Legend of Wrong Mountain: AI Generated Opera

Lingdong Huang, Zheng Jiang, Syuan-Cheng Sun, Tong Bai, Eunsu Kang, Barnabas Poczos
Carnegie Mellon University
Pittsburgh, PA, U.S.A.
lingdonh@andrew.cmu.edu

Abstract

As one of the oldest forms of Chinese Opera since the 16th century, *Kunqu* (崑曲) features literary virtuosity in its scripts, sophisticated vocal techniques in its singing, emotional and elegant yet rigorous bodily motions and facial expressions in its performance. Over its history, *Kunqu* has developed established modes and patterns, which makes it especially suitable for neural networks to learn. In order to generate *Kunqu* literature and performance computationally, we applied multiple machine learning and computer vision techniques. Our result, Legend of Wrong Mountain, is novel twofold: it is the first and only generated opera at the time of this writing, and it is a machine's attempt at *Gesamtkunstwerk* [11], "the Total Artwork". It explores the marriage between contemporary technologies and traditional art form. By studying historical scripts, musical notations and traditional methods for creating *Kunqu*, we tweaked existing algorithms and devised new ones to conform to the traditional rules and norms as closely as possible. We presented this project as a video accompanied by audio.

Keywords

Opera, Kunqu, Audiovisual, Machine Learning, Artificial Intelligence, AI Art, Total Art, Gesamtkunstwerk, LSTM, pix2pix, RNN, Markov Chain, OpenPose

Introduction

Kunqu is one of the oldest form of Chinese Opera. Featuring literary virtuosity in its scripts, sophisticated vocal techniques in its singing, and emotional and elegant yet rigorous bodily motions and facial expressions in its performance, Kunqu originated in the Wu cultural area and dominated the Chinese theatre for hundreds of years from c. 1573 to 1820, and is still being played today. Over its history, Kunqu has developed established modes and patterns, which makes it especially suitable for neural networks to learn. Our goal was to generate novel Kunqu literature and performance computationally.

According to the script generated by machine, our result, Legend of Wrong Mountain, the first Kunqu Opera generated by machine intelligence, is a story about an emperor, his lords and their intrigues. It includes ten scenes with about 15,000 Chinese characters.

We believe the piece is novel twofold: it is possibly the first and only generated Opera at the time of this writing, and it is a machine's attempt at Gesamtkunstwerk, the total work of art.

This paper explains the overall pipeline and detailed description of each process, sharing a clear blueprint of generating a piece of Chinese Kunqu Opera using machine learning. Techniques used in this project include Long Short-Term Memory (LSTM) [5], pix2pix [6], pix2pixHD [12], Recurrent Neural Networks (RNNs) [3], Markov Chain, OpenPose [8] and Detectron [4], as well as tools such as Canny Edge [2], PIL and SketchUp). By studying historical scripts and musical notations and researching traditional methods for creating Kunqu, we tweaked existing algorithms and devised new ones so that our implementation conforms to the traditional rules and norms as closely as possible. Our results were presented in two art forms; an actual woodblock printing script book and a machine learning generated video accompanied by audio.

Methods

The figure 1 shows the pipeline of this project, which has five main sections; music, script, performance, background scene and book generation.

Music Generation

The music of *Kunqu* are composed of melodies with established patterns. Because the melody is the most important part in Kunqu, the materials of Kunqu only contain melody information with lyrics. A hundred images containing traditional *Kunqu* sheet music were downloaded from *Kunquwang*[1] using a scraper script. We present our own algorithms to accomplish this task. First, we locate and identify all the numbers and various symbols in the image. Afterwards, we try to make sense of these symbols and generate a string representation containing all the information to be learned by neural networks [9].

