AUTHOR(S): HUGHES, C.J., ARMSTRONG, M., JONES, R. and CRABB, M.

TITLE: Responsive design for personalised subtitles.

YEAR: 2015

Publisher citation: HUGHES, C.J., ARMSTRONG, M., JONES, R. and CRABB, M. 2015. Responsive design for personalised subtitles. In the Proceedings of the 12th Web for all conference (W4A'15), 18 - 20 May 2015, Florence, Italy. New York: ACM [online], article No. 8. Available from: https://doi.org/10.1145/2745555.2746650

OpenAIR citation: HUGHES, C.J., ARMSTRONG, M., JONES, R. and CRABB, M. 2015. Responsive design for personalised subtitles. In the Proceedings of the 12th Web for all conference (W4A'15), 18 - 20 May 2015, Florence, Italy. New York: ACM, article No. 8. Held on OpenAIR [online]. Available from: https://openair.rgu.ac.uk/

Responsive Design for Personalised Subtitles

Chris J Hughes[1], Mike Armstrong[1], Rhianne Jones[1], Michael Crabb[2]
[1]BBC R&D, Salford, UK
[2]University of Dundee, Dundee UK
chris.hughes@rd.bbc.co.uk, mike.armstrong@bbc.co.uk, rhia.jones@bbc.co.uk, michaelcrabb@acm.org

ABSTRACT
The Internet has become a media rich environment. It is now a major platform for video content, available to a variety of users across a range of devices. Subtitles enhance this experience for many users. However, subtitling techniques are still based on early television systems, which impose limitations on font type, size and line length. These are no longer appropriate in the context of a modern web-based culture.

In this paper we describe a new approach to displaying subtitles alongside the video content. This follows the responsive web design paradigm enabling subtitles to be formatted appropriately for different devices whilst respecting the requirements and preferences of the viewer. We present a prototype responsive video player, and report initial results from a study to evaluate the value perceived by regular subtitle users.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation (e.g.HCI)**]: User Interfaces – *User-centered design, Style guides.*
• *Human-centred computing ~Accessible technologies*

General Terms
Design, Human Factors.

Keywords
Subtitles, captions, accessibility, video, audio, web, VOD.

1. INTRODUCTION
This paper outlines an approach to the rendering of subtitles that adopts techniques used in responsive web site design. The traditional approach to formatting subtitles uses fixed size blocks of text displayed for set periods of time. This runs into difficulty if there is a need to change the line length or the size of the text.

Our responsive approach enables the subtitles to be formatted in the device to fit the display capabilities. It also provides the flexibility to respond to changes made by the user to personalise the subtitles to fit their own requirements.

This is achieved by providing the subtitles with timing for each word and formatting the subtitle blocks in the client device. This means that the block formatting can take account of the device characteristics and user preferences to provide a tailored user experience.

2. BACKGROUND
Ofcom, the UK TV regulator, reported in 2006 that about 7.5 million people had used subtitles to watch television, which equated to around 18% of the audience. Of these about 1.4 million relied on them because of hearing loss [1].

Teletext subtitles first appeared on BBC Television on 2nd September 1979 on a documentary about deaf children called 'Quietly in Switzerland' and live subtitles were first produced in 1984 for the children's programme Blue Peter. The BBC achieved 100% subtitling for all its main channels in 2008 and is now exceeding 98% for programmes on iPlayer.

Despite the move towards digital television and online distribution, the Teletext format still persists in most broadcast systems. As a result its technical limitations, such as a 38 character line length and restricted positioning [2], still constrain the way subtitles are made for television. Generally subtitles are formed as blocks of text split into 1, 2 and sometimes 3 lines, and in the UK colour is used to signify different speakers. The subtitle can be justified to the left, centre or the right of the screen and vertical position is set by the Teletext line number. Each block is transmitted in the television signal along with information as to when it is displayed and removed. Television subtitles are transmitted either as Teletext in the form of text, or as DVB Subtitles where the block is sent as a rendered image [3].

