Website

We created a website to collect sample inputs so we have a variety for our tree structure. Everyone's penmanship is different so the more variety of sample inputs we have the more accurate our application will be.

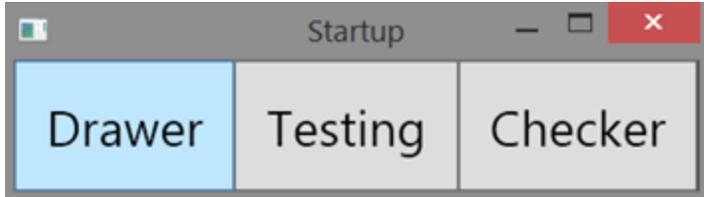You can access the website here: http://kaseypowers.com/Letterform/

Our website randomly chooses a letter for you to draw and then you draw it in the white canvas and click submit, or clear if you wish to start over. Once a user click submit the point data from their sample input and the given letter is stored at the top of a database. Because we are using WordPress there is conveniently a database already integrated into the website so we take advantage of that database.

That table shows how many sample inputs for each letter/digit we have. The table is populated by querying the database.

Application

Checker

The purpose of the checker is to validate the sample user input. The inputs from the website get stored in a database and we retrieve each entry and either delete, or validate it.

We choose to set up a validation checker so we can have complete control over the sample inputs being used to populate our decision tree. If you select Validated Data from the dropdown menu:

You can scroll through the already verified sample inputs.

Drawer

The drawer function is where the user can write a letter and see how our algorithms in place simplify and clean up the data.
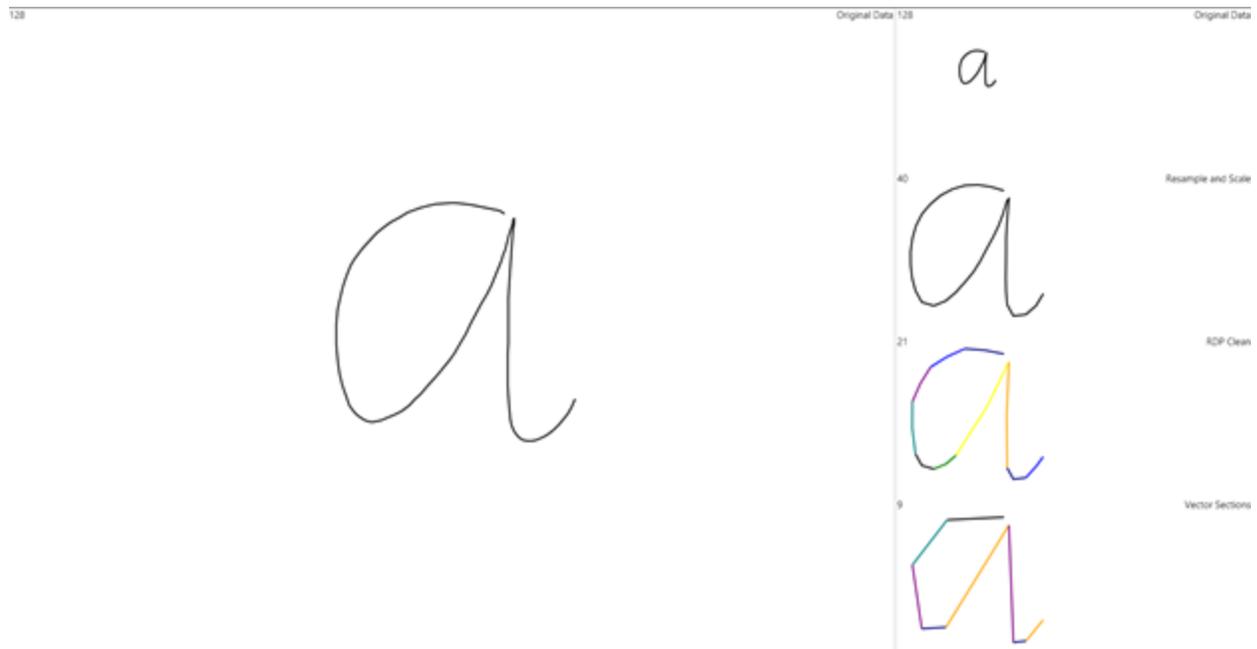
We have a Resampling/Scaling, RDP, and a vector sections algorithm in place to get the best breakdown possible. Those number in the upper left corner of each canvas is the number of points we have simplified the input to, from the above example we have gone from 128 points to 9.

Resampling/Scaling algorithm

Starting with the original recorded data, scale the minimum value to 0 and maximum value to 1. Made more efficient by continuously adding new points on the same scale and checking them to adjust scale as needed.

Next taking the scaled data, resample so that points have a distance of 0.1 units between them. This removes a large amount of the redundant data, keeping the main

shape unchanged. On very messy data some minor cleaning occurs due to the noisy points being close together.

RDP

Using the Ramer-Douglas-Peucker algorithm with a relatively large epsilon value on the output of the previous step, major sections are found then smoothed using a moving average.

Finding major sections of the input was a massive improvement. Previous use of moving average smoothed the entire input, including smoothing out intentional sharp points. RDP allows smoothing the entire input while features large enough to be considered intentional are unchanged.

Vector Sections

Finding the vector sections is the last step. The cleaned input is broken down into a series of slopes. Determining slope by treating two points as a vector and comparing to the 8 directions using dot product. By then combining adjacent points with the same slope, the final output is found.

These sections, making up the building blocks of the character, are then able to be inputted into a decision tree built from a database of sample inputs.

Future

We want to finsih and restructure out tree