Web Browsers on Smart Mobile Devices: A Gap Analysis on the State of the Art

Chinthaka Dharmasiri[1], Rythmal Jayendranath[2], Amila .L. Ariyarathna[3], Pahan .M. Perera[4], Shahani M. Weerawarana[5]

Department of Computer Science and Engineering, University of Moratuwa
Moratuwa, Sri Lanka

[1]chinthakadd@gmail.com [2]rythmalj@gmail.com [3]amila.lakshitha@gmail.com [4]pahan.madushanka@gmail.com
[5]shahani.w@gmail.com

Abstract*— In recent years, we have seen an accelerated development of mobile devices in terms of hardware capabilities, navigation techniques and physical design. Today users are gradually shifting towards choosing a smart mobile device as their preferred device. The technologically advanced capabilities that current smart mobile devices possess have spawned a plethora of engaging applications that are attractive and effective in their usefulness and usability. In contrast, the web browser, which can be considered one's personal gateway to the World Wide Web, has not transformed to keep pace with the technology advancements of smart mobile devices in order to create a better experience to the user. All of these browsers are built on top of the same visualization techniques and layouts used by the desktop oriented web browsers. This is not the ideal form of presentation for a smart mobile device. The underlying issue is that the majority of generic websites have not been designed to cater to the relatively smaller mobile device screens. Web browsers for mobile devices have not been able to effectively address this issue. This has created a significant user experience gap between web browsing applications and the smart mobile device capabilities. In this paper, we have presented a gap analysis framework for smart mobile device web browsers in which we have analysed the aforementioned issue from different perspectives. With this framework, we intend to pin down a set of guidelines for a mobile browser that would lead to a better presentation of generic web sites.

***Keywords*— Content Adaptation, Mobile Cloud, Semantic Web, Smart Mobile Devices, Web Visualization**

I. Introduction

The browsing paradigm introduced in 1993 with the invention of the NCSA Mosaic browser was widely adopted by many browsers that followed [1].This browsing paradigm has made its way to the mobile browsers used in smart mobile devices as well. However, this basic paradigm has several drawbacks and limitations in the context of mobile browsing. This has resulted in creating a significant user experience gap between the presentation of generic websites and the presentation and layout capabilities of smart mobile devices. To address this issue, it is necessary to consider this problem from different perspectives and to develop a potential solution model leading to a new visualization paradigm for mobile web browsing.

This paper presents an analysis framework to highlight and understand the prevailing gaps between smart mobile device capabilities and existing web browsers. Solution techniques to bridge such gaps can be clarified by leveraging such a systematic analysis framework. This problem is looked upon in several perspectives including content restructuring and semantic based organization, content optimization and acceleration, deployment architecture, web rendering and HyperText Markup Language (HTML) parsing and web visualization. This is followed by the sections which discuss the existing browsers and how the future of browsing should be transformed. The recommendations aim to propose how mobile browsers should evolve mainly in the frame of creating a better web experience for smart device users.

A. Development in Smart Mobile Device Technologies

The main reason for the existence of this lag in the web browsing in smart devices is the rapid development in smart device technologies. During the last decade the smart mobile device has transformed from being solely a communication device to a handheld minicomputer. There are various new technologies featured in these devices. Touch interfaces have replaced the traditional keypad and stepped into a new level with multi touch and gestures after the introduction of Apple iPhone in 2007. Developments in motion sensing have been used to bring a phenomenal user experience in gaming and a variety of other applications. With the use of the embedded cameras in mobile phones, eye tracking technologies have come into play very recently which has lifted the document reading and user interaction into a whole new level. Global Positioning Systems (GPS) are used very commonly and it has brought a new era in digital maps such as Google maps. Also mobile communication networks are now in the transition phase from 3G to 4G providing ultra-mobile broadband access for smart mobile devices using Long Term, Evolution (LTE) and Worldwide Interoperability for Microwave Access (WiMAX) technologies. Processing power wise, smart mobile devices have reached the level of "super" smart phones with the use of quad core processors, which has improved the ability of concurrent processing, multitasking and split screen capabilities.

At the rate smart mobile devices are evolving, future seems so unpredictable. Augmented reality, flexible screens, inbuilt projectors, 3d screens and holograms are few of the concepts that we possibly will see in future smart devices. The question we need to ask is "are the prevailing web browsers for smart mobile devices ready to leverage these technological innovations?"

