Mobile Web: Web Manipulation for Small Displays using Multi-level Hierarchy Page Segmentation

Aditya Gupta
ABV-IIITM
Gwalior
INDIA-474010
aditya_gupta@iiitm.ac.in

Anuj Kumar
ABV-IIITM
Gwalior
INDIA-474010
anuj_kumar@iiitm.ac.in

Mayank
ABV-IITM
Gwalior
INDIA-474010
mayank@iiitm.ac.in

V.N.Tripathi
ABV-IIITM
Gwalior
INDIA-474010
vntripathi@iiitm.ac.in

S.Tapaswi
ABV-IIITM
Gwalior
INDIA-474010
stapaswi@hotmail.com

ABSTRACT

Mobile devices have made the world a small place. They are also being used widely, to access the Web. However, because most available web pages are designed for desktop PC in mind, it is inconvenient to browse these large web pages on a mobile device with a small screen. With the enormous increase in mobile users since last decade, to such an extent that it even exceeds the number of desktop users in many developing countries, it becomes crucial to design systems for Web Manipulation, there by providing easier and greater accessibility and device independence. In this paper, we propose a new browsing technique to facilitate efficient web access and provide a better user experience on a device with rather small screen as compared to the usual desktop. It also proposes a novel technique to handle multimedia content, especially the images and video on a web page. In order to solve the problem of incomplete tags in HTML, an equivalent XML is generated for the desired webpage. For efficient browsing, the web page is represented as a multi level hierarchy, with the index of major sections on the first level and the sections indexed again in to sub-sections on the second level and so on, until we reach sub-sections that can be represented as sub-pages that fit in the small screen. We also propose an efficient algorithm for navigating through the sub-pages, thereby providing better user experience. The proposed technique provides better and efficient web browsing on mobile small displays up to user satisfaction.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval – I*nformation filtering, Query formulation, Retrieval Models, Search Process, Selection Process.*

General Terms

Performance, Design, Experimentation, Human Factors.

Keywords

Web browsing, small displays, proxy, TOC, Web page manipulation, multi level hierarchy, DOM

1. INTRODUCTION

It's said metaphorically "The World is going Mobile" and can be viewed on small displays. With the exponential growth of mobile communications along with the pervasive use of web in everyday tasks, there exists a strong need for mobile web access through various hand-held mobile devices. However, the current experiences of accessing HTML documents (designed for desktop PCs) from hand-held devices are very unpleasant ones because of various critical issues pertaining to the hand-held devices. For example, most of the HTML documents are designed for 15-inch or 17-inch standard PC display. However, most of the hand-held devices have 5 ~10 times smaller displays. Furthermore, although the computing power of hand-held devices is rapidly improving, many hand-held devices are not adequate yet to handle multimedia data types in a satisfactory fashion.

In recent years there has been significant and explosive growth of mobile devices, personal digital assistants (PDAs) and smart phones, to such an extent that it even exceeds the number of desktop users in many countries. Such devices provide greater portability, and are being used to leverage the capabilities of the Web and provide users with ubiquitous access to information than ever before [2]. Despite the exponential growth, their usage for accessing the Web today is still largely constraint by their small form factors such as small screen, less memory, low processing power and power consumption problems. PDAs impose limitations on client functionality because of comparatively limited user interface, memory and computing resources [14]. Because most of today's web page has been designed with the desktop computer in mind, and is often too large to fit into the small screen of a mobile device, Web browsing on such small devices becomes utterly cluttered. It requires the user to manually scroll the window (especially the ever teasing horizontal scrolling) to find the content of interest and position the window properly for reading information. This tedious and time-consuming process has largely limited the usefulness of these devices.

