機械知覚&ロボティクスグループ
中部大学　工学部 情報工学科

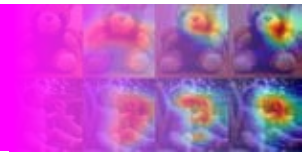
MPRG
MACHINE PERCEPTION AND ROBOTICS GROUP

2022年度　山下研究室　卒業論文発表　アブストラクト

Deep Learning, Loss function, Explainable AI

画像分類タスクにおける損失関数と注視領域の傾向調査

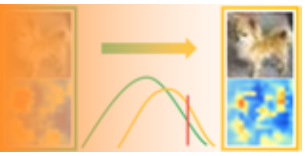
伊藤 天詞



Deep Learning, Drift Detection

ABN と Kolmogorov-Smirnov 検定によるデータのドリフト検知

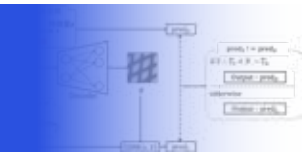
新田 常顧



Deep Learning, Anomaly Detection, autoencoder

Center Loss を導入した VAE による埋め込み空間における異常検知

井上 涼太



Deep Learning, Reinforcement Learning

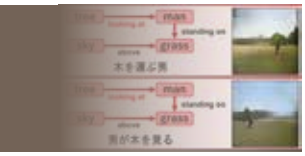
階層型強化学習における効率的な下位方策の分離と獲得

森 勇貴

Scene graph, Image Generation

シーングラフを用いた画像生成法における学習データの内容調査

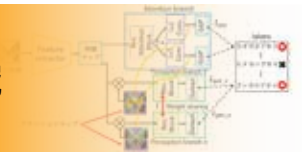
小松 翔太



Deep Learning, Multi-label, Explainable AI

多値ラベル分類 ABN を用いたチョウ斑紋の認識と地域変異に関する研究

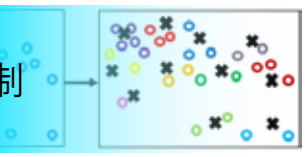
荒井 嵩貴



Deep Learning, Continual Learning

k-means 法を用いたデータ選択による継続学習における破滅的忘却の抑制

林 尚斗



Human-like Guidance, Deep Learning, Visual Saliency

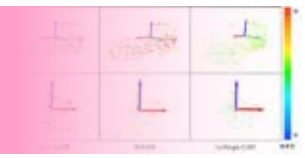
Human-like Guidance に適した視線情報データセットの作成

難波田 雅己

Point Cloud, Object Detection

3 次元点群データの重要度を考慮した物体検出の高精度化と分析

西尾 友佑



Deep Learning, Semantic Segmentation, Point Cloud

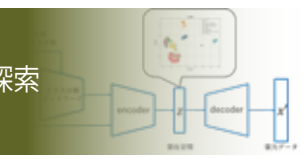
点群データに対するセマンティックセグメンテーションにおける CG データによる事前学習の有効性検証

中村 智也

Deep Learning, Bioinformatics, Autoencoder

Single-cell RNA-seq 解析における Star clustering と GMVAE を用いたデータのクラスタ数の探索

伊藤 啓太



画像分類タスクにおける損失関数と注視領域の傾向調査

EP19013 伊藤天詞　　指導教授：山下隆義

1. はじめに

畳み込みニューラルネットワーク (CNN) による画像認識モデルの学習は，正解ラベルと予測確率の損失を最小化することである．CNN の学習には様々な損失関数が使用される一方で，それらの効果について深く議論されていない．そこで本研究では，各損失関数における認識精度，特徴表現，注視領域の 3 つの観点で傾向を体系的に調査する．

2. 損失関数

画像認識モデルの損失関数は,「出力空間に対するもの」と「特徴空間に対するもの」の 2 つに大別することができる．

2.1. 出力空間に対する損失関数

Cross Entropy Loss　Cross Entropy Loss (CE Loss) は，正解クラス確率を 1 に近づける損失関数である．CE Loss は，N をサンプルの総数，$\boldsymbol{y}$ を正解ラベルが 1 の確率分布，$\hat{\boldsymbol{y}}$ を予測確率分布として，式 (1) で表現される．

$$L_{CE}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_i^T \cdot \log(\hat{\boldsymbol{y}}_i) \tag{1}$$

CE Loss は，$L_{CE}(\boldsymbol{y}, \hat{\boldsymbol{y}})$ が最小となるように学習する．

Complement Entropy Loss　Complement Entropy Loss (CoE Loss) は，Complement Objective Training [1] で提案された不正解クラス確率を平坦化する損失関数である．CoE Loss は，g を正解クラスのインデックスとして，式 (2) で表現される．

$$L_{CoE}(\hat{\boldsymbol{y}}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1, j\neq g}^{k} \frac{\hat{y}_{ij}}{1-\hat{y}_{ig}} \log\left(\frac{\hat{y}_{ij}}{1-\hat{y}_{ig}}\right) \tag{2}$$

CoE Loss は，$L_{CoE}(\hat{\boldsymbol{y}})$ が最大となるように学習する．CoE Loss を適用したモデルは CE Loss 最小化の後に CoE Loss 最大化を行うため，1 つのミニバッチで 2 回重みパラメータを更新する．

2.2. 特徴空間に対する損失関数

特徴空間に対する損失計算は，CE Loss と併せて用いる．

Center Loss　Center Loss は，特徴空間において，同じクラスの特徴量を近づける損失関数である．Center Loss は，入力画像と教師ラベルのペアを N サンプル含む集合 $D := \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$ に関して，学習可能なクラス重心を $\boldsymbol{w}^c$，特徴量を $\boldsymbol{f}_i$ として，式 (3) で表現される．

$$L_{Center} = \frac{1}{2}\sum_{i}^{N} \parallel \boldsymbol{f}_i - \boldsymbol{w}_{y_i}^c \parallel_2^2 \tag{3}$$

Center Loss は，対象クラスの特徴量とそのクラス重心との距離を最小化することで，クラス内分散を抑えることができる．

Prototype Conformity Loss　Prototype Conformity Loss (PC Loss) [2] は，特徴空間において，同じクラスの特徴量を近づけ，異なるクラスの特徴量を離す損失関数である．PC loss は，クラス数を k，正解クラスを y_i，学習可能なクラス重心 $\boldsymbol{w}^c$ と特徴量 $\boldsymbol{f}_i$ の関係を $\alpha_i = \|\boldsymbol{f}_i - \boldsymbol{w}_{y_i}^c\|_2$，$\beta_i = \|\boldsymbol{f}_i - \boldsymbol{w}_j^c\|_2$，$\gamma_i = \|\boldsymbol{w}_{y_i}^c - \boldsymbol{w}_j^c\|_2$ として，式 (4) で表現される．

$$L_{PC} = \sum_{i}^{N}\left\{\alpha_i - \frac{1}{k-1}\sum_{j\neq y_i}(\beta_i + \gamma_i)\right\} \tag{4}$$

PC Loss は，α_i を最小化することで特徴量をクラス重心に集め，$(\beta_i + \gamma_i)$ を最大化することでクラス間を離すように働く．

3. 評価実験

本実験では，CNN に CoE Loss，Center Loss，PC Loss を導入して傾向を調査する．

3.1. 実験概要

本実験では，認識精度だけでなく特徴空間を定量的評価，注視領域を定性的評価する．特徴空間の定量的評価には，クラス間の分離距離を評価する Silhouette score (Sil.) ，同じクラスの凝縮度合と異なるクラスの分散度合を評価する Calinski harabasz index (Cal.) を用いる．注視領域は，Grad-CAM を使用して可視化する．

3.2. 実験条件

ネットワークモデルは ResNet-20，活性化関数は ReLU，最適化手法はモーメンタム SGD，バッチサイズは 128，ドロップアウト率は 30%，学習の更新回数は 200 エポックである．また，事前学習として CE Loss を用いて 100 エポック学習する．学習用データセットは CIFAR-10 を用いる．

3.3. 実験結果

各損失関数における認識精度と特徴空間の定量的評価の比較を表 1 に示す．表 1 より，Center/PC Loss を導入すると認識精度，特徴空間のスコアが同等以上になることを確認した．また，CoE Loss + PC Loss の場合に最高精度となった．

表 1: 認識精度と特徴空間の定量的評価

		test acc.[%]	Sil.	Cal.
w/o CoE	CE	95.00	0.3773	2666.85
	Center	**95.32**	**0.7896**	**15588.98**
	PC	95.21	0.7877	15218.74
w/ CoE	CE	95.19	0.4653	3363.28
	Center	95.26	**0.7928**	14269.11
	PC	**95.36**	0.7924	**14280.56**

3.4. 注視領域の可視化

各損失関数におけるアテンションマップを図 1 に示す．図 1 より，Center/PC Loss を導入すると対象クラスの物体領域に注視することが確認できる．視覚的説明性の向上に併せて認識精度も同等以上になることから，Center/PC Loss を導入したモデルは画像内の小さな範囲に写る物体に対して有効であるといえる．

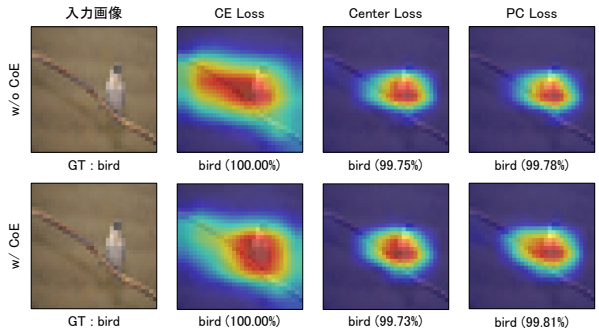


図 1: 各損失関数におけるアテンションマップの例

4. おわりに

本研究では，各損失関数における認識精度，特徴表現，注視領域の 3 つの観点で傾向を調査した．評価実験では，Center/PC Loss を導入した CNN は，物体を局所的に注視し，小さな範囲に写る物体に対して有効な特徴量を獲得できることが確認できた．今後は，Vision Transformer における傾向調査を行う予定である．

参考文献

[1] H. Chen, *et al.*, “COMPLEMENT OBJECTIVE TRAINING”, ICLR, 2019.

