Journal of Creativity Student Vol. 9, No. 3s (2026): Special Edition

Journal of Creativity Student
http://journal.unnes.ac.id/journals/jcs
DOI: https://doi.org/10.15294/jcs.v9i3s.63713

Design of a Vocal Quality Ranking System Using the P.YIN Algorithm Based on Python 3

Agel Bayu Samudro*, Luki Utomo, Joko Tri Susilo, Irawati, Aripin Triyanto

Electrical Engineering Study Program, Faculty of Engineering, Universitas Pamulang, Tangerang Selatan, Indonesia

*Corresponding Author: agel.bayu.samudro@gmail.com

Abstract
In the modern era of the music industry and vocal education, consistent practice is crucial to improving voice quality. However, the vocal evaluation process in recording studio environments is often subjective and not fully supported by objective and measurable evaluation software. This study aims to design and implement a vocal quality rating system called UtaSong based on PYTHON 3. The system utilizes the P.YIN (Probabilistic YIN) algorithm as the core engine for fundamental frequency (F0) extraction, implemented on a Wearnes AW-NE38H Mini Computer. The testing method is carried out by calculating the pitch deviation between the user's vocal and the reference (MIDI) in cent units, which is then automatically converted into a scoring system and grading. The test results indicate that the P.YIN algorithm successfully detects pitch consistently. Acoustic testing proves that a quiet environment (average noise of 38 dB) produces a much more accurate evaluation compared to noisy conditions (65 dB), where background noise affects the pitch deviation rate. In terms of computational performance, the hardware is functionally operational at the minimum threshold, with RAM utilization reaching 87–90% and GPU 98–99%. Therefore, increasing the RAM capacity to at least 8 GB and using a Solid State Drive (SSD) is highly recommended to optimize latency and overall system performance.

Keywords: *p.yin algorithm, pitch detection, python 3, recording studio, utasong, vocal evaluation*

INTRODUCTION

In the modern era of the music industry and vocal education, consistent practice and evaluation are essential for improving a singer's voice quality, breathing techniques, articulation, and pitch accuracy. Recording studio environments and vocal training rooms serve as crucial spaces where this refinement of vocal quality occurs. During music production or professional practice, singers are required to possess precise pitch control that harmonizes with the accompanying instrumental structure. The singing voice is considered the most expressive and complex musical instrument. Singing and expressing emotions are closely intertwined, making it clearly distinguishable when a singer performs in a sad, joyful, soft, or aggressive manner. However, maintaining objectivity in assessing vocal quality and pitch accuracy often presents a unique challenge. Assessments relying solely on human hearing, such as those by music producers or vocal coaches, are sometimes influenced by subjectivity and are difficult to measure or evaluate quantitatively.

To overcome this subjectivity, the utilization of Digital Signal Processing (DSP) technology in audio software becomes highly essential. This technology enables the mathematical analysis of a user's voice characteristics to identify areas requiring improvement, such as pitch accuracy and vocal stability. Among various audio processing methods, the observation and extraction of a singer's pitch constitute the most effective approach to constructing an automatic and accurate vocal quality assessment system. The implementation of a ranking system or real-time scoring provides immediate feedback, allowing singers to instantly evaluate the strengths and weaknesses of their performance. This encourages the habituation of hearing and analyzing one's own voice (self-monitoring), which is a vital component in advanced vocal training. Unfortunately, software or computer systems

specifically designed as vocal evaluation tools with precise pitch deviation measurement metrics remain highly limited in availability within local recording studios in Indonesia. The majority of facilities still rely on manual, intuition-based auditory evaluations or depend on less efficient external hardware.

Based on this background, the problem formulations of this study are stated as follows: 1) how to design and construct a vocal quality ranking system featuring real-time scoring and direct comparison to function as an objective and measurable evaluation tool in recording studio environments; 2) what is the accuracy level of the P.YIN algorithm in extracting the fundamental frequency (F_0) from a singer's voice, and how is the pitch deviation calculated against a MIDI-based reference measured in cents; and 3) how do various environmental acoustic conditions, particularly noise levels, affect the sensitivity of the P.YIN algorithm, and what is the extent of its computational efficiency regarding hardware resource utilization, including CPU and RAM.

