Hand Writing Recognition System for Musical Notes

Alex Silverman   Luis Jimenez   Michael Nechyba   A. Antonio Arroyo   Eric M. Schwartz

Machine Intelligence Laboratory
Deparment of Electrical and Compter Engineering
University of Florida, Gainesville. FL 32611-6200

Abstract — Hand held devices have recently became available that can scan printed text and create computer-generated speech. These devices are generally designed for people with some type of disability. This paper describes the software portion of a similar device to identify isolated small sequences of hand-written or printed sheet music. The software we designed analyses scanned images of sheet music and identifies musical notes by kind and location on the staff. Our program was designed to identify sets which use the note durations (whole notes, half notes and quarter notes, etc.) and the note frequencies (locations relative to the staff). This paper introduces the method implemented to isolate and recollect the information from the data sets. We describe how the information was used and manipulated through statistical models in order to identify the musical notes. The efficiency of the different models was compared and those with the best accuracy were chosen to be implemented in the final program. The final program was tested with multiple data sets. This paper discussed the results of our program along with possible future enhancements.

I. Introduction

This paper describes a project for an isolated-note, small sequence, hand-writing recognition system. In our program we limitted the note durations to only whole, half and quarter notes. Fig. 1 shows some sample data containing only these note durations.

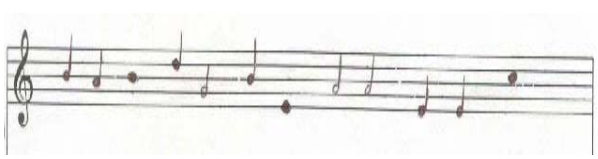

Fig. 1. Sample Data

Presently, there are hand devices that scan printed text and generated audible representations, specifically designed for handicapped people. The objective of this project is to design a scanning program that will identify hand written musical note by kind and location on the staff. Given the complexity hand written music, it was decided to design an initial program that was able to identify only whole, half and quarter notes, the most widely used notes in written music.

II. Musical Notes Recognition System

A. Data Collection

The data was collected by scanning a series of pre-printed staffs, using true-colors; on which different sequences of whole notes, half notes and quarter notes were hand witten. A total of 15 samples per sequence were written along the staff, on different locations up and down, and with little variation of the same note. All, the images were scanned by using a HP scanject 7400C scanner and then saved into image files in PPM format at a resolution of 640 pixels (height) by 58 to 69 pixels (width).. A total of 10 different sample files were used in the experiments describe below.

B. Image Filtering & Note Segmentation

In order to make the data analysis easier, we used a program that converted the image files from RGB format (Fig. 1) to Black and White and then smooth the data. See Fig. 2 for a converted image example. Once the files were converted to black and white a different program was used to erase the staff lines from the image file, leaving just the notes. This new file as shown in Figure 2 provides a relatively clear representation of the notes which was easy to segment.

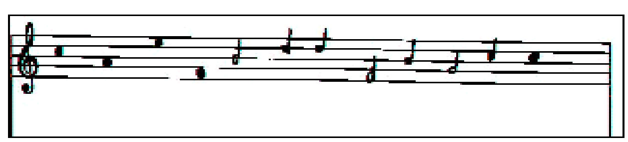

Fig. 2. Black and White Note

The segmentation of the data was done by isolating every note in a 20 by 20 pixels box which would be sent into the statistical program to be analyzed, as shown in Fig. 3. This process was achieved by using our segmentation rogram.

