1 Introduction

According to the World Health Organization [1], there are about 285 million people with vision impairment, of which 39 million are estimated to be totally blind. Visual impairment is defined as a severe degradation in sight capability, making the individual incapable of doing some regular tasks. Advancement of technology has been improving the quality of the life of the people. A part of these improvements is related to Visually Impaired (VI) people, enabling them to perform tasks that they were previously unable to do. One important activity is education and accessing related materials.

The focus of this report is the technology that enables access to science documents for VI through sensory substitution. Enabling access becomes complicated whenever the improvement of vision, such as magnification or changing the color contrast, is not sufficient, and consequently access through other sensory capabilities comes to play. Hearing and the tactile sense are the two main sensory inputs, which can serve to compensate for the vision loss.

Screen reading is the central and unavoidable technology that bridges between the features of documents to be accessed and converter tools (those that convert the features to the desired sensory outputs). A screen reader, in its most idealistic form, would emulate a human reader, who can communicate and comprehend what VI users need to access, and conveniently describe it to them. Having this idealistic goal in mind, we survey the research areas and contributions that can help to approach and realize it in this report. The main focus of this report is on the accessible output feedback to a VI, rather than the input commands by him. That’s because the sense of vision is concerned with capturing the visual aspect of the outer world. However, investigating the ways of receiving command from a VI can also help to provide better interaction, that is, to enable navigating and querying the output more conveniently.

In section 2, we introduce the mainstream access technology for VI, categorized based on the substituted sensory input. Before that, the screen reading technology is introduced in more detail. Section 3 will look into the scientific knowledge and the formats in which the knowledge is recorded. There, the entities appearing in the science documents are categorized to text and non-text graphics based on the approach we may exploit to make them accessible to a VI. That section will specifically try to associate the text and the non-text entities to the access technology introduced in section 2, i.e., how each category can be accessed by a VI using the technology; what is already addressed by the technology, and what is not; what are the open research problems to be addressed before the technology can adopt the solutions. Image understanding, designing efficient and widely acceptable object models to save and process the entities appearing in the science documents, as well as designing convenient user interfaces to provide the decent access for VI are the main areas of research for the access to science problem.

Section 4 tries to focus more on the problem of image understanding that is brought up for tackling access to graphics. Section 5 discusses the issue of “prevention better than cure” for the access to science problem, that is, to generate the documents that are already accessible with the current technology. Finally, section 6 wraps up and concludes the report.
2 Access Technology for VI
In this section, we are going to introduce the access technologies for VI. As mentioned above, there are two main avenues of technology with this regard, i.e., those that improve access via tactile sense and those that exploit the hearing sense. However, before anything can be made accessible through touch or hearing, a context aware facility is required in order to interactively capture those entities from the context. The most notable instance of such a facility on the computing devices are the screen readers. A screen reader is a platform-dependent software that interacts with user and attempts to identify and interpret what is being displayed on the screen (or, more accurately, sent to standard output). This interpretation is then re-presented to the user according to the desirable way of interaction. In the other sense, the screen reader is an input/output software, which can enable a VI user to navigate and use the device by customizing the input commands and output feedback to what a VI user can access. The better and more intelligible a screen reader can interact with the user and the smarter it can interpret the displayed output, the more convenient the user access experience would be. Examples of the screen reader technology are FreedomScientific JAWS (commercial), and NVDA (open source) for Windows, as well as Voice Over for the Apple Mac products. Talks for the Symbian cell phones, Talk Back for Android devices, and Orca for the Linux platform are other examples.

2.1 Tactile Access to Output
Invented by Louis Braille in 1824, the Braille writing system is a form of tactile communication, which has enabled VIs to write and read without sight. In its most common form, each character in the Braille writing system is represented by a matrix of dots with 3 rows by 2 columns. Each of the dots can be raised or flat. The size of the characters and their representation have been designed to be comprehensible by a single touch of a proficient Braille reader. All together, the 6 dots of the Braille can represent 64 characters that is sufficient for the alphabet of many languages. There are 8-dot Braille writing systems to represent more than 64 characters as well. Besides, any larger writing system can be set to utilize more than one Braille cell to represent each of the characters in the system. For instance, the standard Braille encoding uses an escape characters when it is going to represent a number, and the 1-9 and 0 digits are the same as the first 10 letters of alphabet. Many other special-purpose Braille encoding has been developed and standardized. A famous and widely-used example is the Nemeth Braille code for mathematics [2], that uses the standard 6-dot Braille to encode mathematical and scientific notations.