METHOD

This study employed a quantitative approach utilizing laboratory experimental and software engineering methods. The quantitative approach was applied to obtain and analyze data measurably, specifically in measuring the fundamental frequency (F_0) detection accuracy using the P.YIN algorithm, calculating the pitch deviation against a reference pitch in cents, and numerically measuring the hardware resource utilization rate. The laboratory experimental method was utilized to test the system's performance under controlled conditions, whereas the software engineering method was applied during the design, development, implementation, and testing processes of the vocal quality ranking system.

Data collection and preparation were conducted as the initial stage of the research to provide the necessary data for the system development and testing processes. Quantitative data were acquired from a recording studio environment employing two primary scenarios, namely reference data and test data. The reference data consisted of instrumental audio files and MIDI files representing the ideal pitch frequency ranges, which served as the pitch ground truth. The real-time test data comprised vocal audio signals captured directly through a computer microphone while the user sang along to an accompanying backing track. These audio signals were subsequently converted into numerical array data based on each time frame for system processing. Additionally, the study utilized test data in the form of user vocal recording files in .wav or .mp3 formats, which were uploaded into the system for analysis following the completion of the studio recording process.

The system design stage was conducted to determine the software architecture, the data processing flow, and the interaction mechanisms between the user and the system. The system was developed using Python 3 and comprised several core components. On the signal processing backend, the Librosa and NumPy libraries were integrated to support audio signal processing and probability-based pitch extraction. Furthermore, a comparative logic was designed by implementing a time-alignment algorithm, ensuring that both real-time and uploaded vocal data could be precisely compared against the original melody at identical time positions. On the interface side, a desktop web-based UI/UX was designed to provide interactive visualizations, such as spectrograms and pitch movements, allowing analysis results and evaluation scores to be displayed informatively to the user.

The algorithm implementation and quantitative computation stage involved the translation of the system design into program code using Python 3. At this stage, the P.YIN algorithm was implemented to estimate the fundamental frequency (F_0) of the vocal signals. The computational process included the application of the Cumulative Mean Normalized Difference Function (CMNDF) to enhance the precision of fundamental period estimation and reduce the likelihood of octave errors. Subsequently, a Hidden Markov Model (HMM) and Viterbi Decoding were applied to determine the most consistent pitch estimation path over time. The implementation of these methods aimed to maintain pitch detection continuity when the vocal signal experienced interference caused by background noise or the presence of accompanying instrumental sounds.

The data analysis and testing stage was conducted to evaluate the system's performance based on predetermined numerical parameters. The testing was focused on four primary aspects: pitch detection accuracy testing, scoring accuracy testing, acoustic sensitivity testing, and computational performance testing. The pitch detection accuracy testing was performed to ascertain the reliability of the P.YIN algorithm in tracking vocal pitches, both in real-time recording mode and Direct Comparison mode via file uploads. The scoring accuracy testing was conducted by comparing the pitch

deviation values measured in cents against the vocal quality ranking outcomes generated by the system across categories A, B, C, D, and S. Furthermore, the acoustic sensitivity testing was carried out by measuring the voiced frame reading performance under different environmental conditions, specifically a quiet room condition (Quiet Whisper) with an average noise level of 38 dB, and an environment with a higher noise level (Quiet Park) averaging 65 dB. Meanwhile, the computational performance testing was executed to determine the system's processing load on the hardware—a Wearnes AW-NE38H Minicomputer—by monitoring CPU utilization rates, memory (RAM) consumption, storage (HDD) activity, and system processing latency.

