MODELING AND DIGITIZING REPRODUCING PIANO ROLLS

Zhengshan Shi
CCRMA
Stanford University
`kittyshi@ccrma.stanford.edu`

Kumaran Arul
Department of Music
Stanford University
`karul2@stanford.edu`

Julius O. Smith
CCRMA
Stanford University
`jos@ccrma.stanford.edu`

ABSTRACT

Reproducing piano rolls are among the early music storage mediums, preserving fine details of a piano or organ performance on a continuous roll of paper with holes punched onto them. While early acoustic recordings suffer from poor quality sound, reproducing piano rolls benefit from the fidelity of a live piano for playback, and capture all features of a performance in what amounts to an early digital data format. However, due to limited availability of well maintained playback instruments and the condition of fragile paper, rolls have remained elusive and generally inaccessible for study. This paper proposes methods for modeling and digitizing reproducing piano rolls. Starting with an optical scan, we convert the raw image data into the MIDI file format by applying histogram-based image processing and building computational models of the musical expressions encoded on the rolls. Our evaluations show that MIDI emulations from our computational models are accurate on note level and proximate the musical expressions when compared with original playback recordings.

1. INTRODUCTION AND MOTIVATION

The invention of acoustic recordings in the late nineteenth century is widely accepted as a watershed moment in the history of music. However, piano rolls, which were in widespread use from approximately 1905 to 1940, are mistakenly treated as a footnote. Researchers studying early acoustic recordings have significant challenges with transcribing the nuances of a performance due to the poor sound fidelity, high noise artifacts, and limited recording length. Piano rolls did not share these shortcomings and were praised for their faithfulness and accuracy, providing a virtual transcription of a performance by punching holes on a paper scroll. Many important musicians who recorded on piano rolls never made acoustic recordings and were among the oldest generation to be recorded. These include composers like Claude Debussy, Scott Joplin, and Carl Reinecke, among others [13]. Modern digital audio workstations were inspired by this old music storage format and provide graphical display of MIDI (Music Instrument Digital Interface) files in what usually refers to a piano-roll editor. Thus, it can be claimed that piano rolls are among the most significant early commercial recordings and data storage media.

Playing and recording original rolls has been difficult due to the fragile paper and cumbersome demands of maintaining original instruments. Some recent efforts have pursued an alternative approach to accessing rolls by designing dedicated roll scanners to create image files of the rolls. This serves the purpose of archival preservation but does not allow piano rolls to be heard as the musical documents they were originally intended to be.

In this paper, we present a method for faithful automatic playback of reproducing piano rolls from image scans. Our research aims to digitize the data captured through analysis of notes, timings, and dynamics. The latter involves decoding proprietary expression mechanisms controlled through holes on the sides of the rolls. We evaluate our accuracy by comparing the digital file with acoustic recordings of rolls played on original player instruments.

2. RELATED WORK

There has been limited work on digitizing piano rolls by institutional researchers and independent hobbyists. Wayne Stahnke [14] is one of the first to scan piano rolls and convert them into a digital format. Stahnke transferred image information derived from a self-built roll scanner into a proprietary data format preserving the details of the punched holes. Colmenares et al [2] converted Stahnke's pre-processed data for piano rolls into MIDI information by using sparse matrices. A few researchers such as Trimpin [7], Trachtman [15], and Malosio et al [9] have worked on the emulation of piano rolls. Other individuals such as Anthony Robinson [12] developed scanning machines and software that allows manual adjustment of the punch holes for refinement and error correction in the process of MIDI generation. These pioneering efforts remain however largely inaccessible, as key algorithms are not revealed or evaluated and are unavailable for consideration and review. They also typically require intensive manual labor in the transfer process and are usually limited to one format of roll.

Our work aims to fill this gap by proposing techniques to digitize and model reproducing piano rolls including novel methods of transcribing the musical expres-

---

Attribution: Zhengshan Shi, Kumaran Arul, Julius O. Smith. "Modeling and Digitizing Reproducing Piano Rolls", 18th International Society for Music Information Retrieval Conference, Suzhou, China, 2017.

