Journal of Multidisciplinary Engineering Science Studies (JMESS)
ISSN: 2458-925X
Vol. 12 Issue 4, April - 2026

A Reproducible Workflow for AI-Generated Singing MV Production-

Using Image-to-Video Synthesis and FFmpeg-Based Expression Overlay

Hung-Che Shen
Department of Emerging Media Design
I-Shou University
Kaohsiung, Taiwan
e-mail: shungch@isu.edu.tw

***Abstract*—This study introduces a lightweight, fully reproducible workflow for producing AI-generated singing music videos (MVs) by combining image-to-video synthesis with FFmpeg-based facial expression overlay. In contrast to traditional approaches that depend on 3D modeling, character rigging, or data-intensive deep learning models for audio-driven facial animation, the proposed framework prioritizes modularity, timing-deterministic synchronization control, and minimal computational requirements.. The workflow comprises five standardized stages: (1) AI-based character generation, (2) image-to-video motion synthesis, (3) rule-based facial expression state design, (4) time-synchronized FFmpeg overlay compositing, and (5) deterministic audio-video integration. A practical demonstration using the children's song "Two Tigers" (兩隻老虎) at 80 BPM confirms the method's reproducibility. All assets, prompts, and command scripts are publicly released on GitHub. This system establishes a scalable, low-resource baseline for synchronized singing MV production, eliminating the need for neural lip-sync inference or GPU-dependent training while maintaining reproducible temporal synchronization.**

Keywords—Singing MV; Reproducible workflow; Image-to-video synthesis; FFmpeg overlay; Deterministic synchronization

I. INTRODUCTION

Recent advancements in AI-driven image generation and motion synthesis have substantially reduced barriers to digital character animation. Contemporary image-to-video models can animate static portraits in seconds, producing natural motion from text prompts [1]. Although neural lip-sync techniques deliver high-quality audio-visual alignment, they typically demand extensive datasets, GPU-accelerated inference, and intricate preprocessing pipelines [2].

Traditional digital character animation often relies on labor-intensive 3D rigging and manual keyframing [3]. Despite these innovations, lightweight and reproducible workflows for multimedia production remain underexplored. Many existing systems rely on probabilistic neural inference, yielding non-deterministic outputs and strong hardware dependencies.

To address these limitations, this study proposes a deterministic, modular workflow that integrates AI-based motion generation with rule-based facial expression overlay via FFmpeg. Rather than deriving lip motion directly from audio through end-to-end neural networks, the framework deliberately separates motion synthesis, expression state control, and audio synchronization. This decomposition enhances reproducibility, reduces hardware demands, and supports scalable, segment-based construction. This study makes three main contributions:

At the methodological level, we propose a fully standardized, modular, and reproducible standard operating procedure (SOP) for AI-generated singing music video (MV) production. The workflow decomposes the complex task into five deterministic stages—musical timing modeling, character image generation, image-to-video motion synthesis, FFmpeg-based expression overlay, and audio-video integration—eliminating the need for neural lip-sync models or 3D rigging.

At the technical implementation level, we introduce a deterministic synchronization framework that combines Romanized Mandarin (Hanyu Pinyin) prompts for image-to-video synthesis with BPM-derived timing control and rule-based FFmpeg facial expression overlay. This approach achieves frame-accurate, hardware-independent synchronization without relying on probabilistic neural inference or GPU-intensive training.

At the application level, we demonstrate the practicality of the proposed workflow through a complete, publicly released reproducible package (including all prompts, scripts, MIDI files, and generated assets) for the children's song "Two Tigers" (兩隻老虎). This establishes a scalable, low-resource baseline that enables researchers and creators to produce synchronized singing MVs efficiently and repeatably.

Recent studies on singing music video production have primarily followed two distinct paradigms. Rule-based and commercial audio synthesizers, such as MBROLA and VOCALOID, focus on generating singing voices with varying degrees of expressiveness. However, these approaches fail to provide any native support for synchronized visual animation, resulting in audio-only outputs that require separate, labor-intensive post-production for MV creation.

Deep learning-based singing voice synthesis (SVS) models and end-to-end neural lip-sync techniques have advanced the field by delivering natural prosody and precise audio-driven facial movements. However, these approaches fail to address the critical requirements of reproducibility and low-resource deployment, as they demand large-scale audio-visual datasets, GPU-intensive training or inference, and often produce non-deterministic outputs that hinder consistent experimental validation.

Traditional 3D character rigging and manual keyframing offer full artistic control over facial expressions and body motion. However, these approaches fail to scale for rapid AI-assisted production, requiring specialized skills, extensive manual labor, and prohibitively high time costs that make them impractical for researchers and independent creators working under limited computational resources.

