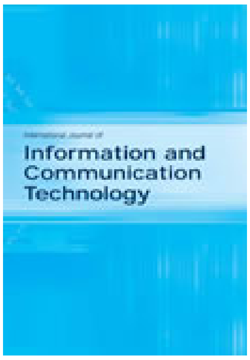


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Extraction of beat features for piano teaching performance based on improved autoencoder

Yating Yang

DOI: 10.1504/IJICT.2025.10072282

Article History:

Received:	20 May 2025
Last revised:	07 June 2025
Accepted:	08 June 2025
Published online:	25 July 2025

Extraction of beat features for piano teaching performance based on improved autoencoder

Yating Yang

School of Music,
Communication University of China,
Nanjing 210000, China
Email: yyt11092025@163.com

Abstract: In the process of piano teaching, accurate extraction of beat features is crucial for assessing the performance level. For the purpose of coping with the issue of insufficient characteristic extraction in the current study, this article first accelerates the convergence speed based on the asymmetric convolutional optimisation autoencoder (ACAE). Then, we obtain the note frequency of the piano performance through the start point detection algorithm, use the relationship between notes and beats to subdivide the beat detection interval, and realise the division of beats through the measurement of confidence level. Finally, the channel attention module (CAM) is introduced into ACAE to complete the adaptive weighting of beat characteristics of every channel, so as to obtain the deep beat characteristics. The experimental outcome demonstrates that the offered approach has a feature classification accuracy of 96.38% and can effectively extract beat features.

Keywords: piano teaching; feature extraction; autocoder; beat division; asymmetric convolution.

Reference to this paper should be made as follows: Yang, Y. (2025) 'Extraction of beat features for piano teaching performance based on improved autoencoder', *Int. J. Information and Communication Technology*, Vol. 26, No. 28, pp.1–16.

Biographical notes: Yating Yang received her PhD from Kookmin University in 2023 in South Korea. She is currently working in School of Music at Communication University of China, Nanjing. Her research interests include piano performance, piano teaching and autocoder learning.

1 Introduction

In piano teaching and performance, the beat is a crucial element, which, like the pulse of the music, gives it life and rhythm. From a pedagogical point of view, whether students can accurately master the beat directly affects their understanding of the overall structure of the piece, the interpretation of rhythmic patterns, and the expression of musical emotion (Griffith et al., 2018). Traditional beat feature extraction methods have limitations when dealing with piano performance data (Johnson et al., 2020). As the artificial intelligence technique quickly growing, the introduction of deep learning algorithms into piano teaching is expected to break through the bottleneck of traditional teaching and bring new vitality and possibilities to piano teaching. As one of the basic elements of music, accurate grasp of the beat is a prerequisite for playing a rhythmic and musical work (Gu, 2022), which not only relates to the overall fluency of the performance, but also directly affects the listener's understanding and feeling of the music (Phanichraksaphong and Tsai, 2021). Therefore, how to effectively extract and analyse students' beat characteristics in piano playing has become a key link in realising intelligent piano teaching and improving teaching quality.

Traditional piano beat feature extraction methodologies include wavelet transform (WT) (Amezquita-Sanchez and Adeli, 2015), empirical modal decomposition (EMD) (Ahmed et al., 2022), principal component analysis (PCA) (Baniya and Lee, 2016), etc. Wang (2018) used WT to implement piano rhythm characteristic extraction and characteristic selection by PCA to improve the accuracy of subsequent recognition models. Liao and Gui (2023) used the shift-invariant singular value decomposition (SVD) algorithm to extract features for piano beats, and formed shift-invariant sparse features for training and test samples by sparse coding. Chiu et al. (2023) used EMD to decompose the audio signal into intrinsic modal function residuals and subsequently extracted the fault features using Hilbert envelope spectral method to improve the accuracy of feature extraction. Cai and Zhang (2022) first analyse the time-frequency characteristics of piano beats by Gabor transform and calculate the Wigner-Ville distribution, and finally identify the beat features from the order spectrum of the music signal.

However, all of the traditional piano beat feature extraction methods are based on shallow learning methods, which usually require dimensionality reduction of complex information. This can lead to poor capability of the resulting features. Deep learning integrates the process of feature extraction into a deep model, which not only improves the accuracy of beat feature extraction, but also greatly saves labour. Qian (2022) used RNN as an unsupervised feature extractor and exploited its excellent feature extraction capability to extract potential representations at the speech frame level. Zhang (2021) designed a deep feature extraction model for musical beats by combining long short memory network (LSTM) and CNN, and utilised the CNN layer of the network to adaptively capture characteristics to reduce the dimensionality as the input to the LSTM layer and train the neural network model, and obtained a small training error. Li (2022) person used BiLSTM to extract music beat features for feature extraction and improved the accuracy by 30.8% relative to the traditional LSTM. Yang et al. (2023) designed a multi-channel CNN to extract multi-scale features of music beats and optimised the model performance by adjusting the weights between the obscured level units. Wu and Chen (2022) proposed a combination of attention mechanism and Bi-LSTM for extracting music beat features, which effectively improves the feature extraction accuracy. Phanichraksaphong and Tsai (2023) proposed the Deformer model to learn a deep representation of audio music data through a denoising process to improve the computational cost of audio sound genre classification.

