Applicability of Gaze Points for Analyzing Priorities of Explanatory Elements in Mathematical Documents

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

Junki Iwagami
iwagami@slab.ces.
kyutech.ac.jp
Kyushu Institute of Technology
Japan

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

Abstract: *For this report, we studied the applicability of gaze points for analyzing the priorities of elements in an explanation of mathematical concepts expressed through figures. There are many informational elements in a mathematical document, and we can understand some of them through the help of attached figures. In non-visual communication, we can obtain such elements through spoken text or tactile information, but can grasp only a few at a single time. Thus, we must rely on the applicability of gaze points for understanding human intentions.*

1. Introduction

We often use figures in a document, and sometimes the document is estimated in the point that how effectively the figures are used. A presentation document can be made by paying careful attention to the adequate selection of figures. On the other hand, graphical contents are barriers in communication for people with visual disability. To remove these barriers, we have to provide the same information in a non-graphical manner. In this report, we attempt to analyze the eye movement to recognize the viewer's intentions when obtaining a set of elements from visual information.

Graphical contents often contain very large amounts of information elements, and among them, the human eye-brain system selects an adequate set of elements. It may be very difficult to grasp the tasks of this system because they occur very quickly and subconsciously. We consider some eye movements to be affected by these intentions. The aim of this report is to find the gaze points, the positions that an individual looks at, when reading a mathematical document, and to study the applicability of the data on such gaze points for judging the priorities of information elements.

A graphical structure of information is often considered. The concepts or properties are represented by nodes, and edges denote their mutual relations. Idea processors are software tools used to assist human consideration. A user creates a directed or undirected graph according to the targets. The user may understand the structure, and using it, can find the next optimal step. The structures of mathematical contents are often complicated, especially when some graphical contents are attached. If a sighted person has sufficient intelligence to understand the situation, they can pick up an adequate set of elements from a mathematical document through certain figures. Our objective is to create a substitution of non-visual information elements equivalent to the above set of elements.

In this study, we attempt to detect a gaze point using an inside-out camera system, which consists of two cameras: an eye-camera and an object camera. Using this hardware, we create a system to estimate the degree of attention for graphical elements and their relations using gaze points when a person reads a mathematical figure [1]. In this case, the nodes of the information graph are mathematical concepts: a point, a line segment, a triangle, and so on, and the edges are their relations. Typically, when trying to understand a mathematical document, we grasp simple sentences (properties, definitions, and so on) one at a time. The nodes of our information graph are therefore small sentences, which we call "explanatory elements," or simply, "explanations," in the sequel. If such explanations are given verbally, through voice or braille output, a person with visual disability can grasp them. In a mathematical document, these explanations may be connected with some graphical elements in a figure containing some other explanations that are not expressed verbally. Moreover, the shapes or positional relations may be important informational elements. We then analyze the movement of the gaze point when the person reads the mathematical document, and consider the role of the corresponding elements in an explanation graph.

2. Detection of Gase Points

We used the same hardware as in our previous report [1]. The applicability of gaze points for an understanding of human intentions must depend on the correctness of the detection. Next, we explain the outline of the detection method. The software must be adjusted for one person, which is not an easy task; however, the gaze points are sufficiently correct after fine adjustments. In this step of our research, we do not have a fixed performance requirement for the detection of gaze points. Therefore, our consideration may differ according to the performance, and we have to consider the ability to understand human intentions along with the detection methods and their performances.

2.1 Inside-out Camera

A gaze point is the point upon which a person's eyes are focused. Gaze points are used to analyze human intentions.

