Web Page Segmentation for Small Screen Devices Using Tag Path Clustering Approach

Ms. S.Aruljothi, Mrs. S. Sivaranjani, Dr.S.Sivakumari
Department of CSE,
Avinashilingam University for Women,
Coimbatore, India.
jothisundararaj@gmail.com

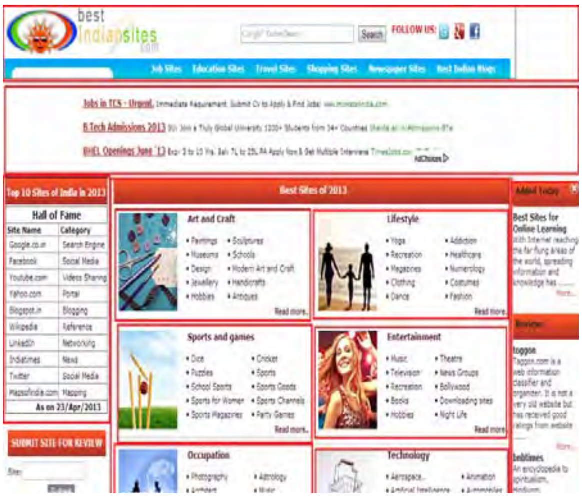
***Abstract*—The web pages breathing these days are developed to be displayed on a Desktop PCs and so viewing them on mobile web browsers is extremely tough. Since mobile devices have restricted resources, small screen device users need to scroll down and across the complicated sites persistently.**

To address the problem of resource limitation of small screen devices, a unique methodology of web page segmentation with tag path clustering is proposed, that reduces the memory space demand of the small hand-held devices. For segmenting web pages, both reappearance key patterns detection technique and page layout information are used to provide better segmentation accuracy.

Keywords- DOM(Document Object Model), key patterns, Tag path clustering, web page segmentation.

I. INTRODUCTION

Handheld devices equipped with browser and a wireless communication will ready to utilize web at anytime and at anywhere. Recently, the demand is high for smart phone, portable hand-held computers and personal digital assistants. In spite of that, mobile devices have small display, limited memory and processing capabilities, low bandwidth, and a few constraints because of equipment limitation exist. The present web page is developed to be displayed on a PC and it is thus not easy to display a whole page on the small screen devices. Web page segmentation is a crucial technology for internet driven applications like search engines and browser on mobile devices. Web page segmentation is the method of dividing a web page into logical blocks in a manner that every block contains unique informative content [9]. Distinguishing segments will be terribly helpful for various fields like sites will be properly displayed on small screen devices, search engines will use such information to provide a much better search results etc,. This system will be used as a preprocessing tool for data extraction, classifying the segmented blocks into informative blocks that contain the page's core contents and noise blocks that contain irrelevant information like menus, advertisement, or copyright statements [18]. Web page segmentation enhance the performance of knowledge extraction by ignoring noise blocks and focusing solely on informative content blocks and additionally facilitating the display of useful information on mobile devices with little screens. [10], [11], [13], [14]. Figure 1 show an example of page segmentation where the segmented blocks are marked with boxes.



Hall of Fame	
Site Name	Category
Google.co.in	Search Engine
Facebook	Social Media
Youtube.com	Videos Sharing
Yahoo.com	Portal
Blogspot.in	Blogging
Wikipedia	Reference
LinkedIn	Networking
Indiatimes	News
Twitter	Social Media
Mapsofindia.com	Mapping
As on 23/Apr/2013	

Figure 1 Sample web page segmentation

Early research on web page segmentation are based on the DOM tree structure of a web page or by using visual information like font size, font color, background color, spaces between paragraph etc., in a web page. Despite some successes, this vision based technique depends heavily on heuristic rules that make it tough to address a dynamic internet setting wherever the structure and visual information of sites are typically dynamical.

The DOM tree structure stores the complete content within the memory and it has to traverse the complete tree whenever once it checks for key pattern and pattern matching. Since mobile devices are memory constrained devices it is troublesome to store the entire content within the memory.

To resolve these issues, this paper proposes a new tag path clustering based approach to web page segmentation within which web page is pictured as a binary visual signal rather than traditional DOM tree structure. For segmenting web pages, both reappearance key pattern detection technique and web page layout information are used [1], [2].