Within a decade of its inception, web documents have become an integral part of our daily lives. It has opened up a window of immeasurable opportunities for information exchange, sharing, connectivity, and interaction among diverse groups of people [3]. Weren't we excited when Google launched GoogleMobile, a website designed specially to cater to the needs of mobile web users. But developing new web pages altogether for the mobile doesn't seem to be a good and feasible idea with millions of websites at hand. Despite a huge number of mobile users, people have been avoiding the mobile Internet because of high costs and poor experiences of the technology. Thus, it is desired to develop novel techniques for web manipulation and developing systems for representing the existing web on the mobile efficiently, and providing an equivalent user experience. It would be possible for users to carry 'their' WWW with them wherever they go - like they wear their watch wherever they go. And accessing the Web would almost be as easy as checking the time [5]. It's like carrying the infinitely large web in small pockets.

The W3C Mobile Web Initiative is a crucial step in the same context. It eventually aims at device independence i.e. making the web accessible by any device under any circumstance and by all people. The motivation comes from the fact that it is highly likely that many people will use handheld-like computers to access web based materials in the coming future; handheld access is the next big thing for the Web. One can imagine all sorts of information that might be really helpful to the mobile user [11].

In this paper first we introduce a relatively new browsing technique. We also propose a novel technique for handling the multimedia content, very important in web pages today. We propose a new browsing technique to facilitate efficient web access and provide a better user experience on a device with rather small screen as compared to the usual desktop. In order to solve the problem of incomplete tags in HTML, an equivalent XML is generated for the desired webpage. This paper also discusses the various challenges posed by such devices in web accessibility.

The rest of the paper is organized as follows. Section 2 gives an overview of the related work in the field of mobile devices. Section 3 presents an overview of the proposed browsing method. Section 4 describes the software architecture used. In Section 5 our approach to the problem of handling a Web page and breaking it into corresponding sub-pages in an iterative manner has been presented. Section 6 gives Simulation Results. Section 7 gives the Conclusions and Future scope is discussed in Section 8.

2. RELATED WORK

In [6], methods were proposed for distilling web objects to reduce the consumption of network bandwidth and client computation. For web pages, the existing methods are mostly based on discarding format information. This method mainly focuses on reducing resource consumption. A technique where the web page is reformatted on the basis of page annotation is described in [8]. The re-authoring technique proposed in [1] required web pages to have sections and section headers. It also describes a technique where the Table of contents (TOC) for corresponding text blocks can be generated using the First Sentence Elision technique. A review on the various techniques of web manipulation such as transcoding, TOC, re-authoring, summarization etc. has been discussed in [3]. The various transcoding techniques for efficient web access using mobile devices are described in [7] in a more elaborative manner. It also gives techniques for handling various multimedia content present in the webpage.

A slicing* tree based approach for page adaptation has been proposed in [4]. It also represents the web page as a multi-level hierarchy rather than a two-level hierarchy. The proposed algorithms improve the processing efficiency significantly. In [5], a visionary side of bringing the web to PDA's and hand held devices is represented. The architecture for the browser is based on the concept of distributed clients. It also reflects the various challenges that such devices pose. The various critical issues involved in bringing the web to mobile devices such as the small screen size, navigation and structure of pages, and user input efforts are also presented in [9]. Methods are also described to improve mobile web accessibility using a case study. It also suggests certain design guidelines for the WAP service providers.

The motivation of handling multimedia content and the techniques that can be used are provided in [15][16]. In [15], a detailed description of wavelet transform is given and how it can be used for image compression is also specified. For handling video on a small screen device [16] describes efficient video compression technique, and compares the performance of video compression techniques such as H.263 and MPEG4. A lot of different wavelet transform techniques are explained and compared in [17] for image compression.

There are many techniques available for page analysis and breaking page in to sections as discussed in [2]. But it represents a web page into a two level hierarchy with a thumbnail representation at the top level for providing a global view and index to a set of sub-pages at the bottom level for detail information. But this technique lacks in providing a quick and better user experience because the user has to wait for sometime till the complete web page is processed and broken down in to logically related units that fit into the screen of a mobile device. On contrary, our paper presents a multi-level representation of a web page. For efficient browsing, the web page is represented as a multi level hierarchy, with the index of major sections on the first level and the sections indexed again in to sub-sections on the second level and so on, until we reach sub-sections that can be represented as sub-pages that fit in the small screen. Thus, when the user requests for a page the first Table of Contents (TOC) of the index page is provided to the user, during this time interval the rest of the sections are being processed simultaneously at the proxy server, without keeping the user waiting, thus trying to attain parallelism in processing and viewing. The technique is described in detail in the subsequent sections.

