International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 17 (2025) pp. 1-14

https://doi.org/10.70917/ijcisim-2025-0280
Article

# Application of Convolutional Neural Network-based Note Recognition and Analysis Technique for Piano Performance in Basic Piano Teaching

**Yihan Zhou ***

School of Primary Education, Hunan First Normal University, Changsha, Hunan, 410000, China;
froggirl73@163.com

**Abstract:** Optical sheet music recognition converts traditional paper sheet music into electronic sheet music, which is convenient for players to learn and practice playing. In this paper, taking the piano notes as the research object, combining the training needs of piano playing technique with the features of piano music and its recognition characteristics, the residual network is used to improve the learning ability of the network and the convolutional layer is adjusted to be a residual depth-separable convolutional network. The recurrent layer is fed with SRU to accelerate the learning speed of the network and accelerate the note classification. The transcription layer uses Focal Loss to deal with note overfitting samples, forming a lightweight note recognition method based on improved CRNN.The NSynth Dataset dataset tests show that the accuracy of the three sub-networks reaches a high level when the model is trained up to 17 times, and there is no overfitting problem in the trained network model.The addition of the SRU module reduces the model training time and The optimal sequence error rate of the whole model for the deformed semantic sheet music is 32.89%. The improved CRNN network can improve students' performance in piano playing test by applying it to piano intelligent assisted playing training.

**Keywords:** CRNN network; residual network; note recognition; Focal Loss; piano playing

## 1. Introduction

Basic piano teaching is the main course of musicology majors in higher teacher training colleges and universities, which integrates music knowledge, theory, technology and the application of music art, is not only the basis for learning all related courses in music, but also directly linked to the classroom teaching and extracurricular music activities of basic music education [1-4]. In basic piano teaching, note recognition training is an important link that cannot be ignored. Notes are the basic elements that make up a piece of music, and each note has its own unique height and time value, and mastering these basics is crucial for students to play the piano [5-6]. For students of basic piano teaching, note recognition is not only the foundation of learning music, but also the starting point for them to move forward smoothly on the path of music, because at this stage, they often feel confused and lost when they face sheet music [7-10]. At this time, note recognition training is particularly important, which can help students establish a basic knowledge of sheet music, so that they can accurately perform musical works [11-12]. And the cultivation of note recognition ability requires meticulous teaching strategies and effective training methods.

With the continuous development of artificial intelligence technology, convolutional neural network (CNN) has achieved great success in the field of image recognition [13]. CNN is a deep learning model, which is mainly used to process data with a grid-like topology, such as images [14]. CNN mainly consists of three kinds of layers: convolutional layer, pooling layer, and fully connected layer. The convolutional layer is mainly used to extract image features and it generates multiple convolutional feature mappings by applying multiple convolutional kernels to the input image [15-17]. The pooling layer, on the other

hand, is designed to reduce the data dimensionality, and the commonly used pooling methods are maximum pooling and average pooling [18]. The fully connected layer, on the other hand, connects the feature vectors output from the convolutional and pooling layers to achieve the classification task [19]. In recent years, CNN has also been widely used in audio processing [20]. Its advantage lies in the fact that it can extract the features in audio and perform classification and recognition. In the recognition and analysis of piano playing notes, CNN mainly processes audio signals so as to realize note detection and classification, which shows the advantages of high precision and real-time, and is of great significance for improving the quality of basic piano teaching [21-25].

Aiming at the shortcomings of existing methods, this paper proposes a lightweight note recognition method based on improved CRNN. The residual network provides a better learning characterization ability for deep neural networks and combines with deep separable convolution to achieve better network learning ability while reducing the model computation. The bidirectional recurrent neural network composed of input SRUs realizes the parallel computation of the network, and Focal Loss handles the unbalanced samples. Overall, the training time duration and recognition accuracy of the CRNN network model are improved. To evaluate the performance of the proposed model in this paper, piano audio files from The NSynth Dataset dataset, and deformed piano sheet music image dataset are used for training and testing.

## 2. Teaching design of basic piano playing techniques

### *2.1. Basic Piano Teaching*

The teaching of basic piano group lessons in colleges and universities should include two major parts: basic theoretical knowledge of the piano and basic playing techniques. Only in this way can students understand and master the discipline of piano art more efficiently and comprehensively [26].

(1) Piano basic theoretical knowledge teaching: basic knowledge of reading music, piano music listening training, artistic expression.

(2) Basic piano playing technique training: basic playing state, basic playing method, basic playing technique.

There are many kinds of playing methods in piano playing, among which the three kinds of playing methods of non-consecutive playing, continuous playing and broken playing are extremely widely used, therefore, the basic piano collective class teaching requires students to strictly master these three kinds of basic playing methods.

Basic piano teaching in colleges and universities requires mastery of the basic piano playing techniques, including: five-finger equilibrium, scales, arpeggios, diatonic (third, sixth), chords, octave intervals, vibrato, vibrato, etc., and combined with the speed and strength of the training.

### *2.2. Piano playing technique training content design*

(1) Content design of technical training in legato and accentuation for college pianos

Legato is a method of playing two or three tones together with a drop roll for beginners in the art of piano in the primary touch learning. The purpose is to train the beginner to strengthen the control of piano tone from the aspect of fingers. After the mastery of legato, the training of musical colors such as piano rhythms can be strengthened in order to let the beginner feel the musical effects of different keystrokes, i.e. fast and slow keystrokes correspond to crisp and lyrical tones respectively.

Accenting is the treatment of the piano work marked with accent marks, especially for the treatment of metronomic and rhythmic accents, the former often appearing in the first beat of each bar. For the piano beginner, you can follow the first beat of the accent, the second beat of the accent and other gradual ways to train.

(2) Content design of technical training in college piano scales and chord exercises

Regarding the training of piano playing scales and transposition, especially the training of piano scales focuses on the mastery of thumb turning. The technique of finger turning involves the thumb actively drilling into the center of the palm and touching the keys with the outside of the tip of the thumb in order to ensure the consistency of the sound produced by each finger touching the keys.

In order to strengthen the piano scale technique training more effectively, we need to train each key in a specialized way, and if necessary, we can fully understand and master the tonality of the tune, so that the beginner piano trainee can have a thorough grasp of the musical knowledge of the key.