Importantly, the proposed workflow is positioned not as a replacement for phoneme-accurate neural lip-sync systems, but as a reproducible and low-resource baseline for rapid AI-assisted singing MV prototyping under constrained computational settings.

II. PROPOSED WORKFLOW

This section describes a procedurally reproducible workflow tailored to practical constraints observed in current image-to-video models (e.g., Grok Imagine), which reliably synchronize lip movements for Mandarin speech but exhibit reduced stability and control when prompted to “sing” directly using Chinese characters [4].

To maximize stability and reproducibility, the workflow employs Romanized Mandarin (Hanyu Pinyin) as the primary prompt format for singing instructions, leveraging the phonetic nature of the Roman alphabet which is often better processed by global LLMs [5]. The demonstration uses the children's song “Two Tigers” (两只老虎), with the lyrical sequence “两只老虎 两只老虎 — 跑得快 跑得快” structured into four modular 6-second segments for a complete 24-second demonstration. The complete workflow consists of five stages:

1) *Musical Timing Model*
2) *Character Image Generation*
3) *Segment-Based Image-to-Video Singing Synthesis*
4) *FFmpeg-Based Expression Overlay and Segment Concatenation*
5) *Audio-Video Integration*

A. Musical Timing Model

The demonstration adopts a 4/4 time signature at 80 BPM, yielding a beat duration of 0.75 seconds (60/80) and a measure duration of 3 seconds (4 beats). Each musical phrase spans two measures (6 seconds), enabling four independent 6-second segments for the full 24-second sequence. This segmentation minimizes temporal drift and ensures modular reproducibility. The use of a fixed BPM to calculate frame-accurate timestamps is a standard practice in digital audio workstations (DAW) and synchronized multimedia systems [6]. Thus:

- First phrase (两只老虎 两只老虎) → 6 seconds
- Second phrase (跑得快 跑得快) → 6 seconds
- Total demo length → 24 seconds

This 24-second design requires four separate 6-second image-to-video prompts, ensuring modular reproducibility while minimizing temporal drift..

B. Stage 1: AI Character Image Generation

A consistent half-body singer portrait serves as the identity anchor. The image is generated using a text-to-image model (Grok Imagine in this case) with carefully designed prompts emphasizing neutral expression, front-facing orientation, even lighting, clear lip and facial landmark visibility, and a simple or transparent background. An example prompt is: “Half-body female singer, front-facing, neutral facial expression, soft studio lighting, clear mouth visibility, upper body framing, simple background.” The resulting base image (character_base.png) remains fixed across all segments (see Fig. 1).

Fig. 1. *Example of AI-generated base singer portrait used as character identity anchor..*

C. Stage 2: Image-to-Video Singing Motion Synthesis

To enhance stability, prompts utilize Romanized lyrics rather than Chinese characters. Each 6-second segment is generated independently.

For Segment 1 (0–6 s): lyrics “两只老虎 两只老虎” (Romanized: liang zhi lao hu liang zhi lao hu).
Prompt example: “Half-body female singer gently singing: liang zhi lao hu liang zhi lao hu, moderate tempo children’s song style, natural rhythmic mouth opening, expressive but soft singing articulation, subtle head nod every 3 seconds, stable camera, realistic motion.”

For Segment 2 (6–12 s): lyrics “跑得快 跑得快” (Romanized: pao de kuai pao de kuai).

Prompt example: “Same female singer continuing to sing: pao de kuai pao de kuai, moderate tempo, natural rhythmic singing articulation, slight expressive emphasis on stressed syllables, subtle shoulder breathing motion, stable camera.” Similar prompts are applied to Segments 3 and 4 to complete the full 24-second demonstration sequence through repeated phrase generation. Independent generation prevents cumulative desynchronization and preserves reproducibility. Outputs are saved as segment_A_6s.mp4 through segment_D_6s.mp4.

D. Stage 4 and Stage 5: FFmpeg-Based Segment Concatenation, Expression Overlay, and Audio Integration

After the two independent 6-second image-to-video segments are generated, they must be concatenated into a unified 12-second visual sequence before audio merging.

1) Video Segment Concatenation

The two files of “segment_A_6s.mp4” and “segment_B_6s.mp4” are concatenated sequentially using FFmpeg. FFmpeg is widely recognized in research for its robust handling of multimedia transcode and filter-based manipulation without significant quality loss [7].

To ensure deterministic behavior, both segments must share identical resolution, frame rate, codec (e.g., H.264), and pixel format. Two standard concatenation approaches are available.

Method 1: Concat Demuxer (Recommended for Identical Encoding Parameters)

First, create a text file named list.txt with the following content:

file 'segment_A_6s.mp4'
file 'segment_B_6s.mp4'

Then execute in Windows’s command line mode :

ffmpeg -f concat -safe 0 -i list.txt -c copy 兩隻老虎_12s.mp4

This method performs stream copying without re-encoding, preserving original visual quality and avoiding generational compression loss.