Compared with the classical deep learning models mentioned above, AE mines the potential feature representations of the data by unsupervised learning and realises data dimensionality reduction, which shows significant advantages in screening complex features and removing feature redundancy. Kumar et al. (2022) achieved end-to-end piano beat recognition by using AE to automatically extract compact and low-dimensional features from raw music beats. Zhao et al. (2023) proposed the Resnet-AE model with the optimal number of levels to remove the redundant features present in the speech signal and extract the deep beat features to improve the recognition accuracy. Han et al. (2024) proposed residual AE to obtain a sparse feature representation by introducing sparsity constraints in the obscured level output forcing the network to extract valid music beat features. Sardari et al. (2022) introduced the idea of degradation process in AE and proposed convolutional AE, which is modelled by reconstructing noise-free samples from samples with added noise so that the extracted music beat features are not affected by noise.

The beat signature of piano performance is a key element in evaluating performance accuracy and musical expression and the traditional beat signature extraction method is susceptible to the interference of factors such as changes in performance speed and differences in note timings. For the goal of improving the accuracy and generalisation of beat feature extraction, this paper proposes a beat feature extraction method for piano teaching performance based on improved AE. The innovations are summarised in the following four aspects.

1 An AE model based on asymmetric convolutional optimisation is proposed. Noise is added to the traditional AE to enhance the generalisation ability of AE and avoid overfitting phenomenon. And the convolutional series operation is added to realise the local sense field and weight sharing. In order to overcome the problem of slow training time of AE, the traditional convolutional kernel structure is decomposed into asymmetric convolutional kernel structure to shorten the model parameters and speed up the convergence.

2 By combining a context-based beat cycle estimation method with a beat tracking algorithm, the beat detection intervals are subdivided by the frequency of notes obtained from the onset detection algorithm, utilising the relationship between the notes and the piano performance beats, and then evaluating the confidence of each detection interval by three confidence measures to realise the segmentation of the piano performance beats.

3 The CAM is introduced on top of ACAE to achieve initial beat feature fusion through feature splicing operations, and utilise CAM to identify discriminative characteristics from the fused representations, enhancing the model's generalisation capability. A series of asymmetric convolution kernels are utilised to compress the characteristic dimensionality, thereby enabling the extraction of deeper abstract characteristics from the beats.

4 Visualisation analysis and comparison experiments were carried out on real datasets, and the visualisation analysis showed that the beat features of piano teaching performance could be effectively extracted. Meanwhile, comparing with the current advanced benchmark approach, the outcome implies that the offered approach outperforms the benchmark method in terms of classification accuracy, macro-F1, and AUC value, and exhibits the best feature classification effect.

2 Relevant technologies

2.1 Convolutional neural network

Fully-connected neural networks have defects in processing image feature information, which leads to the number of parameters in such networks is often very large, and reduces the training speed of the whole model, and also brings the problem of overfitting of the model. Moreover, fully-connected networks have a powerful learning ability that allows the network to memorise every detail in the training data, including noise and outliers, when the number of parameters is too high. As a result, the network will perform very well on the training data, but will perform poorly on the test data or new unseen data, because instead of learning the true distribution and patterns of the data, the network overfits the specific patterns of the training data. Unlike fully connected neural networks, CNNs have strong feature extraction performance in dealing with large-scale images, which can effectively solve the problems encountered by fully connected neural networks in processing image feature information.

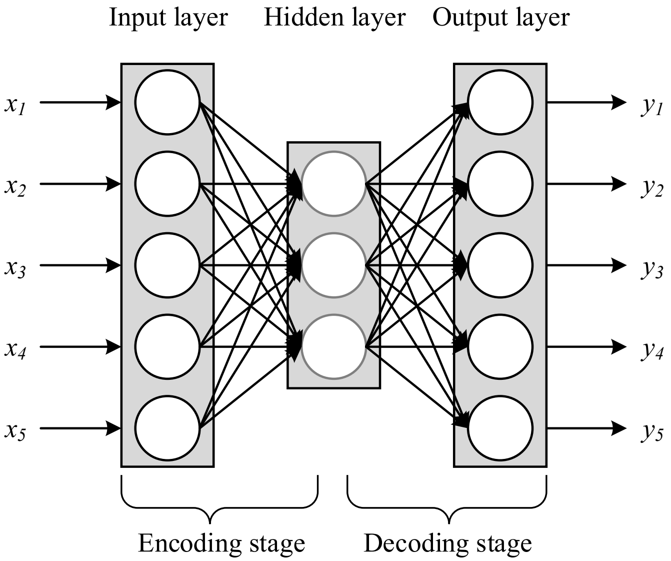
The feature block input to the CNN contains three dimensions of information: width, height, and channel. The structure consists of a stack of input levels, convolutional levels, pooling levels, fully linked levels, activation levels and output levels (Namatēvs, 2017). Among them, the convolutional level and pooling level realise the function of extracting input features, and the weights and bias values to be learned in the network mainly come from the convolutional level, which reflects the computational power of CNN. The local receptive field property determines that the CNN can effectively extract the input features, and the weight sharing and pooling properties enable the CNN to simplify the parameters and computation required for network learning (Cong and Zhou, 2023). Finally in the next level of the network the learned local features will be fused for feature fusion, which avoids the problem of huge number of connections in fully connected neural networks. The idea of weight sharing is to share parameters among different neurons in the form of using the same size of convolution kernel to perform convolution operations, which decreases the amount of parameters to be studied by the network (Taye, 2023).

2.2 Autoencoder

AE is a special type of neural network that is widely used as a neural network structure for data dimensionality reduction tasks (Li et al., 2023), and its basic architecture is shown in Figure 1. AE forces the learning of low-dimensional dense features through the bottleneck layer, which can effectively separate beat-related features from irrelevant features. Compared with AE, CNN (and RNN each have their specific application scenarios and advantages, but also have some shortcomings. CNN is mainly suitable for processing data with a grid structure, such as images. For sequential data, such as text or time series, the structure of CNNs does not capture temporal dependencies well. RNNs may encounter the problem of gradient vanishing or gradient explosion when dealing with long sequences. This is because during backpropagation, the error signal of the RNN needs to be propagated through multiple time steps, which may cause the gradient to become very small or very large, thus making the model difficult to train. The core objective of AE is to capture the chief attributes of the input data *X* in the hidden layer through the encoding process and to try to reconstruct the original input during the decoding process in order to validate the efficacy of AE (Zhang et al., 2022).

