

Non-Visual Diagram Navigation Using Microsoft's InkAnalysis Tool

Dennis Schwab

527 W. Broadway

Winona, MN 55987

(01) 507-474-1604

Denny_Schwab@yahoo.com

ABSTRACT:

Non-visual navigation of diagrams has been a problem for as long as non-sighted people have existed. This paper will discuss the feasibility of using Microsoft's InkAnalysis tool as a means of navigating a diagram in a non-visual way. We will show that it is possible to navigate a graph diagram using the tool.

General Terms

Performance, Human Factors

Keywords

Accessibility, diagram navigation

INTRODUCTION

Diagrams are an intricate piece of any student's education. Kennel states: "In computer science and engineering, diagrams are used to represent interrelations. These diagrams do not represent objects, but connections between objects. Such diagrams are often used to complement a text and are an important part of a report or paper". [6] A diagram allows students to get information in ways that cannot be duplicated by any other means, and these diagrams come in almost every shape and size. The type of diagram that this paper focuses on is one where objects are connected together by drawn lines. Brown and associates explain this idea very effectively: "There are many ways to represent these diagrams, but a generalization for this type of diagram is called a graph. In mathematical terms a graph is composed of a collection of nodes and edges. Examples include universal modeling language (UML) diagrams, process flowcharts, organizational structure hierarchies, and molecular diagrams. Graphs form the basic component of many diagrams and the ability to deliver graphs non-visually is vital if print disabled readers are to be able to use this form of diagram". [3]

For non-sighted individuals, understanding a diagram has always been a tedious undertaking. They must rely on sighted help to get the necessary information across to them in some way. Predominantly this requires Braille printouts or having someone actually describe the diagram for the non-sighted individual. This paper will examine the feasibility of using a tool that will allow the non-visual user to browse a diagram.

As Dr. Francioni of Winona State University and Ann Smith of Saint Mary's University have stated, "many students with disabilities use computer technology in their day-to-day lives for communication and/or access to information that would not otherwise be available to them"[5] That is why the prospect of a

software tool able to represent a graph to the visually impaired would be beneficial to those students. The tool this paper will focus on is InkAnalysis from Microsoft. This software tool can be used on any tablet PC. The fact that the tool can be used on a Microsoft based PC is very useful, as the visually disabled person most likely has a PC or can get access to one. Francioni and Smith continued with "the most basic piece of assistive technology for someone who is blind or who has a severe visual disability is what is called a screen reader. These tools speak the information that is on the screen out loud". [5] Using a screen reader the non-visual student can get access to the information from InkAnalysis and be able to navigate a graph.

Besides giving the non-visual person the ability to access diagrams, the Ink Analysis tool will be a great boon for instructors trying to represent diagrams to the visually impaired student. Bennett and associates call this process image-argument synergy. They define it as the attempt to "tie the experience (the image) to the description (the argument), synthesizing both levels of the academic discourse, and ... offering an insight [to the students] the way the teacher thinks about the task. The true challenge therefore, of designing universally accessible teaching materials, is to both preserve the immediacy of the academic discourse while facilitating its digitization into accessible media." [1] The InkAnalysis tool gives this ability to instructors by allowing them to create diagrams on a tablet PC, have it project overhead for other students, and provide a means for the non-visual student to access the diagram while it is being constructed. This paper explores the possibility of using a modified version of InkAnalysis for non-visual diagram navigation, the changes needed to make the program accessible to a non-sighted user, and future implementation possibilities of this tool.

2. BACKGROUND RESEARCH

The most popular way to depict a diagram to a non-sighted person is by using a tactile printout of the diagram. Tactile diagrams, in the form of special paper with raised lines, is the most prominent way of depicting diagrams for the blind; this type of printout is available in many accessibility related libraries, and is used quite regularly for teaching the non-visual student. [7] As a result, the tactile printout diagram is here to stay for the foreseeable future, but it is not the only type of tactile representation available to the blind. Tactile diagrams can be constructed out of quite a variety of materials depending on the creativity of the creator. [4] Aside from Braille printouts and tactile screen displays, objects like string, toothpicks, glue, or cardboard can be put together to represent a diagram. Tactile representations of diagrams have some rather serious drawbacks in both readability and cost. Brown and associates went on to say that "in a tactile

representation text is represented as Braille. But it is hard to distinguish Braille text from pure graphic elements. Another disadvantage is the cost. A full screen tactile display is expensive. A special printer may be less expensive, but each printout costs money and time”. [3]

