

Tegaki Project: Open-Source Chinese Handwriting Recognition (Presentation for Open Source Developer Meetup #1 organized by Open Source Hong Kong)

Cheung Wai Ho, Chris

chriscsheungnf@gmail.com

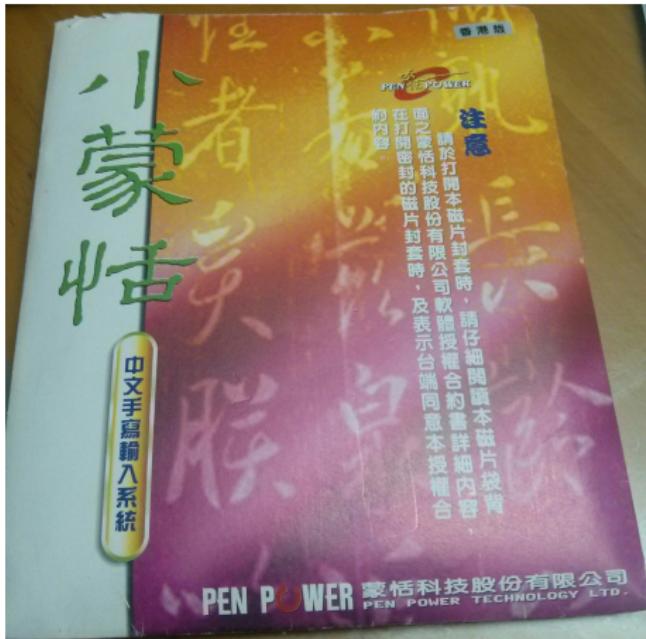
May 11, 2017

Self introduction

- MPhil in Economics, Year 1 student @ CUHK
- Like programming and the concept of open source
- Part-time teaching assistant
- Assisted teaching of Python for UG students majoring in Economics @ CUHK

Discovery of Tegaki Project I

When I am young, I used 小蒙恬 in Windows to type Chinese



Discovery of Tegaki Project II

- Under Linux, I mostly use ibus-table-cangjie3 to type Chinese
- But, sometimes, I do not know how to type some Chinese characters in cangjie3
- Only when I can connect to the Internet, I can use <http://hanzi.unihan.com.cn/QOpen> instead

The screenshot shows the UniHan QPen Web interface. At the top, there is a navigation bar with links for '汉字网' (Hanzi Network), '汉字差异性比较' (Hanzi Difference Comparison), '在线工具' (Online Tools), '汉语教学' (Chinese Language Teaching), '汉字研究' (Hanzi Research), a search bar with placeholder '检索汉字或unicode' (Search for Hanzi or Unicode), and a '检索' (Search) button.

The main content area has a green header titled '巧笔简介' (Introduction to QPen). It contains text about the product, mentioning it is developed by Beijing Shuwen Technology Co., Ltd. and is based on the UniHan QPen software, with features like high recognition accuracy and ease of use. It also notes its compatibility with various platforms and its use of HTML5 technology.

Below the introduction is a search bar containing the characters '世界你'. To the right of the search bar are two small buttons: a red 'X' and a blue square with a white icon.

The central part of the interface is titled '巧笔输入法' (QPen Input Method). On the left, there is a large input field showing the characters '好' and '好' with stroke order guides. To the right of this is a 3x6 grid of characters:

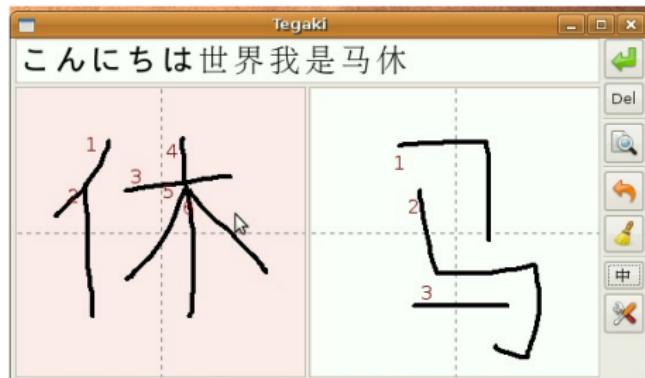
| | | | | | | |
|---|---|---|---|---|---|---|
| 好 | 虹 | 妇 | 奸 | 灼 | 奶 | 妈 |
| 虹 | 如 | 奴 | 妍 | 好 | 妃 | 姐 |
| 妇 | 仔 | 奸 | 汙 | 妞 | | |

Below the grid is a checkbox labeled '识别CJK+汉字 (需字库支持)' (Recognize CJK+Hanzi (Requires font library support)).