The operation of AE can be summarised in two basic steps: firstly, the input data is converted into a low-dimensional code through an encoding process to capture its key characteristics, and secondly, this code is recovered as an output through a decoding process to verify that these characteristics effectively represent the original data (Cemgil et al., 2020). This mechanism makes AE not only suitable for dimensionality reduction tasks, but also makes it perform well in feature extraction in unsupervised learning.

Figure 1 The basic structure of an autoencoder



The encoding stage will encode the input data X to get another representation and the result produced by the encoder is shown in equation (1).

$$Z = f(X) = f_x\left(W_1 X + b_1\right) \tag{1}$$

where f_x is the activation function of the encoder, the decoding stage will be decoded after the encoding of the data, decoding produces the result as shown in equation (2).

$$O = g(X) = g_Z\left(W_2 Z + b_2\right) \tag{2}$$

where g_z is the activation function of the decoder. AE during the training process, $\{W_1, W_2, b_1, b_2\}$ is constantly updated so that the reconstruction error of X and O is small, and the reconstruction loss is shown in Equation (3).

$$E = \sum_{x \in I} J\left(x, g\left(f(x)\right)\right) \tag{3}$$

where J denotes the reconstruction loss function, which will generally be computed using the mean-variance loss function.

Although AE can effectively perform data compression and feature extraction tasks in a variety of contexts, it may encounter several limitations when dealing with some particularly complex data structures. These limitations mainly stem from the fact that the traditional AE structure does not sufficiently emphasise the robustness of the data in the dimensionality reduction process, and that in its encoder part, the complex weight connections between neurons sometimes lead to overfitting of the model to the training data, thus affecting its generalisation ability.

3 Improvement of noise-reducing autoencoder based on asymmetric convolution

The traditional AE is sensitive to the noise in the data and suffers from overfitting and low generalisation level, in this paper, we use the noise-reducing self-encoder (DAE) (Vincent, 2011) with Dropout to address the issue. DAE generates a corrupted dataset by adding Gaussian noise to the original data, and the corrupted data samples are similar to the test set to reduce the bias between the training and test samples, and to enhance the generalisation performance of the whole framework model to other untrained datasets.

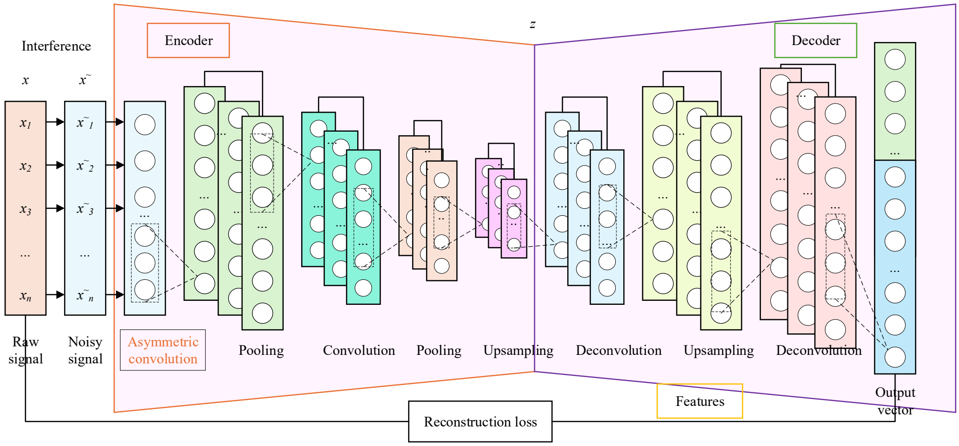
DAE uses densely connected artificial neural networks for feature extraction, encoding and decoding through fully linked levels. The framework of the fully linked neural network is such that the lower neurons can form connections with all the upper neurons, which leads the network to learn the global structure and ignore the local correlations. For this reason, this paper adopts convolutional and pooling levels instead of fully linked levels of DAE. By focusing on local regions through convolutional operations, it helps to extract local features better.

Since convolutional AE uses symmetric convolution operation, it has limitations in extracting features with different scales and directions, this paper introduces an asymmetric structure to extract features with different directions, which enhances the feature representation ability of the model. Compared with the traditional symmetric convolutional kernel structure, this paper decomposes the convolutional kernel to form an asymmetric convolutional kernel structure. The traditional convolutional kernel is usually a symmetric $n \times n$ structure, which requires n^2 parameters. In this paper, we decompose the $n \times n$ structure into $1 \times n$ and $1 \times n$ asymmetric convolutional kernel structures to extract features in the horizontal and vertical directions, respectively. By fusing the features extracted from different convolutional kernels, the features of the model are better expressed, which helps to avoid overfitting to a certain extent. The number of parameters is reduced from n^2 to $2 \times n$, which reduces the computational overhead of the model.

Firstly, a certain range of Gaussian noise is introduced into the training process of ACAE, so that the model learns a clean data representation from the noisy input data and reduces the model's sensitivity to noise, thus effectively improving the quality and usability of the data. CNNs are then used as the main components of the encoder and decoder, allowing the model to be parameter-sharing and locally aware, and able to efficiently extract spatial features from the data. Finally, an asymmetric structure is used to make the encoder and decoder more flexible in design and better adapt to different data distributions and noise types.

