Query-by-Humming: A Comprehensive Literature Survey

Gopala
Assistant Professor
Department of Computer Science
Government Engineering College, Huvina Hadagali, Karnataka, India

Nagappa U Bhajantri
Professor
Department of Computer Science
Government Engineering College, K.R. Pete, Karnataka, India

Abstract

Query-by-Humming (QBH) systems enable users to retrieve music by humming short melodic fragments. This survey consolidates developments from foundational systems to modern deep learning models, covering melody extraction, matching techniques, datasets, system optimization, commercial applications, and future trends. With over 50 key contributions reviewed, we present a holistic understanding of the QBH domain and highlight challenges and future directions.

Keywords: Query by humming, Music information retrieval, Dynamic time warping, Convolution neural network, mean reciprocal rank.

1. **Introduction**

The rapid expansion of digital music libraries, fueled by the rise of streaming platforms and cloud-based storage, has created a growing demand for intuitive and user-friendly music retrieval systems. Traditional search methods that rely on metadata—such as song titles, artist names, or lyrics—are often insufficient when users are unable to recall this specific information. In such scenarios, Query-by-Humming (QBH) systems present a compelling alternative by allowing users to search for music using brief sung or hummed melodic fragments. These systems function by extracting melodic features—most notably the fundamental frequency (F0) and pitch contours—from the user's vocal input, then matching this information against a precompiled database of melodies to identify potential song candidates [12][13][15].

A key strength of QBH systems lies in their robustness to variations in tempo, pitch, rhythm, and vocal timbre, making them especially accessible for casual users and well-suited for natural, flexible human-computer interaction within music information retrieval (MIR). Early research in QBH provided foundational insights into melody extraction, alignment techniques, and similarity measures, paving the way for advanced systems capable of handling polyphonic audio, leveraging machine learning models, and supporting real-time query processing with improved accuracy and responsiveness.

2. **Evolution of QBH Systems**

2.1 **Foundational Stage (1995–2005)**

The work in [12] introduced QBH by leveraging pitch contours and string alignment. The system translated user-hummed queries into pitch sequences and used edit distance to match them against a melody database, laying the foundational principles for symbolic music retrieval.

Building upon this, [13] proposed MELDEX, a system that incorporated not only pitch but also interval and rhythmic features, enhancing the system's robustness to variations in tempo and key introduced by different users. This multi-dimensional approach aimed to improve matching accuracy.

Later, [15] introduced the MUSART testbed, which, although published after the foundational period, played a crucial role by offering standardized evaluation methods for QBH systems. MUSART helped address reproducibility issues in music information retrieval research and enabled consistent comparative benchmarking of different approaches.

2.2 **Pre-Deep Learning Improvements (2006–2015)**

Dynamic Time Warping (DTW) and its optimized variants such as FastDTW emerged as core techniques during this era, enabling efficient alignment of variable-length pitch sequences and greatly enhancing the system's resilience to tempo and rhythm variations between the query and target audio [26]. In parallel, musical parameter-based warping strategies were investigated to improve accuracy and retrieval effectiveness, as shown in [30].

To overcome limitations inherent to alignment-based matching, entropy-based methods were introduced to quantify the informational content of queries and refine similarity measures [4][5]. These innovations, along with subsequence matching techniques, contributed to more robust retrieval capabilities, particularly when dealing with incomplete or noisy user inputs.

Moreover, the foundational principles of Hidden Markov Models (HMMs) continued to play a significant role in QBH development, with [27] providing probabilistic modeling of melodic transitions that improved recognition under uncertain or variable conditions.

This phase also marked a shift toward improving computational efficiency and scalability, laying the groundwork for the deep learning-based advancements that would follow in the next decade.

