Area Type Judgment of Mathematical Document Using Movements of Gaze Point

Ryoji Fukuda, Yuki Ogino
rfukuda@oita-u.ac.jp
Faculty of Engineering,
Oita University
Japan

Kenji Kawata, Takeshi Saitoh
saitoh@ces.kyutech.ac.jp
Kyushu Institute of Technology
Japan

Abstract: *In this study we attempt to determine the area of a mathematical document on which a person is concentrating, using the movements of their eyes. Training data using this analysis were created using a gaze-point extraction method. There are thousands of image files generated by a movie. We also developed a software tool to determine an adequate area for each image file. Our results will be valid for the analysis human intention when a person reads a mathematical document.*

1. Introduction

We often attempt to express visual content using nonvisual methods. In visual content, e.g, a figure in a mathematical document, there are many information elements and the relations among these elements also convey information needed to understand the document. Usually, we only need some of these elements for understanding, and hence there are many unnecessary elements. In nonvisual communication, these elements may complicate document comprehension. In such a case, the importance or priority of an element is an essential factor to improve the understandability of such documents in nonvisual communication. We attempt to understand them using the eye movements of a sighted person [2]. The selection or understanding of graphical elements may be done subconsciously, and eye movements are one physical representation of this process.

To use extracted gaze points for this analysis, we developed an accurate extraction method to determine what part of the figure a person is concentrating on. In this analysis we seek a method to recognize the area type, i.e., a graphical or text area, without using the positional information of the target image. That is, the aim of this study is to develop a technique that can discriminate the area type using only eye movements. There are two main reasons for an approach that uses fewer information elements. One is hardware-related. We use an inside-out camera (two cameras attached to a pair of glasses, see Section 2). However, we need only one camera to obtain eye movements, and simpler hardware may be better for the creation of data. Another reason is the development of various feature values. We expect that the features obtained under these restricted conditions may be different from others obtained using various methods..

We used gaze-point data to obtain training data. Our extraction method for gaze points uses a linear regression, where the extracted vector data are linearly equivalent to the center coordinates of the pupil ellipses. Our extraction data are pairs consisting of a two-dimensional vector and a BMP file of the viewed image. The vector is the gaze position in the corresponding BMP image. We can obtain the area type corresponding to the position where the person is concentrating using some quadrangles in the viewed images. However, these quadrangles slightly change, and we have to adjust them in each BMP file using our custom-built software.

There are thousands of files in one set of data, and we can adjust these positions in a few minutes using our software. Using these training data, we determined some features that can discriminate the area types. We mainly use the directions and sizes of the first derivative of the eye movements. We explain these details in Section 4.

2. Creation of Gaze-Points' Data

We analyzed eye movement to determine the area type. For this analysis, we needed some correct training data, and hence used gaze data. We previously developed an extraction method for gaze points and analyzed their characteristics [1],[2],[3]. Using the same hardware and the methods, we created additional gaze-point data for this study. In this section, we outline the extraction method and the technique to obtain gaze-point data.

2.1 Inside-out Camera

For the extraction of gaze points, we used an inside-out camera. This camera consists of two USB cameras: an eye camera and a scene camera. The eye camera captures the user's left eye and the scene camera captures the user's visual field. The base of this camera is a pair of glasses without the lenses. Three infrared LEDs are also attached to obtain clear pupil images.

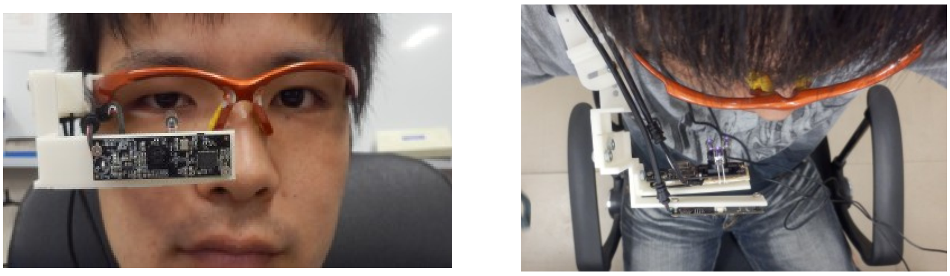


Figure 2.1 Inside-out Camera

2.2 Pupil Extraction

Using the images obtained by the eye camera, we extract a pupil ellipse. This task is divided into the following steps.

