e-ISSN 2385-3042

Annali di Ca' Foscari. Serie orientale
Vol. 61 – Giugno 2025

An SVG-Based Approach for the Development of a Kanji Auto Assessment System

Alessandro Mantelli
Università Ca' Foscari Venezia, Italia

Abstract This paper proposes a vector-graphic approach to designing a kanji assessment system for enhancing instructional practices in Japanese language education. The study details the prototype of an experimental web application, built on a vector-graphics dataset, which captures kanji strokes in real time and compares learner-produced characters against reference models to generate efficacy scores. It also showcases the pedagogical versatility of animated stroke guidance for reinforcing correct writing procedures. Advantages (including potential Chinese support) and limitations are discussed considering practical benefits and appropriate use contexts.

Keywords Automatic handwritten recognition. Digital Humanities. E-learning. Japanese language. Stroke-order recognition.

Edizioni Ca'Foscari

Peer review

Submitted 2025-03-24
Accepted 2025-04-30
Published 2025-07-31

Open access

Citation Mantelli, A. (2025). "An SVG-Based Approach for the Development of a Kanji Auto Assessment System". *Annali di Ca' Foscari. Serie orientale*, 61, 485-508.

1 Introduction

Kanji[1] teaching is a fundamental aspect of Japanese education, commencing in elementary school, where students are systematically introduced to both reading and writing kanji. The Ministry of Education, Culture, Sports, Science and Technology (MEXT) underscores the importance of properly writing kanji as an indispensable skill. It also emphasises the importance of maintaining a consistent instructional approach that adheres to standard forms and precise stroke order, recognizing these elements as the foundation of effective kanji education (MEXT 2016).

Kanji teaching outside Japan may differ from traditional approaches in Japan. Writing instruction is often less emphasised than reading, as evidenced by Riekkinen's analysis of Japanese textbooks, which found that "the exercises in intermediate level books are often presented in an authentic context; however, the majority focus on kanji recognition rather than kanji reproduction" (Riekkinen, Linder 2015). Similar conclusions may be inferred by examining the results of a survey (Kubota 2017, 20) which states that L2 students studying in Japan were more frequently engaged in writing kanji activities and assessments, such as writing on the blackboard. The students' comments also suggest that more time is dedicated to kanji instruction in Japanese contexts than at universities abroad. Back in their country, students tend to learn kanji more independently and primarily focusing on character comprehension for reading, rather than on mastering the detailed stroke order required for writing (Kubota 2017, 6).

Nevertheless, despite the potentially structured nature of the kanji curriculum, students often struggle to maintain accuracy and consistency in their handwriting, with factors such as confidence, proficiency level, and home environment influencing their self-assessments. They may overestimate their ability in writing, rely on reading skills as a basis for judging their proficiency, or misjudge their understanding of radicals and compound kanji words (Aiko 2018, 9-17). These findings suggest that the current methods of feedback, largely reliant on occasional quizzes, self-monitoring, and teacher correction, may not always help students accurately recognise their weaknesses (Aiko 2018, 20). Conventional methods of instruction and assessment heavily depend on human evaluators, which can result in inconsistencies and limit scalability, because as the number of students increases, maintaining uniform standards and providing timely and individualised feedback becomes more

1 Kanji are logographic characters derived from Chinese, used in the Japanese writing system alongside Hiragana and Katakana.

challenging. Automated systems offer a potential solution to this issue as they can provide consistent, scalable, and immediate feedback on kanji writing. This likely explains why companies in Japan over the past decades,[2] have developed various tools for kanji input and recognition, laying the groundwork for efficient and reliable kanji assessment tools.

Specifically, the foundation for automatic assessment systems can be traced back to early efforts in handwriting recognition and natural language processing research in the 1950s and 1960s. For example, the Rand Tablets (Nakagawa, 2024, 4), was the first digital graphic low-cost input device developed by the RAND Corporation. Around the same time, the Project Essay Grade (PEG) system was developed as a device able to auto evaluating essays and create an automating scoring system (Zhang 2021).

A firsthand encounter with a kanji evaluation system took place in 2007 with *Tadashii Kanji Kakitori-kun* 正しい漢字書き取りくん (Kakitori Kun the Correct Kanji Writer) on the Nintendo DS back in 2007 (Nintendo 2007).[3] Despite the limited precision of the stylus, the system was already remarkably effective. In Japan, such technologies have predominantly been developed by companies rather than academic institutions. Recent commercial examples of kanji scoring systems, such as 'Saiten Navi' by Education Soft (Education soft 2023) and 'Digital Kanji Plus' by Neos (Neos 2022), indicate ongoing progress in this field. Collaborative efforts between universities and industry have also emerged: for instance, Masaki Nakagawa, emeritus professor at the Tokyo University of Agriculture and Technology and a pioneer in this field, has worked extensively with various corporations such as i-Labo, which provides kanji recognition SDKs[4] and software for Windows and Linux.[5]

Most of these products are stand-alone software rather than web-based applications, limiting their use to individual users, mainly elementary school children. The i-Labo SDK, even if online, while offering integration options for custom applications, is limited

2 Toshiba introduced the JW-10 word processor for the Japanese language in 1978, based on a 1967 proposal by Toshihiko Kurihara (Doi 2015).

3 Although Its release on the Nintendo 3DS might imply a primarily recreational tool, the software was in fact designed with clear pedagogical objective as it provided structured, immediate feedback on stroke order, character form, and writing accuracy. Indeed, one of the Nintendo DS's selling points has always been its non-game applications. In this context, several reports describe its use in Japanese schools, such as those by Sloan (2008) and Charlton (2009).

4 Acronym of Software Development Kit. SDK is a set of tools to build application for specific types of purposes or platforms.

5 Information gathered from the Nakagawa Laboratory section of the TUAT University site (Nakagawa 2024b).

to approximately 1,000 characters, and the parameters used for evaluation are not modifiable. Furthermore, these software solutions are not freely available; the SDKs come with initial and annual fees, which can be burdensome for universities that might have funding constraints.

Among researchers, notable is the work of Prof. Ido at Gifu Kyoritsu University, who developed a sophisticated system for the automatic scoring of handwritten kanji. As Ido (2015; 2019; 2020; 2022) describes, the system evaluates kanji based on multiple parameters, such as stroke correspondence, length, and intersection angles, using advanced metrics like the order-distance technique. However, the system's proprietary nature (it offered a limited password protected demo) means that its source code has not been released for public use (Ido 2024, 12).