Writing Braille has originally been done with slate and stylus, which corresponds to paper and pencil for a sighted person. Perkins Braille [3] was developed in 1951 by David Abraham, who was a teacher at the Perkins School for the Blind. The machine has 6 keys for each of the dots in the Braille code. The rest of the machine is very similar to a regular typewriter; it has a space key, a backspace key, as well as a line break key. It has two side knobs to advance paper through the machine and a carriage return lever above the keys. Like the print and publication system, embossing Braille has also been modernized by the invention of the computers. Braille codes can be saved in computer, and be embossed on paper by a Braille embosser, which acts very similar to a printer. The most famous of the Braille embossers are Tiger Braille embossers [4]. In addition to embossing Braille, they can emboss graphics and/or combine ink print with the embossed features.

Refreshable Braille display [5] is another sample of modern technology for rendering Braille. It is an electro-mechanical device capable of rendering the Braille characters. Connected to a computing device, a Braille display can represent textual information (a tactile counterpart for a monitor). Since the text to be displayed can be located in any arbitrary place on the screen, as noted above, a screen reader is required to intercept the text and provide it to the Braille display. Since the production cost of the Braille display is too high, they usually present not more than a single row of 40 or 80 cells. If, however, they are worth to
have multiple rows, they can be utilized to render simple graphics as well. Designs for a full braille computer monitor have been patented but not yet produced. [6].

In order to decrease the production cost of Braille displays, a rotating-wheel braille display was developed in 2000 by the National Institute of Standards and Technology (NIST) [7]. The wheel is still in the process of commercialization. In this unit, braille dots are put on the edge of a spinning wheel, which allows the user to read continuously with a stationary finger while the wheel spins at a selected speed.

Apart from the Braille, which is a representation of text, haptic technology [8] has also been used to aid VI. Haptic technology takes advantage of the sense of touch by exposing forces, vibrations, or motions to the user. This mechanical stimulation can be used to represent the objects in a virtual manner. This can be seen as the touch counterpart for graphical representation. Although Braille is also a type of haptic feedback, haptic technology mainly refers to the coarser scale form of feedback, which is not suited to represent detailed entities such as text.

The Geomagic Touch [9] is an example of a commercial haptic device. It is a motorized device that applies force feedback on the user’s hand, allowing the user to feel virtual objects and producing touch sensations as the user manipulates on-screen 3D objects. Applying haptic technology for VI accessibility has been studied to some extent. The authors in [10] reported that the technology was shown to be effective for rendering textures and simple 3D objects both for sighted (blindfolded) and VIs.

2.2 Auditory Access to Output

Auditory access is divided into two main types: speech and non-speech sound (the latter is referred to as sonification). For the automated speech generation out of text, there have been several commercial and non-commercial products that generate a sound very similar to a human voice. Examples include FreedomScientific Eloquence and Espeak. Automated speech generation is technically referred to as speech synthesis or Text-To-Speech (TTS), and the software performing the task is called speech synthesizer or TTS engine. However, since the text to be spoken is not readily available, a screen reader is required to bridge between the text to be spoken and the TTS engine.

Sonification [11] is the use of non-speech sound to convey information. It has also been utilized to make graphics more accessible. Some work on producing sonic patterns for an image exists. For instance, in the vOICe project [12], it has been tried to make a sonification of any arbitrary image, whether saved on the disk or being captured by a regular camera or webcam on the fly. The image is scanned from left to right in one second as the sound beeps are being produced. Change in pitch represents change in elevation, and change in the loudness represents brightness. The more complex the image, the more complex the soundscape would be. Although this might seem to be a little useless sonic noise, practicing to get information about the image out of the sound may yield useful applications, especially for people who are totally blind. According to a selection of user testimonials, after getting used to the technology, listening to the soundscape is becoming like a rough natural colorless vision sense, which has enabled them to live more independently and do tasks that they were previously unable to do [12].