1. Capture an image.
2. Determine the temporal eye position using the combined separability filter.
3. Determine the edge lines using a simple separability filter.
4. Using ellipse estimation, localize the pupil in the image.

By this method, we can obtain the center coordinates of the pupil very quickly.

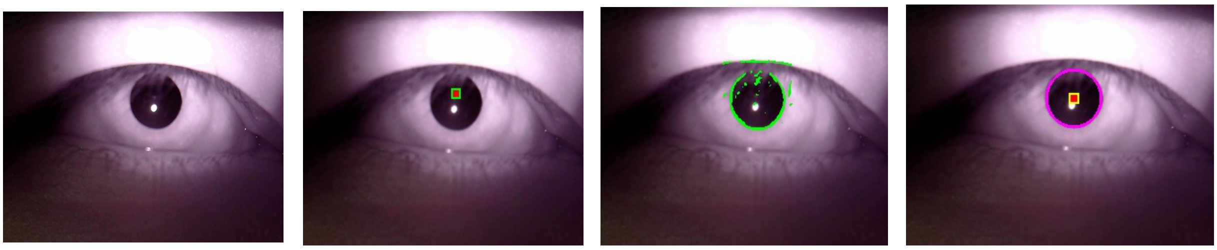


Figure 2.2 Extracted Pupils

2.3 Estimation of Gaze Points

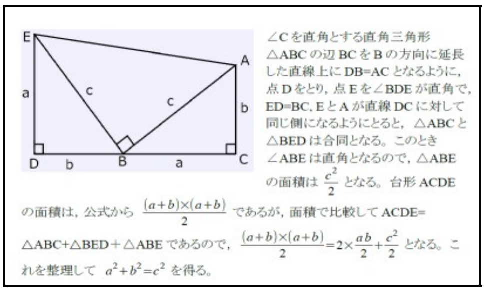
Another advantage of our system is its easy calibration. We use a fingertip image for this calibration. The user moves his/her finger in front of the camera and follows its movement. He/she must be careful not to move his/her head, because we need the various positions of the pupil center from this calibration. We also must consider the background color around the finger position. When the background color of a position is very similar to human skin color, this can affect the detection of the fingertip in the image.

We then obtain several pairs of fingertip and pupil center coordinates. The parameters of the linear transform are the results of a linear regression using these data. Thus, the gaze points are the translated positions of the centers of the pupils. We determine the movement of the eyes using the movement of the gaze points.

Figure 2.3 Calibration

2.4 Documents Used in the Analysis

Our training data are the gaze points of a person who is reading some mathematical documents. The target document is one proof of the Pythagorean theorem (Figure 2.3). The explanation in the text part is translated into voice output using the voice synthesis software "VW Show"(Knowledge Creation Co.).



E
A
a
c
c
b
D
b
B
a
C

∠Cを直角とする直角三角形△ABCの辺BCをBの方向に延長した直線上にDB=ACとなるように，点Dをとり，点Eを∠BDEが直角で，ED=BC, EとAが直線DCに対して同じ側になるようにとると，△ABCと△BEDは合同となる。このとき∠ABEは直角となるので，△ABEの面積は$\frac{c^2}{2}$となる。台形ACDEの面積は，公式から$\frac{(a+b)\times(a+b)}{2}$であるが，面積で比較してACDE=△ABC+△BED＋△ABEであるので，$\frac{(a+b)\times(a+b)}{2}=2\times\frac{ab}{2}+\frac{c^2}{2}$となる。これを整理して$a^2+b^2=c^2$を得る。

Figure 2.4 Sample Document

(English Translation) △ABC is a right angle triangle with the right angle C. Let D be a point such that B is an inner point of line segment DC, DB=AC, and E be a point such that the angle BDE is a right angle, and that E and A are in the same area with respect to the line DC. Then △ABC and △BDE are congruent, and the angle EBA is a right angle. Thus, the area of △EBA is $\frac{C^2}{2}$. The area of quadrangle ACDE is $\frac{(a+b)(a+b)}{2}$. Comparing the areas, we have $\frac{(a+b)(a+b)}{2}=2\times\frac{ab}{2}+\frac{c^2}{2}$. Therefore, $a^2+b^2=c^2$.

