CNN Autoencoder から抽出したボトルネック特徴量を用いた環境音分類

松原 拓未[1] 原 直[1] 阿部 匡伸[1]

概要：環境音分類とは，環境音から音源や収録場所の分類をおこなうことである．環境音分類の研究の多くは，同一の状況で収録されたデータセットを用いておこなわれている．そのため，収録機器や収録条件が学習データと異なった環境音を分類した場合に同等の性能が得られるとは限らない．本報告では，収録機器や収録条件などの収録環境によらない環境音分類を目的として，CNN Autoencoder から抽出したボトルネック特徴量を用いた環境音分類を提案する．評価実験として，2 種類の異なった環境で収録されたデータセットを用いて，提案方式の分類性能を評価した．実験の結果，CNN Autoencoder を用いて抽出したボトルネック特徴量には，収録環境によらず分類をおこなうために有用な情報が含まれていると考えられる．

Environmental Sound Classification using Bottleneck Feature extracted by CNN Autoencoder

TAKUMI MATSUBARA[1] SUNAO HARA[1] MASANOBU ABE[1]

1. はじめに

人間は，車の走行音や人の話し声，鳥の鳴き声などの様々な種類の音に囲まれて生活している．これらの音のことを環境音と呼び，環境音から音源の種類や音の鳴っている時間を検出することは，人間が置かれた状況を理解する上で重要である [1]．環境音の活用例としては，公共施設や観光地における傷害などの犯罪検知や自動車による事故検知，高齢者宅の生活音を利用した緩やかな見守り [2]，聴覚障害者への視覚外にある危険通知，などが考えられる．

環境音から情報を取り出すタスクの 1 つとして環境音分類がある．環境音分類の分野では機械学習を用いたアプローチがおこなわれ，Detection and Classification of Acoustic Scenes and Events (DCASE)[3][4] という国際的ワークショップが 毎年開催されるなど盛んに研究がおこなわれている．機械学習を用いたアプローチの中でも Neural Network (NN) の仕組みを用いた研究がなされており，再帰的な構造を持った Recurrent Neural Network (RNN) を用いた手法 [5] や Convolutional Neural Network (CNN) を用いた手法 [6] などが提案されている．しかしながら，これらの研究は同一の状況で収録がされたデータセットを用いて行われており，収録機器や収録条件などが異なったデータを用いた場合の性能については，評価されていない．

NN の研究分野にはボトルネック特徴量を用いる研究がある．ボトルネック特徴量は学習のおこなわれた NN の中間層から取り出される特徴量のことである，ボトルネック特徴量とは，入力を圧縮したのちに再び入力を復元できるように学習した NN (Autoenocder) の中間層から取り出される特徴量であり，入力から重要な情報だけを取り出すことができる．ボトルネック特徴量を用いた研究は，性別に依存しない話者識別 [7] や雑音に対して頑健な話者識別 [8] などがある．これらの研究は性別や雑音に対して汎化性能のあるモデルを作成するためにボトルネック特徴量を用いている．

本報告では，収録環境によらない環境音分類のための識別器の作成を目標として，CNN を用いて構成した Autoencoder (CNN Autoencoder) でボトルネック特徴量を抽出し環境音分類をおこなう．提案方式では，複数の収録環境のデータセットを用いて学習をおこなった CNN Autoencoder からボトルネック特徴量を抽出する．抽出されたボトルネック特徴量には，どの収録環境のスペクトロ

[1] 岡山大学 大学院ヘルスシステム統合科学研究科

© 2019 Information Processing Society of Japan



図 1　環境音分類の概要

グラムでも復元をおこなうことのできる情報が含まれている．このことから，ボトルネック特徴量を用いて分類をおこなうことで，収録環境によらず入力スペクトログラムにとっての重要な情報のみで分類を行うことができると考えられる．本実験では，収録環境の異なったデータセットの分類における提案方式の評価をおこなうために，収録環境の異なった 2 種類のデータセットを用いて学習をおこなった CNN Autoencoder から抽出したボトルネック特徴量を用いて分類をおこなっていく．

2.　環境音分類タスク

本章では，環境音分類タスクについて使用する特徴量や分類を評価する指標について述べる．また，本実験で分類をおこなう環境音データセットについても述べる．

2.1　環境音分類

環境音から音源の分類をおこなうことについての概要を**図 1** に示す．例としては，ある収録音に車の走行音と音響信号機の音が含まれている場合，収録音から抽出した音響特徴量を識別器に入力して，結果として車の走行音と信号の音が含まれていることを出力することである．

