2024 20th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD) | 979-8-3503-5632-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICNC-FSKD64080.2024.10702215

Non-destructive Ripeness Judgement of Watermelon Based on Mel Spectrogram and ECAPA-DTNN

Jun Liu, Hongbao Shi, Yingjie Xia, Jinping Li
School of Information Science and Engineering
University of Jinan
Jinan, China

Abstract—**In daily life, the majority of consumers generally have little professional knowledge to select ripe watermelons. However, the laboratory non-destructive methods for judging the ripeness of watermelon has limitations and is not suitable for consumers. Obviously, it is important and practical to develop a method for judging the ripeness of watermelon that is convenient for consumers. In this study, we utilized portable smartphones to record watermelon tapping audio signals of different ripeness levels and established a dataset. Then, we preprocessed the recorded audio signals. Next, we extracted Mel spectrogram features from the audio signals in the frequency domain using Short Time Fourier Transform and Mel Filter Bank. Finally, we used these features as input to train the ECAPA-TDNN model. The experimental results show that the accuracy of 89.5% on the test set, demonstrating that this method can achieve non-destructive judging of watermelon ripeness in daily life and is suitable for the majority of consumers.**

Keywords-watermelon ripeness; audio signal; Mel spectrogram; ECAPA-TDNN; non-destructive judgement

I. INTRODUCTION

Watermelon is a fruit that can replenish water during the hot summer. Consumers often use methods such as observation, weighing, and tapping to judge watermelon quality when making a purchase. However, these methods rely heavily on personal experience and are highly subjective, making it difficult for ordinary consumers to accurately judge ripeness.

At present, the main non-destructive detecting technologies for judging watermelon ripeness in laboratories include near-infrared spectroscopy[1], nuclear magnetic resonance(NMR)[2], and acoustic technology[3]. Near-infrared spectroscopy can detect soluble solids in watermelon, such as sugar, acid, cellulose, minerals, and other components, to judge ripeness and quality. However, this method is influenced by the volume of the watermelon and the thickness of its peel, and high-power transmission can affect the fruit's internal quality. NMR is a technique that quantitatively determines the structure of small and large molecules containing hydrogen protons. During the fruit ripening process, the content of hydrogen protons in water and sugar changes, allowing the internal quality of the fruit to be judged. However, NMR is expensive and the equipment is not portable. Acoustic detection technology is usually used to detect holes or cracks in samples, but it can also be applied to fruit detection, as the internal structure and density of fruits change as they grow[4,5]. Therefore, acoustic detection technology can be regarded as the voiceprint of fruits.

Compared with near-infrared spectroscopy and nuclear magnetic resonance detection, acoustic detection technology has the advantages of simplicity, low cost, and easy accessibility. Therefore, in the scenario of consumers selecting watermelons, acoustic detection technology has broader application prospects and promotional value. However, most current research on acoustic detection technology[6-10] is still limited to laboratory environments and requires professionals to use professional equipment for judgement, which is usually not suitable for ordinary consumers. Therefore, we utilize common and portable smartphones available to consumers to collect watermelon tapping audio signals and judge the ripeness of watermelons by analyzing these audio signals.

II. RELATED WORK

Life experience shows that watermelons of different ripeness levels produce different tapping sounds. Ripe watermelons produce crisp tapping sounds, while unripe watermelons produce dull tapping sounds. This difference occurs because sound is produced by the vibration of objects. As the watermelon grows and matures, the internal bubbles gradually enlarge, altering the tapping sound[10].

He[3] first proposed using tapping audio signals to analyze the ripeness of watermelons. By measuring the percussion sound wave curve of watermelon fruits and using Fast Fourier Transform (FFT) to analyze the power spectrum density to assess fruit quality. Later, various research methods were developed, which can be basically divided into two methods:

- The core concept of the first method was that the audio signal characteristics of watermelons at different ripeness levels had different distribution intervals. The ripeness of watermelons was judged by manually dividing the intervals, which avoided model training and did not require a large amount of datas. Xiao[11] used FFT to obtain the power spectrum of the audio signal and assessed the relationship between the frequency at the extreme value of the power spectrum and the ripeness of the watermelon. Chen[12] first used Daubechies wavelet multi-resolution decomposition to obtain the optimal wavelet coefficients, and then used

statistical hypothesis testing to establish an inequality for judging watermelon ripeness. The features extracted by the above two methods are relatively simple. Pamungkas[13] extracted four features to judge the ripeness of watermelon, namely frequency, amplitude, weight, and soluble solid content. It was found that as the watermelon matures, the main frequency and amplitude of the tapping audio signals tend to decrease, while the fruit weight and soluble solid content tend to increase. Based on this, a linear regression equation for ripeness was established.