2.3 **Rise of Deep Learning (2016–Present)**

Deep learning has transformed Query-by-Humming (QBH) systems by introducing powerful neural architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and, more recently, Transformers. These models have significantly enhanced feature extraction capabilities by learning hierarchical representations of audio signals directly from raw or preprocessed inputs, outperforming traditional handcrafted approaches [41][42][43][44][46][48]. CNN-based architectures, as demonstrated in [41], showed strong performance in capturing pitch and timbral nuances, while CRNNs effectively incorporated temporal context. Further refinement was achieved through deep models for singing transcription [46], which led to more accurate melody contour extraction. Pretrained Audio Neural Networks (PANNs), introduced in [48], offered robust generalization across multiple tasks, including QBH.

Moreover, hybrid models have gained traction by combining metadata (e.g., artist, genre) with audio features to improve retrieval precision. These systems often implement progressive filtering—first narrowing the search space using metadata before applying more intensive audio-based matching [6][25][47]. This two-stage strategy enhances scalability and accuracy, particularly in large-scale music databases. An example is provided in [47], where multiple feature types were fused using a voting mechanism, resulting in improved retrieval robustness. These hybrid and context-aware approaches highlight the evolution toward more intelligent and adaptive QBH systems.

3. **Melody Extraction Techniques**

Melody extraction refers to the process of converting a user's sung or hummed input into a symbolic or machine-readable melodic representation. Early techniques relied predominantly on handcrafted features that focused on harmonic and temporal aspects of the audio signal.

Initial studies such as [28] leveraged chroma features to capture harmonic content, enabling effective representation of pitch classes for melody identification and cover song similarity. Building on this, [29] proposed binary chroma similarity combined with local alignment, which improved robustness to variations in key and tempo, making it more tolerant of noisy or imperfect user inputs. Further refinement was presented in [31], where pitch contour features were utilized to preserve the shape and temporal dynamics of melodies, offering improved performance in vocal queries.

With the emergence of deep learning, melody extraction became increasingly data-driven. In [1], Total

Variation Regularization was introduced to suppress background noise and highlight structural melodic boundaries, enhancing clarity in humming signals. Deep learning architectures like HumTrans [3] employed large-scale humming datasets to train transcription models capable of capturing fine-grained pitch variations, thus boosting accuracy and adaptability. The DeepSinging model proposed in [46] provided a specialized neural framework for singing voice transcription, significantly outperforming traditional approaches by leveraging deep representations.

Parallel to these, denoising and segmentation techniques have also played a crucial role. Early work in [14] emphasized encoding timing information and proposed simple denoising strategies to better align rhythmic structures. In [24], discriminative models were used for transcription, which inherently provided noise resilience and improved polyphonic handling. Additionally, [35] introduced methods combining onset detection with segmentation strategies, allowing for more accurate isolation of melodic phrases in noisy or complex audio.

Together, these handcrafted, data-driven, and signal-processing-based approaches have significantly advanced the precision, adaptability, and robustness of melody extraction in modern Query-by-Humming systems.

4. **Matching Algorithms**

Efficient and accurate melody matching forms the backbone of any Query-by-Humming (QBH) system, as it enables the comparison between the user’s hummed input and stored musical representations. Over the years, a range of algorithms—from traditional time-series alignment techniques to modern deep learning-based embeddings—have been employed to improve both accuracy and scalability.

DTW & FastDTW Dynamic Time Warping (DTW) has been pivotal in aligning pitch contours with flexibility in tempo. It was first employed in QBH by [12], enabling alignment between user-hummed queries and stored melodies. Later, [11] extended DTW applications to polyphonic audio, handling more complex musical structures. To enhance computational efficiency, [26] introduced FastDTW and warping indexes, dramatically reducing runtime without a significant loss in alignment accuracy, thus making DTW-based methods more suitable for real-time QBH applications.

HMMs Hidden Markov Models (HMMs) offer a probabilistic framework for modeling sequential data, well-suited to melody modeling. The foundational concepts were introduced in [27], and later adapted for large-scale audio retrieval in [13]. QBH systems were further enhanced by [37], who integrated pitch interval modeling within an HMM framework, allowing robust matching even under noisy, incomplete, or variably sung input.