In other similar work [13, 14], 2D mathematical curves are represented by sonification. The X-axis is represented by time passage while the Y-axis is represented by change in frequency and amplitude. So, the soundscape is a succession of beats that change in frequency and amplitude over time. After some training some VIs can distinguish certain curves [15].

Sonification of images seems to have severe limitations because there is high parallelism in vision, while the hearing sense is much more sequential. The sense of hearing does have some degree of parallelism. People can learn to recognize pitch, amplitude, rhythm, and where the sound is coming from somewhat
accurately, but integrating those dimensions to convey an image in detail seems beyond human capability [16]. Research has shown that there are benefits to sonification for conveying many kinds of information [17], however, its specialization to auditory substitution does not seem to be adequate to replace vision.

3 Science and Education: Access to Materials for Visually Impaired
Throughout the past centuries, scientific knowledge was recorded in the shape of hardcopy materials, e.g., books, magazines, journals. During the recent decades – with the aid of computer technology – the trend of recording science has shifted to mainly digital assets; although there are still many things produced and kept in the format of hardcopy, the digital versions of the same materials are available as well. Even if the digital version is not available, digitizing the hardcopy materials has been enabled through the scanner devices and the Optical Character Recognition (OCR) technology.

As mentioned above, access to computer and digital materials for VI people has been enabled through the screen reader software, the TTS engines, and the Braille displays. We can define the access to science problem as follows: given a scientific document, we want to make it accessible by a VI. The pipeline of the state-of-the-art technology to resolve that is as follows:

1. If the document is hardcopy, scan and convert it to a digital format.
2. If the digital file contains only the scanned images, run it through an OCR engine to convert it to the text.
3. The digital text is intercepted by a screen reader.
4. The screen reader passes the text to a TTS and/or a Braille display interface.
5. The TTS engine/Braille display renders the equivalent speech/Braille characters that can be accessed by a VI.

The above pipeline only works for textual documents, or only the textual portions of the document will be available. The procedure becomes complicated when non-negligible portions of the document are not text, or the text is placed in a non-trivial format, e.g., multiple columns.

For the above pipeline to be applicable to a wider range of problems, we need to look into the scientific documents in more detail. From the perspective of enabling access, there are two types of information presented within an English-language scientific document: ASCII text and non-ASCII graphics. Any other type of information in ASCII which is not a natural language text, such as a computer program, can also be classified as ASCII since all the technology mentioned can handle access character-wise as well. For instance, OCR technology recognizes characters; smallest Braille units are characters; TTS engines can vocalize a single character. However, before any of the information types can be made more accessible, each should be localized and labeled with the correct type through document layout analysis. After it, each piece of the information could be passed through a corresponding pipeline to be made accessible.

3.1 Document Layout Analysis
Document layout analysis can be defined as follows: given a document image, we’d like to find out the sequence of information blocks, e.g., the text blocks, the mathematical notions and formulas, and the graphics, in addition to their coordinates within the page. Although the information flow is usually serial from left to right and from top to bottom within each page, there are many scientific documents with two or more columns, which make this problem challenging. Document layout analysis is divided into, firstly, segmentation of the physical structure, and then deducing the logical structure [18].

Segmentation of the physical structure of the document image is usually preceded by noise reduction, background separation, and skew detection and correction in order to neutralize the noise and other
artifacts introduced during the document generation and scanning process. The physical layout segmentation algorithms are primarily classified based on their order of processing into top-down approaches and bottom-up approaches.

Top-down approaches start with a complete document image and repeatedly split into smaller homogeneous regions. The splitting stops when all the resulting regions correspond to the primitives that describe the document. The primitives could be pixels, connected components, words, etc., depending on the application. As an example, The X-Y Cut algorithm [19], starts dividing a document image into sections based on valleys in their projection profiles on the X and Y axes. The algorithm repeatedly partitions the document by alternately projecting the regions of the current segmentation on the horizontal and vertical axes. The splitting stops when it reaches a particular criterion that specifies the atomicity of a region.