II. RELATED WORK

L. Wu et al [3] projected a block gathering based page segmentation algorithm based on mobile internet features combined with human perception. The algorithm principally used tags which define layout of web page like <div> and <table>. Nyein, S.S [4] introduces an algorithm that extract the main content from the web documents using Content Structure Tree (CST) that is made from the DOM tree and additionally introduces cosine similarity to judge the more important and less important components of the CST tree. J. Kang et al [5] proposed way of webpage segmentation by using repetitive tag patterns within the DOM tree structure of a page. P. Mitra et al [6] outlined web page blocks and devise an algorithm to partition HTML page into constitute web page blocks. They proposed two tools one to section the web pages and another to separate the primary contents. C. Kohlschutter et al [7] conferred the block fusion algorithm to section HTML pages that utilizes text density as a measure to spot the individual text segments of presented page. Chen et al [8] proposed a function based object (FBO) model for content understanding and adaptation. A function type was defined to every object that helps to create a data structure for the page. But is laborious to define the function and grouping rule accurately therefore making tree construction method inflexible. Cai et al [12] discussed how to use web page segmentation to enhance web information retrieval and compared four methods extensively namely fixed length page segmentation, DOM based page segmentation, vision based page segmentation and a combined technique that integrates both the vision based and fixed length properties. Yi, Liu and Li [15] proposed a improvement based on the analysis of both layout and actual content of the online page and used a data structure known as style tree to capture the common presentation style and actual contents. Y. Hwang et al [16] proposed two general solutions namely manual and automatic re authoring by considering the importance of both web components and web page structure. A. Blackwell et al [17] used a thumbnail overview to facilitate the data search process. The thumbnails were annotated to indicate the location of query terms. Lin et al [18] only considered <TABLE> tag and proposed an algorithm which divides the web page into blocks and identify content blocks. It determines the redundant block using entropy of keywords in a block. R. Kumar et al [19] mentioned the heuristic and rule based problem of web page segmentation and proposed a combinatorial approach to unravel this problem by considering two variants, one supported correlation cluster and another supported energy minimizing graph cuts. Y. yang [20] conferred a visual cue based approach to extraction of semantic structure of HTML documents. This approach first measure visual similarity of HTML content objects and applies a pattern detection algorithm to detect frequent pattern and forms blocks according to those patterns.

To sum up, traditional web page segmentation strategies relied on heuristic rule generated by exploiting the stratified features of structural tags and visual information inherent in a page. However, in a dynamic internet setting where the structure of a web page is usually dynamical with the introduction of latest featured tags, the heuristic rules cannot analyze the web pages properly. This needs that we tend to maintain and update the rules once a new standard is declared and the structure of web sites is modified. Additionally the DOM tree based web page segmentation methods store the complete document in memory. Since mobile devices are memory constrained devices, it is terribly difficult to store the entire document within the memory.

In contrast, new web page segmentation is proposed that used tag path clustering based approach before segmentation method that reduces the memory space demand of the small handheld devices. The web page is then segmented either by reappearance based segmentation scheme that recognize key pattern from the web page or by page layout information like <TABLE>, <DIV> and <FRAME> tags.

III. PROPOSED METHOD

The data flow diagram of proposed methodology is shown in figure 2. First the HTML tags are extracted from the web page using HTML parser. In the second section, the HTML tags are pictures as binary visual signals and clustering is performed using spectral clustering algorithm based on similarity matrix calculated between the visual signals. The third section is to detect the key pattern by using reappearance key patterns detection technique. Pattern matching algorithm is applied to checks for key pattern. If the key pattern contains more number of reappearance tags, it will generate implicit node to segment nested blocks. Else the web page is segmented based on page layout information. In the fourth section, the hyperlinks of the divided blocks are displayed on mobile devices and then users choose their own interested hyperlink. The pseudo code of the proposed method is described in algorithm.

