

# Multimedia Clip Generation from Documents for Browsing on Mobile Devices

Berna Erol*	Kathrin Berkner	Siddharth Joshi
RICOH Innovations	RICOH Innovations	Department of Electrical Eng.
California Research Center	California Research Center	Stanford University
2882 Sand Hill Road	2882 Sand Hill Road	Packard 243, 350 Serra Mall
Menlo Park, CA, USA	Menlo Park, CA, USA	Stanford, CA, USA
berna_erol@rii.rioh.com	berkner@rii.rioh.com	sidj@stanford.edu

## ABSTRACT

Small displays on mobile handheld devices, such as PDAs and cellular phones, are the bottlenecks for usability of most content browsing applications. Generally, conventional content such as documents and web pages need to be modified for effective presentation on mobile devices. This paper proposes a novel visualization for documents, called Multimedia Thumbnails, which consists of text and image content converted into playable multimedia clips. A Multimedia Thumbnail utilizes visual and audio channels of small portable devices as well as both spatial and time dimensions to communicate text and image information of a single document. The proposed algorithm for generating Multimedia Thumbnails includes 1) a semantic document analysis step, where salient content from a source document is extracted, 2) an optimization step, where a subset of this extracted content is selected based on time, display, and application constraints, and 3) a composition step, where the selected visual and audible document content is combined into a Multimedia Thumbnail. Scalability of MMNails that allows generation of multimedia clips of various lengths is also described. A user study is presented that evaluates the effectiveness of the proposed Multimedia Thumbnail visualization.

**EDICS Categories:** 3-CONT, 4-ANSY , 3-INTF

**Index Terms:** mobile document, document conversion, adaptive content delivery, document repurposing, multimedia generation, content navigation, multimedia thumbnail.

## 1. INTRODUCTION

Mobile devices are becoming more ubiquitous and used increasingly for tasks that were traditionally performed on desktop and laptop computers. Even though the memory and computational capabilities of these devices will continue to improve, the small display sizes and limited input capabilities for user interaction are likely to remain the major bottlenecks for many mobile applications. These limitations have lead to research on user interfaces and applications, generally addressing two issues: (1) *overcoming non-recognizability* of information on the small screens [1]-[7] and (2) *assisting navigation* when there is limited input capability [8].

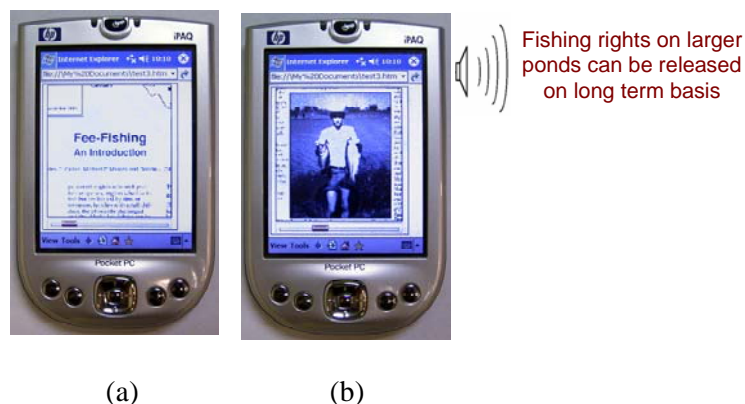
There has been research on addressing these issues for browsing web pages[1]-[3], photos[4][5], and videos on small displays[6], which we review in Section 2. Besides web pages, photos, and video clips, formatted documents are another type of “information carrier.” Current solutions for viewing documents on small displays consist mainly of running a document viewer such as MSWord or Acrobat Reader. However, interactions with such a document viewer through zooming and scrolling are difficult to perform on small mobile devices.

Most techniques used to adapt content to mobile devices reformat content by changing the document layout [1]-[3][7]. Layout of documents, defined by page-breaks, line-breaks, columns, margins, etc., communicates semantic information about the document such as hierarchical structure and reading order, much as hyperlinks in web pages encode structure. Users may find it difficult to recognize a document or navigate through it if that document’s layout has been changed [9].

Automatic browsing through content without reformatting has been applied to photos [4][5]. In contrast to photographs of natural scenes or people, document images contain typically a large amount of high frequency data such as text. Saliency points in documents images that are useful for navigation may include title, authors, abstract, figures, and references section [10], in contrast to people’s faces and a foreground object in typical photographs. Moreover, text in documents is meant to be read in a

predetermined reading order and image and text units may be linked, e.g., a figure picture and a figure caption. These properties are document specific and absent from generic photos.

To address the problem of viewing documents on small displays, we propose a new document visualization called *Multimedia Thumbnail* (MMNail). An MMNail can be seen as a short video clip of a document that gives a guided tour through a document. In an MMNail visualization both visual and audio channels of a mobile device are used to communicate document information. The visual channel is used to present dense spatial information by zooming into and panning over the most important document elements, such as title and figures, while the audio channel is used to communicate speech-synthesizable document information, so called *audible* information, such as keywords and figure captions. In this way, both recognizability and navigation problems are addressed by having text readable and figures comprehensible after zooming and panning and minimizing the navigational input required by the user.



**Figure 1. Multimedia Thumbnails present documents in small displays by automatically zooming into important document parts, such as (a) title and (b) figure, as well as transcoding some document text, such as figure captions, into an audio signal via synthesized speech.**

The MMNail generation algorithm is composed of extracting semantic visual and audible document elements, optimizing the selection of document elements with respect to given time, display, and application constraints, and synthesizing selected document elements into a playable MMNail. We introduced the initial MMNail algorithm in [11], performed an exploratory study of users' document browsing behavior on a mobile device and collected initial feedback on Multimedia Thumbnail examples

[12]. In this paper we improved the MMNail generation algorithm as well as the playback interface based on the initial user feedback. In our previous work we concentrated on scanned documents stored in a raster-scan representation. In this paper we propose a way for generating MMNails for input formats such as HTML and MSWord, utilizing the semantic information present in these representations, as described in Section 3.1. In Section 3.2 we explain how to optimally select document information for a given visual and audio channel. From our earlier user study [12] it became clear that an MMNail optimized for single time duration could not satisfy all user needs. In this paper we address this problem by introducing scalable MMNails with respect to the time duration, as described in Section 3.3. A flexible user interface for MMNail playback is also introduced in Section 4. Examples of MMNails are given in Section 5. In order to evaluate our system design and implementation we performed a new set of user studies. In the new user study we compared viewing of regular formatted documents in a PDF viewer with viewing of Multimedia Thumbnails. Our results indicate statistically significant (a p-value of 0.05 in a paired sample t-test) improvement of document comprehension with MMNail viewing over that with PDF viewing, as reported in detail in Section 6. A summary of our work and discussion of applications and future directions are given in Section 7.

## **2. RELATED WORK**

Research in the area of adaptation of documents to different output devices and allocation of document information to different output channels has diverged into at least two directions. One direction concerns how to reformat document content for small devices, transforming information represented in one visual information channel into another visual channel. The other direction concerns making document information accessible to visually impaired users by transforming visual information into audio information.

In the visual-to-visual transformation category, some solutions focus mainly on readability of text, displaying some text in larger fonts and allowing users to select page elements to be zoomed in or

collapsed [1], summarizing the text content [2], semantic grouping [3] and re-flowing content based on reading order [7]. Solutions that focus both on text and images include Enhanced Thumbnails [13] and SmartNails [14]. Enhanced Thumbnails contain keywords extracted from the source document and pasted onto a low-contrast downsampled page image. SmartNail technology [14] creates an alternative image visualization for a single document page by scaling, cropping, and reflowing page elements, subject to display size constraints. Both techniques include image and text, but the output is a static visual representation of each page. In Multimedia Thumbnails, the output consists of document information from multiple pages represented in a dynamic way using animation and audio.