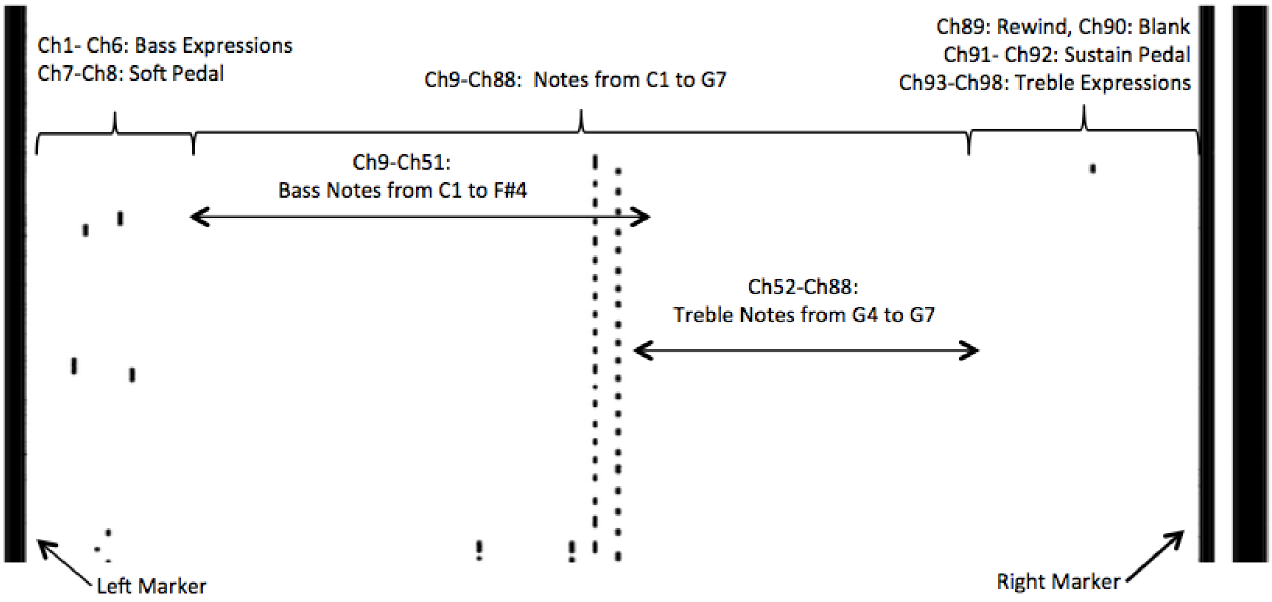

Figure 1. Excerpt from a Welte-Mignon piano roll image scan, consisting of the left marker, channels 1 to 98, and the right marker.

sion markings on those piano rolls. We evaluate our system by comparing the resultant MIDI emulation to the actual acoustic playback of the reproducing piano roll on a player piano.

3. REPRODUCING PIANO ROLLS FORMATS

A piano roll is a continuous roll of paper with perforations that store musical note data. It captured in real time the notes and rhythms of a pianist playing a special recording instrument. Music recorded on rolls are often performed by a player piano, a pneumatic machine that can decode the music data on the perforations and operate the piano action. Reproducing piano rolls are standard piano rolls with expression (dynamic and pedal) capabilities that can also be automatically executed by player piano. Expressions were captured in variable ways but were usually transcribed in shorthand and then coded onto the roll for bass and treble respectively. The complex process allowed for detailed editing of notes, rhythms and expressions.