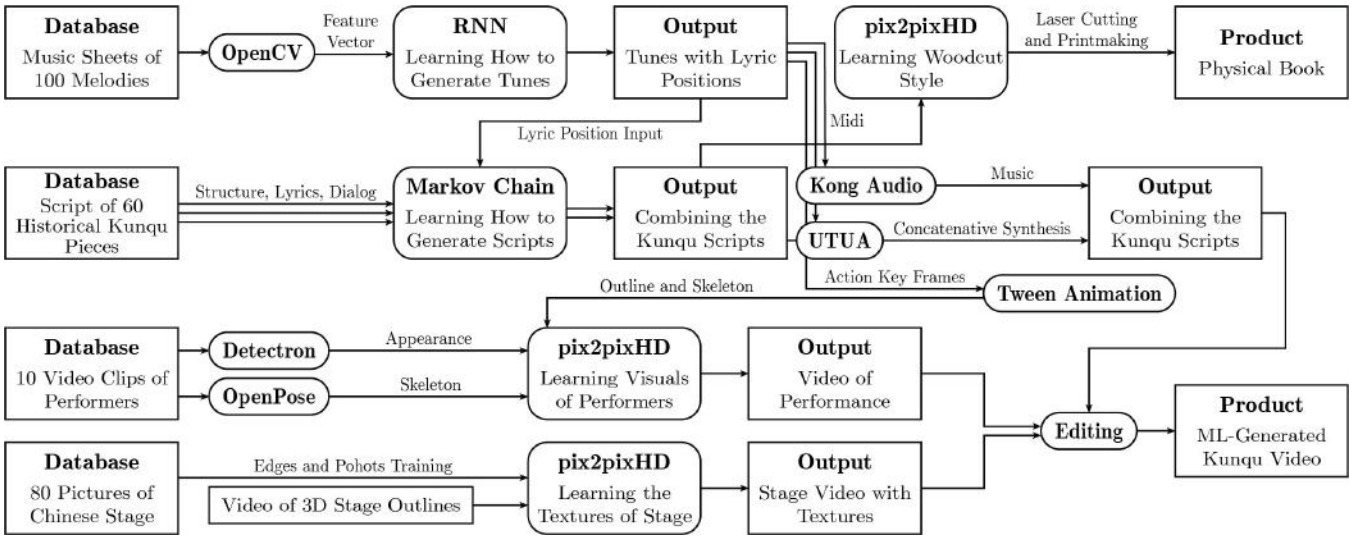

Figure 1: Workflow Pipeline

Music Sheet Analysis In this part we introduce the procedure to extract information from the image data. The *Kunqu* sheets use Numbered Notation (Ziffersystem or 簡譜), a common format for notating Traditional Chinese music, which we wrote a new computer-vision based algorithm to parse. The result is displayed as figure 2. Rather than creating standard bitmaps of numbers and using them to convolute images to guess the numbers, we developed a more precise technique: We isolated the horizontal region where numbers occur for each line, then we used a single bitmap, containing the mean of bitmaps of all 10 numbers, to convolute the region. This estimates coordinates of where any of the numbers occur. These detected numbers are then classified into 0 to 9 by simply multiplying their region element-wise with the standard bitmaps 0 to 9 and selecting the one with the highest score. Since there is a multitude of classes of symbols in the musical notation, each with a different difficulty level to detect, our strategy is therefore to start with the easiest class, erasing each instance from the image after successful detection, and proceeding to the next, more difficult class until we are done.