B. Evolvement of Web Technologies

One major challenge the web browsers had to face over the years was in rendering web content which were created using diversified web technologies. Thus, the introduction of technologies such as HTML5, CSS3 and Web 3.0 is directed towards incorporating a ubiquitous platform for future web development. Moreover, it has become much efficient for web browsers in rendering webpages designed using these novel web technologies [17, 34]. Thus, these technologies have not only made the web more capable, they have made the foundation for a better web standardization.

However, from a user perspective, the underline technologies of a website would not be of any concern if the

978-1-4799-1276-6/13/$31.00 ©2013 IEEE

web browser is unable to visualize the content properly. Therefore, the key challenge for web browsers would be to take a holistic view of the existing web technologies and ensure that supportability for generic web which is developed in different technologies.

II. PERSPECTIVES ON GENERIC WEB ADAPTATION IN MOBILE WEB BROWSERS

A. Content Structuring and Semantic Based Organization

Content adaptation and restructuring is essential in creating cohesive presentation for mobile screens out of generic websites. The mobile browsers need to focus on various web mining techniques for such content adaptation. The idea behind this is to create appropriate web content for the minimal resources available in the mobile devices. This can take two main approaches. While one approach is to restructure the content using the Document Object Model (DOM) structure of the web, the other is the most effective approach which uses the semantic web technologies [20]. In the current mobile browsers, the most common adaptation is the linearization of the web content to minimalize the horizontal navigation as seen in Opera Mini web browser [24]. In this process, browsers should make sure that visually coherent content groups stay together. A further developed approach for this is to create semantically coherent groups by analysing the HTML tags and Cascading Style Sheets (CSS) [18]. Considering the images, browsers can undertake image dimension adaptation and colour manipulation techniques such as palette mapping and contrast enhancement. Also, encoding (UTF-8, Shift_JIS) and conversion processes are done by the browsers to adapt the format of the content. Skyfire does this for video formats for flash content [36]. Flipboard [10] is a mobile application which uses custom scrappers to extract content from news sites and present them in a mobile adapted manner. It is one example for the usage of web mining techniques to uplift the web experience via mobile devices.

The semantic web technologies allow the end user to have adequate amount of freedom in the presentation of web content [27]. Resource Description Framework (RDF) based semantic data is growing at a rapid rate, mainly due to the existence of central RDF repositories such as DBpedia [7] and FreeBase [11]. Google, with the introduction of rich snippets [31] has been experimenting with RDF and microformats. RelFinder visualization tool is using RDF data to build and present relationships between given entities [28]. Web browsers should also be compatible to use semantic data in future. One more aspect should be the ability to cache the RDF data in a local repository which will enhance the performance and customizability. Various tools which use RDF format are being developed to utilize advantages of the semantic web. User experience cannot be uplifted merely by analysing semantic data. It is essential to have a proper method of presenting them. Therefore, mobile browsers should have standard specifications on the presentation of RDF data [27]. It could even be developed to an extent where the user gets to pick the presentation model or use layouts which are generated by separate design layout experts [22]. The browser may have the capability to generate RDF data from HTML content with the use of tools such as Apache Stanbol in the absence of any published RDF data [3].

Knowing the device capabilities is critical when adapting the web content. World Wide Web Consortium (W3C) has created the Device Description Repository [8] where information such as screen dimensions, input mechanisms, supported colours, known limitations and special capabilities of the web enabled devices are stored [19, 37]. CSS adaptation according to device screen size has been enabled by browsers including Safari, Opera and Android. But these are not stable and the author of the content has to decide the initial block size and initial zoom factor. However, mobile device detection and adaptation of content according the device's specification is not properly implemented in any of the existing browsers.

B. Content Acceleration and Optimization

In the context of mobile web browsing, some advanced techniques are needed for accelerating the web content delivery and optimizing them for smart devices.

SPDY [36] is a replacement for Hypertext Transfer Protocol (HTTP) which is designed to speed up transfer of web pages. Since mobile networks typically experience high round-trip times and lower throughput, SPDY is a very good method to improve web page download speeds on mobile networks. SPDY has a header compression technique to eliminate redundant data in HTTP headers and out-of-order request processing, in order to avoid blocking of HTTP responses. TCP connection optimization is also a notable feature in SPDY. Therefore, a modern mobile web browser can accelerate web content transferring using SPDY.