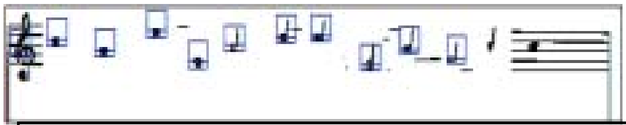

Fig. 3. Notes Segmented

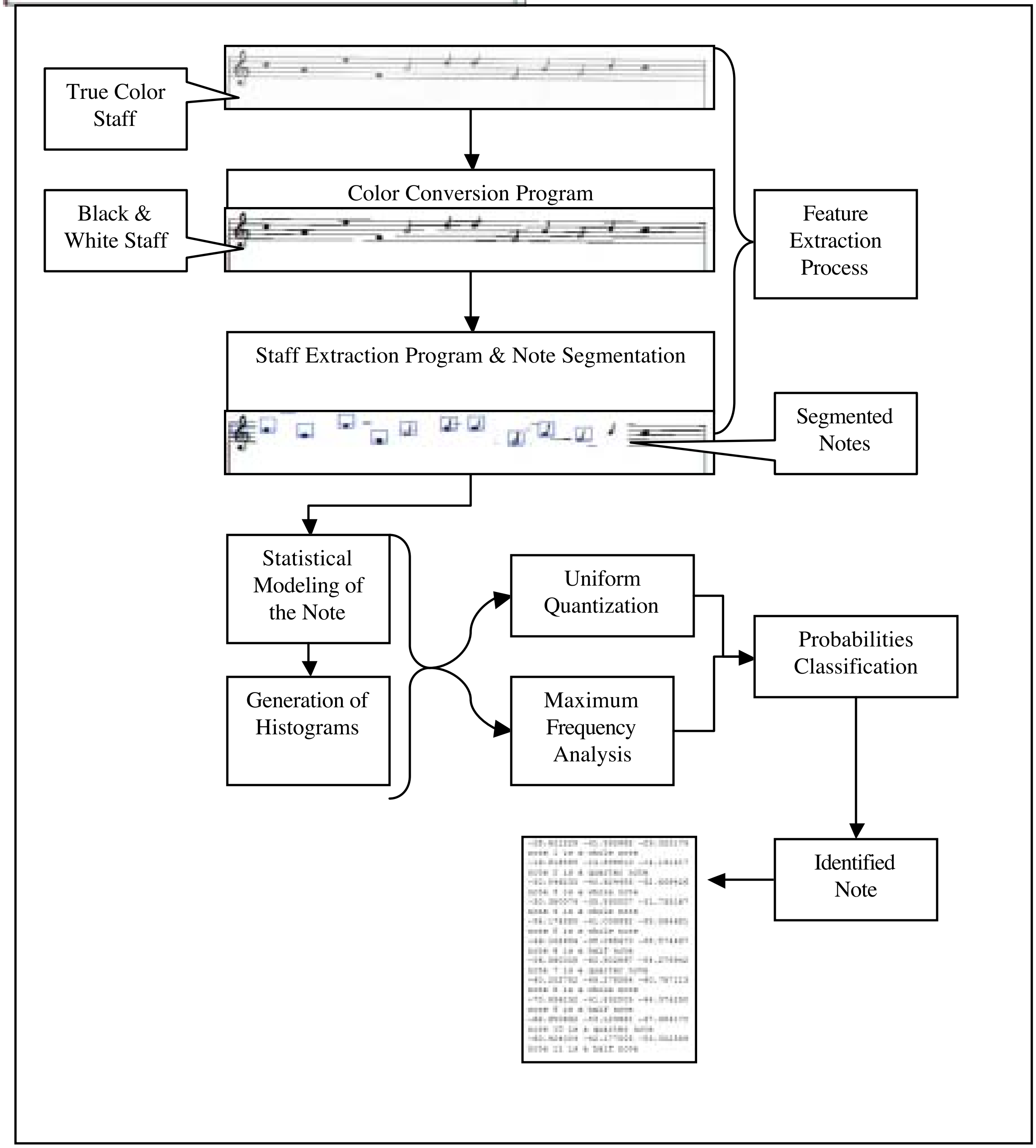

Fig. 4. Analysis Process

C. Training, Testing and Results

Now that the notes were cleaned and segmented, we were ready to extract the features we want to use for our classifier. All of our methods were histogram based methods. The first method we attempted to use was uniform quantization. Different sized cells were tried and the best was found to be splitting the box into a 5 by 5 grid. However, using this method we found there to be too much error in classification of quarter notes and half notes.

Next, we tried Vector Quantization; with Vector Quantization our program could correctly identify whole and quarter notes most of the time. However, it also misclassified about 25% of the half notes as quarter notes. We attempted codebooks from 2 vectors in length to 128 vectors (unnecessary for a 20x20 picture), yet still the best we found was with 16 vectors giving us the 25% error on half notes, and near perfect on whole and quarter notes. The reason for this percentage error can easily be observed by looking at the histograms used in the code book, Fig. 5, Fig. 6, and Fig. 7.