3. PROPOSED BROWSING TECHNIQUE

We aim at finding a better way to enable easy navigation and browsing of a large web page on a device with rather small screen. This is done by breaking the page in to meaningful sections. Every Webpage can be viewed as a collection of different small web pages structured together in a layout. This works beautifully on a Desktop where the screen size is 15-17 inches. Usages of mobile devices have become now an indispensable part of daily life of human beings, thus it has become critical to bring the Web to a user's pocket. We propose an effective web manipulation technique to break the web page in to sections in an iterative manner until we get meaningful units called sub-pages that can fit in the small screen.

The basic idea is to represent a Webpage as a multilevel hierarchy of various sub sections in an iterative manner as shown in Figure 1. The meaningful sub-pages are present at the leaf node. The overview is like a Table of Content (TOC) for the original web page. By clicking on the links, a user can easily navigate to the next level of TOC of the corresponding sub-section, which continues in an iterative manner until the user reaches a meaningful sub-page. Instead of using traditional text based TOC, equivalent thumbnails can also be used for representation, on which each block of semantically related content is represented with a different color [2].

An essential feature that this system can provide is not keeping the user waiting for the processing to finish. For example, when the user sends a request for www.orkut.com, the web page is processed at the proxy server and the TOC for the main page is provided to the user and simultaneously processing takes place at the backend to break those sub-sections further, to derive the meaningful sub pages that fit in the small screen. We also present an efficient algorithm to deal with efficient browsing of the sub pages.

The major contributions of the paper are:

- A novel idea of enabling easy browsing of a large web page on a small screen device through multi level hierarchy representation.
- A novel idea to deal with the multimedia content, especially images.
- The problem of incomplete tags in HTML is being dealt by converting it in to an equivalent XML document and using a Document Object Model (DOM) parser thereafter.
- An efficient algorithm for better sub-page navigation and handling images so that they fit in the small screen.

3.1 Critical issues with mobile devices

Mobile devices are small not only in terms of screen size, but they also have minimal computing power. Apart from the weak computing power, PDAs have several other restrictions that make it difficult to present multimedia information. These are:

- small display
- limited color depth
- small memory
- weak computing power
- poor digital audio interface

A lot of research is still being conducted on developing efficient techniques to provide a better user experience despite all these shortcomings. The common resolution of a PC display is 800 x 600 pixel or more with at least 256 colors. For example the Apple Newton Message Pad only has a monochrome display with a resolution of 320 x 240 pixels [14]. But the kind of portability these devices provide encourages a lot of research to develop systems that can bring the web to these devices keeping in mind all these limitations. Thus, the software architecture should be such which demands minimal processing requirements from the mobile device. For this purpose we used Proxy-Server architecture. It is described in the subsequent section.

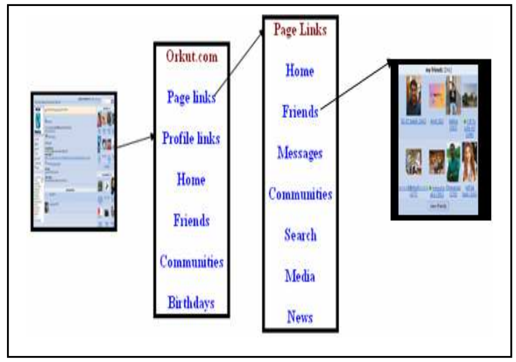

Figure 1. Organizing a web page into a multi level hierarchy with Table of Contents (TOC) to provide a global view.