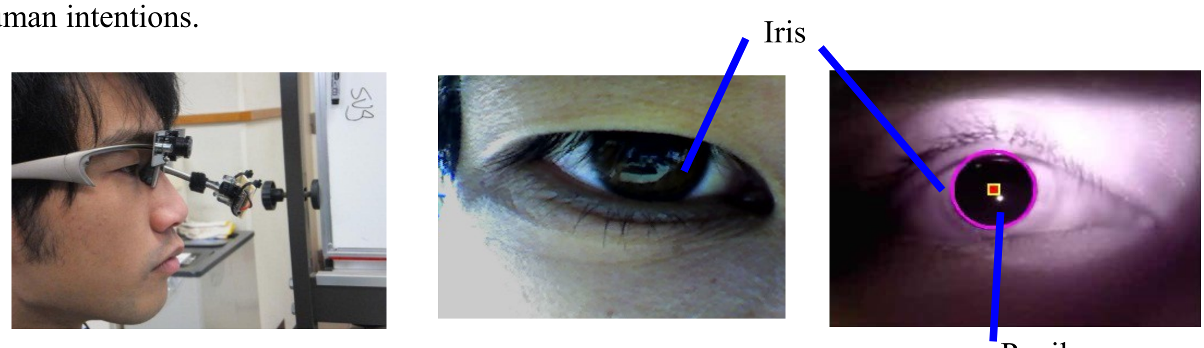

Figure 2.1 Inside-out Camera and Pupil

The eye-camera and screen camera are used to obtain images of the eyeball and target figure, respectively. Together, these two cameras are called an “inside-out camera.” The eye-camera captures images of the eyeball using infrared light, as it has very low intensity and is not visible to the naked eye. Hence, it does not affect the user's vision. We can obtain a clear pupil image using the infrared light and a camera.

2.3 Pupil Extraction

We use the center point of a pupil to estimate gaze points. The first step in gaze estimation is to extract the pupil from an image obtained by the eye-camera using the following procedure.

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.

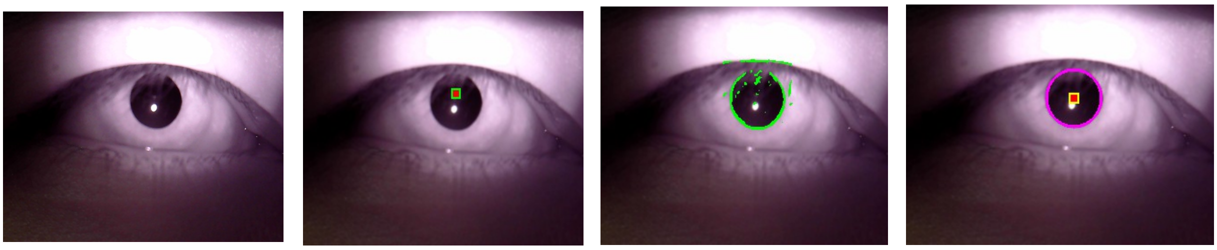

Figure 2.2 Extracted Pupils

2.4 Calibration

It is necessary to adjust the pupil center position in the eye camera image such that it corresponds with the gaze point captured by the scene-camera. In our system, the user adjusts these positions using his/her fingertip in the image captured by the scene-camera. First, the system obtains the two images from both cameras. Next, we detect the pupil and fingertip in these images. For fingertip detection, we use shape and color features until the pupil and fingertip are successfully detected, and a pupil and gaze point pair is obtained. After obtaining several pairs, we estimate the relation with a linear transform using a standard least squares linear regression.