Research conducted by non-Japanese scholars on the development of such systems is considerably less common compared to research originating in Japan. A notable example is the work of Taele, Koh, and Hammond from the University of Texas on a Writing-Based Intelligent Tutoring System (2020). Their research emphasises the importance of providing high-quality reference models to which students' drawings can be compared for effective evaluation.

Given the proprietary nature of existing systems described above, a central motivation for this study was to identify open-source, freely accessible tools for kanji assessment. Such solutions are particularly valuable because they can be extended with additional functionalities. In the context of kanji assessment for example, implementing a database to collect students' error would be highly beneficial, as it would offer insights into prevalent error patterns and raise important pedagogical considerations. Unfortunately, open-source options are scarce. An alternative therefore is to develop systems from scratch that integrate, wherever is possible, open-source technologies along with database capabilities. This approach would also enable seamless integration with existing departmental software. For instance, the department of Asian and North African Studies of Ca Foscari University of Venice has already developed a web application designed to support kanji study and provide downloadable printable exercise sheets, yet it lacks kanji evaluation features (Mantelli 2021). In this context, standalone software, with their inherent limitations, are not suitable for meeting these objectives.

However, developing such a system from scratch poses significant challenges, particularly in resource-constrained environments. This study therefore investigates the viability of assembling a functional kanji assessment platform by strategically integrating publicly available digital tools and datasets. In this context, SVG shapes of kanji are of particular interest, as they offer a promising foundation for creating a scalable and adaptable web-based assessment system.

2 Identifying a Model

As mentioned in the introduction, relying on a robust model to compare students' kanji writing is particularly important for developing an effective assessment system. Previous studies have approached this challenge in various ways. For example, Taele, Koh and Ammond (2020) collected writing data from an expert, specifically a Japanese foreign language instructor, who provided character writing data for 448 characters used in their interface. This data served as templates for assessing students' performance in writing. The data were gathered through a special application and stored in JSON[6] format.

Ido (2019) defined a model where path data represented by Bézier curves[7] and scoring information are embedded within XML[8] files, structured to accommodate character-specific and assessment-specific data.

This study's methodology diverges from conventional approaches by adopting KanjiVG, a widely recognised, community-driven, open-source repository of vector-based kanji glyphs, as its foundational comparison framework. KanjiVG streamlines the development process by providing over 12,000 characters equipped with stroke-order and stroke-path metadata, thereby eliminating the need for manual data collection. Its interoperability is ensured by a Creative Commons license that facilitates redistribution and integration, while its proven adoption in platforms like Jisho, Kanji Alive, and Duolingo underscores its alignment with existing digital ecosystems. These attributes guarantee typographic precision and establish KanjiVG as a scalable, community-driven solution to custom glyph generation methods.

The advantages of vector graphics in kanji related projects are fundamental because unlike raster graphics formats such as JPG[9] or PNG,[10] which represents images as arrays of pixels with associated RGB[11] color values but lack relational logic, vector graphics describe images using geometric shapes (Eisenberg 2020, 22). In the case of

6 Acronym of JavaScript Object Notation, a data Interchange format to share data between applications.

7 Parametric curves that are defined by a set of control points.

8 Acronym of Extensible Markup Language (XML), a markup language and file format for storing, transmitting, and reconstructing data.

9 Acronym of Joint Photographic Experts Group, a digital file format commonly used for photos and other complex images.

10 Acronym of Portable Network Graphics, a raster-graphics file format that supports lossless data compression.

11 Acronym of Red Green Blue, an additive color model in which the red, green, and blue primary colors of light are added together in various ways to reproduce a broad array of colors.

e-ISSN 2385-3042

KanjiVG, kanji shapes are distributed in the Scalable Vector Graphics (SVG) format. As SVG images are defined by spatial coordinates of vertices and paths, they enable an exact mathematical representation of the kanji characters, allowing a correct foundation for creating comparison models. In contrast, raster images are subject to compression and prioritise the overall image rather than the details. Moreover, since SVGs consist of instructions for creating shapes rather than fixed pixel information, they can be scaled infinitely without any loss of quality. This scalability ensures that kanji characters are rendered accurately at various sizes, accommodating different screen resolutions and devices, which is important for a web-based assessment system accessible on multiple platforms.

Another significant advantage of SVG images is their typically smaller file sizes compared to raster images. By storing mathematical instructions instead of pixel data, SVG files are more compact, resulting in faster loading times, an important consideration for enhancing user experience in web applications. Additionally, SVG elements can be easily manipulated using web technologies such as CSS[12] and JavaScript.[13]

SVG shapes are described in XML format and enriched with attributes that define the basic form and the coordinates. An SVG image comprises elements like <circle>, <line>, and <path>, each representing different shapes **[fig. 1]**.

```
<svg width="200" height="200">
  <!-- Group of shapes -->
  <g fill="none" stroke="black" stroke-width="2">
    <!-- Circle -->
    <circle cx="100" cy="100" r="50" />
    <!-- Line -->
    <line x1="50" y1="50" x2="150" y2="150" />
    <!-- Path -->
    <path d="M 60 160 L 100 120 L 140 160 Z" />
  </g>
</svg>
```

Figure 1 Sample of a SVG shape

12 Acronym of Cascading Style Sheets, is a style sheet language used for specifying the presentation and styling of a document written in a markup language such as HTML.

13 Programming language natively present in modern web browsers, enabling the implementation of complex features on web pages and allowing elements to be dynamically modified on the fly without requiring a full page reload.

e-ISSN 2385-3042

In this example, the <path> element uses commands such as M (MoveTo), L (LineTo), and Z (ClosePath) to draw shapes. These commands correspond to movements that can represent kanji strokes. As can be seen in Figure 1, the SVG representation is purely textual, defined by path commands that trace the shape of each kanji stroke. This textual format allows for precise manipulation, scalability, essential attributes for automated shape analysis and stroke pattern recognition.