[2] A. Mustafa, *et al.*, “Adversarial Defense by Restricting the Hidden Space of Deep Neural Networks”, ICCV, 2019.

ABN と Kolmogorov-Smirnov 検定によるデータのドリフト検知

EP19080 新田常顧　　指導教授：山下隆義

1. はじめに

データのドリフトは，運用時のデータ分布がモデル学習時のデータ分布から時間とともに変化することであり，その検知は重要な課題である．ドリフトは，カメラのレンズの経年劣化により発生するノイズや，カメラの交換によって発生する位置ズレや反転などにより生じる．このようなドリフトは，機械学習モデルの性能低下の原因となる．そのため，データ分布の変化を監視することが重要となる．本研究ではモデルの判断根拠の可視化が可能である Attention Branch Network [1] と 2 標本検定によるドリフト検知法を提案する．

2. 関連手法

ドリフト検知は，モデル学習時のデータ分布と運用時のデータ分布を比較することで行われる．Rabanser らの手法 [2] は，それぞれのデータに対して次元削減を行い，2 標本検定により分布を比較してドリフトを検知する．学習済みモデルが出力するクラス確率分布に対して，2 標本検定の一種である Kolmogorov-Smirnov (KS) 検定を用いる検知手法が高精度であることを示した．

3. 提案手法

本研究では，ABN を用いたドリフト検知法を提案する．ABN では，Attention branch から得られるアテンションマップを Attention 機構に入力し，特定領域の特徴を強調して推論を行う．そのため，ドリフトによるアテンションマップの変化によって，入力画像の変化に対する特徴が強調され，ドリフトの検知精度が向上すると考える．例えば，図 3 に示すようにドリフトした画像に対するアテンションマップは，ドリフトなしと比較して変化する．このことから，アテンションマップの違いを考慮することで，ドリフト検知の高精度化が期待できる．ABN を用いたドリフト検知の流れを図 1 に示す．また，提案手法は以下のデータセットの前処理と Step1 から Step3 の流れでドリフト検知を行う．

データセットの前処理

データセットを学習用データ，検証用データ，テスト用データの 3 つに分割する．検証用データは学習用データと同じデータ分布であると仮定し，ドリフトなしのデータとする．テスト用データはモデル運用時のデータ分布と仮定し，ドリフトありのデータとする．

Step1. ABN の学習

学習用データを用いて ABN を学習する．

Step2. クラス確率分布とアテンションマップの算出

ドリフトなしのデータとドリフトありのデータから，指定したサンプル数を取得し，学習済み ABN に入力する．それぞれのデータ群に対して Attention branch の出力するクラス確率分布を (D^{Ab}, D^{Ab}_{drift})，アテンションマップを (D^{Am}, D^{Am}_{drift})，Perception branch の出力するクラス確率分布を (D^{Pb}, D^{Pb}_{drift}) とし，保存する．

Step3. ドリフト検知

Step2 で求めた各分布に対して，式 (1) に示す KS 検定を行い，p 値 (p_{Ab}, p_{Am}, p_{Pb}) を求める．

$$\begin{aligned} p_{Ab} &= \mathrm{KS}(D^{Ab}, D^{Ab}_{drift}) \\ p_{Am} &= \mathrm{KS}(D^{Am}, D^{Am}_{drift}) \\ p_{Pb} &= \mathrm{KS}(D^{Pb}, D^{Pb}_{drift}) \end{aligned} \tag{1}$$

p 値は，2 つの分布の母集団が同じであるという仮定の下で，KS から得られる検定統計量がそれ以上の値となる確率である．次に，式 (2) に示すように p 値の最小値を p_{min} とする．

$$p_{min} = \min(p_{Ab}, p_{Am}, p_{Pb}) \tag{2}$$

最後に，$p_{min} < \alpha$ であればドリフトありと判定する．ここで，α は従来手法と同値の閾値である．

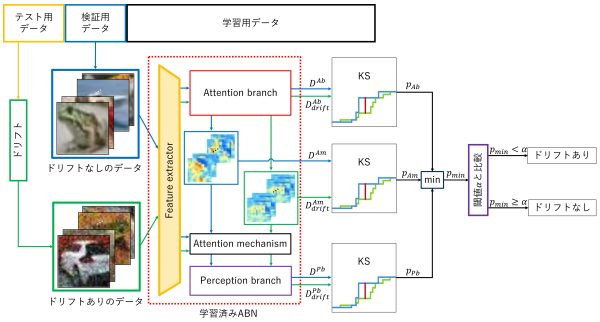


図 1: 提案手法によるドリフト検知の流れ

4. 評価実験

提案手法の有効性を調査するために，従来手法とドリフト検知精度の比較を行う．

4.1. 実験概要

本実験では，MNIST データセットと CIFAR-10 データセットを用いてドリフト検知を行う．各データセットの分割方法やデータの扱いは Rabanser らの手法 [2] に従う．ドリフトの再現には，ガウシアンノイズ，幾何変換，データの不均衡化を用いる．また，ドリフト検知のために必要とするサンプル数を評価するために，ドリフト検知に用いるサンプル数を 6 種類用意する．ここで，評価指標には各種ドリフトの再現手法での平均検知率を用いる．

4.2. 実験結果

Rabanser らの手法と提案手法の実験結果を図 2 に示す．図 2 より，ドリフト検知に用いるサンプル数に関わらず平均検知率が向上している．また，サンプル数が多いほど検知率が向上している．画像にガウシアンノイズを付与した際のアテンションマップを図 3 に示す．図 3 より，ガウシアンノイズをによるドリフトの際は注視領域が狭まっていることが分かる．ABN はアテンションマップを用いて推論を行うため，これらの変化を得ることでドリフトの検知精度が向上していると考えられる．

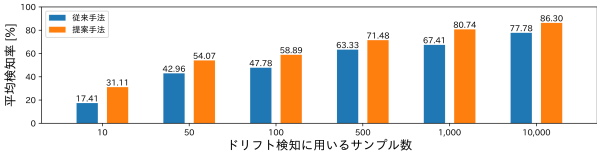
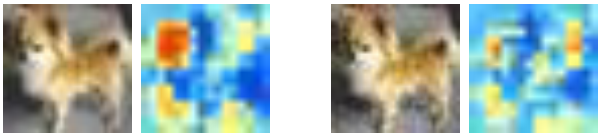


図 2: 提案手法と従来手法の平均検知率の比較



ドリフトなし　　ガウシアンノイズ

図 3: アテンションマップの比較

5. おわりに

本研究では，ABN を用いてドリフトを検知する手法を提案し，ドリフト検知精度が向上することを確認した．また，ドリフトに対するアテンションマップの変化を分析した．今後は，その他のドリフト検知手法との比較や，アテンションマップを用いたドリフト検知，ドリフトの再現手法を追加し，実験を行うことを検討する．

参考文献

[1] H.Fukui, *et al.*, “Attention Branch Network: Learning of Attention Mechanism for Visual Explanation”, CVPR, pp.10705-10714, 2019.

[2] Rabanser, *et al.*, “Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift”, NeurIPS, 2019.

Center Loss を導入した VAE による埋め込み空間における異常検知

EP19016 井上涼太　　指導教授：山下隆義

1. はじめに

正常データのみを用いて学習した Variational Autoencoder (VAE) [1] は，入力に対応した正常データの再構成が可能である．異常データを入力した場合は，正常データに近い画像が再構成され，入出力の類似度が小さいときに異常と検知する．このとき，VAE が再構成に失敗すると，誤検知を誘発する．そこで本研究では，入出力の類似度だけでなく，埋め込み空間における異常検知手法を提案する．埋め込み空間では，学習データから求めた重心を基準として異常データを検知する．また，異常検知の対象となる埋め込み空間のバラつきを Center Loss [2] によって抑制することで，更なる異常検知精度の向上を図る．

2. VAE を用いた異常検知

VAE は変分推論を軸とした生成モデルであり，中間層で得られる特徴量からデータを生成する．学習データ分布を確率分布で表現できるため，データの補間によって学習データに含まれないデータが生成できる．正常データの特徴を上手く捉えた VAE は，正常な特徴のみ再構成するため，異常データを入力した場合でも正常データとして再構成される．従って，入力データの類似度を求めることで異常データを特定できる．

3. 提案手法

埋め込み空間における異常検知を入出力の類似度による異常検知と組み合わせることで，異常検知精度の向上を図る．提案手法による異常検知の流れを図 1 に示す．

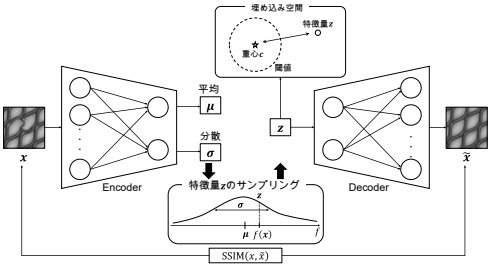


図 1：提案手法による異常検知

3.1. 埋め込み空間における異常検知

埋め込み空間における異常検知は，重心 $\boldsymbol{c}$ とテストデータの特徴量 $\boldsymbol{z}$ とのユークリッド距離 $d(\boldsymbol{z}, \boldsymbol{c})$ を求め，閾値を超える場合は異常とする．重心 $\boldsymbol{c}$ は，VAE に入力して得た全ての学習データの特徴量を平均して求める．

3.2. 画像空間と埋め込み空間を組み合わせた異常検知

従来の画像空間と 3.1. の埋め込み空間を用いて判定する．その流れを以下に示す．

Step1 VAE へ画像 $\boldsymbol{x}$ を入力して画像空間と埋め込み空間，それぞれで異常検知を行う．

$$\mathrm{pred}_S = \mathbf{1}[\mathrm{SSIM}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) > T_S]$$
$$\mathrm{pred}_D = \mathbf{1}[d(\boldsymbol{z}, \boldsymbol{c}) > T_D]$$