- The core concept of the second method was to extract the features of the audio signal and used machine learning to learn the rules from the datas. This method's advantage was that it eliminates the need for manual feature extraction, with the algorithm automatically generating a classification model. Zeng[14] combined the time domain features of the audio signal, including zero crossing rate, short-time energy, and sub-band short-time energy ratio, and then used Support Vector Machine (SVM) for training. Chawgien[15] extracted frequency domain features by Fourier Transform to obtain the spectral features of the audio, then using the maximum frequency as the key feature, and used gradient boosting tree to train the classification model. In addition to analyzing the characteristics of the tapping audio signals in the time domain or frequency domain, Choe[16] conducted research from the perspective of sound speed. He used piezoelectric transducers as transmitters and receivers to measure the speed of sound propagation through the surface of watermelons. He found that the surface sound speed gradually increases as the watermelon matures. He also used Backpropagation neural network to analyze the correlation between sound speed and ripeness.

The first method approach focused on the characteristics of audio signals. Although it avoided processing large amounts of datas, it relied on manual feature selection and was limited to solving specific problems. The effectiveness of this method was greatly affected when the watermelon variety or detection equipment changed. The second method approach used machine learning methods and trained the model, but the feature extraction method used was relatively simple, and the model used was outdated, leading to poor generalization ability. Additionally, most of these methods were tested in controlled laboratory environments. Building datasets required professionals to use professional equipment to collect tapping audio signals, making the process cumbersome and not portable. Obviously, this made it difficult to promote these methods among ordinary consumers.

In order to solve the portability issue among ordinary consumers and better simulate real-world usage scenarios, we utilized common smartphone devices to directly collect the tapping audio signals of watermelons in an outdoor environment. To improve the problem that the extracted feature information is not complex enough, we focused on the frequency domain because audio signals usually contain richer feature information there. We extracted Mel spectrogram features of the audio from the spectrum. Because the Mel spectrogram reduced the dimension of spectral features while retaining the main information, which helped to reduce the complexity and computational overhead of the training model. With the development of deep learning, Convolutional Neural Networks[17] (CNNs) and Time Delay Neural Network[18] (TDNNs) have become widely used network structures in this field. CNNs have demonstrated excellent performance in computer vision and other areas, and research by Albert[19] proved that the CNN architecture is also suitable for acoustic detection of watermelon ripeness. TDNNs are widely used in the audio field. To further improve the performance of TDNN, Desplanques[20] improved the TDNN architecture based on x-vector[21] and proposed the ECAPA-TDNN model, which paid more attention to the attention, information propagation, and aggregation between channels. Duan's[22] research showed that the ECAPA-TDNN network performed well in the field of voiceprint recognition and achieved remarkable results in speaker recognition applications.

In summary, we built a tapping audio dataset collected by smartphones, extracted Mel spectrogram features from the audio signals, and used the ECAPA-TDNN model for training to develop a method that enables ordinary consumers to effectively select ripe watermelons.

III. PROPOSED METHOD

The algorithm flowchart is shown in Fig. 1 The subsequent sections will detail two key components: Mel spectrogram features extraction and the ECAPA-TDNN model structure.

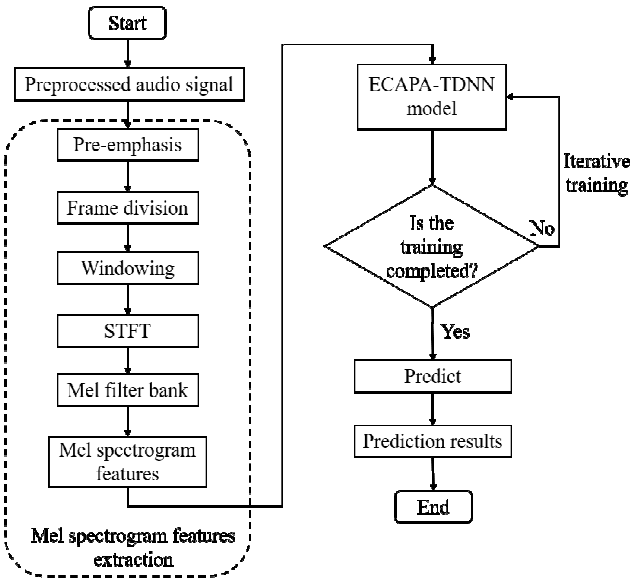

Figure. 1 Algorithm flow chart

A. Mel spectrogram features extraction

The audio signal contains more features in the frequency domain than in the time domain, so it needs to be converted from the time domain to the frequency domain by Fourier Transform. However, for sudden change signals like watermelon tapping audio signals, directly performing Fourier Transform on the entire signal is not meaningful. Therefore, the audio signal needs to be framed and windowed. As shown in

the algorithm flowchart in Fig. 1, the Mel spectrogram features extraction process is divided into the following five steps:

