

Possibilities of displaying moving images in HTML5 including the limitations when compared to Adobe Flash and other Methods

Bachelor Thesis

Fabrice, Feugang Kemegni

Matriculation: 671704

Department: Applied Computer Science B.Sc.
Hochschule Worms

Supervisor:

Prof. Dr. Bernd Ruhland.

Abstract

For several years the dominant technology for displaying moving images (videos and animations) in websites was the use of Adobe Flash, however Adobe Flash has become increasingly criticized and discredited in the recent years due to its instability and its susceptibility to attacks, this to the point that the company Adobe decided to shut down its Adobe Flash Platform by 2020. The new standards of the W3C (Wide Web Consortium) i.e. HTML5 offers new features including elements that facilitate the display of moving images on webpages. HTML5 is now replacing Adobe Flash as the dominant method for the task of displaying moving images in webpages. This thesis will demonstrate the possibilities that HTML5 offers for displaying moving images as well as the limitation when compared to other technologies precisely Adobe Flash.

Content

Declaration of Authenticity.....	5
Eidesstattliche Erklärung	6
Introduction	8
Problem Statement.....	9
Goals	9
Methodology.....	9
1. Overview of moving Images.....	11
1.1 What is an Image	11
1.2 What are moving Images	13
1.3 Video Components	15
1.3.1 Codec 15	
1.3.2 Video Container.....	16
1.4 Reasons for displaying videos in Webpages	17
1.5 Videos in Webpages vs Videos in other Media.....	18
2. Displaying Videos in Websites	19
2.1 Some Video Formats	19
2.1.1 .AVI (Audio Video Interleave).....	19
2.1.2 .FLV, .SWF, .FLA (Flash Video Format, Small Web Format, Flash)	20
2.1.3 .WMV (Windows Media Video)	20
2.1.4 .MOV or QT (Apple QuickTime Movie)	20
2.1.5 .MP4 (Moving Pictures Expert Group 4)	21
2.1.6 .ASF (Advanced Systems Format)	21
2.2 Some methods of displaying videos in Webpages	22
2.2.1 Embed Tag.....	22
2.2.2 YouTube Integration.....	23
2.2.3 Facebook Video Integrated.....	24
2.2.4 Apple QuickTime Plugin.....	27
2.2.5 Windows Media Player	28
2.2.6 HTML5 Video Tag.....	30
2.2.7 Adobe Flash Plugin	31
2.2.8 Vimeo Embed Code.....	32
2.2.9 Other Video Integration Methods.....	32
3. Displaying Videos using Adobe Flash	33
3.1 Adobe Flash	34
3.2 Adobe Flash Player.....	34
3.2.1 Requirements and Installation	35

3.3	Embed a Flash Video	36
3.3.1	Acquire the Video File	36
3.3.2	Convert the File into Flash format.....	37
3.3.3	Embed the Video into the Webpage	37
3.3.4	Embed using HTMLObject	37
3.3.5	Embed using HTMLObject	38
3.3.6	Playback Controls.....	38
3.4	Embed a Flash Animation	39
3.5	Limitations of Adobe Flash	40
3.5.1	Instability	40
3.5.2	Reliability	40
3.5.3	Security	41
3.5.4	Performance.....	41
3.5.5	Complexity	42
3.5.6	Adobe is Proprietary	42
3.5.7	Compatibility Issues	43
4.	Displaying Videos using HTML5	44
4.1	What is HTML5.....	44
4.2	Using HTML to display videos on Webpage	45
4.3	Displaying Videos in HTML5	45
4.4	Displaying Animations in HTML5.....	46
4.5	Sample Program.....	47
5.	Advantages and limitations of displaying Videos using HTML5 over Flash	48
5.1	Advantages of HTML5	48
5.2	Limitations of HTML5	49
6.	Trends and usage Statistics over the Web.....	51
7.	Conclusion	53
8.	List of Figures	54
9.	Bibliography.....	55
10.	List of Abbreviations	59
11.	Attachments.....	60
11.1	The Thesis DVD	60

Declaration of Authenticity

I hereby declare that all material presented in this document is my own work. I fully and specifically acknowledge wherever adapted from other sources. I understand that if at any time it is shown or proven that I have intentionally misrepresented material present here, any decree or credits awarded to me on the basis of the material may be revoked. I declare that all statements and information contained here is true, correct and accurate to the best of my knowledge and belief.

Ludwigshafen, Winter Semester 2017/18

.....

Feugang Kemegni, Fabrice

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus anderen Schriften entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit ist noch nicht veröffentlicht oder anderweitig für Prüfungszwecke vorgelegt worden.

Ludwigshafen, Winter Semester 2017/18

.....

Fabrice, Feugang Kemegni

Introduction

The tendency for websites to display moving images has increased in the recent years to the extent that even personal websites now display videos. The dominant technique for displaying videos and animations in webpages in the past was Adobe Flash. Adobe Flash has become unstable and a constant victim of hacking and software attacks. In the recent years the W3C (World Wide Web Consortium) published the latest the latest version of its standard namely HTML5. HTML5 comes with advanced features and tools including elements that greatly simplify the display of videos and animations on HTML pages (webpages). With these new features, HTML5 has been increasingly gaining grounds in replacing Adobe Flash as the main method to display moving images in websites.

The goal of this thesis is to demonstrate the possibilities that the HTML5 offers for displaying videos and animations (moving images) on websites, and to point out some of its limitations when compared to other methods of displaying moving pictures, particularly Adobe Flash.

In order to achieve this goal, this thesis has been subdivided into chapters and sections. Each chapter builds upon the preceding one and provides basis to understand the following one. Chapter 1 introduces a brief discussion of what an image means and what a video means in the domain of web applications, then discuss some of the components of a video, followed by the reasons why videos and animations are been embedded in webpages. Chapter 2 discusses some common video formats used in websites today and what technologies are required to play them. This is followed by a discussion on some popular methods of displaying videos on websites. Chapter 3 discusses Adobe Flash, and how it is used to embed videos in websites, as well as its limitations and some of the reasons that led to its decline. Chapter 4 discusses HTML5, what it is, how it is used to embed videos on a webpage. Chapter 5 highlights some advantages and imitations of using HTML5 to display videos in webpages as compared to Adobe Flash.

Chapter 6 discusses some usage statistics over the web, what percentage of applications use HTML5, or Adobe Flash to embed their moving images, and how the tendency is growing. This chapter is followed by the conclusion, chapter 7 where I state my apprecia-

tion of the tendency, as well as a little analysis on how the tendency is likely going to grow and why.

Problem Statement

Adobe Flash The dominant method for embedding videos on websites has been in the recent years increasingly criticized due to its instability, it's poor performance on mobile devices and its susceptibility to software attacks. Recently HTML5 has been published as the latest version of HTML, offering new features, including the ability to display videos in webpages. The question is: To what extent can HTML5 replace Adobe Flash in the task of displaying moving images on websites and what are the limitations when compared to Adobe Flash.

Goals

In this thesis, I will demonstrate what possibilities HTML5 offers for displaying moving images in websites, and what are the limitations when compared to other techniques mainly Adobe flash.

Methodology

In the course of developing this subject and solving the task on hand, I will carry some researches through reading from several sources including books, internet forums and websites and I will carry out tests myself on my computer to prove or disprove and support the information obtained from the searches. The principal source of information for this work is the internet because one doesn't learn how to climb a tree from a fish, but instead from a monkey. This means, because the topic of the thesis is essentially web related, the best source of information is the web.

All code snippets found in this thesis except where otherwise stated, have been written by myself, following the instruction found on sources that are stated.

1. Overview of moving Images

This chapter discusses some brief definitions about the structure and main components of a video which are images. The first section discusses what an image is and what it is composed of, the second section build on what have been learn about an image to describe what a video is and the components of a video. The last section of this chapter discusses the reasons why videos are being embedded in websites.

1.1 What is an Image

The definition of an image varies deeply across domains and fields of studies. In computing and image processing, an image is a two-dimensional signal or a mathematical function $f(x, y)$ where x and y are the horizontal and vertical coordinates of pixels. The value of $f(x, y)$ at any point gives the pixel value at that point of an image (www.tutorialspoint.com). The pixel carries the value of a colour, intensity as well as some other image information. Figure 1 below explains the relationship between an image and its pixels.