The network structure of ACAE is shown in Figure 2. A 2-level convolutional level and a 2-layer pooling level are used as the encoding framework, and a 2-level upsampling level and a 2-layer deconvolutional level are used as the decoding framework. In the coding framework, the convolutional level is used as a characteristic extraction level to capture data features. The pooling level is adopted as a characteristic compression level, which can decrease the size of the characteristic picture and the computation amount of the network on the one hand; on the other hand, it can extract the important data features and reduce the noise component effectively. In this paper, the structure of the 1st level convolution kernel is set to 1×3, the structure of the 2nd layer convolution kernel is set to 3×1, and the movement step of the convolution kernel is set to 1.

Figure 2 The structure of the improved autoencoder ACAE (see online version for colours)



ACAE realises feature extraction by reducing the dimensionality of the data on the basis that the main features of the data are preserved. Weighted summation of features extracted from convolution kernels of different sizes strengthens the information of feature extraction. The introduction of its noise also compensates for the poor generalisation ability of AE.

4 Confidence-based correlation coefficients for the division of performance beats in piano teaching

4.1 Piano teaching playing note onset detection

For the purpose of enhancing the accuracy of subsequent beat feature extraction for piano teaching performance, this paper firstly combines the context-based beat cycle estimation method and beat tracking algorithm, and then through the frequency of notes obtained by the onset detection algorithm, subdividing the beat detection intervals by utilising the relationship between the notes and the beats of the piano teaching performance, and then evaluating the confidence level of each detection interval through the three types of confidence measurements to realise the segmentation of the beats of the piano teaching performance. Then the confidence level of each detection interval is evaluated by three confidence measures to realise the segmentation of piano teaching performance beats.

Instantaneous changes in energy have proven to be a very useful measure, and by detecting these changes, it is possible to obtain the onset of the notes played by the piano teacher. In light of the above content, the combination of energy and phase can effectively detect the onset of a note. The specific method is to use the information in the complex domain obtained by the fast Fourier transform as a combination of energy and phase, which can well combine the information related to energy and phase changes, as shown in equation (4).

$$\tilde{S}_k(m) = \tilde{R}_k(m)e^{j\tilde{\Phi}_k(m)} \tag{4}$$

where m is the number of the piano music frame, $\tilde{R}_k(m)$ is the amplitude of the previous frame, and $\tilde{\Phi}_k(m)$ is obtained from the phase difference between the previous frame and the one before it, calculated as follows:

$$\tilde{\Phi}_k(m) = princ\ \arg\left[2\tilde{\varphi}_k(m-1) - \tilde{\varphi}_k(m-2)\right] \tag{5}$$

where *princ*arg is the obtained phase value mapped to the [–π, π] interval. The actual value of the k^{th} frequency band is $R_k(m)e^{j\varphi_k(m)}$. $R_k(m)$ and $\varphi_k(m)$ are the amplitude and phase of the current frame after the short-time Fourier transform, respectively. The features of each frame are calculated as follows.

$$\Gamma(m) = \sum_{k=1}^{k=K} \left|S_k(m) - \tilde{S}_k(m)\right|^2 \tag{6}$$

The note onset detection signal is obtained by calculating the features of all the frames of the piano performance audio and normalising them. Due to various reasons such as noise, the start point detection signal needs to be smoothed and filtered to obtain the final note start point of the piano teaching performance.

4.2 Piano teaching performance beat cycle estimation

Piano teaching performance has continuity and periodicity, so the beat period can be estimated by the onset detection signal. This article proposes a context-based beat period estimation algorithm. The first step of beat period estimation is to compute the autocorrelation function of the onset detection function for each frame. To make the autocorrelation function clearer, each frame is preprocessed by setting an adaptive moving average threshold, as shown below:

$$\bar{\Gamma}(m) = \sum_{q=m-\theta}^{q=m+\theta} \Gamma_i(q) \tag{7}$$

The sliding window size is set to 16 points. Each point of the probe function is then subtracted from the corresponding threshold and half-wave rectified as shown below:

$$\tilde{\Gamma}_i(m) = \frac{\left(\Gamma_i(m) - \bar{\Gamma}_i(m)\right) + \left|\Gamma_i(m) - \bar{\Gamma}_i(m)\right|}{2} \tag{8}$$

Finally, the autocorrelation function is calculated for the preprocessed piano playing beats as shown in equation (9).

$$A(i) = \frac{\sum_{m=1}^{N} \tilde{\Gamma}_i(m)\tilde{\Gamma}_i(m-i)}{|i-N|} \tag{9}$$

where $i = 1, 2, \ldots, N$ stands for the number of points on the frame and N stands for the frame length. The point $\tilde{\Gamma}_i$ in the autocorrelation domain can be mapped to the beat rate, and the mapping relationship is shown in equation (10).

$$BPM = \frac{60}{\tau_i \star 0.0116} \tag{10}$$

Because of the continuity of beats, the beat rate of an array of two neighbouring frames is intrinsically related, and this factor can be taken into account when estimating the beat period of each data frame. In terms of the above considerations, the calculation of the beat rate t_b of the current frame can be dependent on the estimated beat rate t_{b-1} of the previous frame, for which a hidden Markov model can be constructed. Each column of the model's state transfer matrix consists of a Gaussian distribution with standard deviation σ, as shown below:

$$A(t_i, t_j) = P(t_b = t_j \mid t_{b-1} = t_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(t_i - t_j)^2}{2\sigma^2}\right) \tag{11}$$

where σ is fixed to 8 and t_i, t_j range from 0 to 127. The initial probability distribution is a Rayleigh distribution. The observation sequence is the autocorrelation function for each frame, thus mapping the points to the piano teaching beat cycle via equation (10).