- Pre-emphasis: The collected audio signals contains more energy in the low frequency band and less energy in the high frequency band. By applying a high-pass filter, a positive gain characteristic is introduced into the signal, so that the signal in the high frequency part is relatively enhanced.
- Frame division: Because non-stationary signals can be regarded as relatively stable signals in a short time, in order to perform Fourier transform, the signal is divided into segments. The calculation equation of the number of frames is shown in (1).

$$\mathrm{n}=\frac{s\times t}{h}+1 \tag{1}$$

Where s is the sampling rate, t is the duration of the audio, and h is the frame shift.

- Windowing: Truncating the signal in the time domain may cause spectrum leakage, leading to errors in spectrum analysis. Windowing can help reduce spectrum leakage. We select a Hamming window with smooth transition characteristics and multiply it point by point with the original signal to avoid drastic changes. The Hamming window function is shown in (2).

$$H(\mathrm{n})=\begin{cases}0.54-0.46\times\cos\left(\dfrac{2\pi n}{M-1}\right), & 0\le n\le M-1\\ 0, & \text{other}\end{cases} \tag{2}$$

Where n=1,2,⋯,N-1, N represents the total length of the window function; M represents the effective length of the window function; $H(n)$ represents the value of the window function at the nth sampling point.

- Short Time Fourier Transform (STFT): The watermelon tapping sound is a signal with sudden changes. By adjusting the window size and overlap rate in STFT, it can effectively capture these sudden changes in the audio signal.
- Mel Filter Bank: Mel frequency is a nonlinear frequency scale. It is linearly related to the normal frequency in the low frequency band and logarithmically related to the high frequency band. The Mel scale is shown in Fig. 2, and the equation is shown in (3):

$$\mathrm{m}=2595\log_{10}\left(1+\frac{f}{700}\right) \tag{3}$$

Where f is the normal frequency and m is the Mel frequency.

Mel Filter Bank are generated by Mel scale. Each filter in the filter bank is a triangular filter, as shown in Fig. 3 Its expression is shown in (4).

$$H_m(k)=\begin{cases}0, & k<f(m-1)\\ k-f(m-1), & f(m-1)\le k\le f(m)\\ \dfrac{f(m-1)-k}{f(m+1)-f(m)}, & f(m)\le k\le f(m+1)\\ 0, & k>f(m+1)\end{cases} \tag{4}$$

Where f is a triangular filter, $f(m)$ is the center frequency of f, and $f(m)$ is defined by (5):

$$f(m)=\frac{N}{F_S}B^{-1}\left[f_l+m\frac{f_h-f_l}{M+1}\right] \tag{5}$$

$$B^{-1}(x)=700\left(e^{x/1125}-1\right) \tag{6}$$

In (5), N is the number of sample points in a frame signal; F_s is the sampling frequency of the signal; f_l and f_h represent the lowest frequency and the highest frequency in the triangular filter bank respectively; M represents the number of filters.

Finally, the spectrum feature is multiplied and accumulated element by element with the frequency response of each triangular filter. This process calculates the energy value of the frame datas in the frequency band corresponding to each triangular filter.