Conversely, bottom-up approaches start with the primitives and repeatedly group them into larger regions as words, lines, text-blocks, columns, etc. The Docstrum algorithm [20] and the Voronoi diagram based algorithm [21] are examples of effective bottom-up approaches that function by grouping the connected components in a page.

Hybrid algorithms combine the above two approaches. The authors in [22] proposed a hybrid algorithm using a split-and-merge strategy. There, a top-down approach is used to arrive at an over-segmented document image. Regions that are similar and nearby are then merged to form homogeneous regions.

The logical structure of a document image is defined as a mapping from the physical regions in the documents to their logical labels. The labels include text blocks, images, diagrams, etc. Even though the logical layout analysis process is defined to follow the layout segmentation, in practice, the two processes could be combined into a single document understanding framework. One of the popular approaches to define logical structure of a document is to treat the set of regions in a document as a string of symbols. A grammar is defined over these symbols that describes the logical structure of an arbitrary document under consideration. The process of structure analysis then computes the most likely parse (sequence of grammar rules) that generates the observed string of symbols. For an extensive study of document layout analysis approaches, including layout segmentation and logical layout extraction, the reader may refer to [18].

3.2 Accessing Text
Being the dominant type of information within scientific documents, text and accessing it has been well researched. The extreme instance of this problem is when only an image is available. For this case, given that the image quality is sufficiently good (reasonable resolution with minimal noise and skew), there are off-the-shelf OCR tools, such as the Tesseract open source OCR engine, that can extract text out of the image with a high confidence. As mentioned earlier, accessing text has already been enabled with the aid of Braille, Braille displays, Braille embossers, as well as the TTS engines and the screen readers. Although there is always space for improvement, this problem can be considered as solved for the texts (especially the English language).

3.3 Accessing Non-Text Elements
Given that all non-text elements have already been located in the document through the layout analysis phase, converting them to VI-accessible formats is by itself a wide problem. First of all, there are various types of graphics within a document, each of which requires its own processing pipeline to be made accessible. Secondly, unlike text that does have standard accessible formats both in tactile and in auditory formats, devising an object model for each type of non-text elements requires consensus among the
creators. Once a consensus is reached, devising a scheme to render the object models to the audience in an accessible manner requires usability review [23] to prove its efficacy. Non-ASCII mathematical symbols and formulas, diagrams and tables are the samples of abstract graphical entities as opposed to the pictures of the real objects and scenes.

As an example of a standardized object model, we can mention the Math Markup Language (MathML) [24], which was developed by World Wide Web Consortium (W3C). MathML is a formal language for expressing mathematics. It is based on XML to be compatible with a large range of software products. MathML can easily be used by any software to render a human understandable form of those mathematics. For instance, Math Player by Design Science can convert the MathML to a natural language speech output (although there is no standard way of verbalizing math).

Modifying the above pipeline to cover science documents with a remarkable amount of non-text elements, the steps would be as follows:

1. If the document is hardcopy, scan it to an image file.
2. Perform layout analysis on the document image to separate text from non-text parts.
3. The text parts should be processed by an OCR engine.
4. Non-text parts should be classified to subtypes, such as chart, math, etc.
5. Each subtype should be processed with the corresponding image understanding module to extract the underlying data.
6. The data should be saved into the corresponding object model.
7. The object model and the text from step 3 is captured by a screen reader.
8. The screen reader would pass the data or the text to the corresponding module, which converts it to the desired output to be accessed by the VI user.

Note that the above pipeline tries to encompass the hardest scenario of the access to science problem, i.e., when the document is only in hardcopy. If the document already has a digital and more high-level structure, such as a PDF document with vectorized diagrams and text, some steps may trivially be skipped.

As covered earlier, steps 1 and 3 are regarded as solved. The related work for step 2, that is, document layout analysis was introduced in section 3.1. Related work to the Steps 4 and 5, which are concerned with image classification and understanding, will be introduced in more detail in section 4. Designing and standardizing document object models is mainly a software engineering task (not covered here). Related work to design and validation of methods - through usability review - to output or render the data in an accessible manner is covered below. The rest of the steps and their requirements are mainly concerned with purely engineering approaches that are not within the scope of this report.