2.2　環境音分類に利用する音響特徴量

本実験では，分類をおこなうために収録音から対数メルスペクトログラムと動的特徴量（Δ）を抽出する．対数メルスペクトログラムは対数変換をおこなったメルスペクトログラムのことである．メルスペクトログラムとは，人間の音高の知覚的尺度（メル尺度）によるフィルタ群 (メルフィルタバンク) が適用されたスペクトログラムのことである．スペクトログラムは音の波形を短時間フーリエ変換することで得られる特徴量のことであり，周波数成分が時系列順に並んだデータである．動的特徴量はスペクトログラムの時間帯ごとの差分であり，スペクトログラムの時間方向での変化量を示している．

環境音の波形を分析した結果の対数メルスペクトログラムの例を**図 2** に示す．図の縦軸がメルフィルタバンクの次元数で，横軸がフレーム単位の時間を表しており，各座標での色が時間帯ごとの周波数成分の強さを表している．図 2 には主に音響信号の音と車の走行音が含まれており，音響信号機の音が鳴っている時間では，緑の四角で囲んだ部分ように右肩下がりの模様が出ている．また，車の走行音が鳴っている時間では，水色の四角で囲まれている次元数で高い値が出ている．このように音の種類によってスペクトログラムに現れる特徴が異なるため，スペクトログラムを用いて環境音分類が可能であると考えられる．

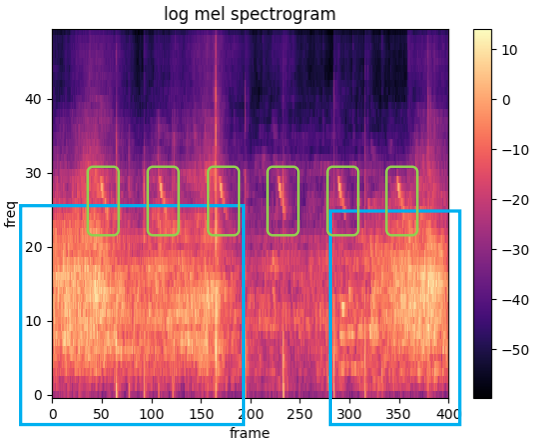


図 2　メルスペクトログラムの例（車と音響信号機）

2.3　F-score による評価

2.3.1　F-score

F-score(F) は予測結果についての評価尺度の 1 つであり，F 値，F-measure とも呼ばれる．F-score は適合率（Precision）と再現率 (Recall) の調和平均によって求められる．適合率は正と予測した場合にその予測が正しい割合のことを示し，再現率は真の結果が正であるものに対して正と予測できた割合を示す．F-score はこれらを総合的に評価することができる．それぞれの評価指標は以下の式で求められる．

$$\mathrm{F} = \frac{2\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{1}$$

$$\mathrm{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\mathrm{Recall} = \frac{TP}{TP + FN} \tag{3}$$

TP，FP，FN は **表 1** で示される関係である．

2.3.2　F-score による環境音分類の評価

環境音分類の結果を F-score を用いて評価するのは，適合率か再現率のどちらか一方だけで評価をおこなうことに問題があるからである．

適合率のみで評価をおこない良い結果を得た場合，予測の正確性は確かめられるがいくつのデータに対して予測ができているかを知ることができない．

また，再現率のみで評価をおこない良い結果を得た場合，多くのデータに対して予測が可能なことを示せるが，分類の予測が正しいものであるかを知ることができない．

人間が生活をおこなっている環境を音から知り活用をしていくためには，できるだけ多くの環境音に対して正確な分類結果を得る必要がある．このため環境音分類の結果は適合率と再現率を総合的に考慮できる F-score を用いて評

© 2019 Information Processing Society of Japan

表 1 予測結果と真の結果の関係

	真の結果が正	真の結果が負
予測結果が正	TP(True Positive)	FP(False Positive)
予測結果が負	FN(False Negative)	TN(True Negative)

表 2 データ数とラベルの数 (TUT データセット)

データ数	brakes squeaking	car	children	large vehicle	people speaking	people walking
547	53	312	60	116	147	209

価をおこなう必要がある.

2.4 使用データセットの概要

提案方式が収録環境によらない環境音分類をおこなうことに有用であることを示すために，2 種類の異なる収録環境で収録された環境音データセットを利用する．本章では使用する環境音データセットの概要を，第 2.4.1 項では TUT Sound Events 2017 detaset[9] (TUT データセット) について，第 2.4.2 項では筆者が所属している研究室で収録を行ったデータセット (OU データセット) についてそれぞれ述べる．第 2.4.3 項では 2 種類のデータセットの違いについて述べる.