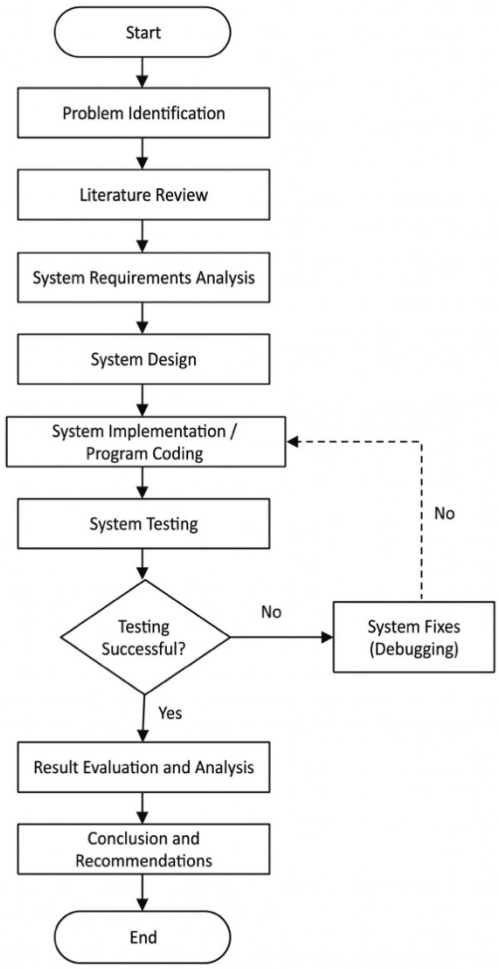

Figure 1. Research Flowchart

RESULTS AND DISCUSSION

The Direct Comparison mode testing was conducted to examine the capability of the UtaSong system in processing and analyzing vocal audio files uploaded directly by users via the Upload feature. This mode was designed to provide vocal performance evaluation in a recording studio environment, where the vocal take process is performed separately and uploaded in .mp3 format for a comparative analysis against a reference pitch. In this test, data were acquired from three user subjects with varying vocal ability backgrounds, namely: Amateur (lacking vocal training fundamentals), Standard (possessing basic singing experience/hobbyist), and Professional (trained vocalist). The three subjects sang the identical song, "Ado – Usseewa," to assess how the system responds and ranks varying levels of pitch precision. The tolerance threshold (*Tolerance*) was kept constant at 200¢ (200 cents). The system processed the uploaded audio files utilizing the P.YIN algorithm as the core engine for fundamental frequency (F_0) extraction. The evaluation method involved calculating the average pitch deviation (*Mean Deviation*) between the user's vocals and the MIDI reference in cents, which was subsequently converted into an automated scoring and ranking system.

Table 1. Specifications of minimum requirements to run the program

Parameter	Subject 1 (Amateur)	Subject 2 (Standard)	Subject 3 (Professional)
Test Song	Ado – Usseewa	Ado – Usseewa	Ado – Usseewa
Format & Sample Rate	.mp3 (11,025 Hz)	.mp3 (11,025 Hz)	.mp3 (11,025 Hz)
Total User Frames	3,512 frames	3,510 frames	3,515 frames
Voiced Frames (Detected F_0)	1,840 frames	2,650 frames	3,120 frames
Valid Comparison Frames	1,840 frames	2,650 frames	3,120 frames
Mean Deviation	245.60 cents	110.35 cents	32.15 cents
Final Score	45.20 / 100	78.50 / 100	96.80 / 100
Grade	D	B	S
Processing Time	115.2 seconds	118.4 seconds	121.5 seconds

Based on the comparative test results in Table 1, a highly apparent mathematical correlation exists between the subject's vocal skill level and the extracted data generated by the P.YIN algorithm. In a high-tempo song (140-150 BPM) requiring rapid syllable transitions, the Professional subject was capable of producing 3,120 voiced frames (approaching the total overall frames). This indicates highly stable breathing and articulation techniques, enabling the P.YIN algorithm to detect the fundamental frequency (F_0) continuously without substantial interruptions. Conversely, the Amateur subject only generated 1,840 voiced frames, indicating numerous unstable notes, failure to meet the P.YIN detection *confidence threshold*, or vocal volumes overshadowed during rapid lyrical phrases.

Deviation measurements in cents substantiated the system's objectivity. The Professional subject recorded a remarkably minimal deviation (32.15 cents), well below the 200-cent tolerance threshold. The Standard subject exhibited a moderate deviation (110.35 cents), whereas the Amateur subject deviated significantly up to 245.60 cents, which musically signifies the singing was out of tune by more than two semitones (piano keys) from the original melody. The logarithmic penalty system successfully differentiated quality proportionally. The minimal deviation in the Professional subject was converted into a Score of 96.80 and awarded the highest rank, Grade S. The Standard subject obtained Grade B (Score 78.50), while the Amateur subject received Grade D (Score 45.20) due to substantial penalty deductions resulting from a high Mean Deviation.

The data processing time tended to be slightly longer for the Professional subject (121.5 seconds) compared to the Amateur subject (115.2 seconds). This is computationally reasonable because the higher the number of successfully detected voiced frames, the greater the number of data points that must be calculated, compared against the reference MIDI, and time-aligned by the system. Overall, this test concluded that the P.YIN algorithm within the UtaSong system is capable of measurably differentiating a singer's proficiency level within a recording studio environment. The system successfully executed its evaluation function objectively by analyzing pitch stability and converting it into an accurate score based on the uploaded audio file format. In the conducted research, test results were obtained, including tests on the Hardware circuit; the UtaSong karaoke program acquired data from Pitch Detection, rhythm, beats, chord progressions, melodies, arrangements, and compositions to evaluate a song, alongside several Pitch values in its data processing using the P.YIN Algorithm as the basis for assessing vocal quality within the system.