Figure 2: Optical Music Recognition

```
|G_4:8+|A_4:8-|C_6:2+|A_5:8-|G_5:8-|E_5:8+|G_5:1-|D_5:16-|D_5:
2-|C_5:2-|A_4:4-|G_4:4-|A_4:8+|D_5:4-|D_5:2-|E_5:2+|G_5:1-|D_5
:2-|C_5:2-|A_4:4-|G_4:4-|A_4:8+|0:8-|D_5:4-|D_5:4-|E_5:4+|G_5:
4-|A_5:1-|E_5:4-|D_5:2+|E_5:8-|E_5:4-|G_5:4-|G_5:4-|E_5:8-|D_5
:8-|D_5:4+|E_5:4-|E_5:2-|E_5:2-|D_5:8-|C_5:8-|C_5:8+|D_5:2-|D_
5:2-|E_5:2-|G_6:4-|0:2-|A_5:6-|G_5:1-|E_5:8+|G_5:1-|D_5:4-|C_5
:8+|C_5:8+|E_5:1-|E_5:2-|E_5:8+|G_5:8-|A_5:8-|E_5:2-|D_5:2-|D_
5:2-|C_5:2-|A_4:16-|A_4:16-|G_4:2-|E_4:8+|G_4:12+|C_5:8+|D_5:1
6-|E_5:4-|E_5:2-|D_5:2-|C_5:8+|C_8:1-|D_5:4+|E_5:4-|0:2-|D_5:2
-|C_6:4-|A_5:2-|B_5:2-|G_5:4-|E_5:8-|D_5:2-|C_5:2+|E_5:8-|D_5:
12-|D_5:4-|C_5:8-|D_5:8-|C_5:16-|D_5:8+|E_5:4-|A_4:8+|C_5:8+|A
_5:1-|G_5:8-|A_5:16+|G_5:8-|E_5:4+|A_5:4-|G_5:2+|G_5:2-|C_5:4-|
-|G_4:16-|C_5:8-|C_5:8+|D_5:8-|C_5:12+|C_5:8+|A_4:4+|A_4:4+|0
:2-|G_4:2-|E_4:8-|G_4:8-|A_4:12+|G_4:4-|A_4:4+|A_4:4-|C_5:4-|D
_5:8+|E_5:16-|D_5:8-|D_5:12-|G_5:2+|C_6:2-|E_5:4-|G_5:8+|G_5:
```

Figure 3: Sample melody represented by our notation

Melody Generation using RNN Recurrent Neural Network is one of the state-of-art tools to generate sequence samples as melody. We used charRNN([7]) to generate the melody for our opera. Instead of using the conventional method of encoding note-down-per-tick as a 128D vector, we observed that a large part of pitch levels will not be used in actual music. For example, we will never use a very low pitch like C1 or C2 in Kunqu. Therefore, we adopted a novel representation method in our intermediate data structure. We concisely represented them with descriptive ASCII strings. For example, an 8th note middle C accompanied by lyric is `|C_4:8+|`. The pitch class information makes the structure easier for the learning model to grasp. The format can also be easily converted into MIDI and vice versa.

As the notation can also be seen as a character sequence, we naturally selected a character-based recurrent neural network (charRNN) to learn it. The output, which has the same format as the input,are sent to the next stages of our generated opera: The melody itself is converted to MIDI and used for rendering the music, while the lyric position information is used for lyric generation, described in later sections.

Script Generation

To learn the hierarchical structure of *Kunqu* scripts, we introduced a nested system with different Markov Chains at each level. The method is capable of producing outputs containing reasonable large-scale structure for chapters, dialogues, prompts and lyrics, as well as smooth sentences within the structure. We also experimented with newer techniques such as LSTM and RNN in place of Markov chains, but eventually found the quality of results to be similar.

We used the first chapter of the first book generated by our finished algorithm as the script for the machine learning opera. The book has the generated title *Legend of Wrong Mountain* (錯山記), and contains 10 chapters in total, where the main plot involves the emperor, his lords and their intrigues.

Learning Structure of Kunqu Libretto Many text generation algorithm suffer from the problem of incoherence caused by short memory. Therefore, the method we propose is a trade off by inserting some domain knowledge in higher-level structures instead of solely relying on machine learning to learn these information. Our method combines systems for generating larger, chapter and paragraph level structures that reasonably resembles the corpus, with those for generating smaller, sentence and character level structures that reads smoothly.

Our dataset is a collection of 60 books of Kunqu script written by various Chinese playwrights from 10th to 19th century. In `.txt` format, the corpus can be easily analyzed with regular expressions for structural information. The dataset contains around 3,000,000 Chinese characters and is 9.1MB in total size.