There are many methods of proving the Pythagorean theorem. This proof was given by James Abram Garfield (the 20th President of the United States) [4].

2.5 Data Creation

Using the document given in the previous subsection, we recorded the gaze points of a person while he/she read the document. There were five test subjects, all of them are male university students. They performed the experiment under the following set of conditions uniformly applied to all experiments (Figure 2.5).

1. They read the document until they understood the contents.
2. The document was printed in a large font, and they read it from a distance of 50 cm.
3. They read the document as the voice output of the document.
4. The system extracted the gaze points.

From the movie created by the scene camera, we obtained several image files. The extraction results are *x*-*y* coordinates in the corresponding image files. During the creation of data, the test subjects focused on not moving their heads. Roughly speaking, the movement of the extracted coordinates nearly equals that of the positions in the original figure.

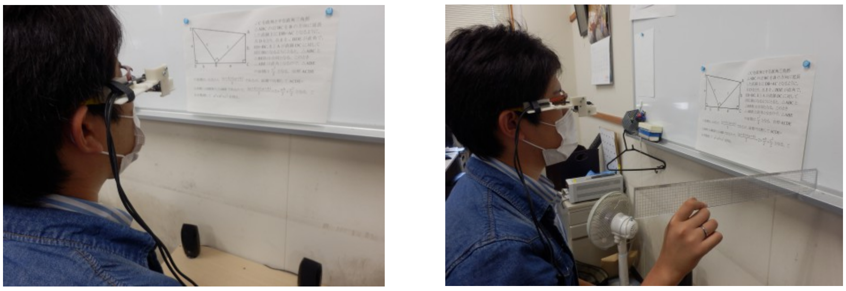


Figure 2.5 Data Creation

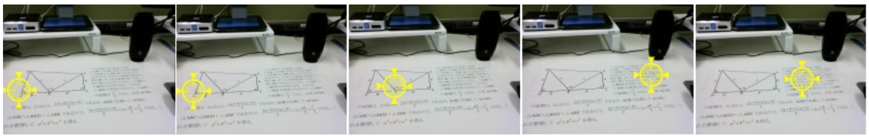


Figure 2.6 Example of Extracted Gaze Points

3. Software Tool for Adjusting Areas

We set several quadrangles that represent text areas or figure areas. For the training data, we determine the area type on which the test subject concentrates on using these quadrangles. In the images obtained by the scene camera, the positions of figures change gradually. We cannot ignore these differences over a long passage of time. Figure 3.1 shows a composite image of two half-transparent images at different times. The number of image files is more than one thousand, therefore, it will take a very long time if we check the coordinates of the quadrangles for all image files. We have to use a realistic approach.

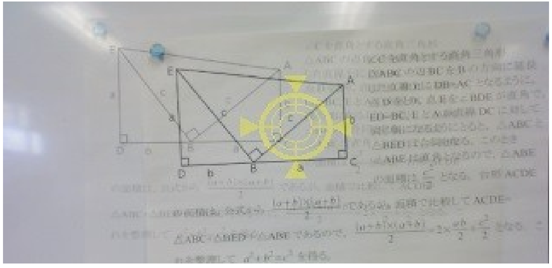


Figure 3.1 Position Change in Viewed Camera Images

3.1 Outline of the System

To obtain the coordinates of all of the vertices of the quadrangles for each image file, we use the position of the mouse cursor. One image of the scene camera is displayed in our system, and this is replaced by next one after a short interval (Figure 3.2). The system checks the coordinates of the position of the mouse cursor for every appearance of a new image. Then, we obtain every coordinate of the vertex of some quadrangle. The test subjects receive instructions not to move his/her head. Then the position of the figure changes very slowly, and we are able to chase one vertex using a mouse. The number of iterations for this task is the same as the

