6-2022

Corrective Feedback Timing in Kanji Writing Instruction Apps

Phoenix Mulgrew
Union College - Schenectady, NY

Corrective Feedback Timing in Kanji Writing Instruction Apps

Phoenix Mulgrew

March 17, 2022

Abstract

The focus of this research paper is to determine the correct time to provide corrective feedback to people who are learning how to write Japanese kanji. To do this, we developed a system that is able to recognize Japanese kanji that is handwritten onto an iPad screen and check for errors such as wrong stroke order. Previous research has achieved success in developing similar systems, but this project is unique because the research question involves the timing of corrective feedback. In particular, we are looking at whether immediate or delayed corrective feedback results in better learning.

Contents

1 Introduction **1**

2 Background and Related Work **3**
- 2.1 Kanji recognition approaches . . . 3
  - 2.1.1 Vision Based Approaches . . . 3
  - 2.1.2 Gesture Based Approaches . . . 4
- 2.2 Instruction Systems . . . 5

3 Methods and Design **5**
- 3.1 Creation of writing system . . . 5
- 3.2 Error types and corrective feedback . . . 6
- 3.3 Experimental Design . . . 6
  - 3.3.1 Pre-Test . . . 7
  - 3.3.2 Learning by Copying . . . 8
  - 3.3.3 Learning from Memory . . . 8
  - 3.3.4 Post Test . . . 10
  - 3.3.5 Retention Test . . . 10
  - 3.3.6 Data Collection . . . 10
- 3.4 Determining success . . . 10

4 Results **10**

5 Future work **13**

6 Conclusion **13**

1 Introduction

Beautiful writing plays an important role in Japanese culture, a role which does not exist in Western culture. Handwriting is thought to reflect one's personality, but not in the same way as Western handwriting

Figure 1: Kanji showing the variety of line types. A is the kanji for ”river”. B is the kanji for ”hold”. C is the feeling for ”feeling”.

analysis where the focus is on the physical characteristics and patterns of handwriting. Rather, by copying revered models and using creative individuality, one's education, cultural refinement, and aesthetic awareness are conveyed through ones distinctive handwriting style. The applied speed, pressure, and rhythm are said to reflect the writer's state of mind. In fact, revered historical writings are often considered works of art to be displayed in museums, and they demonstrate that beauty is often the principle intention when writing Japanese characters, sometimes even at the expense of legibility.

One of the biggest difficulties facing students who are trying to learn Japanese is mastering the writing system of kanji. Although Japanese has three writing systems (kanji, hiragana, and katakana), kanji is the most difficult to learn due to the number of characters and their higher complexity. This is especially true for students whose native language is English because they have no background in a writing system not based on phonetics or one whose characters share no resemblance to latin characters. However, in order to write or read native Japanese texts, students must memorize about 2,000 characters. Furthermore, in addition to learning the pronunciations and meanings of kanji, students must learn the stroke order in which each kanji is written. For many students, this seems pointless and tedious, but practicing the correct stroke order not only helps students memorize all the characters, but also helps them write kanji elegantly, especially once they get good enough to write characters quickly. These include lines that fade out versus lines that come to a stop, and "upticks" where the writer is moving their pen to the next stroke and it leaves a little swish. You can see the different line types in the kanji pictured in Figure 1 when you compare the three lines in (A), particularly the leftmost one. You can also see the upticks in the bottom right in (B) or the bottom left in (C).

Many students are taught the stroke order of kanji by a Japanese language teacher in the classroom, but from personal experience we know that outside of the classroom is when most learning takes place and bad habits form, mainly because of how many times you have to practice writing each character before you have it memorized, and it's easy to get lazy and forgo what you were taught in class.

Because of the complexity of kanji, students will make mistakes when learning how to write them. In addition, students need to receive feedback on their kanji writing so they can learn correctly and improve their technique. If not enforced properly, it is very easy to pick up bad habits or learn kanji incorrectly. Corrective feedback are the critiques that a student receives when they make a mistake. In regards to the timing of feedback, there are two types: delayed and immediate. Delayed feedback is feedback given after the user has written each radical, while the immediate feedback was given after each stroke. The benefits to immediate feedback are that each stroke of the kanji depends on other parts of the kanji being correct. So for example, say the user started writing the kanji from the bottom instead of from the top. In the immediate feedback group, they would receive this feedback immediately and not have time to remember the incorrect way or build bad habits. On the other hand, in the delayed feedback scenario, users may write the entire radical incorrectly and be presented with multiple error messages. In addition, users will have more time

to process their mistakes and remember why they are wrong.