At the same time, need to strengthen the chord connection training, in order to do a good job for the improvisation accompaniment preparation. In addition, is to pay attention to good piano arpeggio training, in the key speed and finger extension size and other aspects to meet the requirements.

## 3. Convolutional neural network-based note recognition for piano performance

### *3.1. Characteristics of piano musical notes and their identification*

Sound is a physical phenomenon that is essentially produced due to the vibration of an object and its transmission in the form of waves to the human ear or audio receiver with the help of solids, liquids and air. Music, as a type of sound signal, is a number of specified notes co-organized according to specific laws to express human thoughts and emotions.

The system of musical characteristics is divided into three stages:

The first is basic features.

The second is complex features.

The third is the overall characteristics.

The overall characteristics include the structure of the whole piece of music, the artistic style it represents and the emotional connotation it expresses. The complex features include the harmony, rhythm and melody of the music. The basic characteristics of music, which are the basis for segmenting musical endpoints and identifying chords, include four main characteristics: timbre, pitch, duration, and intensity. The four basic features have different roles in the analysis and evaluation of music, and they are ranked as follows according to people's sensitivity to music recognition:

(1) Tone: a subjective feeling of the sound of the instrument.

(2) Pitch: The lower the frequency of vibration, the lower the sound produced.

(3) Tone length: is determined by the duration of the vibration produced by the piano strings.

(4) Intensity of sound: It is determined by the amplitude of the vibration produced by the piano strings.

### *3.2. Convolutional Neural Networks*

With the continuous development of Convolutional Neural Network technology (CNN), it is now one of the best known deep learning approaches in which multiple layers are trained in a robust manner. The mathematical model of convolutional neural network can be summarized as follows:

$$x_i^m = \sum_{i \in T_i} x_j^{m-1} \times N_{ij}^m + h_i^m \tag{1}$$

In Eq. $x_i^m$ denotes the feature map of the mth layer, $T_i$ is the image input to the CNN, $x_j^{m-1}$ denotes the jth output of the $m-1$ th hidden layer, $N_{ij}$ is the convolution kernel, and $h_i^m$ is the offset of the layered output. The result of equation (1) is processed by the activation function. The above operation extracts different features from the image data and maintains scale invariance. The pooling layer can consist of a maximum pool or an average pool, which downsamples the data, reduces the number of training parameters, achieves dimensionality reduction, avoids overfitting phenomena, and reduces noise. Eq:

$$X_j^m = f_{down}\left(X_j^{m-1}\right) \tag{2}$$

In the equation, $f_{doun}$ denotes the downsampling function. The CNN convolution and pooling operations are repeated according to a predefined number of network layers. The processed feature vectors are stacked and classified while using fully connected layers. Typically, classification is performed using sof-tmax and SVM classifier functions.The main goal of CNN training is to minimize the value of the loss function. Its mathematical expression is:

$$L(W,b) = -\sum^{N}\sum^{K} g\left(\hat{y}_i = j\right)\log p_i^i \tag{3}$$

In the equation, $w$ is the weight, $b$ is the bias, $g$ is the indicator function, and $j$ is the training sample category. If $\hat{y}_i \neq j$, then $I = 0$ or $\hat{y}_i = j$, then $I = 0$. The probability that the ith training sample is judged to be in category j is $p_i^j$. The loss function and its expectation are used to calculate the difference between the CNN output and the training data, i.e., the residual. The parameters of each neuron layer in the CNN can be optimized and tuned using gradient descent. In PPIR, image data preprocessing, including RGB model or HSV model transformations, is performed first, followed by image denoising and filtering, segmentation, and selection of test and training data. The preprocessed

data is subsequently passed through different layers of the CNN. Optimization of different parameters and adjustment of the number of layers can also improve the image recognition accuracy.

CNN is actually a deeper ANN network system with more nodes, while comparing with the simple ANN. the CNN focuses on the characteristics of the convolutional algorithm to achieve the weights shared with each other, which reduces the order of magnitude of the connections and takes into account the two-dimensional characteristics; at the computational level. the CNN's core characteristics are the same as the BP network of the weights are transmitted in the forward direction with the errors in the forward and reverse direction, and the use of errors to change the weights at weights at each level.

A convolutional network is mainly composed of the following five structures:

(1) Input layer

The entry layer is usually the entry layer of the whole neural network, in the case of a convolutional neural network that manages graphics, it usually represents the pixel matrix of a photograph, e.g. 28*28*1, 32*32*3.

(2) Convolutional layer

As you can tell by the name, the convolutional layer is the most crucial component in a kind of convolutional neural network. Unlike the fully connected layer, the input of each node in the convolutional layer is just a small piece of the previous neural network, usually the size is having 3*3 or 5*5, and different from the fully continuous layer, the input of each node on the convolutional layer is some small piece of the previous neural network. The convolutional layer view allows each individual chunk of the neural network to be analyzed in more depth, thus obtaining features with a higher degree of abstraction.

(3) Pooling layer

The pooling layer can reduce both the features and the width of the matrix without affecting the depth of the 3D matrix. The pooling method operates similarly to converting a known high-resolution image into a lower resolution image. By utilizing the pooling layer, it is possible to gradually reduce the number of nodes in the final full-link layer, which in turn achieves the purpose of reducing the number of parameters of the entire neural network.

(4) Fully connected layer

The convolutional layer and pooling layer can be regarded as a process that automatically performs image feature acquisition, and when the feature acquisition is finished or the spreading is finished, the task is still divided by the full-continuous layer graph.

(5) Softmax layer

Similar to in the fully connected neural network, using the Softmax layer, it is also able to obtain the size of the probability of belonging to different types in the current sample. However, usually, the pooling layer follows the computational layer of convolution and can be used to reduce the dimensionality between the feature mapping and the network parameters. Like the convolutional layer, since the pooling layer is still translationally fixed, the results computed for it refer to other pixels as well. The functions of the pooling layer are mainly: it is used to increase the important feature signals, to reduce the features, to reduce the computational effort, and to reduce the fitted cases.

## *3.3. Design of convolutional neural network*

Existing studies usually use deep networks in the convolutional layer to extract features, and use complex gating units in the recurrent layer for serial computation, but no additional processing for the loss function in the transcription layer, which leads to long training time and low accuracy of CRNN. To address the above problems, this paper proposes a lightweight note recognition method based on improved CRNN.