The competitive environment of the early roll business created multiple different reproducing-piano systems, with variations in the size of rolls, configuration of holes, and pneumatic player designs. Although industry efforts at standardization eventually created some overlap, a thorough inventory of historical roll systems would number over two dozen. Most reproducing rolls of value to scholars are found in the catalogs of a handful of primary players, including Welte-Mignon (the first reproducing roll maker), Ampico, Duo-Art, and Hupfeld. Playing any reproducing roll requires a suitable player piano manufactured specifically for that format of roll. In some cases, as formats evolved over time, multiple players are needed to play back a manufacturer's rolls (early Red, T-100 Welte rolls do not play on later Green, T-98 Welte players, for example).

In this paper, we focus on the Welte-Mignon Licensee format (the third Welte format, a derivation of the T-98 rolls). A typical Welte-Mignon Licensee roll is 11 $1/4$ inches in width and holes spaced 9 per inch across [5], consisting of 98 *channels* [1] of punched holes [11]. As shown in Figure 1, channels 9 through 88 represent notes spanning from C1 to G7, with note F#4 as the splitting note for bass and treble. Channels 1 through 6 control bass (left-hand) expression, and channels 93 through 98 indicate treble (right-hand) expression. The pedal information is included in channel 7 and 8 (soft pedal) as well as channel 91 and 92 (sustain pedal). The expression channels combine to determine the dynamics of the piano performance.

4. ALGORITHM FOR DIGITIZING REPRODUCING PIANO ROLLS

In this section, we describe our method for digitizing and modeling the reproducing rolls. We first obtain piano roll images through a scanner, then construct a template that matches the layout geometry of the piano roll. Based on the template and locations of the perforations, we recover the note matrix that contain onset times for each note and musical expression. Finally we model musical expressions into dynamics and pedal information, that can be preserved in MIDI files.

4.1 Scanning

The first step in digitizing the reproducing-piano rolls is to generate the image file by scanning. The scanner used for this project was purposely built for rolls with a transport

---

[1] For the purpose of this paper, we refer to a "channel" as a column of the punched holes.

mechanism that allows continuous image capture by a contact image sensor (CIS) module which produces a graphic image file in either CIS or bitmap graphic format. The scan produces a directory of bitmap image files, initially preserving the 8-bit grayscale information. Each scanned image is 7296 pixels wide, juxtaposing front and back of the roll onto one side, with a pixel resolution of 324 dots-per-inch (dpi). We convert the grayscale raw image data to binary and invert the digit, so that 0 indicates no hole (white), and 1 indicates a hole coverage (black) at each pixel. On average, the size of a punched hole is 17 by 28 pixels.

4.2 Channel Grid Construction

Given a fixed piano-roll format, the edges of the zones stay sufficiently constant due to the general uniformity of the piano rolls. Thus, a quantitative template specifying piano-roll layout geometry such as the exact location for each channel and the spacing between different zones is necessary. We refer to it as a *Channel Grid*.

However, to the best of our knowledge, there is no such precise grid of hole placement available. Only general historical and empirical evidence from playback are available. Thus, using documentation of roll formats [11], and the notation of the musical works recorded, we derived reference points which were used to create a quantitative template locating the notes and expression holes on the roll.

Histogram-based image processing [6] has been used previously in optical music recognition to determine grid layouts of musical scores. Fujinaga [4] applied the histogram method to successfully detect staff lines on sheet music. We applied a similar method for the scanned piano roll images by projecting the piano-roll image onto its x axis to form a histogram of the holes in each channel. From the x-projection, we process each note-channel histogram to find the center point of each histogram peak relative to the left edge of the piano roll, as illustrated in Figure 2. We take that as defining center lines for each channel. Because the hole-channels are adequately straight and nearly parallel to the edge of the paper, the x projection histogram produces well separated "channel piles". The set of all such channel lines forms a grid on the piano roll in which each channel line is a fixed measured distance from the left edge of the paper. To map these note-center x-coordinates to MIDI note numbers, we use the first note of Waltz in E flat Major, Op.42 No.2 by Frédéric Chopin as an anchor note, namely E♭4 (MIDI note number 63). Under current image resolution, the median gap between each note channel was found to be approximately 33 pixels, which matches the roll specification of 9 holes per inch as mentioned in section 2.