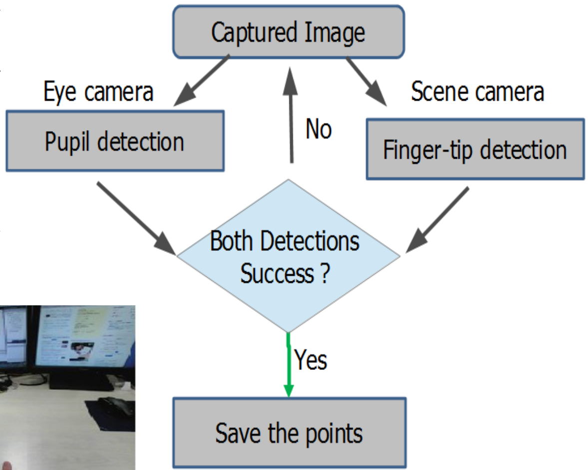

Figure 2.3 Procedure of Calibration

2.5 Detection Output

To obtain the position of a gaze point, two phases are necessary: input and analysis. First, we obtain two moving images using the inside-out camera. The system then estimates the positions of the input gaze points using a linear transform with the parameters obtained in the above calibration. The system obtains several BMP pictures using the scene camera, and we obtain the position of the gaze point for each picture. Our output is a series of pictures with blue cross marks at the corresponding gaze points.

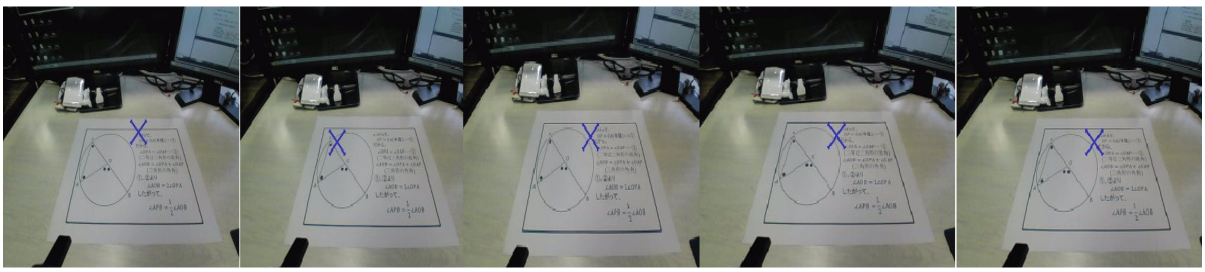

Figure 2.4 Output of the Gaze Points

3. Explanation Graph

We use graph structures when dealing with some complicated consideration targets. The nodes are sets of concepts or opinions, and the edges are their relations. A figure in elementary geometry often contains many elements of information, and a sighted student may obtain all of them. The figure is thus a reminder of these information elements, and he/she may give a brief proof using this figure in a moment. In our previous report [2], we considered the use of information graphs with certain concepts: a circle, a line segment, and points, for example, as the nodes and their relations; and the diameter, edge points, and center, as the edges. However, these structures are too detailed in a standard consideration.

In this report, we assume that the target recipient is a visually handicapped student and understand the concepts of elementary geometry, for example, a student who lost their eyesight after graduating from elementary school. Thus, the graph structure of a figure does not need to be too fine. We consider a graph that consists of several brief explanations. In this section, we explain the rules to express these graphs.

3.1 XML Elements of the Document

An XML element is expressed using a tag name that we define, for example "Point." Such an XML element starts with <Point [*attr*]> and ends with </Point>. In the case where there are no sub elements, it is expressed as <Point [*attr*] />. The part [*attr*] is replaced by a set of attribution definitions. Each attribution has a basic data type. Three attributes of our expression rules, "name," "type," and "refName," are string attributions (EgString). The attribution "name" is an identifier for the element and must be unique in the corresponding group.

Consider the case where the values of "name" are equal for two elements. If these elements are sub-elements of the same parent element, our grammar is contradicted. It is not contradicted if they are sub-elements of different parent elements.

The attribute "type" represents the data type of the element. The value of the "type" is the name of a basic data type or another data type defined according to our rules. The attribute "refName" is the name of another element. If there is a predefined element, we can refer to this element using this attribute. In this case, the data type of the corresponding element must be the same as the data type defined using "type."

Recall the element <Point>. We may describe a <Point> element as

<Point type="EgVec2" name="pt." v2Ctt="(0.0,0.0)"/>,

This statement implies that this element is named "pt" and its corresponding value (as a coordinate pair) is "(0.0, 0.0)." Sequential data are expressed as an "Array" element. We can define several "EgArrayElm" and an element "EgGenTerm." However, this grammar is not required to be calculable, and hence, we can describe concrete values or general formulae in general terms.