3.3.1 Tactile Output Methods

One approach to outputting non-text data in a VI-accessible manner is to emboss the non-text element. Draftsman tactile drawing board [25] is a tool to create tactile drawing and add Braille annotations to it manually. For automating graphic embossing, the first step is to localize the textual elements within the graphic because they should be converted to Braille. Since the resolution capability of the tactile sense is inferior to that of vision, any tactile graphic needs to be simplified. The authors in [26] utilized machine learning methods to locate the text labels. After an OCR phase, the text is converted to Braille. Finally a simplified version of the graph bearing the Braille labels is embossed.
3.3.2 Sonification Output Methods

Using non-speech sound is another way of making graphics accessible. As mentioned above the vOICe project has tried to make a sonic pattern out of any arbitrary image, such as a diagram or an image within a document. As mentioned earlier, some work has used changing frequency in the output sound to represents a single curve in a 2D space, and this design scheme has turned out to be easier for the users to pick up. However, a bare sound pattern is analogous to a curve on a white paper without any annotation. Adding context to the sound pattern has therefore remained an open research question. Sonification, although very useful for improving access, has turned out not to be an effective enabling method when used alone. In the other words, humans are not capable of grasping all the necessary details of a graphic using only sonification. Inserting spoken cues before, between, or after the sonic pattern can be an effective approach to complete what the sonification is trying to convey. Spearcons [27] are spoken cues consisting of compressed and sped-up synthesized speech. The speed is rendered at the fastest speed at which it is still intelligible. Spearcons have proven to be more efficient for exploring and working with menu items in software than previous auditory feedbacks, such as earcons [28]. Using spearcons in order to add context to the sonification of 2D diagrams seems to be a promising approach.

3.3.3 Text Output Methods

Analogous to how a human verbally describes a graphic, one approach to make non-text elements accessible is to make a text description out of it, and have it accessed through the text accessibility technology. However, usability review may need to be performed to see whether the text output is concisely representational.

Using natural language to communicate information about numerical data and graphs is a well-established research area [29], with agreed general architectures [30]. For instance, Math Description Engine (MDE) [13] is an accessibility project by NASA, where a text description of 2D mathematical curves on the X-Y plane is generated from the equation of a curve or a data file. There is, however, no report regarding its usability.

iGraph-Lite [31] provides short verbal descriptions of the information depicted in graphs, as well as a way to also interact with this information. The system attempts to make all the information contained in graphs easily accessible to users by means of keyboard commands, so that the user can explore the graph and infer its intended message given his or her own needs. The system gets the graph data out of the supported document object models, even though it can be extended to encompass image understanding modules in order to access images when the object model is not available. The usability review of iGraph-Lite [32] showed with good statistical power that natural language interfaces to graphs and charts are usable by VI people, who could accurately answer complex questions that they could not answer using only a static textual description of the graph. It also showed that the VI community wants to use these technologies and that people feel that technologies such as iGraph-LITE help provide a more convenient access to graphical information.

4 Image Classification and Understanding

Although access to graphics has been researched to some extent, for the most part it is still inaccessible or the access is not sufficiently convenient. For instance, basic graphics can be embossed, even though tactile representation of more complex graphics might be of little use for a VI; sonification of any arbitrary image has been enabled by the vOICe project, but there is no concrete evidence of its usefulness. Accessing 2D mathematical curves has been enabled by the NASA MDE project outputting the result into three different types, i.e., visual, sonic, and text. However, the input domain is limited to well-defined
equations and data files, while there are many accessibility cases where only an image or a hardcopy is available.