There is demand from audience and regulators to provide the ability to change the size of the subtitles and the demand is likely to increase as more people view TV on smaller screens [4]. The current approach to subtitling constrains the ability to personalise the subtitle display. In the case of text format subtitles like Teletext, increasing the font size will result in more of the video image being obscured and lines of text wrapping round in an uncontrolled manner. In the case of image files like DVB subtitles the subtitle may no longer fit on the screen.

Since the launch of iPlayer in December 2007 there has been a gradual shift of on-demand consumption of television content from PCs and laptops to smart televisions, tablets and mobile phones. By the end of 2013 viewing of iPlayer on tablets had overtaken viewing on computers [5].

The consumption of television on portable devices unlocks the potential for the viewing experience to be tailored to the individual. This comes about though two factors, the capabilities of the device, and the one-to-one relationship that exists between the device and the viewer. These portable devices are far more capable of text and graphical processing than a standard television set. This processing, combined with the association to a single user (in contrast to the often social experience of television viewing) enables adaptations to be provided for the individual's preferences.

3. RESPONSIVE APPROACH

3.1 Adopting Web Design Principles

In the world of web design, the challenge of providing web content to many different types of device has been met by an approach called Responsive Web Design [6] also sometimes called Mobile First. Responsive web sites respond to the device size and capability using a fluid approach to text layout where space is limited, whilst going to fixed column widths where there is plenty of space on the screen. This approach overcomes the problem of needing different versions of a web site for each new device, and it is the approach we have adopted to provide a more appropriate way of displaying subtitles, across a range of display devices.

We have followed practices used in responsive web design to provide a framework for responsive subtitles. This adopts the principles of text flow and line length informed by semantic markup along with style sheets to control the final rendering. The subtitle file can also include rendering information to provide recommended areas of the display in which to display the subtitle so that the subtitle does not occlude faces or graphics. The key difference from the traditional approach is that the words are formed into blocks in the client and so the number of words in the subtitle blocks will vary depending on the space available in the display.

3.2 Personalisation

Once the decision has been taken to render the subtitle layout in the client this opens the possibility of responding to user preferences at the point where the subtitle blocks are rendered. In this way the user could increase the size of the font to make the text more readable and the client would then be able to form subtitle blocks with fewer words in order to prevent the subtitles from obscuring too much of the video.

Aspects of the subtitle display such as font size, font type and text background can then be personalised to the needs of the user. These settings can then be stored alongside the user's identity and could cascade across their different devices to provide their preferred experience every time they use subtitles.

User options can thus be accommodated for a choice of font in order to meet cognitive issues such as dyslexia and font size to cope with varying visual acuity. Colour is used for speaker identification for UK TV subtitles and the choice of the four colours could be re-mapped to accommodate the user's colour vision. The text background could also be varied according to the user preferences for an opaque or semi-transparent background, or even no background with outlined text.

3.3 Key Components

3.3.1 Render Each Subtitle Blocks in the Client

The increasing capability of devices used for viewing television content enables subtitle blocks to be formatted according to a set of rules in the client. If the content of each subtitle block is created and rendered in the client device then the client can take account of the user preferences.

3.3.2 Line Length & Block Size Parameters

Line length options could be included to set maximum and/or minimum line lengths to better fit the reading style of the user. For example the viewer might prefer Teletext style subtitle lines with a smaller font and use less of the available display area. Alternatively, preferences could be set which take account of rate at which subtitles change on screen. The more frequently the subtitle block changes, the more often the user has to look down to read them.

3.3.3 Individual Word Timings

To be able to render the subtitles in the client, we need accurate timings for each word in the audio. This allows us to generate blocks of words of any length with the correct start and stop timings for the block. At the limit the client should be capable of displaying the subtitles as individual words, though this may not be a preferred display option.