Based on the distribution of the punched holes as well as our manual inspection of the roll image, we further partition the piano roll into three zones: zone 1 on the left includes the bass dynamics, zone 2 in the middle contains all the note information for the 80 keys, and zone 3 spans the treble dynamics, as shown in Figure 1. Our entire collection of Welte-Mignon Licensee rolls appears to be compatible with this template. Thus the template formation based on one example is found to be sufficient. We are able to prescribe the edges of the zones according to the plot of channel histograms generated by the system, as described above and further below.

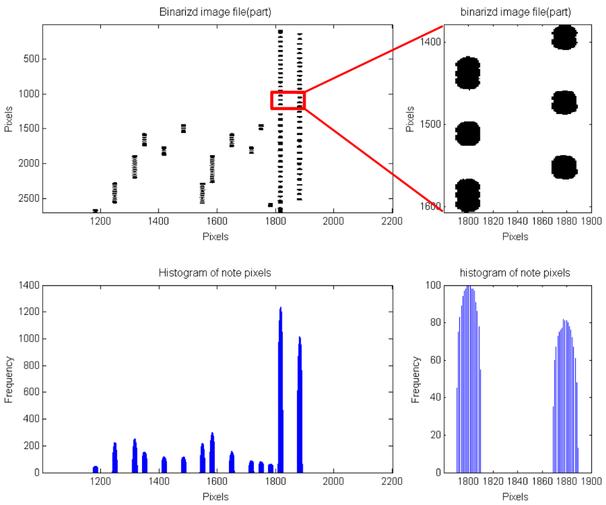

Figure 2. (Top left) Scanned image; (Top right) Zoomed-in view; (Bottom left) Histogram for top left; (Bottom right) Histogram for top right. Note: aspect ratio has been modified for top left and top right images in order to fit into plots.

4.3 Note Identification

We create a *note matrix* based on the channel grid we generated. We first match each hole with a note on the channel grid. We consider a match to occur when a hole intersects with any note on the channel grid. Note that the size of a hole is wider than one pixel. Then we perform a y-projection of the hole to determine the note onset time. The note matrix M is of size 80-by-N, where N is the length of the scanned roll in pixels, and $M_{i,j} = 1$ if a punched hole covers the particular pixel. We then convert the sparse matrix into readable format, row by row, consisting of MIDI note number and note onset times. We scale the note onset time from pixels to milliseconds. We determine the scale factor according to the time information indicated at the beginning of each roll. For example, for a piece with tempo mark 70, the roll should be played at a speed of 7 feet per minute [5]. We thus define the scale factor F to be the length pixels divided by the time it takes to finish the piano roll (in seconds). In the case of the Welte-Mignon Format, $F = 455$ pixels per second.

For long notes, the punching system needs to punch multiple holes closely spaced along the channel grid. Thus to obtain the actual note durations, we define a minimum threshold between two holes as 11 pixels. This threshold was determined empirically based on observations of the reference piano roll. If the gap between holes is smaller than this minimum threshold, the holes are considered to belong to the same note.