3 Leveraging SVG for Kanji Representation

The KanjiVG dataset not only includes the geometric shapes of kanji characters but also embeds metadata such as stroke order, radicals, and stroke vertices. Each kanji character in the KanjiVG repository is organised using a hierarchical structure defined by XML tags, primarily <g> (group) and <path> elements. Each group is enriched with attributes that specify details such as the kanji element (kvg:element), position (kvg:position), and radical (kvg:radical). This structured format encapsulates various aspects of kanji characters, facilitating efficient analysis and manipulation for diverse applications **[fig. 2]**.

```
<svg xmlns="http://www.w3.org/2000/svg" width="109" height="109" viewBox="0 0 109 109">
  <g id="kvg:StrokePaths_05bb6" style="fill:none;stroke:#000000;stroke-width:3;stroke-linecap:round;stroke-linejoin:round;">
    <g id="kvg:05bb6" kvg:element="家">
      <g id="kvg:05bb6-g1" kvg:element="□" kvg:position="top" kvg:radical="general">
        <path id="kvg:05bb6-s1" kvg:type="□a" d="M52.44,11c0.97,0.97,1.69,2.25,1.69,3.68c0,3.42-0.08,4.99-0.08,8.1"/>
        <g id="kvg:05bb6-g2" kvg:element="冖">
          <path id="kvg:05bb6-s2" kvg:type="□" d="M27.09,23.28c0,3.73-2.77,13.98-4.03,16.47"/>
          <path id="kvg:05bb6-s3" kvg:type="□b" d="M27.99,27.18C41.88,25,72.75,21.5,79.81,21.2c12.19-0.52,1.14,7.87-0.34,8.63"/>
        </g>
      </g>
      <g id="kvg:05bb6-g3" kvg:element="豕" kvg:position="bottom">
        <path id="kvg:05bb6-s4" kvg:type="□" d="M32.76,40.23c1.38,0.4,3.92,0.57,
        5.29,0.4c8.08-1,19.7-3.38,25.59-3.84c2.29-0.18,3.41-0.04,4.83-0.11"/>
        <path id="kvg:05bb6-s5" kvg:type="□" d="M51.26,41.5c0.18,1.1-0.06,2.08-0.72,2.94C47.84,48.63,38.92,55.58,26.5,60.5"/>
        <path id="kvg:05bb6-s6" kvg:type=")" d="M48.07,48.96c8.31,7.92,14.18,25.04,6.92,43.51c-2.42,6.16-8.12,0.9-9.3,0.05"/>
        <path id="kvg:05bb6-s7" kvg:type="□" d="M49,56c0.17,1.16-0.14,2.17-0.92,3.04c-2.78,3.84-10.78,9.86-22.33,14.21"/>
        <path id="kvg:05bb6-s8" kvg:type="□" d="M52.5,65.5c0.21,1.3-0.06,2.45-0.84,3.44c-3.27,5.04-13.97,13.61-29.91,19.56"/>
        <path id="kvg:05bb6-s9" kvg:type="□" d="M76.8,45.3c0.14,1.12-0.2,2.09-1.02,2.91c-2.55,3.35-9.29,8.72-18.05,12.19"/>
        <path id="kvg:05bb6-s10" kvg:type="＼" d="M58.79,62.82c3.72,2.92,18.27,15.42,25.25,20.69c2.49,1.88,4.18,3.12,7.08,3.87"/>
      </g>
    </g>
  </g>
</svg>
```

Figure 2 Sample of the kanji 家 (house) from the KanjiVG dataset

As can be observed in figure 2, each individual stroke of a kanji is defined by a <path> element with a d attribute which contains the path data. The path data uses commands like M (move to), L (line to), C (cubic Bézier curve), and Z (close path) to define the exact shape and trajectory of the stroke. The <g> tags group these strokes and structural components, allowing for a clear representation of the kanji's composition. By utilizing these attributes and path data, the KanjiVG dataset provides a comprehensive framework for understanding and manipulating kanji shapes.

As discussed in the previous chapter, a key distinction between SVG and raster images is the ability to access and modify each component of an SVG image, as well as its coordinate data thanks to its XML-based structure. In the context of web design, this capability is primarily enabled by JavaScript, a programming language embedded by default in web browsers. JavaScript facilitates real-time manipulation of the HTML/XML Document Object Model (DOM), allowing for dynamic modifications of the SVG elements. For instance, it is possible to access each part of the SVG and retrieve important kanji information such as the total number of strokes (by counting the <path> nodes) or the direction of each stroke by analyzing the content of the d attribute and calculating the starting and ending points of the node.

Given the numerous advantages of SVG, I have developed a kanji assessment prototype in the form of a web application utilizing KanjiVG and JavaScript. The development of this prototype illustrates the practical applicability of SVG technology and open-source datasets in educational technology aiming to create a functional tool for assisting learners in improving their kanji writing skills.

4 Prototype Design and Development Using KanjiVG and JavaScript

This section explores the practical aspects of designing and developing a kanji assessment prototype. By leveraging the KanjiVG dataset and JavaScript technologies within a web application framework, the prototype seeks to deliver an interactive and accessible tool for evaluating students' kanji writing skills. The discussion encompasses the architectural design, implementation of core functionalities such as stroke recognition and validation, user interface considerations, and the integration of KanjiVG data with the aim to offer perspectives to encourage future research in this field.

4.1 Architectural Design and Technology Stack

Web applications have several advantages over desktop applications (Bhagwan 2021, 12), among others, they can be accessed from any device with a modern browser, eliminating the need for users to install specialised software and aligning with the goal of providing an educational tool that is available to students across various environments.

To facilitate real-time interaction and dynamic content manipulation, the prototype employs JavaScript through the Nuxt.js (`https://nuxt.com/`) framework, which is built on the Vue.js (`https://vuejs.org/`) technology.

While an in-depth discussion of the architectural advantages of Vue.js and Nuxt.js over standard JavaScript is beyond the scope of this work, it is essential to note that the framework allow the creation of modules each with its particular function, such as drawing the kanji, examining the drawn shape, and compare drawn shape over the given model. Furthermore, it facilitates the future implementation of features like database connections. The modular architecture of Nuxt.js thus ensures that the application is easily scalable to accommodate increased complexity.

4.2 User Interface and Interaction Design

The prototype's user interface provides a digital canvas where students can draw kanji characters using a mouse, stylus, or touch input on devices such as tablets and touchscreen computers. The application adapts to different screen sizes and input methods, ensuring a consistent experience across devices using an approach to design called responsive.[14] The interface captures the students' handwriting data in real time, enabling real-time analysis and feedback. At the present moment, the interface presents only the necessary elements for the drawing: a) a square divided in four quadrants by 2 dashed lines, b) buttons for restarting and for undoing the last step, and 3) a button to finish and receive the score.

It is possible to select a different kanji from a list using the menu on the right. Since the software is still a prototype, all operational reports can be debugged by clicking on the debug selector. It is also possible to display the kanji model on the background for practice purpose. The top-left side of the quadrant displays the current input coordinates, also used for debugging purpose **[fig. 3]**.