Given that each timing method has its benefits and drawbacks, we hope to determine the correct time to give feedback to users. Currently, there are few (free) apps for smartphones that help students learn how to write kanji outside of the classroom. This is a missed opportunity because smartphones can be such a help in ensuring students learn how to write kanji correctly. Unlike with regular pencil and paper, apps have already been developed that are able to recognize what kanji was written on the screen and algorithms that can tell if a student wrote a kanji in the correct stroke order. We want to look at the effect of the timing of corrective feedback in such apps.

Our goal was to create an app that teaches students how to write kanji well by building on this previous work. The app gave users feedback on the order that they drew the strokes in and made sure that the strokes were drawn correctly. For this app, feedback was given in written form, with the stroke number and how they can fix it. So for example, if a student starts a stroke in the wrong position, the error message would say "Please redraw stroke 2 with the correct starting position". Examples of both delayed and immediate corrective feedback can be seen in Figure 4. Through these error messages, the user will be able to correct their writing to match the correct kanji.

We ran a study on Union College Japanese language students where we gave users five kanji to learn. For our study, we divided the subjects into two groups. The first group received corrective feedback after every stroke. The second group received corrective feedback after they drew every radical. In particular, the research question was *Does immediate or delayed corrective feedback result in a better understanding of how to write Kanji?* Our hypothesis was that the delayed feedback group would perform better than the immediate feedback group on the post test.

2 Background and Related Work

In Japanese language, kanji are characters adopted from the Chinese language that are used alongside the other scripts, hiragana and katakana. While hiragana and katakana characters are similar to an alphabet where each character represents a sound, kanji often represent not only one (or more) sounds, but also a noun or idea. This is one of the reasons why learning them is so difficult for native and non-native learners alike. In addition, kanji often have more strokes than the other scripts. But many times the strokes are not random or unpatterned. Rather, kanji are often composed of simpler kanji or structures called radicals. The uses and different combinations of these radicals is what creates the different kanji. For example, in figure 2 take the kanji in (A), which means "mouth". This is not only its own kanji, but it also a radical used in other kanji such as the kanji in (B), meaning say, and (C), meaning problem. Additionally, more complex radicals can be built from the simpler ones, like how the kanji (B), say, is used in (D), language, and (E), speak. Learning radicals can help students memorize kanji because the same writing pattern is reused in multiple kanji, which means that the student doesn't have to memorize all the strokes from scratch.

2.1 Kanji recognition approaches

There is a vast amount of research into character and writing recognition by computers divided between offline and online recognition. In general, offline recognition refers to doing analysis after all the data has been collected, while online recognition refers to analyzing the data as it is being created. Knowing this, we can apply it to character recognition algorithms in the following ways. Offline character recognition refers to receiving the input as some sort of image or picture from which text is produced (as all the data is analyzed at once, after it has been collected). Meanwhile, online character recognition takes information such as motions, button presses, pen up and pen down, and other events to produce text and analysis in real time. Due to the nature of this paper being about kanji recognition as the user inputs the strokes, we will only be concerned with previous research in online kanji recognition systems.

2.1.1 Vision Based Approaches

Vision based approaches are kanji recognition systems that focus on recognizing the shape of a written kanji in order to recognize which kanji it is. These approaches do not consider the means by which the user