2.4.1 TUT Sound Events 2017 detaset

TUT Sound Events 2017 detaset (TUT データセット) は DCASE2017 challenge[3] の Task3: Sound event detection in real life audio で使用されたデータセットである．収録は街中で行われており，イヤーマイクロホン（Soundman OKM II Klassik/studio A3）と wave recoder (Roland Edirol R-09) を利用して収録がおこなわれている．サンプリングレートは 44.1 kHz で量子化ビット数は 24 bit である.

このデータセットには，3-5 分の環境音データが 24 個用意されており，それぞれのデータに対して 6 種類のラベルが音の鳴っている時間区間ごとに付けられている．6 種類のラベルは「brakes squeaking」,「car」,「children」,「large vehicle」,「people speaking」,「people walking」であり，同じ時間に複数種類のラベルが重複して存在している場合がある．本報告では，10 秒間ごとに環境音を区切り，その区間に付与されているラベルの分類をおこなう．10 秒ごとのデータ数と各ラベルの数を**表 2** に示す．全体のデータ数よりもラベルの総数が多くなっているのは，1 つのデータに複数のラベルが付与されている場合があるからである．また，このデータセットの音声はステレオ形式で収録が行われているためモノラル形式に変換をして使用している.

2.4.2 OU データセット

筆者が所属している研究室で収録をおこなった環境音データセット [10]（OU データセット）は，タブレット端末（Google Nexus 7）を使用して収録がおこなわれた．また，本実験では OU データセット内の岡山大学周辺と岡山駅周辺で収録された音を実験に用いた．収録をおこなう際には Android 用環境音収録アプリケーションであるオトログマッパー [11] を利用した．このアプリケーションでは，10 秒間の音を 1 つの wav ファイルとして記録することができ，同時に収録端末を所持している人が音を収録する際に聞こえた音の種類を選択して記録することができる．サンプリングレートは 32kHz で量子化ビット数は 16 bit である.

表 3 データ数とラベルの数 (OU データセット)

データ数	車	音楽	音響信号機	人	バイク	足音
1563	880	438	362	566	535	350

本実験で用いるラベル情報は収録時に付与したものではなく，後日 2 名のラベル付与者によって付けられたラベルを利用する [12]．後日付けられたラベルを利用する理由は，収録者の耳には聞こえていてもデータには収録されていない音がある場合や収録者がラベル付与を忘れるなどの可能性があるためである．OU データセット内で本実験で使用するデータ数と各ラベルの数を**表 3** に示す．「人」のラベルに関しては人間の声が聞こえている部分に対して付与されている．TUT データセットと同様にラベルの総数が多くなっているのは，1 つのデータに複数のラベルが付与されている場合があるためである.

2.4.3 データセット間の違い

TUT データセットからメルスペクトログラムの例を**図 3** に示す．このデータには「children」のラベルが付与されている．TUT データセットでは，収録専用の機材を用いて収録された音であるので収録された音には機材の操作によるノイズは含まれていない．また，付与されたラベル以外の音はほぼ含まれていない.

OU データセットから抽出したメルスペクトログラムの例を**図 4** に示す．このデータには「音響信号機」のラベルが付与されている．OU データセットでは，タブレット端末を用いて収録がされているため，マイクと操作をおこなう画面が近く，収録時にラベルを記録するために端末の画面をタップする音などのノイズが含まれているものが多い．タップ音の例としては，図 4 では緑の四角で囲んでいる全周波数帯にかけて線のような模様が表れている部分である．また，今回用いたラベル以外の音が含まれている場合がある．例としてはボールのはねる音や金属がこすれる音などがある.

これらのことから，TUT データセットと比べて OU データセットを分類することは難しいと考えられる.

3. 環境音分類のための識別器

本報告では，CNN Autoencoder 抽出したボトルネック特徴量を用いた環境音分類を提案する.