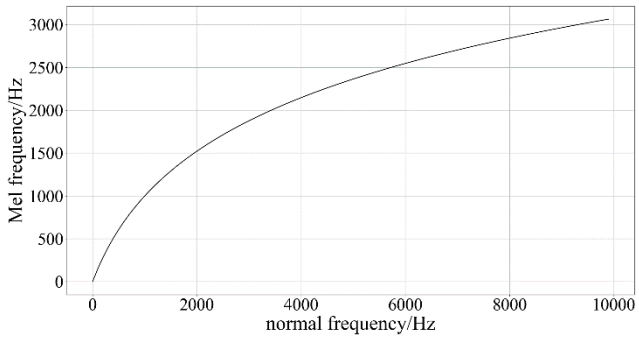

Figure 2. Mel scale

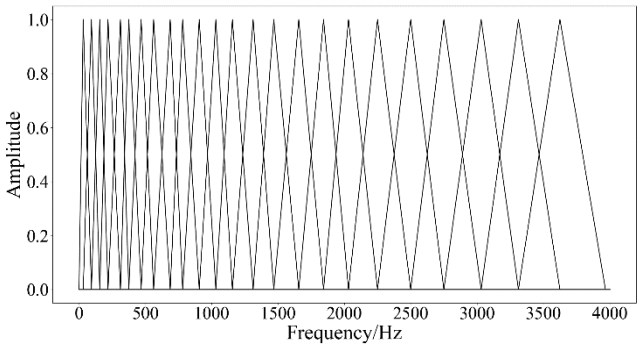

Figure 3. Mel triangular filter bank

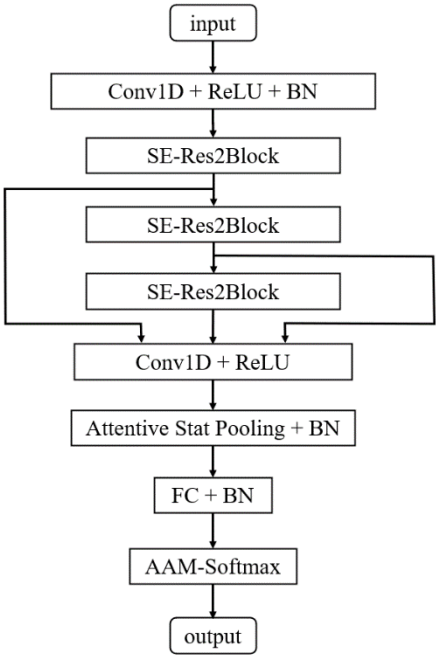

Figure 4. ECAPA-TDNN model architecture

B. *ECAPA-TDNN model architecture*

The architecture of the ECAPA-TDNN model is shown in Fig. 4 ECAPA-TDNN is an architecture based on a time-delay neural network. Its improvements to the x-vector architecture are mainly include the introduction of the SE-Res2Block module, multi-layer feature fusion, and attention statistics pooling (ASP). The core idea of these improvement is to consider the information of multiple frames before and after the current frame, combine the correlation between contextual information, and perform weighted processing on features of different time steps through the self-attention mechanism.

1) *SE-Res2Block*

The architecture of the SE-Res2Block module is shown in Fig. 5, which combines the residual module (Res2Block)[23] with the squeeze-excitation module (SE-Block)[24]. The Res2Block module constructs residual connections between frame-level layers, which can process multi-scale features and increase the receptive field of the network. The SE-Block module consists of two stages: Squeeze and Excitation, which can rescale the frame-level features of each channel according to the global sound properties and enhance the correlation between feature channels.

2) *Multi-layer feature fusion*

Through a convolutional layer, the shallow frame-level features extracted by the first two SE-Res2Block modules are fused with the third frame-level features, which can provide multi-level feature information for the statistical pooling layer, overcoming the shortcoming of only considering deep features in the x-vector architecture.

3) *Attention Statistics Pooling*

The ASP module assigns different weights to each channel according to the importance of each frame-level feature, so that the network can focus more on features that are useful for the current task. By calculating the self-attention score of each

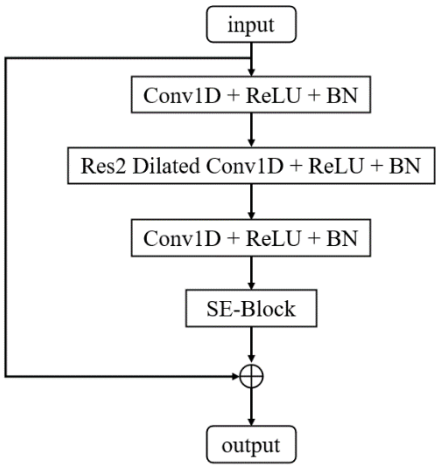

Figure 5. SE-Res2Block architecture

channel, the weighted average and weighted standard deviation of the self-attention scores of each channel over the entire sequence are calculated. Therefore, the final output of the ASP module is the concatenation of the weighted average and weighted standard deviation of each channel along the feature dimension.