ここで，$\tilde{\boldsymbol{x}}$ は再構成画像，T_S は画像空間の閾値，T_D は埋め込み空間の閾値，$\mathbf{1}$ はインジケーター関数である．

Step2 採用する結果を決定するため，同じ入力から得られた結果 pred_S，pred_D を比較する．

Step3 結果が一致する場合はその結果を採用する．結果が異なる場合は類似度とユークリッド距離において閾値から離れている結果を採用する．従って，以下のように求める．

$$\mathrm{Output} = \begin{cases} \mathrm{pred}_S \text{ if } \mathrm{SSIM}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) - T_S > d(\boldsymbol{z}, \boldsymbol{c}) - T_D \\ \mathrm{pred}_D \text{ otherwise} \end{cases}$$

3.3. Center Loss を導入した VAE

正常データの重心位置を安定させるため，Center Loss によって埋め込み空間のバラつきを抑制する．VAE の特徴量から Center Loss を計算し，類似した特徴をまとめる．提案手法の損失関数は以下のようになる．

$$\mathcal{L} = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p(\boldsymbol{x}|\boldsymbol{z})] - D_{KL}[q(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})] + \lambda \frac{1}{2} \sum_{i=1}^{m} \| \boldsymbol{z}_i - \boldsymbol{c}_y \|_2^2$$

ここで，λ はハイパーパラメータ，m はバッチサイズ，y はクラスラベル，$\boldsymbol{c}_y$ はクラス重心である．

4. 評価実験

本実験では，異常検知精度を比較することで，提案手法の有効性を示す．

4.1. 実験条件

本実験では，MVTec-AD データセットを用いて VAE と提案手法を比較する．MVTec-AD は，異常検知用データセットであり，5 種類の Texture と 10 種類の Object クラスから構成される．本実験では，5 種類の Texture クラス，それぞれで学習と異常検知を行う．学習時の設定はバッチサイズを 128，学習回数を 300，学習率を 1.0×10^{-4}，潜在変数を 25 次元，重心数を 1，$\lambda = 1$ とする．

4.2. 異常検知精度

表 1 に従来手法と提案手法の検出率を示す．ここで，Ours1 は画像空間と埋め込み空間を組み合わせた異常検知による結果である．Ours2 は Ours1 に Center Loss を組み込んだ VAE による結果である．表 1 より，埋め込み空間による異常検知を組み合わせることで精度が向上している．これにより，埋め込み空間を活用した提案手法の有効性を示せた．

表 1：検出率の比較

Category	従来手法	Ours1	Ours2
carpet	0.724	0.698	**0.737**
grid	0.636	0.621	**0.695**
leather	0.500	0.527	**0.541**
tile	0.615	**0.698**	0.680
wood	0.601	0.664	**0.664**
平均	0.615	0.641	**0.663**

4.3. 考察

提案手法により正しく異常検知した例を図 2 に示す．ここで，黄色のマスクは異常箇所である．図 2 より，細かい異常箇所が広範囲に分布しており，テクスチャが不規則であることが確認できる．再構成画像全体がぼやけて正しくテクスチャを再構成できていないため，従来手法は誤検知する．一方，埋め込み空間による異常検知では重心との距離が大きくなるため，正常に検知できたと考えられる．Center Loss によって埋め込み空間のバラつきを抑制すると精度が向上することから，Center Loss を VAE に組み込むことは異常検知において有効であると考えられる．

入力画像　　再構成画像

図 2：提案手法により正しく異常検知したテストデータ

5. おわりに

本研究では，埋め込み空間における異常検知を入出力の類似度による異常検知と組み合わせた手法を提案した．提案手法は従来手法よりも異常検知精度が向上することを確認した．また，VAE に Center Loss を導入することで，異常検知精度の更なる向上を確認した．今後は，VAE を学習する際に使用した Center Loss の重心を使用した異常検知を検討する．

参考文献

[1] D. P. Kingma, *et al.*, “Auto-Encoding Variational Bayes”, ICLR, 2014.

[2] Y. Wen, *et al.*, “A Discriminative Feature Learning Approach for Deep Face Recognition”, ECCV, 2016.

階層型強化学習における効率的な下位方策の分離と獲得

EP19105 森　勇貴　　指導教授：山下 隆義

1. はじめに

階層型強化学習は，サブタスクごとの方策である下位方策と，その方策を選択する上位方策により，複雑なタスクを解くことができる．階層型強化学習の代表手法として，下位方策を学習過程で自動獲得する Meta Learning Shared Hierarchies (MLSH) [1] がある．MLSH は下位方策の自動獲得を実現している反面，役割が類似した下位方策が獲得されてしまう．そこで，本研究では下位方策間を分離する損失関数 JS Divergence Loss の導入を提案する．これにより，下位方策間で役割が異なる方策の獲得を促す．

2. Meta Learing Shared Hierarchies

MLSH とは，下位方策を自動獲得する階層型強化学習手法である．上位方策は一定ステップ毎に下位方策を選択し，選択された下位方策が環境に対する行動を出力する．そして上位方策と選択された下位方策のみ，報酬をもとに学習する．また，ここでの上位方策は下位方策が学習するごとに変化するため，一定ステップ経過で重みを初期化し，変化後の下位方策から再度適切な下位方策の選択方法を模索する．これにより，環境とのインタラクションを通じて得る経験を基に獲得することが可能となる．このように獲得した下位方策を用いて，上位方策のみ学習を行うことで複雑なタスクを解くことが可能である．

3. 提案手法

本研究では役割の異なる下位方策獲得を目的とし， JS Divergence Loss L_{JSdiv} を用いた下位方策分離法を提案する．図 1 に提案手法の構造を示す．

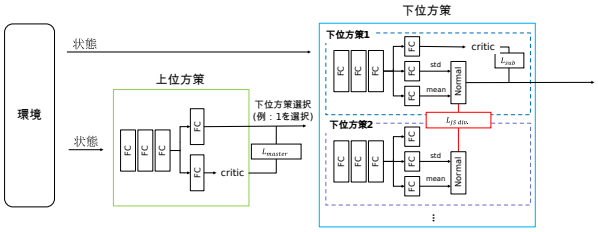


図 1: 提案手法の構造

下位方策は確率分布のため，確率分布間の相違度を示す JS Divergence を用いることで下位方策間の相違度を算出する．JS Divergence を用いた損失関数 L_{JSdiv} を式 (1) に示す．

$$L_{\mathrm{JSdiv}} = js_{\max} - \mathrm{clip}(\mathrm{JS}[\pi_{\mathrm{sub}i}, \pi_{\mathrm{sub}j}], 0, js_{\max}) \qquad (1)$$

ここで，$\pi_{\mathrm{sub}i}, \pi_{\mathrm{sub}j}$ は i, j 番目の下位方策 ($j = \backslash i$)，JS[·] は JS Divergence，$js_{\max}$ は下位方策間の JS Divergence が取りうる最大値である．clip$(\cdot, \min, \max)$ は与えられた値を最大値 max，最小値 min の範囲にクリップする関数である．JS Divergence は相違度を示すため下位方策間の分布が離れているほど高い値となる．そのため，下位方策間での JS Divergence が取りうる最大値 $js_{\max}$ を設定し，下位方策間の JS Divergence による相違度をクリップする．そして最大値 $js_{\max}$ から減算する．これにより下位方策間が離れているほど 0 に近く，下位方策間が類似しているほど高い値となる．また L_{JSdiv} を組み込んだ下位方策の損失関数を式 (2) に示す．

$$L_{\mathrm{sub}} = L_{\mathrm{sub}} + \alpha * L_{\mathrm{JSdiv}} \qquad (2)$$

ここで，L_{sub} は下位方策の損失関数，α は L_{JSdiv} に対する係数である．下位方策学習時に JS Divergence Loss を加算し学習することで，下位方策間を分離し役割を異なる下位方策獲得を促す．

4. 評価実験

AntBandits を用いた実験により，提案手法の有用性を確認する．

4.1. 学習環境

AntBandits は上/右どちらかに生成されるゴールへ向け Ant を制御するタスクである．観測情報は Ant の座標や関節角度等 (27 次元) であり，制御値は Ant8 関節のトルクである．報酬は，エージェントとゴール地点までの距離を負の報酬として与えた．エピソード終了条件は，環境が 2000 ステップ経過した場合である．AntRightUp は，生成されるゴールを右上としゴールへ向け Ant を制御するタスクである．報酬は，エージェントがゴール地点に到達している場合のみ +1 与える．観測情報とエピソード終了条件は AntBandits と同様である．

4.2. 実験概要

AntBandits タスクにて MLSH と提案手法をそれぞれ下位方策 240,000 ステップ，上位方策 400,000 ステップ学習する．獲得した下位方策の相違度の比較と可視化を行い，その傾向を比較する．その後，獲得した下位方策を用いて AntRightUp タスクを上位方策のみ 60,000 ステップ学習する．そして，報酬値の比較を行い，獲得した下位方策の有用性を検証する．本実験では下位方策を 4 つ，JS Divergence の学習係数 α を 0.5，$js_{\max}$ を 5.0 として学習する．学習アルゴリズムには Proximal Policy Optimization[2] を用いる．

4.3. 実験結果

表 1 に JS Divergence による下位方策の相違度，図 2 にその可視化結果，表 2 に AntRightUp タスクに対する報酬値を示す．表 1 より，提案手法は MLSH よりも相違度が 10 倍程度高い．また，図 2 から，MLSH は上に行く下位方策が 2 つ獲得されていることが分かる．一方で提案手法は，停止，上，右，右下と異なる下位方策である．また，表 2 から，提案手法は MLSH よりも高い報酬を獲得した．

表 1: 下位方策の相違度比較

	下位方策 1	下位方策 2	下位方策 3	下位方策 4	平均値
MLSH	521.362	776.375	779.551	893.275	742.641
提案手法	7999.048	5261.252	7582.035	9574.281	7604.154

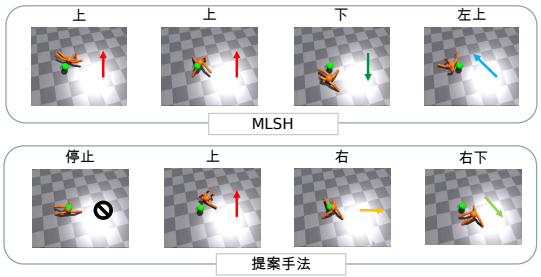


図 2: 下位方策の可視化結果

表 2: AntRightUp タスクに対する報酬値

	最大値	最小値	平均値
MLSH	7.0	0.0	0.07
提案手法	11.0	0.0	0.23

これらの結果から，類似した下位方策を学習する MLSH の問題点を解決することができた．

5. おわりに

本研究では，役割の異なる下位方策獲得を目的とし，MLSH に JS div.Loss を組み込む手法を提案した．実験結果として，類似した下位方策が学習されることを抑制できた．今後は，他環境での実験や方策数を変化させた際に JS div.Loss がどの程度有用であるか調査を行う．

参考文献

[1] K. Frans, *et al.*, “Meta Learning shared hierarchies-meta”, ICLR, 2018.