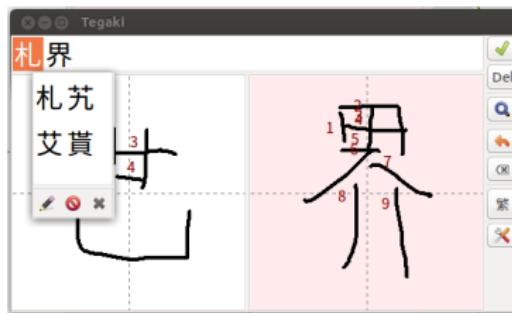
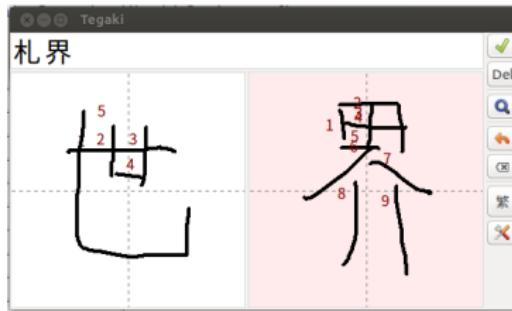
At the bottom of the interface are several small buttons: a red '撤销' (Undo), a blue '重写' (Redo), and other navigation icons.

Discovery of Tegaki Project III

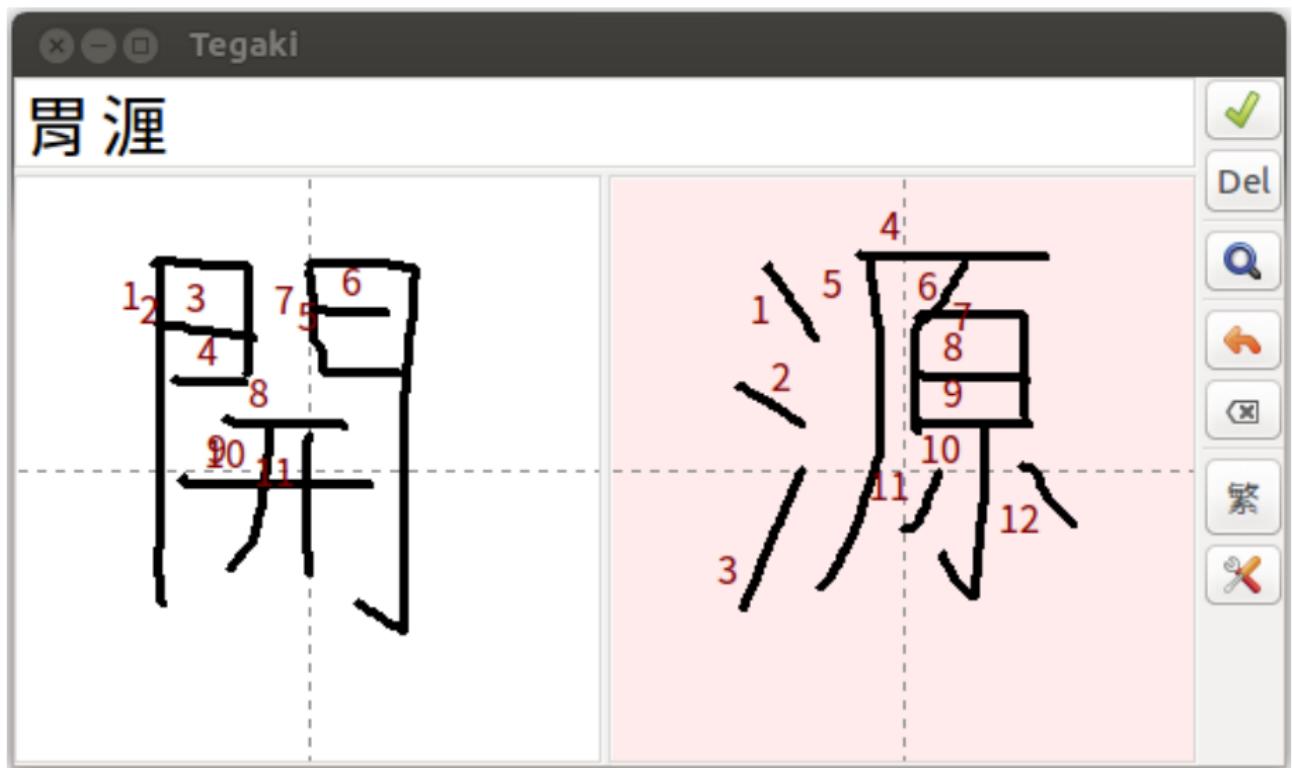
- Local Chinese handwriting input under Linux? Tegaki [1]:
 - is free and open-source
 - is multi-platform
 - focuses on Chinese (simplified and traditional) and Japanese characters
 - supports 2 different recognition engines (zinnia and wagomu)
 - aspires to work on both desktop-PCs and mobile devices