4.3 Histogram generation for beat division based on confidence correlation coefficient

After estimating the beats of the piano performance, a histogram of the beat division is generated based on the confidence correlation coefficient. The loudness of a piano performance over a period of time is related to the amplitude of the audio signal points and the number of signal points, as shown below:

$$L = \left(\sum_{i=0}^{i=N} X_i^2\right)^{0.67} \tag{12}$$

where X_i is the amplitude of the i^{th} point of the audio signal of the piano teaching performance, N is the amount of sampling points of the signal, and the value of L is related to the length of the audio.

The audio signal of the piano teaching performance is centred on the beat position, detecting the position of the point with the highest amplitude within 0.1 seconds around this position, and taking this point as the starting point, intercepting an audio clip of about 0.05 seconds, and obtaining the energy of this piece of audio, i.e., the energy of a single beat. Calculate the energy ratio of the staggered beats: obtain the average of the energy of the two sets $m1$ and $m2$ of beats, and then use equation (13) to obtain the energy ratio of the staggered beats.

$$r = \frac{|m1 - m2|}{(m1 + m2) \times 2} \tag{13}$$

Once the energy ratios are obtained they need to be mapped into coefficients for calculating the displacements, which are calculated as follows:

$$cof = 0.66 \times e^{-12.5 \times r^2} + 0.33 \tag{14}$$

This article gets the histogram of beat distribution by teaching piano to play the position of beat points. The distance between the beat points is the time difference between neighbouring beats, and if the time difference is Δ*t*, then the interval between these two beats is 60 Δ*t*. A histogram of the beat distribution can be obtained by calculating the intervals between all adjacent beat points and normalising it.

5 Improved autoencoder based beat feature extraction for piano teaching performance

After completing the segmentation of piano teaching performance beats, it is necessary to perform feature extraction for each beat. Aiming at the problems of inadequate representation of beat features of piano teaching performance and insufficient model generalisation ability in current feature extraction methods, this paper utilises CAM (Lei et al., 2022) and ACAE (CAM-ACAE) based feature extraction for piano teaching performance beats. In light of ACAE, CAM is introduced to enhance the key features of beats, and feature extraction of piano teaching performance beats is performed under unsupervised learning to significantly improve the recognition of beats. CAM-ACAE mainly consists of an encoder and a decoder. The encoder maps the initial information to the implicit space and captures the implicit characteristic information, while the decoder reconstructs the original piano teaching beats to obtain more generalised deep beat features.

The expressiveness of the obscured level determines the performance of an encoder. The core idea of CAM-ACAE is to feed artificially noise-injected data into the model, with the goal of making the reconstructed data as congruent as feasible with the untainted initial data. Following these ideas, the study introduces noise to create corrupted data, which is then fed into the autoencoder to improve its noise suppression performance via training, with the aim of boosting the model's general robustness against noise interference.

For the purpose of enhancing the CAM-ACAE's characteristic extraction performance, the first level of the encoder adopts a multi-scale convolutional design. The effect of using large convolution kernels is analogous to applying a short-time Fourier transform, and different-sized large convolution kernels operate analogously to time-frequency analysis with multiple window lengths, which enhances the network's ability to process multiple resolutions. The convolution operation can be expressed mathematically in the following manner.

$$y^{l(i,j)} = K_i^l * X^{l(r^j)} = \sum_{j'} K_i^{l(j')} X^{l(j+j')} \tag{15}$$

where $X^{l(r^j)}$ is the j^{th} convolved region in the l^{th} level, W is the width, and $K_i^{l(j')}$ is the j'^{th} weight.

Embedding CAM module in ACAE accomplishes self-adjusting weighting of the channels of the feature maps extracted from the multiscale convolutional levels, allowing the model to focus on information that is more useful for the task at hand. First, the piano beat characteristic picture is sampled by pooling function, and subsequently fed into the shared multilayer perceptron (MLP), and the summed outputs pass through a sigmoid activation function to produce the final channel weighting sequence. The mathematical expression characterising the process is presented below:

$$\begin{aligned} M_c(F) &= \sigma\left(MLP\left(AvgPool(F)\right) + MLP\left(MaxPool(F)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(F_{avg}^c\right)\right) + W_1\left(W_0\left(F_{\max}^c\right)\right)\right) \end{aligned} \tag{16}$$

where F is the input feature; σ is the sigmoid activation operation; W_0 and W_1 are the weights of MLP.

After the weights are assigned by the CAM module, the feature maps are sent to the deep convolution module for further extraction of deep abstract features. This module consists of a series of convolutional levels, and the size of the convolutional kernel is 1 × 3 and 3 × 1 to suppress overfitting while deepening the network. The beat map extracted by CNN for piano teaching needs to be flattened before it can be fed into a classifier consisting of a series of fully linked levels and Softmax activation functions to accomplish the pattern recognition task.