[2] J. Schulman, *et al.*, “Proximal Policy Optimization Algorithms”, *arXiv preprint arXiv:1707.06347*, 2017.

シーングラフを用いた画像生成法における学習データの内容調査

EP19044 小松 翔太　　指導教授：山下隆義

1. はじめに

シーングラフを用いた画像生成では，オブジェクト間の関係をエッジとして表現したシーングラフを入力し，画像を生成する．代表的な手法である Image Generation from Scene Graph (sg2im) [1] は，シーングラフから各オブジェクトのレイアウトを作成し，画像を生成する．しかしながら，オブジェクトの色や向きなどの詳細な情報が不足すると画像を正しく生成できない．この問題は学習データに起因すると考えられる．そこで本研究では，学習時に使用する代表的なデータセットである visual genome (VG) [2] から得られるシーングラフを調査し，どのようなシーングラフの場合に正しく画像生成が可能であるかを明らかにする．

2. Image Generation from Scene Graph

sg2im [1] は，シーングラフから画像生成する代表的な手法である．図 1 に示すように，まずシーングラフの各ノードでグラフ畳み込み処理をすることで，各ノードの特徴量を得る．その後，特徴量から Bounding box と Segmentation mask を含んだシーンレイアウトを作成する．作成したシーンレイアウトを生成モデルに入力して画像を生成する．

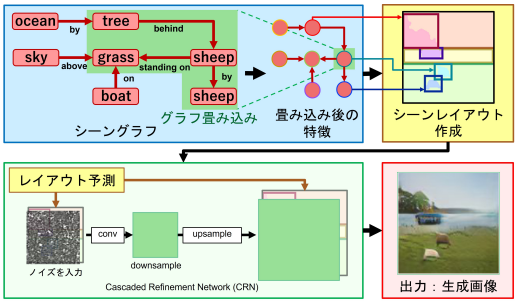


図 1：sg2im の概略図

3. visual genome

VG [2] は，風景画像や動物など様々な画像を含むデータセットである．各画像に対して，オブジェクトに対する Bounding box と，オブジェクトの関係をシーングラフで表現したデータが含まれている．データ内には，オブジェクトを示す単語は 33,877 種類，関係を示す単語は 42,374 種類存在する．

4. 画像生成実験

本実験では，調査内容と画像生成実験に用いる VG データの調整，実験結果について説明する．

4.1. 調査内容

正しく画像を生成できないシーングラフの調査するために，以下の A. と B. の 2 項目について調査を行う．

A. 学習に用いるデータに含まれる総量やオブジェクト，関係の出現数の調査（4.2 節）

B. sg2im の事前学習済みモデルを用いて画像生成を行い，正しい画像を生成できるか確認（4.3 節）

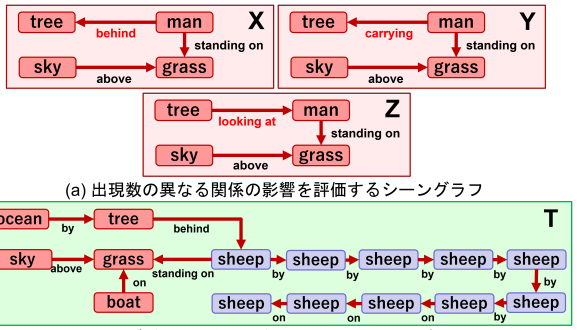
調査 B. で対象とするシーングラフを図 2 に示す．図 2 のシーングラフを用いて，どのようなシーングラフの場合に正しく画像生成が可能かを調査するために，以下の C. と D. の 2 項目の調査を行う．

C. データセット内の出現数が異なる関係に対して正しく画像が生成できるか（シーングラフ X, Y, Z）

D. シーングラフ内のオブジェクト数を変えた時に正しく画像を生成できるか（シーングラフ T）

4.2. 学習データの調査結果 (A, B)

sg2im は，データを train, val, test に分割し，出現回数が極端に少ないオブジェクトと関係を削除する前処理を行う．残ったオブジェクトと関係の種類は 178 と 45 となり，1 つのサンプルにつきオブジェクトは平均 10 個，関係は平均 5 個である．処理後のデータの出現数を図 3 に示す．

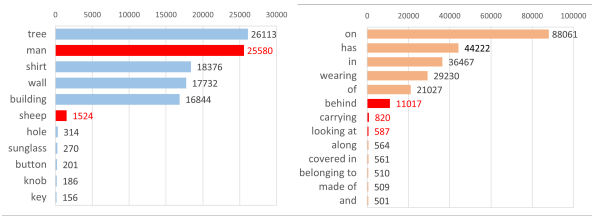


(a) 出現数の異なる関係の影響を評価するシーングラフ

(b) オブジェクト数の影響を評価するシーングラフ

図 2：実験に用いるシーングラフ

図 3 では，各オブジェクト，関係の出現数を上位 5 個と下位 5 個を示している．また，今回の実験で着目するものを赤色で示している．



(a) オブジェクトの出現量　　(b)関係の出現量

図 3：オブジェクト，関係の合計出現数

4.3. 画像生成の調査 (C, D)

図 2(a)X, Y, Z のシーングラフによる生成結果を図 4X, Y, Z に示す．図 4X, Y, Z の生成結果から，オブジェクト間の関係を変えたシーングラフを入力しても，大まかに理解可能な画像が生成されている．よって，データセット内の出現数は生成品質に影響しないと考えられる．次に，図 2(b)T のシーングラフによる生成結果を図 4T に示す．図 4T の生成結果から，1 つのシーングラフに含まれるオブジェクト数が増加すると背景に存在する木が正しく生成できていない．また，どの生成でも羊の向きは変化せず単調な生成であることを確認した．よって，1 つのシーングラフに使用するオブジェクト数が生成画像の品質に影響することが分かった．これはオブジェクトが増えることで関係も増え，シーングラフが複雑になるためであると考える．

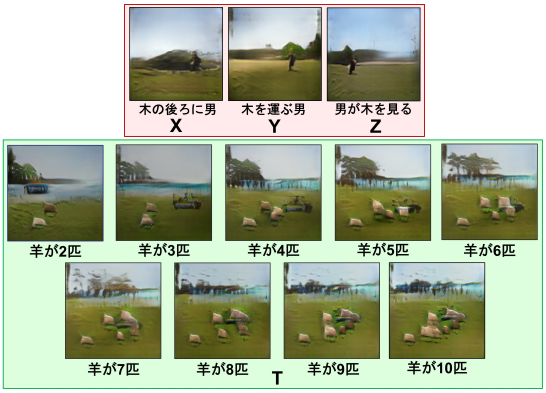


図 4：図 2 のシーングラフによる生成結果

5. おわりに

本研究では，sg2im による学習データ処理後の内容調査を行った．実験より，1 つのシーングラフのオブジェクト数が生成画像に影響することを確認した．今後は，他データセットの調査や今回発見した問題点の改善方法を模索する．

参考文献

[1] J. Johnson *et al.*, “Image generation from scene graphs.” CVPR, 2018.

[2] R. Krishna *et al.*, “Visual genome: Connecting language and vision using crowd-sourced dense image annotations.” IJCV, 2017.