4. SOFTWARE ARCHITECTURE

For efficient processing and better results we used three-tier architecture. Due to the small storage capabilities of PDAs it is not possible to download long documents into its memory. Thus, we developed a proxy HTTP server agent which allows a mobile user to specify his or her viewing preferences, based on the network connection and available resources, and performs the desired requests accordingly. The software architecture proposed introduces a proxy server on each Mobile Support Station (MSS) which accepts and stores the preferences for each of its mobile hosts, acts as the server to the mobile host, and as a client to the Web Server.

A proxy server is commonly used in local area networks as firewalls or accelerators and is more powerful than PDAs. Thus, using a proxy server, web page manipulation can be done more efficiently. This can also support other mobile phones and smart phones, having less processing power as compared to PDAs.

All communication between the mobile and the Web Server is directed through the proxy server as depicted in Figure 2. When a user wishes to view a webpage on his mobile device to access certain information, he types the URL in the mobile web browser. The mobile device sends a request for the above page to a proxy server using a wireless connection. When the proxy receives a request from the mobile, it fetches the webpage from the web server, and produces an equivalent XML document. The XML document is then parsed using a DOM parser, which creates a DOM tree for the desired webpage. It segments the original page in to sections, sub-sections and sub-pages in an iterative manner using a multi-level hierarchy representation of the webpage. The sub pages lie at the leaf nodes of the tree. Thus, when a user clicks on a desired link in the TOC he is taken to the subsequent sub-section (TOC) or the sub-page. The proxy server also runs an efficient algorithm for sub-page navigation. The most frequently accessed WebPages can also be stored in a cache at the proxy server in order to improve the performance.

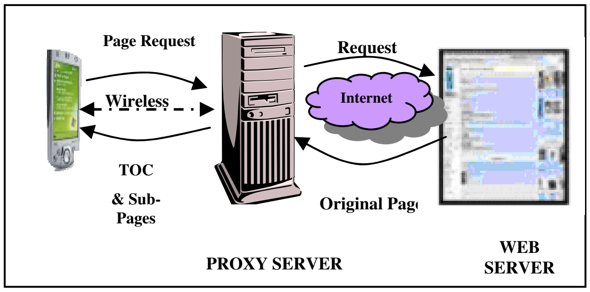

Figure 2. Proxy Server architecture for page manipulation.

5. WEB MANIPULATION : ENABLING MOBILE WEB ACCESS

In this section we discuss our approach to solve the problem of efficient web browsing on mobile devices. Despite the exponential growth, their usage for accessing the Web today is still largely constrained by their small form factors [9]. The critical issues involved with mobile devices are small screen, less memory, low processing power and power consumption problems. In order to deal with the low memory issues proxy server architecture has been used, which also enhances the processing capabilities thereby providing better user experience. For proper browsing the Web page is segmented in to various sub-pages that fit the screen size using a multi level hierarchy representation of the webpage. The Web manipulation technique has four main stages:

- Generating XML documents of the desired web page for easy traversal.
- Traversing XML documents to find out relevant sub-sections or sub-pages and representing them as a multi-level hierarchy (generating DOM tree).
- Creating sub-pages and maintaining the web page style.
- Efficient navigation through the sub-pages (to facilitate minimum scrolling) with the help of the proposed sub-page navigation algorithm.

5.1 HTML to XML conversion

A lot of WebPages contain incomplete HTML tags, i.e. certain HTML tags don't have the closing delimiters. Thus, during parsing it becomes difficult to identify the appropriate sub-sections for some WebPages, thus producing inefficient results. To solve this problem, we first generated an equivalent XML document for the desired Webpage. The WebPages are first fetched from the web server at the proxy server and the corresponding XML documents are generated. The XML documents for frequently accessed WebPages can also be stored in a cache for fast retrieval performance. The cache is updated after a definitive time period to check for updates in WebPages, if any.

5.2 Iterative page splitting: Segmentation in to sub-sections and sub-pages

The generated XML documents contain start and end delimiters for each sub-section. The WebPages are represented as a multi level hierarchy by generating a DOM tree as presented in Figure 3. The XML equivalents of the Webpage is parsed using a DOM (Document Object Model) parser and is segmented in to various subsections in an iterative manner. The sub-sections are further split in to sub-sections if its size is quite large as compared to the screen size, until we get meaningful sub-pages at the leaf nodes.