In previous work we have demonstrated how we can use phonetic models to dynamically retime subtitles on a per word basis [7]. It is also possible to extract per word timings from live subtitle files, where each word is committed individually. We have also found that acceptable results can be achieved with an existing subtitle file, taking each block for which there is a start and end time and dividing the times into regular intervals to match the number of words.

3.3.4 Semantic Information

If subtitle lines and blocks are simply split on line length it can result in awkward looking blocks where subtitles start on the last word on a sentence or the second part of a compound adjective. The client will benefit from semantic markup, in addition to the sentence punctuation, to provide well-formatted blocks. This will also assist with reveal moments, such as joke punch lines to ensure that these are not displayed before the words are spoken.

Markup will also be required to indicate the presence of song lyrics or poetry where there are strong intentional distinctions between each line of text. These can then be used to override the formatting decisions to ensure that the line breaks follow the original text.

3.3.5 Display Area Information

The position of the subtitles would need to be described as a polygon that sets out the area in which the subtitles should be displayed. This information would also signify shot changes in the video, as the subtitle position needs to change along with the change of picture content. This could follow traditional subtitle placement with the option for alternative locations to provide a more dynamic subtitle experience [8].

3.4 Subtitle Enhancements

It is also possible to provide smooth animation for subtitles, such as fading each word in as it is spoken and fading the whole subtitle out in the gaps in the speech. Subtitles can also be placed in speech bubbles if the position of the mouth of the person talking is provided, see Figure 3(b). Enhancements could also provide ways for programme makers to create themed subtitles to go with programme series and individual characters. These enhancements can be provided as optional extended data and alternative style sheets.

A responsive approach can be applied to any other text data displayed alongside pictures or video. Information such as character names, mood and tone of voice could be taken from programme scripts and would assist viewers such as people with autism who already find such information very helpful in following television content via audio description [9]. Additional information might also be in the form of pop-up glosses as common in anime where cultural references are added along with the translation subtitles [10].

4. EXAMPLE IMPLEMENTATION

Our example implementation provides an enhanced HTML5 video player, using JavaScript to display the correct subtitle in time with the video and dynamically re-block the subtitles as a response to user interaction. The video player is embedded into an HTML page using two divisions. The first is contains an HTML5 video object and a second provides a container for the current subtitle. It floats on top of the video player using CSS for rendering style (For example typeface and colour) and position (defined by a polygon), as shown in Figure 1. Other functionality such as the ability to reposition the subtitle division above or below the video are also provided using JavaScript.

Figure 1 - The HTML Video player

Atomised subtitle data (see below) is stored as individual words, mapped to their display timings and speaker identification, providing a base for generating new blocked subtitles as required. If an event occurs that required the subtitles to be re-blocked (such as the video being resized, or the user changing the font size) they are recalculated and stored as HTML strings containing the text and CSS, along with start time for the entire block. In our prototype we assume that it is safe to always have a subtitle displayed and therefore don't currently store the end time, assuming that it is safe to display the subtitle until it is replaced. Should a break be required an empty subtitle can be inserted.

Functions are provided for loading subtitles from either our enhanced subtitle files, which contain per-word timing, or from conventional STL or TTML files which are atomized (see below) as they are loaded.

A JavaScript *timeupdate* event listener periodically checks the position of the video against the start times of the blocked subtitles. If the video position has passed the boundary into a new subtitle then the subtitle division is updated appropriately.

4.1.1 Subtitle Atomisation

Conventional subtitle files contain subtitles that are pre-blocked into 38 character wide lines, each with a block start and end time. As our re-blocking mechanism requires per word timings we need to be able to break the lines down into individual atoms. In previous work we have shown how this can be done phonetically in order to align each word with the audio signal [7].

However we have also found that this can be done with reasonable success by interpolating a time for each word based on its position in the block and the start and end time of the block. In the case of live subtitles, where the subtitler has updated the subtitles as each word is spoken, this can be done more intelligently. Live subtitle files usually contain both individual words, which are blocked only when the subtitler gets to the end of a line. To make use of these subtitles our approach is to load only the individual words, and disregard the repeated blocks. Both methods provide us with a good list of words with individual timings, as shown in Figure 2.