多値ラベル分類 ABN を用いたチョウ斑紋の認識と地域変異に関する研究

EP19004 荒井嵩貴　　指導教授：山下隆義

1. はじめに

日本の本州に生息する蝶類であるギフチョウは，翅の斑紋に地域変異があることが知られている．この地域変異は地形，環境，食草などの要因が関与すると考えられているが，その解明には至っていない．そこで本研究では，ギフチョウの標本画像と各生息地域の食草の分析から食草と斑紋の関係性を明らかにすることを目的とし，Attention Branch Network (ABN) を用いた食草分類とその判断根拠の可視化を行う．

2. Attention Branch Network

ABN [1] は，視覚的説明と画像分類タスクの精度向上を目的とし，推論時における判断根拠の可視化を行う手法である．ABN は入力から特徴マップを獲得する Feature extractor，特徴マップを受け取り，クラス識別とアテンションマップの生成を行う Attention branch (AB)，最終的な各クラス確率を出力する Perception branch (PB) で構成される．

3. 提案手法

ギフチョウサンプルの数は種類や採集場所に依存するため，生息地によりサンプル数に偏りが存在する．そのため，少数クラスの分類が困難になり，認識精度の低下に繋がる．また，複数の生息地域にまたがることもある．そこで本研究では，図 1 に示すクラス不均衡を考慮した損失関数の導入及び，多値ラベルに対応した ABN による分類手法を提案し，これらの問題に対応する．

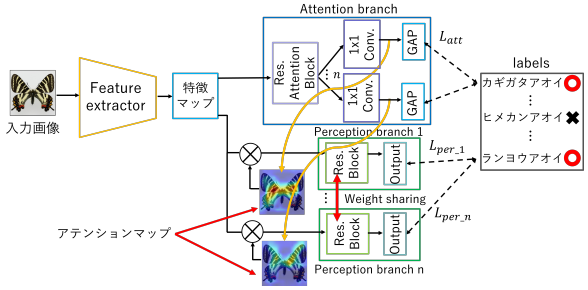


図 1: n ラベル分類に拡張した ABN の構造

提案手法では，AB にて Global Average Polling (GAP) により求めた各クラスの予測結果と正解ラベルから学習し，アテンションマップを出力する．その後，特徴マップとの乗算をクラスごとに行って PB を拡張し，多値ラベル分類を導入する．学習には，クラス不均衡の軽減を目的とした重み付きフォーカルロス [2] を式 (1) の AB の損失関数 L_{att} に導入する．重み付きフォーカルロスを式 (2) に示す．

$$L = L_{att} + \sum_{k=1}^{n} L_{per\_k} \tag{1}$$

$$\begin{aligned} L_{att} = -\sum_{k=1}^{n} w_k \big(& (1-\sigma(\hat{y}_p^k))^{\gamma} \log(\sigma(\hat{y}_p^k)) y^k \\ & + \sigma(\hat{y}_p^k)^{\gamma} \log(1-\sigma(\hat{y}_p^k))(1-y^k) \big) \end{aligned} \tag{2}$$

ここで，w_k はクラス k の重み，$\hat{y}_p^k$ は logits，y^k は正解ラベル，$\sigma(\cdot)$ はシグモイド関数，γ はハイパーパラメータである．PB の損失関数は L_{per_1} から L_{per_n} までの n クラス分の和となる．多値ラベルによる食草のクラス予測後，PB から予測結果を出力する．学習した多値ラベル分類 ABN を用いて，アテンションマップを可視化し，地域ごとに平均した結果を地域変異の調査に用いる．

4. 評価実験

提案手法とサンプル数の調整による評価実験を行う．

4.1. データセット

データセットは藤岡知夫の蝶類コレクションのオスのギフチョウ標本を使用する．ギフチョウが食べる食草である 7 種類のカンアオイを正解ラベルとし，0 から 6 までの ID を割り振る．ラベルごとの内訳を表 1 に示す．本実験では，クラス不均衡を軽減するために，中部地方や近畿地方に分布するヒメカンアオイと新潟県や福島県に分布するユキグニカンアオイのサンプル数を調整した場合と比較する．

表 1: サンプル数調整前後の内訳

ID	正解ラベル	サンプル数 [枚]	
		調整前	調整後
0	カギガタアオイ	235	235
1	カントウカンアオイ	124	124
2	コシノカンアオイ	341	341
3	ヒメカンアオイ	2,075	**400**
4	ミヤコアオイ	340	340
5	ユキグニカンアオイ	145	**245**
6	ランヨウアオイ	359	359

4.2. 実験概要

ギフチョウサンプルを調整せず，重み付きフォーカルロスを用いない場合をベースラインとし，重み付きフォーカルロスのみ導入した場合と，さらにサンプル調整した場合の F 値のスコアとアテンションマップを比較する．モデルは ResNet18 を backbone とした多値ラベル分類 ABN とする．学習回数は 100 エポック，学習率は 0.01，バッチサイズは 8，最適化手法は SGD，ImageNet による事前学習を行う．

4.3. 実験結果

ID ごとの比較結果を表 2 に示す．表 2 より，平均スコアに着目すると，重み付きフォーカルロスのみを導入した場合は 0.4pt 低下したが，さらにサンプル調整をした場合は 1.87pt 向上した．よって，損失関数とサンプル数の両方を考慮することで，クラス不均衡が改善されると考える．

表 2: ID ごとの F 値 [%]

ID	多値ラベル分類 ABN	重み付きフォーカルロス	重み付きフォーカルロス +サンプル調整
0	**100.00**	**100.00**	99.02
1	**100.00**	98.30	88.37
2	80.32	75.47	**82.60**
3	93.83	**94.51**	87.01
4	92.61	87.50	**95.17**
5	53.33	65.21	**82.22**
6	**99.23**	95.52	97.98
平均	88.47	88.07	**90.34**

次に，アテンションマップのラベルごとの平均を図 2 に示す．図 2(a) では羽の左側及び右側中央部分をわずかに注視し，図 2(b) では羽の左上部分を注視する傾向が見られた．

(a) ヒメカンアオイ:3 を食草とするギフチョウ

(b) ユキグニカンアオイ:5 を食草とするギフチョウ

図 2: アテンションマップの平均

5. おわりに

本研究では，重み付きフォーカルロスの導入とサンプル数を考慮した多値ラベル分類 ABN を用いて，食草分類とその判断根拠の可視化を行った．結果から，クラス不均衡の改善とラベルごとで注視領域が変化することを確認した．今後は，食草および採集地の環境情報を考慮した詳細な斑紋の解析を行う予定である．

参考文献

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

[2] N. Sarafianos, *et al.*, “Deep Imbalanced Attribute Classification using Visual Attention Aggregation”, ECCV, 2018.

k-means 法を用いたデータ選択による継続学習における破滅的忘却の抑制

EP19084 林尚斗　　指導教授：山下隆義

1. はじめに

IoT の普及により，オンラインで蓄積した膨大なデータを深層学習モデルの学習に利用することが期待されている．一方，全てのデータを保持することはハードウェアなどの問題から困難である．この問題に対して，学習に利用したデータを破棄しながら継続的に学習を行う継続学習がある．ラベル無しデータを利用する継続学習法である Lifelong Unsupervised Mixup (LUMP) [1] は，破滅的忘却を抑制するためにデータ拡張法の mixup と過去の学習に利用したデータの一部を保存するリプレイバッファを活用する．リプレイバッファに保存するデータは，ランダムで選択されるため，データに偏りが生じ，性能が劣化することがある．そこで本研究では，k-means 法を用いてデータの分布を考慮したデータ選択を行うことで破滅的忘却を抑制する．

2. Lifelong Unsupervised Mixup

継続学習は，一定の間隔で学習に利用したデータを破棄し，新しいデータで学習を行う．学習過程において学習用データが変化するため，新しいタスクのデータに適合し，以前に学習したタスクを解くことができなくなる破滅的忘却が生じる．教師なし継続学習法の LUMP は，現在のタスクのデータとリプレイバッファに保存されている過去に学習したタスクのデータを mixup により合成して学習に利用する．これにより，破滅的忘却を抑制する．

3. 提案手法

LUMP は，リプレイバッファに保存するデータをランダムに選択しているため，データ分布を考慮した学習が難しい．そこで本研究では，LUMP にクラスタリング手法の k-means 法を導入して，データ分布を考慮したデータ選択を行う．提案手法の流れを図 1 に示す．

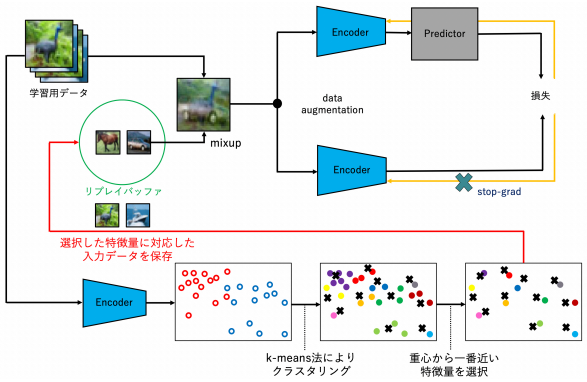


図 1：提案手法の流れ

提案手法では，k-means 法により学習用データの分布を複数のクラスタに分割し，各クラスタから 1 つずつデータを選択することになるため，元のデータ分布を保持したデータ選択を行う．以下に提案手法の流れを示す．

Step1．学習

1 タスク目は学習用データ，2 タスク目以降は学習用データとリプレイバッファのデータを mixup により合成したデータを入力データとし，自己教師あり学習を行う．ここでは，入力データに異なるデータ拡張を適用してそれぞれをエンコーダに入力する．そして，片方のエンコーダの出力とプレディクタの出力のコサイン類似度を損失として求め，片方のエンコーダにのみ勾配を逆伝播し，エンコーダを更新する．

Step2．k-means 法によるデータ選択

学習用データをエンコーダに入力して特徴量を獲得し，k-means 法によりクラスタリングする．k-means 法のクラスタ数は，リプレイバッファに保存するデータ数とする．各クラスタの重心から一番近いデータを選択し，リプレイバッファに保存する．2 タスク目以降でリプレイバッファにデータを保存する時，これまで学習したタスクのデータ数が均等になるように，リプレイバッファのデータを選択したデータで置き換える．

4. 評価実験

提案手法と従来手法 LUMP において，クラス分類問題の分類精度と忘却率を比較する．

4.1. 実験概要

1 タスクずつ継続学習を行い，タスクごとの分類精度を評価する．データセットは 2 クラスずつ 5 個のタスクに分割した Split CIFAR-10，ネットワークは ResNet-18，自己教師あり学習法は SimSiam，学習回数は 200 エポック，バッチサイズは 256，リプレイバッファのサイズは 256 である．評価の際には，k-NN 法を用いてクラス分類を行う．k-NN 法は，テストデータから近い k 個の学習用データに付与された正解ラベルの中で最も多いクラスを推論結果とする．

4.2. 評価指標

評価指標として，平均精度と平均忘却率を用いる．平均精度は，すべてのタスクを学習し終えたモデルを利用した時の各タスクの精度の平均値である．また，平均忘却率は，各タスクの最高精度となった精度とすべてのタスクを学習し終えたモデルの精度の差の平均値である．