3.2 Description for Definitions

We describe a new data type using the element "EgDefinition." This element contains attributes "name" (EgString), "paraNum" (EgInteger), and sub elements of which is called "EgCondition." The attribution "name" is used as a tag name in the description of a corresponding element and its parameters are the corresponding contents. Using these parameters, we explain the definition "EgCondition" using the following XML example.

```
<EgDefinition name="IsosTr" paraNum="1">
    <Triangle name="ABC">
        <Side name="AB"/>
```

```
        <Side name="AC"/>
    </Triangle >
    <EgSurface >
        The isosceles triangle @ABC.
    </EgSurface>
    <EgCondition>
        @AB =@AC
    </EgCondition>
</EgDefinition>
```

This element has one parameter, and the parameter element(a trianble) has two sub elements. An element “EgCondition” describe the text output to explane about the condition of this definition. Its sub elements are definitions, properties, and “EgMess” elements (this represents simple string). If there are only one “EgMess” sub-element, a string is input directly.

In the “EgCondition”, we use parameters defined in this element. A parameter is expressed with “@” followed by the name of target element, and after the name string we need at least one space. In the case where we describe a sub element of a parameter, a period followed by the name of sub element is added after the parameter name. The description “@a.x > @b.x” implies that the x-coordinate of the point “a” is greater than that of the point “b”.

3.3 Expression of Properties

Mathematical properties are described using the element “EgProperty.” This element has attributes “name” and “paraNum” and sub elements “EgCondition,” “EgStatement,” and “EgProof.” These sub-elements are sets of XML elements that may consist of “EgDefinition,” “EgProperty” or some other element. Our grammar for this expression does not require all elements to be given. We define an expression method for various aspects of the documents.

The following XML element is an expression of the relation between a center angle and the corresponding circumference angle. In this element, we describe a condition, statement, and proof.

```
<EgProperty name="CircumAndCenterAngle" paraNum="4">
    <Circle name="c">
        <Point name="center"/>
    </Circle>
    <Angle name="agl1" >
        <Side name="s1"/>
        <Side name="s2"/>
    </Angle>
    <Angle name="agl2" >
        <Side name="s3"/>
        <Side name="s4"/>
    </Angle>
    <Triangle name="tr">
        <Point name="p1"> @c.center </Point>
        <Point name="p1"> @s1.stPt </Point>
        <Point name="p1"> @s2.edPt </Point>
    </Triangle>

    <DgCondition >
        @agl1 is a center angle.
        @agl2 is a circumferential angle.
```

```
        @agl2.s3 includes @agl1.s1.
    </DgCondition>
    <DgStatement >
        @agl1 = 2 @agl2
    </DgCondition>
    <DgProof>
        <BaseAngleIsosTr  paraNum="1">
            <Triangle refName="tr"/>
        </BaseAngleIsosTr>
        <TrOuterAngle  paraNum="1">
            <Triangle refName="tr"/>
        </TrOuterAngle>
    </DgProof>
</EgProperty>
```

4 Analysis of an Explanation Graph

In this section, we analyze the information element graph structure in the document illustrated in Figure 4.1. Originally, this document was written in Japanese, and our gaze point data were collected from a Japanese student. In Figure 4.1, Japanese sentences have been translated to English, keeping the position of the sentences constant.