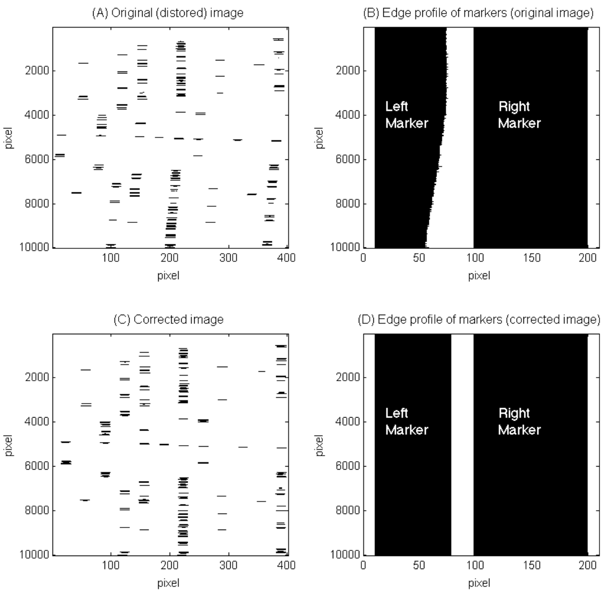

Figure 3. (A) Excerpt from the original (distorted) image (B) Edge profile of the left and right markers for the original image. (C) Excerpt from the corrected image (D) Edge profile of the left and right markers for the corrected image.

4.4 Distortion Correction

Due to instability in the scanner system, some scanned image were warped, as shown in Figure 3A. We fix this distortion by locating reference lines. On each roll, there are two straight bold lines marking the region of the punch holes. We refer to them as *markers*, as shown in Figure 1B. We first locate the two markers in the roll, and then use them as reference point to evaluate if the paper is warped or distorted. Our system iterates through each row and scale the pixels between the two markers such that the markers are enforced to be vertically straight and have a constant distance in between. Thus the imperfection in scanning can be detected and corrected, as in Figure 3C. Figure 3D shows straight left and right marker after the distortion correction.

4.5 Modeling Musical Expressions

Next we decode and model the expression markings on the left and right sides of the piano roll next to the markers. We obtain the location of each expression channel by constructing an expression channel grid similar to the note channel grid, as illustrated in Figure 4.

There are three types of musical expressions in Welte-Mignon piano rolls: constant velocity, changing velocity, and pedal information. The left zone of expression channels correspond to bass notes (notes below F♯4), and the zone on the right specifies the expression for treble notes (notes from G4) with the pedal information controlling the whole register.

For the constant-velocity controls, we simply map each indicated piano-key velocity to a chosen constant MIDI velocity, a seven-bit value between 0 and 127. Based on listening tests and calibration experiments described in the next section, we chose to map the *normal* (default) velocity to MIDI velocity 72, *mezzoforte* to 80, and *forzando* to 88. The piano samples used in this study is the Steinway Grand Piano in Logic Pro X[2].

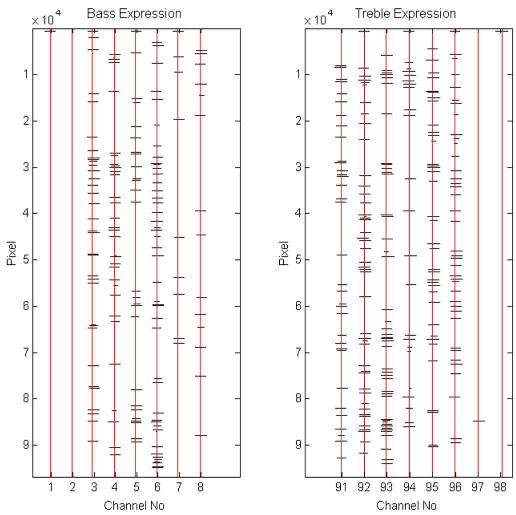

Figure 4. Musical expression holes in black with the expression channel grid in red vertical lines

To model the changing-velocity controls *crescendo* and *decrescendo*, we need some understanding of the expression pneumatics system itself.

In most Welte-Mignon systems, the expression is implemented using *pallet valves* [11]. Specifically, when the system reads a hole on the *crescendo* channel, the *crescendo valve* will be opened, introducing a suction to the expression pneumatic that pulls the pneumatic closed, producing a crescendo in the music. It takes significant time for the air to come into the pneumatic system to take effect. We model this process as an exponential approach to a target value, using one-pole unity-gain lowpass filter having impulse-response time-constant τ that is set to match the observed dynamics.

There are two types of crescendo:

1. a "very slow" crescendo produced by turning on the *crescendo* channel. This control is latching, so that one hole can turn it on.
2. a "fast" crescendo that is *not* latching.

