TAbiWord
Digital Document Annotation and Reflow

University of Maryland at College Park
Department of Computer Science
Undergraduate Honors Thesis

Kevin Conroy
(kmconroy@cs.umd.edu)

March 29, 2004

Advisor: François Guimbretière

Table of Contents

Abstract

Proofreading is a common task that many word processor users perform. Despite advances in word processing programs, many users print documents to paper, annotate them using pen or pencil, and then manually reenter corrections and annotations in the digital source as appropriate. New systems such as the Tablet PC provide a natural interface for document annotation; however, current word processors offer limited support for freeform annotation using a stylus or WACOM tablet. This paper introduces an extension of the AbiWord word processor named TAbiWord, which supports digital document annotation and editing. TAbiWord supports reflow of digital annotations as a document's content, structure, or layout changes. The system allows users to make markings or annotations through the Tablet PC interface or on paper using a Logitech io pen with the PADD system. Our goal is to provide users with a familiar interface for annotating and editing documents on a digital medium, as well as to expand existing paper interfaces, while preserving the flexibility of working with a digital document. TAbiWord aims to bridge the gap that exists between paper and computers throughout the writing process.

Introduction

Recent advancements in word processing programs have helped to make the writing process easier. Word processing applications such as Microsoft Word[1], Open Office[2], and AbiWord[3], allow users to create and change content very easily. With the aid of "print preview" layouts, users are able to see what their documents will look like when printed while they are writing. In spite of these improvements in computer technology, many word processing applications have failed to provide users with a natural interface for one of the most important steps in the writing process: proofreading.

For many users, proofreading a document typically involves printing the document and making annotations and proofreading marks with a red pen. Although this process provides users with a robust interface for document annotation, it also creates a physical separation between the annotations and the digital source. Consequently, once a user has completed the proofreading process, they are forced to type changes and corrections manually from the annotated paper copy back to the digital source. This can be a time-consuming and sometimes frustrating process depending on the size of the document and the number of annotations and corrections that must be copied.

This paper introduces an expansion of the AbiWord[3] word processor for the Tablet PC[10] named TAbiWord. Using the pen-based interface of the Tablet PC, TAbiWord users are able to make freeform digital annotations and markings directly on the screen. The goal of TAbiWord is to simulate and augment the user experience of paper-based proofreading within the word processor while retaining the flexibility of a digital document.

Digital annotation is not a trivial task. Unlike paper-based markings, which remain static and fixed on the page, digital markings must be repositioned as the document's content, layout, or structure changes. In order for repositioning, or "reflow," to be achieved, it is necessary to determine which parts of the document correspond to a given annotation. Once determined, the annotation must be stored in a manner which captures this relationship between digital ink and the document so that the ink may be repositioned correctly as the document changes. During reflow, the text associated with a large annotation may begin to span several lines or pages. In order to preserve the semantic meaning of the marking, it is necessary to split the larger annotation into several smaller annotations which can be each be placed on a separate line or page.

Previous Work

Several systems have explored the issues of digital document annotation. The XLibris system[4], created at Xerox FXPAL, introduced a robust system for digitally annotating a document while reading. Introduced as an "active reading" system[5], XLibris is a tablet computer that displays a document as if it were on paper. Using a stylus or digital pen, users make digital annotations which are repositioned as the document's format changes.

The system supports explicit selection, textual annotation, and freeform digital ink. XLibris also splits large strokes into smaller ones that are associated with a single word or block of text, as well as repositions annotations when the formatting or layout changes. Although the XLibris system addresses many fundamental challenges associated with digital markup, the system was not interested in exploring digital annotation in documents with dynamic content; that is, documents whose text will change as the user continues the writing process. Our system aims to reproduce many of the features of XLibris' "active reading" system in the context of dynamic document content.

To support digital annotation, a system must determine what text is associated with a given marking. This association must be saved so that it may be recovered at a later point. Phelps and Wilensky[6] introduced a framework for anchoring annotations within a document by using a combination of textual context as well as a unique identifier. Although their framework provided a robust system for anchoring information inside of a document, it did not explore how these anchors can be used to maintain the semantic meaning of annotations as the document's content or layout changes. Our system uses unique identifiers, as proposed by Phelps and Wilensky, to anchor our annotations within the document while addressing the issues associated with repositioning annotations.

