1T-08

楽曲ジャンル分類へのEfficientNetV2の適用

坂田　大地†　　小嶋　和徳†　　伊藤　慶明†
岩手県立大学†

1. はじめに

近年ディジタル音楽が普及し，音楽情報検索に関する研究，特に楽曲特徴から自動で楽曲のジャンル分類を行う楽曲ジャンル分類の研究が盛んに行われている．

近年，人工知能分野において深層学習の技術が発展し，様々な深層学習モデルを用いた自動楽曲分類が行われている．自動楽曲ジャンル分類では Convolutional Neural Networks (CNN)が用いられることが多いが，本研究では Google 社が 2021 年に発表した画像認識用の深層学習モデルである EfficientNetV2[1]を楽曲ジャンル分類に適用する方法を提案する．楽曲をスペクトログラム画像とし，EfficientNetV2 の学習および自動分類を行う際，高い分類精度を得るためのスペクトログラム構成方法の検討を行う．

2. 提案手法

本章では提案手法である楽曲ジャンル分類へのEfficientNetV2 の適用方法を説明した後，高い分類精度を得るためのスペクトログラム構成方法について説明する．

2.1. EfficientNetV2 の楽曲分類への適用

EfficientNetV2 とは 2021 年の ICML で発表されたモデルで，ImageNet ILSVRC2012 で 87.3％の Top-1 Accuracy を実現した．

本稿では EfficientNetV2 を楽曲ジャンル分類へ適用する．入力は分類窓長毎にスペクトログラム画像を求め，出力はジャンルの種類数(ジャンル数)の事後確率ベクトルとし，事後確率ベクトルの中で最も高い確率のジャンルをその分類窓の結果とする．本システムでは入出力数が大規模にならないため，EfficientNetV2-S モデルを用いて学習及び分類を行う．EfficientNetV2 は画像認識に使用されるモデルであり，入力は縦画素数×横画素数×チャンネル数(次元)で与えられるため，入力次元数は 224 (縦画素数)×224 (縦画素数)×3 (RGB)の 178,608 次元とした．ジャンル数を 10 としたため，出力は 10 次元のジャンルの事後確率ベクトルとなる．学習には教師データとして，正解ジャンルを 1 とし他を 0 とした One-hot ベクトルを使用した．文献[2]の結果と比較するため画像サイズは 224×224 固定とし，データ数や画像サイズの差異による検証は困難なため，Progressive Learning[1]は使用しない．

TABLE 1 スペクトログラム生成固定条件

変換手法	STFT
サンプリング	22050 Hz (Monaural)
窓関数	ハニング窓
チャンネル数	RGB (3) PNG 画像
出力サイズ	224×224 px

TABLE 2 分析手法

	条件 A	条件 B	条件 C
分析窓長	2048	2048	446
分析窓シフト幅	512	296	296
次元数	130×1025	224×1025	224×224

2.2. スペクトログラムの構成

本稿では縦軸を周波数，横軸を時間，信号の強弱としたスペクトログラム画像を用いる．本研究では，分類を行う窓長を 3.0 sec，分類窓のシフト幅(分類窓シフト)を 1.5 sec とした場合[2]と，データ数を増やすため分類窓シフトを 1.0 sec とした場合のスペクトログラムを作成し，分類精度を評価する．

スペクトログラムの周波数軸のスケールは楽曲解析などに利用されている線形と対数で分類精度の評価を行う．スペクトログラム画像は音声楽曲分析ライブラリの librosa[3]を使用して TABLE 1 に共通の生成条件，周波数分析する際の 3 つの条件を TABLE 2 に示す．A は多くの楽曲解析に使用される条件であり，スペクトログラムを生成した後，画像化の際にリサイズを行ったものである．B は時間軸のみ 224 pixel (px)であるが，周波数軸は A と同じ次元数で生成した後，画像化の際にリサイズを行った．C は多くの画像認識モデルにて利用される 224×224 px になるように生成したものであり，画像化の際にリサイズは行わない．

分類窓シフトを 1.5sec，1.0 sec とした場合，スケールを線形と対数とした場合，3 つの条件，合

An Application of EfficientNetV2 on Music Genre Classification
†Daichi Sakata，Kazunori Kojima，Yoshiaki Itoh・Iwate Prefectural University

計 12 パターンのスペクトログラム画像を生成し，分類精度で評価を行う．

3. 実験結果

3.1. データセット

データセットは楽曲ジャンル分類において多く利用されている，GTZAN を使用した．GTZAN では 1 曲あたり 30 秒，合計 1000 曲で構成される音楽情報検索用のデータセットであり，10 ジャンル 100 曲ずつで構成される．