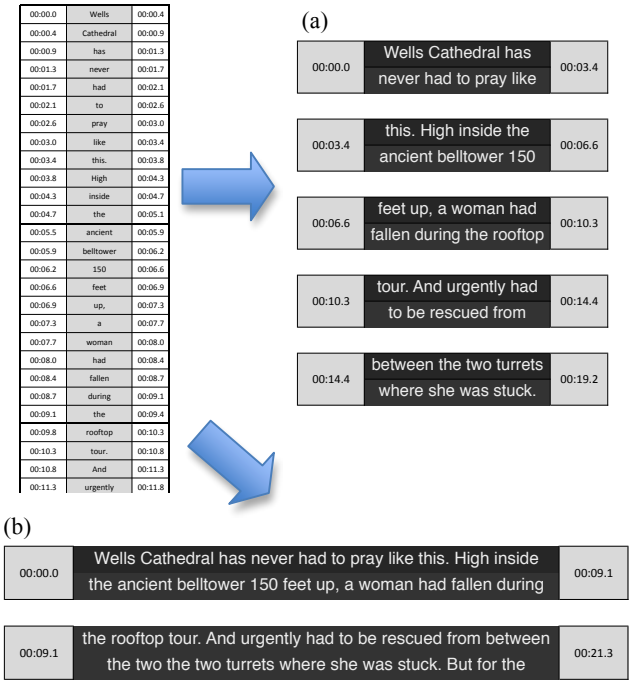

Figure 2 - The atomised subtitle are re-blocked into groups with maximum character length of (a) 22 or (b) 62

4.1.2 Best-fit

It is in the nature of responsive subtitles that the user is given control to change the font size and style that they wish to be used for displaying the subtitles as well as changing the display size. It is therefore up to the renderer to work out how many characters can be fit into each line. Proportional fonts are generally chosen for display to provide a more natural look, however different characters therefore have different display sizes. Although HTML5 provide *maxWidth* and *measureText* functions to determine the dimensions of a string without rendering it, the functionality isn't yet well supported cross browser, which means our player needs a precise way to measure fonts.

We resolve this by providing a lookup table referencing pixel width of different characters to a range of different typefaces and sizes. This was achieved by rendering each displayable ascii character into a web browser. Our tool then took a snapshot of the rendered character and image-processing tools (imagemagick.org) were used to find the tightest bounding box. The bounding boxes were calculated for each character in tern and cached. Groups of characters (such as 'a a') were also rendered to allow the calculation of the size of a space and other punctuation characters.

4.1.3 Re-blocking

The subtitles are re-blocked by adhering to the number of characters that can fit into the display division at the chosen font size. Each word is sequentially appended to a subtitle block until the next word makes the string wider than the division based on the summation of widths for each character in the string. Additional rules avoid leaving orphaned words and forcing a new line if we receive a punctuation mark over half way through the subtitle line. Breaks are always forced when there is a change of speaker. Each line is then assembled into a two line pair to complete the block as shown in Figure 2.

4.1.4 CSS Formatting

The video player uses CSS to control the display style of each subtitle. The viewer is able to select which style sheet is used to display the subtitles. This gives us an opportunity to move away from the conventional text based subtitles and provides a creative opportunity for the subtitles to be either branded alongside the content, or different accessible options provided. One example is to display the subtitles in the manner of speech bubbles, as shown in Figure 3(a).

As the Subtitle division position can also be controlled using CSS it is possible to provide additional mark up within our enhanced subtitle file, which allows a region to be defined where the subtitle should be displayed. The renderer can then find a best fit for the subtitle into the defined region. This could then be used to define clear regions of the screen where the subtitle can be displayed, or used to align the subtitle with the content as shown in Figure 3(b).