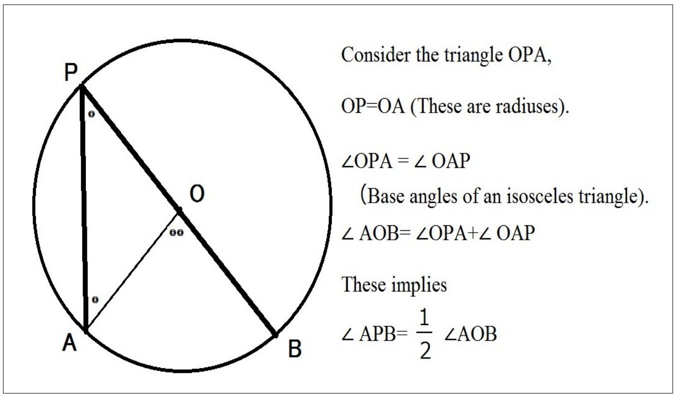

Figure 4.1 Target Document

4.1 Explanation Graph

This document can be expressed using the method detailed in the previous section (a sample expression can be found in the Appendix). Figure 4.2 is a simplified illustration of this explanation graph. The document gives a proof for the relation between a center and circumference angle. This property is stated in the last line of Figure 4.1. The graph in Figure 4.2 is a tree graph where the unique root node is the last sentence of the document.

Figure 4.1 is a simple figure, and the number of information elements is not large. There are many complex figures, especially in the field of elementary geometry. However, the corresponding XML document is correspondingly small because some related elements are attached. For the practical use of such documents, it will be necessary to develop automatic creation functions and an adequate user interface.

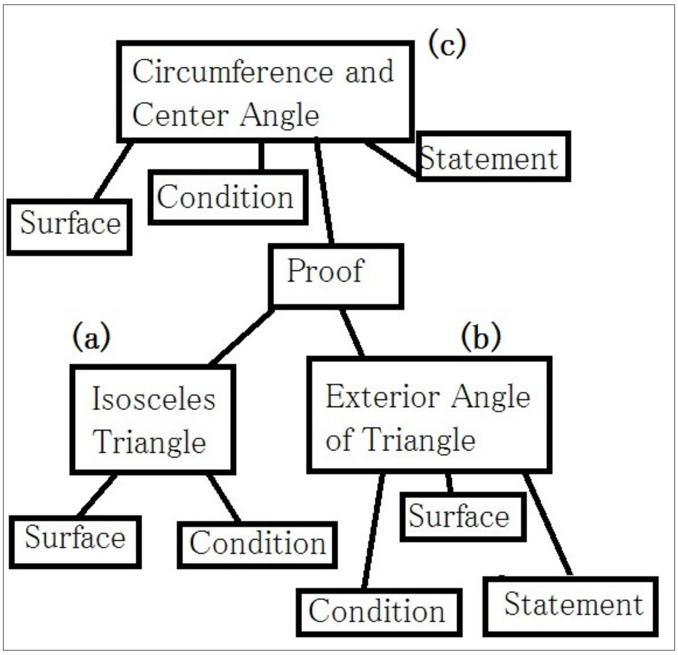

Figure 4.2 Explanation Graph

4.2 Analysis of Gaze Points

The system for detecting gaze points was implemented for a specific person. We obtained one series of gaze points while the test subject read the document (Figure 4.3) and collected the 20 gaze points that are listed in Figure 4.3. At a rough glance, ①–□ connect to node (a), □–□ connect to node (b), and ⑱–⑳ connect to node (c). In this case, the test subject read the text first and mainly concentrated on the first part of the proof.