Any technology, before turning materials to more accessible formats, tries to extract some level of information from the raw data. For example, the MDE accepts only a set of predefined inputs, i.e., the input is already high-level and well-formed information. It can then produce the output in three different formats, which makes those inputs universally accessible. The graph embossing project should locate the text labels and run them through an OCR phase. It can, however, send the other non-text entities as a low-level set of pixels to the embosser after some filtration. The VOICe tool converts a low-level set of pixels appeared in an image directly to a soundscape, i.e., it does not attempt to extract any high-level information at all. We can see that the higher-level of information the technology has/tries to acquire from the input, the more universally accessible outputs it can create from that, e.g., the outputs of MDE are more universal than those of graph embossing or VOICe. However, preparing a higher-level information from the low-level data is not straightforward, which makes the input domain to be bounded by certain constraints. For instance, the input domain of MDE is smaller than that of the graph embossing, which is far smaller than that of the VOICe tool. This is a trade-off that brings up the unsolved problem of image understanding or computer vision, that is, better access requires better information, which requires better image understanding approaches.

By “unsolved” we mean that given an arbitrary image, understanding what is there in the picture is a hard problem. However, myriads of computer vision research has been performed for when the image is constrained to certain types and properties, and/or the what to grasp from the image is a limited set of features. On the other hand, for access to science problem, we’d like to tackle those non-textual entities that appear in scientific documents. Moreover, what we need to glean out of the images is a set of necessary features in order to be converted to multimodal and universally accessible formats. Therefore, the state-of-the-art computer vision methods can be utilized to get a more high-level understanding of those images appear in science texts in order to bridge the gap of science materials to more universally accessible documents.

Given a set of non-textual entities within a document, the first step is to classify them into well-defined categories, e.g., charts, tables, mathematical notions, real objects, etc. Once those categories are identified, they can be sent to their corresponding pipeline in order to process them and extract the information they are trying to present.

A large body of research and proposed methods for image classification are available within the literature [33-37], which can be utilized for classification of graphical entities appearing in science documents. Specifically, in some work on chart image classification and understanding [38, 39], the researchers tried to extract the underlying tabular data out of the chart image. The majority of existing approaches to chart image classification and understanding were developed within the scenario of image features extraction followed by a feature-based comparison [38]. The latter varies from comparison of a test image to training images (Learning-based approach, such as [37]) to comparison of a test image to abstract models representing particular classes (Model-based approach, such as [40, 41]). For the image features to be used, some proposed work, such as[39], has utilized lower-level features like pixels and square patches of them, while others have used higher-level features, e.g., authors in [40, 41] detect basic shapes in the image and check if they satisfy a certain set of constraints corresponding to a certain chart type.

Given that the charts or other graphical entities are classified to their corresponding types, dealing with each type requires its own procedure. One of the tools that might need to be revisited is the OCR facility, for when graphs contain textual entities. However, performing OCR on text documents is somewhat
different from doing it within a diagram or other graphical entities like natural images. For text document images, these assumptions usually hold: all text lines are parallel; text size is uniform; it is easy to separate the background; big text blocks of several text lines (paragraphs of text) are present. For natural images the following assumptions usually hold: there is a few text strings on an image; text strings are small; background is complicated; text strings have lower contrast compared to the text in document images. Diagram images belong to none of these classes. They are somewhere in between (text strings are small, but have a good contrast). OCR engines perform well on the text document images, even though for other types of images with sparser amount of text, a preprocessing text locating step improves the result remarkably. For instance, the authors in [42] proposed an algorithm for separating text strings in engineering drawings and diagrams. Another similar approach has been proposed in [38] for the chart images, which has reportedly improved the result of OCR by a factor of 15.

As mentioned above, information extraction for each type of graphics can be done through a unique procedure. For instance, information extraction from images of linear and quadratic 2D curves was studied in [43]. The algorithm presented in [37], is capable of detecting special data points and solid line curves together with its axes, labels and data symbols, leading to 2D plot data understanding and information extraction. The proposed approach in [38] to the chart image understanding problem is development of the chart-interpretation kernel, which can adapt to various chart types through the addition of new models. Nevertheless, information extraction out of an arbitrary diagram has remained unsolved for the most part and requires more elaboration and research commitments.

Dealing with non-ASCII mathematical notions and formulas whenever the document object model is not available is another instance of computer vision problem. With this regard, Infty [44] is a practical system, which can convert mathematical document images to other formats, such as MathML. The system can separate mathematics, both inline and within an independent line, from the ordinary text, and simultaneously convert both to the desired formats. The system also has a manual error correction module. The reported experimental result over about 500 pages with mathematical text indicates close performance to the OCR of the ordinary text, that is, 95.4% compared to 99.5%.