The generated DOM tree can be parsed now to generate the Table of Contents for the corresponding sub-sections and generating the sub-pages. The way user receives the results of this parsing is shown in Figure 1. The TOC and Sub-pages are generated at the proxy server, there by providing a quick response.

In our approach, we try to achieve parallelism in the processing and representation of results. The page is processed and as soon as we segment the webpage in to sub-sections on one level the corresponding TOC are generated and the results are send to the user, while the user is browsing, the sub-sections are further segmented in to smaller sub-sections and sub-pages in the second level. This approach provides a better user experience as the user does not have to wait until the complete processing of the web page takes place. The frequently accessed web pages can be segmented and stored in cache already, to enhance performance and provide a quicker response to the user.

HTML standard permits the author to use Cascading Style Sheet (CSS) [10] and style inheritance to specify content style. CSS defines the style of a tag, a tag class or tag instance. Extracting content from a block of HTML file may lose some style information because the style information from CSS and inheritance is not located together with the content. In order to keep the original appearance of a block, we need to retrieve its style information. For the CSS case, since CSS is usually specified in *<HEADER>* using *<STYLE>* or *<LINK>*, we could copy the *<HEADER>* section of a web page into each generated sub-page [2]. This technique described in [2] provides an opportunity to maintain the look and feel of the web page in the sub-pages as well. So, while generating the sub-pages the header information of the original page is also copied so as to maintain the style of the web page. Thus, the sub-pages look like mere sub-sections of a web page.

5.3 Representing sub-pages and handling multimedia content

There are several interesting questions that are critical to this problem domain. Especially, how to divide a large page into several smaller ones so that each of the sub-pages fits into the screen? What parts of a page are "good" to be put into one sub-page ? What is the right number of sub-pages in one page,

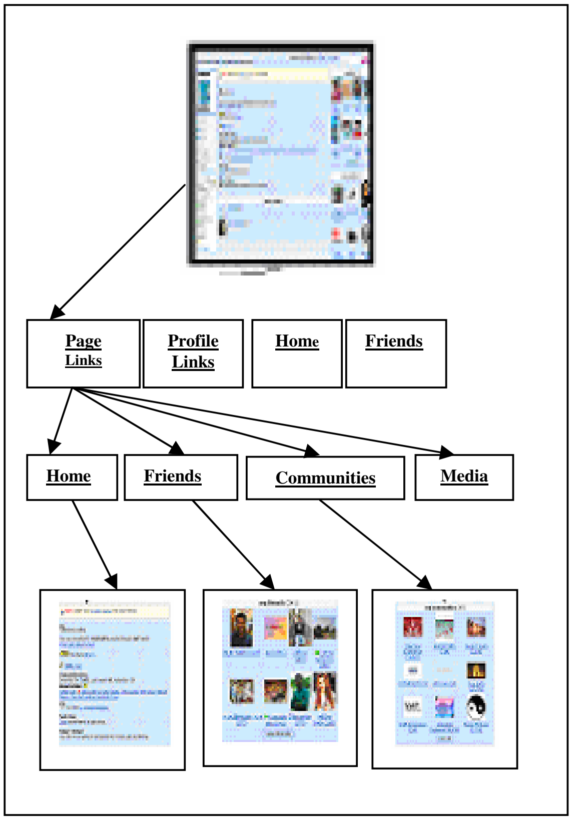

Figure 3. Proposed approach for web page manipulation

since too many sub-pages in one page is hard for the users to browse, and too few may result in too many pen-taps during navigation? [4]. We have developed an efficient algorithm to answer this question. Since, the screen size of a mobile device is quite small we cannot go about segmenting the page in to very small sub-pages, which can be a big trade off. Thus after limiting the tree generation process, and generating the sub-pages we designed an efficient algorithm for the representation of the sub-pages and the multimedia content, especially the images.