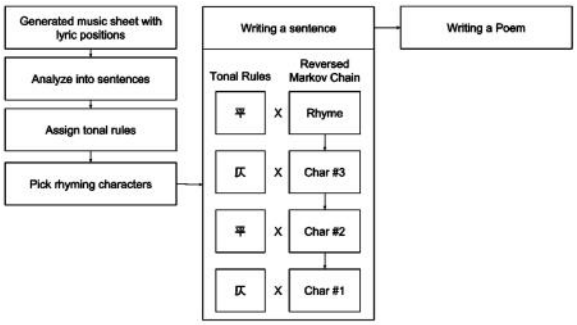

Figure 4: Poem generation pipeline

Qu Poetry Generation The sung parts of *Kunqu* were typically written in *Qu* (曲), a form of classical Chinese poetry consisting of characters that conforms to tonal rules from melodies. To generate them, our algorithm first analyze the corresponding melody generated in the previous step, and split them into sentences. Then, we reversed each sentence, such that the last, rhyming character is first determined, and all the previous characters are predicted backwards using a Markov chain trained on the reversed corpus. To ensure that the tonal rules are met, our algorithm multiplied the probability for outputting a character given by traditional Markov chain with the normalized distance between its tone and the tonal requirement (*Ping (平)*, *Ze (仄)*, or *no requirement (中)*) for the given position (figure 4). Thus, our generated poems read smoothly, rhyme, and adhere to the traditional tonal rules. Sample output and rough translation can be found in figure 5.

```
【簫聲漸】

古今去看。看那一派簫聲漸。
暮雨紛紛片翅飛。燕對鶯亂遮羞臉。
染征調塵。浪浪熟油奩檢。

The Flute Fades

In times new and old,
Hearken to the fading music of flutes, and behold
As the twilight rain splattered on their wings,
Swallows flew in disarray across her bashful face.
And the burning oil from the dressing case
Tainted the the ashes on the warriors' road.
```

Figure 5: Sample generated poem and rough translation

(a) Original (b) Label Map (c) Instance Map

Figure 6: Detection Results

Plot Summary The story of our generated script mainly revolves around the Emperor and his court: There was once a lord who was greatly trusted by the Emperor yet coveted the throne in secrecy. While he was conspiring with his friend, his family started to fell apart: he left his wife for breaking a bath tub, while his daughter-in-law turned out to be a drug addict. Kicked out by his father, the lord's son ventured to the capital city to take the imperial exam, flirting with various women on his way. Meanwhile, a mysterious couple assassinated a friend of the Emperor's envoy, and the fates of all the characters were tied together when they chanced upon each other in the very same road-side bar...

Performance Generation

Detecting people & pose with OpenPose & Detectron To understand character information of Kunqu, several video clips of different Kunqu characters were picked as the dataset. We used OpenPose to get the pose information (coordinates of keypoints on human bodies) and Detectron to get the semantic information (silhouette of human bodies) in parallel. OpenPose is a real-time multi-person keypoint detection library for body, face, hands, and foot estimation. After that, generated maps and send them to Pix2pixHD. The procedure is illustrated in figure 6.

Procedurally generating new poses and segmentation data We used python and PIL to generate animated images of new actor poses from scratch. A custom, minimal 3D

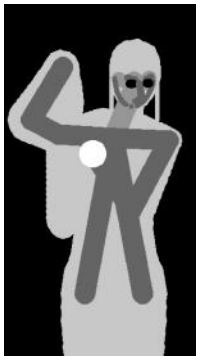

Figure 7: Generated Pose

engine that calculates rotation and offset for each joint in a skeleton was made. And for the costumes, pseudo-3D effects are created with polygons, with simple physics implemented using linear interpolation and numerical integration. A GUI is also design for quick annotation and generation of poses. The sample is displayed in figure 7.

Figure 8: Generated Actors

Generating costumes and faces with pix2pixHD pix2pix and pix2pixHD learn the correspondence between two sets of images. These procedurally generated semantics were then fed back into pix2pixHD which rendered them back to the photographic appearance of the performers. We had 9 pix2pixHD networks trained on characters of a variety of gender, age and costumes from the input videos. Thus by feeding the network 9 sequences of frames each containing a different generated animation, we obtained 9 corresponding photographic performances, ready to be used in the final video. 6 sample results can be seen in figure 8.