One of the most important capabilities of a system which supports freeform digital ink annotations in the context of dynamic content is the ability to preserve the meaning of annotations during repositioning or reflow. Brush et. al[7] recently conducted user studies to determine how users expect digital reflow in a document with dynamic content to work. Based on these studies, Bageron and Moscovich[8] introduced Callisto, a plug-in for Internet Explorer that allows users to make freeform annotations on Web pages. Although the Callisto framework addresses many of the problems with digital ink reflow, such as repositioning and anchoring, it does not address the challenges of digital reflow in a document with dynamic content, nor does it focus on the proofreading process.

Annotations can be viewed conceptually as an additional dimension or layer of the document. This dimension is best conceptualized as a marked sheet of overhead transparency that is placed over a paper document; the text on the paper is visible, except in the locations where opaque markings on the transparency block the text. Phelps and Wilensky[9] generalized this principle by creating the notion of multivalent documents which can contain many layers with many different types of information. Our system uses the idea of a multivalent layer of ink that is placed over the document to simulate the experience of paper-based annotations.

Microsoft Word 2003[1] provides an interface for making digital ink annotations on the Tablet PC within a document. These annotations, however, only reposition vertically, causing many markings to change or lose meaning as the document's content or layout changes. Our system attempts to preserve the semantic meaning of annotations by reposting ink both vertically and horizontally.

TAbiWord

TAbiWord is an extension of AbiWord[3], an open source word processing program that is similar to Microsoft Word[1]. TAbiWord adds support for digital annotation and reflow by modifying several key aspects of AbiWord's 400,000+ line rendering, layout, and document storage framework.

Through TAbiWord, we have adopted and expanded techniques used in previous systems, such as XLibris[4], to enable annotation repositioning in documents with dynamic content.

Annotation Interfaces

TAbiWord provides users with several methods for annotating documents. Users may make digital markings with their mouse, a WACOM tablet, or the stylus on a Tablet PC[10]. When TAbiWord starts, annotation mode is enabled by default. Users may type on the given blank document or open an existing document and make changes to it. As the user moves his or her mouse or stylus over the screen, the cursor is replaced with a representative icon of the chosen writing utility. When the user presses down with the stylus or clicks-and-drags the mouse, a mark is made. This is a very natural interface for the stylus, since it mimics the manner in which paper-based annotations are made; when the pen is in contact with the screen or paper, marks are made on the document.

Users may switch between the annotation cursor and the normal mouse cursor by selecting an enable option from a drop down menu.

TAbiWord has also been extended to work with the Paper-Augmented Digital Documents (PADD)[11] infrastructure. The motivation behind PADD is to allow for the cohabitation of paper and computers by having users make markings on paper with a Logitech io digital pen[12]. The io pen is a ballpoint pen with a small camera on its tip that is capable of recognizing and recording the location of markings made on Anoto paper[13]. Once the io pen is plugged into the computer, the PADD system transfers the markings made by the user back to the digital source.

By integration with PADD, users are able to type documents in TAbiWord, print them, and annotate the paper documents with a Logitech io digital pen. When the pen is plugged into the computer, users may open the digital copy in TAbiWord and import annotations from the io pen by selecting an item from a drop-down menu.

Since TAbiWord supports digital document annotation, PADD integration merely requires an additional step before inserting a stroke, or the digital representation of a mark: Logitech io strokes are converted to the Microsoft Tablet PC API[14] stroke format and the stroke is translated and scaled from paper-based coordinates to the digital coordinates used with the TAbiWord annotation interface. This allows TAbiWord to process strokes in a uniform manner, regardless of the source.

By supporting both digital and paper-based annotation, the user is able to select the interface most appropriate for their task.

Text-Annotation Association

One of the most difficult issues in digital annotation is determining the underlying text with which the annotation should be associated. When an annotation is made on the paper or digital document and later repositioned in the digital copy, the semantic meaning of the mark must be preserved. It is important to ensure that each marking is associated with the correct text since many markings, such as the insertion mark, capitalize mark, or delete mark, can have a drastic change in meaning if shifted even by a single letter.

Our initial approach has focused on the ANSI standard proofreading marks[15] taught in grade schools. Many of these markings are contextually associated with a single letter or word but may be drawn over several words or even several lines depending on the size of the pen and font. These marks often denote the letter or word of interest through a sharp angle as seen most clearly in the use of an upside-down “v” insertion mark. This abrupt change in direction provides a clear point to determine the context of the mark.