LSH Locality Sensitive Hashing (LSH) provides a highly scalable approach for approximate nearest-neighbor searches in high-dimensional spaces. This was utilized in [10] for real-time QBH retrieval, allowing the system to quickly narrow down potential matches in massive music repositories. Earlier, [34] demonstrated LSH's efficacy in melody indexing, enabling quick candidate filtering before applying more computationally intensive similarity measures.

Embedding-based Approaches Recent developments in deep learning have shifted focus toward embedding-based similarity matching. Models trained to map audio inputs into a lower-dimensional latent space have enabled rapid similarity calculations using simple distance metrics like cosine similarity or dot product. [3] trained embeddings on large humming datasets to improve the representational capacity of humming queries. A unified approach integrating melodic and spectral features was explored in [6], while [42] developed deep architectures for learning high-level audio representations, forming the foundation for embedding-based matching in most current QBH systems.

Together, these diverse strategies—from sequence alignment and probabilistic modeling to hash-based indexing and deep embeddings—enable modern QBH systems to achieve high accuracy, scalability, and robustness across a variety of musical genres, query conditions, and dataset sizes.

5. **Dataset Creation**

A solid foundation of well-structured and diverse datasets is crucial for the design, training, benchmarking, and continual improvement of Query-by-Humming (QBH) systems. These datasets capture the real-world variability of humming queries, including tonal differences, timing fluctuations, background noise, and user diversity. They support the training of machine learning models, the evaluation of retrieval performance, and the development of robust systems capable of handling noisy or incomplete inputs.

CHAD The CHAD dataset is a pioneering collection of manually aligned hum-song pairs, offering precise pitch and timing annotations. It provides a valuable testbed for evaluating temporal alignment methods such as DTW and FastDTW. The dataset's time-aligned structure makes it especially useful for examining how well QBH systems match melodic contours and rhythms under controlled conditions [2][3].

HumTrans The HumTrans dataset [3] is an open-source resource specifically designed for training deep learning models on melody transcription and QBH tasks. It features a diverse set of hummed inputs with variations in tempo, pitch, accent, and background noise. These properties help improve model generalization, especially for real-world applications where humming may be inconsistent and acoustically noisy.

MTG-QBH The MTG-QBH dataset [16] includes natural, real-user humming recordings that reflect broad tonal and stylistic diversity. Its inclusion of spontaneous and expressive queries makes it particularly valuable for testing the robustness of QBH systems against variability in human performance. It also enables the study of cross-lingual and user-specific retrieval patterns.

MIREX-QBSH The MIREX-QBSH dataset [15], used in the Music Information Retrieval Evaluation eXchange, provides a standardized platform for evaluating QBH systems. It includes a wide range of test queries and associated metadata, allowing researchers to directly compare the performance of different retrieval algorithms under consistent and reproducible conditions.

Crowd-sourced Techniques Recent research has explored scalable dataset creation via crowd-sourcing platforms. Efforts in [2][3] have utilized crowd-sourced humming samples and applied semi-supervised learning techniques to ensure label accuracy at scale. This approach enables the collection of large, diverse, and realistic humming datasets without incurring prohibitive annotation costs, fostering the growth of more inclusive and representative datasets.

Collectively, these datasets underpin the development and evaluation of QBH systems, enabling advances in feature extraction, melody matching, and real-world deployment. They also support reproducibility and benchmarking, which are essential for academic and industrial progress in music information retrieval.

6. System Optimization and Scalability

Scalability and performance optimization are essential for enabling Query-by-Humming (QBH) systems to function effectively in real-time and commercial settings. These improvements span various aspects, including indexing, computational acceleration, and memory efficiency.