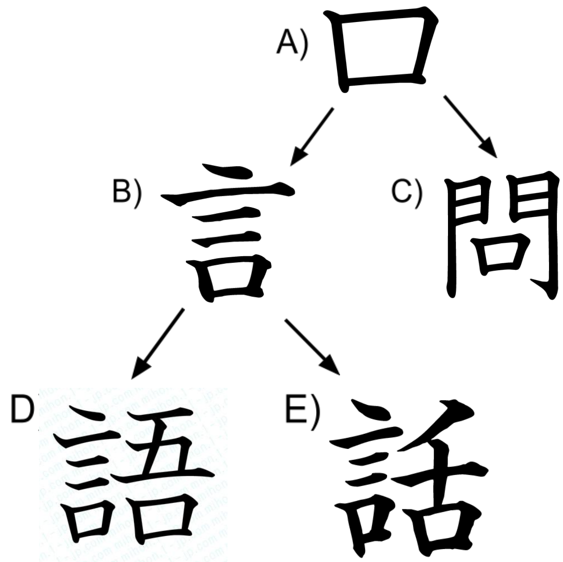

Figure 2: Kanji showing the progression on Kanji and radicals. The kanji meanings are as follows: A) mouth B) say C) problem D) language E) speak/talk

entered the kanji and as such they do not have to deal with features such as individual stroke order. Instead, the character as a whole is considered when trying to identify it and as such can be either offline, such as using a picture as input, or online, such as using handwritten input but running the algorithm after the user has finished. One such offline system is the one created by Korpi et al. [4], where they developed an app that can translate pictures of kanji the user takes and save information about it for the user to study later. When they involve writing, such systems are ideal for native users because the accuracy will not decrease when users write with a less common stroke order or write sloppily. However, from personal experience, this sort of kanji recognition approach is very unhelpful for students trying to learn Japanese because of the lack of feedback provided on the techniques they need to master. If these sorts of systems are used with learners, there is a high risk that a student will write a kanji incorrectly but the system will still correctly recognize the kanji. This would not only impede a student's progress but could also negatively reinforce bad habits.

2.1.2 Gesture Based Approaches

As opposed to vision based approaches, gesture based systems retain information such as stroke order when recognizing a kanji the user has written and so are online approaches. Previous work by Sezgin et al. [6] demonstrated a successful structured symbol construction technique for Chinese kanji through the implementation of a radical recognition system that uses geometric-based sketch recognition. In addition, research work by Chen et al. [2] showed that the development of a stroke order assessment system is achievable. Work by Junjian Tang and Jun Guo [8] showed that a stroke recognition algorithm could determine if the inputted stroke order was correct. These systems generally follow a path as such: First, a sampling of points is collected, and from those points distinct lines and line features (such as intersections and curves) are extracted. Stroke order can then be taken into account as well. Next, hierarchical structures and radicals are composed based on the lines before finally the kanji as a whole is analyzed. However, these systems are not designed to give any output to the user except for recognizing the character and/or turning their handwriting into text. Because of this, when these systems cannot recognize a character or incorrectly guess a kanji, the user does not know whether the issue lies with how they wrote the kanji (as in incorrect stroke order or improperly drawn lines) or whether they drew the wrong character completely. If a student is solely relying on this sort of system to learn how to write kanji, based on personal experience, they will likely become easily frustrated by this lack of error feedback and it can turn learning into a guessing game

of trying to figure out what they did wrong.

2.2 Instruction Systems

With how complex teaching kanji is and how much individual practice is required to master writing, computerized instruction systems are being seen as a helpful tool to help students learn. In general, there are two types of systems: input-focused, where the system provides explicit instruction and comprehension exercises, and output-focused, where the system provides the same instruction together with production exercises. Research by Margherita Berti [1] demonstrated that instruction systems that are output-focused rather than input-focused are more effective in the acquisition of a second language. However, for many kanji acquisition systems, the focus is on learning meanings, pronunciation, and vocabulary words containing each kanji rather than the writing. In addition, research by Lin et al. [5] showed the improvement in memorization of kanji can be significantly aided through the use of mnemonic stories. Kanji meaning, vocabulary, and pronunciation are important to learn if one wants to master kanji however they leave out the fact that at its core kanji is the writing component of the Japanese language. The failure to incorporate writing into these systems is a hindrance to truly mastering kanji.