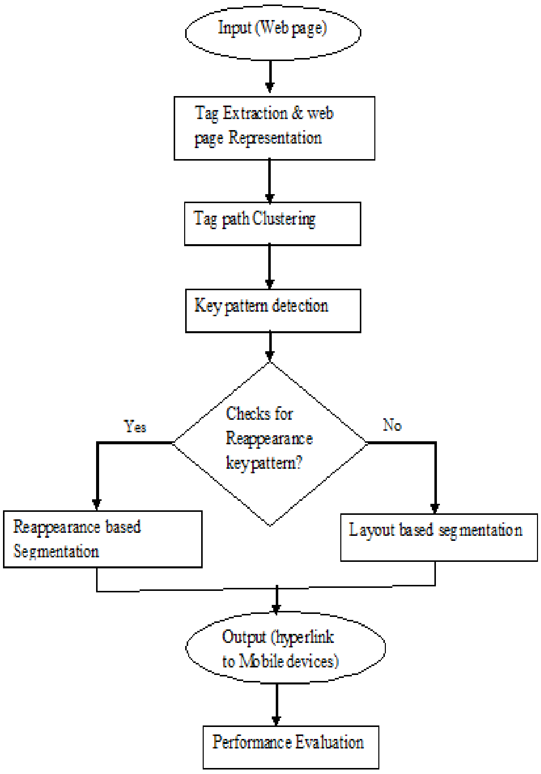

Input (Web page)
Tag Extraction & web page Representation
Tag path Clustering
Key pattern detection
Checks for Reappearance key pattern?
Yes
No
Reappearance based Segmentation
Layout based segmentation
Output (hyperlink to Mobile devices)
Performance Evaluation

Figure 2 Data flow diagram

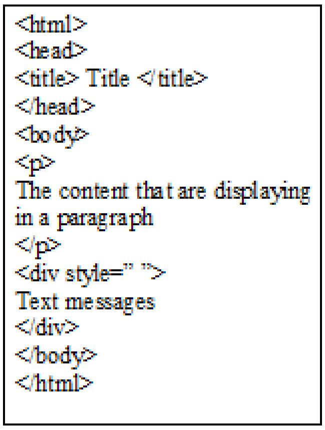
Algorithm web page segmentation

```
Input: Web page
Output: segmented Blocks
Extract HTML tag from the web page
Find tag path for HTML tags
Function tag path clustering (tag list)
{
//Represent tags as Binary visual signals
Extract Visual signals from tag path
Compute similarity Measure between visual signals
Form pair wise similarity matrix
Apply spectral clustering algorithm
}
    For each cluster
{
        Function Reappearance pattern detection
        {
        Checks for key pattern
            If found key pattern
            {
            Use reappearance based segmentation algorithm
            }
            Else
            {
            Use layout based segmentation algorithm
            }

        Generate hyperlink for the segmented blocks
        }
}
Display the hyperlink on small screen devices.
```

A. Extraction of web elements

The web pages are taken from any web sites and keep in a folder. The tags are extracted from the HTML source code using HTML parser. The parser parses out the HTML source code and constructs a DOM (Document Object Model) tree. The DOM tree is commonly used for traversing the web page within which every component is referred as node. The nodes are either HTML tags or contents consisting of texts and images. An example of HTML source code and its DOM tree illustration is shown in the figure 3 (a), (b).



```
<html>
<head>
<title> Title </title>
</head>
<body>
<p>
The content that are displaying
in a paragraph
</p>
<div style=" ">
Text messages
</div>
</body>
</html>
```

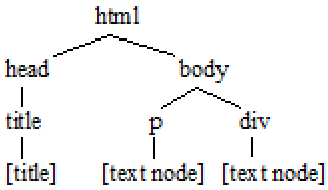


Figure 3 (a) HTML source code   Figure 3 (b) Corresponding DOM tree illustration

B. Tag path clustering

The visual information rendered on a web page, like fonts and layout is conveyed by HTML tags. A given link tag will have different appearance once it follows different paths within the DOM tree. For every tag occurrence, there is an HTML tag path, containing an ordered sequence of ancestor nodes within the DOM tree. Every HTML tag maps to an HTML path. Each tag path defines a unique visual signal. Figure 4 shows finding tag path from HTML code and figure 5 shows extracting visual signals from the given tag path of web page. A visual similarity measure that captures how closely the visual signals appear and interleave is introduced which ends up in a pair wise similarity matrix between visual signals. The pair wise similarity matrix can be fed into a clustering algorithm. The spectral clustering algorithm produces clustering results based on the pair wise similarity matrix. A cluster containing *n* visual signals indicates that those *n* visual signals are from a similar data region with high probability.