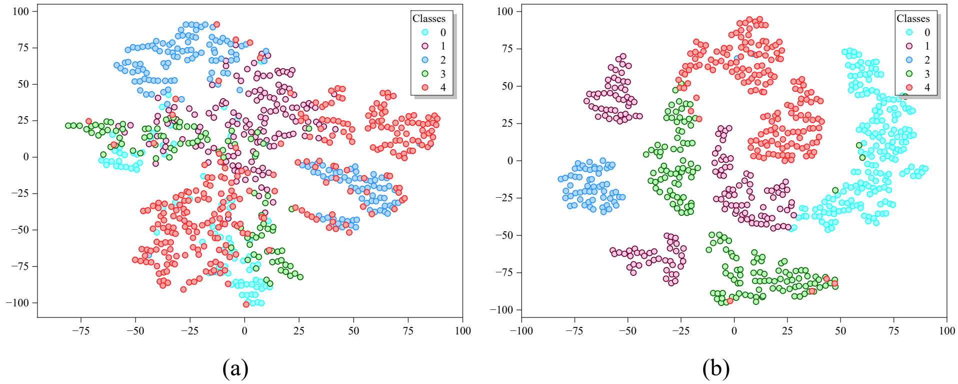
The training of CAM-ACAE is divided into two phases. The first phase is an unsupervised learning phase, in which the parameters are optimised by reducing the error between the reconstructed data and the original input data. After unsupervised learning, a fine-tuning phase is performed in which the output of the encoder's hidden features, i.e., the maximal pooling level, is fed into the fully linked level and the classifier to make classification predictions, and the parameters of the network are fine-tuned by reducing the classification error, which is accomplished with only a small amount of labelled data. The fully linked level is prone to overfitting because of its many parameters. The global average pooling level implements both spreading and dimensionality reduction function through computing the mean value of each characteristic picture. The global average pooling operation involves no learnable parameters, which substantially lowers the likelihood of overfitting. Additionally, it computes the spatial information summation, which boosts the model's stability against input spatial variations.

6 Experimental results and analyses

In this paper, the piano teaching dataset MAESTRO was selected as the experimental dataset. The dataset consists of 3,697 audio and MIDI files, which are aligned with a precision of 3 milliseconds and sliced into individual musical compositions with annotations such as composer, title, and year of performance. MAESTRO includes information on keystroke velocity and sustain, slow pedal, and bass pedal position. The hardware environment for the experiment is Intel Core i7-11800K CPU and NVIDIA GeForce GTX 3060 GPU, the development language is Python 3.6, and the deep learning structure adopted is TensorFlow 2.6.0. The batch_size in the experiment is 16, the initial studying rate is 0.0005, and the amount of training rounds is 100.

In the MAESTRO dataset, a total of five types of features are included, labelled as 0, 1, 2, 3, and 4. For the purpose of visualising the characteristic extraction ability of the offered approach, this paper adopts the t-SNE technique for feature visualisation to compare the features extracted by the CAM-ACAE model with the original data. The t-SNE technique is good at maintaining the localised structure of the data, and t-SNE tries to keep similar data points close to each other in the low-dimensional space during the dimensionality reduction process. If the feature extraction is effective, then data points of the same category will form clear clusters in the low-dimensional space generated by t-SNE. The use of t-SNE visualisation can intuitively reflect the effectiveness of CAM-ACAE in feature extraction, as shown in Figure 3. From the figure, it is observed that the data features extracted by CAM-ACAE show a better feature recognition effect in the t-SNE visualisation results. Although a few sample points were misclassified, overall the vast majority of sample points were successfully clustered. This result implies that the CAM-ACAE model can effectively capture the beat characteristics of piano teaching performance, and exhibits excellent performance.

Figure 3 Comparison before and after CAM-ACAE feature extraction, (a) distribution of original piano beat features (b) distribution of piano beat features after CAM-ACAE feature extraction (see online version for colours)



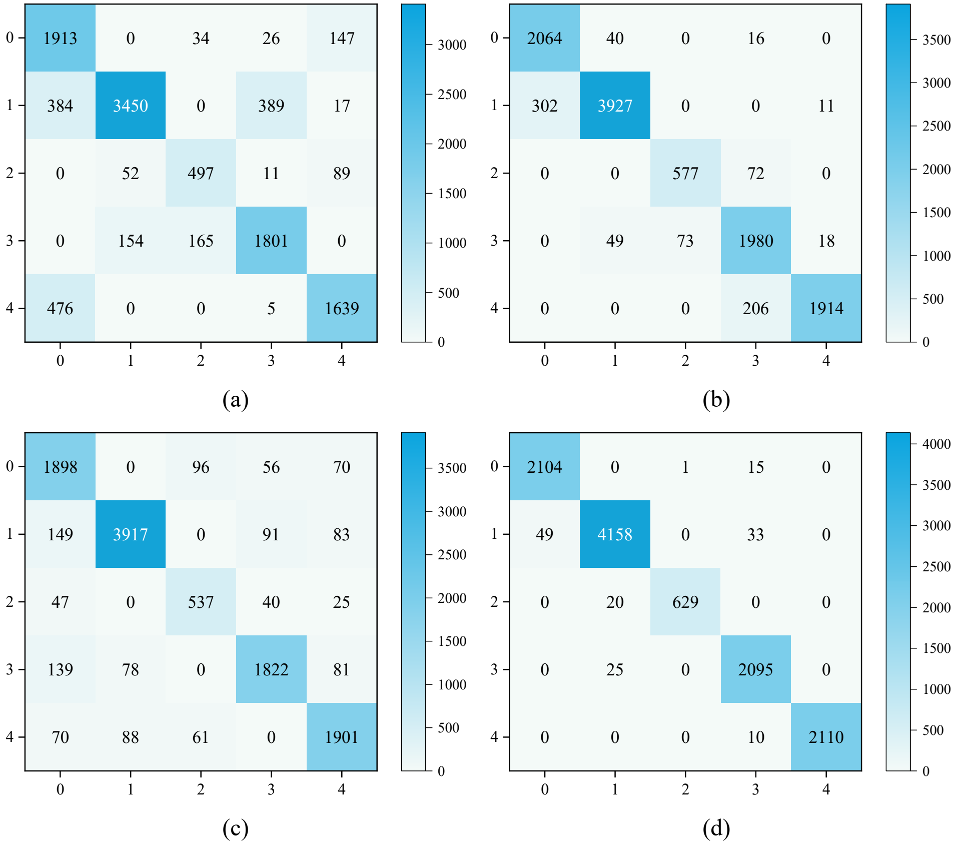
(a)