Some of the most relevant prior art to our current work in the visual-to-visual category is described in [4][5], where a method for non-interactive picture browsing on mobile devices was proposed. The goal there was to find salient, face and text regions on a picture automatically and then apply zoom and pan motions on this picture to automatically provide informative close ups to the user. This method concentrates on representing photos, whereas our method focuses on representing high-resolution multi-page document content. Moreover, the automated picture browsing technique shows only visual information, whereas we employ visual and audio channel for communicating document information.

There has also been some work on re-targeting audiovisual content, which was produced to be viewed in large displays, to small displays. The method in [6] converts high-resolution video clips to play on small displays by adding virtual zoom and pan operations in order to retain the recognizability of the content based on automatically extracted salient image regions and motion activity. Unlike our work where we transform visual information to audiovisual information, the research in [6] concentrates on conversion of audiovisual content to another audiovisual representation.

Works that fits into the visual-to-audio transformation category include document browsers that support synthesizing text to speech, such as Adobe Acrobat PDF reader for visually impaired users [15]. Furthermore, some work has been done on developing Web browsers for blind and visually impaired users [16][17]. The focus in [17] is to map a graphical HTML document into a 3D virtual sound space,

where non-speech auditory cues differentiate HTML documents. Their goal is to transform as much information as possible into the audio channel. In contrast, MMNails contain document information optimally selected for communication through both, visual and audio channels.

Multimedia Thumbnails can be seen as being the first technique that transforms purely visual, multi-page, formatted document information into an audiovisual representation that exploits both the visual and the audio channel of a mobile device. Allocation of the visual and audio channel is performed optimally with respect to information content, display size, time duration, and user's task.

### 3. GENERATION OF MULTIMEDIA THUMBNAILS

The proposed Multimedia Thumbnail generation algorithm accepts a scanned documents as well as an electronic document as an input and is comprised of the following steps as depicted in Figure 2:

1. **Semantic Analysis and Extraction:** Visual document elements and audible document elements are automatically identified and extracted from the source document. These extracted document elements are semantically labeled, for instance as abstract, section heading, or figure caption. A reading order is also estimated.
2. **MMNail Optimization:** A subset of extracted document elements is selected to be included in the Multimedia Thumbnail. Information attributes that indicate the importance of each document element are determined based on user and task constraints. Time attributes, that determine the time required to present each document element, are computed based on display constraints. Then, given a time duration constraint, those attributes are used to optimize the selection of document elements to be included in the MMNail.
3. **Composition into a playable form and MMNail representation:** Selected Multimedia Thumbnail elements, animation instructions or rendered animations, and synthesized speech are represented in a file with or without the original document file.

Each of these steps is explained in more detail in the next subsections.

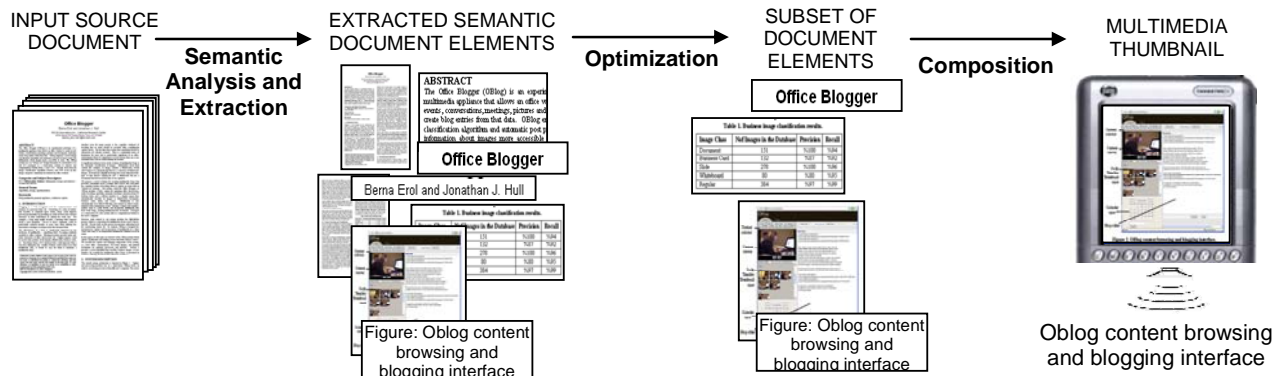


Figure 2. Overview of MMNail generation steps.

### 3.1 Semantic Analysis and Extraction

Semantic document elements are the building blocks of Multimedia Thumbnails. We divide these elements into three groups depending on the element presentation type:

- Purely visual elements that can be presented to the user only through the visual channel;
- Purely audible elements that can be presented to the user only through the audio channel; and
- Audiovisual elements that can be presented to the user synchronously through both visual and audio channels.

Examples of purely visual elements include page thumbnails and figures without any captions. Examples of purely audible elements include elements that can be communicated easily in the audio channel without a visual representation such as keywords and number of pages. An example of an audiovisual element is a figure with a caption, where the figure can be presented through the video channel and the caption can be synthesized to speech and presented through the audio channel.

#### 3.1.1 Extraction of Document Elements

In order to automatically generate a Multimedia Thumbnail, semantic document elements and their locations on the page as well as the reading order should be extracted. If a document is in a scanned

image format, postscript, or PDF document, a preprocessing step is applied to the document that includes layout analysis and optical character recognition via commercial software. The software also automatically determines a reading order based on the layout. The output of the preprocessor, which is a collection of document elements, is further analyzed to assign semantic labels to visual document elements, such as title, section heading, and figure captions. Publication name and date are generally difficult to automatically extract. If this information is not present in the file header, it can be provided to the algorithm as metadata.

In addition to identifying visual information in this way, the analysis step also determines audible document information from the document image and metadata. Examples of audible information include figure captions, keywords, publication date, and publication name that can be synthesized to speech. We extract keywords from a document with TF-IDF analysis [18].

### ***3.1.2 Pre-processing for Real-Time Rendered Documents***

In static documents such as postscript and PDF, the layout of a document page is already known as well as the coordinates of text and figures. However, this is not the case in symbolic source documents that are rendered real-time, such as HTML pages and MSWord documents. In real-time rendered documents, the layout of a page changes depending on the page size and the selected printer properties. Nevertheless, these symbolic source representations potentially contain very valuable semantic information about their contents, such as text and formatting information of titles, headers, and figures that is usually difficult to obtain accurately from image-based representations.

In most commonly used symbolic representations, it is possible to extract document elements and their semantic tags from the file by either simple parsing of the description (e.g., HTML) or using the APIs that allow access to the proprietary representations (e.g., MSWord, MSPowerPoint). The extracted information is stored in an XML description file. The coordinates of those elements are not known until after the file is rendered for display or printing. In some cases a Document Object Model (DOM) [19]

can be used to obtain this information. Nevertheless, DOM is not supported by all document representation formats. Our solution is to generate a static visual representation of a symbolic source document by printing the file and storing the coordinates of the document elements in a second XML description file. The generated XML description is less accurate in terms of semantic labels of document elements, for example links between figures and figure captions do not exist. On the other hand it contains accurate location information for images and text. Merging of the information contained in the two XML files is performed by content matching. Content matching is performed based on the document element type. For matching of figure content, bitmaps corresponding to figure areas are extracted from original files using the location information present in the XML files and color layout similarity is employed. For matching of text elements, a tri-word-gram word similarity measure is used that is similar to the method employed in [20]. The output of content matching is a description of the document content that contains semantic labels of document elements, such as title, section heading, and caption, as well as their absolute coordinates on the page.