There has been previous research work that uses gesture or vision based approaches to recognise students' handwritten characters. Previous research work in the Hashigo paper [7] developed a computer-assisted instruction (CAI) system that not only recognizes students' strokes but also provides feedback on their handwriting. In order to evaluate their system, they had two groups of students use their system, one familiar with kanji and one not. The system was able to identify that the advanced users wrote kanji correctly (stroke order wise) while the novices did not. Then, they had the novice users undergo instruction by their training system which had a preview, learn, and review section for each lesson. They then had those users take another test and they compared the accuracy results with the results from the first test they took. But, their work focused on building an artificial intelligence system to teach the student how to write kanji and did not focus on determining the effect of the timing of corrective feedback. Despite some of these systems being developed for Chinese characters, the concepts and methods would be the same for Japanese kanji given their similarity.

3 Methods and Design

This research had two main steps that corresponded to different parts of the project.

3.1 Creation of writing system

The first step was the creation of a system where the user is allowed to hand-write kanji onto the screen and the algorithm analyzes it for accuracy. To do this, we wanted to implement the methods used in the GRAIL algorithm [3] but expand their work to include accuracy checks on all the small nuances of kanji such as "upticks". The method they follow is as such. First, the user's stroke is thinned. This is when the user's stroke is analyzed in the background to remove extra data points that lie within a certain range of the last point made. This is important to do because it's possible to generate lots of data points, especially if the user is drawing slowly. However, the app shouldn't incorrectly find a corner because the user has drawn many points close together. Next, each line is given a direction. As Groner [3] demonstrated, only four directions were necessary to be able to differentiate between different lines: up, down, left, and right. After the points have direction, the app will find the location of corners. A corner is found whenever the pen moves in the same direction for two points, changes direction by at least 90 degrees, and then moves in the same direction for two or more points. Finally, the last step is to analyze size and position features of the line. These involve gathering the starting and stopping points of the line, and determining features of the stroke like the width, height, and aspect ratio. These features allow us to answer questions like, "Would you describe the symbol as being fat or skinny?" As a stroke is drawn, its x (horizontal) and y (vertical) extremes are continuously updated.

3.2 Error types and corrective feedback

When a user is learning how to write each kanji, the types of errors and corrective feedback are based on the mistakes they make. This included:

1. Wrong stroke order - if they write the correct strokes but in the wrong order
2. Too many strokes - if they draw the kanji correctly but then add an additional line
3. Too few strokes - if they press the submit button and they have not written all of the necessary strokes
4. Wrong starting position of stroke - they place the pen down in the wrong position relative to the rest of the kanji
5. Wrong ending position of stroke - they lift the pen in the wrong position relative to the rest of the kanji
6. Lack of an expected corner - they were supposed to draw a corner but they didn't
7. Corner in wrong location in stroke - they drew a corner but it was not in the correct position relative to the rest of the kanji
8. Additional corner where there shouldn't be - they added an additional corner to a line where there shouldn't be one
9. Wrong line shape - the vertical and horizontal extremes of the stroke do not match how the stroke should be drawn (ie not too wide, too tall, etc)

When the user makes a mistake, the system presents them with an error message that tells them what they did wrong and how they can fix their mistake. Users are either given this message after each stroke or after each radical, depending on what group they are in. The user only "passes" each kanji after they are able to write each kanji without making any mistakes.

3.3 Experimental Design

To test the effectiveness of the app, we recruited volunteers from Union College Japanese classes at the 100 level, 200 level, and 300 level. Each volunteer was tested on the same kanji - the difference was if they received feedback immediately after they make a mistake or after they complete each radical. In total, the users were taught 5 kanji, with one kanji used as training. Here, training meant for them to have a practice kanji for them to get used to the app which was not used in the study. The kanji in the experiment were separated into three groups: the group that should already be familiar to both 200 and 300 level students, the group that should be new to just the 300 level students, and the group that should be new to all groups. The kanji we used can be seen in Figure 3

The study was split into five sections: Pre-test, Learning by Copying, Learning from Memory, Post-test, and Retention test. Each stage is as follows:

1. Step 1: Pre-test - participants were tested on their kanji to establish a baseline. If they didn't know, they could just press submit. It was ok to guess or only write part of the kanji if they didn't remember the entire kanji.
2. Step 2: Learning by Copying - participants were taught 5 kanji. The kanji they were expected to write were displayed on the screen with the correct stroke order. There was a box for them to write the kanji in. Participants were instructed to try and make their kanji large enough to fill the box with their answer. They were also given feedback on their kanji writing. They received feedback [after every stroke, after each radical], as seen in Figure 4. Participants had to write each Kanji with no mistakes before they could progress to the next kanji.