IV. EXPERIMENTS

A. *Datasets*

The watermelon variety used in this experiment is "Renfeng Selenium-rich Watermelon" (Fig. 6), and a smartphone is used as the audio collection device. Based on the different planting dates of watermelon batches, and the tasting and scoring results from over ten researchers, watermelon ripeness is defined by farmers and categorized into four levels in the field: fully ripe, nearly ripe, early ripe, and unripe.

1) *Recording audio signals*

One researcher used a smartphone to record the sound 2 cm above the equator of the watermelon, while another researcher held the watermelon and tapped it twice at the equator and navel (Fig. 7). The whole process was repeated twice, so the audio signal of a watermelon contained 8 taps (Fig. 8), and the duration of each audio did not exceed 4 seconds.


Figure 6. Experimental watermelon

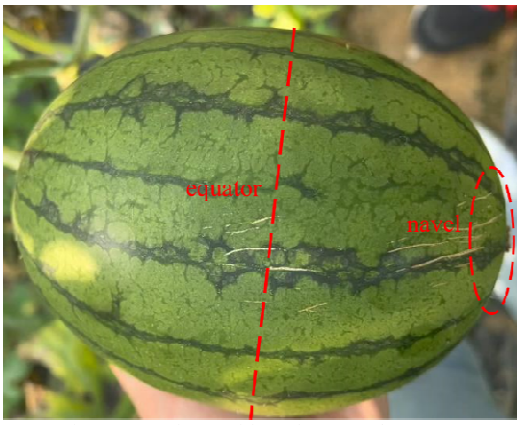

Figure 7. Taping position of watermelon

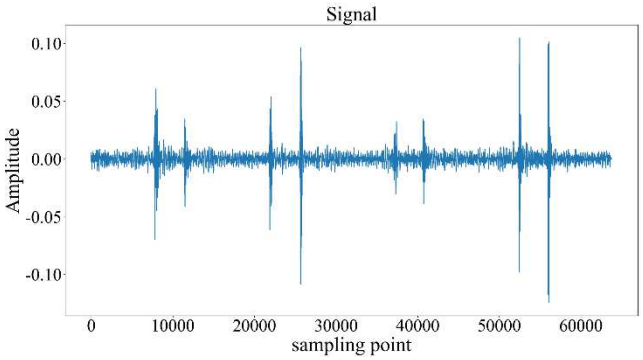

Figure 8. Tap audio waveform

A total of 187 watermelon tapping audio signals were collected in this experiment. Among them, there were 81 tapping audio signals of fully ripe watermelons, 33 tapping audio signals of nearly ripe watermelons, 62 tapping audio signals of early ripe watermelons, and 11 tapping audio signals of unripe watermelons.

2) Data preprocessing

This experiment preprocessed the collected audio data through the following steps:

- Converted the collected audio signal from dual channel to mono channel;
- Standardized the sampling rate to 16000 Hz;
- Stored the audio in *.wav format;
- Performed noise reduction on the processed audio to filter out the friction sound of plant leaves and environmental noise such as vehicles. Fig. 9 is the audio waveform after noise reduction of Fig. 8;
- Standardized the audio duration to 4 seconds by adding silence at the beginning and end;
- The processed audio signals were randomly divided into 149 training set and 38 test set in an 8:2 ratio, as shown in Table 1;
- Datas augmentation was performed to meet the requirements of deep learning by adjusting the volume. This method involved randomly altering the volume by ±15dB. The distribution of the training set after datas enhancement is shown in Table 2.

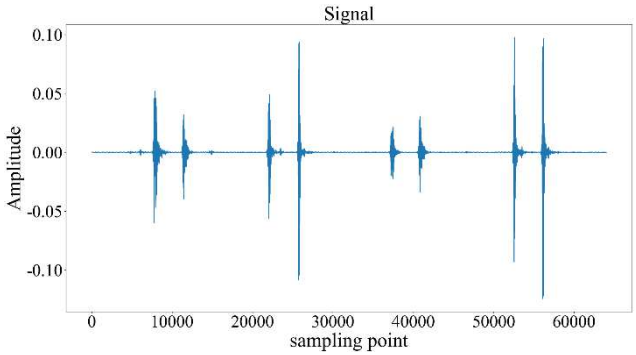

Figure 9. Audio waveform after noise reduction

TABLE 1. TRAIN SET AND TEST SET

Ripeness	Training set	Test set
fully ripe	64	17
nearly ripe	27	6
early ripe	50	12
unripe	8	3
total	149	38

TABLE 2. TRAIN SET BEFORE AND AFTER DATA AUGMENTATION

Ripeness	Before data augmentation	After data augmentation
fully ripe	64	320
nearly ripe	27	270
early ripe	50	300
unripe	8	240
total	149	1130

B. Feature extraction

Using the audio signal in Fig. 9 as an example, its frequency spectrum is shown in Fig. 10 and its Mel spectrogram features is shown in Fig. 11.

