

A DIGITAL TRANSFORMATIVE EDUCATION METHOD FOR JAPANESE CLASSICAL LITERATURE – HYAKUNIN-ISSHU E-LEARNING APPLICATION –

Eri Yokoyama[†]

Hiroshi Sunaga[†]

[†]Osaka Institute of Technology

ABSTRACT

We propose a new e-learning application to promote digital-transformative education in classical literature classes. To encourage the studies of Hyakunin Isshu, a famous Japanese collection of Wak a poems, this application uses a quest-game format. The basic idea of the game is that a player tries to navigate a virtual labyrinth full of Hyakunin Isshu picture cards by answering related questions. In the process, the player is expected to learn about the background of the poems and the writers. This application, where Android's Bitmap drawing function is used to display a two-dimensional labyrinth three-dimensionally, identifies the location of the player within the labyrinth and manages questions related to randomly-placed picture cards as well as the player's performance at answering them. One of its key technical features is the function to convert the shapes of the picture card images from rectangles to trapeziums (or trapezoids in American English) following the perspective technique. We have also created a PC version of this application where JavaScript is used for implementation to achieve an equivalent effect. Their prototypes are generating a lot of interest among their users.

Keywords: Android application, JavaScript application, classical literature education, Hyakuin-Isshu

1. BACKGROUND OF THE STUDY

We have studied and implemented Web and smartphone applications based on the concept of Services Computing [1] - [9], covering various fields such as business, enterprise, personal life, e-learning, and entertainment. Basically these research results are for prototyping purposes, but for practical use we have introduced e-learning applications in general education courses, in particular, literature classes in the Faculty of Information Science and Technology at Osaka Institute of Technology. To motivate science students in general education classes, we thought that a digital-transformative approach is necessary. Thus we have introduced them and our earlier approaches have been recognized as successful [5][9]. This paper aims at creating a new application that exceeds the previous

ones. It is required that the new one should make students feel and think more deeply about classical literature. To this end, we introduce a "quest" element to the application to further engage students in the details of literary works. We take up Hyakunin Isshu (a series of classical Japanese poems selected by Teika Fujiwara in the 13th century) as the material for the quest. In the following sections, we will discuss the general image and technical features of this application.

2. APPLICATION IMAGE

Fig. 1 shows the interior of the labyrinth. The basic idea of the game is that a player tries to navigate a virtual labyrinth full of Hyakunin Isshu picture cards by answering related questions. In the process, the player is expected to learn about the background of the poems

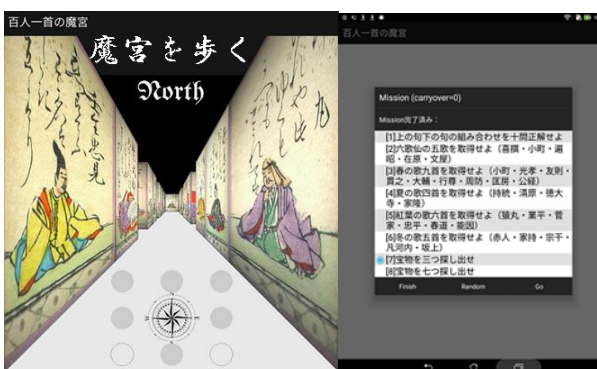


Fig. 1: Labyrinth of Hyakunin-Isshu

and the writers.

When the player walking in the labyrinth selects a Hyakunin Isshu picture card by touching it, the first half of the poem and a four-choice related question pop-up. The player needs to answer these questions to complete their missions.

Here are some of the missions prepared:

- Correctly match up ten pairs of the first and second halves of the poems.
- Find all the poems related to spring and match them with the second halves of the poems.
- Variations of the above (relating to other seasons)
- Find all the poems by a group of great poets called "Rokkasen" and match them with the second halves of the poems.

- Find treasure concealed in some picture cards.