A. Familiar to all levels: 水 (only used for training)

B. Familiar to 200 and 300 levels: 前

C. Familiar to 300 level: 深, 調

D. Familiar to none: 報, 氏

Figure 3: The kanji that were included in the study. a) were the kanji familiar to all students, and only used for training. b) were unfamiliar to 100 level, familiar to 200 and 300 levels. c) were unfamiliar to both 100 and 200 level, but familiar to the 300 level. d) were unfamiliar to everyone.

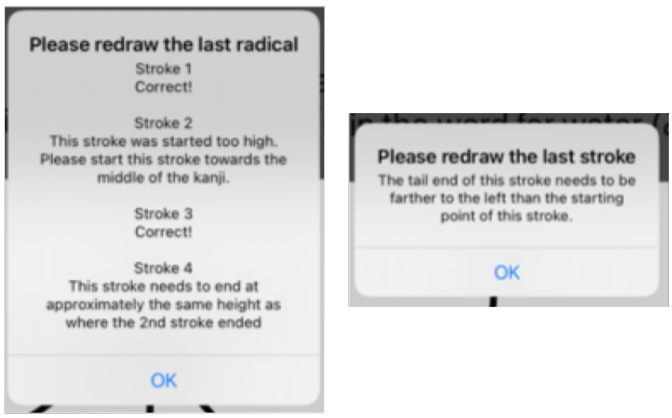

Figure 4: Sample example corrective feedback messages. The left message is what appears for delayed feedback while the right message is what appears for immediate feedback users.

3. Learning from Memory: Participants were taught the same kanji as in Learning from Copying, except this time there was no image for them to reference. Instead, they had to write the kanji from memory. However, if they couldn't remember a kanji, a hint button would appear after they had written 5 strokes incorrectly. When pressed, the kanji they were supposed to be writing would appear on the screen until they started drawing again. Participants were still given feedback [after each stroke, after each radical]. They had to write the kanji correctly before they could move on.

4. Post Test: Participants were asked to write the kanji they were taught. There were no error notifications. They had to write the kanji completely from memory but they could skip kanjis if they couldn't remember how to write one.

5. Retention Test: Participants were asked to return one day later and retake the post test to see how much they retained over 24 hours. There were no error notifications but they could skip kanjis if they couldn't remember how to write one.

3.3.1 Pre-Test

During the pretest, users saw a screen with the name of the kanji and a box for them to write the kanji in. They were asked to write all five kanji in random order. The app kept track of:

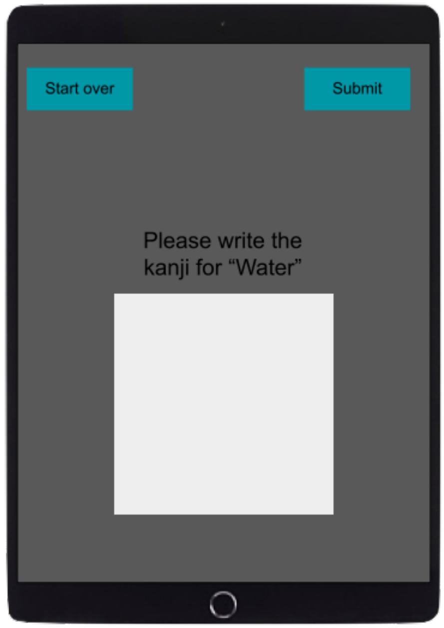

Figure 5: This is how the iPad looked for the Pre-test, Learning from Memory, and the Post Test.

1. If they start writing, how many of the strokes did they get correct and incorrect for their final attempt.
2. For the strokes they got incorrect, what was the reason.
3. Takes a screenshot after they press the submit button.

The reasons for the errors are the ones mentioned above. Users had the option to skip the kanji, meaning they got all the strokes wrong. The pretest looked like figure 5.