C. Experimental parameters

In audio processing tasks, the Hamming window is usually set with a length of 1024 sampling points. Given a sampling rate of 16000Hz and using (7), this corresponds to a time length of 64 milliseconds per Hamming window. To ensure overlap between adjacent windows, the frame shift was set to 320 sampling points. For human auditory perception alignment, the number of Mel Filter Bank was set to 64. The lowest frequency in the triangular filter bank was 50Hz, and the highest was 14000Hz. In model training, BatchSize was set to 256, utilized the cross-entropy loss function. Optimization utilized the Adam optimizer for iterative parameter updates, with a cosine annealing algorithm adjusting the learning rate. The initial learning rate was set to 1e-3 and the weight decay parameter was 6e-6. The experimental environment was Nvidia GeForce RTX 3060 GPU (12G).

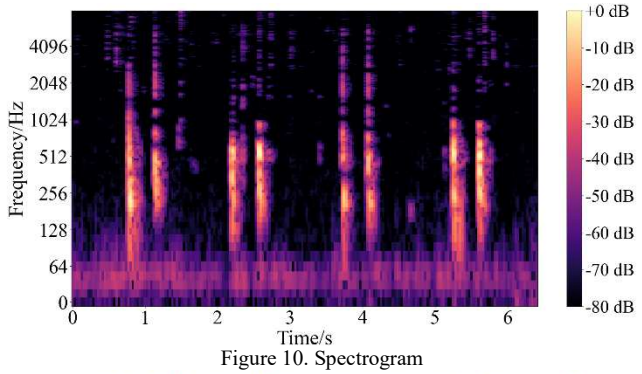

Figure 10. Spectrogram

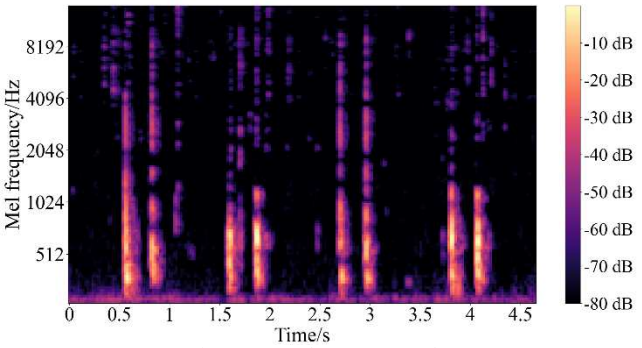

Figure 11. Mel spectrogram features

$$t = \frac{l}{s} \tag{7}$$

Where t is the duration of the audio, s is the sampling rate, and l is the Hamming window length.

D. Evaluation indicators

The performance of the model is usually evaluated using a confusion matrix, which has four possible results: true positives (*TP*), true negatives (*TN*), false positives (*FP*), and false negatives (*FN*). Based on the confusion matrix, four performance metrics can be determined: accuracy, precision, recall, and F1 score as defined in (8-11). Since this experiment is a multi-classification task and the dataset is unevenly distributed, macro-Precision (P_m), macro-Recall (R_m), and macro-F1 ($F1_m$) are used as the evaluation indicators of the model, as defined in (12-14).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F1 = \frac{2 * P * R}{P + R} \tag{11}$$

For the convenience of equation description, the labels, "fully ripe", "nearly ripe", "early ripe", and "unripe" are represented by the numbers 1, 2, 3, and 4, respectively.

$$P_m = \frac{1}{4}\sum_{i=1}^{4} Precision(i) \tag{12}$$

$$R_m = \frac{1}{4}\sum_{i=1}^{4} Recall(i) \tag{13}$$

$$F1_m = \frac{1}{4}\sum_{i=1}^{4} F1(i) \tag{14}$$

E. Comparative Experiment

1) Comparison of different methods of feature extraction

In audio signal processing, frequency domain analysis can provide richer feature information than time domain. A common method is to use FFT to process audio signals and obtain spectral features. Furthermore, by applying Mel Filter Bank generates Mel spectrogram features. Finally, by performing discrete cosine transform on Mel spectrogram coefficients can obtain MFCC (Mel-Frequency Cepstral Coefficients) features.