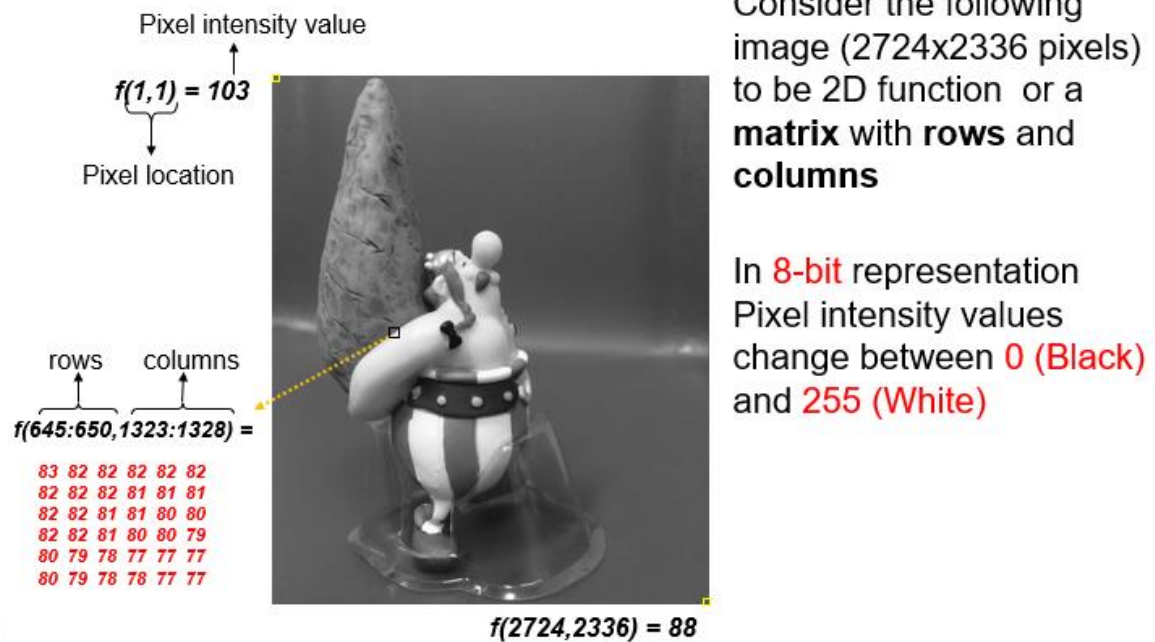


Figure 1 : Digital Image

This a 2750x2336 Image viewed as a 2D function. The little yellow box at the top left of the image represents a pixel at coordinate $f(1,1)$ and the value 103 which represents a grayscale or colour. In this example the pixels are 8-bit, which means the pixel values range between 0 to 255 (decimal representation of 8-bit range).

Reference

(Gholamreza 2015).

A detailed discussion on image component – pixel is out of the scope of this thesis. To avoid ambiguity, the terms “image” and “picture” are used interchangeably to refer to the same thing, hence “moving image” means “moving picture” and vice versa.

1.2 What are moving Images

In this thesis, the terms video, moving images, moving picture, and animation are used interchangeably to mean the same thing. A moving image or moving picture is a term that encompasses videos, animations and other files that display images changing over time. A video could be understood as a stack of superimposed images, each image in the stack is marked with a timestamp in the time interval. Supposing a stack of 60 images, and a time interval of 60 seconds, placing these stacked images in a device that will display each image of the stack at the specified timestamp will result in a video. Such a device is called a video player. The result of displaying these images at the stated timestamp is what we see as a video. Hence requesting a particular image to the playing device by providing its timestamp is the equivalent of jumping to a time interval in the video just like what we do when we fast-forward a video in YouTube or VLC player. Figure 2 below illustrate a basic video consisting of stacked images.

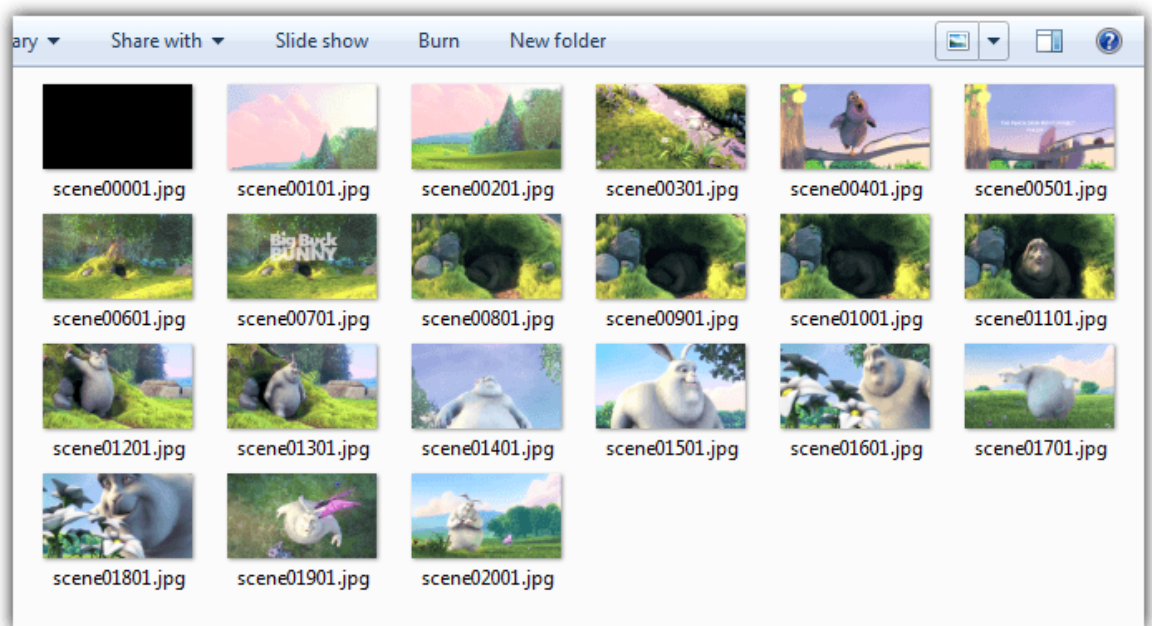


Figure 2: Video Frames

The 21 Pictures/Images in the figure will constitute frames when grouped together to form a video. Each of the images (frames in the figure above could be marked with a timestamp e.g. 1 second, this can then be viewed as a video 21 seconds long.

Reference

(Raymond.cc, 2016)