3.3.2 Learning by Copying

For this step, users progressed through the five kanji randomly. As soon as they saw the first screen, a timer started in the background. This timer measured how long it took each user to progress through Learning by Copying and Learning from Memory. Users were shown the same screen as in the pre-test, except this time they were also shown a diagram of the kanji with stroke orders and directions. Users had the ability to start over if they needed to. They had to press the submit button to notify the system that they had finished writing the kanji. This was to prevent the system from assuming that the user was finished if they wrote the kanji correctly (ie they might add an additional stroke, which would then make it incorrect). The app would not count the number of errors, as this is the learning stage. However, the app did notify the user of errors either after each stroke, or after each radical. They had to write the kanji correctly before they could move on to the next one. This was to try and prevent them from making lucky guesses and not actually learning. The screen looked like Figure 6.

3.3.3 Learning from Memory

In this stage, the user was asked to write the same five kanji as in Learning by Copying. The user was asked to write each kanji one by one. At this point, they were expected to draw the kanji from memory. However, if they write five or more incorrect strokes, a hint button would appear. The hint button displayed the image of the kanji they were working on until they begin drawing again. They had to write the kanji correctly with no reference picture before they could move on. The users still received error messages after each stroke or after each radical. After this stage was finished and they completed the 5th kanji, the timer stopped. The screen looked like Figure 5.

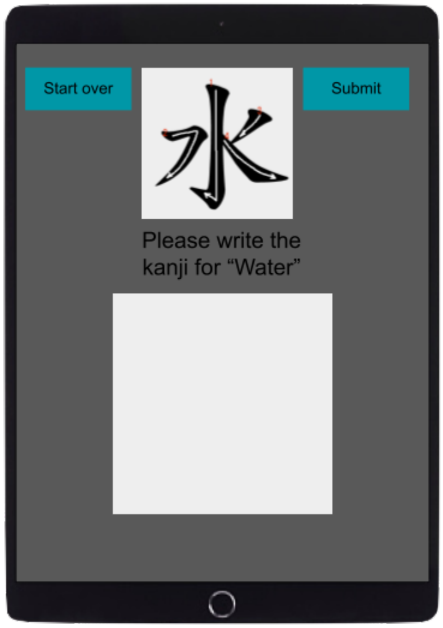

Figure 6: This is how the iPad looked for Learning by Copying.

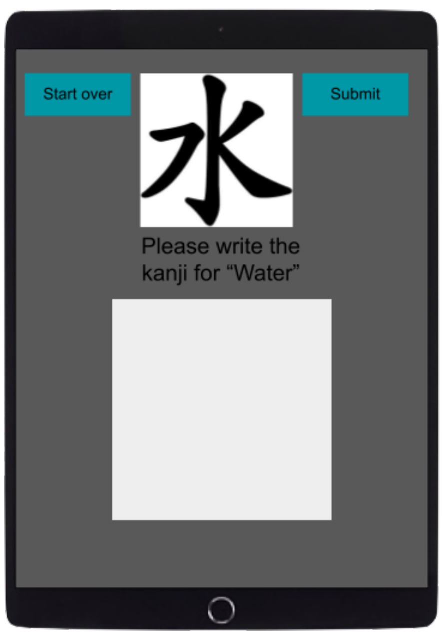

Figure 7: This is how the iPad looked for Learning from Memory.

3.3.4 Post Test

For the post-test, users were asked to write each of the five kanji, but this time with no error messages. They did have the option to start over, but not to skip. The app kept track of the number of strokes they draw correctly and incorrectly and the reasons for the errors (same as in the pretest). After they clicked submit, the app took a screenshot before moving on to the next kanji without giving them any feedback. The screen looked like Figure 5.

3.3.5 Retention Test

Users were asked to return after 24 hours to retake the post test. We kept track of all the same metrics as before and the test was in the same format as the post test.

3.3.6 Data Collection

Data was collected in string format and screenshots during the pre and post tests. This allowed us to see which strokes were correct and incorrect for each kanji, as well as how many total they got correct. An example of the data can be seen in Figure 8. In addition, we also collected data on how long it took users to progress through Learning by Copying and Learning from Memory.