In order to determine the sharpest change in direction of an annotation, the system examines the points that form the stroke. Using the law of cosines, it is possible to determine the angle formed by three points on a plane. To find the sharpest change in direction of an annotation, the system examines the points in consecutive groups of 3 and finds the location at which the smallest angle appears, as calculated by the law of cosines.

It is important to note, however, that due to the natural motion of the human hand, there are often small, sharp changes in direction as a user starts to make or ends a stroke. Although these changes in direction are often noticed by the human eye, the sudden variations in data collected by the Tablet PC stylus or Logitech io digital pen can lead to a semantically incorrect calculation of the sharpest angle. As a result, it is necessary to examine the points in each stroke using a sliding window to reduce the influence of these statistical outliers. User tests on a Tablet PC show that a sliding window of size 17 or greater will tend to calculate the change of direction in the stroke correctly and thus, the contextual position of the proofreading mark within the document, as illustrated in Figure 1.

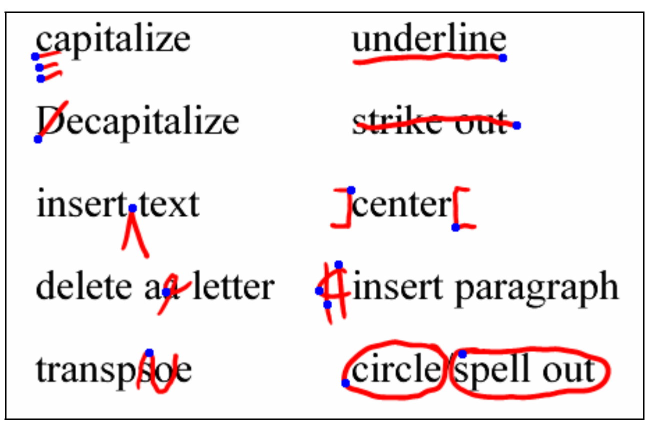

Figure 1: Common proofreading marks and their corresponding insertion point (blue dots).

Storing Annotations and Document Structure

Once the text-annotation association has been calculated, it must be saved so that it is possible to determine where to reposition the annotation after the document changes. Previous systems, such as XLibris[4] and Calliso[8], have stored annotations made to a document in a separate file or database. This design allows a system to support document annotation on a file that a user may not have permission to modify. The focus of our project, however, is to improve the writing process. Thus, it is assumed that the users of the TAbiWord system have permission to modify the documents that they are annotating.

Previous systems have used several methods for saving the association between text and annotation. The most popular method has been to save the annotation with a copy of the text to the left and/or right of the annotation. By saving the surrounding text, annotations can be repositioned by searching through the document until the surrounding text is found. In a document with fixed content, this approach can be sufficiently robust to ensure the correct repositioning of markings. In a document with dynamic content, however, it can be difficult to determine the correct position of annotation placement if the contents of the document change dramatically. If the system is unable to determine the text with which an annotation should be associated, the annotation is said to be "orphaned." In order to avoid orphaning, we have taken a different approach for annotation storage than previous systems.

To help preserve the correct association between ink markings and contextual document position, our system stores the ink markings in an intra-document anchor[6] rather than in a separate file or database. This affords several advantages in digital reflow. By storing annotations directly in the document structure, other areas of the document may be changed freely without losing the contextual position of existing strokes. This is termed intra-document anchoring. Intra-document anchoring also allows the system to be more responsive as the writing process continues since it is not necessary to recalculate the association between text and annotation. The issue of orphaned markings is also avoided, as it is only necessary to find the text associated with a marking at the time of annotation. This helps reduce errors in reflow.

Annotation information is inserted directly into the document at the associated semantic position. Following the design decisions that AbiWord developers made for hyperlink and image storage, it is only necessary to store a unique stroke identifier at the contextual position inside of the document text. The serialized stroke objects can be stored in one large block in a special data segment at the end of document. This reduces the number of expensive serialization operations that must occur, and thus helping to improve system responsiveness.

An example of a TAbiWord document and how it is structured after it is saved can be seen in Figures 2 and 3.