(b)

For the goal of measuring the superiority of the offered method, in this paper, the feature recognition results of CAM-ACAE are compared with the classification performance of the most competitive feature extraction methods MSCNN (Yang et al., 2023), Resnet-AE (Zhao et al., 2023), and RDU-AE (Han et al., 2024). The confusion matrix results of each method are given below to show the classification performance of each method on different beat feature classes, as shown in Figure 4. From the confusion matrices of each model, it is obvious that there are more misclassifications in the sample classification in the application of the MSCNN algorithm, especially in the classification of beat feature category 2 and beat feature category 4. In addition, samples from several other categories were also misclassified. When the Resnet-AE method was used, the classification accuracy of the samples in beat feature category 2 and beat feature category 4 could be improved, although the diagnostic performance in some categories was improved. Similar misclassification phenomenon was also observed in RDU-AE. In contrast, CAM-ACAE was able to effectively recognise and differentiate various types of beats, and significantly reduced the classification errors, which further confirmed the advantages of CAM-ACAE in terms of the accuracy and reliability of beat-specific classification.

The above experimental outcome cannot visualise the characteristic classification performance of the approaches as numerical values. Therefore, in order to further analyse the experimental results, the classification accuracies of each type of beat features under different methods are listed in Table 1. The accuracies of different methods in each beat feature category. Among them, CAM-ACAE outperforms the other three methods in classification accuracy for all beat characteristic categories. Even in the relatively weak beat feature category 2, the method achieves 94.39% classification accuracy. This result implies that CAM-ACAE has strong stability in classifying various beat feature categories.

Figure 4 The confusion matrix results of each method, (a) MSCNN (b) Resnet-AE (c) RDU-AE (d) CAM-ACAE (see online version for colours)



(a) (b) (c) (d)

Table 1 Classification accuracy of each class of beat features under different methods

Piano beat characterisation categories	*MSCNN*	*Resnet-AE*	*RDU-AE*	*CAM-ACAE*
0	0.8024	0.8753	0.9128	0.9725
1	0.8369	0.8816	0.9262	0.9607
2	0.7658	0.8274	0.8891	0.9439
3	0.8495	0.8594	0.9034	0.9582
4	0.7731	0.8967	0.9067	0.9713

In addition, this paper also uses the accuracy, macro-F1, and AUC values to comprehensively analyse the performance of beat features of different methods, and the results of the comparison experiments are implied in Table 2. The accuracy and macro-F1 of CAM-ACAE are 0.9638 and 0.9406, which are improved by 13.44% and 12.56%

compared to MSCNN, 8.03% and 8.94% compared to Resnet-AE, and 4.81% and 3.83% compared to RDU-AE, respectively. Comparing the AUC values again, the AUC value of CAM-ACAE is 0.9804, which is the closest to 1, indicating that CAM-ACAE is more capable of categorising the beat features of piano playing. This is because CAM-ACAE not only improves AE based on the idea of non-stacked convolution, but also divides the beats of piano playing through the confidence correlation coefficient, and the integration of the CAM module in ACAE significantly improves the feature classification accuracy of the beats of piano playing, and exhibits the best feature classification effect.

Table 2 Feature classification performance of different methods

Method	*MSCNN*	*Resnet-AE*	*RDU-AE*	*CAM-ACAE*
Accuracy	0.8294	0.8835	0.9157	0.9638
Macro-F1	0.8155	0.8512	0.9023	0.9406
AUC	0.8963	0.9381	0.9519	0.9804

7 Conclusions

In the context of the booming growth of digital piano teaching, accurate extraction of playing beat features is the key to improve teaching quality and realise personalised teaching. In order to solve the problems of inadequate representation of piano playing beat features and insufficient generalisation ability in current research, this paper firstly proposes an AE model based on asymmetric convolutional optimisation. Noise is added to traditional AE to enhance the generalisation ability of AE. And the convolution series operation is added to realise the local sense field and weight sharing. To overcome the issue of slow training time of convolution, the traditional convolution kernel structure is decomposed into an asymmetric convolution kernel structure to shorten the model parameters and speed up the convergence. The frequency of notes played by the piano, obtained by the start detection algorithm, is then used to subdivide the beat detection intervals by using the relationship between notes and beats, and then the confidence level of each detection interval is evaluated by a confidence measure to realise the division of beats. Finally, CAM is introduced on the basis of ACAE to complete the adjustable weighting of all channels of the characteristics captured by the multi-scale asymmetric convolutional level, so as to obtain more generalised deep beat features. Comparative experiments were conducted on real datasets, and the results show that the feature classification accuracy of the proposed method and macro-F1 is at least 3.83% higher compared with the benchmark method, which can extract the beat features in piano playing more accurately, and it has an important application value in promoting the development of intelligent piano teaching.

This paper focuses on the research and study of the beat feature extraction method for piano teaching and playing, and although some progress has been made, there are still a lot of shortcomings due to the limited research level and time. Since the rhythm of piano playing is different under different style types, the personalisation of pianos with different rhythms can be attempted when designing the size of the sliding window of the characteristic fusion level, and finally the feature dimensions can be standardised to further enhance the generalisation of the characteristic extraction of piano playing beats.

Declarations

All authors declare that they have no conflicts of interest.

References

Ahmed, A., Serrestou, Y., Raoof, K. and Diouris, J-F. (2022) 'Empirical mode decomposition-based feature extraction for environmental sound classification', *Sensors*, Vol. 22, No. 20, pp.1–19.