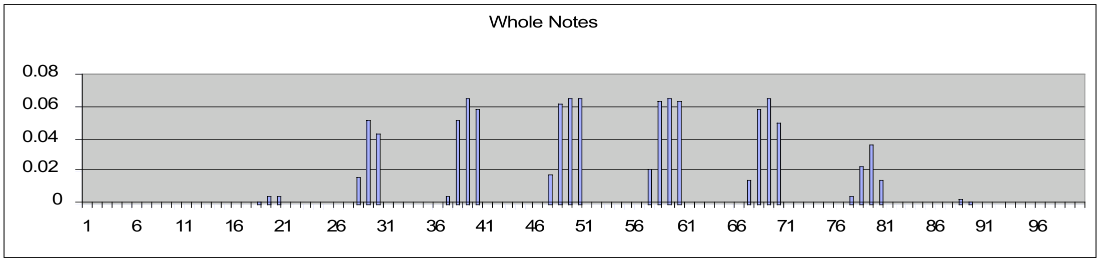

Fig. 5. Whole Notes Histogram

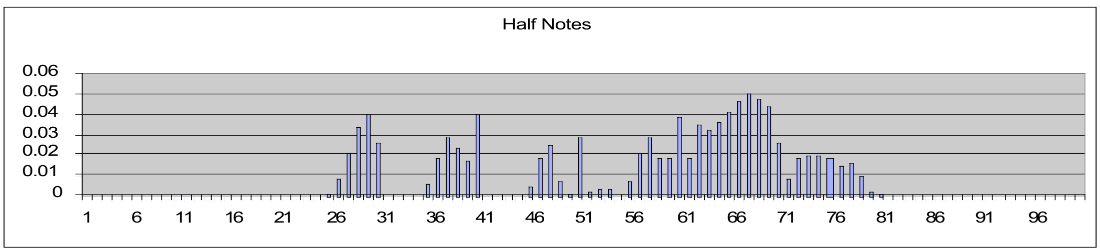

Fig. 6. Half Notes Histogram

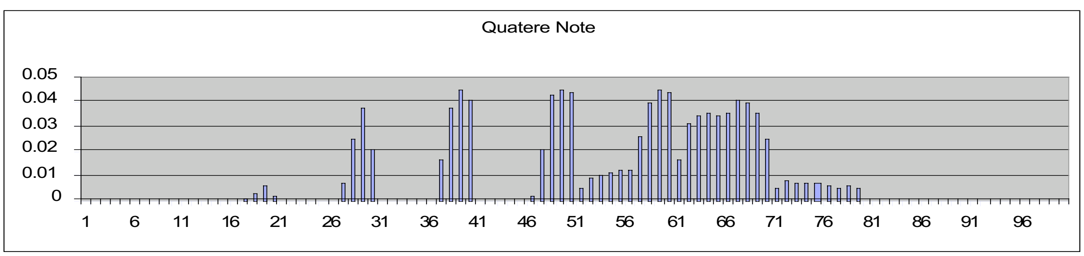

Fig. 7. Quarter Notes Histogram

Finally, the third method we tried is based on 1 dimensional histogram of the Maximum number of consecutive black pixels found in each column of the 20x20 image. Thus the histogram went from 0 to 20 pixels. So, in order to train the histograms, we used training data which consisted of 2 sets of each type of note, drawn on plain white paper. The training data was cleaned up, segmented, and the histograms trained, Fig. 8, Fig.9, and Fig.10. Classifying is done by multiplying the number of columns that fall under a given category times the log probability of that category being generated from the model, and repeated for 0 to 20 black pixels. Now, we had almost 100% accuracy for whole and quarter notes, but half notes were only down to 17% error. While this was not terrible, we still wanted better accuracy.