Pitch Detection Algorithm (P.YIN) Testing

The P.YIN pitch detection algorithm testing was conducted using real-time audio data acquired from a live karaoke session featuring the song "Chase – Batta." In this test, a user was instructed to sing the song using the UtaSong system's hardware under imperfect singing conditions, wherein numerous notes were off-pitch and deviated significantly from the reference pitch, aiming to test the P.YIN algorithm's capability to detect various Pitch variations realistically and consistently.

The testing process commenced when the user sang the song through an Advance Professional Double UHF Wireless Microphone (B); the audio signal was subsequently forwarded to a Wearnes AW-NE38H Core i3-4130 Mini Computer with 4GB DDR3 RAM (D) for direct processing by the P.YIN algorithm. The analytical process results were displayed visually via a BenQ G615 HDPL LED Monitor (A), while the musical accompaniment was output through a Robot RB300 BT 5.1 Karaoke Speaker (C). A Keyboard peripheral (E) was utilized to support the system's operations throughout the testing session. Based on the results displayed by the system, the analytical process proceeded through several sequentially recorded stages as follows:

Audio Loaded — The system successfully loaded the user's audio file with a size of 318.4s at a sample rate of 11,025 Hz, signifying that the audio input from the wireless microphone was successfully received and read into the system's memory properly.

Silence Trimming Skipped — The Silence Trimming process was bypassed by the system with a total user frame count of ±3,396 frames, wherein the system retained the initial recording point intact to maintain synchronization between the user's audio and the reference audio.

Running P.YIN on User Vocal — The P.YIN algorithm was executed directly on the user's vocal data, while the reference Pitch was retrieved from a previously stored system cache (*Fetching Reference Pitch from Cache*), ensuring the comparative process could be performed efficiently without necessitating the reprocessing of the reference audio.

Pitch Detection Complete — The pitch detection process was successfully concluded with the following results: the user's audio produced 3,397 frames with 871 voiced frames (frames detected as containing pitch), whereas the reference audio produced 1,968 frames with 957 voiced frames. These data indicate that the P.YIN algorithm successfully identified voiced segments accurately from the total processed audio frames.

Score Calculated — The system calculated a score based on 957 valid frames resulting from the matching between the user's vocal and the reference, yielding a Mean Deviation value of 282.41 cents. This high Mean Deviation value directly reflects the substantial average pitch deviation committed by the user while singing.

Analysis Complete — The entire analytical process was successfully completed with a total processing time of 50.7 seconds, signifying that the system is capable of executing the complete pitch detection and comparison process on the Wearnes Mini Computer hardware.

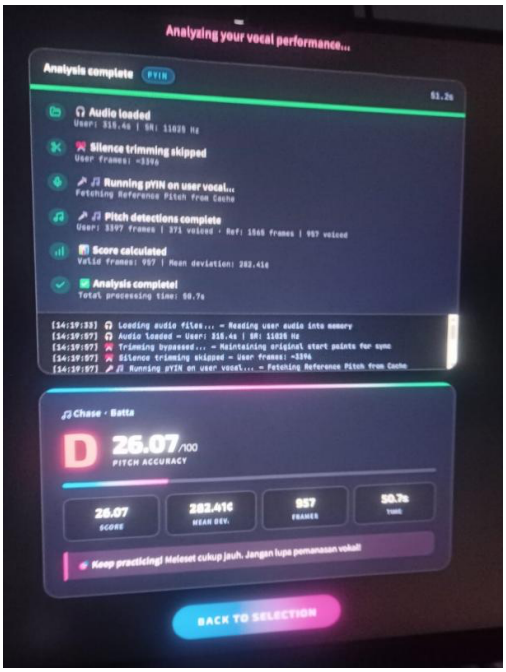

Figure 2. Real-time Evaluation Display of "Chase – batta" Karaoke

The final test results demonstrated that the system awarded a Pitch Accuracy score of 26.07 out of 100 with a D ranking, which reflects the user's singing condition missing significantly from the reference pitch. The system automatically displayed a feedback message to the user stating "Keep practicing! Meleset cukup jauh. Jangan lupa pemanasan vokal!" (*Keep practicing! Missed quite far. Don't forget vocal warm-ups!*), signifying that the score-based evaluation feature operated as designed. Table 2 below summarizes the technical data of the P.YIN algorithm test results obtained from the live Karaoke session:

