'=1}^D \frac{\partial \tanh(h_{t'}^l(z_{t'}^l))_d^2}{\partial (z_{t'}^l)_d} \right] \right) \quad (10)$$

なお、 D は中間特徴ベクトル z^l の次元数を表す。このような重み係数を利用することで、中間特徴ベクトルの変動により対応する注意マップの変動量が大きい場合に、中間特徴ベクトルの変動を大きく阻害することが可能となる。

まとめると、Knowledge Distillation に基づいた損失 $\mathcal{L}_{\text{attn}}$ は式 (11) により定義される。

$$\mathcal{L}_{\text{attn}} = \sum_{l=1}^L \mathcal{L}_{\text{attn}}^{\text{map},l} + \mathcal{L}_{\text{attn}}^{\text{source},l} \quad (11)$$

4.4 全体の損失

これまでに述べた損失と UCIR により提案された損失をまとめると、本フレームワークで用いられる損失関数は式 (12) により定義される。

$$\mathcal{L} = \alpha \mathcal{L}_{\text{ce}} + (1 - \alpha) \mathcal{L}_{\text{ssl}} + \beta_t (\lambda_{\text{kd}} \mathcal{L}_{\text{kd}} + \mathcal{L}_{\text{attn}}) + \lambda_{\text{mr}} \mathcal{L}_{\text{mr}} \quad (12)$$

ここで、 $\alpha \in [0, 1]$ は損失 \mathcal{L}_{ce} と \mathcal{L}_{ssl} とのバランスを調整するハイパーパラメータを表す。なお、損失 \mathcal{L}_{ssl} の計算には複雑なデータ拡張手法を利用し生成した画像により計算し、他の損失はより単純なデータ拡張手法を適用し生成した画像により計算する。

5. 評価実験

5.1 実験設定

本研究では、CIFAR100 [23] をデータセットとして利用した。CIFAR100 は、100 クラスの画像分類データセットであり、 32×32 の画像が学習用に 50,000 枚、評価用に 10,000 枚含まれている。また、一般的な CIL の実験設定 [7] に従い、無作為に抽出した 50 種のクラスを 1 番目のタスクとして扱った。加えて、残りのクラスを 10 クラス毎に分割することで他のタスクを生成し、合計 6 タスクから成るタスク列を生成した。そして、モデルが各タスクについて学習した後に、これまでに学習したすべてのクラスに対応する評価用データを用い Accuracy を計算した。本研究では、シナリオに含まれる全 6 タスクそれぞれの学習後に

表 1 AIA による各手法の比較

Table 1 The comparison among the four types of methods in AIA.

method	AIA
UCIR [14]	65.01 ± 0.26
Variation 1	66.61 ± 0.21
Variation 2	66.84 ± 0.28
All	67.14 ± 0.36

得られる Accuracy の平均値を評価指標として用いた。この評価指標は、Average Incremental Accuracy (AIA) [7] と呼ばれており、CIL 手法の評価指標として広く用いられている。

また、特徴抽出器として ResNet32 [1] を用い、提案フレームワークで用いる注意機構モジュールを各残差ブロックに SE block [12] と同様の方法で導入した。加えて、クラスに依存しない特徴の学習に用いる射影器と予測器については、それぞれ 2 層の全結合層からなる多層パーセプトロンを用いた。また、最適化手法として、確率的勾配降下法を利用し、最初のタスクのみその学習率を 0.2、エポック数を 400 と設定し、残りのタスクは学習率を 0.02、エポック数を 200 とした。加えて、モーメントと Weight decay の値としてそれぞれ 0.9、 $5e-4$ を用い、cosine decay により学習率スケジューリングを行った。また、バッチサイズとして 128 を設定し、保持サンプル数を各クラス当たり 20 とした。さらに、SimSiam で用いられたデータ拡張手法を複雑なデータ拡張手法として利用し、単純なデータ拡張手法として画像の水平反転と random crop を用いた。他のハイパーパラメータについては、UCIR [14] により導入されるハイパーパラメータとして、 $(\lambda_{\text{kd}}, \lambda_{\text{mr}}, m, K) = (1.0, 0.5, 0.7, 2)$ を設定した。また、本フレームワークにより導入されるハイパーパラメータとして、 $(\alpha, \lambda_{\text{attn}}^{\text{map,new}}, \lambda_{\text{attn}}^{\text{map,old}}, \lambda_{\text{attn}}^{\text{source}}) = (0.3, 4.5, 3.5, 0.5)$ を設定した。