### 3.3.1. Feature extraction

For the sheet music image, the feature map extracted by the convolutional layer will be transformed into a sequence of features, which will then be passed into the recurrent layer for learning. Therefore, in this paper, improvements are made in the convolutional layer to both speed up the training and ensure the effectiveness of the extracted features.

It is assumed that the size of the standard convolutional kernel is $D_K \times D_K \times M$ , which represent the width, height, and dimension of the convolutional kernel, respectively. The $K$ represents the dimensional size of the convolution kernel. Since convolution kernels are usually constructed with equal width and height, the length and height are represented by $D_K$ . The parametric number of $N$ standard convolutions is $D_K \times D_K \times M \times N$ and the computational amount is

$D_K \times D_K \times M \times N \times D_W \times D_H$.

Similarly set the size of the convolution kernel for deep convolution $D_K \times D_K \times M$ . The convolution kernel size for point-by-point convolution is $1 \times 1 \times M$ , with a total of $N$ , and the number of parameters for depth separable convolution is $D_K \times D_K \times M + M \times N$ . The depth convolution performs a total of $D_W \times D_H$ multiplications and additions, and the $N$ point-by-point convolutions have to do a total of $D_W \times D_H$ multiplications and additions, and thus the number of computations for the depth separable convolution is $D_K \times D_K \times M \times D_W \times D_H + M \times N \times D_W \times D_H$. The ratio of the depth separable convolutional computation to the standard convolutional computation is shown in Equation (4):

$$\frac{D_K \times D_K \times M \times D_w \times D_h + M \times N \times D_w \times D_h}{D_K \times D_K \times M \times N \times D_w \times D_h} = \frac{1}{N} + \frac{1}{D_K^2} \tag{4}$$

In order to improve the learning ability of the network, residual network is introduced on this basis. Residual network is proposed to solve the network degradation problem when there are too many hidden layers in deep neural network, usually the input-output mapping relationship of neural network is denoted as $H(x) = F(x), x$ as the input of the neural network, $F(x)$ denotes the function fitted by neural network, and $H(x)$ is the output result of the neural network. And the residual function is defined as $H(x) = F(x) + x$ , the input data is directly added to the output as a constant mapping by jump connection, and the stacking layer learns new features based on the input features to have better performance.

In this paper, we combine the ideas of depth separable convolution and residual network to improve the convolutional layer as residual depth separable convolutional network. The depth separable convolution defined in this paper uses LeakyRule activation function.

### 3.3.2. Note Classification Prediction

LSTM and GRU have a strong dependency problem, where the input of the neural unit depends on the output of the previous neural unit, which also leads to the use of serial computation for the whole network, and slow training and inference.

SRU improves the gating unit to lift the strong dependence of the network on the hidden state, and realizes the parallel computation of the network by obtaining the gating parameter matrix in advance, which dramatically accelerates the learning speed of the network [27].

The structure of the SRU at the moment $t$ , $x_t$ denotes the input, $h_t$ denotes the output state, $c_t$ denotes the internal state, $\sigma$ is the Sigmoid activation function, and $v_f$ and $v_r$ are parameter vectors mapping to $c_{t-1}$. $g_t$ is a linear transformation on $x_t$: $g_t = wx_t, w$ is its parameter matrix. $f_t$ denotes the forgetting gate and $r_t$ the reset gate.

To mitigate the recursion, its two gating units $f_t$ and $r_t$ no longer depend on the hidden state $h_{t-1}$ of the previous moment, but on the internal state $c_{t-1}$ of the previous moment. The forgetting gate $f_t$ is computed as shown in equation (5). The reset gate $r_t$ is computed as shown in equation (6):

$$f_t = \sigma\left(w_f x_t + v_f \odot c_{t-1} + b_f\right) \tag{5}$$

$$r_t = \sigma\left(w_r x_t + v_r \odot c_{t-1} + b_r\right) \tag{6}$$

$b_f$ and $b_r$ are the bias cells of $f_t$ and $r_t$ , respectively. The $c_t$ synthesizes the information of the past state and the current input, and reduces the computation by using the Hadamard product instead of the matrix product, as defined in Equation (7):

$$c_t = f_t \odot c_{t-1} + \left(1 - f_t\right) \odot g_t \tag{7}$$

The $h_t$ is jump-connected, as shown in Eq. (8), and the input $x_t$ is directly included in the computation, which aims to optimize the gradient propagation, so that it does not make the gradient disappear when the depth of the network is increased due to the propagation distance being too far. Eq:

$$h_t = r_t \odot c_t + (1 - r_t) \odot x_t \tag{8}$$

The SRU model contains two layers of bi-directional SRUs, each layer of bi-directional SRUs has a total of 512 hidden units, which firstly pass the information in chronological order, and then the last output unit passes the information in reverse chronological order. The outputs of the two-layer bidirectional SRUs are finally predicted and classified by dot-multiplication computation.

### 3.3.3. Sample Learning

Traditional CTC does not train well for datasets that are extremely unbalanced or contain a large number of low-frequency samples.Focal Loss solves this problem well, and it overcomes the overfitting and underfitting problems due to unbalanced datasets. Based on focus theory and cross entropy, the FocalLoss loss function is defined as shown in Equation (9):

$$L_{Focal\_Loss}(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \tag{9}$$

Where $p_t$ is the size of the prediction probability, $\alpha_t$ and $\gamma$ are the adjustable factors, and $L_{Focal\_Loss}(p_t)$ represents the loss function when the prediction probability is $p_t$.

The $p_t$ in the formula reflects how close the predicted sequence is to the true sequence. The $p_t$ reflects the ease of classification. $\alpha_t$ is used to regulate the ratio between the loss of positive and negative samples to suppress the imbalance in the number of positive and negative samples, and $\gamma$ is used to control the imbalance in the number of simple/hard-to-distinguish samples. The $(1 - p_t)^{\gamma}$ is the modulation factor of this loss function, which tends to 0 for accurately categorized samples $p_t \to 1$ and to 1 for inaccurately categorized samples $p_t \to 0$.