### ***3.1.3 Selection of Presentation Channels***

Some document elements, such as a title can be easily presented only through the visual channel by zooming in and panning over the title, presented only through the audio channel by synthesizing title to speech, or presented through both channels. *Possible* and *preferred* presentation channels for document elements are presented in Table 1 based on our previous user study [12]. For example, users prefer to have “author names” presented only in the visual channel because the text-to-speech engine often wrongly pronounces names. In contrary, users like very much to have title and figures with captions in both visual and audio channels. Users also indicated that they like to hear the number of pages and page numbers in the audio channel. More details can be found in [12].

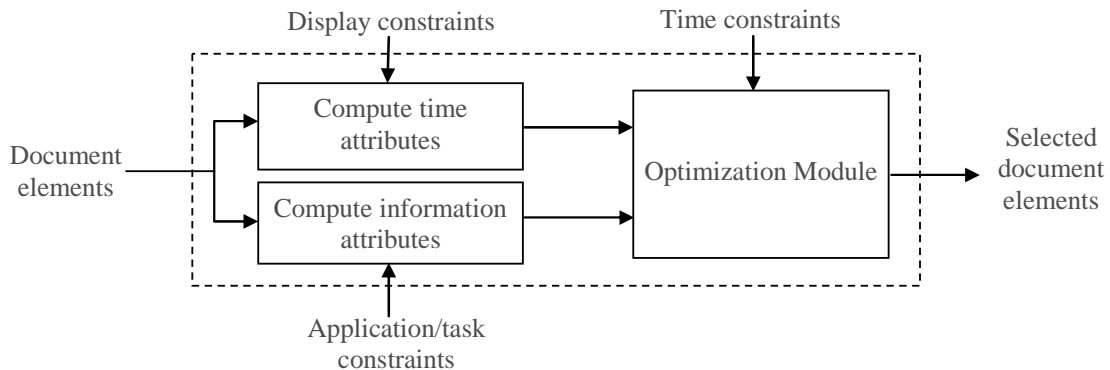
**Table 1. Preferred presentation channels for document elements based on our user study in [12].**

<b>Document element</b>	<b>Possible presentation channels</b>	<b>Preferred presentation channel</b>
Title	Visual and Audio	Audiovisual
Figure with captions	Visual and Audio	Audiovisual
Figure with no captions	Visual	Visual
Section headings	Visual and Audio	Audiovisual
Abstract	Visual and Audio	Visual
References	Visual and Audio	Visual
Page thumbnail	Visual	Visual
Author names	Visual and Audio	Visual
Publication name	Visual and Audio	Audio
Publication date	Visual and Audio	Audio
Keywords	Audio	Audio
Page number	Visual and Audio	Audio
Number of pages	Audio	Audio

### **3.2 MMNail Optimization**

Initially we designed an ad-hoc algorithm that selected certain parts of documents based on the target MMNail duration. For example, if MMNail is 20 seconds we included title and page thumbnails and if it is 30 seconds, we included title, thumbnails and figures, and so on. However after generating MMNails for various different documents and receiving feedback from users it became clear that (1) documents vary in content greatly, e.g., some documents have readable titles that do not need zooming in, some documents have many figures with long captions, and some other documents don't have any figures, and (2) MMNail generation depends on user preferences and target applications [12]. Therefore, it is important to develop an MMNail generator which is easily expandable to include different document elements and configurable for personal preferences and target application.

In this section, we present a generalized optimization framework which selects document elements to form an MMNail based on time, application, user, and display size constraints. An overview of the optimizer is presented in Figure 3. In the optimizer, first, for each document element we compute a time attribute, i.e. time required to display the element, and an information attribute, i.e. information content of the element. The computation of attributes is described in detail in Sections 3.2.1 and 3.2.2. Display constraints of the viewing device are taken into account when computing time attributes. For example, it takes longer time to present a text paragraph in a readable form in a small viewing area than in a large viewing area due to increased amount of zooming and scrolling in the small viewing area. Similarly, target application and user preferences need to be taken into account when computing information attributes. For example, for some tasks the abstract or keyword elements can have higher importance than other elements such as a body text paragraph.



**Figure 3. Overview of the MMNail optimizer.**

The optimization module selects elements from the set of document elements, such that the total information content of the MMNail is maximized, subject to a time constraint. Let the information content of an element  $e$  be denoted by  $I(e)$ , the time required to present  $e$  by  $t(e)$ , the set of available document elements by  $E$ , and the target MMNail duration by  $T$ . The optimization problem is

$$\begin{aligned}
 & \text{maximize} && \sum_{e \in E} x(e) I(e) \\
 & \text{subject to} && \sum_{e \in E} x(e) t(e) \leq T \quad (1)
 \end{aligned}$$

$$x(e) \in \{0,1\}, e \in E,$$

with optimization variables  $x(e)$ ,  $e \in E$ . For an element  $e$ ,  $x(e)=1$  means  $e$  is selected to be included in the MMNail, and  $x(e)=0$  means  $e$  is not selected for inclusion.

The problem (1) is a ‘0-1 knapsack’ problem. Therefore it is a hard combinatorial optimization problem [23]. In our implementation, a greedy approximation is employed to solve (1). First, the document elements  $e \in E$  are sorted according to the ratio  $I(e)/t(e)$  in descending order, i.e.,

$$\frac{I(e_1)}{t(e_1)} \geq \dots \geq \frac{I(e_m)}{t(e_m)},$$

where  $m$  is the number of elements in  $E$ . Then the document elements are included

into the MMNail, starting from the first element, until the sum of the time attributes of the included elements is the same or very close to the target duration  $T$ . For practical purposes, this approximation of problem (1) should work quite well, as we expect the individual elements to have much shorter display time than the total MMNail duration.

### ***3.2.1 Time Attributes***

The time attribute of a document element can be interpreted as the approximate duration that is sufficient for a user to comprehend that element. Computation of time attributes depends on the type of the document element.

The time attribute of a text document element (e.g., title) is determined to be the duration of the visual effects necessary to show the text segment to the user at a readable resolution. In our experiments, text was determined to be at least 6 pixels high in order to be readable on an LCD (Apple Cinema) display. If text is not readable once the whole document is fitted into the display area (i.e. in a thumbnail view), a zoom operation is performed. If even zooming into the text such that the entire text region still fits on the display is not sufficient for readability, then zooming into a part of the text is performed. A pan operation is carried out in order to show the user the remainder of the text. In order to compute time attributes for text elements, first the document image is down-sampled to fit the display area. Then a zoom factor  $Z(e)$  is determined as the factor that is necessary to scale the height of the smallest font in the text to the

minimum readable height. Finally the time attribute for a visual element  $e$  that contains text is computed as

$$t(e) = \begin{cases} SSC \times n_e, & Z(e) = 1 \\ SSC \times n_e + Z_C, & Z(e) > 1 \end{cases} \quad (2)$$

where  $n_e$  is number of characters in  $e$ ,  $Z_C$  is zoom time (in our implementation this is fixed to be 1 second), and  $SSC$  (Speech Synthesis Constant) is the average time required to play back the synthesized audio character.  $SSC$  is computed as first synthesizing a text segment containing  $k$  characters, next measuring the total time it takes for the synthesized speech to be spoken out,  $\tau$ , and then computing  $SSC = \tau/k$ . The  $SSC$  constant may change depending on the language choice, synthesizer that is used, and the synthesizer options (female vs. male voice, accent type, talk speed, etc). Using the AT&T speech SDK [21],  $SSC$  is found to be equal to 75 ms when a female voice was used. The computation of  $t(e)$  remains the same even if an element cannot be shown with one zoom operation and both zoom and pan operations are required. In such cases, the complete presentation of the element consists of first zooming into a portion of the text, for example the first  $m_e$  out of a total of  $n_e$  characters, and keeping the focus on the text for  $SSC \times m_e$  seconds. Then the remainder of the time, i.e.  $SSC \times (n_e - m_e)$ , is spent on the pan operation.