WebP is an image optimization technique which provides lossless and lossy compression for images on the web. Mobile web browsers can offer high performance and better user experience by converting web resources to a mobile friendly format using these techniques [40].

Web caching and Web prefetching are two important techniques used to reduce the response time perceived by mobile users. Since many web resources change infrequently, mobile web browsers can reuse local copies of resources, which reduce time for downloading web content again and again. Caching mechanisms can be implemented in client side, client side proxy or in server side proxy. Web prefetching is an advanced technique which downloads web content before they are requested. With these two techniques, user-perceived network latency for a web page can be significantly reduced or even eliminated [41].

C. Deployment Architecture

The context of mobile web browser differs from one another. Here the context refers to all the factors that relate to the browser that the web page is displayed on. For a successful adaptation of generic web, the contextual aspects such as screen size and bandwidth should be considered. Therefore, the mobile web browser should have a delegated entity which can perform the adaptation. There are many deployment techniques for this entity that are being experimented. The proxy or intermediary server is one such model [20]. A proxy standing in between the client browser and the web maps the content according to the device. A separate context information server can also be placed in order to acquire device information of the user agent [12]. This is one existing solution for the problem that has been tried out. There are several transcoders developed to transform the web content on top of the web browser itself as well.

Another approach, being quite new to mobile browsing is the mobile cloud computing [5, 6]. This has encapsulated the proxy server model and added some new benefits for generic web visualization. The mobile cloud could enable screen virtualization [11, 21] offloading the rendering engine, adaptation of caching mechanism, usage of semantic processing servers and much more capabilities. Further, high availability can be achieved through clustering, which is an advantage over the proxy intermediary.

D. Web Visualization / Layout

Mobile Web Visualization mainly involves in graphical presentation of web content on Smart devices. Delivering the same desktop view for the mobile screen is not the most effective way for the mobile devices. In fact, there are drawbacks such as the "cluttered view", for the users in browsing generic websites. An effective strategy for content visualization enables representing websites in units of semantic content with their relationships while creating cohesive views for a much easier navigation for the user.

Yet the basic layout technique is a top down approach in which the page width, font and image sizes, and etc. are adapted to the device screen. This improves the web page accessibility and navigation. However, this has limitations and creates inefficiencies with larger content.

Reading view detects the main article of a web page and views that text in a separate page without any distractible items. Therefore, the limited screen space can be fully utilized to display useful web content.

Splitting the whole web page and presenting the content as subpages is a possible solution as most web pages do not fit to smart devices. This is a suitable view for readers, successfully implemented by Flipboard despite the difficulty to see the overview of the web content at once. Therefore, page splitting can organize the content but it should be improved to preserve the overall content presentation.

Then Content Outlining approach can solve the problem associated with page splitting. It gives an overall idea about the content of the web page. Typical Table of Content (TOC) and Sitemap of a web page are some of the examples for this approach [2]. Thumbnail TOC view has addressed the limitation of content amount in the aforementioned approach. It basically presents a summarized content along with the topic in the TOC [5].

Apart from these techniques, relationship among the entities of the website can be built with the semantic extraction of the content. It opens out several new visualization techniques such as the mind maps, Relationship Trees and Concept Maps.

E. Web Rendering and HTML Parsing

The rendering engine performs the transformation of HTML pages to DOM trees with a parser algorithm. In the context of generic web pages, this is not by any means a simple task. There should be error tolerance to the common syntax errors and invalid tag definitions while supporting the scripting which obstruct the static HTML content. W3C DOM standard model [39] has defined a set of properties that all browsers should support. There are certain browser dependant properties which can create JavaScript errors in rendering the page.

In the DOM tree construction, adaptation techniques can be included to create a DOM tree which is a better fitting to the respective mobile screen when it is rendered. This manipulation of the original tree structure for the HTML page can take place either before the rendering or after. An Extensible Stylesheet Language Transformations (XSLT) transformation can be applied after the extraction of the semantically coherent units from the page. Alternatively, the semantic extraction can be applied on the DOM tree constructed by the rendering engine [15].

The web rendering implementation should have a generic style decorator for DOM objects. There are several other Extensible Markup Language (XML) declarations such as XHTML, XForms, SVG, MathML, and etc. which should also be handled by the browser.