第 3.2 項で CNN について，第 3.3 項で CNN Autoen-

© 2019 Information Processing Society of Japan

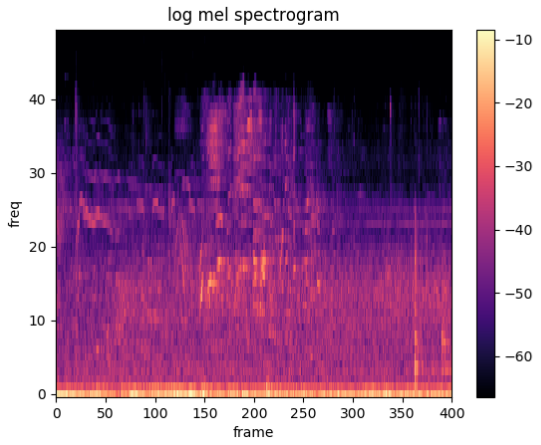


図 3 TUT データのメルスペクトログラムの例（children）

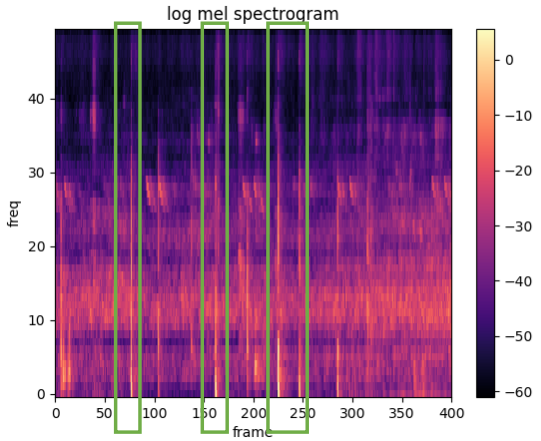


図 4 OU データのメルスペクトログラムの例（信号）

coder から抽出するボトルネック特徴量についてそれぞれ述べる．また，本実験の評価手法である k-fold cross validation について第 3.4 項で述べる．

3.1 識別器の概要

提案方式の概要図を**図 5** に示す．提案方式は，まず CNN Autoencoder の学習をおこなう．次に，作成した CNN Autoencoder を用いて，環境音のスペクトログラムからボトルネック特徴量を抽出する．その後，ボトルネック特徴量を用いて分類をおこなう．

CNN Autoencoder の学習部では，収録した環境音の分析から得たスペクトログラムを入力特徴量として出力が入力特徴量と同じになるように CNN の学習をおこなう．学習をおこなう CNN は内部で入力を 1 度圧縮したのちに復元をおこなう構造にする．

分類器では，まず環境音のスペクトログラムを CNN Autoencoder に入力しボトルネック特徴量の抽出をおこなう．その後取り出されたボトルネック特徴量を 2 層の NN を用いて分類の学習をおこなう．

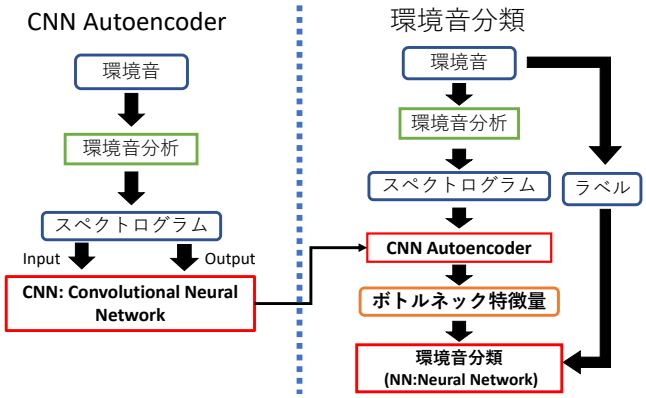


図 5 提案方式の概要図

3.2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) とは畳み込み層とプーリング層を持つ NN のことである．畳み込み（convolution）層では，入力に対してフィルタによる重みつき和を一定の幅で動かしながらおこなう．また，プーリング層では入力の一定範囲ごとに代表値が取り出される．プーリングは，一定範囲で最も大きな値を取り出す最大プーリング (max pooling) を用いる．

CNN による環境音分類では，入力スペクトログラムに畳み込みとプーリングをおこない，畳み込まれた特徴量を用いて分類をおこなう．CNN の構造を用いることによって，第 2.2 節で述べた周波数方向と時間方向で現れる特徴を畳み込みの処理によって考慮することができる．

3.3 CNN Autoencoder から抽出するボトルネック特徴量

CNN Autoencoder とは，CNN の構造を用いて学習がおこなわれた Autoencoder のことである．CNN Autoencoder の概要図を**図 8** に示す．Autoencoder とは入力と同じ値を出力するように学習された NN のことであり，ある中間層をほかの中間層よりも次元数を少なくした構造（ボトルネック構造）で学習される．Autencoder に入力をおこなった際にボトルネック構造の層から値を抽出することで，入力を復元するために必要な情報，つまり入力についての重要な情報を保持したまま次元数が削減された情報を取り出すことができる．この取り出された値のことをボトルネック特徴量と表す．

CNN の構造を用いて Autoencoder の学習をおこなうことによって 2 次元の特徴量からボトルネック特徴量を取り出すことができる．本報告では周波数方向と時間方向の 2 次元の特徴量であるスペクトログラムからボトルネック特徴量を取り出すために CNN Autoencoder を利用する．

CNN の構造である畳み込みとプーリングを繰り返すと特徴量が圧縮されていくが，Autoencoder を学習するためには圧縮された特徴量を拡大することで復元する必要がある．そのため「deconvolution」と 「up sanpling」とい