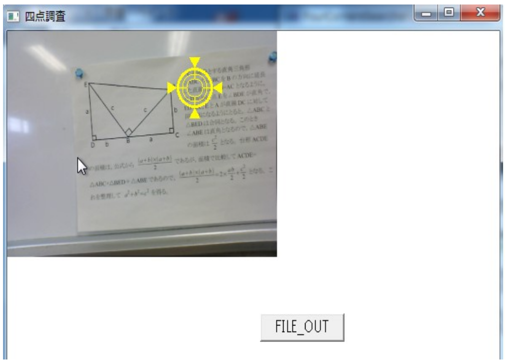


Figure 3.2 Screen Shot of the System

number of vertices. There are only a few area quadrangles, it took reasonable time to complete the series of tasks.

3.2 Procedure for Obtaining Coordinates

To begin, we input the base names of the image file names. The names of the image files consist of a base name part and an integer part. These files are automatically created by the gaze-points' estimation system. The system also requires the last file number. In the case where the estimation system fails to obtain a gaze point, the corresponding image file is removed. This happens, for example, when the eyes are closed.

By clicking the left mouse button, the system loads the first image file, and the next image is loaded at regular time intervals. Before beginning this task, the user determines one vertex, and he/she moves the mouse cursor according to the change of image position.

Table 3.1 Sample Data

Size of Image	320 x 240
File Number	1600
Time Interval	100ms
Vertices Number	8
Total Time	1280 s

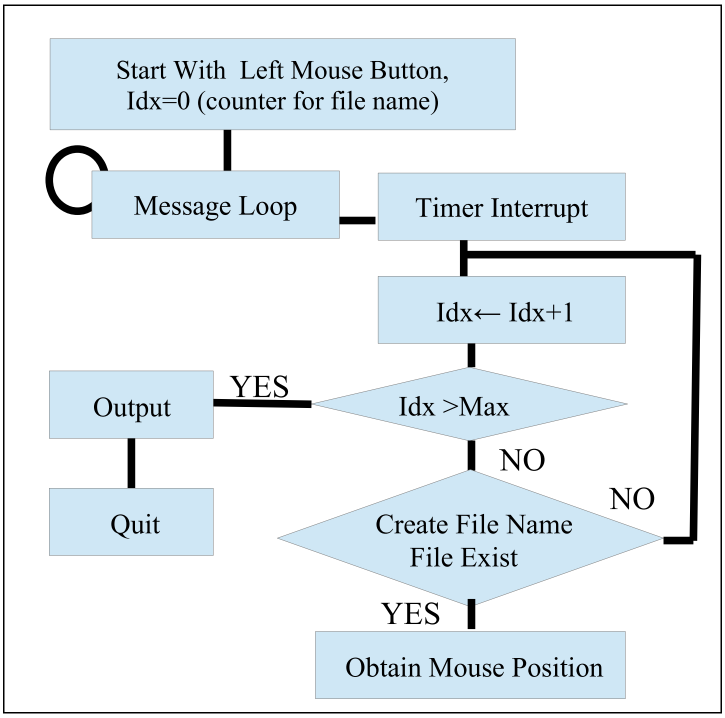


Figure 3.3 Flow Chart To Obtain Coordinates

4. Feature Values for the Judgment of Areas

The purpose of this study is to analyze human intention by eye movement. We developed an extraction method for the gaze points, and we analyzed the intention by using positional data [2], [3]. In these analyses we mainly use the relation between the gaze-point position and that of a target figure element. In general, the analysis of intention must be very delicate and we need many clues for this problem.

Sometimes, we are able to imagine someone's intention according to their eye movement. For example, consider the case in which a persons wife is harbouring doubt about a comment from her husband. If he cannot look into her eyes or avoids eye contact with her, his wife may feel strongly that some parts of his comment are not true.

In this study, we consider two area types: the graphical area and text area in some mathematical documents. In a text area, there are some sentences. These are definitions, proofs, explanations, etc. In a graphical area, there may be some triangles, quadrangles, and some additional lines.

The human approach or brain tasks may be different from each other. One usually reads a sentence in the text area. Then, in the standard case, he/she looks at every word in sequential order. On the other hand, one concentrates on some special elements or the relations between them. These differences may appear in the eye movements; therefore, we consider some feature values in this section.