Overall, Focal Loss increases the weight of the misclassified samples in the loss function, which makes the loss function tend to the difficult-to-classify samples and helps to improve the accuracy of the difficult-to-classify samples.

## 4. Implementation of note recognition for piano playing

### *4.1. Data set construction*

The dataset used in this paper is divided into two parts: audio files for the piano from Google's off-the-shelf dataset, The NSynth Dataset, and chord files generated using MIDI data organized according to the twelve-mean rhythm.

The NSynth Dataset is a large-scale, high-quality, annotated and labeled note dataset for eleven instruments, including the erhu, flute and piano. There are three sources of sound production for these note data, including acoustic, electronic, and synthesized instruments.

Since this study is only about the piano, a total of 35,480 piano musical notes are selected in this paper, of which 30,000 note signals are used as the training set, 4,000 samples are used as the test set, and 1,480 samples are used as the test set.

In these samples contain data from eight kinds of pianos, and there are five notes from the same pitch and the same instrument, and the difference lies in the speed of the notes or the degree of each note pressed. Specifically the five levels 25, 50, 75, 100 and 127, the meaning behind which is the amount of energy contained in a single note.

### *4.2. Data pre-processing*

The current datasets are all audio files in wav or MIDI format, while the input to the convolutional neural network should be image data. Therefore, in this paper, MIDI music is converted to wav format first, and then the constant Q transform is applied to the audio in wav format to finally obtain the spectrogram.

(1) Constant Q transformation

The actual musical signal in the processing, usually on the time frame of the signal using the function for weighting, also known as adding window, constant Q transform and Fourier transform of the biggest difference is that this window function can be dynamically changed according to the center frequency changes. Let us have a signal $x(n)$ whose constant Q transform is:

$$x(k) = \frac{1}{N_k} \sum_{n=0}^{N_k - 1} x(n) w_k(n) e^{-j2\pi n \frac{f_k}{f_s}} \tag{10}$$

Where $k$ is the frequency ordinate, $N_k$ is the window length whose value is related to $k$, $w_k(n)$ is the corresponding window function i.e., filter, $f_s$ is the sampling frequency, $f_k$ is the center frequency of the tonal scale, which is computed using the following formula:

$$f_k = 2^{\frac{k}{B}} \cdot f_{\min} \tag{11}$$

where $f_{\min}$ denotes the lowest frequency value of the signal to be processed and $B$ denotes the total number of spectra in each octave. From the above equation:

$$f_{k+1} - f_k = f_k \left( 2^{\frac{1}{B}} - 1 \right) \tag{12}$$

Let $\delta_{f_k} = f_{k+1} - f_k$, the bandwidth of the frequency band at $f_k$, be obtained:

$$Q = \frac{f_k}{\delta_k} = \frac{1}{2^{\frac{1}{B}} - 1} \tag{13}$$

Since the value of Q is required to be constant in a constant Q change, the value of Q is determined by B.

Up to this point, the window length $N_k$ can be obtained as:

$$N_k = Q \frac{f_s}{f_k} \tag{14}$$

So the larger the center frequency $f_k$, the smaller the window length $N_k$.

The constant Q transform is performed on the already obtained wav file. The specific realization steps are as follows:

Step1: Read the existing wav file with sampling frequency $f_s$ of 56.0k Hz.

Step2: For the frequency resolution of a time-frequency signal, it is related to the number of frequency line bands B within an octave.

Step3: For the time resolution, it is related to the frame shift. Therefore, if the sampling frequency is 56.0k Hz, the frame shift is 512 samples.

Step4: For the lowest frequency $f_{min}$, $f_{min}$ takes the lowest note of the piano.

Step5: For the total number of frequency points $n\_bins$, the size of its value directly corresponds to the dimension of the first dimension of the input data, which is also the width of the spectrogram. In the case of the lowest frequency $f_{min}$ is determined, the size of its value should be satisfied to cover the entire band of piano frequency.

(2) Speech Spectrogram Processing

In order to reduce the difficulty of network training and improve the recognition accuracy, the band segment corresponding to the spectrogram is segmented for the fundamental frequency to be recognized, i.e., the fundamental frequency is set to be $f$, and its MIDI pitch calculation formula is as follows:

$$P = 69 + 12 \log_2 \frac{f}{440} \tag{15}$$

Intercept horizontally on a 440*400*3 spectrogram to get a sub-spectrogram.

## *4.3. Model training*

The lightweight note recognition network model with improved CRNN proposed in this paper is trained using the produced sample note images.

Experimental environment: the network model was composed by writing a program component network model through Python, Keras was chosen to train the model, the operating system was Windows 10, the processor was Inter(R) Core(TM) i7-6300HQ @2.30GHz, and the running memory was 8G.

Improved CRNN for Lightweight Note Recognition Method Initial stage of network training, the input sample note images were scaled at the input layer to 90*90*5. The number of iterations for the note sample images was set to 80. The initial learning rate of the network model was set to 0.03. The model batch processed sample data of 16 at a time. The random inactivation probability of neurons during model training was 30%. Adams was chosen as the optimizer and loss value and accuracy were chosen to analyze the network performance.

After the model training, the accuracy curves of the three sub-networks of the model with the number of iterations during the training process were obtained. The box-and-line plots of the recognition accuracy of digits, pitches and time values are shown in Fig. 1, in which the curves are normally symmetrically distributed.

Combining the distribution characteristics of the box plot and the recognition accuracy data, it can be obtained that in the 17th time of model training, the recognition accuracy in the three sub-networks of the model reaches more than 0.9. In order, they are time value, pitch and number, which are related to the three features, indicating that the network model has extracted the main note features.

The accuracy of the three sub-networks did not show fluctuating changes from 17 times of training to the end of training, and maintained a high level. The accuracy of the model for recognizing the time value, pitch, and number is 0.953, 0.924, and 0.927, respectively.The trained network model has no overfitting or underfitting problem, which indicates that the method in this paper can be used for note feature extraction.