14 The term, coined by Ethan Marcotte (2011), refers to design techniques that enable interface elements to automatically adapt to the size and format of the window or screen.

e-ISSN 2385-3042

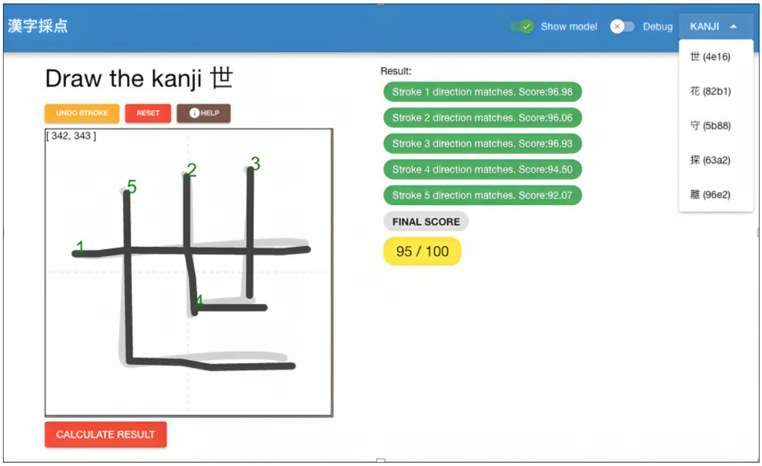

Figure 3 Kanji Evaluation prototype with result

Each time the student draws a stroke the stroke number is displayed above it. By pressing the [Calculate Result] button, the final score is calculated. The system also checks for any incorrect stroke directions or discrepancies in the number of strokes as seen in figure 4.

The inclusion of visual indicators and interactive elements aims to engage students actively in the learning process. By providing real-time feedback and an accessible interface, the prototype should encourage students to practice kanji writing more frequently and with greater attention to detail as the ability to receive immediate evaluation fosters a more responsive learning environment, where students can iteratively improve their skills through practice and correction. Studies on motivation and design demonstrate a connection between inner motivation and user engagement (Deci, Ryan 2000, 73; Deterding 2012, 17). In particular, features such as targeted feedback loops can stimulate intrinsic motivation, as learners perceive their efforts translating directly into noticeable improvements (Malone, Lepper 1987).

The system's function to display the kanji model in the background serves as a reference for students, aiding in the reinforcement of correct stroke order and structure. This optional feature can be particularly beneficial for beginners' learners who are still familiarizing themselves with kanji characters **[fig. 4]**.

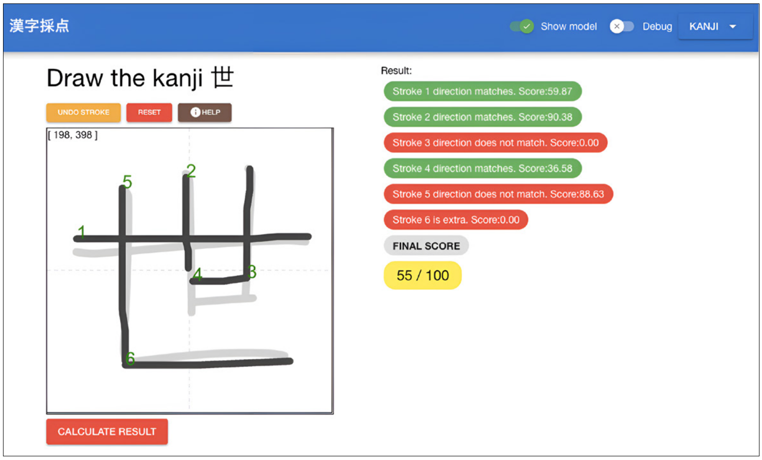

Figure 4 Handling Missing or Extra Strokes: The system detects discrepancies in the number of strokes. Missing strokes are flagged, and extra strokes are identified, contributing to the overall assessment

4.3 Integration of KanjiVG Data

The KanjiVG dataset serves as the foundation for the prototype's kanji models. Each kanji character is represented as an SVG file containing detailed information about its strokes, including order, direction, and path data. Integrating this data into the application involves parsing the SVG files and extracting relevant information for comparison with the student's input. By analyzing the SVG structure, the system determines the total number of strokes, their sequence, and their geometric properties. Since the SVG stroke data is defined in local coordinates, it is necessary to convert these into absolute pixel coordinates that correspond to the user's drawing canvas. This ensures accurate alignment between the model and the student's input.

4.3.1 Implementation of Core Functionalities

The core functionalities of the prototype revolve around capturing the student's handwriting input and comparing it to the reference kanji model to evaluate correctness in terms of stroke order, direction, and structure. This process involves several computational steps:

1. Definition of a common coordinate system
   To compare the student's strokes with the model accurately, both datasets (the student's and the model's) are transformed into a common coordinate system. This involves calculating

the bounding box (Abramov 2021)[15] for both the student's input and the model kanji. The top-left corner of each bounding box is shifted to the origin (0,0) of the coordinate system. This translation centers both the student's strokes and the model, eliminating positional discrepancies and ensuring that the comparison focuses on the shape and structure rather than position.

2. Data Capture and Preprocessing
   As the student draws each stroke, the application records a sequence of points representing the stroke's trajectory. These points are captured in real time, storing the spatial coordinates (x and y) of each point. This continuous recording ensures that the geometric characteristics of the strokes are preserved for successive analysis.
3. Normalisation Through Sampling
   The raw input data may vary significantly between users due to differences in drawing speed, scale, and starting position. To facilitate a fair comparison, the input data is normalised by sampling a consistent number of points for each stroke. This sampling reduces the dataset to a manageable size while preserving the essential characteristics of each stroke, ensuring uniform representation across all inputs.
4. Scaling with Affine Transformation
   An affine transformation is applied to scale the student's strokes to match the size of the model kanji. This process adjusts for differences in scale, ensuring that size variations do not affect the assessment. The scaling factors are calculated based on the dimensions of the bounding boxes, maintaining the proportionality of the strokes and focusing the evaluation on the shape and structure rather than the size.
5. Stroke Direction Analysis
   To determine the direction of each stroke, the system analyses the initial segment of the stroke. Typically, the first 20 points are sampled to capture the initial movement, providing a reliable indication of the stroke's starting direction. The angle of the initial segment is calculated using trigonometric functions, resulting in a direction vector for the stroke. This angle is then compared to the corresponding stroke in the model. The system allows for a predefined tolerance (e.g. ±20 degrees) when comparing stroke directions. If the student's stroke direction falls within this acceptable range, it is

15 A geometric rectangle that serving as a point of reference for object detection, encloses or surrounds an object or a group of objects in a digital image.

considered correct, accommodating minor deviations that are natural in handwriting.