4.1 Approximation by a Polynomial

The gaze-point data are given as a sequence of three values: (frame number, x-coordinate, y-coordinate). In the case where the system fails to detect the gaze point, the data for this frame is not stored. Then the frame number is proportional to the elapsed time. We approximate the x and y-coordinates by cubic functions of the frame number by the following procedure.

1. Consider two sequences: $\{(t_k, x_k)\}_{k=1}^{N}, \{(t_k, y_k)\}_{k=1}^{N}$,where t_k is the frame numbers.
2. Fix a frame number k_0 ($5<k_0<N-5$) and consider subsequences $\{(t_k, x_k)\}_{k=k_0-5}^{k_0+5}$, $\{(t_k, y_k)\}_{k=k_0-5}^{k_0+5}$.
3. Obtain the cubic functions $x(t)$ and $y(t)$ using polynomial regression.

Then, we approximate the first and second derivatives of the the coordinates (with respect to *t*) at the point (x_{k_0}, y_{k_0}) using the approximated functions $x(t)$ and $y(t)$.

4.2 Eye-Movement Speed

When one reads a sentence, the gaze point moves from the left to the right slowly. Let denotes the approximated move of the gaze point. Then, the speed of the movement is defined by

$$Spd(t)=|\vec{x}'(t)| \ .$$

On the other hand, when we look at a figure (graphical area), we often concentrate on several elements and compare them. Then $Spd(t)$ often takes on a large value compared to that when reading a sentence. We will show their features by a run of the item.

1. $Spd(t)$ does not take on large value when one read a sentence,except for the line feed cases.
2. Sometimes, $Spd(t)$ takes on a large value when one looks at a figure.
3. The average value of $Spd(t)$ changes according to the situation.

Using the quadrangle defining the area type, we obtained some sequences of gaze points

$$B_j=\{\vec{x}(t_k)\}_{S_j\le k<E_j} \ . \qquad .$$

For each *j*, the area type ("Text" or "Graphic") of the gaze point that belongs to the block B_j is unique. Then we consider the following feature value Lsr :

$$Lsr(j)=\frac{\#\{k : T_{Lsr}\le Spd(t_k); t_k\in B_j\}}{(E_j-S_j)} \ ,$$

where T_{Lsr} is the threshold for the large-speed value.

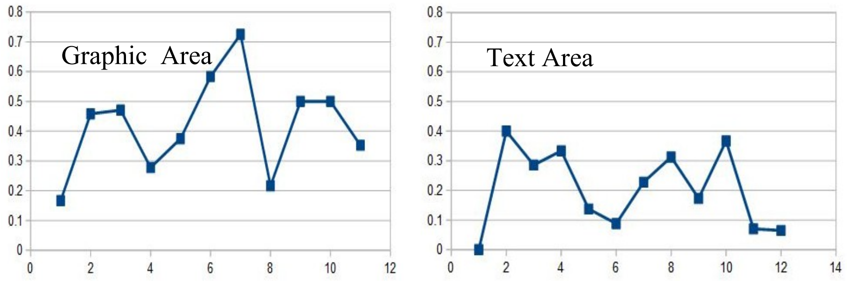


Figure 4.1 Lsr values for Both Areas

4.3 Number of Horizontal Turns

In many cases, a gaze point moves slowly from left to right while reading sentences. Then, the direction of movement does not change frequently, and we think that a count of the "horizontal turn" is valid for this judgment. We use the same notation used in Subsection 4.1 for $Sq(j)=\{\vec{x}(t_k)\}_{S_j\le k<E_j}$ and define

$$Htr(j)=\frac{\#\{k: x'(t_k)\times x'(t_{k+1})\le 0\,;t_k,t_{k+1}\in B_j\}}{(E_j-S_j-1)} .$$

The graphs in Figure 4.2 represent these values for both area types. It has a real chance to be effective: however, the ability is not stable and the average value will not exhibit good performance.