Algorithm: Sub-page-navigation

Input:

S is a sub-page;
T is threshold, in accordance with the screen size;
M_w is the width of mobile screen;
M_h is the height of mobile screen;

Output: Easier browsing through sub-pages

1: **begin**
2: Analyze/traverse S to count the no. of lines
3: NL = number of lines in the sub-page
4: **if** S contains Image **then**
{
I_w= width of image;
I_h = Height of image;
If $I_w > M_w$ **or** $I_h > M_h$ **then**
{
Compress the Image to the screen size and reduce color;
}
}
5: **if** NL > T **then**
{
if(NL > 2T)
{
Summarize(S);
provide link for full page view;
}
else
{
Display S till T lines, and provide NEXT link for further reading;
Sub-page-navigation(S-T, T, M_w , M_h) ;
}
}
9: **else**
10: Display (S);
11: **end**

This algorithm provides an efficient way to navigate through the sub-pages and to handle images as well. The sub-pages that were generated during the iterative tree generation process might contain text or images that might not fit the small screen size. The sub pages are small but in order to avoid any scrolling, an efficient navigation algorithm is proposed.

The sub-page is traversed and the number of lines in the page is counted. This is then compared to a threshold value T, set from observation or depending on the mobile screen size. In our experimentation we assumed the value of T as 7. If the number of lines is greater than 2T, the user is provided a summary of the sub-page, thus giving the user an overview of the sub-page. For summarization natural language technique described in [13] were used. If the user is genuinely interested he can navigate through the sub-page by clicking on the full page view link. If the number of lines is more than the threshold T but less than 2T, the sub-page is further segmented in to T number of lines and kept for display with a NEXT link for further navigation; the remaining sub-page is than passed again as input to the algorithm, the algorithm being recursive in nature. The NEXT and PREVIOUS links help the user to move ahead or read back.

If the sub-page contains any image, the image dimensions are also checked with the size of the screen. If the image doesn't fit in with the screen size, the image is compressed appropriately. For image compression we used Sliding-Window based wavelet coding. Wavelet image compression offers several advantages over existing methods such as the DCT adopted in the JPEG standard, and has now been adopted by MPEG-4 [15]. Images which contain text or numbers can only be reduced by a small amount before their contents become illegible [12]. The image compression is done at the proxy server while generating the sub-pages and thus the compressed images are embedded in the sub-page or send as a different sub-page in itself depending on the size of the image. In the next sub-section we describe in more detail how to handle the multimedia content in web pages on a mobile device which had minimal graphics and video support for the user.

5.4 Handling image and video

When the proxy finds an image tag in the sub-pages generated from the original page received from the server, it reads the URI (Uniform Resource Identifier) of the image file to be fetched. It checks the content-type and content-length information received from the server. If the image is small enough to be handled on the mobile device, the image is sent to the mobile unmodified. But if the image is larger than what can be handled by the device, it is reduced in size or color as per the mobile requirements. The image is scaled down in size, or the number of colors is reduced, or both without sacrificing semantics [7]. To reduce the image size and maintain the image quality, Sliding-Window based wavelet coding is employed. This provides a great opportunity of representing the multimedia content especially images which forms an important part of web pages these days, on the small screen device. The image compression technique is applied when the sub-pages are generated (if the sub-pages contain an image larger than the screen size). Thus while generating the sub-pages the images are compressed to proper display size to fit in the small screen. All the processing is done on the proxy server and the compressed images are send to the user either as a part of the sub-pages or as different sub-pages altogether. This gives us an efficient and fast method for handling images present in the desired web page.

Dealing with videos on small screen devices is rather complex. This problem is more serious seeing the limited bandwidth available for transmission of video content on mobile devices than required. A video consists of segments, scenes and shots. The shots comprise of frames having similar properties. The fuzzy clustering techniques are used since frames can belong to the clusters to different degrees (membership values) [7]. Developing efficient video compression techniques catering to the needs of mobile devices also forms a part of our future study.