Indexing Locality Sensitive Hashing (LSH) and subsequence indexing techniques have significantly improved the speed of candidate retrieval in large-scale music databases. An LSH-based indexing framework was introduced in [10], reducing retrieval latency while maintaining high accuracy. Further enhancements were proposed in [34], where LSH was applied to both MIDI and audio data, enabling real-time performance in extensive music collections.

Parallelization & GPU Acceleration To meet the high computational requirements of deep learning-based feature extraction, the use of parallel processing and GPU acceleration has become standard practice. The foundational ideas for real-time audio feature computation were established in [22], and have since evolved into fully parallelized pipelines capable of processing humming inputs in real time.

Model Compression For deployment on edge devices like smartphones, model compression and pruning techniques have been explored to reduce computational load without compromising performance. In [6], it was shown that compressed models can still maintain high accuracy in melody extraction tasks. Earlier lightweight implementations such as the CubyHum system in [19] demonstrated the feasibility of mobile QBH systems, setting a standard for future on-device applications.

Hybrid Systems Multi-stage filtering has been adopted to improve both scalability and retrieval accuracy. A progressive filtering strategy was introduced in [25], where queries are first narrowed down using metadata (e.g., genre or artist), followed by more computationally intensive similarity matching. More recently, [47] proposed combining diverse features and ensemble-based voting techniques within hybrid QBH frameworks, enhancing robustness and retrieval precision.

These innovations collectively contribute to the practical deployment of QBH systems, enabling them to deliver fast and accurate results in both cloud-based platforms and resource-constrained environments like mobile devices.

7. **Evaluation Metrics**

To evaluate the performance of Query-by-Humming (QBH) systems, the Mean Reciprocal Rank (MRR) and Top-X Hit Rate metrics are commonly used [15, 36]. MRR calculates the average of the reciprocal ranks of the correct results in the retrieval list, giving an indication of how quickly the correct match appears in the ranking. A higher MRR reflects that the system retrieves relevant results closer to the top. The Top-X Hit Rate measures the proportion of queries where the correct result appears within the top X retrieved items, providing insights into the system's retrieval precision, particularly under practical limitations.

These metrics are essential in QBH systems, especially when users may only hum a brief segment of a song, as they capture both the accuracy of the retrieval process and user satisfaction. Moreover, evaluation frameworks like MUSART provide standardized benchmarking tools for comparing retrieval performance across various systems, datasets, and feature extraction methods [15]. These frameworks facilitate reproducibility and fairness in performance comparisons by offering structured datasets, predefined tasks, and evaluation scripts.

8. **Commercial Applications**

Query-by-Humming (QBH) systems have successfully moved from academic research to real-world commercial use, demonstrating their practical value in music information retrieval. A notable example is Google's "Hum to Search" feature, launched in 2020, which allows users to hum a melody and identify the corresponding song using embedding-based audio retrieval techniques powered by deep learning models [7]. This service is designed to be robust to pitch and rhythm variations, ensuring accurate matches even with off-key humming.

Other well-known platforms such as Midomi and SoundHound have long offered real-time humming and singing recognition, enabling users to search for music simply by vocalizing a tune [21]. These applications utilize advanced audio fingerprinting and machine learning models to identify matches from extensive music databases in seconds.

The integration of QBH capabilities into music streaming services and mobile applications further enhances user accessibility and interaction, particularly when users cannot recall lyrics or song titles [21]. As a result, QBH not only improves the music search experience but also introduces new opportunities for music discovery, recommendation, and engagement across various user demographics and scenarios.

9. **Challenges and Limitations**

Despite the promising capabilities of Query-by-Humming (QBH) systems, several significant challenges hinder their performance and scalability. One of the foremost issues is the high variability in user humming, which can differ widely in pitch accuracy, rhythm consistency, tempo, and articulation. Such inconsistencies introduce a large degree of unpredictability in the input, making it difficult for systems to generate stable feature representations for matching [8].