Scene Generation

Traditionally, Chinese Opera were often performed in tea houses. To honor this important part of Kunqu opera culture, we generated a virtual tea house to stage the performances using a combination of manual annotation and neural networks.

To construct the input for the network, we first built a 3D model of an traditional Chinese opera house using Google SketchUp, containing only the general outlines and without any textural information. Then we wrote a Ruby script to move the virtual camera in simulation of the perspective of a potential opera-goer, and exported the line work of the scene as a sequence of frames.

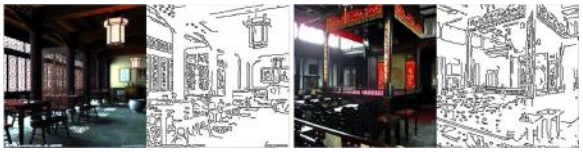

Figure 9: Canny edge detection result as training data

To train the network, 80 photographs of real Chinese stages and interiors were used as the dataset. We extracted the edges of these photos using Canny edge detection, and paired them with original photographs as input to a pix2pixHD network (figure 9). Pix2pixHD would thus output believable textures and lighting automatically according to the line work (figure 11). Finally, the sequence of generated images were prepended to the final video as a video walk-through of the space ere the beginning of the performance.

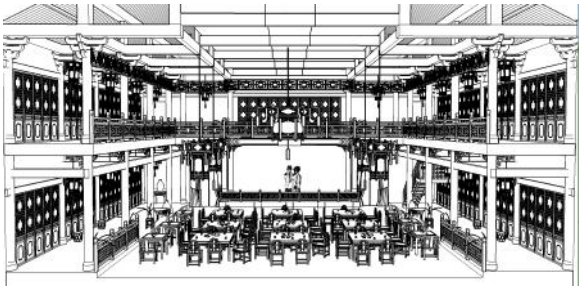

Figure 10: Tea House Reconstruction in Google SketchUp

Book Generation

We printed and bound 8 physical books containing the script of the opera, in traditional woodcut press style. First, a pix2pix network is trained on images of Chinese characters and their woodcut counterparts, which is then used to render the characters in our script. A custom software is then written to typeset all the pages in our book, before they were sent to a laser cutter (figure 12). The laser cutter prepared the printing blocks by engraving the pages onto 6mm wood pieces. Then, following traditional techniques, we applied black ink on the blocks and relief-printed onto rice paper. Finally, the books were bound with thread and needles.

Result

This project has presented in two different forms of art. First as a eight-minute-long video accompanied by sound and as eight printed books of the opera script.

At the beginning of the video, the camera slowly moves through the interior of a tea house while introductory texts are superimposed on the image (figure 13). The camera eventually stops at a chair facing the stage, where the performance