New technologies have come to the forefront in diagram rendering. One of these new technologies is audio tablets, which work alongside the Braille printouts. With this tool a touch-sensitive PC stores the data for the diagram, and, when the Braille printout is placed on the screen, the non-visual user can touch parts of the diagram to get auditory feedback. [3] The type of diagram rendering here is quite a good step up from the traditional tactile diagrams, because it allows for voice feedback of the information as well as the tactile information. One example of this, as explained by Brown and associates, is “AudioGraf, a diagram-reader which represents simple diagrams in an audio-tactile way. This way blind users can read and interpret simple diagrams without the help of sighted persons”. [3]

There are several other technologies that help to make diagrams accessible to the non-visual user. One such technology, developed by Blenkhom and Evans [2], is explained by Yu and Brewster as: “audio-tactile representation for data flow diagrams. They transform the data flow diagram into an audio-tactile matrix”. [8] A third tool, described by Brown and associates, is “Kekul’e – a tool developed to enable audio browsing of molecular structures”. [3] The number of accessibility related tools out there are quite numerous, and we do not take time here to introduce them all. All of these tools use either tactile representations, or auditory feedback, to give information about the diagram to the non-sighted user.

3. THE TOOL

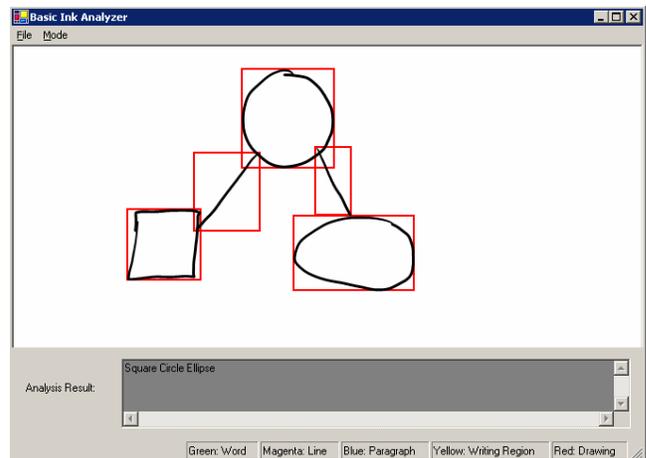
Ink Analysis is a product of Microsoft, and is a tool that can be used with a Tablet PC. The tool does not currently have the features that allow it to be navigated by a blind individual, but we will show the feasibility of adding those features to this product. Ink Analysis is a diagram recognition tool; it recognizes that a square is drawn with the pen, then a line is drawn to where a circle is drawn, and then a line is drawn to where a triangle is drawn. The tool will recognize that a square was drawn, a circle was drawn, a triangle was drawn, that the square is connected to the circle, and that the circle is connected to the triangle. The tool holds all of the data for a diagram in a tree that it creates in the background. By retrieving data from the tree we are able to make the InkAnalysis tool accessible for a non-visual user. Once we retrieve the required data from the tree, we can output that data in such a way that the user can access it with a screen reader.

The tree, that InkAnalysis creates, holds a large variety of data, which includes a lot more than just the shapes and lines drawn in figure 1; it includes many other types of nodes and links, but we are not interested in these for the purposes of this paper. To access the data we need, we added in some keyboard events that will run when a certain key is pressed. For example, if a user presses the ‘L’ key the next link to the left will be displayed to the screen. By pressing a combination of the ‘L’ key, ‘R’ key, and ‘N’ keys the user will be able to navigate the entire diagram. The ‘R’ key will allow the user to select the next most right link from a certain object, and the ‘N’ key will take him to that ‘next’ object. So, for example, if the current node was on the circle in figure 1, then the ‘L’ key would say “Link to square,” the ‘R’ key

would say “Link to ellipse,” and the ‘N’ key would take you along which ever of these two links is currently selected.

The tree holds several types of nodes that can be accessed through method calls. For the purpose of this paper we are only interested in the InkDrawingNodes, which are the actual shapes and lines drawn in the diagram. The InkDrawingNodes, and all other types of nodes, have links associated with them; these links depict how each node is connected to each other node. For example, a line is always the source of a square or any other geometric shape. So, again looking at Figure 1, the line between the square and the circle is the source for these two shapes, and the two shapes are the destination for the line. The idea of the line being the source for each of the nodes, it is connected to, is not very intuitive, because most people would think that one of the nodes would be the source for the line. It is worth noting that each line has 2 links, and each shape has a number of links equal to the number of lines attached to it.

Figure 1:



A diagram of a tree having a root node (circle) with 2 child nodes (square and ellipse).