A string of fast crescendo holes can used to speed up the slow crescendo, thereby providing many ultimate crescendo rates, as well as nonuniform crescendos. They are like little bursts of additional suction along the way as the slow crescendo develops.

Let the observed time-constant of the slow crescendo be denoted by τ_s (s for "slow"). Then the slow-crescendo one-pole filter has digital transfer function

$$H_s(z) = \frac{1-p_s}{1-p_s z^{-1}} \tag{1}$$

---

[2] https://www.apple.com/logic-pro/

where its pole p_s is defined as

$$p_s = e^{-T/\tau_s}$$

with T denoting the digital sampling interval in seconds (typically $T = 1/f_s$, where f_s denotes the sampling rate, and $f_s = 44100$ Hz or greater). The time-domain difference-equation used to implement the slow crescendo is given by

$$y_n = (1 - p_s)\, x_n + p_s\, y_{n-1}, \quad n = 0, 1, 2, \ldots ,$$

where x_n is set to the target velocity at sample n, which is *forzando* for a crescendo, and either *normal* or *mezzoforte* for a decrescendo, depending on the last constant-velocity setting.

The fast crescendo is modeled exactly like the slow crescendo, but using a smaller time-constant $\tau_f \ll \tau_s$.

5. EVALUATION

We used MIDI file tools for Matlab[3] to convert our piano-roll matrix into MIDI format, and we synthesized the MIDI file in Logic Pro X using the Steinway Piano software-instrument that comes with Logic. We evaluated the success of our algorithm by comparing the synthesized audio to a recording of the player-piano, both driven from the same piano roll. Due to limited access to the machine and rolls, we only include one roll in our discussion here. However, given that each roll format possesses a fix template, we can assume that it will work for many additional rolls.

5.1 Recording Setup

We set up a recording environment for the player piano in a concert hall. The player we used is called a "push-up" because one pushes it up to a real piano where it plays the piano using padded wooden mechanical "fingers" (see Figure 5), as described further below. We recorded the push-up's performance on a 9' Steinway grand piano. The piano-roll chosen was the Chopin Op.42 Waltz in A$\flat$, performed by Katherine Bacon, and published by Welte in 1924.

Figure 5. The Push-up Player

For the acoustic recording, we set up two cardioid microphones above the Steinway grand piano, one on the left, capturing most of the energy from the bass, and one on the right, for the treble, as shown in Figure 6. The push-up player is aligned at the piano and the roll is set in the player, attaching the lead to the take up spool. The playback speed is set manually on the player. There can be some variation in playback on different players due to subtle differences in condition, however, most features of rolls are reproduced consistently on well functioning instruments. The instrument used in this project has been evaluated by player piano technicians to be in good working condition.

Figure 6. Acoustic Recording Setup of the player piano

5.2 Note Similarity Measurement

An example MIDI file[4] is shown in Figure 7, with the raw image file on the top, and the MIDI file displayed in "piano roll" editor window in the program Logic Pro X for Mac OS X. We can see that the overall shape and trend of the notes are visually identical.

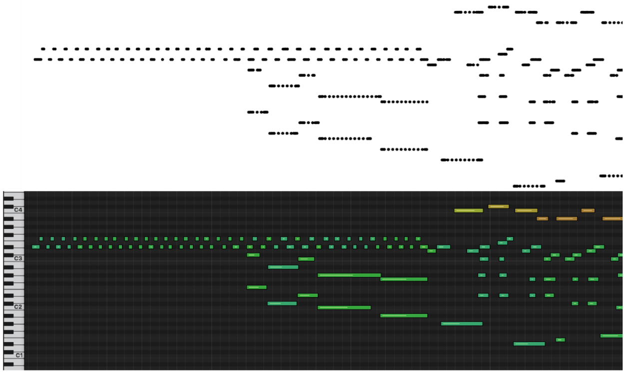

Figure 7. Visual inspection of the scanned image (top) and the "piano roll" image of the synthesized MIDI in Logic Pro X (bottom).