Deep learning models usually consist of several layers, each of which contains several parameters. The dimension of the input feature will directly affect the size of the first layer weight matrix, and then affect the number of parameters of the overall model. We compared the number of parameters (Table 3) and training results (Table 4) when the input features are Mel spectrogram, spectrogram, and MFCC.

The following conclusions were drawn from the experimental results:

- The size of spectrum features, Mel spectrogram features and MFCC features, gradually decreases, which leads to a gradual decrease in the number of parameters of the ECAPA-TDNN model trained under these three features.
- In terms of accuracy, macro-Precisio, macro-Recall and macro-F1, Mel spectrogram features were superior to the other two features. This also proves that Mel spectrogram features is closer to human hearing and performs better in human-oriented audio tasks.
- The model complexity of the spectrum feature was the highest, but the classification accuracy was the lowest,

TABLE 3. COMPARISON OF THREE FEATURES(MB)

	Mel spectrogram	Spec-trogram	MFCC
Input feature	0.02	0.10	0.01
Model parameter	23.45	25.33	23.21

TABLE 4. MODEL TRAINING RESULTS FOR THREE FEATURES(%)

Feature	accuracy	P_m	R_m	$F1_m$
Mel-spectrogram	89.5	90.0	88.7	88.8
Spectrogram	84.2	84.6	85.2	84.7
MFCC	86.8	82.4	81.9	82.1

which may be due to the extraction of too much redundant information.

- Although MFCC was widely used in the field of audio deep learning, its performance in this experiment was relatively low because the discrete cosine transform is a linear transform that discards some highly nonlinear information in the audio signal, resulting in a decrease in training accuracy.

2) Comparison of different models

We compared the performance of four models, namely SVM[25], CNN[17], x-vector[21], and ECAPA-TDNN[20], on the watermelon ripeness audio classification task using Mel spectrogram features as input (Table 5).

The experimental results showed that when the input feature was Mel spectrogram, the ECAPA-TDNN model still performed better than the SVM model in the watermelon ripeness audio classification task despite the small dataset. Additionally, the ECAPA-TDNN model also demonstrated better performance than the CNN and x-vector models. This advantage is mainly attributed to the attention mechanism, and the more detailed processing of information propagation and aggregation between channels of the ECAPA-TDNN model. These features enable the model to more accurately capture important features and contextual information in the audio signal.

TABLE 5. TRAINING RESULTS OF DIFFERENT MODELS WITH MEL SPECTROGRAM AS INPUT FEATURE(%)

Model	P_m	R_m	$F1_m$	P_m
SVM	63.9	48.8	62.9	52.8
CNN	55.3	51.5	51.0	51.2
x-vector	68.4	51.0	57.5	53.9
ECAPA-TDNN	89.5	90.0	88.7	88.8

V. CONCLUSION

We extracted the Mel spectrogram features of audio signal and used the ECAPA-TDNN model to conduct acoustic nondestructive testing of watermelon ripeness. It successfully achieved accurate judgement of four types of watermelon ripeness: fully ripe, nearly ripe, early ripe, and unripe. While we provide a feasible solution, promoting this method among consumers requires the development of corresponding mobile application for practical testing in real-world applications.

Future work involves the following three aspects:

- It is possible to consider increasing watermelon varieties, using different smartphones to collect audio signals, exploring different model structures and training methods to improve the generalization ability of the model.
- It is also meaningful research to expand the application scope of non-destructive ripeness judgement to other fruits in the agricultural field.
- Combining audio analysis with images analysis is expected to improve the accuracy of ripeness assessment. The texture and fuzz of watermelon can also indicate its ripeness, and consumers can easily take photos of watermelons with their smartphones.

REFERENCES

[1] S. F. Wang, P. Han, G. L. Cui, D. Wang, S. S. Liu, et al. “Near infrared spectroscopy detection of watermelon soluble solids content based on SPXY algorithm,” Spectroscopy and Spectral Analysis, 2019, 39(03): vol. 39, pp. 738-742

[2] G. Jayaprakasha and B. S. J. T. Patil, “A metabolomics approach to identify and quantify the phytochemicals in watermelons by quantitative 1HNMR,” Talanta, 2016, vol. 153, pp. 268-277

[3] D. J. He, Z. W. Li and H. Q. Wang, “Study on the acoustic wave characteristics of watermelon impact sound,” Journal of Northwest A&F University (Natural Science Edition), 1994, pp. 105-107