Let's say the player chooses the first item on the mission list above. When they choose a picture card by touching it, the first half of the poem pops up along with a four-choice question asking for the second half of the poem (Fig.2). If the player's choice is correct, the selected answer will turn blue. Then, an explanation will be shown. When a question is answered, the card disappears and a blue wall is revealed. If the player touches the blue wall, an explanation related to this

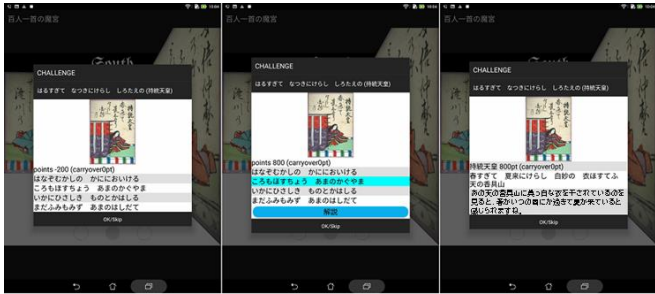


Fig. 2: Answering a question



Fig. 3: Blue wall and Popped up explanations

poem will pop-up (Fig.3).

If a treasure hunting mission is chosen, the player needs to walk in the labyrinth and answer as many questions as possible. If a treasure is found, it will be depicted as in Fig. 4. Because it is difficult to find treasure if the player randomly chooses cards, they can switch the map display from 3-D to 2-D, which shows the location of all the cards and treasures, but will have to spend points.

When one mission is completed, a pop-up showing the game status will be displayed (Fig. 5). The player can stop the game there, but higher scores will be earned if more missions are completed.

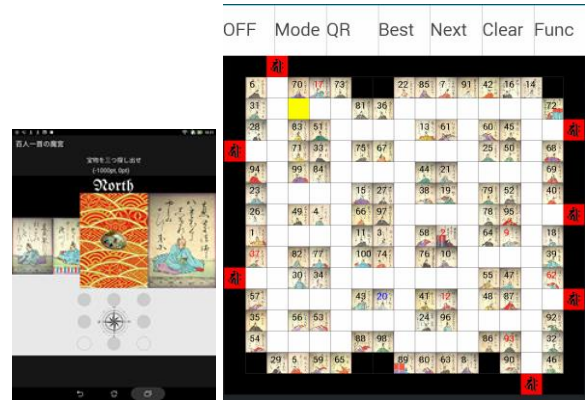


Fig. 4: Treasure card and 2-D map



Fig. 5: Mission complete display

3. TECHNIQUES FOR ANDROID APPLICATION

3.1. Management of labyrinth data

To implement this application, position data in the labyrinth are necessary. To create a labyrinth in the 2-D form as shown in Fig.4, a Web-based data management application is prepared (Fig.6). This page first provides a plain grid plane that is composed of more than one hundred squares. By touching a square (y-coordinate i and x-coordinate j) and selecting a color, the square is labelled as a path (0), wall (1) or picture card (2). Coordinate i, j is converted to a numerical value ($i \times \text{width} + j$) and vice visa. An example of this sequence is "... 2,2,2,1,1,1,2,0,0,0,2,2 ...". When a picture file is selected from the picture list page for a position with value "2", an image is drawn in this position. Corresponding to the position in the numeral sequence, the file name is set like "... Hyakuninisshu_030.jpg, Hyakuninisshu_020.jpg, Hyakuninisshu_070.jpg,....". These data are accessed from the Android application when the application is installed.



Fig. 6: Web-based Data management application

3.2. Implementation in Android

The android application reads in the data prepared in the previous section at initialization time. The one hundred picture files are contained in the assets folder of this application. As for the 2-D drawing, a similar programming technique is applied to the Web application. The numeral sequence is converted to 2-dimensional coordinate data. The application draws a square and sets colors in accordance with the grid value. If numeral 2 is found, the file name sequence will be read and the corresponding file name will be retrieved.

In this application, squares, images and texts are drawn on a Canvas [11] instance. A square is drawn by the drawRect() method. An image is first retrieved as a Bitmap by reading a picture file stored in the assets folder. For example, from the file name sequence, Hyakuninissu_030.jpg is designated and converted to a Bitmap. It is drawn by the drawBitmap() method.