4.3. 評価結果

LUMP の評価結果を表 1，提案手法を表 2 に示す．表 1，表 2 から，提案手法は LUMP と比較して最終タスクまで学習した時の平均精度を 0.98pt 向上し，平均忘却率を 5.67pt 抑えることができた．以上より，k-means 法を利用してデータ選択したリプレイバッファを用いて学習することは，破滅的忘却の抑制に有効であるといえる．

表 1：LUMP の精度 [%] と忘却率 [pt]

		評価タスク					平均
		1	2	3	4	5	
学習タスク	1	**96.90**	-	-	-	-	96.90
	2	88.50	**89.80**	-	-	-	89.15
	3	88.25	83.99	**95.20**	-	-	89.14
	4	87.95	80.25	89.14	**98.80**	-	89.04
	5	92.85	79.00	86.90	92.65	**98.15**	89.91
忘却率		4.05	10.80	8.30	6.15	-	7.33

表 2：提案手法の精度 [%] と忘却率 [pt]

		評価タスク					平均
		1	2	3	4	5	
学習タスク	1	**96.65**	-	-	-	-	96.65
	2	95.50	**83.20**	-	-	-	89.35
	3	95.15	80.25	86.15	-	-	87.18
	4	92.20	80.50	**88.45**	**96.90**	-	89.51
	5	94.90	81.70	87.60	94.35	**95.90**	90.89
忘却率		1.75	1.50	0.85	2.55	-	1.66

5. おわりに

本研究では，k-means 法を用いてデータ選択したリプレイバッファによる継続学習法を提案した．評価実験により，提案手法が LUMP と比べ高精度かつ低い忘却率となり，破滅的忘却をより抑制することを示した．今後は，勾配空間に k-means 法を適用したデータ選択を導入した継続学習を行う予定である．

参考文献

[1] D. Madaan, *et al.*, “Representational Continuity for Unsupervised Continual Learning”, ICLR, 2022.

Human-like Guidance に適した視線情報データセットの作成

EP19078 難波田雅己　　指導教授：山下隆義

1. はじめに

自動車の次世代ナビゲーションとして，人間と同じ様な誘導を行う Human-like Guidance (HℓG) [1] の研究が取り組まれている．HℓG を実現するには，運転者が注視すべき物体を推定し，その物体に着目した案内をする必要がある．そのため，シーン状況に合わせて運転者がどこを注視するか推定することが重要となる．これまでに，HℓG に特化した視線情報を含むデータセットは存在しない．そこで本研究では，HℓG ナビゲーションに特化した視線情報データセットを作成する．また，作成したデータセットを利用した視線推定の実験を行い，有効性を検証する．

2. HℓG のためのデータセット作成

HℓG に特化した視線データセットの作成を目的とし，運転シーンにおける人の視線を収集する．実世界で様々なシーンのデータを収集することは困難であるため，自動運転開発シミュレータである CARLA を用いる．視線データの収集には，眼球の動きが計測可能な Tobii Pro X3-120 を使用する．これらを用いて作成したデータセットを Driving Gaze Dataset と呼ぶ．図 1 に視線データの収集の流れを示す．

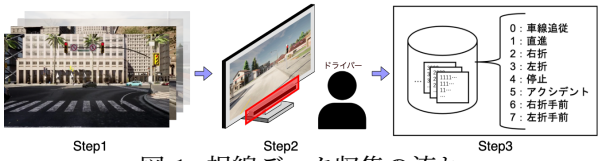


図 1: 視線データ収集の流れ

視線の収集方法は，以下の通りである．

Step1：運転シーンの動画像作成

CARLA の自動運転機能を用いて様々なシーンの動画像を収集する．1 つの動画像は 60fps で約 3 分間とし，20 本作成する．

Step2：視線収集

2 種類の視線データを収集する．1 つ目は通常運転時の視線情報である．既存のカーナビゲーションのように「100m 先を右方向です」といった案内に沿った視線 (自然な視線) の収集を目的とする．2 つ目はナビゲーションに適した視線情報である．事前に動画像を見せ指示文に関連した物体に視線を向けるように指示して,「前方車に追従してください」といった人間のような道案内を行った視線 (恣意的視線) の収集を目的とする．それぞれのデータは人間毎の視線情報の多様性を考慮するため，複数人で収集する．

Step3：ラベルデータの付与

シーン状況を Straight (車線追従/直進)，Turn (右折/左折/右折手前/左折手前)，Stop(停止/アクシデント) と定義し，各フレームに対してラベル付けを行う．また，ラベルの数値で埋められたラベルマップデータを作成し，デコーダに入力する特徴マップと連結する．

3. 評価実験

視覚的顕著性予測の代表的な手法である Dilated Inception Network (DINet)[2] を用いて，Driving Gaze Dataset の有効性を検証する．

3.1. 実験概要

Driving Gaze Dataset には自然な視線，恣意的視線がそれぞれ 6 人分含まれ，学習用は約 100,000 フレーム，評価用は約 3,200 フレームである．これらのデータを用いた 4 つの手法について比較，評価を行う．1 つ目は自然な視線のみでの学習（手法 1），2 つ目は恣意的視線のみでの学習（手法 2）である．3 つ目は手法 2 にラベルマップデータを付与して学習する（手法 3）．4 つ目は手法 3 に用いる動画像データについて，各シーン（Straight/Turn/Stop）の割合を 1:8:1 に調整して学習する（手法 4）．評価指標には CC，sAUC を使用する．

3.2. 実験結果

表 1 に各手法の評価結果，図 2 に各手法の視線推定結果を示す．図 2 から，手法 1 では画面中央付近に視線が集中した．手法 2 では右折時で左前方車両に視線を推定する傾向を確認した．しかし，遠方車両に対しては追従する視線が確認できず，手法 1 と同様に中央付近に視線を推定する結果であった．また，表 1 から手法 1 より低い精度であることを確認した．手法 3 では右折時の左前方車両への視線に加え，前方車両追従時に遠方車両を追従する視線を確認した．それに加え，表 1 から手法 2 より高い精度であることを確認した．手法 4 では，手法 3 と同様の推定結果を獲得しつつ，表 1 から手法 3 よりさらに高い精度であることを確認した．

表 1: 各評価指標での定量的評価

評価指標	手法 1	手法 2	手法 3	手法 4
CC	0.4685	0.4325	0.4306	**0.4802**
sAUC	0.5882	0.5512	0.6002	**0.6446**

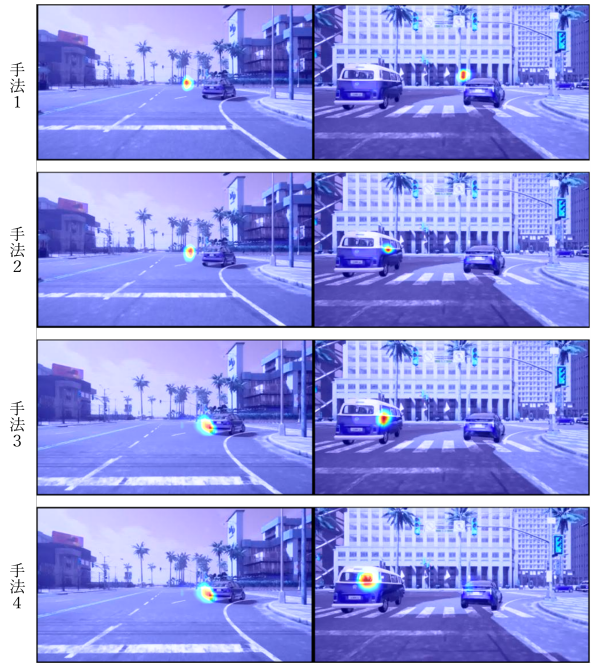


図 2: 各データにおける視線推定結果

3.3. 考察

手法 1 の結果は，人間の視線情報にあるセンターバイアスが原因と考えられる．センターバイアスは，人間が動画像を見る時に周辺視野を用いて情報を得るため，視線が中央に集まる現象である．一方，恣意的視線は視線データ収集時に視線の誘導を行うため，センターバイアスが減少し，手法 2 において左前方車への視線推定が可能となった．手法 3 は，どの状況であるかのシーン情報を追加することで特有の視線特徴を捉えやすくなり，右折時の左前方車両に対する視線に加えて，遠方車両を追従する視線を推定したと考えられる．手法 4 では，センターバイアスの主な原因である直進シーンを削減することでクラス均衡を抑制し，精度が向上したと考えられる．

4. おわりに

本研究では，HℓG に特化した視線情報データセットを作成し，DINet を用いて有効性を調査した．実験結果から，恣意的視線とラベルデータを入力することでナビゲーションにつながる視線を獲得し，各シーンの割合を調整することで精度の向上を確認した．今後は，時系列情報を考慮したモデル等の他モデルでの実験を行い，精度の比較を行う．

参考文献

[1] B.Wang, *et al.*, “Landmarks Based Human-like Guidance for Driving Navigation in an Urban Environment”, ITCS, 2018.

[2] S.Yang, *et al.*, “A Dilated Inception Network for Visual Saliency Prediction”, TMM, 2019.