[4] D. F. Jie and X. Wei, “Review on the recent progress of non-destructive detection technology for internal quality of watermelon,” Computers and Electronics in Agriculture, 2018, vol. 151, pp. 156-164

[5] E. Coffey, “Acoustic resonance testing,” Proc of the 2012 Future of Instrumentation International Workshop Proceedings. Gatlinburg, TN: IEEE, 2012, pp. 1-2

[6] Y. X. Zhang, Y. Zhao, and J. Tao, “Research on non-destructive detection technology of watermelon ripeness based on audio characteristics,” Journal of Hebei Agricultural University, 2011, vol. 34, pp. 114-118

[7] Y. H. Zhang, X. Y. Deng, Z. Xu, and P. Yuan, “Watermelon ripeness detection via extreme learning machine with kernel principal component analysis based on acoustic signals,” International Journal of Pattern Recognition and Artificial Intelligence, 2019, vol. 33, pp. 1951002

[8] A. Alipasandi, A. Mahmoudi, B. Sturm, H. Behfar, and S. Zohrabi, “Application of meta-heuristic feature selection method in low-cost portable device for watermelon classification using signal processing techniques,” Computers and Electronics in Agriculture, 2023, vol. 205, pp. 107578

[9] X. B. Zou, J. J. Zhang, X. W. Huang, K. Y. Zheng, S. B. Wu, et al. “Discrimination of watermelon ripeness based on fusion technology of audio and Near-Infrared spectroscopy,” Transactions of the Chinese Society of Agricultural Engineering, 2019, vol. 35, pp. 301-307

[10] J. H. Mao. “Research on acoustic detection technology and device for watermelon ripeness and internal hollowing,” Hangzhou: Zhejiang University, 2017

[11] K. Xiao, G. D. Gao, G. F. Teng, Y. X. Zhang, and Y. C. Jia, “Non-destructive detection technology of watermelon ripeness based on audio,” Agricultural Mechanization Research, 2009, vol. 31, pp. 150-152+155

[12] X. Chen, P. P. Yuan, and X. Y. Deng, “Watermelon ripeness detection by wavelet multiresolution decomposition of acoustic impulse response signals,” Postharvest Biology and Technology, 2018, vol. 142, pp. 135-141

[13] W. A. Pamungkas and N. Bintoro, “Evaluation of watermelon ripeness using self-developed ripening detector,” Proc of the Earth and Environmental Science. Purwokerto: IOP, 2021, vol. 653, pp. 012020

[14] W. Zeng, X. Huang, S. Müller Arisona, and IV. McLoughlin, “Classifying water- melon ripeness by analysing acoustic signals using mobile devices,” Personal and Ubiquitous Computing, 2014, vol. 18, pp. 1753-1762

[15] K. Chawgien and S. Kiattisin, “Machine learning techniques for classifying the sweetness of watermelon using acoustic signal and image processing,” Computers and Electronics in Agriculture, 2021, vol. 181, pp. 105938

[16] U. Choe, H. Kang, J. Ham, K. Ri, and U. Choe, “Maturity assessment of watermelon by acoustic method,” Scientia Horticulturae, 2022, vol. 293, pp. 110735

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, 1998, vol. 86, pp. 2278-2324

[18] V. Peddinti, D. Povey, and S. Khudanpur, “A time delay neural network architecture for efficient modeling of long temporal contexts,” Proc of the 16th Conference in the annual series of Interspeech events. Dresden: Interspeech, 2015, pp. 3214-3218

[19] D. Albert-Weiß, E. Hajdini, M. Heinrich, and A. Osman, “Cnn for ripeness classfication of watermelon fruits based on acoustic testing,” Proc of the Virtual 3rd International Symposium on Structural Health Monitoring and Non- destructive Testing. Quebeq: eJNDT, pp. 25-26

[20] B. Desplanques, J. Thienpondt, and K. Demuynck, “Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification,” ArXiv, 2020, abs/2005.07143

[21] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2018, pp. 5329-5333.

[22] Y. B. Duan, “Research on robust voiceprint confirmation method based on deep learning,” Shanghai: Shanghai Normal University, 2023

[23] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proc of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016, pp. 770-778

[24] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” Proc of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018, pp. 7132-7141

[25] M. A. Hearst, S. T. Dumais, E. Osuna, and J. Platt, “Support vector machines,” IEEE Intelligent Systems and their applications, 1998, vol. 13, pp. 18-28.