© 2019 Information Processing Society of Japan

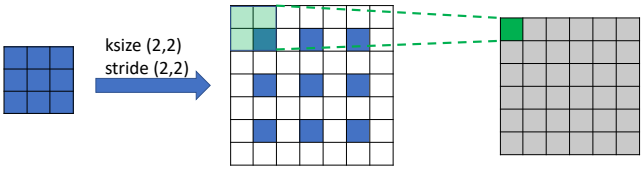


図 6　deconvolution の例

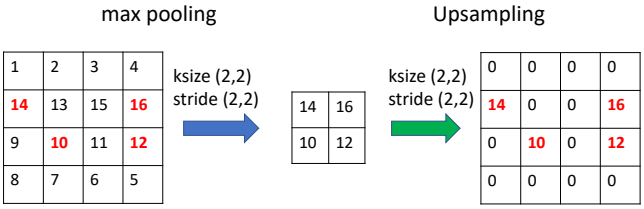


図 7　max pooling と up sampling の関係

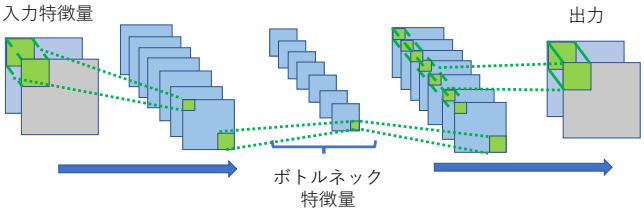


図 8　CNN Autoencoder とボトルネック特徴量

う手法を用いる．deconvolution の例を**図 6**，max pooling と up sampling の関係の例を**図 7** にそれぞれ示す．deconvolution は入力の各値の周囲を一定範囲で 0 埋めして拡張した特徴量を畳み込みをおこなうこと手法であり，up sampling は最大プーリングされた値とプーリングされる前の特徴量での座標を利用して特徴量の拡張をおこなう手法である．

3.4　k-fold cross validation

本報告の実験では，2 種類のデータセットを用いて提案方式の検討をおこなう．用意したデータセットを用いて分類の学習をおこなう場合，学習データと評価データに分ける必要がある．データを分ける際に学習データと評価データで各ラベルのデータ数に偏りが発生した場合，正しく学習や評価をおこなうことができない可能性がある．データの偏りにより正しい評価が行えなくなることを避けるために k-fold cross validation （k-分割交差検証）をおこなう．k- fold cross validation は以下の手順でおこなわれる．

(1) 使用するデータセットを均等に k 個の部分集合に分割する
(2) 分割したうちの 1 つを評価データ，残りを学習データとして学習，評価をおこなう
(3) すべての部分集合が評価データとして選ばれるように 2 の手順を k 回おこなう
(4) k 回の結果の平均を全体の結果として扱う

k-fold cross validation により，使用するデータセットを最大限に活用して手法の評価をおこなうことができる．

表 4　CNN Autoencoder の各層の条件

Shared Layer			
layer type	input size	ksize	stride
$L1$ convolution	$400 \times 50 \times 2$	(7,7)	(3,1)
$L2$ max pooling	$132 \times 44 \times 64$	(2,2)	(2,2)
$L3$ convolution	$66 \times 22 \times 64$	(5,5)	(2,1)
$L4$ max pooling	$31 \times 18 \times 128$	(2,2)	(2,2)
$L5$ convolution	$15 \times 9 \times 128$	(3,3)	(2,1)
$L6$ max pooling	$7 \times 7 \times 256$	(2,2)	(2,2)
$L7$ up sampling	$3 \times 3 \times 256$	(2,2)	(2,2)
$L8$ deconvolution	$7 \times 7 \times 256$	(3,3)	(2,1)
$L9$ up sampling	$15 \times 9 \times 128$	(2,2)	(2,2)
$L10$ deconvolution	$31 \times 18 \times 128$	(5,5)	(2,1)
$L11$ up sampling	$66 \times 22 \times 64$	(2,2)	(2,2)
$L12$ deconvolution	$132 \times 44 \times 64$	(7,7)	(3,1)

表 5　実験条件

特徴量抽出	
特徴量	対数メルスペクトログラム，動的特徴量（Δ）
メルフィルタバンクの次元数	50
フレーム長	40 ms
フレームシフト	20 ms
Δ 特徴量推定幅	前後 4 フレーム
CNN Autoencoder 学習条件	
各 fold の最大 epoch 数	200
ミニバッチサイズ	20
損失関数	mean_square_error
活性化関数	Rectified Linear Function