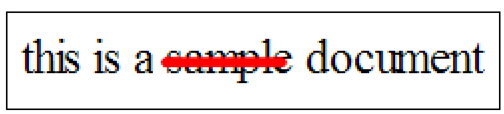

Figure 2: A sample TAbiWord document.

```
<tabiword>
      <section>
            <p style="Normal"><c props="lang:en-US">this is a
            <ink stroke="1">sample</ink> document</c></p>
      </section>
      <data>
            <d name="ink" mime-type="text/ink-xml" base64="yes">
            YmFzZTY0OkFOZ0JIQU9BZ0FRZEFsSUdBVEJjNHMrenFvcFlUSWtQW
            mtNSjZkbk1XYjlnNFp1WFZrcU5hWUxMRlZwSkdkSG1KY0hKcjlaQX
            FtZlJyN0MxaHE0RERrZ1VSUDhCUlV0R1MxY05BQUFBQlFRTFpHVm1
            IZ1FFZ3ZHZ0NpRUVoL0NMWjRSQTAxQ0MvbWNUK1p4UUFpOXdBQVlB
            QXdCaS8vLy9BemdEbmdBS0hBSU9SZDBaRUExcmcxd0FBaTk0QUFZQ
            UF3Q0svLy8vQXpnRG5nQUtJeFNDL29kTCtoYVFBc0FBZ3Y1bkUvbW
            NVQUFBQWpFNEFBUTZBT3NBQUFNNEE1NEFDaGtDRGxVcFZsQU5hNE5
            jQUFJeE9RQUROZ2ZnQUFNNEE1NEE=</d>
      </data>
</tabiword>
```


Figure 3: Example of the TAbiWord document structure.
(Note: Some extraneous information has been removed).

Repositioning/Reflow

TAbiWord is unique from previous systems in that it supports reflow in documents with dynamic content. This is achieved through the use of intra-document anchors and a repositioning algorithm that is integrated with the document rendering and layout engine.

When TAbiWord displays a document, it iterates through a list of document elements, rendering each one in turn. These elements include text, formatting, images, tables, and other document elements. As each element is rendered, its properties such as position and size are used to determine the rendering position of the next element. Since the position of any given document element is dependent on the rendering/layout properties of every preceding document element, it is very difficult to predict what the position of any given document element will be after the user changes the document's content or layout. As a result, digital reflow is performed reactively.

As the TAbiWord system renders the document, it calculates the position at which each element should be rendered. This position is a function of the width of the last element rendered, as well as any necessary formatting such as font size, justification, and margins. When the TAbiWord system encounters an ink-element, it suggests the position at which the stroke should be rendered. Since the system uses intra-document anchors that have been inserted around the semantically correct text, one can assume that the position provided by the formatting engine from the TAbiWord system is the correct position to which the ink should reflow. We use the difference between the new position and the old position to create an affine transformation which is applied to the stroke to translate and scale the stroke to the appropriate position in the document. In this way, the multivalent ink layer that is above the text of the document is synchronized with any changes made to the underlying document, pressuring the association between the annotation and the text, as illustrated in Figure 4.

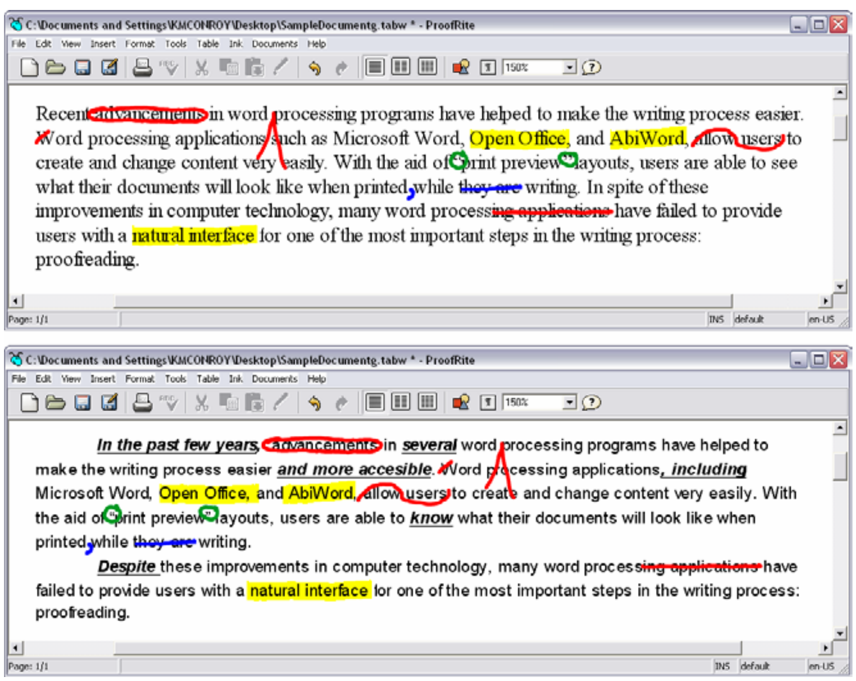

Figure 4: Text and annotation before reflow (top) and after reflow (bottom).