F. Accessibility

The web accessibility guidelines provided by the W3C are to ensure that web is accessible by everyone including the people with disabilities [37]. The accessibility is generally coupled with the navigation techniques. The mobile browsers can adapt various navigation patterns based on the technologies provided by the device itself. Motion sensing, touch, gestures and also the voice recognition support can be utilized to increase the accessibility. The mobile browser is needed to support the input of HTML forms, popup windows and other user interface components written according different specifications.

III. Gap Analysis Framework for Smart Mobile Device Web Browsers

Table 1 presents a comprehensive summary of above mentioned perspectives and how different mobile web browsers have addressed them.

Table 1: Perspectives Implemented by Current Mobile Browsers

Perspective	Indices	Adapted Method	Supporting Web Browser
Web visualization / layout	Visualization techniques	Top-down Font resizing	Opera [24]
		Reading view	Firefox, Safari [32], Dolphin (Webzine)
		Content splitting	FlipBoard
		Content outlining	n/a (research state)
	Interface Description Language (IDL) Definitions	XPIDL	Mozilla
Content restructuring / semantic based organization	Use of RDF meta data	RDF processing	Rhizomer [30], BrownSause [4], Sgvizler [33]
		Accessing RDF repository(single/multi)	RelFinder [28]
		RDF caching	n/a (research state)
		RDF presentation standards	RelFinder, Rhizomer
		Supporting RDF extensions	Firefox(research)[29]
	Semantic Extraction	Content linearization and minimization	Opera Mini, Google Wireless Transcoder
		Atomic unit structuring	Safari
		Image and video adaptation	Skyfire
Content acceleration / optimization	Protocol	SPDY	Google Chrome, Firefox, Opera Mini, Amazon Silk [19]
	Caching and Prefetching	Study user patterns for content prefetching	Amazon Silk
		Server push	Google Chrome

	Image Optimization	WebP	Chrome, Android browser, UC browser [38], Safari
Deployment Architecture	Cloud Based	Amazon EC2 Cloud based architecture (Cache, Proxy, Rendering)	Amazon Silk
		Cloud Architected with JavaScript Rendering Engine (Flash supported)	Maxthon [23] Puffin [26]
		Cloud Architecture with Presto engine	Opera Mini
		Data Compression Proxy (using SPDY)	Google Chrome
	Proxy-based		Skyfire
Web Rendering / HTML parsing	DOM Compliance	Fully adapted W3C standard	Mozilla
		Browser Dependent DOM Properties	IE
	XML/XSLT support	Support all W3C Recommended XML formats including XForms	Mozilla
		XML, XSLT, and XPath (After IE5)	IE
		XML, XSLT, and XPath	Google Chrome
		XML, XSLT, XPath (After version 9)	Opera
Accessibility	Navigation	Vertical Navigation, Flipping Multi touch zooming Gesture navigation	Opera FlipBoard Dolphin
	Voice Search		Google Chrome, Dolphin
	Screen Readers	Apple VoiceOver	Safari
	Haptics	n/a (research state)	n/a (research state)

IV. Future Direction

These different perspectives have opened several paths to help bridge the gap between generic websites and smart mobile device web browsers.

A rigorous research on applying semantic based techniques in web content adaptation is required as future work. If the entities of a website and their relationship across the pages can be extracted, use of RDF data enables them to be mapped to a better visualization technique in order to suit the mobile device.

The visualization techniques that can present a website with a summarized overview should create a paradigm shift from the conventional web browsing.

Also it is important to study how different navigation techniques can be incorporated with web visualization to enhance the accessibility to people including the differently-abled.

A further research on mobile cloud computing should be conducted, especially on offloading web rendering with the use of server-side scripting technologies as well. It will be useful to design cloud based architecture to implement web content adaptation utilizing versatility of the cloud.

Web technologies have emerged towards integrating the diversified web content into one foundation. This seems to be the future direction of web with the introduction of HTML5, CSS3 and Web 2.0. Such technologies are built in order to standardize the web rendering process by eliminating the content type incompatibilities. However, generic web in definition specifies webpages of different technologies. The majority of webpages that prevail are not updated to occupy these novel technologies. The web technologies will continue to evolve with time. However, if generic web is to survive, web browsers must ensure consistency in treating all these different websites in web rendering. Thus, it illustrates the fact that the web browsers have to support the generic web to appear with same behavioural consistency despite the evolvement of technologies in use.