We measure the similarity of the audio content by calculating a similarity matrix [3] of the spectrogram between the MIDI-synthesized audio file and the audio recording. We then apply dynamic time warping [10] to align the MIDI file with audio and retrieve a path between the two. Figure 8 shows the similarity matrix comparing the spectrogram of the audio recording and MIDI-synthesized au-

---

[3] https://github.com/kts/matlab-midi

[4] MIDI synthesized audio file on Chopin Waltz Op.42 No.2 can be heard at https://tinyurl.com/kvyxc3s

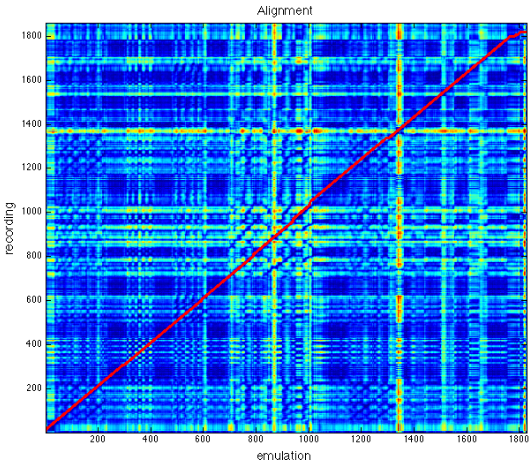

Figure 8. The spectrogram similarity matrix comparing the live audio recording and the synthesized MIDI, with an overlay of the time alignment line (red). Note: a straight diagonal line means perfect alignment.

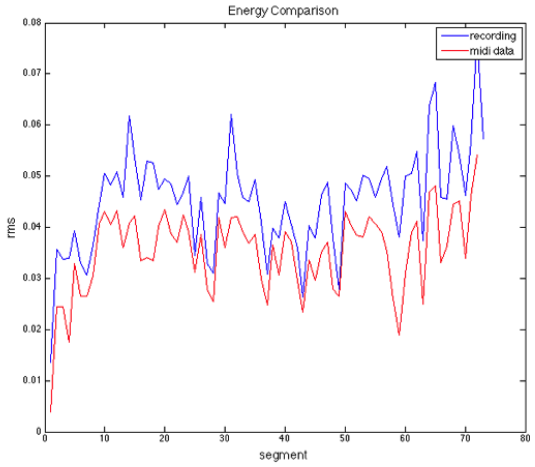

Figure 9. Comparison of the root mean square energy for live audio recording (blue) and synthesized MIDI (red).

dio, with the red diagonal line representing the time alignment between the recording and the MIDI. We see that the diagonal line is near straight and slope 1, meaning that we get almost perfect alignment between the audio and the MIDI. We found that there is some variation and curvature towards the end of the diagonal line and that the overall length of the MIDI emulation is 8-second longer than recording of the player piano.

5.3 Dynamics Measurement

To measure the success of modeling the dynamic expressions, we calculated the root mean square energy of both audio file. As in Figure 9, we see that their dynamic levels are similar, but we see that the original playback has a wider dynamic range than our synthesized MIDI file. We plan to optimize our setting of the dynamic variability to match the acoustic recording, but that will be the center (default) setting of a knob that can be varied from "flat" (no dynamics at all) to "exaggerated" (expanded dynamics). With this control users can adjust to taste. For example, it is nice to be able to make flatter renderings for noisy listening environments such as cars.

Note that every player piano will give a slightly different result due to variabilities in manufacturing and settings. This is another reason to make the end-result easily adjustable.

We do not yet include pedal information because the pedal on the player piano was not working perfectly at the time of our recording. We presently do not have control over how much pedal and pedal delay.