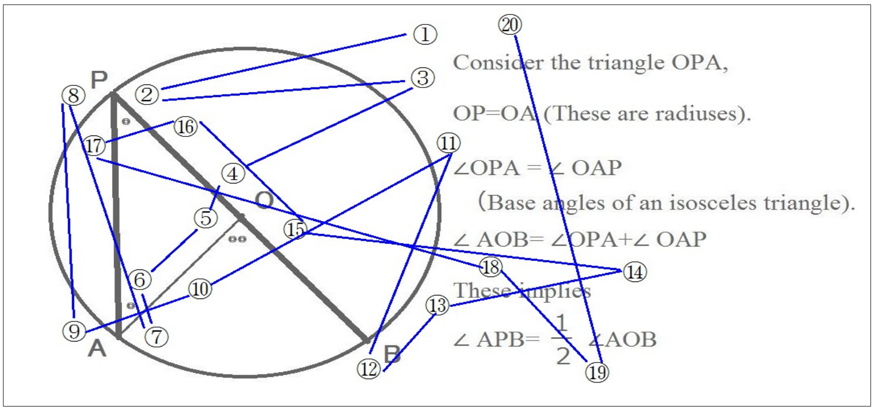

Figure 4.3 Move of Gaze Points

5. Conclusion

In this paper, we analyzed the graph structure of a mathematical document using the gaze points of a test subject while it was read. We created the graph structure considering the explanation of the document. We defined rules to create explanation graphs. Comparing the graph structure of the explanation to gaze-point movements, we observed how the test subject read the document.

However, this gaze point detection system is not yet online, and is not sufficiently generalized for anonymous test subjects. Moreover, we need an automatic creation and optimization method for the explanation graphs. We will then be able to determine any clear relations between the graph structures and gaze-point movements. These are directions for future research.

References

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

[2] Fukuda, R., Expression Rules of Directed Graphs for Non-visual Communication, Lecture Notes in Computer Science, Computers Helping People with Special Needs, 13th International Conference, ICCHP 2012,pp. 182-185

APPENDIX

```
<EgDefinition name="IsosTr" paraNum="1">
    <Triangle name="ABC">
        <Side name="AB"/>
        <Side name="AC"/>
    </Triangle >
    <EgSurface >
        The isosceles triangle @ABC.
    </EgSurface>
    <EgCondition>
        @AB =@AC
    </EgCondition>
</EgDefinition>
<EgProperty name="TrExteriorAngle" paraNum="5">
    <Triangle name="tr"/>
    <Angle name="in1"/>
    <Angle name="in2"/>
    <Angle name="in3"/>
    <Angle name="out"/>
    <EgCondition>
        @out is the outer angle
        corresponding to @in1.
    </EgConsdition>
    <EgStatement>
        @out = @in2 + @in3
    </EgStatement>
</EgProperty>

<EgProperty name="BaseAngleIsosTr" paraNum="1">
    <IsosTr name="ABC">
        <Side name="AB"/>
        <Side name="AC"/>
        <Angle name="CAB"/>
        <Angle name="CBA"/>
    </IsosTr >
    <EgStatement name="st">
        @CAB=@CBA
    </EgStatement>
    <EgSurfae refName="st"/>
    <EgProof name="pr">
        If @AB=@BC, then @CAB = @CBA.
    </EgProof>
    <EgDetail refName="pr"/>
</EgProperty>
<EgProperty name="CircumAndCenterAngle"
paraNum="4">
    <Circle name="c">
        <Point name="center"/>
    </Circle>
    <Angle name="agl1" >
        <Side name="s1"/>
        <Side name="s2"/>
    </Angle>
    <Angle name="agl2" >
        <Side name="s3"/>
        <Side name="s4"/>
    </Angle>
    <Triangle name="tr">
        <Point name="p1"> @c.center
</Point>
        <Point name="p1"> @s1.stPt </Point>
        <Point name="p1"> @s2.edPt </Point>
    </Triangle>
    <DgCondition >
        @agl1 is a center angle.
        @agl2 is a circumferential angle.
        @agl2.s3 includes @agl1.s1.
    </DgCondition>
    <DgStatement >
        @agl1 = 2 @agl2
    </DgCondition>
    <DgProof>
        <BaseAngleIsosTr  paraNum="1">
            <Triangle refName="tr"/>
        </BaseAngleIsosTr>
        <TrOuterAngle  paraNum="1">
            <Triangle refName="tr"/>
        </BaseAngleIsosTr>
    </DgProof>
</EgProperty>
```