The time attribute for an audible text document element  $e$ , e.g., a keyword, is computed as

$$t(e) = SSC \times n_e, \quad (3)$$

where  $SSC$  is the speech synthesis constant and  $n_e$  is the number of characters in the document element.

For computing time attributes for figures without any captions, we make the assumption that complex figures take a longer time to comprehend. The complexity of a visual figure element  $e$  is measured by the figure entropy  $H(e)$  that is computed extracting bits from a low-bitrate layer of the JPEG2000 compressed image as described in [22]. The time attribute for a figure element is computed as  $t(e) = \alpha H(e) / \bar{H}$ , where  $H(e)$  is the figure entropy,  $\bar{H}$  is the mean entropy, and  $\alpha$  is a time constant.

$\overline{H}$  is empirically determined by measuring the average entropy for a large collection of document figures. Time required to comprehend a photo might be different than that of a graph or a table, therefore, different  $\alpha$  can be used for these different figure types. Moreover, high level content analysis, such as face detection, can be applied to assign time attributes to figures. We do not perform content analysis or distinguish different figure types in this paper and  $\alpha$  is fixed to 4 seconds, which is the average time a user spends on a figure in our experiments.

An audiovisual element  $e$  is composed of an audio component,  $A(e)$ , and a visual component,  $V(e)$ . A time attribute for an audiovisual element is computed as the maximum of time attributes for its visual and audible components:  $t(e) = \max(t(V(e)), t(A(e)))$ , where  $t(V(e))$  is computed as in (2) and  $t(A(e))$  as in (3). For example,  $t(e)$  of a figure element is computed as the maximum of time required to comprehend the figure and the duration of synthesized figure caption.

### ***3.2.2 Information Attributes***

An information attribute determines how much information a particular document element contains for the user. Clearly this very much depends on the user's viewing/browsing style, target application, and the task on hand. For example, information in the abstract could be very important if the task is to understand the document, but it may not be as important if the task is merely to determine if the document has been seen before.

In order to understand how important the information is in different parts of a document, we performed an exploratory and descriptive user study that we reported in [12]. The users were given two tasks: a document browsing task, where users were asked to browse documents in a small viewing area in order to determine if they have seen these documents before, and a document understanding task, where users were asked to browse documents for a limited time in order to answer questions about the document content. The users' navigation behaviors were recorded and analyzed in order to understand what document parts were viewed during browsing [12]. Table 2 shows the percentage of users who

viewed various document parts when performing the two tasks. This experiment gave us an idea about how much users value different document elements. For example, 100% of the users read the title in the document understanding task, whereas very few users looked at the references, publication name and the date. We use these results to assign information attributes to text elements. For example in the document understanding task, the title is assigned the information value of 1.0 based on 100% viewing, and references are given the value 0.13 based on 13% viewing.

**Table 2. Percentage of users who viewed different parts of the source documents for document search and understanding tasks.**

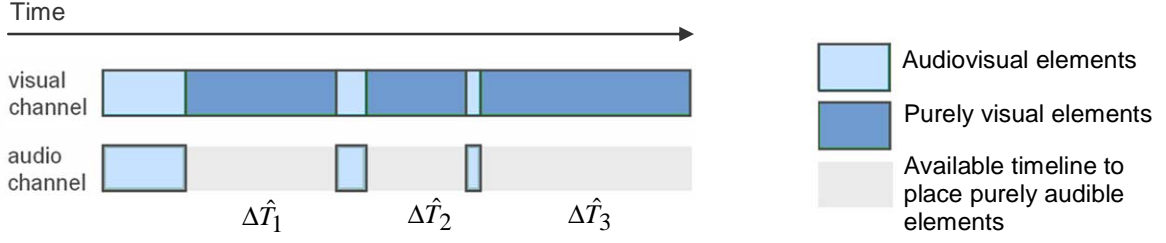
Document Part	Viewing percentage for search task	Viewing percentage for understanding task
Title	83%	100%
Abstract	13%	87%
Figures	38%	93%
First page thumbnail	83%	73%
References	8%	13%
Publication name	4%	7%
Publication date	4%	7%

### 3.2.3 Two-Stage Optimization

After the time and the information attributes are computed for the visual, audible, and audiovisual elements, the optimizer produces the best MMNail by selecting a combination of elements. The best thumbnail is one that maximizes the total information content of the thumbnail and can be displayed in the given time.

A document element  $e$  belongs to either the set of purely visual elements  $E_v$ , the set of purely audible elements  $E_a$ , or the set of synchronized audiovisual elements  $E_{av}$ . A Multimedia Thumbnail has two presentation channels, visual and audio. Purely visual elements and purely audible elements can be played simultaneously through the visual and audio channel, respectively. On the other hand, displaying a

synchronized audiovisual element requires both channels. For a feasible MMNail, playing of an audiovisual element can not coincide with playing of a visual or an audible element at any time.



**Figure 4. Audio and visual channels after the first stage of the optimization where some parts of the audio channel, with durations  $\Delta\hat{T}_1, \dots, \Delta\hat{T}_n$ , are not filled.**

Our method to produce an MMNail consists of two stages. In the first stage we select purely visual and synchronized audiovisual elements to fill the video channel. This leaves the audio channel partially filled, as illustrated in Figure 4. In the second stage we select purely audible elements to fill the partially filled audio channel.

The optimization problem of the first stage is of type (1), using a the set of visual and audiovisual document elements,  $E_v \cup E_{av}$ , as the element set.

$$\begin{aligned}
 & \text{maximize} && \sum_{e \in E_v \cup E_{av}} x(e) I(e) \\
 & \text{subject to} && \sum_{e \in E_v \cup E_{av}} x(e)t(e) \leq T \quad (4) \\
 & && x(e) \in \{0,1\}, e \in E_v \cup E_{av}.
 \end{aligned}$$

We solve this problem approximately using the linear programming relaxation as shown for the problem (1). The selected purely visual and synchronized audiovisual elements are placed in time in the order they occur in the document. The first stage optimization almost fills the visual channel, and fills the audio channel partially.

In the second stage we need to select purely audio elements to fill the audio channel which has separate empty time intervals. Let the total time duration to be filled in the audio channel be  $\hat{T}$ . If the

selected purely audible elements have a total display time of approximately  $\hat{T}$ , it may still be difficult to place the elements in the audio channel because the empty time duration  $\hat{T}$  is not contiguous. Therefore, a conservative approach is taken and instead of  $\hat{T}$  we use  $\beta\hat{T}$ , where  $\beta \in [0,1]$ . Further, we only consider the purely audio elements which do not have a long display time to be included in the MMNail. We do this by comparing the display time of an element with the average length of the empty intervals of the audio channel  $\hat{T}/R$ , where  $R$  is the number of empty intervals, and consider the element set  $\hat{E}_a = \{e \in E_a \mid t(e) \leq \gamma\hat{T}/R\}$ , where  $\gamma \in [0, R]$ . The optimization problem of the second stage becomes

$$\begin{aligned}
& \text{maximize} && \sum_{e \in \hat{E}_a} x(e)I(e) \\
& \text{subject to} && \sum_{e \in \hat{E}_a} x(e)t(e) \leq \beta\hat{T} \quad (5) \\
& && x(e) \in \{0,1\}, e \in \hat{E}_a,
\end{aligned}$$

with optimization variables  $x(e)$ ,  $e \in \hat{E}_a$ . This problem is of the type (1) and it is approximately solved by the greedy approximation as shown earlier.