5.5 Navigating the web pages

The sub-pages are generated and we have also designed an efficient algorithm for navigating through the sub-pages. A lot of WebPages these days contain anchors, i.e. the links to the content in the same page itself. Now, while breaking the page into sub-pages this might happen that the link (say at the top of the page) and the content (say at the bottom of the page) are separated into two different sub-sections or sub-pages. In this case we need to convert the anchor tags in to corresponding hyperlinks that link to the corresponding sub-page.

In order to avoid unnecessary navigation in sub-pages, another novel idea is to use text summarization. This helps by providing a summary of the sub-page initially, thus giving the user an overview. He can navigate through the sub-page only if he is genuinely interested. In this approach, the content is not simply separated into separate layers; the textual part of the content is summarized using natural language techniques [13].

6. SIMULATION RESULTS

The system was successfully implemented and demonstrated on the Nokia Mobile simulator with few test cases. The results were satisfactory for the web page segmentation and the small images in a web-page. The Simulation Results are shown in Figure 4 (a), Figure 4(b), Figure 4 (c), Figure 4(d) and Figure 4(e).

7. CONCLUSIONS

The number of mobile users is increasing exponentially; to such an extent that it even exceeds the number of desktop users in many countries. The number of Internet users has also doubled since last few years. Thus, looking in to the possibility of an amalgamation of the two is important. Mobile web holds a lot of promise, by providing increased mobility and encouraging device independence. But the web is enormous, or to say *"it's infinite"* thus we cannot go about building websites specifically for mobile devices. Efficient web manipulation techniques can be used to adapt the existing websites to the small screen of a mobile device.

In this paper we proposed an efficient web manipulation technique by using a multi level hierarchy representation of a webpage. We also aimed at increasing the accuracy of the results by converting the HTML in to an equivalent XML document, so that the problem of missing tags is solved, and segmentation of page becomes easier. An efficient algorithm is also proposed for better sub-page navigation and browsing. We also emphasized on techniques of managing the multimedia content specially images, video. Such a new browsing method overcomes the limitations of a mobile device with a small screen and makes them truly useful for information access.

The very idea of bringing the "***infinite web to our pockets"*** and accessing information whenever and wherever we wish to, is enticing in itself. The future according to Bill Gates is called *"Web Lifestyle"*. It has three pillars: computers, interactive cable services, and wireless information services (mobile devices). Thus, mobile devices pose a great promise in enhancing information accessibility and availability, using efficient web manipulation techniques. So, let's carry the web in our pockets.

8. FUTURE SCOPE

We used the Table of Contents technique for representing the links to various sub-sections. In the future study an equivalent thumbnail representation can be aimed at. Developing efficient video compression techniques catering to the needs of mobile devices is also desired as handling multimedia content is crucial, as it forms an important part of web interaction these days. Specific systems can be developed for people with disabilities by providing multi modal input for web access. This deals with developing speech to text and text to speech systems, so that the user can speak the URL and the system can produce the desired results. More research is required in building user specific systems, where a particular user's profile can be created and stored on a proxy server, also maintaining a cache of the WebPages frequently accessed by the user. This would provide user specific web accessibility. Specific applications can be developed with "*perceptive context*", such as the presence, attention, and activity of users, as well as provide conversational cues such as face pose and expression. Web manipulation techniques can also be clubbed with theory of captology or persuasive technology and persuade the users for mobile web access.

9. REFERENCES

[1] Bickmore, T.W. and Schilit, B.N. Digestor: "*Device-independent Access to the World Wide Web*", Proc. of the 6th WWW Conference, 1997, pp655-663.

[2] Chen, Y., Ma, W., Zhang, H., "*Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices*", the Twelfth International World Wide Web Conference (2003).

[3] Alam, H., Rahman, F., "*Web Document Manipulation for Small Screen Devices: A Review*", Second International Workshop on Web Document Analysis (WDA2003), Edinburgh, August 3 (2003)

[4] Xiao, X., Luo, Q., Hong, D., Fu, H., "*Slicing*-Tree Based Web Page Transformation for Small Dislplays*", ACM Fourteenth Conference on Information and Knowledge Management (CIKM 2005), Bremen, Germany, 2005.

