

# Counting Strokes in Chinese Characters

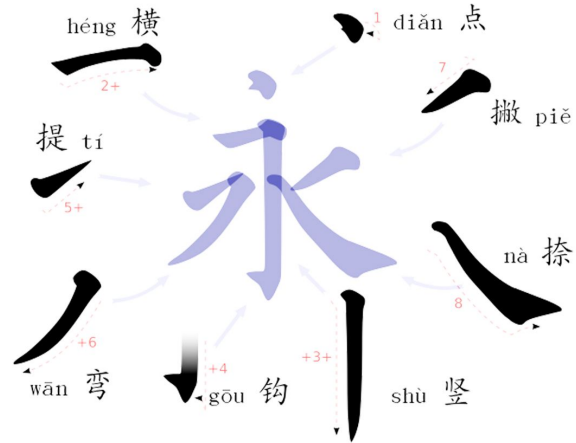
Yien Xu, yien.xu@wisc.edu

Yuqi Lin, ylin273@wisc.edu

Scott Lai, qlai5@wisc.edu

## Problem Statement

As one of the oldest known language system in the world, Chinese is the only prolific writing system that is still in use today (Olson, 2014). Unlike the alphabet or syllabary, Chinese characters are comprised of different strokes, and each character is like a single word in English which has a meaning itself. There are five basic stroke features, including Héng (Horizontal Stroke), Shù (Vertical Stroke), Piě (Down Stroke to The Left), Diǎn (Dot), Zhé (Horizontal Stroke with A Vertical Turn) (Ministry, 2001) (Lavarini & Franco, 1999). Under each basic stroke, there are subordinate stroke features. In total there are 36 stroke features, or principle strokes (See the figures of stroke features in Appendix). These simple stroke features compose tens of thousands of Chinese characters with complex forms.



Eight Principle strokes extracted from 永, means “eternity”.  
The total number of strokes of 永 is 5.

As Chinese international students, we hope to create a method that helps people who are interested in Chinese learn about this magical language. In this project, our goal is to use machine learning algorithms to count the total number of strokes in a Chinese character depicted as an image. We divide our project into three parts. The first part of the project is to fit the training data to make the machine learn how many strokes each Chinese character has. The second part is to use the learned knowledge to predict the number of strokes of a given image of a Chinese character. The third part is to evaluate our model to see how well our model can perform to count the number of strokes of Chinese characters that are not present in our training data.

## Dataset

We need to get an extensive dataset for the purpose of training. We find a dataset containing 15 million 28x28 PNG files of 52,835 Chinese characters with different fonts (Burkimsher, 2018). This dataset is created based on 36 fundamental components, such as “丿” (Piě), “一” (Héngzhé), and “丶” (Diǎn). Burkimsher gets this dataset by generating Chinese fonts from Chinese Font Design, a website containing a great amount of Chineses font collection. We will use a subset (one font) of this collection of 28\*28 pixel images as our inputs.

Another tool we plan to use is the Unicode Han Database (UniHan, 汉) from the Unicode website. The Unicode standard provides encodings of all characters used in the world's written language. UniHan is a

large database, under Unicode standard, containing strokes, structural analysis, and definitions of Chinese fonts (Jenkins, Cook & Lunde, 2018 & Unicode Han Indices, n.d.). In this project, we plan to use UniHan as our database to look up the number of strokes of a specific Chinese Character.

Because the total number of Chinese characters that exist in the world is fixed, our training data and test data both come from our collection of 52,835 Chinese characters. We plan to apply stratified sampling to our dataset to maintain the original class proportion in resulting subsets and thereby split the dataset into the training set, containing 90% of all characters, and test set, containing the rest (10%). Note that the training set will be either split into a combination of training and validation parts or used for cross-validation depending on the type of the model we choose.

### **Research Approach**

We plan to apply three machine learning models to reach our goal - k Nearest Neighbors algorithm (kNN), logistic regression and Convolutional Neural Networks (CNN). Intuitively, a CNN will outperform the other two models and be sufficient enough to count the number of strokes in Chinese characters with relatively high accuracy. The reason why we choose to keep them is that we plan to make them as benchmarks to which later models can refer to.