Amezquita-Sanchez, J.P. and Adeli, H. (2015) 'A new music-empirical wavelet transform methodology for time-frequency analysis of noisy nonlinear and non-stationary signals', *Digital Signal Processing*, Vol. 45, pp.55–68.

Baniya, B.K. and Lee, J. (2016) 'Importance of audio feature reduction in automatic music genre classification', *Multimedia Tools and Applications*, Vol. 75, pp.3013–3026.

Cai, X. and Zhang, H. (2022) 'Music genre classification based on auditory image, spectral and acoustic features', *Multimedia Systems*, Vol. 28, No. 3, pp.779–791.

Cemgil, T., Ghaisas, S., Dvijotham, K., Gowal, S. and Kohli, P. (2020) 'The autoencoding variational autoencoder', *Advances in Neural Information Processing Systems*, Vol. 33, pp.15077–15087.

Chiu, C-Y., Müller, M., Davies, M.E., Su, A.W-Y. and Yang, Y.-H. (2023) 'Local periodicity-based beat tracking for expressive classical piano music', *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 31, pp.2824–2835.

Cong, S. and Zhou, Y. (2023) 'A review of convolutional neural network architectures and their optimizations', *Artificial Intelligence Review*, Vol. 56, No. 3, pp.1905–1969.

Griffith, K.R., Ramos, A.L., Hill, K.E. and Miguel, C.F. (2018) 'Using equivalence-based instruction to teach piano skills to college students', *Journal of Applied Behavior Analysis*, Vol. 51, No. 2, pp.207–219.

Gu, W. (2022) 'Recognition algorithm of piano playing music in intelligent background', *Mobile Information Systems*, Vol. 20, No. 5, pp.12–29.

Han, X., Chen, W. and Zhou, C. (2024) 'Musical genre classification based on deep residual auto-encoder and support vector machine', *Journal of Information Processing Systems*, Vol. 20, No. 1, pp.13–23.

Johnson, D., Damian, D. and Tzanetakis, G. (2020) 'Detecting hand posture in piano playing using depth data', *Computer Music Journal*, Vol. 43, No. 1, pp.59–78.

Kumar, A., Solanki, S.S. and Chandra, M. (2022) 'Stacked auto-encoders based visual features for speech/music classification', *Expert Systems with Applications*, Vol. 10, pp.21–34.

Lei, D., Ran, G., Zhang, L. and Li, W. (2022) 'A spatiotemporal fusion method based on multiscale feature extraction and spatial channel attention mechanism', *Remote Sensing*, Vol. 14, No. 3, pp.46–58.

Li, P., Pei, Y. and Li, J. (2023) 'A comprehensive survey on design and application of autoencoder in deep learning', *Applied Soft Computing*, Vol. 138, pp.21–34.

Li, W. (2022) 'Analysis of piano performance characteristics by deep learning and artificial intelligence and its application in piano teaching', *Frontiers in Psychology*, Vol. 12, pp.75–87.

Liao, Y. and Gui, Z. (2023) 'An intelligent sparse feature extraction approach for music data component recognition and analysis of hybrid instruments', *Journal of Intelligent & Fuzzy Systems*, Vol. 45, No. 5, pp.7785–7796.

Namatēvs, I. (2017) 'Deep convolutional neural networks: structure, feature extraction and training', *Information Technology and Management Science*, Vol. 20, No. 1, pp.40–47.

Phanichraksaphong, V. and Tsai, W-H. (2021) 'Automatic evaluation of piano performances for STEAM education', *Applied Sciences*, Vol. 11, No. 24, pp.31–45.

Phanichraksaphong, V. and Tsai, W-H. (2023) 'Automatic assessment of piano performances using timbre and pitch features', *Electronics*, Vol. 12, No. 8, pp.1–13.

Qian, Z. (2022) 'Feature extraction method of piano performance technique based on recurrent neural network', *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)*, Vol. 14, No. 2, pp.1–14.

Sardari, S., Nakisa, B., Rastgoo, M.N. and Eklund, P. (2022) 'Audio based depression detection using convolutional autoencoder', *Expert Systems with Applications*, Vol. 19, pp.26–37.

Taye, M.M. (2023) 'Theoretical understanding of convolutional neural network: concepts, architectures, applications, future directions', *Computation*, Vol. 11, No. 3, pp.1–23.

Vincent, P. (2011) 'A connection between score matching and denoising autoencoders', *Neural Computation*, Vol. 23, No. 7, pp.1661–1674.

Wang, X. (2018) 'Research on the improved method of fundamental frequency extraction for music automatic recognition of piano music', *Journal of Intelligent & Fuzzy Systems*, Vol. 35, No. 3, pp.2777–2783.

Wu, G. and Chen, W. (2022) 'Construction and application of a piano playing pitch recognition model based on neural network', *Computational Intelligence and Neuroscience*, Vol. 11, No. 5, pp.1–11.

Yang, J., Zhou, Y. and Lu, Y. (2023) 'Multimedia identification and analysis algorithm of piano performance music based on deep learning', *Journal of Electrical Systems*, Vol. 19, No. 4, pp.1–11.

Zhang, C., Geng, Y., Han, Z., Liu, Y., Fu, H. and Hu, Q. (2022) 'Autoencoder in autoencoder networks', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 35, No. 2, pp.2263–2275.

Zhang, J. (2021) 'Music feature extraction and classification algorithm based on deep learning', *Scientific Programming*, Vol. 20, No. 4, pp.16–29.

Zhao, X., Wang, Y. and Cai, X. (2023) 'A ResNet-based audio-visual fusion model for piano skill evaluation', *Applied Sciences*, Vol. 13, No. 13, pp.1–14.