In our implementation we set  $\beta = 1/2$  and  $\gamma = 1$ . For most documents, setting  $\beta = 1/2$  results in selection of audio elements such that a  $\Delta\hat{T}_n$  exists where each audio element can be placed in a continuous audio segment. If this is not the case, the second stage of the optimization is solved again for a smaller  $\gamma$  until all selected audio element's complete time span can be preserved.

Instead of the two stage optimization approach, we can formulate a single optimization problem to choose the visual, audiovisual, and the audible elements simultaneously. In this case, the optimization problem is

$$\begin{aligned}
& \text{maximize} && \sum_{e \in E_a \cup E_v \cup E_{av}} x(e) I(e) \\
& \text{subject to} && \sum_{e \in E_a \cup E_{av}} x(e)t(e) \leq T \quad (6)
\end{aligned}$$

$$\sum_{e \in E_v \cup E_{av}} x(e)t(e) \leq T$$

$$x(e) \in \{0,1\}, e \in E_a \cup E_v \cup E_{av},$$

where  $x(e)$ ,  $e \in E_a \cup E_v \cup E_{av}$ , are the optimization variables.

The greedy approximation described to solve the relaxed problem (1) can not be applied to problem (6), but (6) can be relaxed to a linear program (LP), so that any generic LP solver can be applied. The advantage of solving the two stage optimization problem is that the calibration of the information attributes of the purely audible elements becomes independent of the information attributes of the visual elements. This means that multiplying the information content of all the audible elements by a positive number does not affect the solution of the two stage optimization problem; but will change the solution of the problem (6).

Readers should note that the two stage optimization gives selection of purely visual elements a priority over that of purely audible elements. If it is desired that audible elements have priority over visual elements (e.g. for consumption of document content while driving a car), the first stage of the optimization can be used to select audiovisual and purely audible elements, and the second stage is used to optimize selection of purely visual elements.

### **3.3 Composition into a Playable Form and MMNail Representation**

The optimization output is the list of visual, audible, and audiovisual elements that are included in the MMNail as well as actions, such as hold, zoom, or pan, to be performed with each element. The visual and audiovisual document elements are sorted based on the reading order. Since synchronized audio and visual elements are combined and optimized as audiovisual elements, their synchronization is preserved. The remaining audio channel is filled by first sorting selected audio elements and available audio segments based on their durations. Then, starting with the element with the longest duration, each audio element is placed into the smallest audio channel segment that can fit the audio element.

In our implementation, audible information is converted to mp3 audio clips using the AT&T Natural Voices Text-to-Speech SDK [21], high-resolution document pages are stored in Flash format, and visual animations are rendered real-time using ActionScript 2.0. This composition allows us to use a standard Flash representation, avoid video compression artifacts, and keep the file small by performing visual rendering only during playback. Alternatively, an MMNail can be represented as a stand-alone video clip such as MPEG-4 [25]. The advantage of this type of representation is that any standard video player can be used to playback an MMNail visualization. It is important to note that MMNails can be stored as user data in the source document or in a separate file.

Even though the above representations are “user-friendly” in the sense of taking advantage of using standard playback software, results of our initial user study in [12] show that some flexibility in handling MMNail content is needed in order to allow easier adaptation to user, task, and application parameters. Flexibility can be addressed in various directions such as time duration of an MMNail, usage of audio and visual channels, content selection, etc., requiring a scalable representation. Scalability issues have been researched intensively in the area of still-image and video compression [24][25].

In this paper we present a way for representing MMNails in a time-scalable configuration. This allows users to view a few seconds or several minutes long MMNails without having to regenerate and store separate representations for a few seconds or a few minutes MMNails. Next, a modified optimization module that supports generating time-scalable MMNails is presented. A discussion on other scalable MMNail features is given in Section 7.

### ***3.3.1 Time Scalability***

In this section we expand the MMNail optimization module presented in Section 3.2.3 such that it allows time scalability, i.e. creation of MMNail visualizations for a set of  $N$  time durations  $T_1, T_2, \dots, T_N$  with  $T_N > \dots > T_2 > T_1$ . Our goal for scalability is to ensure that elements included in a shorter MMNail with duration  $T_i$  are included in any longer MMNail with duration  $T_n > T_i$ . This time scalability is achieved by

iteratively applying the two-stage approach from Section 3.2.3 to decreasing time durations  $T_N > \dots > T_2 > T_1$  as follows:

For iteration  $n=N, \dots, 1$ :

For the first stage,

$$\begin{aligned} & \text{maximize} && \sum_{e \in E_v^{(n)} \cup E_{av}^{(n)}} x_n(e) I(e) \\ & \text{subject to} && \sum_{e \in E_v^{(n)} \cup E_{av}^{(n)}} x_n(e) t(e) \leq T_n \quad (7) \\ & && x(e) \in \{0,1\}, \quad e \in E_v^{(n)} \cup E_{av}^{(n)}, \end{aligned}$$

where  $E_q^{(n)} = \begin{cases} \{e \in E_q^{(n+1)} | x_{n+1}^*(e) = 1\} & , \quad n=1, \dots, N-1 \\ E_q & , \quad n=N \end{cases}$ , for  $q \in \{v, av\}$ , and  $x_{n+1}^*$  is a solution of (7) in iteration  $n+1$ .

For the second stage,

$$\begin{aligned} & \text{maximize} && \sum_{e \in \hat{E}_a^{(n)}} x_n(e) I(e) \\ & \text{subject to} && \sum_{e \in \hat{E}_a^{(n)}} x_n(e) t(e) \leq \beta \hat{T}_n \quad (8) \\ & && x_n(e) \in \{0,1\}, \quad e \in \hat{E}_a^{(n)}, \end{aligned}$$

where  $\beta \in [0,1]$ ,  $\hat{T}_n$  is the total time duration available at iteration  $n$  after the first stage to be filled in the audio channel,  $\hat{E}_a^{(n)} = \begin{cases} \{e \in \hat{E}_a^{(n+1)} | x_{n+1}^{**}(e) = 1, t(e) \leq \gamma_n \hat{T}_n / R_n\} & , \quad n=1, \dots, N-1 \\ \{e \in E_a | t(e) \leq \gamma_N \hat{T}_N / R_N\} & , \quad n=N \end{cases}$ ,  $\gamma_n \in [0, R_n]$ ,  $R_n$  is the number of empty audio intervals after the first stage in iteration  $n$ , and  $x_{n+1}^{**}$  is a solution of (8) in iteration  $n+1$ .

A solution  $\{x_n^*, x_n^{**}\}$  describes contents of a set of time-scalable MMNails for time durations  $T_1, T_2, \dots, T_N$ , having the property that if a document element  $e$  is included in MMNail with duration  $T_i$ , it is also included in the MMNail with duration  $T_n > T_i$ .