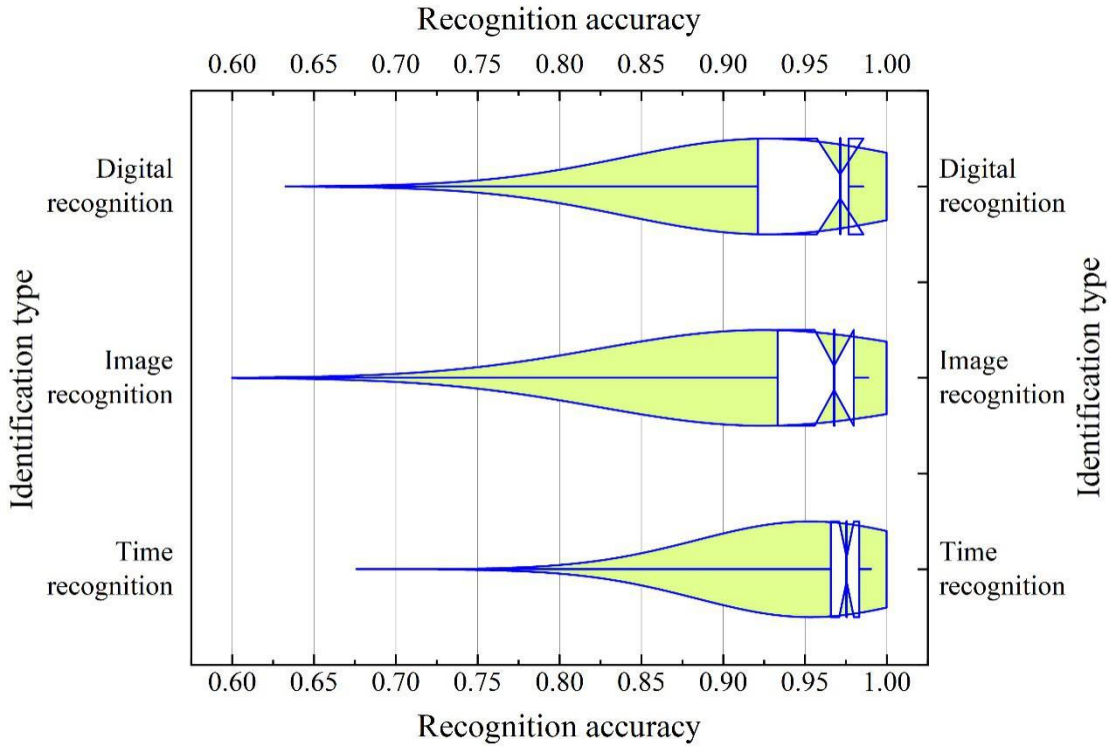


***Figure 1.*** The number of Numbers, the number, the time value of the box diagram.

## *4.4. Model realization*

### 4.4.1. Evaluation indicators

In this paper, 2 metrics, note error rate and sequence error rate, are used to evaluate the model

performance. Note Error Rate (SER): represents the ratio of the average of the edit distance from the model's predicted output note encoding sequence to the true value sequence to the length of the true value sequence. Its mathematical representation is shown in Equation (16):

$$SER = \frac{I + D + S}{N} \times 100\% \tag{16}$$

where edit distance is the sum of insertion $(I)$, deletion $(D)$ and substitution operation $(S)$, and $N$ is the sequence length.

Sequence Error Rate (ER): indicates the proportion of the erroneous sequence $E$ in the model prediction output to all the test sequences $T$, and any note recognition error in the sequence is regarded as sequence error. Its mathematical representation is shown in equation (17):

$$ER = \frac{E}{T} \times 100\% \tag{17}$$

### 4.4.2. SRU effectiveness analysis

In order to verify the effectiveness of the SRU module in reducing the model training time consumption, a training time consumption comparison is done for four network models, C-BiLSTM, C-SE-BiLSTM, C-SE-BiSRU, and C-SE-SRU.

All three models are iterated 30000 times under the same experimental environment. The training elapsed time curve of each model is shown in Fig. 2. From the model time-consuming curves in the figure, it can be seen that the average time consumed per 80 rounds of training for the C-BiLSTM network, C-SE-BiLSTM network, C-SE-BiSRU network, and C-SE-SRU network is 1.37 min, 1.50 min, 0.83 min, and 0.67 min, respectively.Obviously, the training time consumed for the C-SE-SRU network is lower than that for the the three networks and the training time is about one-half of the original C-BiLSTM network.

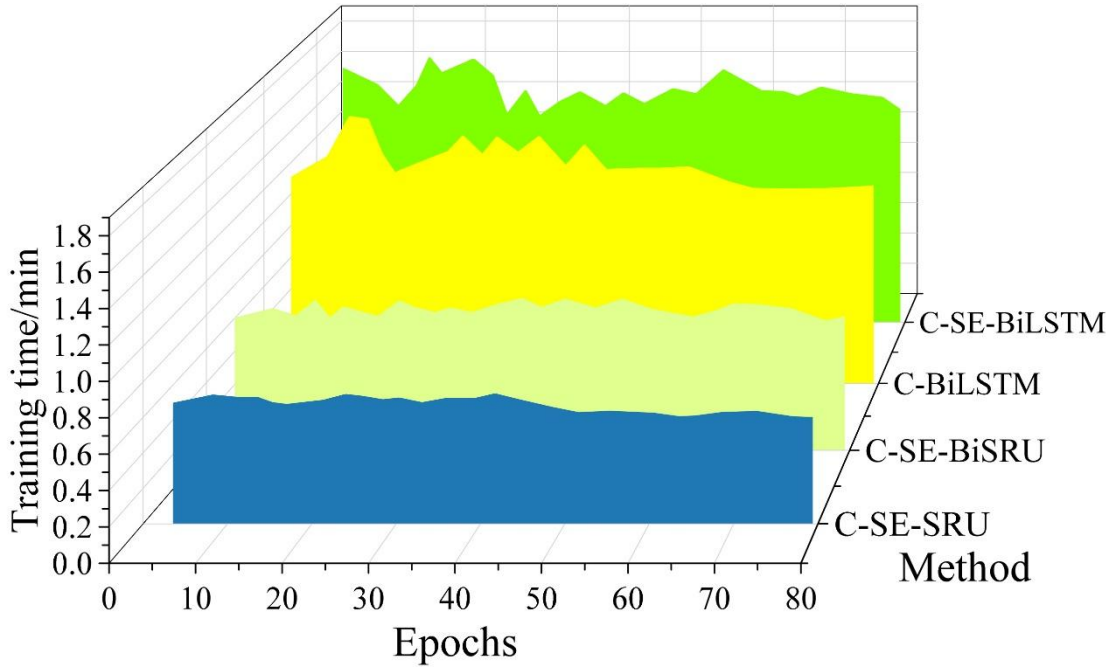


***Figure 2.*** Model training time curve.

### 4.4.3. SER/ER comparison

In this paper, different combinations of training data and test data are combined, both of which output 2 types of encoding form results, semantic and semantic-independent. The detailed data comparison between this paper's method and the original CRNN method is shown in Table 1, where S and SI are the

original and deformed training data, respectively.