CODE NO	HTML SOURCE CODE	TAG PATH
1	<html>	Html
2	<body>	html/body
3	<table>	html/body/table
4	<tr>	html/body/table/tr
5	<td> Cell 1 </td> </tr>	html/body/table/tr/td
6	<tr>	html/body/table/tr
7	<td> Cell 2 </td></tr>	html/body/table/tr/td
-	</table></body></head></html>	-

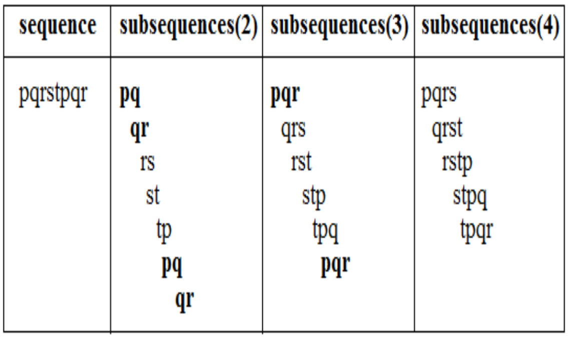
Figure 4 finding tag path from the HTML Code

CODE NO	UNIQUE TAG PATH	VISUAL SIGNAL
1	Html	[1 0 0 0 0 0 0]
2	html/body	[0 1 0 0 0 0 0]
3	html/body/table	[0 0 1 0 0 0 0]
4,6	html/body/table/tr	[0 0 0 1 0 1 0]
5,7	html/body/table/tr/td	[0 0 0 0 1 0 1]

Figure 5 Visual signal extractions

C. Segmentation Process

After tag path clustering, the web page is segmented either by reappearance based segmentation or by layout based segmentation. Reappearance pattern detection technique is employed to seek out pattern in a sequence that consists of HTML tags. Reappearance is defined as sub list of length *i* (>1) occurring doubly or more in a length *j*. Maximum length of reappearance for a list of length is *j*/2, satisfying the formula 1<*i*<*j*/2. Figure 6 shows a way to realize reappearance from a sequence.



sequence	subsequences(2)	subsequences(3)	subsequences(4)
pqrstpqr	**pq**	**pqr**	pqrs
	qr	qrs	qrst
	rs	rst	rstp
	st	stp	stpq
	tp	tpq	tpqr
	pq	**pqr**	
	qr		

Figure 6 Finding reappearance from a sequence.

In this example, the reappearance subsequence are [*pq*], [*qr*], and [*pqr*] because it occurs twice in the sequence. Take into account this reappearance subsequence as patterns. A key pattern is a repetitive pattern in a sequence that is longest and most frequent. For the sequence *pqrstpqt*, the repetitions obtained by using reappearance detection technique are [*pq*], [*qr*] and [*pqr*]. These key patterns are used to build implicit nodes by separating the subsequence containing key pattern. For doing this, a pattern matching algorithm is applied and key pattern is matched from left to right sequentially with the given series. Once a match is found at position *x*, the method continues to seek out succeeding match within the remaining part of the series. If there is another match at position *y*, the subsequence starting from *x* through *y*-1 is sorted with new implicit node. Figure 7 shows generation of implicit node by using key patterns.

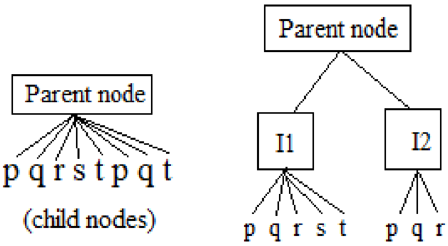


Figure 7 Generation of implicit node by using key pattern

If there is no key pattern, it will segment based on page layout information like <TABLE>, <TR>, <TD> <DIV>, <FRAME>, etc., that are not used to solely compose the table, however conjointly to outline the layout of the web page. Such tags are identified and divided as a block.

D. Hyperlink creation and displaying in mobile device

The web page is split into blocks, after segmentation process. Informative blocks are determined by evaluating the quantity of information within the blocks, which might be done by assigning an importance weight to every node considering the amount of reappearance of node patterns in a web page. Normalize the importance weight for every node by the maximum number of reappearance for making the calculation easy. From that informative divided block, hyperlink is created and displayed on the mobile devices. Users choose their own interested hyperlink. The interested information alone is exhibited to the users.