Splitting Large Annotations for Reflow

As a document's content or layout changes, text that appears on a single line may be repositioned and span several lines. If the text associated with an annotation begins to span multiple lines, it is necessary to split the annotation into smaller markings which can remain attached to individual words or blocks of text. Previous systems have addressed the problem of splitting annotations into smaller parts in several ways. TAbiWord is similar to XLibris[4] in that it splits annotations into multiple marks that are each associated with a single word. By breaking each marking into one or more single-word annotations, we are able to ensure that the user can change the document's content, structure, or layout in any arbitrary manner while still ensuring that the stroke will retain its semantic meaning.

Splitting large markings into small markings is particularly important in documents with dynamic content. As a document changes, there is no guarantee that the text that holds the semantic meaning of the marking will remain on the same line. Thus, markings have to be broken into smaller markings which can be associated with a smaller block of associated text.

This allows long strokes, such as the cross out mark, to reflow over multiple lines, as illustrated in Figures 5 and 6.

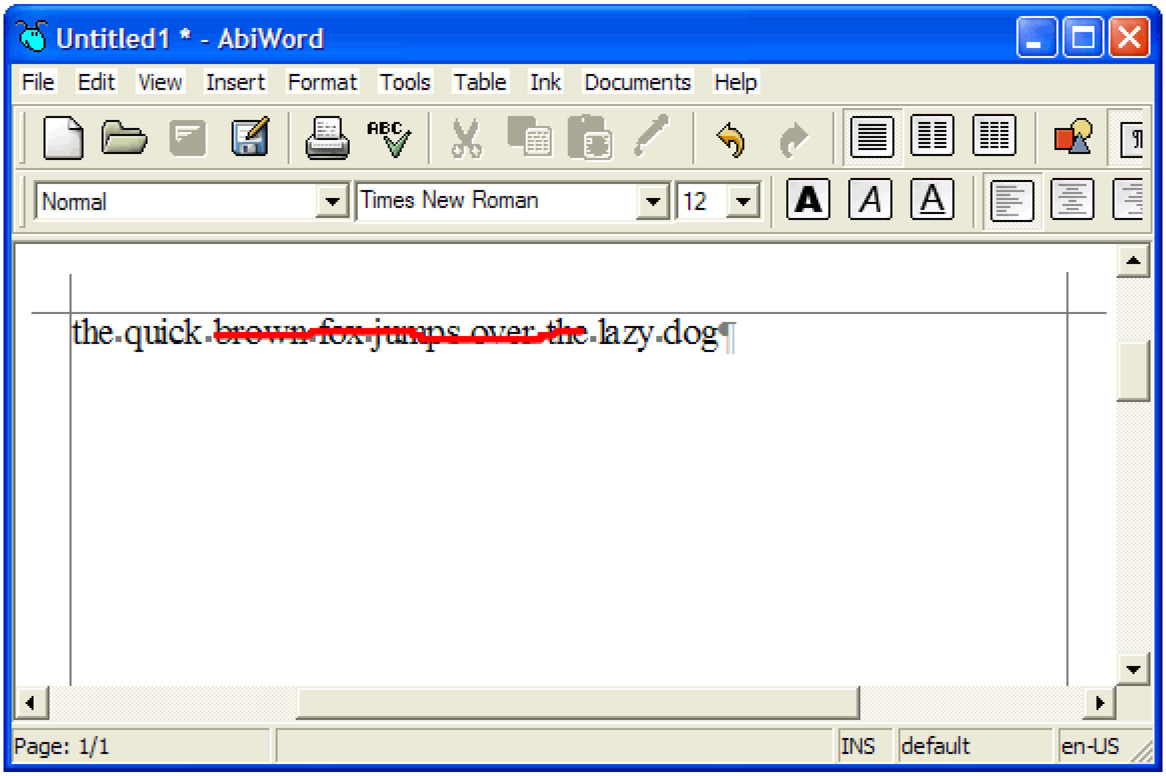

Figure 5: Text and annotation before splitting and reflow.