From the data in the table, it can be learned that this paper's method has the lowest note error rate of 0.32% for the original semantic score, and the lowest sequence error rate of 32.89% for the deformed semantic score. Under different training and testing scenarios, the method proposed in this paper outperforms the original CRNN model in terms of recognition performance. In particular, when the model trained on the original score is tested using the deformed score, the symbol error rate and sequence error rate of the output semantic-independent coding decreases from 56.64% and 93.25% to 9.15% and 33.4%, which also indicates that the model proposed in this paper has better generalization.

***Table 1.*** Detailed data of the method and the original crate method.

| Test data | | | | | | | | Test data | |
|---|---|---|---|---|---|---|---|---|---|
| Primordial | | | | Deformation | | | | | |
| Semantics | | Semantic independence | | Semantics | | Semantic independence | | | |
| SER% | ER/% | SER% | ER/% | SER% | ER/% | SER% | ER/% | - | |
| 0.76 | 12.35 | 1.14 | 27.35 | 60.12 | 98.21 | 56.64 | 93.25 | CRNN | S |
| 0.32 | 9.52 | 0.52 | 13.9 | 39.16 | 47.6 | 9.15 | 33.4 | Ours | |
| SER% | ER/% | SER% | ER/% | SER% | ER/% | SER% | ER/% | - | |
| 3.56 | 47.04 | 1.01 | 32.45 | 4.15 | 39.25 | 1.27 | 30.78 | CRNN | SI |
| 1.15 | 18.62 | 0.55 | 20.01 | 1.36 | 32.89 | 0.91 | 19.21 | Ours | |

### 4.4.4. Song Recognition Test

In order to verify the effectiveness of the method proposed in this paper, the short scores of some songs are tested, the notes in the short scores are extracted, the note recognition is performed, the recognition accuracy and the recognition speed are tested and compared with other methods.

The total number of notes and the number of notes recognized for a variety of songs are shown in Figure 3, and the comparison methods include LSTM, CS2S, and R2-CRNN.

From the figure, it can be seen that the number of notes recognized curve of this paper's method is higher than other methods. For the song Flower Butterfly, the number of notes recognized by this paper's method differs from the total number of actual notes by 3 notes, which indicates that this paper's method has good recognition accuracy.

Compared with the R2-CRNN method, for the song Jingle bells, the number of notes recognized by this method and the R2-CRNN method is 97, while for the song Thousand Miles Away, the number of notes recognized by this method and the R2-CRNN method is 276 and 256, respectively, which shows that the method is more broadly recognizable.

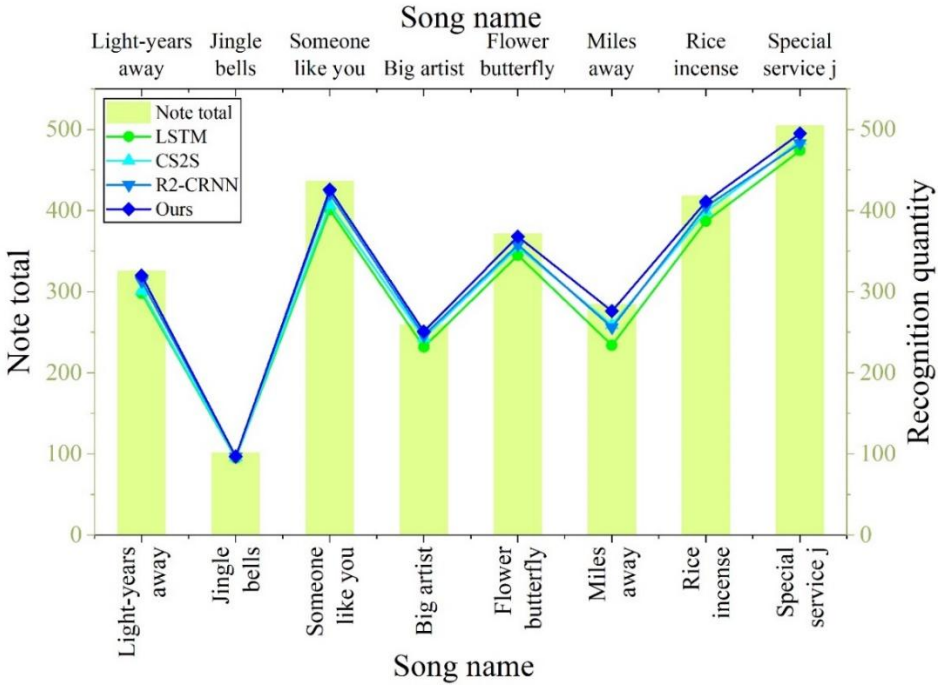

***Figure 3.*** The number of notes of multiple songs and the number of notes.

The performance comparison of different note recognition methods is shown in Fig. 4, the recognition correct rate of LSTM for the song Thousand Miles Away is 82.4%, which is worse than other note recognition methods.

The note recognition correct rate of this paper's method for the selected songs are above 96%, reflecting the excellent recognition accuracy of this paper's method.