IV. EXPERIMENTAL RESULTS

The web page is segmented using tag path clustering method and its performance is evaluated using three different types of web sites. The web sites used for analysis are shown in table I. The results of segmented page blocks are shown in figure 6 (a), (b) and (c).

TABLE I Web sites used for evaluation

S.no	Web sites
1.	http://www.amazon.com
2.	http://www.ebay.com
3.	http://www.tomcat.apache.org

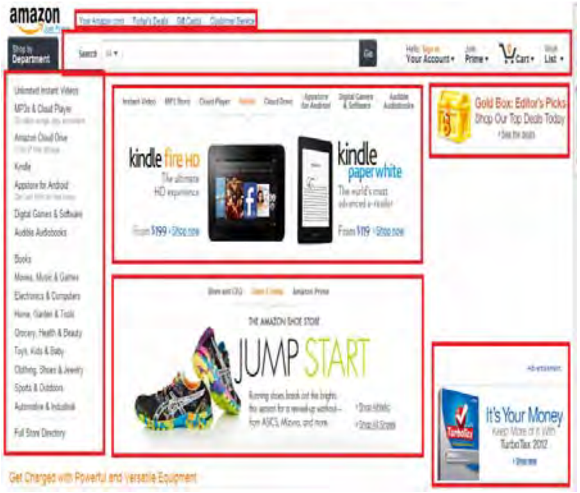


Figure 6 (a) Segmented Amazon web page

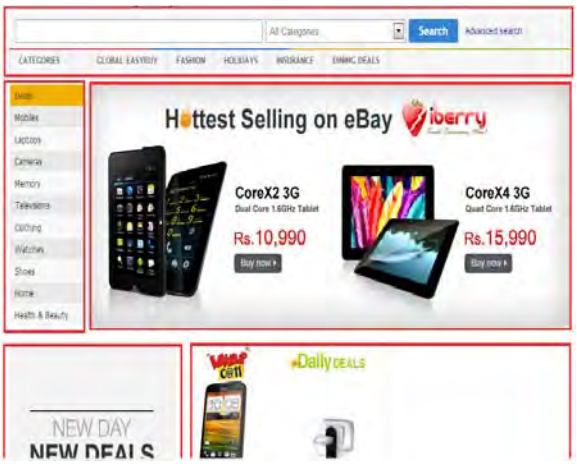
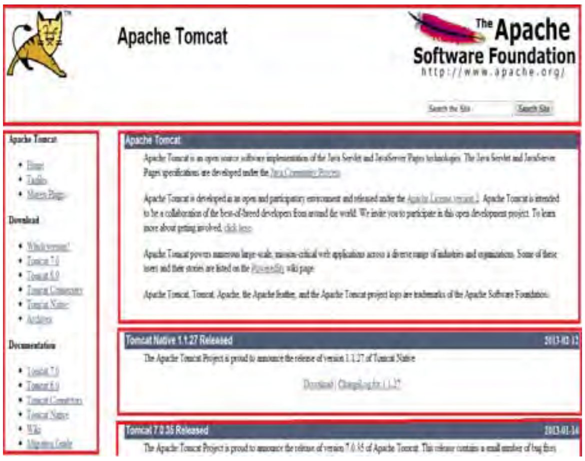


Figure 6 (b) Segmented eBay web page



Apache Tomcat

- Home
- Taglibs
- Maven Plugin

Download

- Which version?
- Tomcat 7.0
- Tomcat 6.0
- Tomcat Connectors
- Tomcat Native
- Archives

Documentation

- Tomcat 7.0
- Tomcat 6.0
- Tomcat Connectors
- Tomcat Native
- Wiki
- Migration Guide

Apache Tomcat

Apache Tomcat is an open source software implementation of the Java Servlet and JavaServer Pages technologies. The Java Servlet and JavaServer Pages specifications are developed under the Java Community Process.

Apache Tomcat is developed in an open and participatory environment and released under the Apache License version 2. Apache Tomcat is intended to be a collaboration of the best-of-breed developers from around the world. We invite you to participate in this open development project. To learn more about getting involved, click here.

Apache Tomcat powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations. Some of these users and their stories are listed on the PoweredBy wiki page.