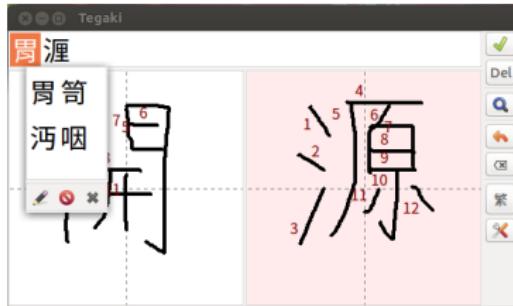
Problem: Bad Chinese (traditional) handwriting recognition I



Problem: Bad Chinese (traditional) handwriting recognition II



Problem: Bad Chinese (traditional) handwriting recognition III



Solution: Bad Chinese (traditional) handwriting recognition I

- Traditional Chinese model (Zinnia engine) available at <https://github.com/tegaki/tegaki/releases/download/v0.3/tegaki-zinnia-traditional-chinese-0.3.zip> is used
- The model seems to use rare Chinese characters for training
- Why don't we as local Chinese train our own model?

Solution: Bad Chinese (traditional) handwriting recognition II

- In Tegaki Project, there is tegaki-train, allowing us to create our own model



Solution: What I have done

- Create model for Arabic numerals as a test
- Scrap common Chinese (traditional) characters from Chinese Character Frequency Statistics for Hong Kong, Mainland China and Taiwan at <http://humanum.arts.cuhk.edu.hk/Lexis/chifreq/>
- Combine these characters into one UTF-8 file for importing into tegaki-train

Another problem: I am doing it wrongly or tegaki-train is buggy

After I have imported these common characters into the program, I get the error:

```
Traceback (most recent call last):
  File "./tegaki-train", line 654, in _on_open
    charcol = dialog.get_character_collection()
  File "./tegaki-train", line 572, in get_character_collection
    open_character_collection(self.get_filename ())
  File "./tegaki-train", line 577, in open_character_collection
    return CharacterCollection(filename)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 347, in __init__
    self.read(path, gzip=gzip, bz2=bz2)
  File "/usr/lib/python2.7/dist-packages/tegaki/character.py", line 996, in read
    parser.ParseFile(file)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 1026, in _start_element
    self.add_set(self._curr_set_name)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 596, in add_set
    self.add_sets([set_name])
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 606, in add_sets
    self._em("INSERT INTO character_sets(name) VALUES (?)", set_names)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 365, in _em
    return self._c.executemany(req, *a, **kw)
sqlite3.ProgrammingError: You must not use 8-bit bytestrings unless you use a text_factory that can interpret 8-bit bytestrings (like text_factory = str). It is highly recommended that you instead just switch your application to Unicode strings.
```

Another problem: I am doing it wrongly or tegaki-train is buggy

After I have provided sample for a Chinese character, I get the error:

```
Traceback (most recent call last):
  File "./tegaki-train", line 851, in _on_add_sample_done
    self._charcol.append_character(charset, character)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 778, in append_character
    self.append_characters(set_name, [character])
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 786, in append_characters
    self.append_character_rows(set_name, rows)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 795, in append_character_rows
    VALUES (?, ?, ?, ?, ?) "", tupls)
  File "/usr/lib/python2.7/dist-packages/tegaki/charcol.py", line 365, in _em
    return self._c.executemany(req, *a, **kw)
sqlite3.ProgrammingError: You must not use 8-bit bytestrings unless you use a text_factory that can interpret
8-bit bytestrings (like text_factory = str). It is highly recommended that you instead just switch your
application to Unicode strings.
```

My plan

- Try to resolve my problem or tegaki-train's problem with some of you if possible
- Start to train our own Chinese (traditional) model using tegaki-train with the help of Hong Kong people and Taiwanese interested in this project

Joining this project

- You can obtain tegaki-train by:
 - git clone <https://github.com/tegaki/tegaki.git>
 - Move to the folder "tegaki-train", where the source code locates
 - To install tegaki-train: sudo python setup.py install
 - To open tegaki-train: ./bin/tegaki-train
- My project is at <https://github.com/chrischeungnf/tegaki-zinnia-traditional-chinese-local>

Reference



Tegaki Project

Tegaki - Open-Source Chinese and Japanese Handwriting Recognition
Retrieved from: <https://tegaki.github.io/>