4. ADDITIONS TO CODE

Several changes to the code needed to be made to allow InkAnalysis to be accessible to the non-visual user. The first of these was registering that a keyboard event happened. To do this we linked the keyboard event to the output region of the form. This allowed for the associated KeyDown event method to be called whenever a key was pressed. The next addition we made was to add the KeyDown method to the functions associated with the form and place the logic for when each of the keys were pressed. To achieve this we added the method to the file, and then we added checks for when each of the keys were pressed; this allowed us to be able to disregard any other keys that might be pressed. Next we added in the logic for getting the requested node from the tree. We decided that all we were interested in were DrawingNodes, so we were able to call a method that would return all of the DrawingNodes in the diagram as an array. With this array of nodes, we are able to increment an index to retrieve each node along a given link. Here we would get the current index, which would give us the current node, and then we would be able to access the links associated with that node. The links associated with a node are held in an array, and we access these links through an index. Once we have this array of links we can,

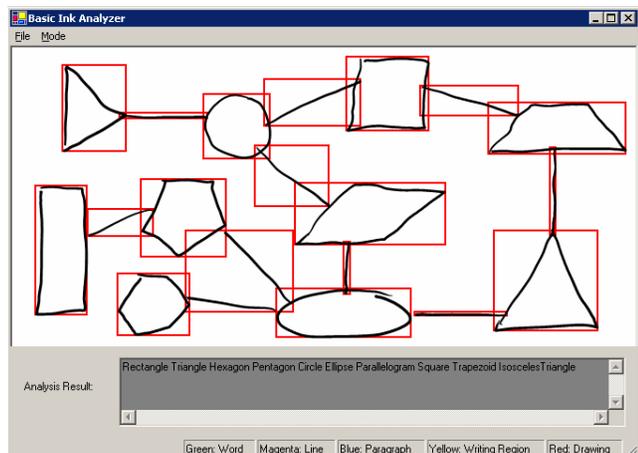
by using the 'R' and 'L' keys, transition between all of the nodes attached to the current node. Finally we output the current requested node to the textbox, that Ink Analysis uses for displaying the recognized name of the objects drawn. The output text box is accessible to a screen reader, so the non-visual user is able to read the node information.

5. USING THE TOOL

Figure 1 can be easily traversed with the modified Ink Analysis tool by simply hitting the 'L' key to find out what links are available, and then hitting next, the 'N' key, to get to the corresponding node. Once again the user can hit the 'L' and 'R' keys to find the link they wish to traverse, and then hitting the 'N' key to get there. The tool will show, depending on which node is the current one, that there are links to a square, an ellipse, and a circle. It will also show that the square has 2 nodes attached to it, and that the other two shapes only have 1.

Navigating the diagram is completely accessible to a non-visual user by having only keyboard commands. We made sure that all of the functionality of our modifications were completely accessible through keyboard commands. We also tried to make the commands intuitive from a user's perspective, thus we have 'L' for moving left one link, 'R' for moving right one link, and 'N' for going to the next link. We should also note that we added 'S' for saving the current diagram and 'O' for opening an existing diagram. These functions are necessary if a student wants to look at a diagram at a later date without it having to be drawn again.

Figure 2:



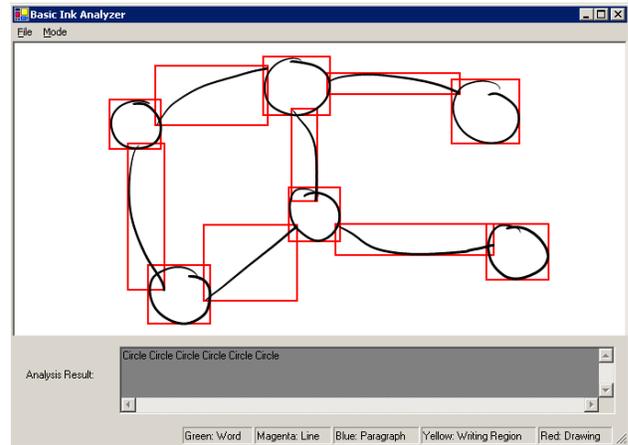
A more complex diagram with many shapes.

In figure 2 we have a much more complicated diagram; this graph has several nodes that are connected to many other nodes. To traverse the diagram above a non-sighted user has to use several steps to figure out which nodes are attached to which, and to tell how big the tree is. To traverse this tree the user will have to use the 'L' and 'R' keys to go through all of the possible links for a given node, and use the 'N' key to go to that node. Eventually the user will go to every node in this manner, and will know how everything in the graph is connected. There still will be no problem in traversing the diagram, because each node is a different shape; this allows the non-visual user to know that there is only 1 circle, so that circle will be the same one no matter how many times it is traversed. Having a shape restriction is not at all

realistic from a graph perspective, because a graph usually has either circles or squares as nodes. We will discuss ways around this restriction in the next section, along with some other future considerations.