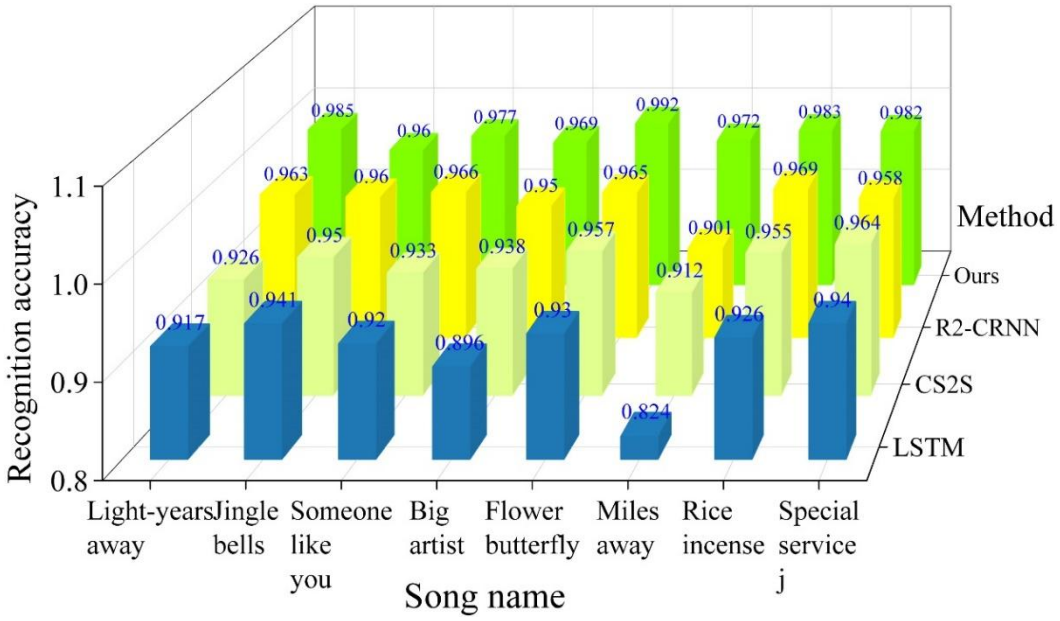


***Figure 4.*** Performance comparison of different notes.

Comparison of the recognition time of different note recognition methods is shown in Figure 5, for the song Jingle bells, the recognition time of each method is 3.58s, 4.26s, 5.15s and 6.3s respectively, and the recognition time of each note recognition method is controlled within 7s. The reason may be because the melody of the song is relatively simple, and the recognition difficulty is low.

The recognition time of this paper's method for the selected songs does not exceed 7s, and the recognition time for the song Light Years Away reaches 6.85s, which is 8.41s and 6.8s less than that of LSTM and CS3S, respectively.

The performance of this paper's method in terms of the number of recognized notes, the correct recognition rate, and the recognition time consumed can show that this paper's method has better recognition accuracy and recognition speed.

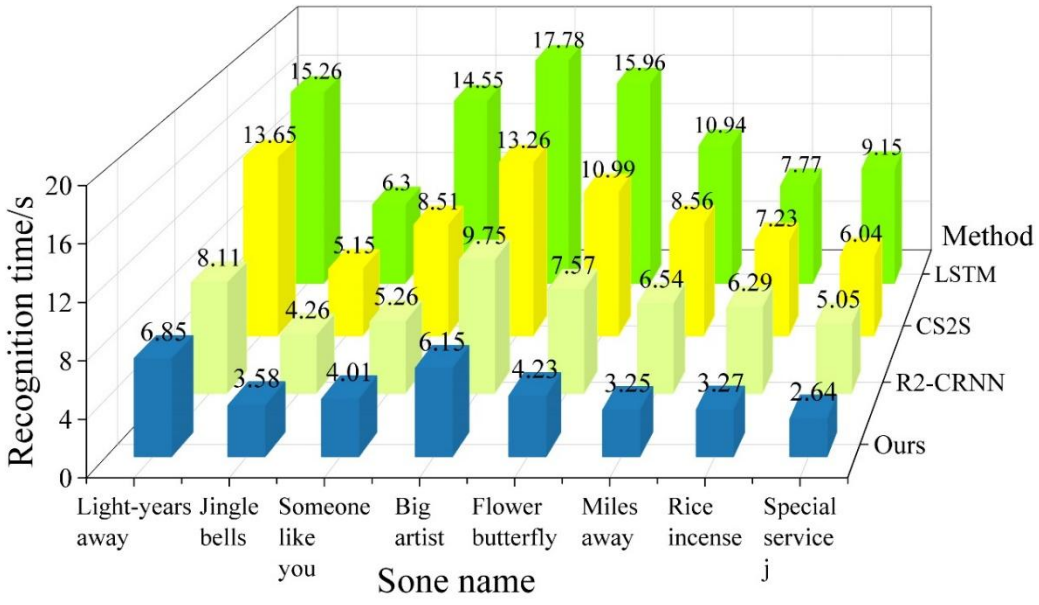


***Figure 5.*** The recognition method of different notes is compared.

## *4.5. Modeling Instructional Applications*

By applying the convolutional neural network-based note recognition and analysis technology for piano playing designed in this paper to the field of piano teaching, it can be designed to form an intelligently assisted piano playing technology. In order to analyze the impact on students in terms of learning after the application of intelligently assisted piano playing learning, a post-class test was conducted on the experimental and control classes. Independent samples t-test was performed on the performance data of both classes using SPSS 24.0 software. The results of the sample test for the piano playing technique test scores are shown in Table 2.

From the data in the table, it can be seen that the mean scores of the experimental class over the control class were 92.51 and 83.07 respectively.The significance of the F-test of the Levene test of the variance equation is 0.000, which is greater than 0.05, so it is the data of the test of chi-square of the variance. The t-test significance (two-tailed) value of the mean equation is 0.012, which is less than 0.05, indicating that there is a significant difference in the piano playing performance assisted by intelligence. That is, the learning effect of piano playing in the experimental class is significantly better than that of the control class. It can be seen that the application of intelligent piano teaching based on the lightweight note recognition method with improved CRNN, under the basic piano teaching, is very effective and worth implementing as it can help the learning performance of the students in this school, which determines the children's learning outcomes and efficiency to a great extent.