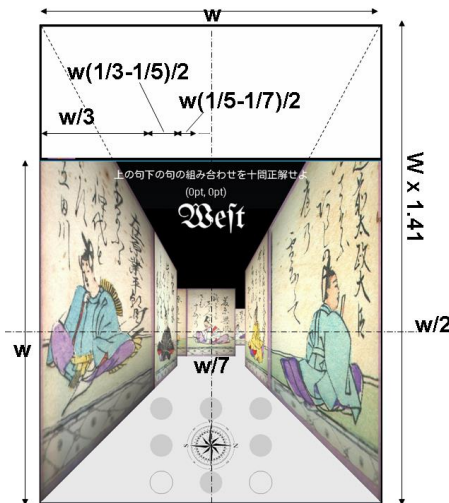


Fig. 7: 3-D description

To move the player in the labyrinth, there are control buttons for each direction, i.e., forward, forward left, forward right, left turn, right turn, backward, backward left, backward right and 180 degree turn. By

judging the player's current position and path information, the player moves in the designated direction.

For 3-D expression, pictures must be drawn stereographically and in perspective. Pictures in front of the player are drawn as rectangles, but pictures on either side are drawn as tetragons whose upper and lower hems are diagonal and cross at the vanishing point. As shown in Fig. 7, from the player's viewpoint, the nearest picture is depicted with a width of one third of the display width (w). The next one has a width of $w*(1/3-1/5)$, and the n th one has a width of $w*(1/(2n-1)-1/(2n+1))$. The aspect ratio is 1.407.

Pictures on either side of the player are distorted. To transform a rectangle to a trapezoid, the Android system provides the Matrix class [12]. As shown in Fig. 8, the mapping between all the squares in the original rectangle and all the trapezoid is defined by using linear equations.

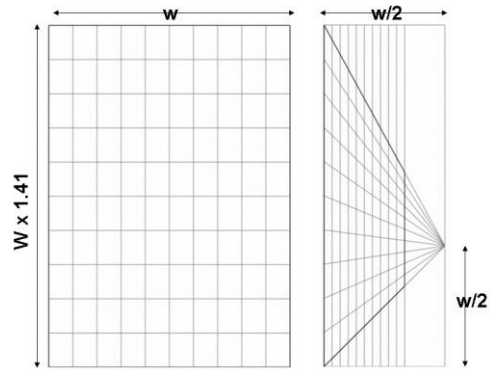


Fig. 8: Trapezoid transformation

3.3. Presenting questions

At the application initialization time, in accordance with the order of Hyakunin Isshu poems, necessary data are lined up. The data include the authors, the first halves, the second halves, the poem texts and modern translations, and they are dealt with as String arrays. The poem numbers of the six great poets, spring-related poems, summer-related poems, autumn-related poems, winter-related poems and autumn leaves related poems are also managed as arrays.

While walking in the labyrinth after having selected a mission, the player may find a picture related to their mission. By touching the picture, a question is presented in the form of the AlertDialog (a kind of pop-up window). From the poem number of this picture, the first half of the poem is depicted as a TextView element. As one of the four choices, the second half of this poem is set to an array. Then, three other poems are randomly selected and set to the array. These four array elements are reordered randomly, and depicted as RadioButtons. At this time, the position of the correct answer number is

stored, and it is judged whether or not the player's choice corresponds to this answer.

3.4. PC version

We have also created a PC version of this application where HTML-5 [13] and JavaScript [14] are used for implementation to achieve an equivalent effect. The PC version fully utilizes the large screens of PCs and offers a game mode which is unsuitable for Android. The PC version is implemented in the JavaScript language. The programming algorithm is similar but there are several differences.

First, picture file handling requires elaboration. Because picture file reading to make Image() instances must be completed before the picture drawing starts, the processing is deferred until all the image loading completions have been notified by the onload() callback function of the Image() class.