[5] Gessler, S. and Kotulla, A. "*PDAs as mobile WWW browsers*", in Proceedings of 2nd International WWW Conference, 53-59, 1994.

[6] Fox, A., Gribble, S.D., et al. , "*Adapting to Network and Client Variation Using Infrastructural Proxies: Lessons and Perspectives*". IEEE Personal Communication, V5, I4, 1998, pp10-19.

[7] Bharadvaj, H., Joshi A., Auephanwiriyakul S., "*An Active Transcoding Proxy to Support Mobile Web Access*", IEEE (1998)

[8] Hori, M., Kondoh, G., Ono, K., Hirose, S. and Singhal, S., "*Annotation-Based Web Content Transcoding*". Proc. of WWW-9, Amsterdam, Holland, May 2000.

[9] Buchanan, G., Farrant, S., Jones, M., Thimbleby, H., Marsden, G., Pazzani, M., "*Improving Mobile Internet Usability*", the Tenth International World Wide Web Conference, May 1-5, Hong Kong (2003).

[10] *W3C, "Cascading Style Sheets"*. http://www.w3.org/Style/CSS/

[11] Jones, M., Marsden, G., Mohd-Nasir, N., Boone, K., "*Improving Web Interaction on Small Displays*", The Eighth International World Wide Web Conference (1999)

[12] Bickmore, T., Girgensohn, A., Sullivan W. J., "*Web Page Filtering and Re-Authoring for Mobile Users*", Computer. Journal, 42(6), 1999

[13] Berger, A., Mittal, V., "*OCELOT: A System for Summarizing Web Pages*.", Proceedings of the 23rd Annual Conference on Research and Development in Information Retrieval (ACMSIGIR), pp. 144-151, 2000.

[14] Lauff, M., Gellersen, H.W., "*Multimedia Client Implementation on Personal Digital Assistants*", Proc. of European Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS'97), Darmstadt, Sept. 1997, Lecture Notes in Computer Science, Springer-Verlag.

[15] M. Ravasi, M. Mattavelli, D. J. Mlynek, A. Buttar , S. Soudagar , "*Wavelet image compression for mobile/portable Applications*", ICCE International Conference on Consumer Electronics, 1999.

[16] E. Jackson, R Peplow, "*Video Compression System for Mobile Devices,*" South African Telecommunications Networking and Applications Conference, George, 2003

[17] Xiao, P., "*Image Compression by wavelet transform – A Thesis*", http://etd-submit.etsu.edu/etd/theses/available/etd-0711101-121206/unrestricted/xiaop0720.pdf , August 2001, pp. 29-40

Figure 4 (a) Mobile Device

User selects the web Application from his Menu Bar

User can select the 'Go to Address' option to request desired page

Figure 4 (b) User Enters Desired URL

Web page request is sent to the proxy server through the mobile Server. TOC is generated and sub-pages that can be easily browsed on the small screen.

Web Page already downloaded on the server is being called using the local host server

Figure 4 (c) The desired Web-page is displayed on the mobile device.

Figure 4 (d) The user is provided the table of contents of the web-page, where the TOC links to the sub-pages.

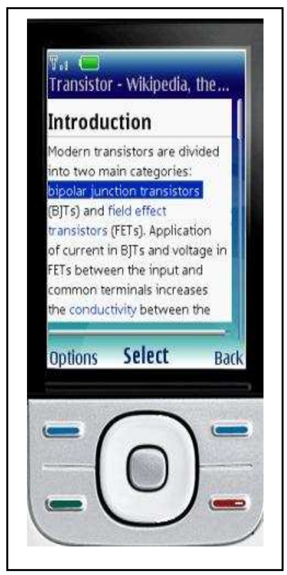

Figure 4 (e) The sub-pages are browsed by clicking on the links (for ex. Introduction). The sub- pages are displayed along with the images that are small in size and manageable.