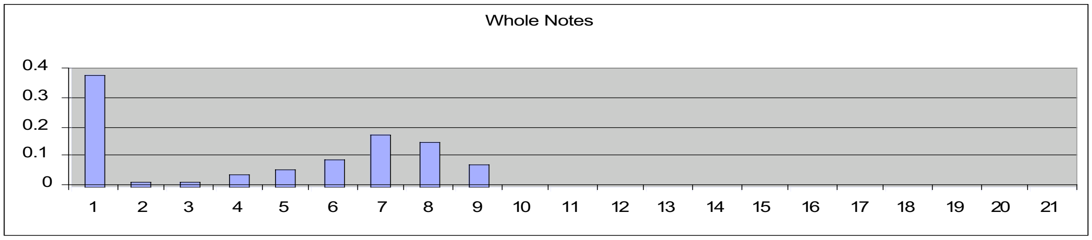

Fig. 8. Whole Notes Histogram

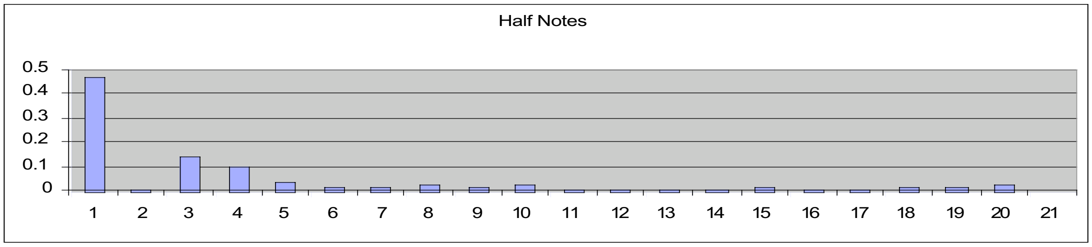

Fig. 9. Half Notes Histogram

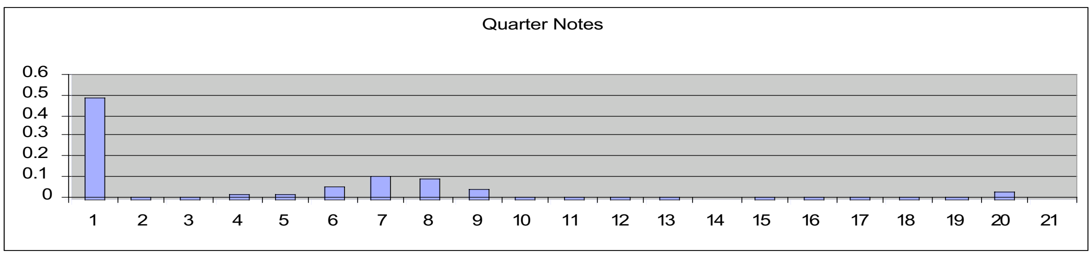

Fig. 10. Quarter Notes Histogram

We then tried slight variations on the aforementioned method; such as using the number of black pixels in a black segment of a column as the feature (thus a half note ideally would have had two semi-short black segments per column, whereas the other method just uses the longest segment per column). In addition, we also tried both of these on rows instead of columns and also rows and columns thus using a two dimensional feature space. However, none of these variations provided any significant increase in accuracy. Lastly, we decided to stick with the analysis of maximum length segment to determine whether we thought it was a whole note of not. To have a higher probability of the correct data being generated from the half or quarter note model, we go into a rule based analysis. This rule based analysis is based on the following method.

First, the program looks down each column in the bottom half of the 20x20 image, for a black segment followed by a white segment, followed by another black segment. If it meets that criterion than for each white pixel in the white segment, it looks across the row to see if the white pixel is part of a white segment surrounded by two black segments. Now, based on the number of pixels that meet both criterions, we use that number versus a threshold value to determine half versus quarter notes. Over our test data, we found that we got 2 notes wrong out of 75 (25 quarter, 25 half, 25 whole), for an over all accuracy of 97.3%.

Table I

Notes	Test	Correct	Incorrect	% Error
Whole	25	25	0	0
Half	25	25	0	0
Quarters	25	23	2	8
Total	75	73	2	97.3

III. Conclusion

In this paper, we trained and tested hand writing recognition system for musical notes using a mixture of vector quantization and maximum frequency of black pixels. All the different method tried for data analysis were described in order to make it clear how we came to the design of our final system. We achieve a classification error of 2.7% over the test data, despite the low resolution of the images used. The combination of different statistical techniques helped us to reduce the error to this low number.

IV. Future Improvements

In the future, we would like to change the last method to a statistical model, using the number of pixels that met the criterion as a feature rather than tested versus a threshold value. Also, once the minimum percentage error possible is achieved the next step would be to add other kinds of notes into program. By doing this, it is possible that the current statistical method to identify the notes would not be enough; therefore, different techniques would be used for the analysis of the new notes. The final goal of this project would be to have a program capable of identifying every different kind of musical note from an image file and being able to correctly interpret the music.