Background noise is another critical limitation, especially in mobile or uncontrolled environments. Noisy recordings—due to ambient sounds, overlapping speech, or recording artifacts—can severely distort the input signal, compromising the feature extraction process and leading to incorrect or failed retrievals [5]. Although noise-robust models and preprocessing techniques exist, they are not always effective under all acoustic conditions.

Polyphonic complexity presents a further challenge, particularly when humming queries are matched against full-length polyphonic audio tracks containing multiple instruments and vocal layers. Distinguishing the target melody from such complex mixtures requires advanced source separation or attention-based modeling, which remains computationally demanding and error-prone [8].

Additionally, multilingual melodies and culturally diverse musical expressions introduce challenges related to melodic interpretation and representation. Differences in musical scales, tuning systems, and ornamentation styles across languages and cultures can hinder the system's ability to generalize effectively [32][33]. As global usage of QBH expands, addressing these cross-cultural limitations becomes increasingly important for inclusive and equitable performance.

10. **Future Directions**

The future of Query-by-Humming (QBH) systems lies in addressing current limitations while embracing emerging technologies to enhance scalability, accuracy, and user experience. One key area of advancement is the adoption of advanced deep learning architectures, such as Siamese networks, transformers, and convolutional-recurrent models, which have shown potential for capturing temporal and tonal nuances in melodic content [42][46]. These models can better represent and compare humming queries with target songs, improving retrieval accuracy even under noisy or varied input conditions.

Cross-modal retrieval is another promising direction, where humming queries can be matched not just to audio, but to lyrics, sheet music, or even visual representations like music videos [6]. This approach broadens the scope of QBH applications and enhances accessibility across different content types and user preferences.

With increasing user expectations for seamless interaction, real-time processing capabilities are becoming essential. Lightweight architectures and edge-computing optimizations are being developed to support instantaneous feedback during humming input, reducing latency and improving user satisfaction [19][25].

Personalization also holds promise, enabling systems to adapt to an individual's vocal patterns, pitch tendencies, and musical tastes. By incorporating user-specific feedback and preference modelling, QBH systems can offer more relevant and accurate results [6][47].

The use of synthetic queries—artificially generated humming-like audio fragments—has emerged as a means to augment training datasets and enhance system robustness to diverse inputs. Techniques such as pitch shifting, time-stretching, and noise injection help simulate real-world user queries during model training [20][44].

Finally, integration with music platforms like Spotify, YouTube Music, or SoundCloud can turn QBH into a mainstream search feature, enabling users to find songs by humming directly within their favorite streaming environments. This not only boosts user engagement but also fosters broader adoption of QBH technology [23][50].

11. **Conclusion**

Query-by-Humming (QBH) systems have witnessed a transformative evolution, progressing from early string-matching and template-based approaches to modern deep learning-driven frameworks. This shift has enabled greater robustness to user variability, improved accuracy, and broader applicability in real-world scenarios. The integration of sophisticated AI models, such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and embedding-based retrieval mechanisms, has significantly enhanced the system's ability to handle diverse and noisy humming inputs.

The future development and widespread adoption of QBH systems hinge on several critical factors. First, the availability of large, diverse, and well-annotated datasets is essential for training and evaluating deep models under realistic conditions. Second, scalability must be prioritized to ensure fast and accurate retrieval across millions of songs, particularly in commercial applications. Innovations in indexing techniques, approximate nearest neighbour (ANN) search, and model compression will play a vital role in this regard.

Lastly, user-centric design principles are crucial for fostering engagement and ensuring that QBH tools remain intuitive and accessible. Features such as personalization, real-time interaction, and integration with existing streaming platforms will determine how effectively QBH technologies can bridge the gap between user intent and musical discovery [23][49][50].

Ultimately, the convergence of technical innovation, usability, and music industry collaboration will define the next generation of QBH systems, pushing the boundaries of music information retrieval and redefining how users interact with sound.