6. Euclidean Distance Calculation
   After normalisation and sampling, the student's strokes are aligned with the corresponding strokes in the model based on stroke order. For each pair of corresponding strokes, the Euclidean distance between each point in the student's stroke and the corresponding point in the model stroke is calculated. This calculation provides a quantitative measure of the similarity between the strokes. Smaller distances indicate a closer match, reflecting higher accuracy in the student's writing.
7. Score Calculation
   Once the Euclidean distances have been computed for each pair of corresponding strokes, the system converts these distances into a score on a scale from 0 to 100, reflecting the student's adherence to the model stroke. If extra strokes are detected or if any stroke is drawn in the wrong direction, that stroke is assigned a score of 0. The final score is determined by taking the average of all individual stroke scores.

4.3.2 Advantages of Standardised Datasets for Kanji Assessment and Extensibility

Standardised datasets, such as KanjiVG, provide significant advantages in developing kanji learning tools, such as the integration with compatible existing libraries. It is the case of Dmak[16] a JavaScript library developed by Matthieu Bilbille that enables kanji stroke order animation. This library uses the same SVG data provided by KanjiVG, allowing students to view the correct sequence and formation of kanji strokes in real time. It also offers configurable options for animation speed and stroke colours, making it adaptable to various learning needs.

This functionality is particularly beneficial from a language-learning perspective, as it provides immediate visual feedback and reinforces proper writing techniques. When a student encounters difficulty with a particular character, accessing the Stroke Animation can clarify uncertainties and enhance retention through visual demonstration **[fig. 5]**. In the developed prototype, this functionality can be triggered by pressing the [help] button, providing learners with the animation of the kanji in question.

16 Acronym of Draw Me A Kanji (the website is available at `https://mbilbille.github.io/dmak/`).

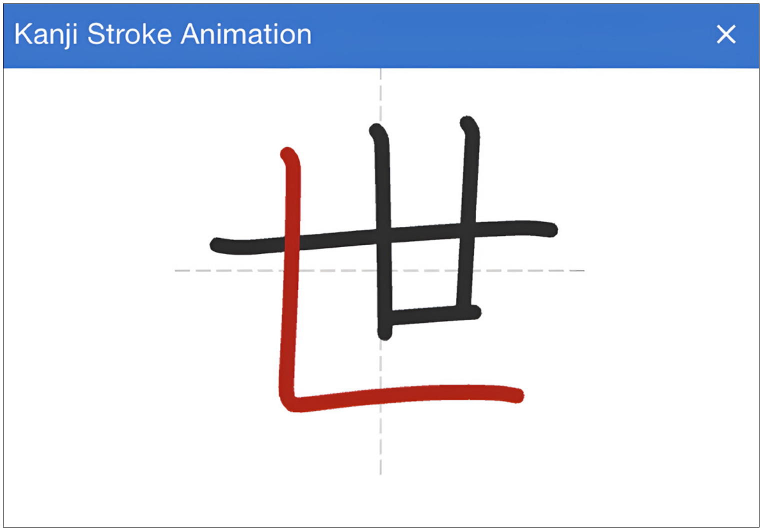

Figure 5 Kanji Stroke Animation functionality developed with the DMAK library

The inclusion of Dmak aligns with pedagogical strategies that emphasise multimodal learning (Bouchey, Castek, Thygeson 2021, 36). By catering to visual and kinesthetic learners, the prototype fosters a more engaging and effective educational experience. As previously mentioned, Dmak's complete support of KanjiVG facilitates seamless integration, reducing development complexity and allowing for the efficient incorporation of advanced features without extensive code modifications.

The KanjiVG data format is also compatible with character datasets for other languages, such as Chinese. Consequently, developing a character assessment tool that leverages this standardised framework yields considerable advantages by expanding its applicability beyond Japanese. For instance, an existing project called HanziVG (Groß 2018) provides SVG representations of Chinese characters (Hanzi), offering detailed stroke order and path data compatible with the KanjiVG dataset structure.

Although stroke order and character structure may differ considerably between Japanese and Chinese scripts, using SVG with the same data schema such as the organisation of the hierarchical data structure in <g> (groups), enables the prototype to adapt these distinctions seamlessly without the need to write new code to manage the different datasets.

Furthermore, kvg:-prefixed attributes (e.g. kvg:element, kvg:radical, kvg:original, kvg:position) **[fig. 2]** offer important character information that can be provided to the learners if needed.

In particular, details about the kanji radical (kvg:radical) and its semantic origin (kvg:original) can be highly beneficial.

In other words, it is possible to lay the basis for the creation of a multi-language writing assessment system that accommodate the unique conventions and nuances inherent each set. In this way, a prototype initially designed for kanji writing assessment can evolve into a more general solution, meeting the diverse needs of learners studying multiple East Asian languages that utilise logographic scripts.

Incorporating Dmak and HanziVG into the prototype exemplifies the advantages of utilizing established resources to enhance functionality and user experience. The integration of Stroke Animations through Dmak enriches the learning process by providing dynamic, interactive content that reinforces proper kanji writing techniques. Extending support to Chinese characters via HanziVG showcases the prototype's flexibility and potential for adaptation to various writing systems. These enhancements not only improve the efficacy of the assessment system but also align with broader educational objectives of supporting diverse learner needs.

5 Technological Limits of the Actual Prototype

While the prototype exhibits considerable potential as an interactive and accurate kanji assessment system, it currently faces several technological limitations that must be addressed to enhance its overall effectiveness. Addressing these constraints is important not only to establish a robust framework for future iterations but also to ensure the system's accuracy in evaluating kanji writing, particularly through the integration of multi-modal assessment metrics (e.g. stroke-order fidelity, stroke-direction precision, and shape similarity algorithms). The following sections outline these limitations and propose actionable solutions where feasible.

5.1 Overreliance of Shape Distance Metrics as Primary Evaluation Method

The current evaluation approach relies predominantly on computing the Euclidean distance between corresponding points of a student's input and a reference kanji model, following normalisation and scaling. While this method effectively captures general shape accuracy, it omits several critical facets of kanji writing. In particular, the system does not assess stroke connectivity (gaps or overlaps between adjacent strokes), the terminal stroke angle of each stroke, and temporal parameters such as execution speed or pause intervals.