4.　CNN Autoencoder に関する予備実験

本章では，ボトルネック特徴量を抽出するための CNN Autoencoder の学習条件と作成した CNN Autoencoder についての評価を行う．第 4.1 節では CNN Autoencoder の学習条件や使用する特徴量について述べる．第 4.2 節では作成した CNN Autoencoder が入力の復元をおこなうことができているかを評価する．

4.1　実験条件

本実験では，第 2.4 節 で述べた 2 種類のデータセットを用いて学習，評価をおこなう．CNN Autoencoder の学習をおこなうために使用したネットワーク構造を**表 4** に，その他の学習条件や使用する特徴量についてを**表 5** に示す．

本実験では，convolution(畳み込み) 層と max pooling(最大プーリング層) の組み合わせの 3 層で入力特徴量を圧縮し，その後の deconvolution 層と up sampling 層の組み合わせを上記と同じ 3 層で入力特徴量の復元するように学習をおこなう．各データセットの 3/4 を用いて学習をおこない残りの 1/4 を評価データとして学習をおこなった．

© 2019 Information Processing Society of Japan

4.2 CNN Autoencoder の評価実験

本実験では，第 2.4 節で述べた 2 種類のデータセットを用いて CNN Autoencoder の学習をおこない，学習に用いていないデータを入力として CNN Autoencoder の出力とメルスペクトログラムを比較する．また， CNN Autoencoder は各データセットごとに学習をおこなったモデル (AE_depend) と，2 種類のデータセットを用いて学習をおこなったモデル (AE_independ) の 2 種類の比較をおこなう．

各データセットごとの比較結果の一部を**図 9** と**図 10** に示す．それぞれの図は上から順に入力特徴量，AE_depend の出力，AE_independ の出力を表している．今回載せた図のように，単独のデータセットで学習をおこなった場合，混ぜて学習をおこなった場合の双方で元の入力特徴量をおおむね復元できていることが確認できた．このことから，今回作成した CNN Autoencoder から抽出したボトルネック特徴量には，入力特徴量を復元するのに重要な情報を保持していると考えられる．

5. 提案方式の分類性能の評価

本章では，CNN Autoencoder から得られるボトルネック特徴量を用いた環境音分類と従来手法である CNN を利用した環境音分類の比較実験をおこなう．第 5.1 節では本実験の条件について述べる．第 5.2 節では各手法での分類結果を示し，その結果について考察を述べる．

5.1 実験条件

本実験では，提案方式であるボトルネック特徴量を用いた環境音分類と従来手法の分類性能の比較をおこなう．従来手法では，ボトルネック特徴量を使わずに CNN で入力特徴量を圧縮する．なお，後段の全結合による分類部分は，提案方式と同等の構造を持ち，「データセット毎に学習する場合 (CNN_depend)」 と「データセット共通に学習する場合 (CNN_independ)」とがある．提案方式は，「AE_depend から抽出したボトルネック特徴量を用いて分類の学習をおこなう手法 (AE+NN_depend)」，「AE_independ から抽出をしたボトルネック特徴量を用いて分類の学習おこなう手法 (AE+NN_independ)」である．

本実験では，第 4 と同様に，第 2.4 節で述べた 2 つのデータセットの分類をおこなう．各データセットごとの分類するデータ数は TUT データセットが 547 個，OU データセットは 1563 個，ラベルの総数は，TUT データセットは 897 個，OU データセットは 3131 個である．

提案方式で用いる CNN Autoencoder は表 4 のネットワーク構成で学習をおこなう．CNN Autoencoder の中間層である 表 4 の $L6$ 層の出力をボトルネック特徴量として扱い，取り出したボトルネック特徴量を用いて各データセットごとにラベルへの分類の学習を全結合層 2 層でおこなう．分類の学習をおこなう全結合層 2 層の条件を**表 6** に示す．

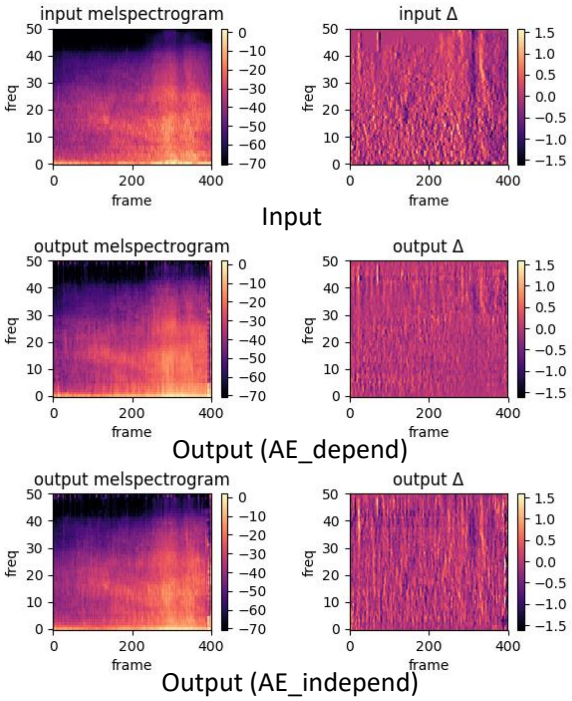


図 9 TUT データ: 対数メルスペクトログラム（左），Δ(右)