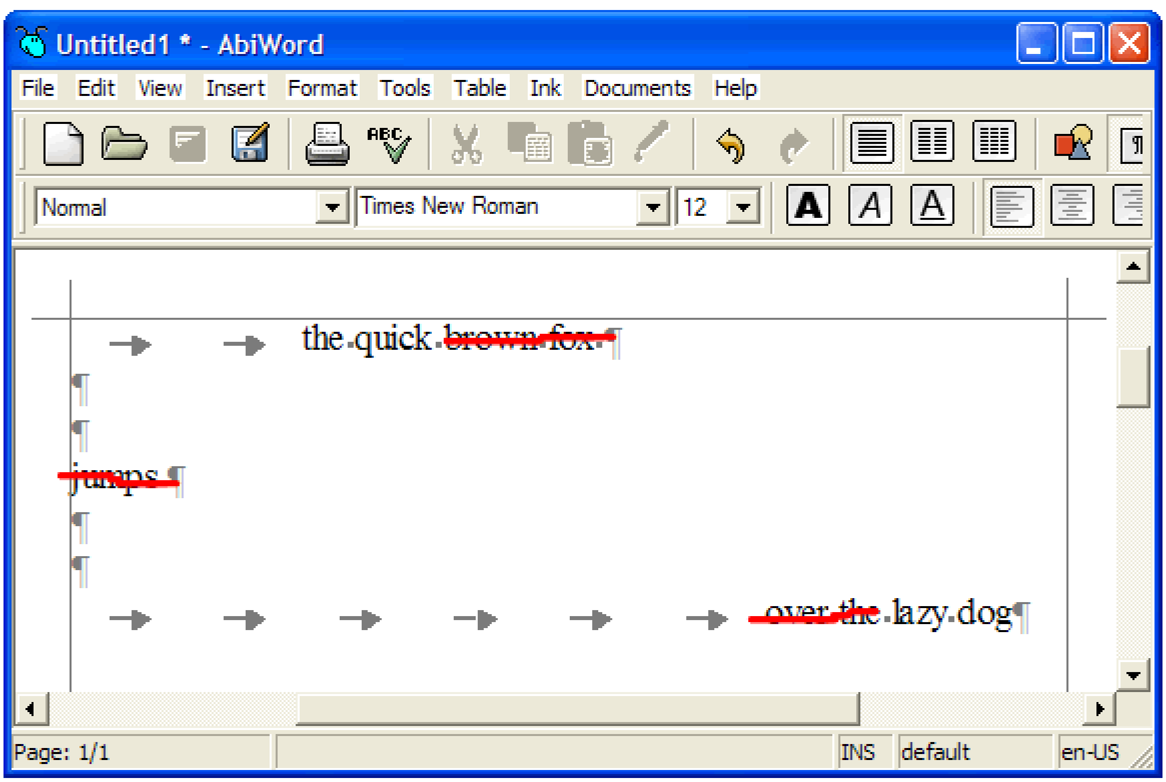

Figure 6: Text and annotation after splitting and reflow.

Analysis

By allowing users to make annotations on the Tablet PC with a stylus and allowing these annotations to reflow, TAbiWord helps reduce the separation that typically appears between annotations made in the proofreading process and the next revision of the document. Users who do not want to use paper can now perform every step of the writing process on the computer using familiar methods. The use of a stylus frees the user to interact with digital documents as if they were paper while digital reflow allows the user to retain the ability to change the document easily without losing information.

By integrating with the PADD architecture, TAbiWord simplifies the writing process by allowing users to continue to use their existing methods of interacting with documents, while removing the process of manually translating information from paper back to the computer.

The user desire to interact with documents on paper is clear; as computers have become more widespread, the amount of paper consumption has increased![16] The ability to synchronize a digital document with a paper document means that the user no longer has to consider a document that exists in separate media as a separate version. Paper-based information can be transferred back to the digital source quickly and easily, freeing the user to continue the writing process rather than attempting to synchronize information between the two media. As seen in Figure 7, this allows paper-based annotations to be retained across multiple revisions, a feature that existing paper-computer systems do not allow. PADD integration bridges a significant gap between paper and computers.