5.4 Discussion

We found that the MIDI file matched the original recording quite well, both visually in the graphs and audibly in recordings. As pointed out in the similarity measurement section, we observe that the MIDI roll is not quite at the same speed as the instrument playback of the roll. The MIDI roll appears to be "slowing down" towards the end compared to the original playback. Our research suggests that this is likely a deliberate design from the factory to compensate for increasing tempo change due to the changing diameter of the spool as it unwinds the paper upon playback. Wider spacing of the holes towards the end of the roll would keep the speed of the playback constant [1]. This would be consistent with our observation which finds the original roll playback faster than the MIDI of the paper roll itself. This observation will be explored with further evidence as more rolls are scanned and digitized.

6. CONCLUSION AND FUTURE WORK

We proposed methods for decoding reproducing piano roll images into MIDI files. We also proposed an apparently novel method for interpreting the expression markings on a piano roll. Though the system is designed to recognize the Welte-Mignon piano-roll format, our note matrix template is adaptable to all other systems with small modifications. However, the expression template is not adaptable from system to system. We also developed models for the Duo-art format. We have not yet evaluated this model with a playback comparison due to limited availability of an appropriate instrument. However, future work is planned for interpreting all piano-roll types, and including comparing different player instruments of the same type in order to measure variabilities. We further plan to create a master punch-matrix for the system types for correcting errors and repunching the piano rolls, thereby "restoring" them. We plan to develop a batch processing system for all the roll images created by the new scanning device that is presently being built for the Stanford Music Library [8]. Finally, we plan to release these digitized piano rolls on the Web as a free online resource.

7. ACKNOWLEDGEMENT

We'd like to thank Anthony Robinson and Jerry McBride for providing valuable information about reproducing piano rolls. We thank the Stanford Archive of Recorded Sound for providing the player piano as well as piano rolls.

8. REFERENCES

[1] Mechanical music digest archives. `http://www.mmdigest.com/Archives`. Accessed: 2017-04-18.

[2] Gustavo Colmenares, René Escalante, Juan F Sans, and Rina Surós. Computational modeling of reproducing-piano rolls. *Computer Music Journal*, 35(1):58–75, 2011.

[3] Jonathan Foote. Visualizing music and audio using self-similarity. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, pages 77–80. ACM, 1999.

[4] Ichiro Fujinaga. *Optical music recognition using projections*. McGill University, 1990.

[5] The Pianola Institute. The reproducing piano - welte-mignon. `http://www.pianola.org/reproducing/reproducing_welte.cfm`. Accessed: 2017-04-15.

[6] Jagat Narain Kapur, Prasanna K Sahoo, and Andrew KC Wong. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer vision, graphics, and image processing*, 29(3):273–285, 1985.

[7] Sasha Leitman. Trimpin: An interview. *Computer Music Journal*, 35(4):12–27, 2011.

[8] Stanford University Libraries. About the player piano project. `https://library.stanford.edu/projects/player-piano-project/about-project`. Accessed: 2017-05-16.

[9] Matteo Malosio, Flavio Pedrazzini, and Perego Niccol. The sisar project. *The Music Box*, 26(6):221–223, 2014.

[10] Meinard Müller. *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Springer, 2015.

[11] Arthur A Reblitz. *Player Piano: Servicing and Rebuilding*. Vestal Press, 1997.

[12] Anthony Robinson. Anthony's roll scanning. `http://semitone440.co.uk/scanner/`. Accessed: 2017-02-17.

[13] Larry Sitsky. *The Classical Reproducing Piano Roll: A Catelogue*. New York: Greenwood Press, 1990.

[14] Wayne Stahnke. Stahnke's roll archival methodology. `http://mmd.foxtail.com/Archives/Authors/Aut318.html`. Accessed: 2017-04-15.

[15] Warren Trachtman. Recovering inherent roll punch matrix spacing information and using it to dynamically self-correct roll scans. `http://www.trachtman.org/rollscans`. Accessed: 2017-03-15.