3.4 Determining success

After we collected data from all participants, we compiled the data and compared the pre and post test results between the immediate and delayed feedback groups. We were looking to see if one group improved more from the pre-test to the post test and retention test than the other group. We were also looking at the timing data between the two groups. We will know if the project succeeded if one group did significantly better than the other group on either the post test, the retention test, or the time it took them to learn the kanji. We hypothesized that the group that receives delayed corrective feedback would perform better than the immediate feedback group because users would have more time to process their mistakes and remember why they are wrong.

4 Results

After recruiting participants from the Japanese language classes, we were left with 15 participants. From those, we screened out the participants who could already write the kanjis before doing the experiment. We used a post-experiment questionnaire asking about the kanjis and skill level in writing in Chinese. After this, we were left with eight participants whose data was suitable to use for analysis. You can see a summary of the results in 9. Both the immediate and the delayed feedback group had four participants. Compiling their respective results together, the immediate feedback group drew 61 strokes correctly out of 208 on the post test, and 43 correctly on the retention test. The delayed feedback group drew 25 out of 208 correctly on the post test and 18 correct on the retention test. This leaves us with a p-value of 0.30793, and since $0.30793 > 0.05$, our results are not statistically significant. Despite this, the immediate feedback group did tend to do better than the delayed feedback group. A possible explanation for this could be that the delayed feedback group participants became more frustrated with the app. In fact, this was a recurring issue throughout the study. Users reported that the app was frustrating and difficult to use because it was too sensitive to mistakes. In addition, the delayed feedback users were more prone to being frustrated because when they drew a stroke incorrectly, they had to keep redrawing the entire radical until they drew the entire radical correctly. There was also a difference between the average amount of time it took users to progress through the learning stages of the experiment. Immediate feedback participants finished in an average of 9.5 minutes while delayed feedback participants finished in an average of 25.8 minutes. However,this difference is likely in part caused by the delayed feedback participants needing to redraw the entire radical.

Before Kanji
Stroke 1: Please start this stroke higher
Stroke 2: Please start this stroke higher
Stroke 3: Please make this stroke longer
Stroke 4: Please make this stroke longer
Stroke 5: Correct!
Stroke 6: Correct!
Stroke 7: Correct!
Stroke 8: Correct!
Stroke 9: Correct!
Result: 5 / 9 strokes correct

Investigate Kanji
Stroke 1: This stroke should end more to the right than where it started
Stroke 2: Correct!
Stroke 3: Please make this stroke longer
Stroke 4: Please make this stroke longer
Stroke 5: Please make this stroke longer
Stroke 6: Correct!
Stroke 7: This stroke's end should intersect with the end of the previous stroke
Stroke 8: Please start this stroke lower
Stroke 9: Correct!
Stroke 10: Correct!
Stroke 11: Correct!
Stroke 12: Correct!
Stroke 13: Correct!
Stroke 14: Correct!
Stroke 15: Correct!
Result: 9 / 15 strokes correct

Summary: 27 / 53 strokes correct for day 2 post-test

Figure 8: A sample of how data was collected. Shows whether each stroke is correct or not, and why.

Table 1: Stroke CF (n=4)

Kanji	Strokes in Kanji	Pre-test	Post-test	Retention Test
氏 (Surname)	5	0	8	3
報 (Report)	12	0	17	12
調 (Investigate)	15	5	25	10
深 (Deep)	11	0	9	13
前 (Before)	9	0	2	5
Summary	**52 (208 total)**	**5**	**61**	**43**

Table 2: Radical CF (n=4)

Kanji	Strokes in Kanji	Pre-test	Post-test	Retention Test
氏 (Surname)	5	0	4	3
報 (Report)	12	0	8	9
調 (Investigate)	15	0	1	0
深 (Deep)	11	0	0	0
前 (Before)	9	0	12	6
Summary	**52 (208 total)**	**0**	**25**	**18**

Figure 9: Summary of the number of strokes correct from the four immediate feedback participants (Table 1) and the four delayed feedback participants (Table 2).