#### *k Nearest Neighbors*

The kNN algorithm is used for our first benchmark. Each Chinese character in our dataset is represented as a data point in a 784-dimensional space. The number of dimensions, 784 (28x28), results from the number of pixels of one Chinese character image in our dataset. We plan to use Euclidean distance as a distance metric between two data points. In addition, we aim to implement cross-validation method in order to determine the best value of k. We predict that the evaluation accuracy is not likely to be high, given that kNN suffers from the curse of dimensionality. Therefore, our kNN model is merely served as a benchmark.

#### *Logistic Regression*

Apart from kNN, we believe logistic regression is also a good model for benchmarking. Like kNN, we plan to use 784 features in the logistic regression model, where each feature represents one pixel in the corresponding image of one Chinese character. Besides fitting the model with 784 features, we plan to make use of the spatial structure of an image. We plan to train another logistic regression model with max pooling on the training images. Doing so not only reduces the number of features but also increases the utilization of spatial information. However, we are not sure whether max pooling will improve test accuracy or not.





































#### *Convolutional Neural Network*

CNNs are extremely famous for processing image data because they are able to keep the structural information of the image while training. Hence, we plan to use a CNN as a final step to boost our model accuracy. We have not yet decided how many and what kind of layers to use in this model. We will experiment with various configurations of layers since it should not be time consuming to train the model by using the GPU we currently have.

## Expected Outcomes and Future Application

We hope to train models that can count the total number of strokes in a Chinese character given an image of it. We expect that our CNN will achieve the highest level of accuracy, with logistic regression following and then kNN ranking the third. Regarding the real world application of our stroke counting model, we can integrate them into a screenshot software so that Chinese language learners can easily keep track of how many strokes there are in a new character that they have just learned.

## Appendix

	31C	31D	31E
0	 31C0	 31D0	 31E0
1	 31C1	 31D1	 31E1
2	 31C2	 31D2	 31E2
3	 31C3	 31D3	 31E3
4	 31C4	 31D4	
5	 31C5	 31D5	
6	 31C6	 31D6	
7	 31C7	 31D7	
8	 31C8	 31D8	
9	 31C9	 31D9	
A	 31CA	 31DA	
B	 31CB	 31DB	
C	 31CC	 31DC	
D	 31CD	 31DD	
E	 31CE	 31DE	
F	 31CF	 31DF	

This figure shows the 36 stroke features of Chinese characters (Unicode, Inc.).

## References

Burkimsher, P. (2018, June 25). Making of a Chinese Characters dataset – Noteworthy - The Journal Blog. Retrieved from <https://blog.usejournal.com/making-of-a-chinese-characters-dataset-92d4065cc7cc>

Chinese Calligraphy. (n.d.). [Figure]. Retrieved from <http://jmsc.hku.hk/jmsc61110/chinese-calligraphy/>

Jenkins, J. H., Cook, R., & Lunde, K. (2018, May 18). Unicode Han Database (UniHan). Retrieved October 25, 2018, from <http://www.unicode.org/reports/tr38/#N10046>

Lavarini, D., & Franco, A. D. (1999). Learn Chinese. Retrieved from <https://www.clearchinese.com/chinese-writing/strokes.htm>

Ministry of Education of the People's Republic of China. (2001, December 19). Chinese Character Turning Stroke Standard of GB 13000.1 Character Set. Retrieved from <http://www.moe.edu.cn/ewebeditor/uploadfile/2015/01/12/20150112170016626.pdf>

Olson, D. R. (2014, March 14). Chinese writing. Retrieved from <https://www.britannica.com/topic/Chinese-writing>

Unicode, Inc. (n.d.) The Unicode Standard 11.0. Retrieved from <http://unicode.org/charts/PDF/U31C0.pdf>

Unicode, Inc. (n.d.) Han Indices. Retrieved from <https://www.unicode.org/versions/IICoreRSIndex.pdf>