Apache Tomcat, Tomcat, Apache, the Apache feather, and the Apache Tomcat project logo are trademarks of the Apache Software Foundation.

Tomcat Native 1.1.27 Released 2013-02-12

The Apache Tomcat Project is proud to announce the release of version 1.1.27 of Tomcat Native.

Download | ChangeLog for 1.1.27

Tomcat 7.0.35 Released 2013-01-14

The Apache Tomcat Project is proud to announce the release of version 7.0.35 of Apache Tomcat. This release contains a small number of bug fixes

Figure 6 (c) Segmented Apache tomcat web page

The Performance can be evaluated with the help of two parameters namely precision and recall that is denoted as P and R respectively.

$$\text{Precision} = \frac{\text{correctly segmented blocks}}{\text{blocks segmented by the algorithm}} \qquad (1)$$

$$\text{Recall} = \frac{\text{correctly segmented blocks}}{\text{ideal blocks}} \qquad (2)$$

To measure the accuracy, F-measure is employed which is denoted as F-M. The F-Measure can be calculated as,

$$\text{F - Measure(F - M)} = \frac{2 * \text{Precision} * \text{Recall}}{(\text{Precision} + \text{Recall})} \qquad (3)$$

The comparison results are analyzed and represented in table II. Based on the data in the table, it is shown that the segmentation with tag path clustering had better precision, recall and F-Measure value than segmentation without tag path clustering.

TABLE II Comparison of Proposed Methodology

WEB SITES	**Segmentation without tag path clustering**			**Segmentation with tag path clustering**		
	P	**R**	**F-M**	**P**	**R**	**F-M**
Amazon. Com	0.75	0.65	0.70	0.87	0.75	0.81
eBay.com	0.84	0.74	0.79	0.88	0.78	0.82
Tomcat. Apache.org	0.87	0.79	0.82	0.91	0.84	0.87
Mean	0.82	0.73	0.77	0.88	0.77	0.83

Table III shows the comparison of bandwidth utilization using with and without interface. Using the proposed technique as an interface, the consumption of bandwidth is reduced. The bandwidth calculation was made by considering the initial page loading size in kb with and without interface. The bandwidth is saved, when we display the divided blocks to the mobile devices.

TABLE III Consumption of Bandwidth

Web sites	**Without interface**	**With interface**	**Bandwidth saving**
Amazon.com	245KB	1KB	74.55%
eBay.com	110KB	2KB	79.925%
Tomcat.apache.org	23KB	1KB	61.45%

The memory utilization of three different web sites was calculated for segmentation without and with tag path clustering and it is compared that is shown in the figure 7.

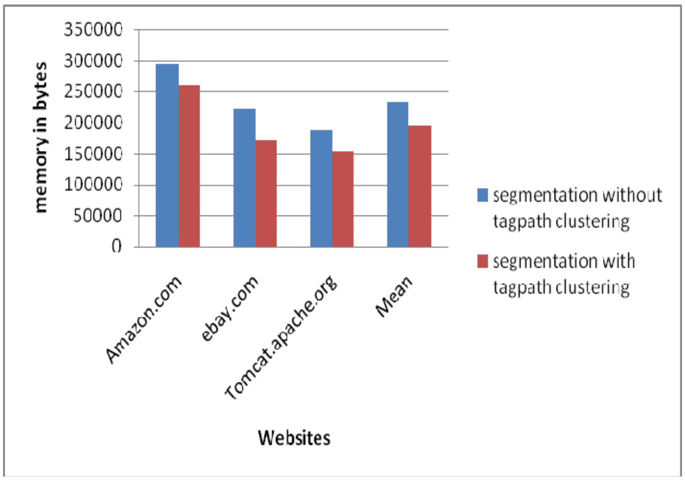


Figure 8: Memory comparison of segmentation with and without tag path clustering

The segmentation method with tag path clustering had occupied less memory than segmentation without tag path clustering.

V. CONCLUSION

A tag path clustering based approach to dynamic web page segmentation for mobile devices was proposed. The proposed method is appropriate for dynamic net setting either by recognizing reappearance tag patterns or page layout information. This approach can able to resolve the issue of resource limitation of mobile devices by reducing the bandwidth consumption and memory utilization, thereby making the mobile browsing easier. In future, semantic analysis modules can even be incorporated to provide personalization of users browsing activities on small screen devices.