V. Conclusions

The analysis framework presented in this paper focuses on six different perspectives in order to cover all the aspects required for the generic web content to reach the mobile browser in a coherent, visually and environmentally effective manner. In the instantiation of our framework for the existing browsers, the different aspects that web browsers have focused on were highlighted. However, there is yet a lot to be achieved in terms of content visualization and semantic based organization, which will create a paradigm shift in the mobile browsers. The already available RDF data could be used to create more flexibility in the content structuring and adaptation rather than merely focusing on the HTML syntax based web mining techniques.

In the deployment architecture, different browsers have adapted different approaches. Most browsers are creating clients with JavaScript engines whilst a new mobile cloud paradigm has also emerged. Both architectures have their own benefits, but in terms of web content adaptation, the cloud computing can bring several benefits extending from the proxy-based solutions that existed for some time now. Next we discussed on visual presentation of the web content. Although all the major players of the mobile browsing have carried out various research studies on this, they seem to be fenced by the traditional browsing paradigm. A web visualization paradigm that is aligned with the latest navigation techniques, the relationship amongst the entities created on a semantic base, and the context of the device would enable a more organized presentation of generic websites. The content adaptation (semantic unit extraction) needs to be implemented as a technique at a certain point of the process such as DOM tree creation that can bring better content units to the presentation.

Content acceleration is an area which has been a concern of mobile browsers. HTTP is no longer considered good enough and browsers are shifting towards SPDY which is more efficient. Several caching and prefetching techniques are also being adapted to optimize the process. Considering the findings emanating from the analysis framework, current browsers would have to make some major changes. The existing browsers' stability and the web's high dependency on those browsers may create reluctance in the browsing world to take this leap.

The insights derived from the analysis framework presented in this paper would be an excellent launch pad to design requisite changes in order to create a better web browsing experience for smart mobile device users.

REFERENCES

[1] "About NCSA Mosaic". [Online]. Available: http://www.ncsa.illinois.edu/Projects/mosaic.html/. [Apr. 05, 2013].

[2] B. Adipat, D. Zhang, and L. Zhou, "Adapting Content Presentation for Mobile Web Browsing," in Information Systems, Technology and Management, S. K. Prasad, H. M. Vin, S. Sahni, M. P. Jaiswal, and B. Thipakorn, Eds. Springer Berlin Heidelberg, 2010. pp. 293–303.

[3] "Apache Stanbol". [Online]. Available: http://stanbol.apache.org/, [Apr. 07, 2013].

[4] "BrownSauce RDF Browser". [Online]. Available: http://brownsauce.sourceforge.net/, [Apr. 07, 2013].

[5] Y. Chen, W.-Y. Ma, and H.-J. Zhang, "Detecting web page structure for adaptive viewing on small form factor devices," in Proceedings of the 12th international conference on World Wide Web, New York, NY, USA, 2003, pp. 225–233.

[6] B.G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: elastic execution between mobile device and cloud," in Proceedings of the sixth conference on Computer systems, New York, NY, USA, 2011, pp. 301–314.

[7] "DBpedia" [Online]. Available: http://dbpedia.org/About/, [Apr. 07, 2013].

[8] "Device Description Repository Simple API." [Online]. Available: http://www.w3.org/TR/DDR-Simple-API/, [Apr. 06, 2013].

[9] "Dolphin Browser (HD)" [Online]. Available: http://home.dolphin-browser.com/tunny/Feature.htm/, [Apr. 01, 2013].

[10] "Flipboard" [Online]. Available: https://flipboard.com/, [Apr. 01, 2013].

[11] "Freebase" [Online]. Available: http://www.freebase.com/, [Apr. 01, 2013].

[12] Gasimov, F. Magagna, and J. Sutanto, "CAMB: context -aware mobile browser," in Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, New York, NY, USA, 2010, pp. 22:1–22:5.

[13] Neal Gompa, "What is LTE?". [Online]. Available: http://www.extremetech.com/mobile/110711-what-is-lte/, [Apr. 01, 2013].

[14] J. Hendler, "Web 3.0 Emerging," *Computer*, vol. 42, no. 1, pp. 111–113. 2009.