5 Future work

Going forward, some possible ways this research could be continued are as follows. One question that was not addressed in this research was the impact of level of Japanese on how well participants could learn kanji, or how easy/difficult the app is to use. Another way to continue this research would be to recreate the app with a focus on user easy of use. We could then investigate the question of whether delayed or immediate feedback is more beneficial to learning, or look into the user experience while learning.

6 Conclusion

Learning how to write kanji is a crucial aspect that Japanese learners must master, especially if they are hoping to read or write Japanese or become fluent. Despite this, there are few (free) apps that currently exist to teach students how to write kanji using an input-focused method where the student is expected to draw / write on the screen. This is a major missed opportunity because software today has the capability to act as the teacher and point out mistakes. Because of this deficiency, we believe that our system could make an impact on the Kanji learning world. This system incorporated not only the correct stroke order, but also the finer nuances that make kanji look truly authentic and beautiful. In addition, we are hoping to determine the best time to give users corrective feedback. Through a user study, we found that users who were given immediate corrective feedback performed slightly better than the users who were given delayed corrective feedback. While the results were not statistically significant, this research is not only important, but it has the ability to impact thousands of Japanese Language students.

References

[1] Margherita Berti. "Second Language Acquisition in Action: Principles from Practice, by A. Nava and L. Pedrazzini (2018)". In: *Instructed Second Language Acquisition* 3.1 (Nov. 2020), pp. 111–114. DOI: 10.1558/isla.37254. URL: https://journal.equinoxpub.com/ISLA/article/view/11322.

[2] G. Chen, Yu-Du Jheng, and L. Lin. "Computer-based Assessment for the Stroke Order of Chinese Characters Writing". In: *Second International Conference on Innovative Computing, Informatio and Control (ICICIC 2007)* (2007), pp. 160–160. DOI: 10.1145/971478.971487.

[3] Gabriel F. Groner. "Real-Time Recognition of Handprinted Text". In: *Proceedings of the November 7-10, 1966, Fall Joint Computer Conference*. AFIPS '66 (Fall). San Francisco, California: Association for Computing Machinery, 1966, pp. 591–601. ISBN: 9781450378932. DOI: 10.1145/1464291.1464355. URL: https://doi.org/10.1145/1464291.1464355.

[4] Kiia Korpi and Kiyoharu Aizawa. "Kanji Snap: An OCR-Based Smartphone Application for Learning Japanese Kanji Characters". In: *Proceedings of the 21st ACM International Conference on Multimedia*. MM '13. Barcelona, Spain: Association for Computing Machinery, 2013, pp. 403–404. ISBN: 9781450324045. DOI: 10.1145/2502081.2502241. URL: https://doi.org/10.1145/2502081.2502241.

[5] Norman Lin, Shoji Kajita, and Kenji Mase. "A Multi-Modal Mobile Device for Learning Japanese Kanji Characters through Mnemonic Stories". In: *Proceedings of the 9th International Conference on Multimodal Interfaces*. ICMI '07. Nagoya, Aichi, Japan: Association for Computing Machinery, 2007, pp. 335–338. ISBN: 9781595938176. DOI: 10.1145/1322192.1322250. URL: https://doi.org/10.1145/1322192.1322250.

[6] Tevfik Metin Sezgin, Thomas Stahovich, and Randall Davis. "Sketch Based Interfaces: Early Processing for Sketch Understanding". In: *Proceedings of the 2001 Workshop on Perceptive User Interfaces*. PUI '01. Orlando, Florida, USA: Association for Computing Machinery, 2001, pp. 1–8. ISBN: 9781450374736. DOI: 10.1145/971478.971487. URL: https://doi.org/10.1145/971478.971487.

[7] Paul Taele and Tracy Hammond. "Hashigo: A Next-Generation Sketch Interactive System for Japanese Kanji." In: *IAAI*. Ed. by Karen Zita Haigh and Nestor Rychtyckyj. AAAI, 2009. URL: http://dblp.uni-trier.de/db/conf/iaai/iaai2009.html#TaeleH09.

[8] Junjian Tang and Jun Guo. "A New Method for Stroke Order Recognition of Handwritten Chinese Characters". In: *Proceedings of the 2018 the 2nd International Conference on Video and Image Processing* (2018).