3.2. 実験条件

分類窓シフト 1.5 sec の場合，1 曲につき 19 枚のスペクトログラム画像が得られるため，全体で 19,000 枚得られ，そのうち 80 %の 15,200 枚，1 ジャンルにつき 1,520 枚を学習データとして使用し，残りの 3,800 枚，1 ジャンルにつき 380 枚をテストデータとして使用した．分類窓シフト 1.0 sec の場合，スペクトログラム画像は 28,000 枚得られ，そのうち 80 %の 22,400 枚，1 ジャンルにつき 2,240 枚を学習データとして使用し，残りの 5,600 枚，1 ジャンルにつき 560 枚をテストデータとして使用した．スペクトログラム画像 1 枚毎に判定を行い，テストデータ全体の正解率で分類精度を評価した．なお，学習及び評価には 5 分割の交差検証（5-Fold Cross Validation)を使用し，5 つの平均正解率を分類精度とした．ただし，スペクトログラム画像はそれぞれのジャンルが同量選択され，入力形式に差異が出ないよう，同じシード値で実行した．実装には Tensorflow[4] を利用し，学習時の Graphics Processing Unit (GPU)には NVIDIA 社の GeForce RTX 2080 を使用した．

3.3. 実験結果と考察

提案手法を用いて初期から学習を行い，評価実験を行った．分類窓シフトを 1.5 sec，1.0 sec とした場合の結果を Fig. 1，Fig. 2 にそれぞれ示す．まず，周波数のスケールについて，分類精度はいずれも線形よりも対数の方が高くなった．これは対数の方が線形よりも高周波成分を表現できているためと考える．次に分類窓シフトは 1.0 sec の方が高くなった．分類窓シフトを小さくすることで，学習のデータ数が増えたためと考える．条件 A，B，C について比較する．条件 B は条件 A より分類精度が高く，これは分析窓シフトが小さいため，より詳細なスペクトログラムが生成できたためと考える．条件 C が最も分類精度が高くなった．これは周波数軸をリサイズなどで圧縮することなく表現できたためと考える．

本手法での最良の結果は対数スケールの分類窓

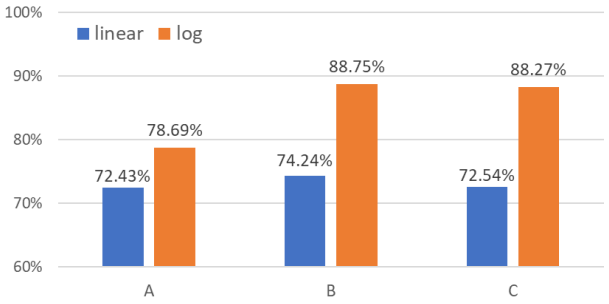

Fig. 1 分類窓シフト 1.5 sec 結果

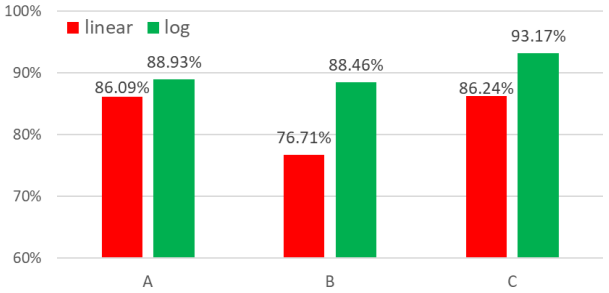

Fig. 2 分類窓シフト 1.0 sec 結果

シフト 1.0 sec の条件 C の 93.17%であった．最新の研究では SE-Block を用いた手法[2]の精度が 92.02%であったため，同等以上の結果を得ることができた．今後は分類窓シフトや分析窓長，分析窓シフト等を更に短くした場合について評価実験を行う予定である．また，今回は用いなかった Progressive Learning や転移学習等を用いることで更なる認識精度の向上を目指したい．

4. まとめ

本稿では EfficientNetV2 を使用した楽曲ジャンル分類手法を提案し，SE-CNN と比較し同等以上の精度を得ることができ，提案手法の楽曲ジャンル分類における有効性を確認できた．今後は分類窓シフトや分析窓長，分析窓シフト等を短くした場合，Progressive Learning や転移学習などの実装も試みたい．

謝辞

本研究の一部は JSPS 科研費 21K12611 の助成を受けて実施した．

参考文献

[1] Mingxing Tan, et al., "EfficientNetV2: Smaller Models and Faster Training," 38th ICML, PMLR 139:10096-10106, 2021.
[2] Yijie Xu, et al., ”A deep music genres classification model based on CNN with Squeeze & Excitation Block,” APSIPA Annual Summit and Conference, Dec., 2020
[3] McFee Brian *et al.*, “librosa: Audio and music signal analysis in python.” In Proceedings of the 14th python in science conference, pp. 18-25. 2015.
[4] Martín Abadi *et al.*, TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, 2015, http://download.tensorflow.org/paper/whitepaper2015.pdf