Figure 3 - (a) CSS styled Subtitles, (b) CSS positioned subtitles

5. INITIAL STUDY

A study was carried out with 26 participants to examine the personalisation of responsive subtitles. Participants were shown a series of video clips and asked to adjust the size of the fonts to a level that they were comfortable with. Participants were then asked to explain the reasons behind their choices of font size.

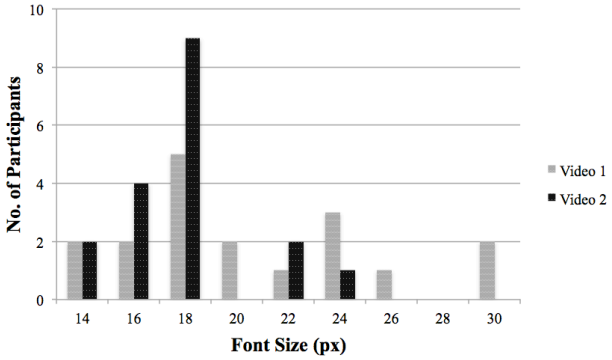

Figure 4 - Comparison of chosen subtitle size between participants across two different video clips

Initial results from this have shown that the size at which participants will set subtitles varies substantially, as shown in Figure 4. Reasons given for this by participants included both the size of the font and the amount of text being displayed in a subtitle at one time.

"I like it like this with more information. It makes more sense rather than splitting up into shorter sentences"

"Its great being able to make it bigger, its almost like I can hear it better"

This study demonstrated that there is a need to adjust font sizes and the text on screen based on individual user needs. Future work should further explore this in a more controlled manner, examining the relationship between font size and text available on screen.

6. CONCLUSIONS

This paper presented responsive subtitling, a method for displaying subtitles. This takes advantage of the principles used in Responsive Web Design. We have discussed the components needed and the challenges of enabling subtitles to be displayed in this way. A sample implementation of this method is included. Initial results from a study indicate that participants see benefit of providing personalisation of subtitles using this system and future work will examine this in more detail.

7. ACKNOWLEDGMENTS

This work has been undertaken at BBC Research and Development with encouragement and support from Gareth Ford-Williams, Head of Accessibility for BBC Future Media.

The fourth author is supported by RCUK Digital Economy Research Hub EP/G066019/1 – SIDE: Social Inclusion through the Digital Economy.

8. REFERENCES

[1] ' Television access services: Review of the Code and guidance' Ofcom, 2006, Retrieved from: http://stakeholders.ofcom.org.uk/binaries/consultations/accesservs/summary/access.pdf, January 2015

[2] 'Enhanced Teletext specification', European Standard (Telecommunications series) EN 300 706, 2003-04

[3] 'Digital Video Broadcasting (DVB); Subtitling systems', European Standard (Telecommunications series) EN 300 743 V1.5.1, 2014

[4] FCC (2008). Display of Captioning on Equipment Used to View Video Programming. , Retrieved from: https://apps.fcc.gov/edocs_public/attachmatch/DOC-314815A1.pdf, January 2015

[5] Taylor-Watt, D., 'Christmas 2013 on BBC iPlayer', BBC Blog, January 2014, Retrieved from http://www.bbc.co.uk/blogs/internet/entries/a8e5725c-76ac-32ba-8f9f-af66393cba03, January 2015

[6] Marcotte, E., 'Responsive Web Design', A book Apart, ISBN 978-1-9375571-8-8, 2011

[7] Shotton, M., 'Candidate Techniques for Improving Live Subtitle Quality', BBC White Paper 256, 2013

[8] Brooks, M., and Armstrong, M. Enhancing subtitles. In TVX2014 (2014).

[9] Garman, J., 'Autistic spectrum, captions and audio description', 2011, Retrieved from: http://mindfulresearch.co.uk/2011/08/29/autistic-spectrum-captions-and-audio-description, January 2015

[10] Caffrey, C., 'Investigating the effects of an abusive subtitling procedure on the perception of TV anime using eye tracker and questionnaire', Thesis (PhD), 2009