12. **References**

[1] Ranjan, S., & Srivastava, V. (2023). Incorporating Total Variation Regularization in the design of an intelligent Query by Humming system. arXiv preprint arXiv:2302.04577.□

[2] Amatov, A., Lamanov, D., Titov, M., Vovk, I., Makarov, I., & Kudinov, M. (2023). A Semi-Supervised Deep Learning Approach to Dataset Collection for Query-By-Humming Task. arXiv preprint arXiv:2312.01092. □

[3] Liu, S., Li, X., Li, D., & Shan, Y. (2023). HumTrans: A Novel Open-Source Dataset for Humming Melody Transcription and Beyond. arXiv preprint arXiv:2309.09623.□

[4] Chen, L., & Cheung, D. W. (2020). Subsequence Matching with Gaps-Range-Tolerances in Time Series. Proceedings of the VLDB Endowment, 13(7), 1031-1044.□

[5] Chen, L., & Cheung, D. W. (2015). Embedding-based subsequence matching of time series. The VLDB Journal, 24(4), 511-535.□

[6] Salamon, J., Gómez, E., Ellis, D. P. W., & Bello, J. P. (2013). Melody extraction from polyphonic music signals: Approaches, applications, and challenges. IEEE Signal Processing Magazine, 31(2), 118-134.□

[7] Google Research. (2020). The Machine Learning Behind Hum to Search. Retrieved from https://research.google/blog/the-machine-learning-behind-hum-to-search/ □

[8] Duchi, J., & Phipps, B. (2005). Query By Humming: Finding Songs in a Polyphonic Database. Stanford University.□

[9] Tripathy, A. K., Chhatre, N., Surendranath, N., & Kalsi, M. (2009). Query by Humming System. International Journal of Recent Trends in Engineering, 2(5), 373-377.

[10] Sharma, N. (2024). Query by Humming via Locality Sensitive Hashing. Brown University.

[11] Du, J. (2015). A Method of Query-by-Humming System for Polyphonic Audio. University of Rochester.

[12] Ghias, A., Logan, J., Chamberlin, D., & Smith, B. C. (1995). Query by humming: musical information retrieval in an audio database. Proceedings of the third ACM international conference on Multimedia, 231-236.

[13] Wang, A. (2003). An industrial-strength audio search algorithm. Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR), 7-13.

[14] Pardo, B., & Birmingham, W. P. (2002). Encoding timing information for musical query matching. Proceedings of the International Conference on Music Information Retrieval (ISMIR), 267-268.

[15] Müller, M. (2007). Information Retrieval for Music and Motion. Springer.

[16] Kim, Y. E., & Whitman, B. (2002). Singer identification in popular music recordings using voice coding features. Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR), 164-169.

[17] Goto, M. (2001). A real-time music-scene-description system: predominant-f0

[17] Goto, M. (2001). A real-time music-scene-description system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. *Speech Communication*, 43(4), 311–329.

[18] Typke, R., Wiering, F., & Veltkamp, R. C. (2005). A survey of music information retrieval systems. *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 153–160.

[19] Pauws, S. (2002). CubyHum: A fully operational query by humming system. *ISMIR 2002: International Symposium on Music Information Retrieval.*

[20] Müllensiefen, D., & Frieler, K. (2007). Modelling experts' notions of melodic similarity. *Musicae Scientiae*, 11(2_suppl), 183–210.

[21] SoundHound Inc. (2020). Midomi – Discover music by singing or humming. Retrieved from https://www.midomi.com

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

[23] Casey, M., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., & Slaney, M. (2008). Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, 96(4), 668–696.

[24] Poliner, G. E., & Ellis, D. P. W. (2007). A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007, 1–9.

[25] Jang, J. S. R., & Lee, H. R. (2001). A general framework of progressive filtering and its application to query by singing/humming. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2), 461–471.

[26] Zhu, Y., & Shasha, D. (2003). Warping indexes with envelope transforms for query by humming. *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, 181–192.