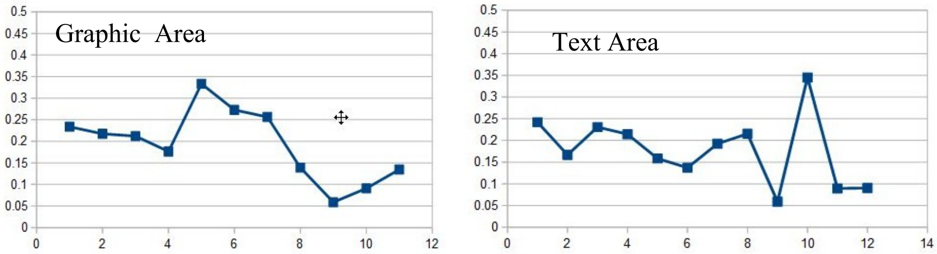


Figure 4.2 *Htr* Values for Both Areas

4.4 Horizontal Movement

We attempt to evaluate the horizontal eye movement using another feature value. Let $\{\vec{d}_j\}_{j=0}^{7}$ be a family of unit vectors equiangularly allocated on the unit circle (Figure 4.3). Using same notation as the previous subsection, we define

$$Hmr(j)=\frac{\#\{k:(\vec{x}'(t_k)\cdot\vec{d}_i \text{ takes max value at } i=0 \text{ or } 4)\}}{(E_j-S_j)} .$$

For a text area, this feature value sometimes takes on a large value, except for some unusual cases. It does not take on a large value for graphical areas in almost all cases; however, it may be difficult to set a certain threshold for this determination (Figure 4.4). These feature values are developed using some behaviors when reading text. One may move his/her gaze point from left to right slowly when he/she reads text document. However, some extra intentions or accidental movements may be incorporated, thus we need some higher dimensional or strictly statistical analysis to improve the determination of these eye movements.

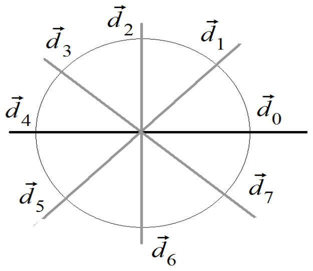


Figure 4.3 Eight directions

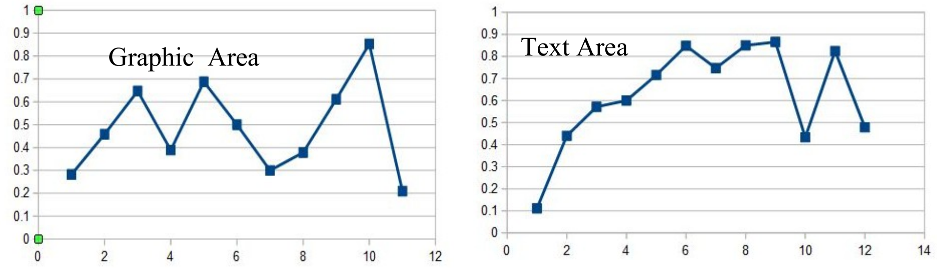


Figure 4.4 *Hmr* values for Both Areas

5. Conclusions

We analyzed the movements of gaze points for the judgment of area types, i.e., the text area and graphical area, in mathematical documents. We used an extraction method for the gaze points and developed an area-adjusting system for the training data. We consider three feature values to judge area types. We discussed their abilities in the previous section; however, we have several options for improvements: considering the situation, finding other features, or analyzing the ability of combined feature values. We hope to pursue some of these improvements in our future work.

References

[1] Iwagami J., Saitoh T., Easy Calibration for Gaze Estimation Using Inside-Out Camera , 20th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV2014), 57, 014, pp.292-297

[2] Fukuda R.,, Iwagami J.,, Saitoh T.,, Applicability of Gaze Points for Analyzing Priorities of Explanatory Elements in Mathematical Documents, Proceedings of th 19th ATCM, 2014, pp.254-260

[3] Fukuda R., Iwagami J., Saitoh T., Evaluating Importance of Information Elements in Graphical Content Using Gaze Points, Proceedings of the 18th ATCM, 2013, pp.254-260

[4] James A Garfield, (No title colum), The New England Journal of Education 3: 161, 1876