5 Designing Accessibility Standards: Prevention better than Cure!

Viewing to the access to science problem from a different perspective, we can specify and validate design schemes for generating scientific documents, where recorded knowledge is by design universally accessible. The most notable example of this kind is W3C efforts and publications for making the webpages universally accessible [45]. For instance, there, it is instructed to provide sufficient text description for any graphic embedded in the documents. As mentioned earlier, development of the XML-based formal math language, namely MathML, is a remarkable instance of the W3C work.

The validated and widely acceptable standards and guidelines can not only serve as a resource for generation of accessible documents, but also can be utilized as the standard object models for saving and processing the entities in question. For example, Infty system, after recognition of the mathematical notions can convert them to MathML. The output MathML can easily be accessed by other software, such as Math Player, or a screen reader, to be spoken to a VI user for instance.

Nevertheless, designing and validating such document generation schemes is by itself a big and largely unsolved challenge. First of all, any design scheme should be validated through an empirical usability review procedure. Secondly, once we reach a validated design, how to incorporate it to the current document generation technologies and products is another challenge. Thirdly, how to encourage the users of the document generation technology who might not have any insight into blindness to go through the hassles of using those guidelines and facility to create accessible materials is another issue.
Although defining efficient design schemes for document generation and trying to standardize them will definitely improve access to science, it may never totally alleviate the problem. Apart from the mentioned reasons for difficulty of reaching and enforcing a widely acceptable standard, the legacy problem and the disparity between the non-disabled document producer and the VI user will persist. The legacy problem refers to the cases in which the documents were generated before standard generation schemes were defined and accepted. The disparity between the users is referred to those usual cases where the person generating the document is not disabled and overlooks the fact that the audience might be a VI. Furthermore, some cases of accessibility problem are inherent, e.g., a picture of a real scene or a real object is inherently inaccessible to a VI.

6 Conclusions

This report has focused on the access technology adopted to access science documents for VI through sensory substitution. Screen reading technology has been recognized as the central technology that bridges between the VI and the documents. A screen reader interacts with the user, and tries to identify and interpret the user-desired entities to be accessed. It finally delivers those interpretations to the user in a form that he or she can conveniently access.

Those entities to be identified and interpreted by the screen reader are classified as text and non-text graphics. The basic type of alternative sensory outputs that can be used to deliver the interpretation to VI are tactile and the hearing sense. Coarse tactile feedback, which is convenient to render 3D objects and surfaces, is referred to as haptic technology. Feedback targeted to the hearing sense is also classified as speech and non-speech (sonification). Accessing text (or an image of text) and outputting it as Braille or speech has been a well-studied problem, and practical and commercial systems has been developed.

Tackling the accessibility of non-text graphics, three categories of research problems have been recognized: document image understanding when only an image is available; devising efficient and widely-acceptable object models to encapsulate the information from the documents/images; and devising a scheme to render the object models to the target audience (VI here), which requires usability review to prove its efficacy. The solutions reached out of any of the three research areas can then gradually be incorporated into the technology of screen reading with idealistic goal of making screen readers as smart as a human reader. Although substantial amount of related work in the three areas has been accomplished, more research commitments and interdisciplinary collaborations still seems to be required to approach the idealistic goal.

Furthermore, the current screen reading technology is mainly concerned with capturing plain text interactively and passing it to a speech synthesizer or a Braille display. Sonification has also been minimally used to distinguish the context, e.g., when the cursor lands on an edit field in a webpage, a typewriter sound is played. Nevertheless, although the related research is still ongoing, much of the useful research accomplishments, such as processing non-ASCII math or basic charts, or providing haptic feedback, have not been incorporated into the technology yet.

References


---

1 A research topic that has not been discussed here deals with smarter ways of input and interaction with screen readers beyond using keyboard and mouse, e.g., speech input, which can yield smarter and more convenient screen reading technology as well.


38. Mishchenko, A. and N. Vassilieva. Chart image understanding and numerical data extraction. in Digital Information Management (ICDIM), 2011 Sixth International Conference on. 2011. IEEE.