In kanji writing, certain strokes are designed to connect seamlessly, and failing to evaluate this aspect can lead to incomplete assessment. Although a lack of connection currently results in a reduced score due to imperfect adherence to the model, this disconnection should be recognised as a more severe error. By assigning a significantly lower score to disconnected strokes, the system could better reflect the structural integrity of the character and identify errors that compromise its fundamental composition. Another important factor involves the endpoints of strokes, which may terminate in a sharp flick (*hane* はね) or taper outward (*harai* はらい). In the current implementation, any deviation from the model's endpoint is penalised only as a minor discrepancy. Ido (2019) recommends the use of measurement vectors to evaluate the relationships between strokes, specifically by gauging how two points, such as a stroke's endpoint and its brush-out trajectory, align with predefined criteria. Incorporating a similar method into the prototype could yield a more nuanced and precise evaluation of students' written kanji.

This type of aspects enters in a more general assessment criteria which guidelines are detailed by the Agency for Cultural Affairs of Japan. The perfect adherence to these criteria involves a notable challenge. A solution proposed by Ido (2015, 133) is the implementation of an *Aggressive DP Matching* (*ADPM*) technique for substroke correspondence and an order coordinate system for stroke alignment. The ADPM technique allows precise measurement of deviations in stroke placement and geometry though if its implementation may be complex and time-consuming.

Another aspect not yet addressed in the present prototype is stroke speed and temporal data. The speed at which a student writes a stroke can provide valuable insights into their confidence and familiarity with the character. While input devices may affect the writing speed, a single student employing the same device could rewrite a kanji more rapidly over time, thus highlighting improvement in writing fluency. Storing these timing data in a database and making them available to the educator would provide an overall indication of each student's progress. A practical implementation might involve starting a timer when the student begins writing and recording the input-down and input-up events of each stroke, then saving these metrics alongside stroke path data in the database. The interface could subsequently present these metrics, along with the overall score to highlight progress over multiple sections.

5.1.1 Complex Kanji Assessment

Assessing complex kanji characters poses another challenge. As the number of strokes and the intricacy of their configurations increase,

e-ISSN 2385-3042

the current computational methods may not scale effectively. Tolerance thresholds and algorithms might require refinement to handle advanced characters accurately. For instance, difficulties arise in accurately determining stroke direction when dealing with very short shapes, resulting in both false errors and false positives. To solve this problem, more complex character should be tested and the stroke direction analysis mechanism should be improved by for example, dynamically changing the threshold angle according with the length of the stroke.

5.1.2 Improvement of the Interface and of the Functionalities

The user interface could also be enhanced to promote a more engaging learning experience. As the prototype is still in the early development phases, features such detailed feedback, interactive tutorials, and customizable settings have not been implemented, potentially limiting usability. An intuitive, user-centred interface would support learner engagement and strengthen educational outcomes. Additionally, establishing a database of common errors and providing targeted feedback would enable learners to identify and correct habitual mistakes, while providing educators with valuable insights into how best to support them, thereby improving overall learning effectiveness.

As the system evolves, optimizing performance will become increasingly important. Efficient algorithms must be developed to handle complex kanji without substantial performance degradation. Managing larger data loads, supporting multiple languages, and accommodating broader user bases necessitate careful scalability considerations. Such technical optimisations will be fundamental to maintaining the system's relevance and effectiveness over time.

6 Scope and Practical Limitations of the Present Approach for Kanji Auto-Assessment

While the developed prototype shows significant potential as an educational tool for kanji learning, it is important to critically evaluate its real-world applicability, particularly within classroom settings and formal examinations. This discussion explores the limitations inherent in implementing the auto-assessment system in educational environments, considering factors such as hardware requirements, user experience with input devices, scalability for large classes, and alternative assessment methodologies.

6.1 Practical Limitations in Classroom Settings

A primary challenge in deploying the prototype within a classroom context is the need to provide each student with compatible hardware. The system relies on digital input devices capable of accurately capturing handwriting data, which raises concerns regarding accessibility and cost. Equipping a classroom of, for instance, fifty students with individual computers or tablets represents a substantial logistical and financial undertaking. While the system offers innovative potential for kanji assessment, its classroom implementation requires strategic resource allocation, particularly in institutions with limited budgets. Successful integration depends on scalable technical infrastructure and pedagogical alignment to ensure the technology enhances, rather than interrupts, instructional workflows. Potential challenges such as intermittent connectivity or device compatibility issues can be mitigated through pre-emptive testing and contingency planning. Furthermore, dedicating initial training sessions to familiarise students with the interface would help maintain focus on core learning outcomes while minimizing operational friction.

6.2 Challenges with Input Devices

The effectiveness of the auto-assessment system is intrinsically linked to the input devices employed by students. During testing phases, various devices were evaluated, including standard computer mice, entry-level Wacom drawing tablets, and iPads equipped with styluses. Each device presents unique trade-offs that impact usability and accuracy.

The use of a computer mouse, while commonplace and familiar to most students, is suboptimal for handwriting tasks. Its inherent lack of precision and the unnatural motion required can hinder the accurate rendering of kanji strokes, thereby affecting the reliability of the assessment. Students may find it challenging to produce the intricate shapes and lines characteristic of kanji, which can cause frustration and disengagement.

Wacom-like drawing tablets offer greater precision and are specifically designed for digital handwriting. In the trial, an entry-level Wacom tablet was selected for classroom use due to its affordability. However, these tablets lack an integrated display, creating a spatial disjunction between the tablet's surface and the computer screen. Users must adapt to this mismatch, which introduces a hand-eye coordination challenge during operation. The writing experience thus becomes less intuitive, potentially undermining assessment accuracy. Calibration efforts improved performance in standalone

e-ISSN 2385-3042

desktop applications but not in the browser-based web application. Achieving minimal device proficiency ultimately required several hours. As Barracosa (2021) notes, mastering such tablets demands training in spatial coordination skills, with adaptation times varying significantly across learners and device quality; in some cases, proficiency may take several days.

Apple iPads-like devices with styluses provide a more natural and intuitive interface, closely mimicking the traditional pen-and-paper experience. The direct interaction with the screen allows for more accurate stroke production and a smoother user experience. Nonetheless, the high cost associated with these devices presents a significant barrier to widespread adoption. Supplying each student with an iPad and stylus is often impractical, particularly in public education systems operating under financial constraints. In addition, studies indicate that any chosen tablet requires some degree of a learning curve (Milanovic 2006; Oviatt, Arthur, Cohen 2006), although trials suggest that this adjustment period is longer for non-display tablets such as non-display-based Wacom models. Oviatt, Arthur and Cohen (2006) also highlight persistent student reluctance to abandon paper, alongside higher error rates and longer task completion times on digital devices. Nonetheless, outcomes can vary considerably depending on both device type and individual student performance levels.