[15] "How browsers work." [Online]. Available: http://taligarsiel.com/Projects/howbrowserswork1.htm#w3c/, [Apr. 05, 2013].

[16] "How does Google modify web pages for mobile viewing?" [Online]. Available: http://support.google.com/webmasters/bin/answer.py?hl=en&answer=35312/, [Apr. 08, 2013].

[17] "HTML5". [Online]. Available: http://www.w3.org/TR/html5/, [Apr. 5, 2013]

[18] Jeff J.S. Huang, Stephen J.H. Yang, Zac S.C. Chen and Frank C.C. Wu," Web content adaptation for mobile device:A fuzzy-based approach", Knowledge Management & E-Learning: An International Journal (KM&EL), Vol 4, No 1, 2012.).

[19] "Introducing Amazon Silk" [Online]. Available: http://amazonsilk.wordpress.com/2011/09/28/introducing-amazon-silk/, [Apr. 01, 2013].

[20] T. Laakko and T. Hiltunen, "Adapting Web content to mobile user agents," IEEE Internet Computing, vol. 9, no. 2, pp. 46–53, 2005.

[21] Y. Lu, S. Li, and H. Shen, "Virtualized Screen: A Third Element for Cloud amp; #x2013; Mobile Convergence," IEEE Multimedia, vol. 18, no. 2, pp. 4–11, 2011.

[22] Jos'e Antonio Mac'ias,"Enhancing Interaction Design on the Semantic Web: A Case Study", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 42, NO. 6, NOVEMBER 2012.

[23] "Maxthon's Cloud Browser Sets You Free". [Online]. Available: http://www.maxthon.com/, [Apr. 10, 2013].

[24] "Opera Browser". [Online]. Available: http://operamini.com/mobile/specs/, [Apr. 02, 2013].

[25] M. Poh, "5 Key Features to Expect in Future Smartphones." [Online]. Available: http://www.hongkiat.com/blog/future-smartphone-features/, [Apr. 02, 2013].

[26] "Puffin Browser". [Online]. Available: http://www.puffinbrowser.com/, [Apr. 10, 2013].

[27] D. A. Quan and R. Karger, "How to make a semantic web browser," in Proceedings of the 13th international conference on World Wide Web, New York, NY, USA, 2004, pp. 255–265.

[28] "RefFinder". [Online]. Available: http://www.visualdataweb.org/relfinder-.php/, [Apr. 07, 2013].

[29] "Resource Description Framework (RDF)". [Online]. Available: http://www-archive.mozilla.org/rdf/doc/, [Apr. 05, 2013].

[30] "Rhizomer". [Online]. Available: http://rhizomik.net/html/rhizomer/, [Apr. 07, 2013].

[31] "Rich Snippets". [Online]. Available: http://support.google.com/webmasters/bin/answer.py?hl=en&answer=99170/, [Apr. 05, 2013].

[32] "Safari". [Online]. Available: http://www.apple.com/safari/what-is.html/, [Apr. 10, 2013]

[33] "sgvizler, Javascript SPARQL result set visualizer". [Online]. Available: https://code.google.com/p/sgvizler/, [Apr. 09, 2013].

[34] L. Sikos, *Web Standards: Mastering HTML5, CSS3, and XML*, 1st ed. Berkely, CA, USA: Apress, 2011.

[35] "Skyfire Rocket Optimizer" [Online]. Available: http://www.skyfire.com/, [Apr. 10, 2013].

[36] "SPDY Performance on Mobile Networks". [Online]. Available: https://developers.google.com/speed/articles/spdy-for-mobile/, [Apr. 08, 2013].

[37] "The Web and Mobile Devices". [Online]. Available: http://www.w3.org/Mobile/, [Apr. 05, 2013].

[38] "UC Browser". [Online]. Available: http://www.ucweb.com/, [Apr. 11, 2013].

[39] "W3C Document Object Model." [Online]. Available: http://www.w3.org/DOM/, [Apr. 05, 2013].

[40] "WebP, A new image format for the Web" [Online]. Available: https://developers.google.com/speed/webp/, [Apr. 09, 2013].

[41] "Web Caching and Web Prefetching" [Online]. Available: http://www.mobilerevamp.org/2011/08/22/web-caching-and-web-prefetching/, [Apr. 08, 2013].