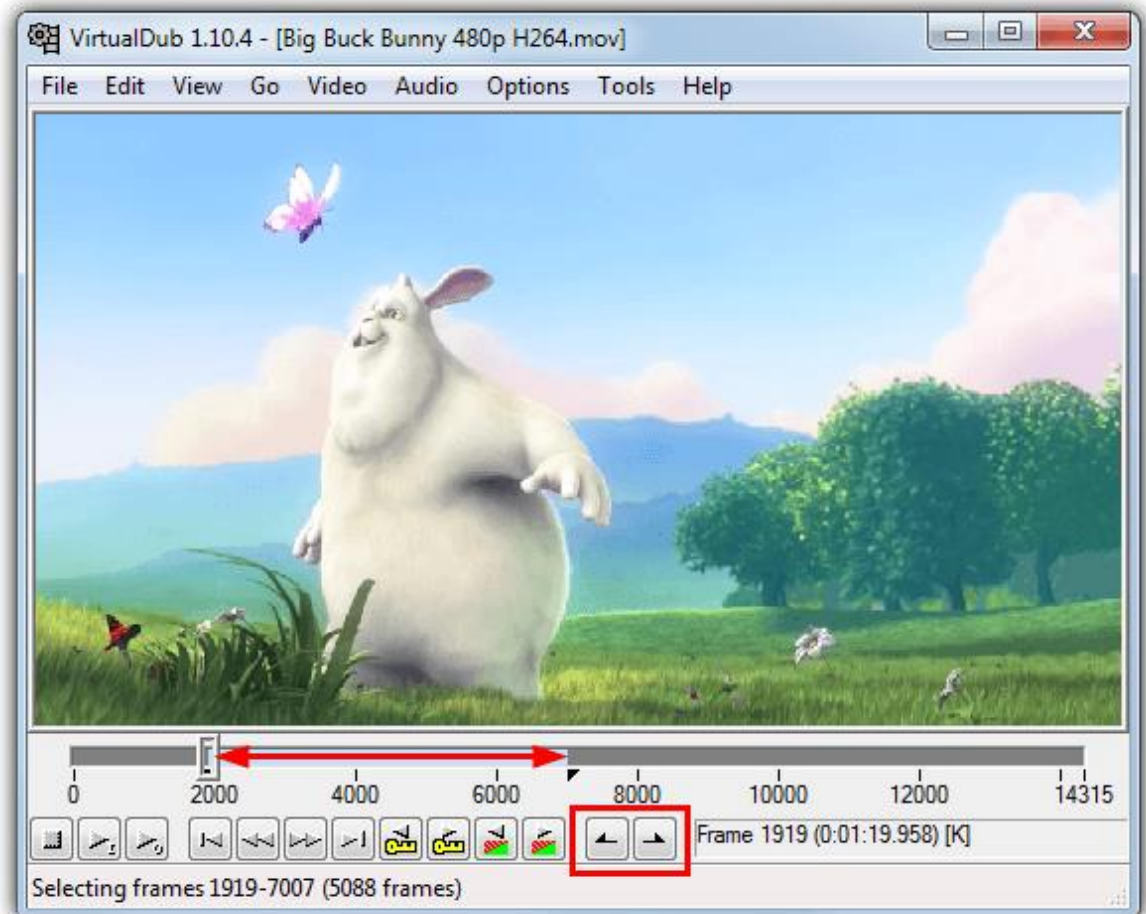


Figure 3: Frame Clipping

The Figure illustrate how images/frames are stacked together to produce a video.

Reference

(Raymond.cc, 2016)

Each image in such a stack/video is called a frame. Typical videos are around 24-30 frames per second. This means 24-30 images are displayed each second, where the term Frame Per Second (FPS) (Raymond.cc, 2016) Video files simply store all frames together and a video player plays them in order.

With such a description of what images and videos are, it is now possible to illustrate some of the components or elements of a video.

1.3 Video Components

A video is stored in a compressed form to reduce the amount of storage space it may use. A video file typically consists of a video container. The video container consists of video data in video coding format, audio data in audio coding format as well as other synchronization information like subtitles, metadata such as title etc. There are hundreds of types of videos and video formats and each type/format serves a purpose better than the others. The video format is the structure in which a video is stored in a computer file. A video can be converted from one format into another format using algorithms (Imagen, 2015).

1.3.1 Codec

A codec is a computer algorithm that encodes/decodes or compresses/decompresses digital stream data like videos or audios, it also interprets the data stream (e.g. video/audio) and determines how best to play the video on the output medium (Videomaker.com, 2010). Electronic devices capable of displaying videos all come with some preinstalled codecs. Computers for example come with some codecs and users are often required to download and install some additional codecs, this is often the case when the computer encounters errors like “file extension not supported” or “cannot play media”.

1.3.2 Video Container

A video container is a bundle of media files, often consisting of a video and an audio codecs and sometimes some additional information like subtitles and captions (Imagen, 2015). Containers allow users to choose one codec for the video and another codec for the audio.

1.4 Reasons for displaying videos in Webpages

There are hundred reasons why videos are being displayed on websites, below are some of these reasons:

Shopping websites must display images and videos on their websites so as to show to the potential customers what their products and services look like. An example here is in the vacation industry, where videos of beaches, hotels, mountains, cruising tours must be shown to users to motivate them into committing to carry purchase.

If not displaying an unpurchased product, companies may also need to illustrate how the customers should or could use their purchased products. These include videos of installation guides, after sale supports where a video featuring a trained personal demonstrating how to assemble a chair, bicycle, a tent or any product that could have been purchased in a disassembled state and requiring the customer be assembled the product themselves.

The academics also make use of videos in their webpages. This could be videos for tutorials, videos of welcome, new Students orientation, visitor's orientation etc. Thus, requiring videos to be embedded in their websites.

Television channels also have some branches that are online and requiring live broadcasts or archives or shows. These either require streaming of videos either live streaming or not. Viewers have the opportunity to choose among viewing their favourite shows live on Television or on web browser. This increasing the need to display videos on webpages.

Video sharing websites make extensive use of videos in their websites and these videos constitute a great percentage of the content of their websites. YouTube, Netflix, even Facebook make extensive use of videos.

Videos are also used as advertisements in websites. These often come in the form of animations and GIFs (Graphics Interchange Format) and other forms that involve the display of moving images.

As stated before, the list of reasons why websites may contain videos is endless, above the above points only provide some common reasons why videos may be found in a website.

1.5 Videos in Webpages vs Videos in other Media

Videos in a website differ from videos in other media in several ways. One of the main difference is that a video in a website requires internet connection and buffering. A website requires the video to be buffered, i.e. a certain percentage of the video must be loaded before the video starts playing. This in comparison to a video played on the computer from a video player does not require the video to be buffered, the video is instantly played. Web videos require internet in order to load the video content from the source or server to the user.

Refreshing the browser page makes the video restart from the beginning, and the amount of already buffered video is lost. The video restarts each time the connection to the internet is lost. This means if you are watching a video over a web browser, then the internet goes off and on, the video needs to restart as the browser needs to re-establish a connection with the website.

Low bandwidth leads to poor viewing quality as the video will frequently freeze while it is being buffered, this may lead to inconveniences and bad user experience compared to a video been viewed from a video player program on the computer.

Displaying a video on a website requires the dimensions of the video to be customized. Not all the videos are being embedded the same way in to a website. The videos width, height, scale and other parameters need to be edited, otherwise the video will be either cropped, hidden or wrongly scaled, distorted. In comparison to playing a video over a video player in the computer or television, where no additional setting is required for videos of different sizes. The video sizes and ratios are automatically adapted to the display.

These differences highlight the attention required for displaying videos in websites, which make it a little more difficult. In the next chapter, I discuss some video formats used over the internet and how they differ among each other and some of the methods used to display videos on webpages.

2. Displaying Videos in Websites

In the previous chapter, I defined what a video in the context of computing is, I also highlighted the structure of a video and gave some reasons why videos are displayed in websites. This chapter discusses some common video formats in use in web applications, and some popular methods used in display videos in websites.

2.1 Some Video Formats

As previously introduced, there are dozens of video formats, each with its characteristics, abilities and each being appropriate for a specific purpose. Video formats are often referred to as video containers. Below are 6 commonly encountered videos formats used in websites (Imagen, 2015). A more comprehensive list of video formats can be found under: <http://websitehelpers.com/video/formats.html> https://developer.mozilla.org/en-US/docs/Web/HTML/Supported_media_formats.

2.1.1 .AVI (Audio Video Interleave)

Developed and published by Microsoft in 1992 AVI is one of the oldest video formats. Due to its simple architecture, AVI files are able to run on several architectures like Windows, Macintosh, Linux, and also supported by popular web browsers. AVI files are opened with: Microsoft Windows Media Player, Apple QuickTime Player, VideoLAN VLC, media player as well as several other media players (Sharpened Productions, 2010).

2.1.2 .FLV, .SWF, .FLA (Flash Video Format, Small Web Format, Flash)

These are files that are encoded by the Adobe Flash Software, and can be played via the Adobe Flash Player, Adobe Flash web browser plugins and several third-party programs. Since virtually every browser has the Adobe Flash plugin installed, it has become the most common online video viewing platform on the web. Almost all video sharing sites including YouTube has been streaming videos in Flash format. The Flash video format is also what many video-sharing sites convert videos to, from formats that were uploaded by their users in something other than Flash. This is because videos in the flash format remain in high quality even after compression to a smaller file size, which means that the videos on the web load quickly and won't spend a lot of time using up bandwidth.

2.1.3 .WMV (Windows Media Video)

Developed by Microsoft, it was originally designed for web streaming applications, but can now cater to more specialized content. WMV files are the tiniest video files over the Web, as their file size decreases significantly after compression, resulting to poor video quality. However, this small file sized format allows users to upload and share their videos through the e-mail system.

Being a Microsoft software, the Windows Media Player is the main application that is used to play WMV files on all Microsoft's Windows operating systems, but there are also WMV players available for free for the Macintosh operating system and other platforms (FileInfo, 2016).

2.1.4 .MOV or QT (Apple QuickTime Movie)

Developed by Apple. Inc, the QuickTime file format is a popular type of video sharing and viewing format amongst Macintosh users. It is often used on the Web, and for saving movie and video files. There is a free version of the QuickTime Player available for

the Windows Operating System among many other players. MOV files are of high quality and are usually big in file size (Movie, 2017)

2.1.5 .MP4 (Moving Pictures Expert Group 4)

MP4 is an abbreviated term for MPEG version 4, a standard developed by the Moving Pictures Expert Group (MPEG) who was responsible for setting industry standards regarding digital audio and video, and is commonly used for sharing video files on the Web. This video format uses separate compression for audio and video. The video track is compressed with MPEG-4 or H.264 video encoding, while the audio track is compressed using AAC compression. It is also a great file sharing format for the Web as MP4 file sizes are relatively small but the quality remains high even after compression (File, 2016).