6. FUTURE WORK

Figure 3:



A graph with several circles that are all interconnected.

A diagram like figure 3 needs to have some type of identifier for each of the objects. In order for a non-visual user to navigate the diagram they need to be able to tell the nodes apart from one another; it doesn't do much good to keep outputting "square" or "circle" when that is all that the diagram holds. To fix this problem an identifier could be assigned to each node, and, in this fashion, the user would be able to use this value to determine if they have traversed the node before. We are currently not adding functionality to the Ink Analysis tool to handle this type of diagram; we are leaving this for future work.

Other future problems, that should be solved, are to include a self-speaking interface and the idea of a node having a location. A non-visual user will probably have a speech synthesizer, but a self-speaking unit is easier to integrate into a system without such a resource. Also, by having a self-speaking interface, the InkAnalysis tool can define exactly what output the user might want, and it wouldn't be restricted by only being able to show what is in the output textbox. The idea of the location of a node is important, because currently the Ink Analysis tool does not necessarily start at the "root" of a tree; it starts at the left-most highest node in the graph. By having the location of the node we could make sure that the top most node is always sent back, but this has drawbacks in that it will not work for a sideways drawn tree. There is quite a bit of research remaining on how to find the exact root of a drawn tree diagram.

7. CONCLUSION

This paper has summarized the feasibility of using the Ink Analysis tool as a diagram browser for non-visual users. Stand alone Ink Analysis from Microsoft is a diagram recognition utility that will recognize the different parts of a diagram as it is drawn; this does not allow a non-visual user to access the drawing, and by no means can that individual draw a diagram. With a few code additions the tool can be made to allow a non-visual user to navigate a diagram someone else had drawn. A non-visual user

can navigate a diagram by using a combination of the 'L', 'R', and 'N' keys to traverse the different parts of the diagram.

There are several uses of the modified version of the Ink Analysis tool. The tool has implications on both a student's and an instructor's academic life. For a student, the modified Ink Analysis tool gives the ability to navigate diagrams to get a better understanding of how everything works together. A student might take a saved drawing, open it up, and browse the diagram at their leisure. For the instructor, the modified Ink Analysis tool gives the ability of getting the non-visual student access to in-class diagrams in real time. The instructor could open the modified Ink Analysis tool, start drawing a diagram, and the student could browse the diagram as it is being created; this would require having the student having access to the same tool as the instructor.

8. ACKNOWLEDGEMENTS

Special thanks to Herry Sutanto at Microsoft for his help in setting things up and providing assistance during the implementation phase. Also thanks to Dr. Francioni for her guidance on this project.

9. REFERENCES

- [1] Bennett, S., Hewitt, J., Kraithman, D. & Britton, C. 2003, 'Making Chalk and Talk Accessible', Proceedings of the 2003 conference on Universal usability (Vancouver, Canada), ACM Press, New York.
- [2] Blenkhorn P., Evans D. G., Using speech and touch to enable blind people to access schematic diagrams. *Journal of Network and Computer Applications*, 1998. 21: p. 17-29.
- [3] Brown, Andy; Pettifer, Steve; Stevens, Robert: Evaluation of a NonVisual Molecule Browser. ASSETS 2004 Conference.
- [4] Edman, P.K., Tactile graphics. 1991: American Foundation for the Blind.
- [5] Francioni, J.M. and A.C. Smith (2002). Computer Science Accessibility for Students with Visual Disabilities. Proceedings of 33rd SIGCSE Technical Symposium on Computer Science Education, Northern Kentucky, February 2002, pp. 91-95.
- [6] Kennel, Andrea R.: AudioGraf: A Diagram-Reader for the Blind. In: ASSETS 96: The Second Annual ACM Conference on Assistive Technologies. The Association for Computing Machinery, 1996, pp 51-56.
- [7] Ramloll, R. and Brewster, S.A. (2002). A Generic Approach for Augmenting Tactile Diagrams with Spatial Non-Speech Sounds, Proceedings of ACM CHI2002 (Minneapolis, MN), Vol. II, ACM Press, pp 770-771.
- [8] Wai Yu, Stephen Brewster, Multimodal virtual reality versus printed medium in visualization for blind people, Proceedings of the fifth international ACM conference on Assistive technologies, July 08-10, 2002, Edinburgh, Scotland