#### 4. PLAYBACK INTERFACE

We implement an MMNail playback interface in Flash that is compatible with smart phones running the Pocket PC OS 2003 and Windows Mobile operating systems. First a general document browser interface that displays a conventional thumbnail of the first page of each document is shown in Figure 5.a. When a user selects a document thumbnail the MMNail visualization for the selected document is displayed in the interface given in Figure 5.b. The user has control over playback with a control bar, which he can use to start, stop, go backward and forward in the MMNail timeline. In addition to the control bar, the user can also use the keypad of the device to pause and play MMNails and to move to the beginning of the previous or next MMNail animation segment. This way, users have more control over the playback. They are able to skip less interesting parts and spend more time on more interesting sections.



Figure 5. Interface for (a) document browsing and (b) document viewing

#### 5. EXAMPLE MULTIMEDIA THUMBNAILS

In this section we present examples of automatically generated Multimedia Thumbnails for different time durations. The MMNail example shown in Figure 6 is generated by setting the time duration in the optimizer to be 40 seconds. The visual channel is taken by document pages, zooming into the title, some sections, and many figures. The audio channel is occupied by document title, section headings, and figure captions. Figure 7 shows an automatically generated MMNail of the same document with a 10 second duration constraint. Because this duration is quite short, there is only time to zoom into the title of the

document. For the rest of the pages in the document only the page thumbnails are selected to be displayed. The audio channel is mostly occupied by the title and the keywords. More MMNail examples can be downloaded at [26].

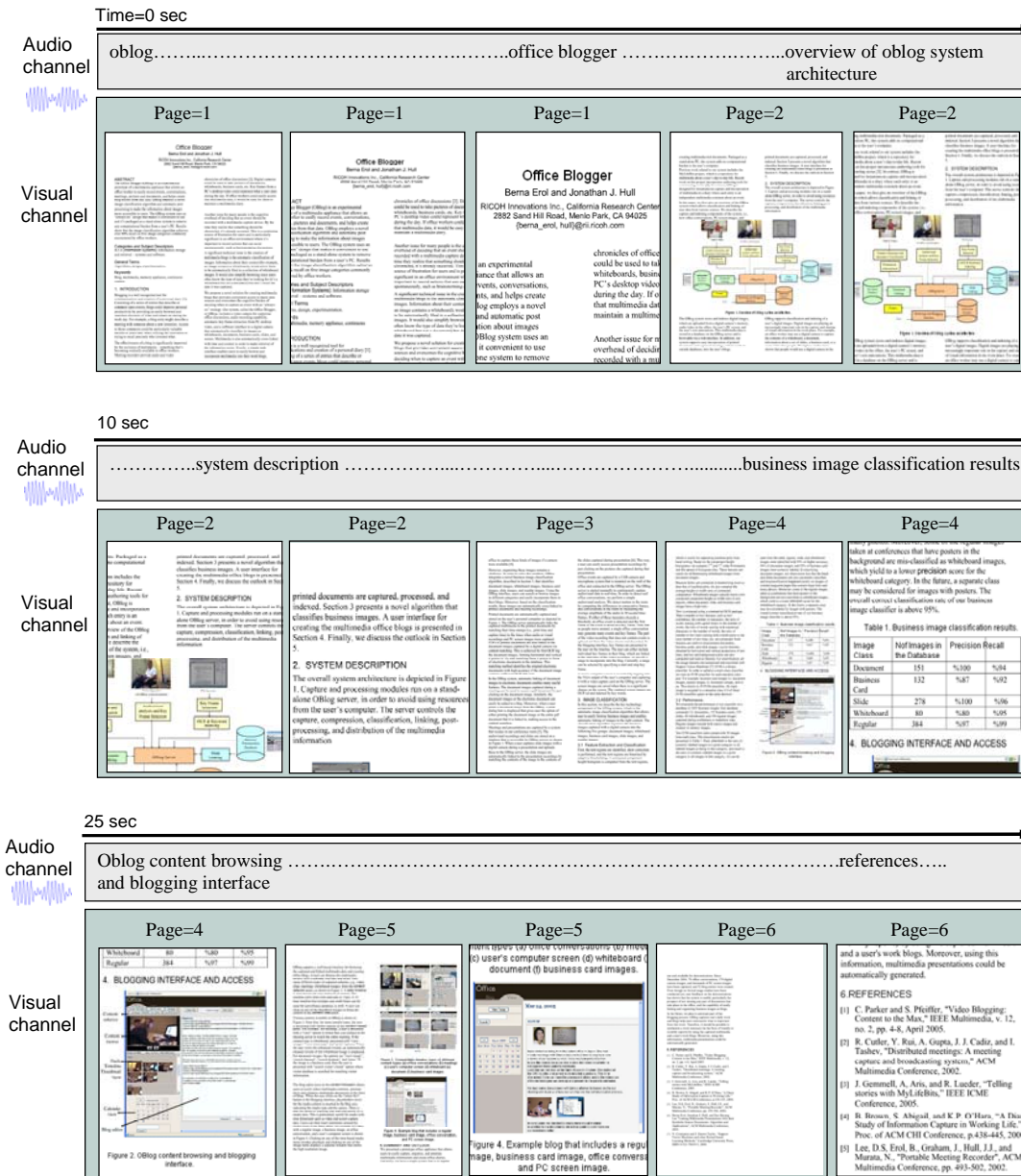
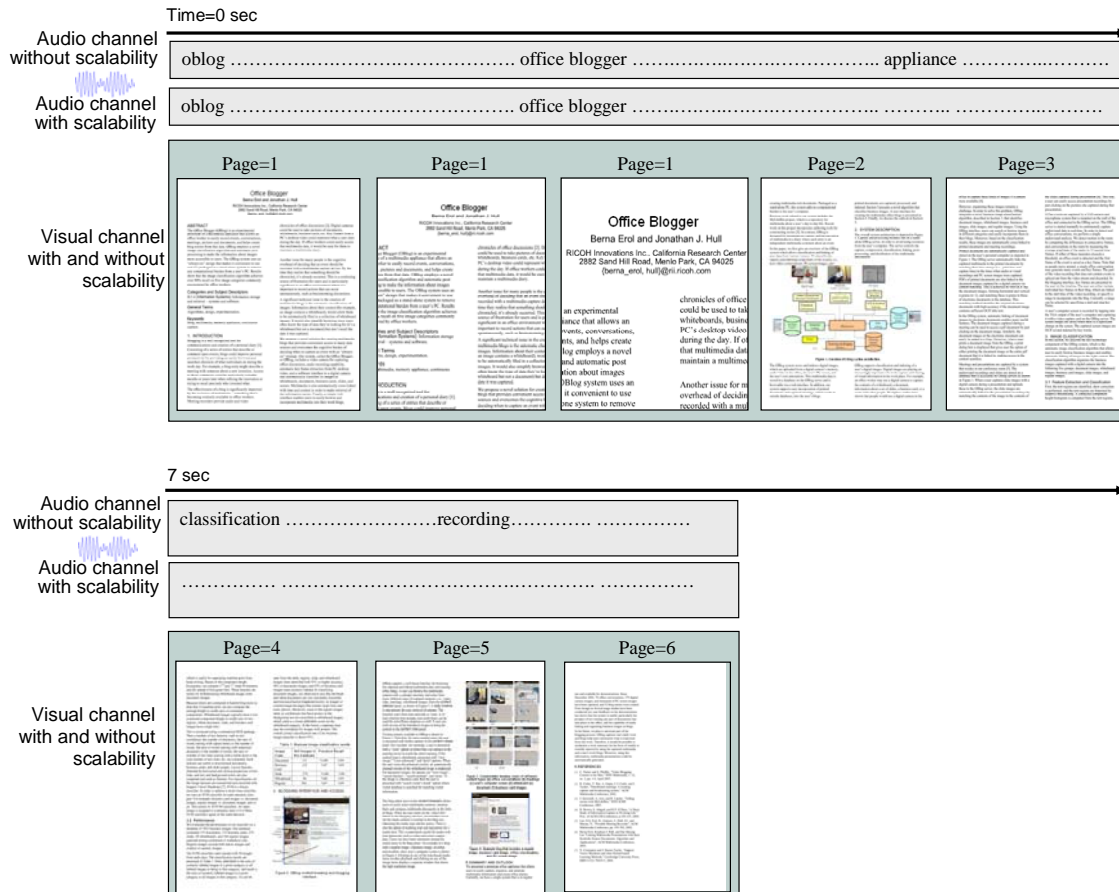


Figure 6. Example of a 40 second Multimedia Thumbnail.