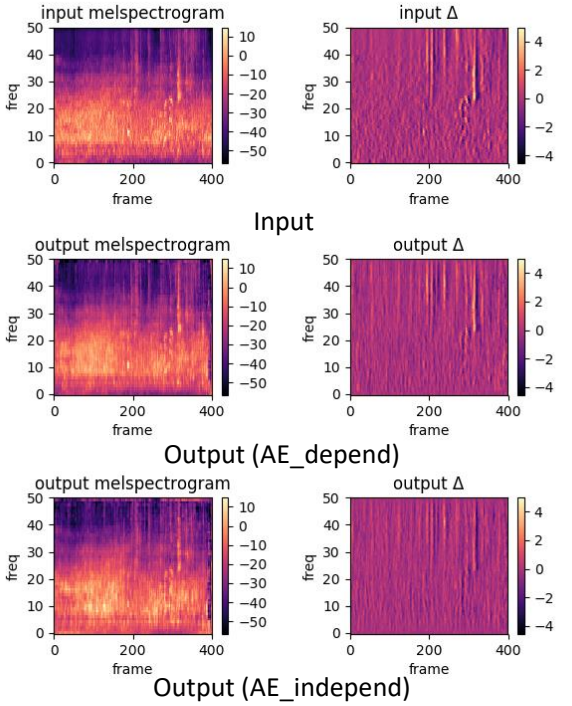


図 10 OU データ: 対数メルスペクトログラム（左），Δ(右)

表 6 全結合層の条件

layer type	input size	output size
$L1$ linear	2304	1024
$L2$ linear	1024	6

従来手法のネットワーク各層の条件は，入力の畳み込み

© 2019 Information Processing Society of Japan

表 7　実験条件

分類ネットワーク学習条件	
各 fold の最大 epoch 数	100
ミニバッチサイズ	20
損失関数	sigmoid_cross_entropy
活性化関数	Rectified Linear Function
結合重みの初期化	ガウス分布 (平均: 0, 標準偏差: 1)

表 8　各手法ごとの分類結果

F-score	TUT	OU
AE+NN_depend	**0.737**	**0.668**
AE+NN_independ	0.731	0.660
CNN_depend	0.715	0.665
CNN_independ	0.693	0.642

をおこなう部分は表 4 の $L1$ から $L6$ と同じ設定にし，畳み込んだ後の分類をおこなう全結合層の設定は表 6 と同じにする．また，CNN_independ については各データセットごとに付与されているラベルが異なるため畳み込み層をデータセット共通で学習し，全結合層はそれぞれのデータセットで学習をおこなった．評価は 4-fold cross validation を用いて行い，評価尺度は F-score (F) を用いる．

特徴量抽出とネットワークの学習条件を**表 7** に示す．評価に用いる CNN Autoencoder のモデルは各 fold ごとに評価データセットを入力した時に損失関数の結果が最も小さくなった epoch の時のパラメータを用いる．分類器のモデルは F-score が最も大きな値になった epoch の時のパラメータを用いる．

5.2　実験結果

各手法ごとの分類結果を **表 8** に示す．**TUT** の列は各手法ごとに TUT データセットの分類をおこなった結果，同様に **OU** の列は各手法ごとに OU データセットの分類をおこなった結果を示す．また，各データセットごとに最も高い F-score を太字で表している．分類結果について，第 5.2.1 項で学習をおこなうネットワークと特徴量の違い，第 5.2.2 項で学習に用いるデータセットの観点で比較をおこなう．第 5.2.3 項で全体の結果についての考察をおこなう．

5.2.1　ネットワークと特徴量の違いによる比較

提案方式と従来手法について F-score の比較をおこなうと，depend と independ のどちらにおいても，提案方式である AE+NN を用いて分類をおこなった場合のほうが，データセットによらず F-score が向上している．depend の場合は，TUT データセットで 0.022，OU データセットで 0.003，independ の場合は，TUT データセットで 0.038，OU データセットで 0.018 F-score が向上している．