2.1.6 .ASF (Advanced Systems Format)

Another offering from Microsoft, the ASF container normally houses files compressed with Microsoft's WMA (Windows Media Audio) and WMV (Windows Media Video) codecs. Just to confuse the matter further, the files are usually given the .wmv or .wma suffix and not the expected .asf. This file format opens with: VideoLAN VLC media player and Microsoft Windows Media Player, as well as some other third-party video players (FileInfo, 2016).

These 6 Video formats listed above is not the complete list of all video file formats available, instead a list frequently encountered formats over the web. There are many more formats available, but listing all of them is out of scope for this thesis. The idea of listing these is just to familiarize the reader with some terms that will reoccur in future chapters.

2.2 Some methods of displaying videos in Webpages

This section describes some popular methods used to display (embed) videos and animations in websites. Each way you embed your video will depend on the video format on hand. Most of the formats referred here have been discussed in the previous section. The desired format may require the use of a video player or plugin to be able to play the video. Video players are easily downloadable over the Internet. Below are some popular methods for embedding videos into websites (Freemake.com, 2017). Some additional methods for displaying videos in webpages are summarized by (Ronald, P, 2012).

2.2.1 Embed Tag

This method is frequently used for short videos (about 10 seconds long), and when the bandwidth and buffering times is no issue. Here only the html embed tag is required including the source of the video. The browser will handle the rest. The code below illustrates how to embed a video using the html embed tag.

```
<embed src="my-mp4.mp4"
       width="500"
       height="auto">
```

HTML Embed Tag



```
<embed src="videos/mp4.mp4"
width="560"
height="316">
```

Figure 4: Displaying a Video using html embed tag

The only requirements are the relative path of the video and the dimensions (also optional). To avoid distortion of the video, it is advised to set only one attribute of the video (only the height or only the width) in the tag, then set the value of the other attribute to 'auto' in the stylesheet file, especially when you are not sure of the exact dimensions of the video.

Source

own construction.

2.2.2 YouTube Integration

YouTube is often used to embed videos when the normal embed tag takes too long to load the video. This method is fast and free, but requires that the video be first uploaded to YouTube, then referencing the id of the uploaded YouTube video on your website. To use this method, you need to create a YouTube account, then upload the video to this account, YouTube automatically generates the embed Tag that you need to insert in your website code.

The trade off with this method is that you lose all control over your video, as anyone else on youtube can freely download and also embed your video to their site.



Figure 5: Embed a Youtube Video

Once you have uploaded a video to youtube, youtube offers the possibility to embed the video to your website, right click on the video and choose "Copy embed code", paste the content of the link into your website code.

Source

own construction.

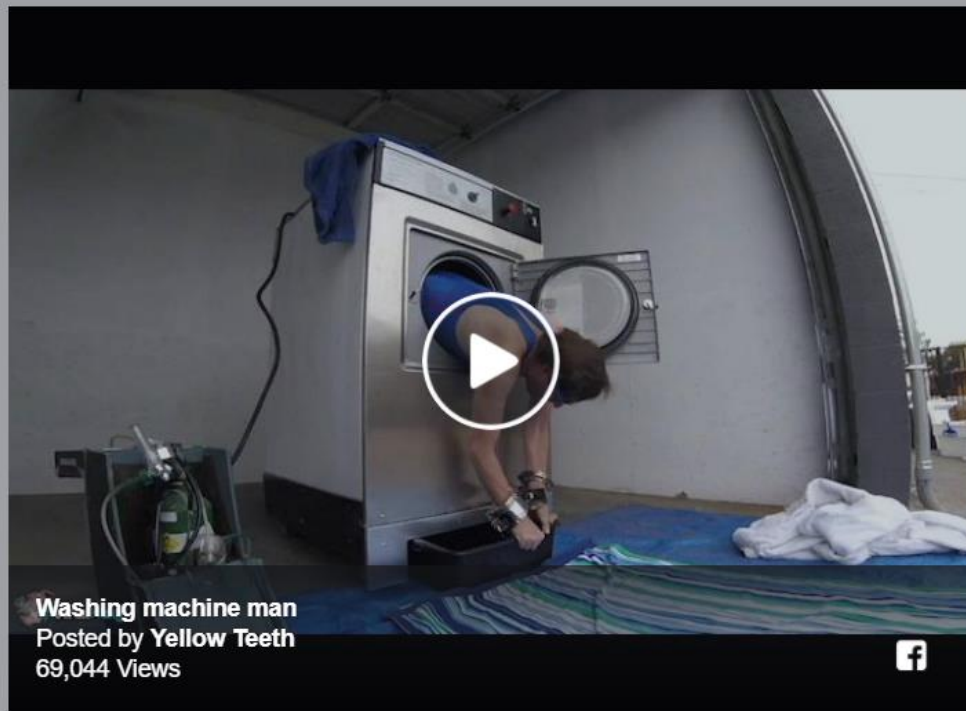
2.2.3 Facebook Video Integrated

Another method to display a video on a webpage is using the Facebook's embed method. The problem of privacy pops up here because any one can embed any Facebook

video on his website. It is very simple to embed a Facebook video on a webpage. To do that, go to Facebook and play the video you want to embed on your website. Every video on Facebook has an option at its top left corner, click on the option button and choose 'Embed', a dialog opens where you can copy the html tag of the video, you then paste this code in your website. The code snippet below shows how to embed a Facebook into a website. This is just the code copied from Facebook, and the figure following the code snippet shows how the embedded video is displayed on the browser (Facebook Developers).

```
<iframe
src="https://www.facebook.com/plugins/video.php?href=https%3A%2F%2Fwww.facebook.com%2Fyellowteethpage%2Fvideos%2F1468747389909130%2F&show_text=0&width=560"
width="560" height="411"
style="border:none;overflow:hidden"
scrolling="no"
frameborder="0"
allowTransparency="true"
allowFullScreen="true">
</iframe>
```

Facebook Video



```
<iframe src="https://www.facebook.com/plugins/video.php?href=https%3A%2F%
width="560" height="411"
style="border:none;overflow:hidden"
scrolling="no"
frameborder="0"
allowTransparency="true"
allowFullScreen="true">
</iframe>
```

Figure 6: Embedding a Facebook Video in to the website

While viewing any video on Facebook, the video has an 'option' button at its top corner, click on this button and choose 'Embed', a dialog opens where you can copy the html tag of the video, you then need only to paste this code in your website.

Source

own construction.

You can then put the code on any webpages and the Facebook video will play. The video player size can optionally be changed, as well as other parameters like auto play or loop (Facebook Developers).

2.2.4 Apple QuickTime Plugin

The html `<embed>` tag allows media file types other than those directly supported by the browser to be handled with an external application or plugin. In this case, the external application is QuickTime Player and the plugin is the QuickTime Plugin (Apple Support, 2016).

To embed a video on your website using this method, you can either download the QuickTime plugin and place it in your website folder or you can reference the link of the download source of the player in the html options. The code snippet below demonstrates the code required to embed a video on a webpage using QuickTime. The code snippet is followed by that shows how the embedded video looks like. Additional instructions on how to install the Apple QuickTime Player can be found following this link:

[how to install or uninstall QuickTime](#). The QuickTime plugin that you need to download in to your website can be downloaded under:

https://support.mozilla.org/en-US/kb/quicktime-plugin-play-audio-and-video#w_testing-quicktime.