For the same document and the same durations, MMNail examples are also generated with time scalability constraint as described in Section 3.3.1. In this case, the 40-second MMNail remains unchanged since presented content is selected from all document elements. On the other hand, the

contents of the 10-second MMNail slightly changes since the selection of document elements is based on the contents of the 40-second MMNail instead of the whole document. As can be seen from Figure 7, the visual channel of the time-scalable MMNail remains the same, but the audio channel does not contain some of the previously included document elements, such as most of the keywords.



**Figure 7. Example of a 10 second Multimedia Thumbnail.**

## 6. USER EVALUATION

We performed a user study in order to evaluate the effectiveness of MMNail document visualization compared to viewing a document in a PDF viewer on a small size viewing area. Our hypothesis is that MMNail viewing provides users with document comprehension superior to that of PDF viewing.

## 6.1 Evaluation Setup

A total of 16 users participated in our user study. Most users had technical backgrounds and even though all of them were very familiar with use of computers only six participants regularly used hand held devices for tasks other than making phone calls.

One of the two documents shown to the users was a technical paper on a photo browsing system and the other one was a magazine article on San Francisco's Angel Island. The first paper is 8 pages long and the second article is 4 pages long. Content included text, graphics, and pictures.

In our user study, a laptop computer is used to show both an MMNail viewer and PDF viewer to the users. A meeting recorder system [27] is used to capture the laptop screen and conversations with the users. A laptop is used instead of a handheld device because of its adequate speed and ease of screen capture. Only a limited viewing area (320×240 pixels) is used on the laptop computer and users' interactions with PDF viewer interface were limited to zoom-in, zoom-out, and pan, similar to typical interactions on handheld devices. For the MMNail visualization the interface described in Section 5 is used on the same laptop using the same viewing area.

Each user viewed one document in MMNail format and the other document, which has different content than the first one, in PDF format. We employed a rotation design by alternating the order of the viewing environments in order to keep the dependency on viewing order as small as possible. Users are given limited time to view and interact with each of these documents: 2 minutes 40 seconds for MMNail and PDF versions of the technical document and 2 minutes for both versions of the magazine article.

## 6.2 Task and Evaluation Measures

The users' overall task is to understand each document's content and to recognize visually the parts of the document after a limited-time browsing. In order to evaluate the user's comprehension of the content, after document browsing users were given a quiz that contained the following two parts:

1. Textual content quiz: Users are given a list of 20 topics, out of which 10 topics are covered in the document and 10 topics are not covered in the document. Topics included phrases such as “group annotation” and “mobile blogging” for the technical paper on photo editing. Users are asked to mark the topics that they think appeared in the document.
2. Visual content quiz: Users are given a list of 10 visual document elements, e.g., page thumbnails or figures, out of which 5 elements are extracted from the document and 5 do not belong to the document. These visualizations are presented in pairs to users, who are asked to pick which one of the visual representations is included in the document.

It is important to note that the document topics were extracted by a professional librarian who saw only the paper copies of the documents and did not know how the MMNail algorithm works. This removes the possibility of any bias in the preparation of quiz questions.

### 6.3 Comprehension Quiz Results

**Table 3. Summary of quiz scores when users view MMNails vs. PDF files.**

Viewing method/ <i>Document type</i>	Textual content		Visual content	
	Score average	Score $\sigma$	Score average	Score $\sigma$
MMNail/ <i>Technical document</i>	54	18	41	15
PDF viewer/ <i>Technical document</i>	39	15	33	23
MMNail/ <i>Magazine article</i>	71	21	45	7
PDF viewer/ <i>Magazine article</i>	58	15	48	9
MMNail/ <i>Average of both documents</i>	62	19	43	11
PDF viewer/ <i>Average of both documents</i>	49	15	41	16

Table 3 presents the results of quiz scoring for both MMNail and PDF viewing. Quiz scoring is done such that a wrong answer cancels a right answer. It can be seen from the table that users obtained an average score of 62% for the MMNail viewing vs. 49% for PDF browsing of documents when they were

asked questions about the textual content. Quiz scores were 43% for MMNail viewing and 41% for PDF viewing of documents when users are quizzed about the visual content of the documents. As can be seen from these results, viewing MMNail documents, users performed slightly better in visual quiz and significantly better in textual quiz compared to viewing PDF documents. Statistical analysis was performed using a paired sample t-test showing statistically significant difference between the textual content comprehension using MMNails and PDF viewer with a p-value of  $p = 0.05$ . For visual content comprehension no statistically significant difference can be reported.

## 6.4 Questionnaire Results

In addition to the quiz we gave each participant a questionnaire in order to receive feedback on the overall viewing experience of MMNails. The questionnaire results are presented in Table 4. In average, users rated MMNail viewing experience as a pleasant experience and indicated that MMNail viewing is better than PDF viewing. They also found the audio channel and automatic navigation useful. Variance of their scoring was highest in the case of usefulness of audio, which may be an indication of benefiting from audio channel being more of a personal choice.

**Table 4. User’s scoring of MMNail viewing experience on a scale of 1 to 10.**

Questionnaire topics	Score average	Score $\sigma$
<i>Overall viewing experience</i> [1: unpleasant, 10:pleasant]	8.25	1.43
<i>MMNail compared to PDF viewing</i> [1: MMNail worse than PDF, 10:MMNail much better than PDF]	8.24	1.86
<i>Usefulness of audio synthesis</i> [1:not useful, 10:very useful]	7.88	2.69
<i>Usefulness of automatic visual navigation</i> [1:not useful, 10:very useful]	8.47	2.07

## 6.5 Users’ Comments

In an exit questionnaire users were asked to comment on their MMNail viewing experiences. Generally users commented that document skimming with MMNails was less stressful compared to manually

skimming of PDF files. Because important document parts were already identified in the MMNail, users felt that they were viewing the parts of the document that were relevant. Users also thought audio was good for capturing the browsing person's attention, and it was easy to remember a topic when it was both seen and heard. As a shortcoming, users stated that sometimes they felt that they were not in control of the viewing experience or lost in the document. They suggested user interface components that would overcome these limitations as summarized in the next section.

## **7. SUMMARY AND OUTLOOK**

Even though we may expect that the computational power and memory of handheld mobile devices will improve significantly in the future, the small display size may be unavoidable and will likely continue to impose significant constraints on mobile applications. In this paper we presented a novel method for viewing formatted, high-resolution, and multi-page documents on handheld devices with small displays and limited navigational capability. An MMNail visualization can be seen as a conversion of two dimensional (spatial) document information to four dimensions (spatial+time+audio) for more efficient and effective information presentation. This conversion takes place in three stages where we first analyze the contents of the document and determine informative semantic document elements, then optimize the presentation of document elements given time, display, and application constraints, and finally compose selected document elements into a playable Multimedia Thumbnail.