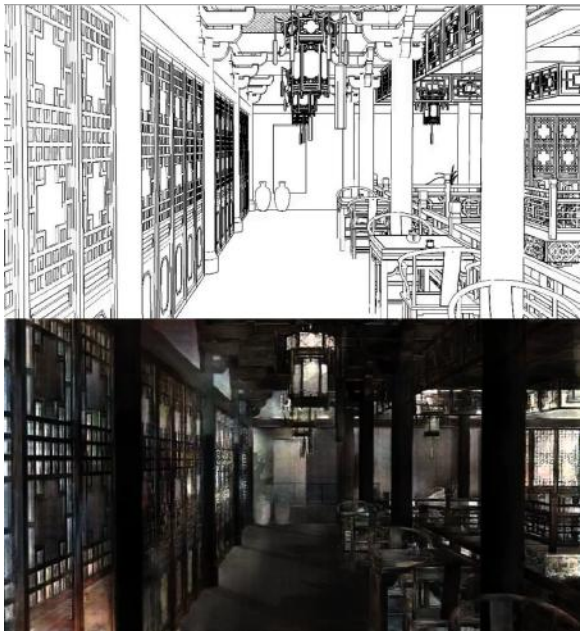

Figure 11: Above: frame exported from Sketchup; Below: rendering by pix2pixHD

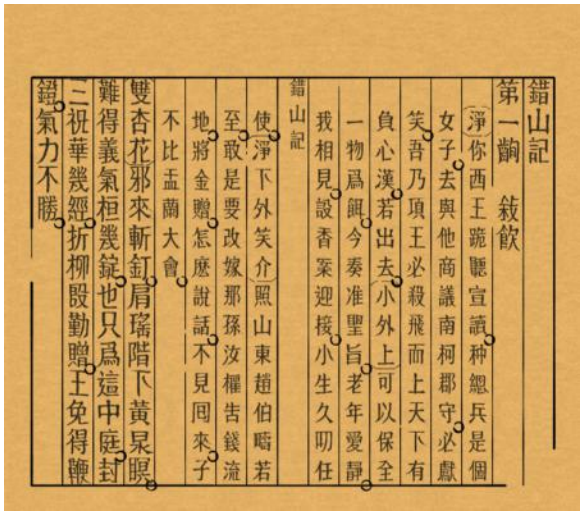

Figure 12: Generated rendering of a page from the script to be sent to laser-cuter

begins. Actors enter the stage in turns and sing with synthesized voices and dance to the generated music. Donning various costumes, the 5 of them perform dialog as well as monologue during different points of the performance. (figure 14 and figure 15)

The books are printed using laser-cut woodblocks on rice paper, and bound together using thread and needles by hand. Each of the books contain the first chapter of the script, and are typeset in the style of traditional Chinese prints. (figure 16)

Figure 13: Tea House

Figure 14: Still from performance

Figure 15: Another still from performance

Conclusion

This paper presents our techniques in generating Chinese Opera using machine learning methods. The project was presented at the *Art + Machine Learning* exhibition in Pittsburgh on May 1st, 2018. The printed and hand-bound books were offered to the audience for viewing at the show as further enhancement of their experiences.

In order for people not familiar with the Chinese language to enjoy our piece, we translated the lyrics and added them as subtitles to the video. Accompanied by rich audio and visuals, the somewhat nonsensical dialogs provided much room for viewer's own interpretation. We've heard people describe it as a love story, while others thought it was more about intrigues.

Figure 16: Book

For future work, we plan to study Chinese literature theory in depth, taking tones of the Chinese characters into consideration while generating music and lyrics. The music theory would be introduced to make the outputs more realistic as well. Also, we would like to apply vid2vid [10] to reduce flicker between adjacent frames and dig into pitch shift technique to generate more convincing vocals.

References

[1] 2018. Kunquwang. http://www.52Kunqu.com.

[2] Canny, J. 1986. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 8(6):679–698.

[3] Gers, F. A., and Schmidhuber, J. 2000. Recurrent nets that time and count. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, volume 3, 189–194. IEEE.

[4] Girshick, R.; Radosavovic, I.; Gkioxari, G.; Dollár, P.; and He, K. 2018. Detectron. https://github.com/facebookresearch/detectron.

[5] Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

[6] Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2016. Image-to-image translation with conditional adversarial networks. *arxiv*.

[7] Karpathy, A. 2015. Char-rnn.

[8] Kazemi, V., and Sullivan, J. 2014. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1867–1874.

[9] Sturm, B. L.; Santos, J. F.; Ben-Tal, O.; and Korshunova, I. 2016. Music transcription modelling and composition using deep learning. *arXiv preprint arXiv:1604.08723*.

[10] Ting-Chun Wang, Ming-Yu Liu, J.-Y. Z. G. L. A. T. J. K. B. C. 2018. Video-to-video synthesis. *arxiv*.

[11] Trahndorff, K. F. E. 1827. Ästhetik oder lehre von weltanschauung und kunst.

[12] Wang, T.-C.; Liu, M.-Y.; Zhu, J.-Y.; Tao, A.; Kautz, J.; and Catanzaro, B. 2017. High-resolution image synthesis and semantic manipulation with conditional gans. *arXiv preprint arXiv:1711.11585*.