Below is a code snippet of how to embed a video to be played by the Apple QuickTime Player. The Browser output is displayed in the figure following the code snippet.

```
<object width="560" height="315" classid="clsid:CFCDAA03-8BE4-11cf-B84B-0020AFBBCCFA">
  <param name="controls" value="ImageWindow" />
  <param name="autostart" value="false" />
  <param name="play" value="false" />
  <param name="start" value="false" />
  <param name="src" value="videos/mov.mov" />

  <embed src="videos/mov.mov"
    width="560"
    height="auto"
    autoplay="false"
    controller="true"
    play="false"
    start="false"
    type="application/video/quicktime"
    pluginspage="http://www.apple.com/quicktime/download/">
</object>
```

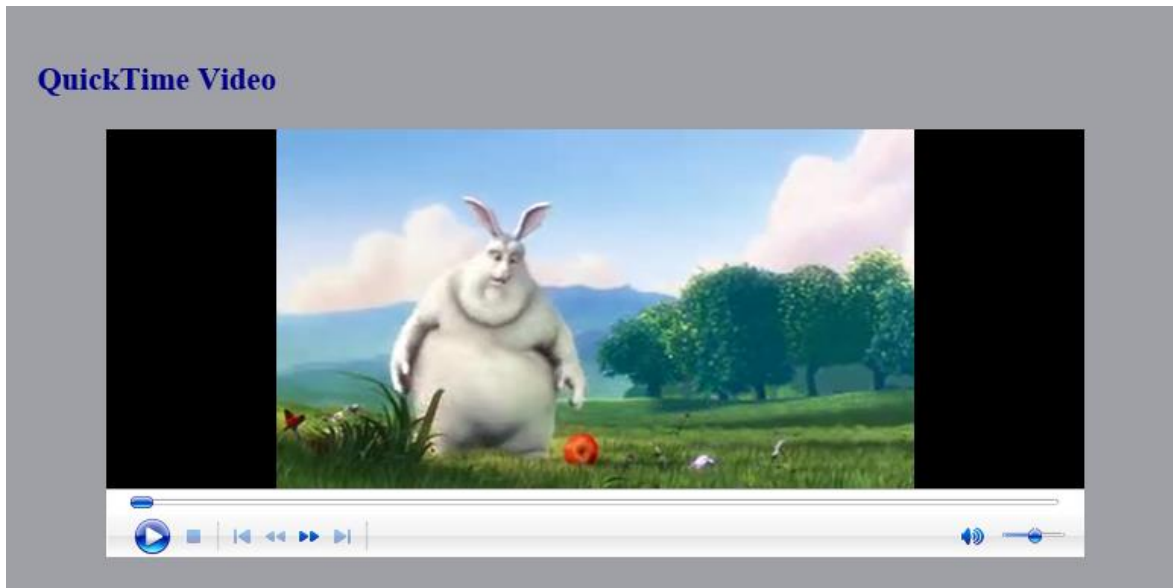


Figure 7: Output of video embedded using Apple's QuickTime.

The Apple Quicktime player comes with its controls, therefore viewers can directly interact with the video by forwarding, rewinding, pausing the video.

Source

own construction.

2.2.5 Windows Media Player

This is typically used to embed videos of format .wmv. To embed videos using this method, you either need to download the Windows Media plugin and place it in the website folder, or reference the link of the player in the attributes of the embed tag (Media-college.com).

The snippet below shows the code required to embed the video file named wmv.wmv found in the website repository. The figure following the snippet demonstrates the browser output.

```
<OBJECT id='mediaPlayer1' width='180' height='50'
  standby='Loading Microsoft Windows Media Player components...'
  type='application/x-oleobject'>
  <param name='fileName' value='Stream URL or Full File Path Goes
Here'>
  <param name='animationatStart' value='true'>
  <param name='transparentatStart' value='true'>
  <param name='autoStart' value='false'>
  <param name='showControls' value='true'>
  <param name='ShowAudioControls' value='true'>
  <param name='ShowStatusBar' value='true'>
```

```
<param name='loop' value="false">
<EMBED type='application/x-mplayer2'
  plugin-
spage='http://microsoft.com/windows/mediaplayer/en/download/'
  id='mediaPlayer'
  name='mediaPlayer'
  displaysize='4'
  autosize='-1'
  showcontrols="true"
  showtracker='-1'
  showdisplay='0'
  showstatusbar='-1'
  videoborder3d='-1'
  width="420"
  height="380"
  src="videos/wmv.wmv"
  autostart="true"
  designtimesp='5311'
  loop="false">
</OBJECT>
```

The browser's output is shown in the figure below.

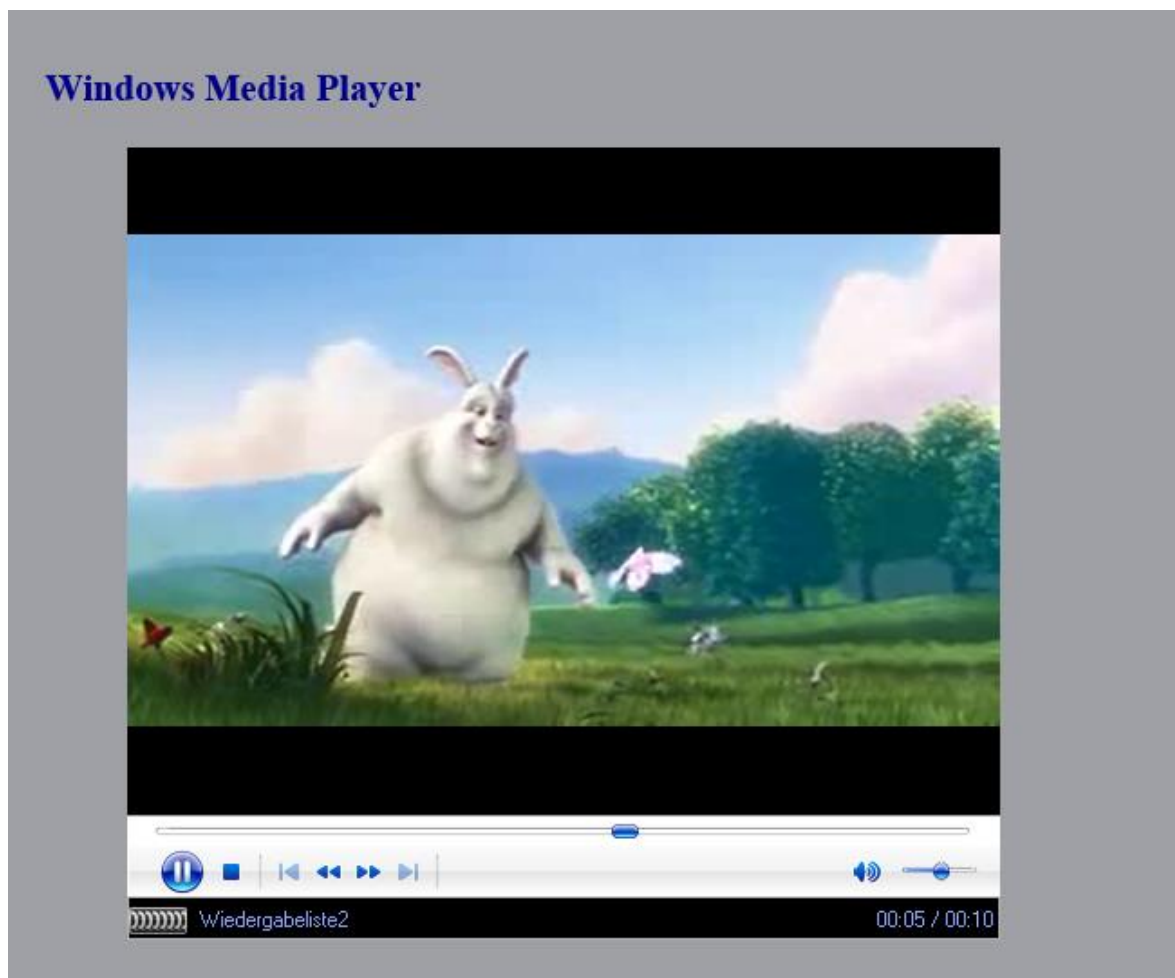


Figure 8: Embedding a Video using the Windows Media Player Plugin.

Windows Media Player plugin also comes with Playback controls to interact with the video.

Source

own construction.