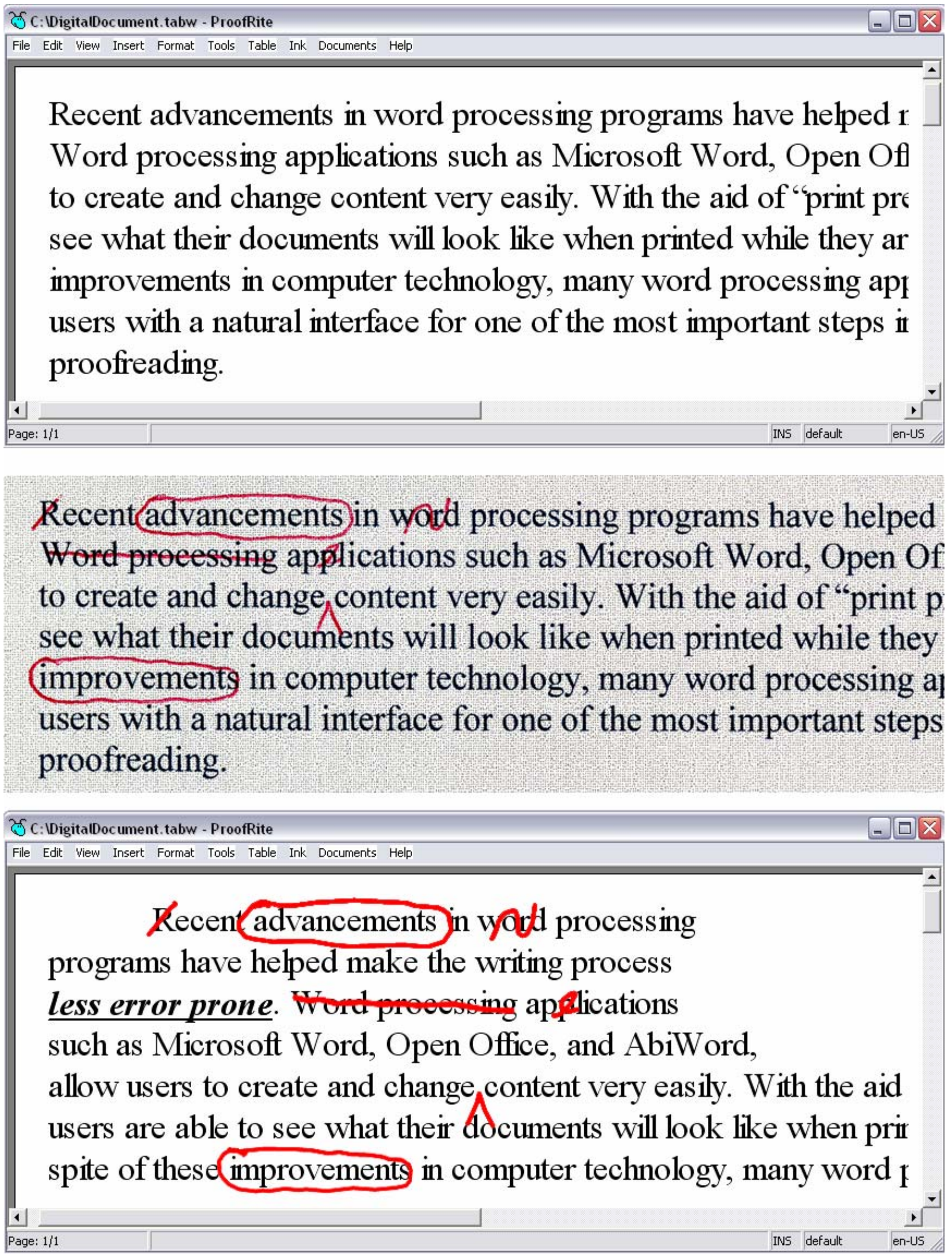

Figure 7: Top: TAbiWord document. Middle: Same document printed and annotated on Anoto paper. Bottom: Strokes reflowed in TAbiWord.

Future Work

Like previous systems, our method for text-annotation association is sufficiently robust for proofreading markup, but is limited in its ability to address freeform writing and annotations. Users may have a wide range of handwriting sizes and styles, as well as their own system of annotation. It is clear that a simple text-annotation association algorithm is not sufficiently accurate to support digital reflow. Additional work must be done regarding methods to to associate freeform markings with their semantically correct text before any system can correctly handle user expectations for digital reflow and markup.

Although we have not conducted any formal user studies of our system, several early test users have expressed interest in the ability to automatically apply annotated corrections. An "auto-correct" or "apply annotations" feature is limited by the accuracy of handwriting and annotation recognition. In order to implement a robust recognition engine, it is necessary not only to examine the annotation, but also the surrounding text and annotations. It may be possible to provide better annotation recognition by examining the position of the stroke relative to the text (i.e. the cross-out mark goes through text while the underline mark goes underneath text).