6.3 Scalability Concerns for Large Classes

Implementing the auto-assessment system in large classes amplifies the previously mentioned challenges. Managing and supporting the technology infrastructure required for numerous students places additional strain on institutional resources. The logistical complexities of distributing, maintaining, and troubleshooting a multitude of devices can detract from the primary educational mission.

Moreover, ensuring consistent performance and user experience across all devices is a non-trivial endeavour. Variability in hardware capabilities and student proficiency with technology can introduce discrepancies in assessment outcomes. These inconsistencies risk compromising evaluation fairness and accuracy, thereby calling into question the system's reliability and validity.

6.4 Suitability for Formal Examinations

Given these considerations, the current approach is best suited for self-learning contexts, while its viability in formal examination contexts remains constrained. Compared to traditional assessments, in which students write kanji characters by hand on paper, this approach can accurately capture the number of strokes and their direction, parameters that are difficult to evaluate in handwritten submissions. From this perspective, it is beneficial for practice purposes.

However, the logistical challenges of equipping examination environments with appropriate technology are substantial. The risk of technical failures during high-stakes assessments introduces substantial reliability concerns. Consequently, relying on the prototype for formal evaluations may not be practical or advisable.

6.5 Alternative Approaches: Image Processing Techniques

If handwritten exams are preferred to avoid the logistical and financial challenges associated with providing digital input devices, alternative methodologies that align more closely with traditional examination practices warrant consideration. One such approach involves leveraging image processing techniques to analyse handwritten kanji captured through digital imaging, thereby allowing educators to evaluate scanned or photographed submissions.

While this method preserves the authenticity of the conventional pen-and-paper experience and may eliminate the need for specialised hardware, its implementation is far from trivial. Tools like OpenCV (`https://opencv.org/`) and other computer vision libraries can facilitate image analysis, yet accurately interpreting handwritten kanji remains a complex task. Variations in handwriting styles, ink density, lighting conditions, and image resolution can all influence the reliability of image-based assessments. Ensuring that the system can consistently identify kanji, distinguish similar shapes, and evaluate character quality requires careful calibration and robust algorithmic design.

Moreover, integrating machine learning models capable of handling these nuances involves substantial development effort. Collecting and annotating training data, refining models to handle diverse handwriting samples, and maintaining acceptable levels of accuracy and speed are nontrivial challenges. Although advancements in computer vision and machine learning continue to expand the possibilities, educators and developers must recognise the significant technical hurdles and resource investments these solutions entail.

7 Conclusion

This study introduces a novel open-source approach to kanji auto-assessment, leveraging KanjiVG's vector data to prioritise accessibility and adaptability – a departure from proprietary or hardware-dependent systems. While challenges in formal examination contexts (e.g. device variability, training time) warrant further refinement, the prototype's innovative reuse of standardised glyphs offers immediate value as a low-cost, device-flexible tool for self-directed learning, particularly in resource-constrained settings. However, factors such as hardware requirements, usability issues related to digital input devices, and the logistical complexities of equipping numerous students can limit the feasibility of widespread classroom integration. Nonetheless, these constraints do not diminish the prototype's strengths in other learning contexts.

In particular, the system shows promise as a self-learning tool, where students can practice kanji on a variety of devices, even those with less precise input methods such as a standard mouse. For instance, although the mouse may not perfectly replicate pen-and-paper writing, it can still help learners focus on the correct stroke order and overall character shape. This focus can reinforce fundamentals and encourage regular, low-pressure practice outside the formal classroom setting.

Additionally, the modular architecture, built on open SVG data and extensible via libraries like Dmak, provides a scaffold for community-driven enhancements – from multilingual support (e.g. Chinese Hanzi) to advanced feedback mechanisms. Educators can already deploy it for targeted exercises, while its open-source foundation ensures future integrations (e.g. machine learning) require minimal structural overhaul.

Critically, the perceived 'limitations' reflect deliberate trade-offs to serve underrepresented educational needs: its simplicity makes it viable for casual learners, remote education, and institutions prioritizing affordability over commercial solutions. Rather than competing with high-stakes assessment tools, it fills a gap in everyday skill reinforcement.

Ultimately, balancing technological innovation with the realities of educational environments is likely the most critical factor. By supporting diverse input methods and prioritizing ease of use, the system can continue to evolve as a practical and sustainable resource that fosters robust kanji-writing skills and meets a variety of learner needs, even if it, at this stage, remains better suited to self-directed study than to formal testing.

Bibliography

Abramov, M. (2021). "Image Processing Techniques: What Are Bounding Boxes?". *Keymakr*. https://keymakr.com/blog/what-are-bounding-boxes/.

Aiko, M. (2018). "The Relationships Between the Accuracy of Self-Evaluation, Kanji Proficiency and the Learning Environment for Adolescent Japanese Heritage Language Learners". *Journal of Language and Education*, 4(2), 6-23. https://doi.org/10.17323/2411-7390-2018-4-2-6-23

Barracosa, K. (2021). "How to Get Used to a Wacom Tablet After Always Using a Mouse". *Officeworks*. https://www.officeworks.com.au/noteworthy/post/create-how-to-get-used-to-a-wacom-tablet.

Bhagwan, S.M. (2021). "Investigation of Progressive Web Applications and Understanding Its Working Mechanism, Advantages and Disadvantages". *International Journal on Future Revolution in Computer Science & Communication Engineering*, 7(4), 12.

Bouchey, B.; Castek, J.; Thygeson, J (2021). "Multimodal Learning". Ryoo, J.; Winkelmann, K. (eds), *Innovative Learning Environments in STEM Higher Education*, 35-54. Cham: Springer International Publishing (SpringerBriefs in Statistics). http://link.springer.com/10.1007/978-3-030-58948-6_3

Charlton, J. (2009). "Nintendo DS Mandatory in Select Japanese Schools – News". *Nintendo World Report*. http://www.nintendoworldreport.com/news/17973/nintendo-ds-mandatory-in-select-japanese-schools.

Deci, E.L.; Ryan, R.M. (2000). "The 'What' and 'Why' of Goal Pursuits: Human Needs and the Self-Determination of Behavior". *Psychological Inquiry*, 11(4), 227-68. https://doi.org/10.1207/S15327965PLI1104_01.