2.2.6 HTML5 Video Tag

HTML5 is the most straightforward method. The new <video> tag in HTML5 mark-up allows web developers to add videos into a webpage without any special plugins. Below are the code snippet and the browser output when embedding a video using the HTML5 method (w3schools.com)

```
<video id="html5-video-player"
  controls
  autoplay
  loop
  muted
  preload="none"
  poster="images/icon.jpg"
  width="560" height="315"
>
  <source src="videos/mp4.mp4" type="video/mp4">
  <source src="videos/another_video.mp4" type="video/mp4">
  This is the fallback text, in case the browser doesn't support HTML5.
  e.g.
  Your browser does not support the video tag.
</video>
```

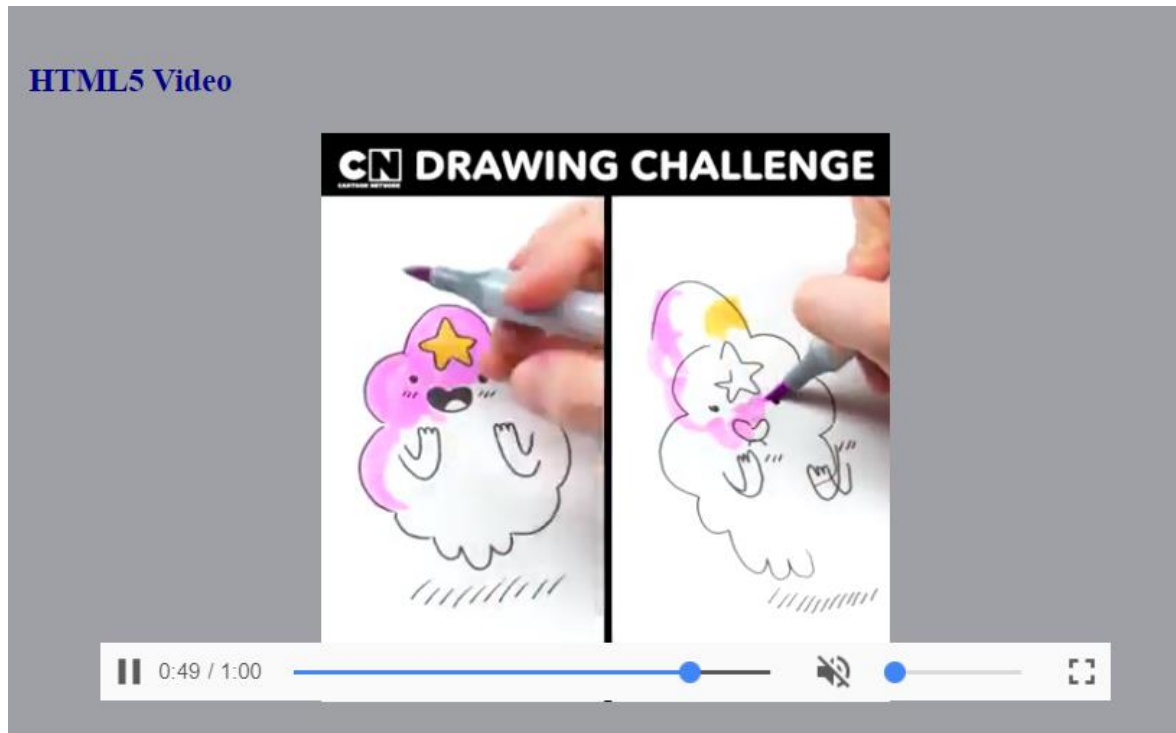


Figure 9: Embedding a Video using the HTML5 method.

HTML5 tag provides Playback controls as well as a poster attribute, this poster is an image that will poster the video on the player once the video is stopped.

Source

own construction.

A detailed discussion of this method is presented in section 4.3.

2.2.7 Adobe Flash Plugin

To embed a Flash video (.flv, .fla, .swf), you either need to install the flash player plugin and place it in your website's folder or simply reference the link of the Flash Player as an attribute in the ebbbed tag. Adobe Flash player plugin can be downloaded in the following link <https://get.adobe.com/flashplayer/otherversions/> (Help.adobe.com)

A Detailed illustration of embedding videos on a webpage using the Adobe Flash Method is presented in the section 3.3

2.2.8 Vimeo Embed Code

Vimeo is YouTube's main rival, especially in the area of video embedment options. Vimeo videos are also viewed by millions of web users and can be easily integrated into other websites or blogs. Vimeo player is very handsome and streamlined. The blue "Embed" button in the upper right of the video leads to a dialog box with the embedding options. Basic account holders are allowed to customize player size, text colors and add other elements, while Vimeo Plus members enjoy the complete control over the player, including options to choose what happens next and where the video can appear (Help Center, 2016).

2.2.9 Other Video Integration Methods

There are several other Methods for displaying videos in websites, most of which are web based and provide step by step instructions to even novice users on how to embed a video to their website. Some of these methods include:

<https://oembed.com/>

<http://www.cocoonsoftware.com/en/>

<http://www.freevideocoding.com/>

<http://videolightbox.com/>

<http://easyhtml5video.com/>

3. Displaying Videos using Adobe Flash

As stated in the problem statement and goal of this thesis, Adobe Flash and HTML5 are the point of interest. The previous chapters have already provided some foundations on which to build upon. It is already clear what a video is, types and formats of videos, why and how to embed a video in a website.

This chapter provides a detailed discussion on Adobe Flash. In the first section, I describe Adobe Flash player and how to use it, in the second section I provide a sample program that illustrate how to use Adobe Flash player to displays a video in a website.

3.1 Adobe Flash

Adobe Flash consists of 2 platforms, the Flash Player and the Adobe Air. Flash Player runs on web browsers while Adobe Air runs outside browsers. Adobe Flash display content such as web applications, interactive online advertising and animations created with Flash Professional and Adobe Flex (Adobe.com, 2017).

Another variant is the Adobe Shockwave player which differs from Flash Player in that Shockwave player displays content such as high-performance multiuser games, interactive product simulations, online entertainment and training applications created with Adobe Director software (Adobe.com, 2017). The Flash Platform displays text, vector graphics and raster graphic to provide animations, video games and applications. It allows streaming of video and audio and can capture mouse, keyboard, microphone and camera input. Artists may use the Adobe Animation to produce Flash Graphics, Software developers may use Adobe Flash Builder, Flash Catalyst and Flash Develop or any text editor combined with the Apache Flex SDK to produce applications and video games.

End users can view Flash content via Flash Player. Adobe Flash Lite enables viewing Flash content on older smartphones.

3.2 Adobe Flash Player

Adobe Flash Player as described above is a video player plugin for web browsers. In most web browsers it is built in, so users do not need to install nor to activate it. It may be downloaded under: <https://get.adobe.com/flashplayer/otherversions/>. It is a freeware to use with software created on the Adobe Flash Platform. These include viewing multimedia and executing internet applications, streaming videos and audios. Adobe Flash can run on a web browser as a plug-in or can be ran on supported mobile devices. Adobe Flash plays file formats FLA, FLV, SWF (Adobe.com, 2017).

3.2.1 Requirements and Installation

Adobe Flash player can be download in the following link:

<https://get.adobe.com/flashplayer/> (Helpx.adobe.com, 2017).

The systems requirements are also displayed on the requirement's page:

<http://www.adobe.com/products/flashplayer/tech-specs.html> . These requirements include:

Windows

- Microsoft Windows: 32-or-64 bits
- Microsoft Windows xp 32 bits,
- Windows 7, Windows 8.x and Windows 10
- Latest versions of Internet Explorer
- Microsoft Edge
- Mozilla Firefox
- Google Chrome and Opera

Mac OS

- Latest versions of Safari, Mozilla Firefox, Google Chrome, and Opera
- Mac OS X v10.9, or later

Linux

- YUM, TAR.GZ, RPM and APT packages for NPAPI and PPAPI
- Latest versions of Firefox or Google Chrome

As stated on the download page, in most browsers the plugin is built-in but may be disabled. In case the plugin is disabled on a browser, it may be enabled following the steps under: <https://support.google.com/chrome/answer/6258784>.

The first section of this chapter has provided some information about what adobe flash player is and how it may be acquired. However, users often do not need to install nor to enable the Flash Player because it is preinstalled in the browser. In case the Flash Player

is missing, it is very easy to install it because an error message is always shown to the user together with the steps required to install the browser, which the user simply needs to follow.

In the following section, I will go into details on how to display a video on a website using Adobe Flash Player.

3.3 Embed a Flash Video

In the Attachment submitted with this thesis is a copy of a website that contains all samples of the methods of embedding videos in a website. This section will provide a step by step explanations on how to embed a Flash video i.e. embedding a video using the flash method.

3.3.1 Acquire the Video File

The first step is to acquire the video. The video needs a scale and some ratio, otherwise it will not be displayed correctly in the browser. Videos often occupy the whole screen width in a website and videos are rarely square in shape. The best way to go is to configure the camera to have some proportions. 4:3 is widely used, this implies the width 400px width and 300px height and multiples of these.