5.2 実験結果

提案フレームワークは、クラスに依存しない特徴の学習、注意機構による特徴変換メカニズム、および、注意機構による特徴変換メカニズムの安定性向上方法により構成される。そこで、これら 3 つの構成要素による効果を示すため、以下に示す 4 種類の手法について比較実験を行った。

- UCIR : ベースライン手法。
- Variation 1 : UCIR にクラスに依存しない特徴の学習を組み込んだ手法。
- Variation 2: Variation 1 に注意機構による特徴変換メカニズムを組み込んだ手法。
- All : UCIR に提案フレームワークを組み込んだ手法。

ここで、比較実験の結果を表 1 に示す。なお、表 1 の AIA の値は 3 度の試行を行い導出した AIA の平均値と標準偏差を表す。表 1 より、提案フレームワークの導入により

表 2 可塑性と安定性に関する比較

Table 2 The verification of the effectiveness for plasticity and stability

method	可塑性	安定性
UCIR	66.15 ± 0.33	63.83 ± 0.46
All	66.31 ± 0.48	66.03 ± 0.51

UCIR の AIA を 2.13% 向上させていることがわかる。加えて、UCIR と Variation 1 を比べると、その AIA は 1.60% 異なり、これは自己教師情報が CIL シナリオにおいて有益な最適化対象であることを示していると考えられる。

次に、本フレームワークの導入によるモデルの可塑性と安定性への影響を検証するために、新規クラスに対する Accuracy の平均値と最初のタスクに含まれるクラスに対する Accuracy の平均値について、UCIR と All とを比較した。新規クラスに対する Accuracy の平均値は、モデルが新規タスクを学習するために十分可塑である場合、高い値を取ると考えられる。一方、最初のタスクに含まれるクラスに対する Accuracy の平均値は、モデルが忘却の影響を受けないほどに安定であれば、高い値を取ると考えられる。表 2 に比較の結果を示す。表 2 の 2 列目が新規クラスに対する Accuracy の平均値、3 列目が最初のタスクに含まれるクラスに対する Accuracy の平均値を表している。この結果から、提案フレームワークが可塑性を犠牲にすることなく安定性を向上させることに成功していることがわかる。

6. おわりに

本論文では、クラスに依存しない特徴の学習、注意機構による特徴変換メカニズム、および、特徴変換メカニズムの安定性向上手法の 3 つの要素により構成される新たな CIL フレームワークを提案した。また、ベンチマークデータを用いた評価実験を通して、本フレームワークが UCIR の AIA を向上させることを示した。今後の課題として、本フレームワークを他の CIL 手法に適用し、提案フレームワークの汎用性を確認することが挙げられる。

謝辞 本研究の一部は JSPS 科研費 JP21J21785 の助成による。

参考文献

[1] He, K., Zhang, X., Ren, S. and Sun, J.: Deep Residual Learning for Image Recognition, *CVPR*, pp. 770–778 (2016).

[2] Dosovitskiy, A., Beyer, L., Kolesnikov, A. et al.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, *ICLR* (2020).

[3] Masana, M., Liu, X., Twardowski, B. et al.: Class-incremental learning: survey and performance evaluation on image classification, *arXiv preprint arXiv:2010.15277* (2020).