Method 2: Filter-Based Concatenation (If Re-encoding Is Required)

If encoding parameters differ, the following command ensures compatibility:

*ffmpeg -i segment_A_6s.mp4 -i segment_B_6s.mp4 *
*-filter_complex "[0:v][1:v]concat=n=2:v=1:a=0[v]" *
-map "[v]" -c:v libx264 -pix_fmt yuv420p 兩隻老虎_12s.mp4

This method re-encodes the output using H.264 and standardizes the pixel format for playback compatibility. The resulting file, 兩隻老虎_12s.mp4, contains the complete 12-second visual sequence without audio.

2) Audio–Video Synchronization and Merging

Since the final MV must use the externally synthesized UTAU singing voice, the original video audio must be removed before merging. To eliminate the embedded audio stream from the 12-second concatenated video, the following FFmpeg command is used:

ffmpeg -i 兩隻老虎_12s.mp4 -c copy -an 兩隻老虎_12s_silent.mp4

Explanation:

- -c copy performs stream copy without re-encoding the video.
- -an removes the audio stream entirely.
- The output file 兩隻老虎_12s_silent.mp4 contains video only.

This step ensures that no pitch-inaccurate or speech-based audio remains in the final production pipeline. After producing the silent video file, the UTAU-synthesized singing voice 兩隻老虎.WAV is merged using the following FFmpeg command:

ffmpeg -i 兩隻老虎_12s_silent.mp4 -i 兩隻老虎.WAV \

-c:v copy -c:a aac -shortest 兩隻老虎.MP4

Explanation:

- -c:v copy preserves the existing video stream without re-encoding.
- -c:a aac encodes the WAV audio into AAC format for MP4 compatibility.
- -shortest ensures that the final output duration matches the shorter of the two streams, preventing unintended trailing silence or black frames.

If strict duration control is required, an explicit time constraint may be applied:

ffmpeg -i 兩隻老虎_12s.mp4 -i 兩隻老虎.WAV \

-t 12 -c:v copy -c:a aac 兩隻老虎.MP4

This guarantees exact alignment with the 12-second Musical Timing Model.

3) Deterministic Output Assurance

Because the Musical Timing Model defines an exact 12-second duration and both video segments are generated independently at 6 seconds each, synchronization errors are minimized. The explicit timing control during concatenation and multiplexing ensures timing-deterministic synchronization of the final audiovisual composition:

- Frame-accurate visual continuity
- Tempo-aligned audio synchronization
- Reproducible final output across different computing environments

The final deliverable file “兩隻老虎.MP4” represents the completed AI-generated singing music video reconstructed under the proposed Romanized prompt workflow.

III. DEMONSTRATION AND REPRODUCIBLE EXPERIMENT

The complete demonstration package, including all assets, is publicly available at: https://github.com/shungch-code/AI-Generated-Singing-MV-Production . This repository enables independent reconstruction of the singing MV for “Two Tigers” (兩隻老虎).

The repository organizes assets into musical data (MIDI, lyrics, WAV), visual assets (base image and

four segments), and prompt documentation (exact text files for each generation step). By archiving prompts, timing references, and processing commands, the package ensures methodological transparency and full reproducibility.

A. Demonstration Overview

The selected test case consists of the lyrical sequence:

"兩隻老虎 兩隻老虎 — 跑得快 跑得快"

The configuration parameters are defined as follows: The time signature is 4/4, and the tempo is fixed at 80 beats per minute. Under this configuration, each beat has a duration of 0.75 seconds, and each measure spans 3 seconds. Every two measures form a 6-second musical phrase.

The complete demonstration extends this structure into four modular 6-second segments, resulting in a total duration of 24 seconds. Each segment is generated independently using the Romanized prompt strategy described in Section II. The final MV is produced by concatenating the four visual segments, removing any embedded audio, and merging the externally synthesized singing voice generated via UTAU. Because all temporal parameters are mathematically defined, the audiovisual synchronization remains deterministic and reproducible.

B. Repository Structure

The GitHub repository is organized to support full experimental reconstruction. The directory includes three primary categories of assets:

1) Musical Data

- 兩隻老虎.MIDI. This file contains complete note timing and pitch information at BPM = 80. The MIDI serves as the authoritative timing reference for both singing synthesis and visual segmentation.
- 兩隻老虎歌詞.TXT. This file provides lyric segmentation aligned precisely with MIDI note boundaries. It ensures phoneme-to-note correspondence and facilitates Romanized prompt construction.
- 兩隻老虎.WAV This is the singing voice synthesized using UTAU. The waveform is generated directly from the MIDI file and maintains strict tempo alignment.

Together, these three files establish deterministic musical timing, pitch structure, and phoneme segmentation, forming the reproducible foundation of the experiment.