Table 2. Singing test table under off-pitch conditions to test the sensitivity accuracy of the P.YIN algorithm

Parameter	Value
Tested Song	Chase – Batta
User Audio Size	318.4s
Sample Rate	11,025 Hz
Total User Frames	3,397 frames
User Voiced Frames	871 frames
Total Reference Frames	1,968 frames
Reference Voiced Frames	957 frames
Valid Comparison Frames	957 frames
Mean Deviation	282.41 cents
Pitch Accuracy Score	26.07 / 100
Grade	D
Total Processing Time	50.7 seconds

The comparative accuracy and scoring system testing was conducted to measure the capability of the UtaSong system in generating accurate and consistent scores across various differing environmental testing conditions. The primary focus in this subchapter was to test the sensitivity of the P.YIN algorithm toward noise (background noise), specifically by comparing the score results obtained by a user singing in a quiet environmental condition versus a noisy environmental condition with a television turned on nearby. To reinforce the testing's validity, the room's acoustic condition in each scenario was objectively measured using a Sound Meter application on a smartphone device to acquire noise level data in decibels (dB).

Room Acoustic Condition Measurement

Prior to the commencement of the karaoke session, the room's acoustic condition was initially measured utilizing the Sound Meter application for 60 seconds to obtain a representative overview of the testing environment's noise level. Table 3 summarizes the Sound Meter measurement results for each environmental condition.

Table 3. Singing experiment during quiet conditions at 38 dB noise

Acoustic Parameter	Scenario 1 (Quiet)	Scenario 2 (With Noise/TV)
Current dB Level	34 dB	42 dB
Minimum Value	30 dB	42 dB
Average Value	38 dB	65 dB
Maximum Value	93 dB	79 dB
Condition Classification	Quiet Whisper	Quiet Park

In Scenario 1 (Quiet), the room's average noise level was at 38 dB classified as "Quiet Whisper," which is equivalent to a highly tranquil room condition resembling a soft whisper noise level or a library room. The Sound Meter waveform graph pattern demonstrated waves consistently moving within the 30–40 dB range throughout the measurement duration, with only a few momentary transient spikes that did not represent the overall background condition.

In Scenario 2 (With Noise/TV), the room's average noise level increased significantly to 65 dB classified as "Quiet Park," which is equivalent to a room condition with normal conversation or an active television sound. The Sound Meter waveform graph in this scenario displayed a wave pattern dominantly and consistently remaining within the 60–70 dB range throughout the measurement duration, indicating the presence of continuous and uninterrupted background noise during the karaoke session. The recorded minimum value of 42 dB demonstrated that even at the quietest moments, the noise level in this scenario remained much higher compared to the average condition of Scenario 1.

Testing Scenarios

The testing was conducted under two different environmental condition scenarios but utilized the identical user with equivalent singing style and capabilities, ensuring that the differentiating variable was solely the environmental acoustic condition, not differences in the user's vocal performance. Scenario 1 — Noisy Environment Condition (High Noise): The user sang the song "Chase

– Batta" in a noisy environmental condition with a television turned on around the testing room. The background sound from the television potentially entered the capture of the Advance Professional Double UHF Wireless Microphone (B) and mixed with the user's vocal signal, potentially affecting the quality of the audio data received by the Wearnes AW-NE38H Mini Computer (D) for processing by the P.YIN algorithm.

Scenario 2 — Quiet Environment Condition (Low Noise): The user sang the song "Blow My Gale – Uma Musume" in a relatively quiet environmental condition without significant background noise interference, thereby allowing the microphone to capture the user's vocal sound more cleanly and focused. Table 4 below presents a comparison of the technical data obtained from both testing scenarios:

Table 4. Singing experiment during quiet conditions at 38 dB noise

Parameter	Scenario 1 (Noise 65 dB / TV On)	Scenario 2 (Quiet 38 dB)
Song	Chase – Batta	Blow My Gale – Uma Musume
Acoustic Classification	Quiet Park	Quiet Whisper
Avg Room Noise	65 dB	38 dB
User Audio Size	314.9s	302.4s
Sample Rate	11,025 Hz	11,025 Hz
Total User Frames	3,391 frames	3,257 frames
Reference Voiced Frames	1,565 frames	1,070 frames
Valid Comparison Frames	957 frames	1,070 frames
Mean Deviation	301.9 cents	287.78 cents
Pitch Accuracy Score	54.30 / 100	63.44 / 100
Grade	C	C
Total Processing Time	67.1 seconds	54.0 seconds
System Message	"Getting there. Meleset 301,9¢ — jangan ragu-ragu pas nyanyi."	"Getting there. Meleset cukup jauh. Jangan lupa pemanasan vokal!"