Deterding, S. (2012). "Gamification: Designing for Motivation". *interactions*, 19(4), 14. https://doi.org/10/gd22z5

Doi, M. (2015). "Word Processing for the Japanese Language". *Engineering and Technology History Wiki*. https://ethw.org/Word_Processing_for_the_Japanese_Language.

Education soft (2023). "Saiten nabi: gakkōmuke no dejitaru saiten shisutemu" 採点ナビ 学校向けのデジタル採点システム (Digital Scoring System for Schools). https://www.kyoikusw.co.jp/campaign-pages/saiten-navi/.

Eisenberg, J.D. (2020). *SVG Essentials: Producing Scalable Vector Graphics with XML*. [Reprint]. Beijing; Köln: O'Reilly.

Ido Nobuhiko 井戸伸彦 (2015). "Ga to hen to dōtei shita nochi no tegaki kanji nyūryoku no saiten hyōka hōhō" 画と辺と同定した後の手書き漢字入力の採点評価方法 (Method of Scoring Evaluation for Handwritten Kanji Input after Identifying Strokes and Edges). IEICE Technical Report.

Ido Nobuhiko 井戸伸彦 (2019). "Hito te ni yoru saiten o kanzen ni daitai suru tegaki nyūryoku kanji jidō saiten shisutemu ni okeru kasū furī no jitsugen hōhō" 人手による採点を完全に代替する手書き入力漢字 自動採点システムにおける画数フリーの実現方法 (Implementing a Stroke-Free Approach in a Handwritten Kanji Automatic Scoring System Fully Replacing Manual Grading). JSiSE Research Report.

Ido Nobuhiko 井戸伸彦 (2020). "Tegaki nyūryoku kanji jidō saiten shisutemu ni okeru saiten jōhō no kōchiku hōhō" 手書き入力漢字自動採点システムにおける採点情報の構築方法 (Method to Build Scoring Data in a Handwritten Kanji Automatic Scoring System). IEICE Technical Report.

Ido Nobuhiko 井戸伸彦 (2022). "Tegaki nyūryoku kanji jidō saiten shisutemu ni okeru kabusoku no kaku no dōtei hōhō" 手書き入力漢字自動採点システムにお

ける 過不足の画の同定方法 (Method of Identifying Missing or Extra Strokes in a Handwritten Kanji Automatic Scoring System). IEICE Technical Report.

Ido Nobuhiko 井戸伸彦 (unpublished). "Development of an Automatic Scoring System for Handwritten Kanji Input that Implements the Standards for Correctness of Kanji in Agency for Cultural Affairs Documents". *Workshop: Kanji Writing Automatic Assessment Systems Development, Applications, and Future Perspectives in Education* (17 October 2024).

Kubota, M. (2017). "Post Study Abroad Investigation of Kanji Knowledge in Japanese as a Second Language Learners". *System*, 69, 143-52. https://doi.org/10.1016/j.system.2017.07.006.

Malone, T.; Lepper, M. (1987). "Making Learning Fun: A Taxonomy of Intrinsic Motivations for Learning". Snow, R.E.; Farr, M.J. (eds), *Aptitude, Learning and Instruction*. 1st ed. London: Routledge, 223-53. https://ocw.metu.edu.tr/pluginfile.php/2340/mod_resource/content/0/ceit706/week3/MakingLearningFun-ATaxonomyOfIntrinsicMotivationsForLearning.pdf.

Mantelli, A. (2021). "Development of a Kanji Reference Tool and its Moodle Integration". *EL.LE. Educazione linguistica. Language Education*, 10, 117-40. https://doi.org/10.30687/ELLE/2280-6792/2021/01.

Marcotte, E. (2011). *Responsive Web Design. New York: A Book Apart*. Brief Books for People Who Make Websites 4.

MEXT (2016). "Sankō shiryō: gakkō kyōiku ni okeru kanji shidō no arikata ni tsuite" 参考資料　学校教育における漢字指導の在り方について (Reference Materials: On Methods of Kanji Instruction in School Education).

Milanovic, Z. (2006). "Lecturing with a Virtual Whiteboard". *The Physics Teacher*, 44(6), 354-7. https://doi.org/10.1119/1.2336136.

Nakagawa, M. (unpublished). "Handwriting Recognition and Its Applications". *Kanji Writing Automatic Assessment Systems Development, Applications, and Future Perspectives in Education* (17 October 2024).

Nakagawa, M. (2024b). *Nakagawa Laboratory*. Tokyo University of Agriculture and Technology. https://web.tuat.ac.jp/~nakagawa/en/nakagawa.html.

Neos (2022). "Jidō saiten de machigai ga hitome de wakaru, okosama no jigaku jishū o sapōto: Aoba Shuppan teikyō no dejitaru kyōzai 'Dejitaru kanji purasu' 'Dejitaru keisan purasu' o Neos ga kaihatsu" 自動採点で間違いが一目でわかる、お子様の自学自習をサポート　青葉出版提供のデジタル教材　『デジタル漢字プラス』『デジタル計算プラス』をネオスが開発 (Automatic Scoring System Shows Mistakes at a Glance, Supporting Independent Study). https://news.neoscorp.jp/news-aob-digitaldrill/.

Nintendo (2007). "Tadashii kanji kakitori-kun" 正しい漢字かきとりくん (Correct Kanji Writing-Kun). https://www.nintendo.co.jp/ds/software/a8kj/index.html

Oviatt, S.; Arthur. A.; Cohen, J.(2006). "Quiet Interfaces That Help Students Think". *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, 191-200. Montreux (Switzerland): ACM. https://doi.org/10.1145/1166253.1166284.

Riekkinen, J. (2015). *A Content Analysis of Kanji Textbooks Targeted for Second Language Learners of Japanese* [MA thesis]. Stockholm: Stockholm University.

Sloan, D. (2008). "Japanese Students Schooled with Nintendo". *Reuters*, sect. Technology. https://www.reuters.com/article/technology/japanese-students-schooled-with-nintendo-idUSS27142984/.

Taele, P.; Koh, J.I.; Hammond, T. (2020). "Kanji Workbook: A Writing-Based Intelligent Tutoring System for Learning Proper Japanese Kanji Writing Technique with Instructor-Emulated Assessment". *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(08), 13382-9. `https://doi.org/10.1609/aaai.v34i08.7053`.

Zhang, S. (2021). "Review of Automated Writing Evaluation Systems". *Journal of China Computer-Assisted Language Learning*, 1(1), 170-6. `https://doi.org/10.1515/jccall-2021-2007`.