[4] Mermillod, M., Bugaiska, A. and Bonin, P.: The

stability-plasticity dilemma: investigating the continuum from catastrophic forgetting to age-limited learning effects, *Frontiers in Psychology*, Vol. 4, p. 504 (2013).

[5] McCloskey, M. and Cohen, N. J.: Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem, *Psychology of Learning and Motivation*, Vol. 24, Academic Press, pp. 109–165 (1989).

[6] Hinton, G., Vinyals, O. and Dean, J.: Distilling the knowledge in a neural network, *arXiv preprint arXiv:1503.02531* (2015).

[7] Rebuffi, S. A., Kolesnikov, A., Sperl, G. and Lampert, C. H.: icarl: Incremental classifier and representation learning, *CVPR*, pp. 2001–2010 (2017).

[8] Chaudhry, A., Dokania, P. K., Ajanthan, T. and Torr, P. H.: Riemannian walk for incremental learning: Understanding forgetting and intransigence, *ECCV*, pp. 532–547 (2018).

[9] Liu, X., Zhang, F., Hou, Z. et al.: Self-supervised Learning: Generative or Contrastive, *IEEE TKDE*, (online), DOI: 10.1109/TKDE.2021.3090866 (2021).

[10] Chen, X. and He, K.: Exploring simple siamese representation learning, *CVPR*, pp. 15750–15758 (2021).

[11] Pei, W., Mayer, A., Tu, K. and Yue, C.: Attention please: Your attention check questions in survey studies can be automatically answered, *WWW*, pp. 1182–1193 (2020).

[12] Hu, J., Shen, L. and Sun, G.: Squeeze-and-excitation networks, *CVPR*, pp. 7132–7141 (2018).

[13] Chaudhry, A., Rohrbach, M., Elhoseiny, M. et al.: On tiny episodic memories in continual learning, *arXiv preprint arXiv:1902.10486* (2019).

[14] Hou, S., Pan, X., Loy, C. C. et al.: Learning a unified classifier incrementally via rebalancing, *CVPR*, pp. 831–839 (2019).

[15] Kurmi, V. K., Patro, B. N., Subramanian, V. K. and Namboodiri, V. P.: Do not Forget to Attend to Uncertainty while Mitigating Catastrophic Forgetting, *WACV*, pp. 736–745 (2021).

[16] Mai, Z., Li, R., Kim, H. and Sanner, S.: Supervised Contrastive Replay: Revisiting the Nearest Class Mean Classifier in Online Class-Incremental Continual Learning, *CVPR*, pp. 3589–3599 (2021).

[17] Rajasegaran, J., Khan, S., Hayat, M. et al.: itaml: An incremental task-agnostic meta-learning approach, *CVPR*, pp. 13588–13597 (2020).

[18] Gallardo, J., Hayes, T. L. and Kanan, C.: Self-Supervised Training Enhances Online Continual Learning, *arXiv preprint arXiv:2103.14010* (2021).

[19] Zhang, S., Shen, G., Huang, J. and Deng, Z.-H.: Self-Supervised Learning Aided Class-Incremental Lifelong Learning, *arXiv preprint arXiv:2006.05882* (2020).

[20] Zhu, F., Zhang, X.-Y., Wang, C. et al.: Prototype Augmentation and Self-Supervision for Incremental Learning, *CVPR*, pp. 5871–5880 (2021).

[21] Ji, Z., Li, J., Wang, Q. and Zhang, Z.: Complementary Calibration: Boosting General Continual Learning with Collaborative Distillation and Self-Supervision, *arXiv preprint arXiv:2109.02426* (2021).

[22] Khosla, P., Teterwak, P., Wang, C. et al.: Supervised Contrastive Learning, *NeurIPS*, Vol. 33, pp. 18661–18673 (2020).

[23] Krizhevsky, A.: Learning multiple layers of features from tiny images, Technical report, University of Toronto (2009).