***Table 2.*** Test results of piano performance test results.

| Group statistics | | | | | | | |
|---|---|---|---|---|---|---|---|
| | Class | | Average | Standard deviation | | Standard error mean | |
| Grade | Laboratory class | | 92.51 | 7.526 | | 0.592 | |
| | Cross-reference class | | 83.07 | 9.101 | | 1.426 | |
| Independent sample inspection | | | | | | | |
| | Levin variance equivalence test | | Average equivalent t test | | | | |
| | F | sig | T | df | Sig. | Mean difference | Standard error difference |
| Assumed equal variance | 3.296 | 0.000 | 3.001 | 101 | 0.012 | 4.527 | 1.226 |
| Unassuming equal variance | | | 3.014 | 96.859 | 0.012 | 4.527 | 1.226 |

## 5. Conclusions and shortcomings

### *5.1. Conclusion*

In order to optimize the recognition accuracy and recognition speed of convolutional neural network for piano playing notes, this paper forms a lightweight note recognition method based on improved CRNN by improving the feature extraction of convolutional layer, classifying notes using SRU and improving sample learning. The method achieves more than 92% recognition accuracy for time value, pitch, and number in model training, respectively.

The main contribution of this work is to optimize the feature extraction and reduce the computation of convolutional neural network by introducing the idea of depth-separable convolution and residual network, and adopting LeakyRule activation function as the depth-separable convolution strategy. Focal CTC is used to equalize the learning samples to broaden the recognition dimension of piano music notes. Especially in piano deformed semantic score note recognition, the recognition SER and ER of this paper's method are 1.36%. 32.89%, respectively.

### *5.2. Inadequacies*

Of course this work has some limitations. In order to further improve the performance note recognition model, the following aspects can be investigated in the future:

(1) The current lightweighting work is based on the CRNN model, which may be able to use networks such as Transformer to have a better enhancement in the optical score recognition task.

(2) In the face of uneven sample data processing, this paper applies the Focal CTC strategy for processing, and there is still room for improvement.

(3) For music review, the selection of dataset can still be adjusted and optimized.

**References**


1. Zhu, J. (2020). Reform of piano basic course teaching for college music performance major–Research on the application of flipped classroom in teaching. In 2020 Conference on Educational Science and Educational Skills (pp. 693-699).
2. Zhengshan, X., & Sondhiratna, T. (2024). Teaching methods of basic piano courses in preschool education in higher vocational colleges. Asia Pacific Journal of Religions and Cultures, 8(1), 385-395.
3. Young, M. M. (2016). A national survey of university-level group piano programs. MTNA e-Journal, 7(3), 13.
4. Zitong, W., & Pattananon, N. (2023). Teaching Contents and Teaching Methods of Basic Piano Courses in China. Journal of Modern Learning Development, 8(11), 361-369.
5. Mehta, A. A., & Bhatt, M. S. (2015, January). Optical music notes recognition for printed piano music score sheet. In 2015 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-6). IEEE.
6. Sezen, H. (2021). Note Reading Methods Used in Piano Education of 4 to 6 Years Old Children. Educational Research and Reviews, 16(7), 310-324.
7. Yin, X. (2023). Educational innovation of piano teaching course in universities. Education and Information Technologies, 28(9), 11335-11350.
8. Chen, X., & Tang, J. (2014). Research on Piano Music Signal Recognition Based on Short-Time Fourier Analysis. Advanced Materials Research, 853, 680-685.
9. Wang, Z. (2025). Recognition algorithm of piano playing music notes based on improved hidden Markov model. Egyptian Informatics Journal, 31, 100746.
10. Gadre, S. (2022, August). Piano Notes Recognition using Digital Signal Processing. In 2022 International Conference on Signal and Information Processing (IConSIP) (pp. 1-5). IEEE.
11. Shang, R. (2022). A Deep Learning-Enabled Composition System Based on Piano Score Recognition. Mobile Information Systems, 2022(1), 9132697.
12. Nakamura, E., Yoshii, K., & Dixon, S. (2017). Note value recognition for piano transcription using markov random fields. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(9), 1846-1858.
13. Ma'arif, A., Rahmaniar, W., Fathurrahman, H. I. K., & Frisky, A. Z. K. (2022). Understanding of Convolutional Neural Network (CNN): A Review. International Journal of Robotics & Control Systems, 2(4).
14. Rawat, W., & Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. Neural computation, 29(9), 2352-2449.
15. Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. Insights into imaging, 9(4), 611-629.
16. Sarıgül, M., Ozyildirim, B. M., & Avci, M. (2019). Differential convolutional neural network. Neural Networks, 116, 279-287.
17. Uchida, K., Tanaka, M., & Okutomi, M. (2018). Coupled convolution layer for convolutional neural network. Neural Networks, 105, 197-205.
18. Saha, S. (2018). A comprehensive guide to convolutional neural networks—the ELI5 way. Towards data science, 15, 15.

19. Xu, G. (2017, June). Deep convolutional neural network to detect J-UNIWARD. In Proceedings of the 5th ACM workshop on information hiding and multimedia security (pp. 67-73).
20. Han, Y., Kim, J., & Lee, K. (2016). Deep convolutional neural networks for predominant instrument recognition in polyphonic music. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(1), 208-221.
21. Basha, S. S., Dubey, S. R., Pulabaigari, V., & Mukherjee, S. (2020). Impact of fully connected layers on performance of convolutional neural networks for image classification. Neurocomputing, 378, 112-119.
22. Böck, S., & Schedl, M. (2012, March). Polyphonic piano note transcription with recurrent neural networks. In 2012 IEEE international conference on acoustics, speech and signal processing (ICASSP) (pp. 121-124). IEEE.
23. Liao, Y. (2022). Educational evaluation of piano performance by the deep learning neural network model. Mobile Information Systems, 2022(1), 6975824.
24. Wu, R. (2021, June). Research on automatic recognition algorithm of piano music based on convolution neural network. In Journal of physics: conference series (Vol. 1941, No. 1, p. 012086). IOP Publishing.
25. Ru, T. (2025). Deep Learning-Based Automatic Piano Note Recognition and Performance Generation System. International Journal of High Speed Electronics and Systems, 2540814.
26. Yang Jing & Jung Gook Young. (2023). College Piano Teaching Based on Multimedia Technology. International Journal of Web-Based Learning and Teaching Technologies (IJWLTT),18(2),1-10.
27. Haoyang Zhao,Lianzhong Huang,Ranqi Ma,Kai Wang,Jianlin Cao,Tiancheng Wang... & Xiangjun Chen. (2025). Adaptive online modeling of ship maneuvering based on multiscale convolutional hybrid SRU models. Ocean Engineering,333,121486-121486.