3 次元点群データの重要度を考慮した物体検出の高精度化と分析

EP19079　西尾友佑　　指導教授：山下　隆義

1. はじめに

物体検出は，歩行者や車両などの物体を検出するタスクであり，自動運転支援などの重要な機能の 1 つである．自動運転支援では，悪天候や夜間でも高い検出精度を実現する必要がある．そこで，LiDAR で取得した 3 次元点群データから対象物体を検出する手法を用いることが多い．代表的な手法として PointRCNN[1] がある．PointRCNN はより高い精度で物体検出できるものの，どの領域に着目したか定かではない．そこで，本研究では PointRCNN の中間モジュールに Context Attntion Network(CAN)[2] を導入し，検出時の判断根拠を明らかにする．

2. 提案手法

本研究では物体検出時に注目した点群の重要度を算出し，判断根拠とする．提案手法の詳細を以下で説明する．

2.1 処理の流れ

図 1 に提案手法の処理の流れを示す．ここで，図 1 の B はバッチサイズ，N は入力点群，N_1 は RoI Pooling 数，D は特徴次元数である．提案手法は Two-Stage 型のネットワークで構成される．まず，1st-Stage では，入力点群 X を PointRCNN に入力し矩形領域 (BBox) X_1 を推定する．そして，BBox X_1 内の点群に対して RoI Pooling を行い，特徴量 X_2 と点群座標 $\hat{\mathbf{x}}$ を獲得する．2nd-Stage では，特徴量 X_2 と点群座標 $\hat{\mathbf{x}}$ を結合した特徴量 F を CAN に入力する．CAN から出力されたアテンション S と特徴量 F を内積して特徴量 F_s を獲得する．最後に，特徴量 F_s を用いて BBox X_1 の位置や向きを修正する．PointRCNN の中間モジュールに CAN を導入することで，各点の重要度を算出しながら BBox を修正できる．また，各点に対するアテンション S から，各点の重要度を可視化できる．

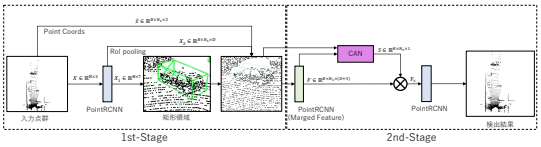


図 1：提案手法のネットワーク構造

2.2 CAN の導入

CAN の構造を図 2 に示す．ここで，CAN はサンプリング，グルーピング，プーリングをする SAG，SAG の特徴量を point-wise conv. で次元方向にアップサンプリングする FP 及び，全結合層で各点の重みを出力する FC で構成される．また，図 2 の N は点群数，N_1, N_2 は SAG のサンプリング数，D, D_1, D_2, D_3 は次元数である．まず，1 層目の SAG に点群 $\mathbf{x}$ と特徴量 F を入力し，特徴量 F_1 を抽出する．ここで，異なるグルーピングを考慮するために，1 層目の SAG を 3 つの Head に拡張し，各 Head で得た特徴量を連結する．これにより，各 Head 内で異なる特徴表現を獲得できる．次に，2 層目の SAG で特徴量 F_1 を入力して特徴量 F_2 を獲得する．1 層目の FP で特徴量 F_3 を抽出後，2 層目の FP で特徴量 F_{p2} を獲得する．そして，特徴量 F_4 を FC に入力しアテンション S を出力する．

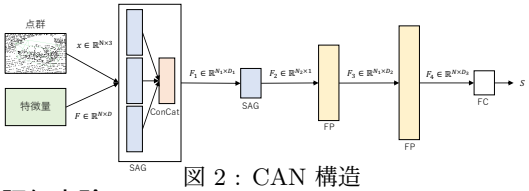


図 2：CAN 構造

3. 評価実験

提案手法の有効性を評価実験により検証する．

3.1 実験条件

本実験では，最適化手法を Adam，初期学習率を 0.01，エポック数を 80，バッチサイズを 12 とする．学習データには KITTI 3D Object Detection Benchmark を用いる．対象クラスは Car，Pedestrian，Cyclist である．評価データには，KITTI test 3D Object Detection Benchmark と KITTI test Bev Object Detection Benchmark を用いる．評価指標には BBox (3d) と BBox の向き (Aos)，俯瞰視点 (Bev) での Average Precision(AP) を用いる．また，CAN の SAG 内の head 数による精度と重要度の違いを確認するために，1 層目の Head 数を 1 つにした Single CAN とし，CAN と PointRCNN と比較する．また，これらによるアテンションの違いを分析する．

3.2 定量的評価

従来手法と提案手法の比較結果を表 1 を示す．まず，表 1 の PointRCNN と CAN を比較すると，Cyclist (Easy) の 3d で約 0.5pt，Cyclist (Hard) の 3d で約 1pt 向上した．また，Pedestrian (Moderate) の Aos で約 6pt 向上した．次に，表 1 の PointRCNN と Single CAN を比較すると，Bev の Cyclist で約 1pt，3d の Cyclist で約 2pt，Aos の Cyclist (Easy,Moderate) で約 1pt，Aos の Pedestrian (Easy，Moderate) で約 5pt 向上した．CAN と Single CAN を比較すると，Single CAN が全体の精度が良い．これ異なるグルーピングを考慮しない方が精度面では良いことが分かった．

表 1：提案手法と従来手法の精度比較

	methods	Car			Cyclist			Pedestrian		
		Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Bev	PointRCNN	**90.12**	**87.59**	86.23	86.99	72.88	66.80	**68.09**	**61.48**	54.26
	CAN	89.58	80.09	79.71	**88.44**	69.14	67.86	65.41	57.22	**55.62**
	Single CAN	89.13	86.34	**86.52**	87.86	**74.06**	**67.97**	66.19	59.43	52.56
3d	PointRCNN	**88.69**	**78.59**	**77.82**	86.31	70.69	65.28	**65.01**	**57.69**	**51.87**
	CAN	87.63	77.37	76.92	87.19	68.14	66.85	63.13	55.54	47.72
	Single CAN	86.87	77.04	75.70	**87.83**	**73.55**	**67.31**	60.33	56.52	50.03
Aos	PointRCNN	**90.73**	**89.54**	**89.05**	88.89	76.21	**73.79**	67.50	61.28	**73.79**
	CAN	90.56	88.96	80.77	89.37	70.90	70.04	**73.56**	65.16	57.30
	Single CAN	90.59	89.09	88.33	**89.68**	**77.34**	70.30	70.19	**66.34**	59.52

3.3 各点群の重要度の可視化

CAN で得た各点の重みをアテンションとして可視化した画像を図 3 に示す．各点の色はアテンションの大小を表し，青から赤にかけて注目度が大きくなる．また，矢印は物体の座標系を表す．図 3(b) の CAN のアテンションから Car は側面の後方部分，Cyclist は自転車のサドルとカゴに注視している．しかし，Single CAN の場合は赤い点ががなく，全体を平均的に注視している．これらの結果より，CAN の方が，物体検出に重要な箇所を注視していることがわかった．

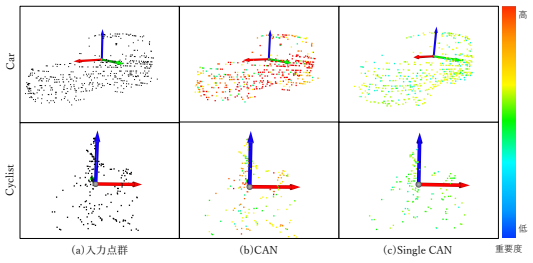


図 3：各点群の重要度の可視化

4. おわりに

本研究では点群の重要度を算出するために，PointRCNN に CAN，Single CAN を導入したネットワークを提案した．CAN で求めた各点の重要度から物体検出時に重要となる箇所が分かった．しかしながら，検出精度は CAN よりも Single CAN の方が高かった．今後は，異なるアテンション機構の導入による精度検証や傾向を調査する．

参考文献

[1] S.shi *et al.* “pointrcnn: 3d object proposal generation and detection from point cloud”. In *CVPR*, 2019.

[2] W.Zhang *et al.* “pcan: 3d attention map learning using contextual information for point cloud based retrieval”. In *CVPR*, 2019.

点群データに対するセマンティックセグメンテーションにおける CG データによる事前学習の有効性検証

EP19074 中村 智也　　指導教授：山下 隆義

1. はじめに

セマンティックセグメンテーションは，画像内の全画素や点群の各点に対してラベルやカテゴリを関連付けるタスクである．学習には大量の実環境データが必要であり，不十分な場合は精度の低下を招く．そこで，CG により大量のデータを生成して学習に用いることで，実環境データを補填することが期待される．本研究では，点群データに対するセマンティックセグメンテーションにおいて，CG データによる事前学習が有効であるか検証する．

2. セマンティックセグメンテーション

点群データに対するセマンティックセグメンテーションの代表的な手法である SalsaNext[1] は，3 次元点群を擬似画像へ変換し処理を行う．SalsaNext は Context Module, Pixel-Shuffle Layer, Dilated Convolution を導入し，識別精度の向上とリアルタイム性を実現している．SalsaNext のネットワーク構造を図 1 に示す．

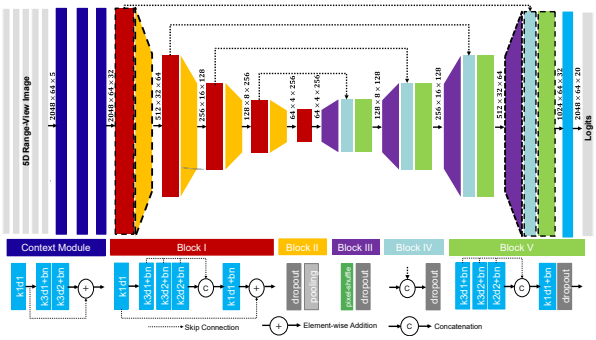


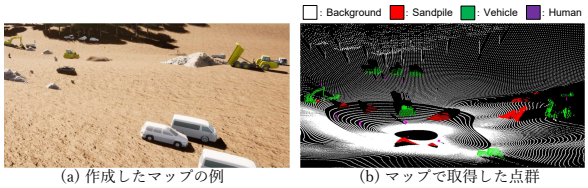
図 1：SalsaNext のネットワーク構造

3. 提案手法

本研究では，CG データセットを用いて事前学習モデルを作成し，実環境データセットでファインチューニングすることで事前学習の有効性を検証する．

3.1 CARLA を用いた CG データセット作成

セマンティックセグメンテーションの学習において，データセットが小規模であると，過学習などにより汎化能力が向上しない．そこで，実環境データセットに類似したデータを CG で作成する．CG データセットを用いて事前学習することで汎用的なモデルを獲得する．そして，実環境データセットでファインチューニングすることで実環境のドメインに適応させる．本研究では，CARLA を用いて工事現場を模したマップを作成する．作成したマップ例を図 2(a) に示す．マップ上では，ランダムに盛り土や建機，人を配置して点群を取得する．図 2(a) のマップで取得した点群データを図 2(b) に示す．生成したマップ数は 7，フレーム数は 3458 である．クラスは Background，Sandpile，Vehicle，Human である．