[27] Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2), 257–286.

[28] Ellis, D. P. W., & Poliner, G. E. (2007). Identifying 'Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking. *IEEE ICASSP*, 1429–1432.

[29] Serra, J., Gómez, E., Herrera, P., & Serra, X. (2008). Chroma binary similarity and local alignment applied to cover song identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(6), 1138–1151.

[30] Lemström, K., & Laine, P. (2008). Musical information retrieval using musical parameters. *Journal of New Music Research*, 37(4), 353–361.

[31] Salamon, J., Rocha, B., & Gómez, E. (2012). Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(6), 1759–1770.

[32] Orio, N., & Schwarz, D. (2001). Alignment of monophonic and polyphonic music to a score. *Proceedings of the International Computer Music Conference (ICMC)*, 155–158.

[33] Yang, C., Kao, M., & Chen, H. (2005). Music retrieval using melody matching. *Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 77–84.

[34] Ryynänen, M., & Klapuri, A. (2008). Query by humming of MIDI and audio using locality sensitive hashing. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2249–2252.

[35] Wang, Y., Wang, M., & Li, X. (2010). Query-by-humming system based on improved pitch extraction and similarity matching algorithm. *International Conference on Computer Application and System Modeling*, 309–313.

[36] Cano, P., Kaltenbrunner, M., & Widmer, G. (2005). Near-duplicate detection for audio using chroma-based representations. *IEEE ICASSP*, 233–236.

[37] Tang, Z., Zhao, H., & Wu, S. (2012). Improved QBH system based on pitch interval and HMM. *International Journal of Computer Applications*, 39(16), 1–6.

[38] Wu, C. H., Lee, C. H., & Chuang, C. M. (2009). Music retrieval based on query-by-singing/humming (QBSH) and automatic alignment of music fragments. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4), 659–674.

[39] Lee, K., & Slaney, M. (2008). Acoustic chord transcription and key extraction from audio using key-dependent HMMs trained on synthesized audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2), 291–301.

[40] Lee, C. H., & Wu, C. H. (2006). Automatic semantic annotation for music content based on music emotion. *IEEE Transactions on Audio, Speech, and Language Processing*, 13(4), 1121–1129.

[41] Lim, J. H., Kim, J. S., & Lee, K. (2014). Query by humming system using convolutional neural network. *ICMLA*, 283–288.

[42] Humphrey, E. J., Bello, J. P., & LeCun, Y. (2013). Feature learning and deep architectures: New directions for music informatics. *Journal of Intelligent Information Systems*, 41(3), 461–481.

[43] Choi, K., Fazekas, G., Sandler, M., & Cho, K. (2017). Convolutional recurrent neural networks for music classification. *IEEE ICASSP*, 2392–2396.

[44] Lattner, S., Mörchen, F., & Eck, D. (2018). Learning interpretable representations of music with deep generative models. *ISMIR*, 379–386.

[45] Huang, Y. C., Yeh, C. H., & Yang, Y. H. (2021). Pop Music Highlighter: Marking the Emotion Keypoints. *Proceedings of the 29th ACM International Conference on Multimedia*, 1864–1872.

[46] Kum, H., & Lee, S. (2022). DeepSinging: A deep neural architecture for singing melody transcription. *IEEE Transactions on Affective Computing*, 13(2), 950–961.

[47] Chen, S., Wu, L., & Liu, H. (2023). Feature fusion and voting mechanisms for robust QBH systems. *Journal of Intelligent Information Systems*, 62(1), 99–117.

[48] Xu, Y., Kong, Q., Wang, W., & Plumbley, M. D. (2021). PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28, 288–304.

[49] Schedl, M., Gómez, E., & Urbano, J. (2014). Music information retrieval: Recent developments and future challenges. *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1171–1174.

[50] Serra, X., & Gómez, E. (2020). Music information research: What is it, and why do we need it?. *Communications of the ACM*, 63(6), 80–88.