It may also be possible to improve recognition by adopting a learning algorithm that examines the user's typical annotations and the changes that the user makes to the document after creating an annotation. Such an algorithm would help identify ambiguous marks, such as the horizontal line which is used to denote both the cross-out mark and the underline mark. If the user begins to a delete a word from a block of text with a horizontal line, then the annotation may be the cross out mark, implying that the user wants to delete the remaining text associated with the mark. Alternatively, if the user applies the underline font format, then the system may want to remove the associated underline annotation.

Although reflow is an important aspect of digital annotation, the ability to preserve annotations made on paper in the digital source presents another layer of data with which the user can interact during the writing process. This presents several questions regarding the user experience of interacting with digital annotations. When should markings be applied to the text? In what ways do users wish to interact with their markings (i.e. move, erase, change color, etc)? What is the best way to present annotations from several different sources? Does the presence of annotations from the last proofreading affect the user experience of word processing? Until now, annotations made on paper have always been fixed and static, so users have no experience or prior expectations of how to interact with annotations that reflow. The question of how users wish to interact with digital annotations is fundamental to the writing process. Further studies must be done to determine how word processors that support digital annotation and reflow can best serve the user.

Conclusion

This paper introduced the TAbiWord word processor which supports digital annotation and reflow through both paper-based and computer-based interfaces. Through the use of intra-document anchors and the inclusion of annotation reflow in the document rendering engine, it is possible to support digital reflow in documents with dynamic content, structure, and layout. Paper-simulation on the Tablet PC as well as PADD integration for paper-based annotations allows users to bridge the gap between paper and computers in the writing process. Although the system works well for proofreading marks, there is still much work to be done in the area of digital annotation interaction. Once digital documents can be synchronized with paper documents, word processors must be expanded to support new aspects of the writing processing.

Online Resources

The source code for TAbiWord is distributed under the GNU General Public License and is available at *http://sourceforge.net/projects/tabiword*. A video demonstration of TAbiWord is available at *http://www.cs.umd.edu/~kmconroy/*.

Acknowledgements

The author wishes to thank François Guimbretière and Dave Levin for their help with PADD integration and comments on early drafts of this paper; Samantha Ahalt and Cathryn Conroy for comments on early drafts of this paper; Dom Lachowicz, Marc Maurer, and the rest of the AbiWord development team for technical assistance.

Bibliography

[1] Microsoft Word. *http://www.microsoft.com/word/.*

[2] Open Office, *http://www.openoffice.org.*

[3] AbiWord. *http://www.abiword.com.*

[4] Golovchinsky, G. and L. Denoue. Moving markup: repositioning freeform annotations. *Proceedings of UIST 2002*, pp. 21 - 30.

[5] Schilit, B.N., G. Golovchinsky, and M.N. Price. Beyond paper: supporting active reading with free form digital ink annotations. *Proceedings of CHI'98*, pp. 249 - 256.

[6] Phelps, T.A., and Wilensky, R. Robust intra-document locations. In *Proceedings of WWW9 World Wide Web Conference*. Amsterdam, The Netherlands, May 2000.

[7] Brush, A.J. Bernheim, Bargeron, David, Gupta, Anoop, and Cadiz, JJ. Robust Annotation Positioning in Digital Documents. In *Proceedings of CHI 2001*. pp. 285 – 292.

[8] Bargeron, David and Moscovich, Tomer. Reflowing digital ink annotations. *Proceedings of CHI 2003*, pp. 385 – 393.

[9] Phelps, T.A., and Wilensky, R. Multivalent Annotation. In *Proceedings of the Frist European Conference on Research and Advanced Technology for Digital Libraries*. Pisa Italy, September 1997.

[10] Tablet PC. *http://www.tabletpc.com.*

[11] Guimbretière, François. Paper Augmented Digital Documents. *Proceedings of UIST 2003*, pp. 51-60.

[12] Logitech, io digital pen. *http://www.logitech.com.*

[13] Anoto. *http://www.anoto.com/*

[14] Tablet PC SDK. *http://msdn.microsoft.com/tabletpc/*

[15] American National Standards Institute. *American national standard proof corrections.* New York: American National Standards Institute, 1981.

[16] Sellen, A.J. and R.H.R. Harper. *The Myth of the Paperless Office*, 1st ed. MIT Press, 2001.