5.2.2　学習に用いるデータセットによる比較

depend と independ について F-score の比較をおこなうと提案方式と従来手法のどちらにおいても，環境に依存しない independ の手法をもちいた場合，データセットによらず F-score が低下した．各データセットごとの F-score の低下値を**表 9** に示す．**TUT**，**OU** の列にそれぞれのデータセットでの F-score の低下値を示している．

表 9　independ モデルによる F-score の低下値

F-score	TUT	OU
AE+NN_independ	**0.006**	**0.008**
CNN_independ	0.022	0.023

表 9 から，提案方式である AE+NN_independ を用いた場合，収録環境に非依存なモデルの学習をおこなった際の分類性能の低下を抑えられていることが確認できる．

5.2.3　実験結果のまとめ

提案方式を用いた場合，第 5.2.1 項から従来手法より高い分類性能を得ることができることが確認できた．また，第 5.2.2 から収録環境に非依存なモデルの学習をおこなった際の分類性能の低下を抑えられることが確認できた．これらのことから，CNN Autoencoder を用いて抽出したボトルネック特徴量には，収録環境によらず分類をおこなうために有用な情報が含まれていると考えられる．

6.　まとめ

本報告では，収録環境によらない環境音分類を目標として CNN Autoencoder から抽出したボトルネック特徴量を用いた環境音分類について検討をおこなった．本実験では，異なる環境で収録された 2 種類のデータセットを用いることで提案方式の評価をおこなった．実験結果では提案方式を用いることで単独のデータセットで学習した識別器とほぼ同等の分類性のを得ることができた．このことから，提案方式は複数の収録環境で得られたデータセットを用いて分類の学習をおこなう場合に有効であると考えられる．

今後の課題としては，各データセットごとで学習をおこなった分類部分で，学習で用いていないデータセットの分類をおこなうことが考えられる．本実験ではボトルネック特徴量を抽出したのちは各データセットごとに分類をおこなうための全結合層を学習している．これは，付与されているラベルの種類が異なっているためにおこなったが，「car」と「車」，「people walking」と「足音」など同じようなラベルがつけられている音がある．そのため，これらのラベルについて学習に使用していないデータセットでも分類することが可能であれば提案方式の有効性をより示すことができると考えられる．

謝辞

本研究は JSPS 科研費 18K02862 の助成を受けて実施したものである．

© 2019 Information Processing Society of Japan

参考文献

[1] 大石康智：あらゆる音の検出・識別を目指して –音響イベント検出研究の現在と未来–，日本音響学会 2014 年秋季研究発表会，pp.1521–1524 (2014)

[2] NEC: NEC、音で起きている状況がわかる「音状況認識技術」を開発，NEC，入手先 ⟨https://jpn.nec.com/press/201611/20161128_03.html⟩，（参照 2019-04-15）

[3] Virtanen, T., Mesaros, A., Heittola, T., et al: Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017), (2017)

[4] Plumbley, M. D., Kroos, C., Bello, J., et al: Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)，(2018)

[5] Zohrer, M. and Pernkopf, F: Virtual Adversarial Training and Data Augmentation for Acoustic Event Detection with Gated Recurrent Neural, Proc. INTERSPEECH，pp.493–497 (2017)

[6] Guo, J., Xu, N., Li, L. J., et al: Attention based CLDNNs for short-duration acoustic scene classification, Proc. INTERSPEECH, pp.469–473 (2017)

[7] Ranjan, S. and Hansen, H. L. J: Improved Gender Independent Speaker Recognition Using ConvolutionalNeural Network Based Bottleneck Features, Proc. INTERSPEECH, pp. 1009–1013 (2017)

[8] Yu, H., Tan, Z., Ma, Z., et al: Adversarial Network Bottleneck Features for Noise Robust Speaker Verification, Proc. INTERSPEECH, pp. 1492–1496 (2017)

[9] Mesaros, A., Heittola, T., Virtanen T: TUT Database for Acoustic Scene Classification and Sound Event Detection, In 24th European Signal Processing Conference 2016 (EUSIPCO 2016) (2016)

[10] Hara, S., Hatakeyama, A., Kobayashi, S., et al: Sound sencing using smartphones as a crowsourcing approach, Proc. APSIPA ASC 2017 (2017)

[11] 原直，阿部匡伸，曽根原登：クラウドソーシングによる環境音収集システムを用いた予備収録実験，2015 年日本音響学会秋季研究発表会，pp. 147-148 (2015)

[12] Tanaka, T., Hara, S., Abe, M: A CLASSIFICATION METHOD FOR CROWDED SITUATION USING ENVIRONMENTAL SOUNDS BASED ON GMM-UBM, ASA/ASJ joint meetng (2016)

© 2019 Information Processing Society of Japan