Overall, this test concluded that the P.YIN algorithm-based UtaSong system was capable of operating under both environmental conditions without experiencing failures or system errors. Nevertheless, environmental acoustic conditions demonstrably affected three primary system aspects, namely the number of detected voiced frames, Pitch estimation accuracy (*Mean Deviation*), and processing time. Noise at the 65 dB level (Quiet Park) caused noise to be co-detected as voiced signals, increasing the Mean Deviation, and prolonging computational time. Meanwhile, quiet conditions at 38 dB (Quiet Whisper) produced cleaner detection, a higher number of valid frames, and more efficient processing times. Based on all these findings, the utilization of the UtaSong system is highly recommended in room conditions with noise levels below 40 dB, or equivalent to a Quiet Whisper classification, to ensure the P.YIN algorithm can operate optimally and yield the most accurate and reliable vocal performance evaluations.

CONCLUSION

Based on the research findings, it can be concluded that the UtaSong system was successfully constructed as an objective vocal quality evaluation tool through signal processing utilizing a PYTHON 3-based P.YIN algorithm featuring real-time scoring and direct comparison capabilities. The system logarithmically calculates the pitch deviation between the user's voice frequency and a reference pitch in cents, then converts it into a proportional penalty system to generate vocal quality rankings in S, A, B, C, and D categories. The test results demonstrated that the P.YIN algorithm exhibits high sensitivity toward environmental acoustic conditions, particularly noise levels, wherein room conditions with an average noise of 65 dB caused background sounds to be co-detected as voiced frames, thereby significantly increasing the Mean Deviation up to 301.9 cents and prolonging computational time. Based on these results, the UtaSong system operates more optimally in relatively quiet studio conditions with noise levels below 40 dB. From a hardware performance perspective, the system could run functionally without errors, but the P.YIN algorithm imposed a relatively high computational load on devices with the minimum specifications of an Intel Core i3 and 4 GB RAM, indicated by a RAM utilization of 87–90%, GPU of 98–99%, and a result processing time ranging from 50 to 116 seconds depending on the analyzed audio duration. Therefore, subsequent developments are recommended to

upgrade hardware specifications, notably increasing the minimum RAM capacity to 8 GB and utilizing SSD storage media, to mitigate potential bottlenecks in the audio processing pipeline. In addition to hardware upgrades, the system also needs to be developed with a noise reduction module via noise gate methods or vocal isolation filters, along with PYTHON code optimization utilizing approaches such as multithreading or restricting the pitch extraction buffer size to decrease processing time and enhance feedback response. Furthermore, the UI/UX interface is recommended to be developed with more informative data visualizations, such as spectrogram graphs and detailed pitch comparisons, while maintaining its concept as a professional monitoring tool and avoiding an overly game-like appearance.

DECLARATION OF GENERATIVE AI

During the preparation of this work, the author(s) used Grammarly to assist with paraphrasing and language refinement throughout the article and Google Gemini to generate Figure 1. After using these tools, the author(s) carefully reviewed and edited the content and figure as needed and take(s) full responsibility for the accuracy, integrity, and originality of the content of the publication.

CONFLICTING INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING INFORMATION STATEMENT

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

AUTHORSHIP CONTRIBUTION STATEMENT

Agel Bayu Samudro: Conceptualization, Methodology, Writing - Original Draft. **Luki Utomo**: Data Curation, Investigation, Validation. **Joko Tri Susilo**: Methodology, Formal Analysis, Writing - Review & Editing. **Irawati**: Supervision, Resources, Validation. **Aripin Triyanto**: Supervision, Writing - Review & Editing, Project Administration.

REFERENCES

Anisah, P., Lestari, B., Anggreini, A., Fajriani, R., & Assaidah, A. (2023). Analisis Filter Digital pada Variasi Pengolah Suara Aplikasi TikTok. *Jurnal Penelitian Sains, 25*(1), 29–35.