(a) 作成したマップの例　(b) マップで取得した点群

図 2：取得した点群データの例

3.2 事前学習

事前学習モデルとは，大規模なデータセットを用いて学習したモデルである．本実験では，CG データセットを用いて学習を行い，事前学習モデルを作成する．

3.3 実環境データでの学習

CG データセットを用いて事前学習モデルを作成した後，実環境データセットを用いてファインチューニングを行い，実環境データに対する精度の向上を図る．実環境データは，工事現場で取得した点群データである．実環境データセット内のフレーム数は 383 である．

4. 評価実験

本実験では，事前学習モデルの有無による精度を比較する．

4.1 実験概要

事前学習モデルの学習には CG データセット内の 5 シーン (2568 フレーム) を用いる．学習回数は 100 エポックである．学習時のネットワークに SalsaNext，最適化手法に Adam，初期学習率は 1.0×10^{-4} とする．事前学習モデルの評価には 2 シーン (890 フレーム) を用いる．ファインチューニングには実環境データセット内の 350 フレームを用いる．学習回数は 300 エポックである．学習時のネットワークに SalsaNext，最適化手法に Adam，初期学習率は 1.0×10^{-4} とする．評価には残りの 33 フレームを用いる．評価指標には IoU を用いる．

4.2 実験結果

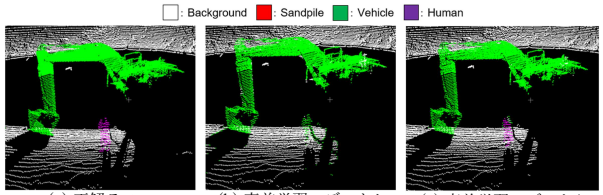
事前学習モデルを CG データで評価した結果を表 1 に示す．表 1 の Vehicle は，他クラスよりも精度が低い結果である．事前学習モデルの有無による評価結果を表 2 に示す．表 2 より事前学習モデルを使用することで Background は 0.4pt，Sandpile は 4.4pt，Vehicle は 1.6pt，Human は 24.6pt，Average は 7.8pt 精度が向上した．事前学習モデルの有無による検出結果を図 3 に示す．図 3 より，事前学習モデルを使用することで，Human の認識率が向上した．

表 1：事前学習モデルの定量的評価

Background	Sandpile	Vehicle	Human	Average
96.7	59.5	31.8	61.4	62.3

表 2：事前学習モデルの有無による定量的評価の比較

事前学習	Background	Sandpile	Vehicle	Human	Average
	93.9	52.1	34.3	0.0	45.0
✓	**94.3**	**56.5**	**35.9**	**24.6**	**52.8**



(a) 正解ラベル　(b) 事前学習モデルなし　(c) 事前学習モデルあり

図 3：事前学習モデルの有無による検出結果の比較

5. おわりに

本稿では，点群データに対するセマンティックセグメンテーションにおける CG データによる事前学習の有効性を検証した．評価実験により，CG データを用いた事前学習は有効であることを確認した．今後は，クラス間の点群数の不均衡問題について改善する方法を検討する．

参考文献

[1] Tiago Cortinhal, et al, “SalsaNext: Fast, Uncertainty-aware Semantic Segmentation of LiDAR Point Clouds for Autonomous Driving”, ISVC2020

Single-cell RNA-seq 解析における Star clustering と GMVAE を用いたデータのクラスタ数の探索

EP19011 伊藤啓太　　指導教授：山下隆義

1. はじめに

病気や障害の原因は，異常な細胞の遺伝子情報をタンパク質として発現していることである．病気や障害の原因究明には，タンパク質を発現している異常な細胞を特定する必要がある．細胞の解析手法には，深層生成モデルの Gaussian-mixture Variational AutoEncoder (GMVAE) [1] を用いた Single-cell RNA-seq 解析手法がある．しかし，遺伝子発現量データは細胞の種類が不明である．そのため，GMVAE に真のクラスタ数を入力することが不可能であり，適切な潜在空間の獲得が困難である．そこで，本研究では，Star clustering によりクラスタ数を予測し，GMVAE に入力する．これにより，GMVAE を用いて遺伝子発現量データの最適な潜在空間を獲得できる．

2. 先行研究

Single-cell RNA-seq 解析とは，1 つの細胞に含まれる各遺伝子の発現量を解析する手法である．Grønbech ら [2] は，生の遺伝子発現量データを GMVAE で処理することで，各細胞の潜在的な表現を獲得可能であることを示した．GMVAE には，クラス分類ネットワークが含まれ，データの真のクラスタ数を入力する必要がある．しかし，データにおける真のクラスタ数は不明なため，最適な潜在空間の獲得が困難となる場合がある．

3. 提案手法

本研究では，真のクラスタ数が不明なデータに対して最適なクラスタ数を探索し，GMVAE に用いる手法を提案する．提案手法によるクラスタ数の探索手順を以下に示す．

1. Star clustering を用いてデータのクラスタ数を予測する
2. 予測したクラスタ数を図 1 に示す GMVAE に入力する
3. GMVAE を学習し，データのクラスタ数を予測する
4. GMVAE に入力したクラスタ数と GMVAE で予測したクラスタ数が一致する場合，GMVAE に入力したクラスタ数を最適なクラスタ数とする
5. GMVAE に入力したクラスタ数と GMVAE で予測したクラスタ数が不一致な場合，GMVAE で予測したクラスタ数を GMVAE に入力する．そして，3 から 5 の手順を最適なクラスタ数を予測するまで繰り返す

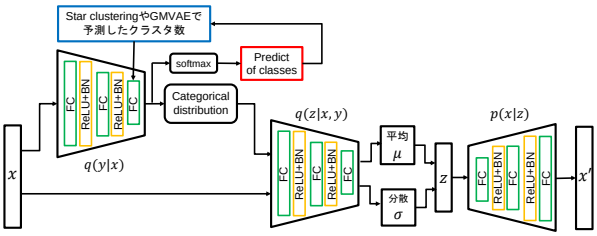


図 1：GMVAE のモデル図

提案手法で探索した最適なクラスタ数による GMVAE を用いることで，データの最適な潜在空間が獲得可能となる．

4. 評価実験

本実験では，提案手法を用いて真のクラスタ数が不明なデータの最適なクラスタ数を探索可能かを評価する．

4.1. 実験概要

本実験は，末梢血単核細胞データ（クラスタ数 9）を対象とする．GMVAE の尤度関数は，負の 2 項分布を使用する．そして，200 エポックの Warm up を実施し，500 エポック学習させる．評価には，Accuracy，Normalization Mutual Information score(NMI) と Entropy を使用する．NMI はクラスタリングの性能を評価し，値は 1 に近いほどクラスタリング性能が高いことを示す．Entropy は予測クラスタに属するデータの真のラベルの偏り度合いを評価し，値が 0 に近いほどクラスタリングができていることを示す．

4.2. 実験結果

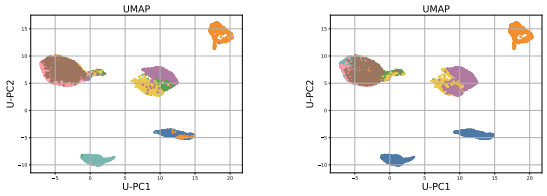
従来手法との比較結果を表 1 に示す．従来手法は真のクラスタ数である 9 を入力とした時の評価結果である．クラスタ数 10 は，提案手法により求めた最適なクラスタ数である 10 を入力とした時の評価結果である．表 1 から，クラスタ数 10 の Accuracy と NMI がクラスタ数 9 よりも高く，クラスタ数 10 の Entropy がクラスタ数 9 よりも低いことが分かる．これらから，提案手法で求めたクラスタ数である 10 が GMVAE で予測する最適なクラスタ数であると考える．

表 1：評価結果

クラスタ数	9（従来手法）	10（提案手法）
Accuracy	0.537	0.661
NMI	0.607	0.683
Entropy	1.150	1.076

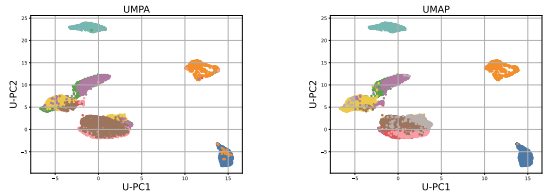
4.3. ラベル分布の比較

従来手法および提案手法の真のラベル分布と予測ラベル分布を図 2，図 3 に示す．提案手法は，青色と水色のクラスタを個々のクラスタと予測したが，従来手法は 1 つの青色のクラスタと予測した．また，図 3 から提案手法は，黄色のクラスタを黄色と灰色の 2 つのクラスタに分離した．これにより，クラスタ数を 1 つ多く予測したと考える．これは，黄色の細胞種が 2 つの細胞種の特徴を持っているためである．これらから，提案手法はデータに即した予測結果であるといえる．



(a) 真のラベル　　(b) 予測ラベル

図 2：従来手法



(a) 真のラベル　　(b) 予測ラベル

図 3：提案手法

5. おわりに

本研究では，GMVAE を用いたクラスタ数の探索手法により，GMVAE のクラスタリング性能が向上し，データに即した予測が可能であることと，予測クラスタ数を真のクラスタ数に近づけることが可能であることを示した．今後は，疾患のあるデータと疾患のないデータを用いて潜在空間を比較することで，疾患原因の細胞種の特定をしていく．

参考文献

[1] N. Dilokthanakul, *et al.*, “Deep unsupervised clustering with gaussian mixture variational autoencoders”, arXiv, 2016.

[2] C. Grønbech, *et al.*, “scVAE: variational auto-encoders for single-cell gene expression data”, Bioinformatics, 2020.