2) Visual Assets

To ensure full reproducibility, all visual-generation parameters and prompt configurations are documented within the repository.

- character_base.png Base singer portrait generated via Grok Imagine.
- **segment_A_6s.mp4**
- **segment_B_6s.mp4**
- **segment_C_6s.mp4**
- **segment_D_6s.mp4**

Each segment corresponds to a 6-second musical phrase and is generated independently to minimize temporal drift.

3) Prompt Documentation

- **prompt_character.txt**
- **prompt_segment_A.txt**
- **prompt_segment_B.txt**
- **prompt_segment_C.txt**
- **prompt_segment_D.txt**

These files archive the exact text prompts used during image and motion generation. Since generative systems are sensitive to textual variation, prompt logging ensures methodological transparency and reproducibility.

By publishing musical data, visual assets, and prompt documentation in a structured format, the study enables independent researchers to reconstruct the full AI-generated singing MV without ambiguity. This repository-based design reinforces the reproducible engineering contribution of the proposed workflow.

IV. DISCUSSION

A. Engineering Contribution

This work contributes a reproducible engineering pipeline rather than a novel neural architecture. Such modular approaches are increasingly favored in "AI-native" creative workflows to maintain human-in-the-loop control over stochastic outputs [8]. It leverages speech-synchronized image-to-video generation, stabilizes Mandarin singing via Romanized prompts, enforces deterministic segmentation, and employs FFmpeg for transparent alignment, thereby converting inherently non-deterministic generation into a modular, partially deterministic process.

B. Practical Advantages

The design remains model-agnostic, incurs low computational overhead, supports scalable long-form production through short segments, and adheres to reproducible research standards via public assets.

C. Current Limitations

Despite its reproducibility, several limitations remain:

- Lip articulation reflects singing-like exaggeration but does not strictly follow phoneme-level vowel shaping.
- Pitch variation is not visually encoded in mouth morphology.
- Emotional expression remains prompt-driven rather than acoustically driven.
- Model output may vary slightly across platform updates [9].

Thus, while the method produces perceptually plausible singing animation suitable for lightweight multimedia production, it does not achieve phoneme-accurate audiovisual articulation comparable to dedicated neural lip-sync systems [10].

D. Future Work

Potential directions include automated syllable-to-beat alignment, phoneme-level mouth shape correction, pitch-informed expression modulation, hybrid integration with neural lip-sync models, and quantitative evaluation of synchronization accuracy. Accordingly, the contribution of this work lies in workflow engineering and reproducible systems design rather than phoneme-level audiovisual modeling innovation.

V. CONCLUSION

This paper presents a reproducible workflow for AI-generated singing music video production that accommodates current generative model limitations. By reformulating Mandarin singing animation through Romanized prompts, short-segment synthesis, deterministic beat-aligned segmentation, and FFmpeg-based assembly, the method achieves stable, synchronized results without specialized lip-sync training.

The demonstration using “Two Tigers” (兩隻老虎) validates the pipeline across four independent segments, with full reproducibility supported by the released repository. The framework's methodological focus—emphasizing modularity, transparency, and low resource demands—offers a practical foundation for research and application in AI-assisted media production, digital music visualization, and reproducible computational creativity. Future refinements in articulation modeling and synchronization metrics could further align prompt-driven approaches with data-driven audiovisual standards. Beyond the presented case study, the proposed workflow may serve as a reproducible baseline benchmark for future research in low-resource audiovisual singing synthesis, AI-assisted digital performance generation, and human–AI co-creative multimedia production.

REFERENCES

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. Advances in Neural Information Processing Systems, 27, 2672-2680.

[2] Chen, L., Srivastava, R. K., Duan, Z., & Xu, C. (2019). Hierarchical cross-modal talking face generation with dynamic sequence-conditioning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 8579-8588.

[3] Magnenat-Thalmann, N., & Thalmann, D. (Eds.). (2005). Handbook of virtual humans. John Wiley & Sons.

[4] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020.

[5] Yu, J., Xu, Y., Koh, J. Y., Zhang, H., Pang, T. J., Qin, J., Ku, A., Xu, Y., Li, J., Yu, J., Wang, H., Huang, V., Anyasodor, I., Tay, Y., & Wu, K. (2022). Scaling up GANs for text-to-image synthesis. arXiv preprint arXiv:2203.04715.

[6] Roads, C. (1996). The computer music tutorial. MIT Press.

[7] Tomar, S. (2006). Converting video formats with FFmpeg. Linux Journal, 2006(146), 10.

[8] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260. https://doi.org/10.1126/science.aaa8415

[9] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Lowe, L., & Jan Leike. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730-27744.

[10] Chung, J. S., Jamaludin, A., & Zisserman, A. (2017). You said that? British Machine Vision Conference (BMVC).