In the paper we also addressed the issue of producing a navigation path through a document that is time scalable so that users can view MMNails of various durations without the need for re-generation. It is also possible to support other scalabilities such as computational scalability, e.g., where computation resources are sparse zooming and panning animations can be omitted, content scalability, e.g., where audio or visual content may be partially or completely omitted. Different scalability levels can be combined in *profiles* that users can specify based on target application, platform, location, etc. For example, when a person is driving, the profile for driving can be selected where the document info is

communicated mostly through audio, while they are not driving, a profile that gives more information through visual channel can be selected.

An explanatory user study is conducted to receive feedback and evaluate the effectiveness of the MMNail visualization. It was shown that MMNail viewing provided a statistically significant ( $p=0.05$ ) improvement in content understanding over PDF viewing of the documents. Moreover, users rated MMNail viewing experience compared to PDF viewing very favorably ( $>8/10$ ). The playback interface was found to be simple enough to get used to quickly yet powerful enough to achieve the task easily.

In light of the user feedback, many improvements are possible to MMNails. In order to prevent users' feeling of being lost in the document, moving to a different document page can be visualized more explicitly by adding an animation of page flipping. A small overview of the document pages can also be presented at the top or the bottom of the screen and the MMNail path can be highlighted as it plays. A timer can be displayed to give a user an idea about how much time left for viewing each section. Therefore, if the user is done reading before the allowed time, they can simply skip and move on to the next section. Moreover, MMNails can be used in combination with a document or web page viewer to obtain the benefits of both automatic navigation and manual navigation. The document viewer may show the most important document elements with the right alignment and zoom factor based on the display size and user's estimated navigation time, but user may take over the playback at anytime to perform manual navigation.

Applications of Multimedia Thumbnail span a wide spectrum. MMNails can be used to obtain a quick overview of the document before e-mailing or printing from a handheld device. The visualization can be extended to browse a large collection of documents. Moreover, when some document information, such as color and small text, is communicated through the audio channel, it can be useful for color-blind people or people with vision problems. MMNails can be employed at scanners to give the user an idea about the scan quality by zooming in to the most problematic areas of the scanned document, such as smallest text. In addition to document viewing, an MMNail visualization can be utilized for browsing

web pages and presentation slides on handheld devices. As can be seen from these examples, transformation of visual to audiovisual content via Multimedia Thumbnails enables many innovative applications and opens new research directions.

## **8. ACKNOWLEDGEMENTS**

The authors would like to thank Jonathan J. Hull, Martin Boliek, Michael Gormish, Peter E. Hart, and David G. Stork for many useful discussions, Kurt Piersol and Bradley Rhodes for their suggestions on the playback interface, Daniel Van Olst for his support with the implementation, librarian Rowan Fairgrove for preparing the quizzes, and employees of Ricoh Innovations for participating in the user study.

## **REFERENCES**

- [1] H. Lam, P. Baudisch, "Summary Thumbnails: Readable Overviews for Small Screen Web Browsers," Proceedings of the CHI, pp. 681-690, 2005.
- [2] J. Otterbacher, D. R. Radev, O. Kareem, "News to go: hierarchical text summarization for mobile devices," Proc. of ACM SIGIR, 589-596, 2006.
- [3] Y. Chen, X. Xie, W.Y. Ma, H. Zhang, "Adapting Web Pages for Small-Screen Devices," IEEE Internet Computing pp. 50-56, 2005.
- [4] X. Fan, X. Xie, W-Y. Ma, H-J. Zhang, "Visual Attention based Image Browsing on Mobile Devices," Proceedings of ICME, vol.1, pp. 53-56, Baltimore, MD, July 2003.
- [5] Z. Xie, H. Liu, W.-Y. Ma, H.-J. Zhang, "Browsing Large Pictures Under Limited Display Sizes," IEEE Transactions on Multimedia, vol. 8, no. 4, pp. 707-715, 2006.
- [6] F. Liu, M. Gleicher, "Video Retargeting: Automating Pan and Scan," Proceedings of International Conference of ACM Multimedia, pp. 241-250, 2006.
- [7] T.M. Breuel, W.C. Janssen, K. Popat, H.S. Baird, "Paper to PDA," Proceedings of the International Conference on Pattern Recognition, pp. 476-480, 2002.
- [8] J. Wang, S. Zhai, J. Canny, "Camera Phone Based Motion Sensing: Interaction Techniques, Applications and Performance Study," ACM UIST Conference, pp. 101-110, 2006.
- [9] C.C. Marshall and C. Ruotolo, "Reading-in-the-Small: a study of reading on small form factor devices," In Proc. of the Joint IEEE and ACM Conference on Digital Libraries," pp. 56-64, 2002.

- [10] G., Maderlechner. A. Schreyer, P. Suda, "Information extraction from document images using attention based layout segmentation," Proceedings of DLIA, pp. 216-219, 1999.
- [11] B. Erol, K. Berkner, S. Joshi, J.J. Hull "Computing a Multimedia Representation for Documents Given Time and Display Constraints," Proceedings of ICME 2006, pp. 2133-2136, 2006.
- [12] B. Erol, K. Berkner, S. Joshi, "Multimedia Thumbnails for Documents," Proceedings of ACM Multimedia Conference, pp. 231-240, Santa Barbara, CA, 2006.
- [13] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrison, and P. Pirolli. "Using Thumbnails to Search the Web." Proc. CHI 2001, Seattle, Apr. 2001, pp. 198-205, 2001.
- [14] K. Berkner, E. L. Schwartz, C. Marle, "SmartNails - Image and Display Dependent Thumbnails," Proceedings of SPIE, vol. 5296, pp. 53-65, San Jose, 2004.
- [15] Adobe, PDF access for visually impaired, available at <http://www.adobe.com/>.
- [16] P. Parente, "Audio enriched links: web page previews for blind users," Proc. of International ACM SIGACCESS Conference on Computers and Accessibility, no 77-78, pp. 2-8, 2004.
- [17] P. Roth, L. Petrucci, T. Pun, A. Assimacopoulos, "Auditory browser for blind and visually impaired users," Proceedings of CHI, pp. 218-219, 1999.
- [18] G. Salton, Automatic Text Processing, Addison-Wesley, 1989.
- [19] World Wide Web Consortium, Document Object Model Level 1 Specification, ISBN-10: 1583482547, Iuniverse Inc, 2000.
- [20] B. Erol, J.J. Hull, J. Graham, and D.S. Lee, "Prescient Paper: Multimedia Document Creation with Document Image Matching", IEEE International Conference on Pattern Recognition, 2004.
- [21] AT&T Natural Voices Speech SDK, <http://www.naturalvoices.att.com/>
- [22] R. Neelamani, K. Berkner, "Adaptive Representation of JPEG 2000 Images using Header-based Processing", Proceedings of IEEE ICIP, pp. 381-384, 2002.
- [23] R.L. Rivest, H.H. Cormen, C.E. Leiserson, Introduction to Algorithms, MIT Pres, MC-Graw-Hill, Cambridge Massachusetts, 1997.
- [24] ISO/IEC 15444-1:2000, Information Technology – JPEG2000 Image Coding System.
- [25] ISO/IEC 14496-10, "Information technology—Coding of audiovisual objects—Part 10: Advanced video coding," 2003.
- [26] Multimedia Thumbnail examples can be downloaded at <http://rii.ricoh.com/~berna/mmnails> .
- [27] Lee, D.S, Erol, B., Graham, J., Hull, J.J., and Murata, N., "Portable Meeting Recorder", ACM Multimedia Conference, pp. 493-502, 2002.