Cui, C., Wang, J., Wu, Z., Ni, J., & Qian, T. (2016). The Socio-Spatial Distribution of Leisure Venues: A Case Study of Karaoke Bars in Nanjing, China. *ISPRS International Journal of Geo-Information, 5*(9), 150.

Grinstein, E., Tengan, E., Çakmak, B., Dietzen, T., Nunes, L., van Waterschoot, T., Brookes, M., & Naylor, P. A. (2024). Steered Response Power for Sound Source Localization: A Tutorial Review. *EURASIP Journal on Audio, Speech, and Music Processing, 2024*, 24.

Hasian, I., & Rachma, A. (2019). *Perancangan dan Implementasi Sistem Informasi Olah Vokal Menggunakan Metode PSOLA* [Skripsi, Sekolah Tinggi Manajemen Informatika dan Komputer Indo Daya Suvana].

Kim, J. W., Salamon, J., Li, P., & Bello, J. P. (2018). *CREPE: A Convolutional Representation for Pitch Estimation*. arXiv.

Kumar, A., Priya, Kumar, P., & Haque, M. A. (2023). An Exploration of Human-Computer Interaction and User-Centric Design Principles. *Journal of Propulsion Technology, 44*(5), 1832–1840.

Lindblom, B., & Sundberg, J. (2014). The Human Voice inAnisah, P., Lestari, B., Anggreini, A., Fajriani, R., & Assaidah, A. (2023). Analisis filter digital pada variasi pengolah suara aplikasi TikTok. *Jurnal Penelitian Sains*, *25*(1), 29.

Cui, C., Wang, J., Wu, Z., Ni, J., & Qian, T. (2016). The socio-spatial distribution of leisure venues: A case study of karaoke bars in Nanjing, China. *ISPRS International Journal of Geo-Information*, *5*(9).

Grinstein, E., Tengan, E., Çakmak, B., Dietzen, T., Nunes, L., van Waterschoot, T., Brookes, M., & Naylor, P. A. (2024). *Steered response power for sound source localization: A tutorial review*.

Hasian, I., & Rachma, A. (2019). *Irene Hasian dan Azhari perancangan dan implementasi sistem informasi olah vocal menggunakan metode PSOLA*. Sekolah Tinggi Manajemen Informatika dan Komputer Indo Daya Suvana.

Kim, J. W., Salamon, J., Li, P., & Bello, J. P. (2018). *CREPE: A convolutional representation for pitch estimation*. arXiv.

Kumar, A., Priya, Kumar, P., & Haque, M. A. (2023). An exploration of human-computer interaction and user-centric design principles. *Journal of Propulsion Technology*, *44*(5).

Lindblom, B., & Sundberg, J. (2014). The human voice in speech and singing. In *Springer handbook of speech processing* (pp. 703–746). Springer.

Lucas, C. R., Debnath, D., & Hines, A. (2024). *Improving long-term F0 representation using post-processing techniques*. GitHub.

Mahmudi, R., Bahasa, F., & Seni, D. (2023). Karya lagu Dewa 19 salam tinjauan aransemen (ciri progresi akor yang mendominasi). *Jurnal Seni Musik*, *3*(2).

Mauch, M., & Dixon, S. (2014). *pYIN: A fundamental frequency estimator using probabilistic threshold distributions*. Sound Software.

Mayor, O., Bonada, J., & Loscos, A. (2009). *Performance analysis and scoring of the singing voice*. ResearchGate.

Oppenheim, A. V., & Schafer, R. W. (2010). *Discrete-time signal processing* (3rd ed.). Prentice Hall.

Oxenham, A. J. (2013). Revisiting place and temporal theories of pitch. *Acoustical Science and Technology*, *34*(6), 388–396.

Prasetiyo, A. (2020). *Analisis formulasi ritme lagu pada buku ajar Suzuki violin method vol. I*.

Rahman, H. N. (2015). *Peningkatan kemampuan membaca ritme melalui metode Takadimi-Orff*.

Rossing, T. D., Moore, F. R., & Wheeler, P. A. (2002). *The science of sound* (3rd ed.). Addison Wesley.

Saputra, K. (2026). Distortion of Sampelong: Komposisi metal berdasarkan logu Sampelong dan struktur sonata form. *Jurnal Penelitian Nusantara*, *2*(2), 328–333.

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5), 293–302.

Zhang, M. (2014). *A real time karaoke scoring system based on pitch detection*. Semantic Scholar.