Next, the Matrix class is not offered by JavaScript. The conversion from a rectangle to trapezoid is neither easy nor visually perfect. The depicted picture is cut in the direction of the x axis with a 1-pixel width, and the height is scaled down in accordance with the diagonal line equation of the upper and lower hems of the picture. Regrettably, the transformed image looks a little unnatural.

4. CONCLUSION AND FUTURE WORK

Although the PC version has just been introduced in the classroom, it has already received some favorable comments. The Android version was exhibited at Open Campuses and Open Laboratories at our university, and it generated a lot of interest among the participants. Because the position of poem cards in the labyrinth is randomly changed at each trial, users can learn and enjoy Hyakunin Isshu poems with renewed interest. Also, answer choices for each question regarding the same poem are changed each time, leading to the same effect.

We conducted a pop quiz on our students before and after they studied with this application. The percentage of correct answers was 40.6% for the former and 59.3% for the latter. Much of this improvement can be attributed to its quest game format. The effect was most notable on the question "Choose the poems created by Rokkasen". This is a tough question for students who have only learned Hyakunin Isshu poems by rote memorization. This application includes not only simple match-up questions but also ones that require a deeper understanding of the poems, for example question about choosing spring (or autumn) poems, where some knowledge of season words is essential. Also the point system, particularly the one about searching for treasures, has turned out to be helpful in boosting the students' motivation. From these viewpoints, this application is expected to enhance learning effect in the

classroom. As a final note, the application's framework can be used in various other situations, such as in museums and festivals.

5. REFERENCES

- [1] Hiroshi Sunaga, "Recording and Sharing Techniques for Figures and Lines Drawn Using the Android View Class", IEICE Technical Report, SC2015-29 (2016-03)
- [2] Hiroshi Sunaga, Keishuke Nakajima, "Data Management for 3D Floor Plan Drawn on HTML5 Canvas - Floor Plan Data Management -", IEICE Technical Report, SC2017-15 (2017-08)
- [3] Hiroshi Sunaga, Eri Yokoyama, "Development of E-learning Applications Using Image Content and Introduction to Literature Classes", 4th Fall Meeting of Federation of Imaging Society, 3A01(2017-12)
- [4] Hiroshi Sunaga, Ayaka Morikawa, "Memory Usage Reduction Method for Bitmap Drawing in Android Programming - Evacuation Guidance Application -", IEICE Technical Report, SC2018-18 (2018-08)
- [5] Eri Yokoyama, Hiroshi Sunaga, Makoto Hirayama, "COOPERATIVE E-LEARNING APPLICATIONS BASED ON HTML-5 CANVAS FOR JAPANESE CLASSICAL LITERATURE EDUCATION", IEVC2019 (2019-08)
- [6] Hiroshi Sunaga, "E-Learning Applications for Neural Networks and Genetic Algorithms", IEICE Technical Report, SC2019-15 (2019-08)
- [7] Yuma Otsuchi, Hiroshi Sunaga, Eri Yokoyama, "Implementation and Utilization of Application for Learning Cursive Style Japanese Syllabaries ?Kana Matching-", IPSJ SIG-CH 2019 Poster-43
- [8] Chiharu Yamashita, Hiroshi Sunaga, "E-Learning Application for Japanese Program Language Nadeshiko", IEICE Technical Report, SC2019-45 (2020-03)
- [9] Eri Yokoyama, Hiroshi Sunaga, Makoto Hirayama, "Cooperative E-learning Applications Based on HTML-5 Canvas for Japanese Classical Literature Education", IEEEJ Transactions on Image Electronics and Visual Computing Vol.8 No.2 (2020)
- [10] Osaka Institute of Technology, Faculty of Information Science and Technology, <http://www.oit.ac.jp/is/>.
- [11] <https://developer.android.com/reference/android/graphics/Canvas>
- [12] <https://developer.android.com/reference/android/graphics/Matrix>
- [13] WHATWG and W3C, "HTML5", <http://www.html5.jp/>
- [14] JavaScript (ECMAScript), ECMA-262