In case you are not capturing the video yourself but are using an available video, you may need to resize the video so as to obtain such scales. This can be done online through free online video resizers like: <https://ezgif.com/resize-video> or free software like Filmora video editor: <https://filmora.wondershare.com/video-editing-tips/how-to-resize-video-files.html> .

3.3.2 Convert the File into Flash format

Adobe Flash Player recognises files with the following formats (discussed in section 2.1.2 .swf, .fla and .flv, so the video to be embedded must be converted into one of these formats. This can be done using Adobe Dreamweaver:

<http://www.adobe.com/products/dreamweaver/free-trial-download.html>

or with the help of any video converter software like Format Factory: <https://format-factory.en.softonic.com/download>. Some Video recording devices like mobile phones and digital cameras can also directly convert videos to the desired format.

3.3.3 Embed the Video into the Webpage

Once the video is ready i.e. scaled and converted it can be embedded into the website (Help.adobe.com) . The code snippet below shows how to do that. The result of doing so can be viewed in the program submitted with this thesis.

3.3.4 Embed using HTMLObject

```
<object type="application/x-shockwave-flash" height="auto" width="100%">
  <!-- link to the video-->
  <param name="movie" value="videos/my-mp4.swf"/>

  <!-- Sets the Window Mode property of the see documentation-->
  <param name="wmode" value="opaque"/>

  <!-- play the video automatically ?-->
  <param name="play" value="false"/>

  <!-- Should the video restart once it is over? -->
  <param name="loop " value="false"/>

  <!-- should playback controls be displayed? -->
  <param name="menu" value="true"/>

  <!-- Trade-off between performance and video quality-->
  <param name="quality" value="best"/>

  <!-- Should the Flash Player be able to occupy the whole screen-->
  <param name="allowFullScreen " value="true"/>
</object>
```

3.3.5 Embed using HTMLObject

This is fundamentally the same method, with the exception that the actual video is placed in the html embed tag which is surrounded by the HTML Object Tag. The distinction between the two methods is not clear, however using one or the other method is a question of programmer's taste (Help.adobe.com) The code Snippet below shows how this is done.

```
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
  code-
  base="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.c
  ab#version=7,0,0,0"
    id=0 align="middle" width=100% height="auto">
      <param name="menu" value="true" />
      <param name="quality" value="high" />
      <param name="bgcolor" value="#FFFFFF" />
      <param name="wmode" value="transparent" />
      <param name="controls" value="true"/>
      <param name=movie value="videos/.my-mp4.swf">
      <param name=allowFullScreen value=true>
      <param name=flashvars value="videos/my-mp4.swf">
      <embed src="videos/my-mp4.swf"
        wmode="transparent"
        play="false"
        menu="true"
        quality="high"
        bgcolor="black"
        width="100%" height="auto"
        name="player"
        align="middle"
        allowScriptAccess="sameDomain"
        type="application/x-shockwave-flash"
        pluginspage="http://www.macromedia.com/go/getflashplayer" >
    </object>
```

A list of supported parameters/attributes and their corresponding values can be found here
<https://helpx.adobe.com/flash/kb/flash-object-embed-tag-attributes.html>

3.3.6 Playback Controls

Playback controls include buttons like buttons to pause/play, stop, resume, forward, mute etc. By default, the embedded video doesn't display the playback controls event

when you set the parameter

```
<param name="menu" value="true"/>
```

The controls can be views by right clicking on the video. Thus, the video will automatically start playing when the webpage has finished loading (it will not if the parameter play=false had been given).

The playback controls can be included with the video in one of these 3 ways

1. Embedding the video directly from adobe Dreamweaver.
2. Converting the video and including the controls directly.
3. Program the control using JavaScript.

3.4 Embed a Flash Animation

Flash animations are created using Adobe Animate CC that can be downloaded under: <https://creative.adobe.com/products/download/animate>. Once Adobe Animate CC is installed, create your animation and save under the format SWF or FLA or FLV. To embed the animation, follow the same steps as in section 3.3 above.

3.5 Limitations of Adobe Flash

Using Adobe Flash layer to embed videos in to webpages is one of the most popular methods over the internet. Until recent years it was undoubtedly the most efficient way to do that. However, because of some of its limitations, it has been increasingly criticized, and its popularity has fallen. Some of the drawbacks of using adobe flash player include the following.

3.5.1 Instability

Users have been constantly complaining that their computer or browsers crashed when viewing flash content. With each sets of complaints, Adobe often came out with some software patches and updates to fix the problems. Some of these complaints can be viewed in the adobe forums. Some users in the (Forums.adobe.com, 2015) complain that Adobe Flash randomly crashed independently of the browser used.

3.5.2 Reliability

In 2010, Steve Jobs in an open letter directed to the Adobe platform openly points to some criticisms of the Flash platform. Amongst these criticisms, there is this point *“Third, there’s reliability, security and performance.*

Symantec recently highlighted Flash for having one of the worst security records in 2009. We also know first-hand that Flash is the number one reason Macs crash. We have been working with Adobe to fix these problems, but they have persisted for several years now. We don’t want to reduce the reliability and security of our iPhones, iPods and iPads by adding Flash.

In addition, Flash has not performed well on mobile devices. We have routinely asked Adobe to show us Flash performing well on a mobile device, any mobile device, for a few years now. We have never seen it. Adobe publicly said that Flash would ship on a

smartphone in early 2009, then the second half of 2009, then the first half of 2010, and now they say the second half of 2010. We think it will eventually ship, but we're glad we didn't hold our breath. Who knows how it will perform?" (Jobs, 2010)

3.5.3 Security

Flash Player is one of the most widely distributed piece of software in the world, and as such is target to malicious hackers. No matter how hard Adobe has been actively working on Flash Player's security, hackers have been working on counter effort even harder. "Exploit kits" packets of code that take advantage of these flash player's vulnerabilities in browser to push malware or ransomware have used Flash to futz with countless sites. Furthermore, Flash grants default access to some device resources including the camera and the file system which could be seen as a security hole (krill, 2016)

3.5.4 Performance

Sites that make use of Flash technology have always been slower than the sites that do not use it. (Hachman, 2015) Carried out an experiment to demonstrate the performance drills of Adobe Flash. He tested major browsers (Chrome 44, Windows 10's Edge 12, Firefox 39, Internet Explorer 11, and Opera 31—all the latest versions at press time) abilities to handle Flash. The test ran on a windows 10 machine with sufficient RAM with several websites containing a number of flash videos and flash ads. He measured the browsers CPU consumption with flash enabled then with flash disabled. The results were then the difference in CPU consumptions and the conclusions were as follows. About 56.9% increase in CPU consumption in Microsoft Edge, about 74.6 % CPU consumption in Opera, about 25% CPU consumption in Firefox (with tabs crashing), and about 71% CPU consumption in Chrome.

The above test is just one among several tests carried out by experts on the subject over the internet. According to (Fast Company, 2015), Flash is consuming too much ma-

chine resources, which is a major problem with new generation slim laptops that are lighter, thinner than ever before and relying on power efficiency for long battery life.

3.5.5 Complexity

If you followed the steps on how to embed a flash video, you won't miss the complexity involved in doing so. In the description, I subdivided that in only five steps, and this is because I had repeated the experiment several times in order to get familiar with the process. When comparing this with other steps like uploading Facebook or Youtube videos, it is relatively complex.

The playback controls are another issue. Most other ways of embedding a video provide easy playback controls. With Flash, you need to have converted the video a certain way like using Dreamweaver, otherwise visitors need to be aware that they have to right click on the video to view the playback controls, which doesn't reduce complexity.

3.5.6 Adobe is Proprietary

(Jobs, 2010) in his open letter to the Adobe Platform states:
“Adobe’s Flash products are 100% proprietary. They are only available from Adobe, and Adobe has sole authority as to their future enhancement, pricing, etc. While Adobe’s Flash products are widely available, this does not mean they are open, since they are controlled entirely by Adobe and available only from Adobe. By almost any definition, Flash is a closed system.” (Jobs, 2010) This means that programmers cannot extend the Flash Player. Only Adobe programmers can extend the Adobe products, bring updates and fixes to any issue discovered on Adobe products.

3.5.7 Compatibility Issues

According to (Stellnberger) the constant updates in versions of the Adobe Flash Player creates compatibility issues as videos converted for new versions of Adobe Flash are often not backward compatible, thus videos converted for new versions might not run on prior versions of Adobe Flash.