VI. REFERENCES

[1] Kalaivani, Rajkumar, "*Dynamic Web Page segmentation Based on Detecting Reappearance and Layout of Tag Patterns for Small Screen Devices*", IEEE conference proceedings, Page(s): 508 - 513, April *2012.*

[2] Kalaivani, Rajkumar, "*Reappearance Layout based Web Page Segmentation for Small Screen Devices*", International Journal of Computer Applications, *Vol. 49– No.20, July 2012.*

[3] L.Wu, N.Liu Y.He, Y.Ke, "*A Block Gathering Based on Mobile Web Page Segmentation Algorithm*", Proceedings of IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2011.

[4] Nyein, S.S., "*Mining Contents in Web Page Using Cosine Similarity*", Proceedings of IEEE International Conference on Computer Research and Development, vol: 2, page(s): 472- 475, 2011.

[5] Jinbeom Kang, Jaeyoung Yang, Joongmin Choi, "*Repetition-based Web Page Segmentation by Detecting Tag Patterns for Small-Screen Devices*", IEEE Transactions on Consumer Electronics, IEEE, vol. 56, no. 2, pp.980-986, 2010.

[6] SandipDebnath, Mitra. P, "*Automatic Identification of Informative Sections of Web-pages*",IEEE Transactions on knowledge and Data Engineering,vol. 17, no. 9, 2008.

[7] C.Kohlscutter, W.Nedjl "*A Densitometric Approach to Web Page Segmentation*", Proceedings of the 17th ACM conference on Information and Knowledge management, Pages 1173-1182, 2008.

[8] D. Chakrabarti, R. Kumar, "*A Graph-Theoretic Approach to Webpage Segmentation*", Proceedings of the 17th ACM international conference on World Wide Web, Page(s): 21–25, 2008.

[9] G. Hattori, K. Hoashi, K. Matsumoto, and F. Sugaya, "*Robust web page segmentation for mobile terminal using content distances and page layout information*," Proc. 16th Intl. Conf. on World Wide Web, pp. 361–370, 2007.

[10] W. Lee, S. Kang, S. Lim, M. Shin, and Y. Kim, "*Adaptive hierarchical surrogate for searching web with mobile devices*," IEEE Trans. Consumer Electron., vol. 53, no. 2, pp. 796-803, 2007.

[11] Y. Chen, X. Xie, W. Ma, and H. Zhang, "*Adapting web pages for small screen devices*," IEEE Internet Computing, vol. 9, no. 1, pp. 40-56, 2005.

[12] Cai, Yu,Wen, "*Block-based Web Search*", Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pages 456–463, 2004.

[13] Y. Chen,W.-Y. Ma, and H.-J. Zhang, "*Detecting web page structure for adaptive viewing on small form factor devices*," Proc. 12th Intl. Conf. on World Wide Web, pp. 225–233, 2003.

[14] Zheng and M. Atiquzzaman, "*A novel scheme for streaming multimedia to personal wireless handheld devices*," IEEE Trans. Consumer Electron., vol. 49, no. 1, pp. 32-40, 2003.

[15] L. Yi, B. Liu, and X. Li, "*Eliminating noisy information in web pages for data mining*," Proc. 9th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, pp. 296-305, 2003.

[16] Yonghyun Hwang et al., "*Structure-Aware Web Transcoding for Mobile Device*",IEEE Transactions on Internet computing, vol. 7, no.14-21, 2003.

[17] N. Milic-Fraying, R. Sommerer, K. Rodden, and A.Blackwell., "*SearchMobil: web viewing and search for mobile devices*",12th International World Wide Web Conference (WWW 2003), poster, Budapest, Hungary, May 2003.

[18] S. Lin and J. Ho, "*Discovering informative content blocks from Web documents*", ACM, Proceedings of 8th International Conference on Knowledge Discovery and Data Mining, pp. 588-593, 2002.

[19] Chen, J., Zhou, B., Shi, J., Zhang, H.-J., and Wu, Q., "*Function-Based Object Model towards Website Adaptation*", In Proceedings of the 10th International World Wide Web Conference, 2001.

[20] Y. Yang and H. Zhang, "*HTML page analysis based on visual cues*", Proc. 16th Intl. Conf. on Document Analysis and Recognition, p. 859, 2001.