In this chapter, I have presented a brief discussion about Adobe Flash, this included a description of what Adobe is, what Adobe Flash is, how to embed a video in a webpage using Adobe Flash, followed by some discussions about the advantages and limitations of Adobe Flash. It is to note that because this thesis is basically web oriented, most of the information sources come from the internet, because You don't learn how to swim from a monkey, instead you learn that from a fish.

4. Displaying Videos using HTML5

This chapter will describe how to embed a video using HTML5. The first section of this chapter provides a description of HTML5, what it is, followed by what is required to use HTML5. Afterwards a detailed example of how to embed a video in a webpage using the HTML5 method. Because videos cannot be displayed on PDF documents, a sample program is included in the web application that is attached to this thesis.

4.1 What is HTML5

HTML5 is the 5th version of HTML first published in October 2014 by the W3C (World Wide Web Consortium) to improve the language with support for the latest media (W3.org, 2017). It extends, improves and rationalizes the markup available for documents and introduces APIs (Application Programming Interfaces) for complex web applications. It also includes features designed with low-powered devices in mind thus making it candidate for cross platform mobile applications. HTML5 is supported by most major browsers (Video.js).

New syntactic and tags have been added to handle multimedia and graphic content, for example the `<video></video>` and `<audio></audio>` Tags. Support for SVG (Scalable Vector Graphics) content and MathML for mathematical formulas. Some other useful tags included in HTML5 include

```
<header> </header>,  
<footer> </footer>,  
<aside> </aside>,  
<nav> </nav>  
<figure> </figure>
```

4.2 Using HTML to display videos on Webpage

To display a video in a webpage using HTML5 is very simple. As stated in the introduction of this chapter, HTML5 has enriched the previous HTML tags with tags to support the consumption of multimedia elements. Such tag is the video tag `<video></video>` tag. HTML5 the following formats (MDN web docs, 2017).

- Mp4
- OGG
- WEBM
- MP3
- WAVE

4.3 Displaying Videos in HTML5

The following code snippet illustrates how easy it is to embed a video content into the webpage (w3schools.com).

```
<video id="html5-video-player"
  controls
  autoplay
  loop
  muted
  preload="none"
  poster="images/icon.jpg"
>
```

```
<source src="videos/my-mp4.mp4" type="video/mp4">
<source src="videos/mov_bbb.webm" type="video/webm">
```

This is the fallback text, in case the browser doesn't support HTML5.

eg

Your browser does not support the video tag.

```
</video>
```

4.4 Displaying Animations in HTML5

Animations in HTML5 are created using the combination of JavaScript, CSS and HTML tags (Mozilla Developer Network, 2017), in comparison to Adobe Flash animations which are created in Adobe Animate CC and played using Adobe Flash Player. An animation could also be saved in the form of video and embedded into the webpage as a video (Jennifer, 2017).

Creating animations is not covered by this thesis. Animation APIs as well as several other sources can be found in under:

https://developer.mozilla.org/en-US/docs/Web/API/Web_Animations_API

https://www.w3schools.com/css/css3_animations.asp

4.5 Sample Program

Because moving images cannot be displayed on a PDF, I have submitted a web application as an attachment with this Thesis.

5. Advantages and limitations of displaying Videos using HTML5 over Flash

This chapter presents some advantages of using HTML5 as compared to Adobe Flash to display moving images on webpages, as well as the Limitations. The first section gives some advantages of HTML5, the second section presents limitations of HTML5 as compared to Adobe Flash.

5.1 Advantages of HTML5

It is extremely easy to embed a video using HTML5. As can be seen in the code snippet in chapter 4.3 Above, all that is required is a video of one of the 3 formats MP4, OGG, WEBM (video formats are discussed in section 2.1). Considering that most video capture devices like cameras and mobile phones can be easily configured to capture video in the format mp4, this greatly eases things even more. Once you have your video in the correct format, all you need is to embed it in the source attribute of the video tag.

Built-in Video controls. As seen in the previous 2 chapters, embedding a video using flash often requires that the video controls be programmed manually. HTML5 comes with built-in controls which in addition to facility increases usability and elegance.

One other advantage of embedding videos using HTML5 is that it is an “Open Standard”. Open here means it is not proprietary, thus its functionalities can be extended by any one (Jobs, 2010).

Another advantage is that of performance. HTML5 optimizes performance because it is designed for such purpose. It is designed to be cross-platform compatible and takes into consideration devices performances. Adobe Flash on the other hand was designed mainly for desktop computers using mice (Jobs, 2010), which already constitutes a barrier. With the advent of mobile devices and touchscreens, adobe has constantly modified its platform to support mobile devices and touch screens. Because of this, its performance is limited and cannot be compared with that of HTML5.

Animations can easily be created using the html canvas element, CSS and JavaScript (W3schools.com) . This makes HTML5 extremely useful. When compared to older versions of HTML5 which didn't really provide the possibility to animate HTML tags, HTML5 offers superior animation possibilities. HTML tags themselves can be animated as well as their style sheet attributes.

HTML5 contains JS Web Worker API which allows JavaScript and Browser interface to run in separate threads. The offline caching provided by HTML5 is also a great advantage (Mozilla Developer Network, 2017).

HTML5 provides more compatibility than Adobe Flash. The fact that HTML5 is universally supported across mobile devices (in some cases even exclusively supported) makes HTML5 the best solution for mobile devices (Danny Winokur, 2011). HTML5 runs on practically every web browsers including Firefox, Internet Explorer, Chrome, Safari and Opera. In addition, HTML5 does not require any update. On the other hand, using Adobe Flash, users constantly have to update their versions of the software in order to catch up with new fixes, patches and features.

HTML5 is often used as fallback option when embedding flash videos, so that in case the Flash video fails to play, the HTML5 video is used as spare. This proves how efficient and reliable HTML5.

5.2 Limitations of HTML5

Although it is obvious that new technologies primarily advantageous because they exploit the existing market situation to satisfy every development industries and HTML5 being no exception, it is however to call out that new technologies also have their limitations. In the case of HTML5 some of the limitations to the extent to which it can be used to display moving images are listed below.

The very first limitation is that of security. The fundamental problem with client-side computing is that users have control over the code running on their browser. Most web browsers offer debugging tools, for example pressing F12 on Firefox and Chrome opens the debugging tool in which the visitor has access to the code. He can then insert break point anywhere and edit the code to their will. Even when these Developers Tools are not

built in, they can be easily installed. E.g. [Firebug https://addons.mozilla.org/en-US/firefox/addon/firebug/](https://addons.mozilla.org/en-US/firefox/addon/firebug/) (a JavaScript debugger plugin on Firefox). With this tool, users have access to files, filenames and source code (and even video files that were supposed to be private) that they can use at their will. Although Adobe Flash has security issues, it is not this much easy to access the source code of an animation or video that runs in a flash Player.

Another drawback from HTML5 might be that of format incompatibility. The new audio and video tags makes it easy to embed audios and videos with less than 5 lines of codes into a website. However, the problem of media format still pops up. The embedded videos play well on Chrome, Firefox, Safari but not on Internet Explorer. This is to say that although HTML5 is supported by mostly all the major browsers, because this is computing some particular player may play in almost every web browser but not on a single exceptional player, and the developer might need to use fallback methods or to test each embedded video over every possible browser.

Graphics and Animations are restricted. With adobe flash player, we are able to develop rich animations and graphics. Although it is possible to create animations in HTML5 using JavaScript and CSS, these animations are restricted to certain types. Thus, HTML5 cannot be used to produce the same types of animations that would be produced using Adobe Flash.

HTML5 being a new technology, the recommendations have to be fully accepted and implemented by all web browser developers (thus still not universally supported), which is not always the case. According to (Yakura) versions of Opera browser before Opera 12 still does not have support for HTML5, versions of Internet Explorer prior to IE 8 do not support HTML5. This is part of the reason that Flash is still very popular today.

6. Trends and usage Statistics over the Web

This chapter provides some brief statistics about how the trends is moving from Adobe Flash to HTML5.

The usage of Adobe Flash has been in such a decline in the recent years that almost every major web bodies have decided to discontinue the use of Adobe Flash. Adobe, the company has decided to discontinue any further developments of the Adobe Flash Platform by 2020.

According to (LaForge, 2016) HTML5 provides improved security, reduced power consumption and is faster than Adobe Flash. He cites *“Adobe Flash Player played a pivotal role in the adoption of video, gaming and animation on the web. Today, sites typically use technologies like HTML5, giving you improved security, reduced power consumption and faster page load times. Going forward, Chrome will de-emphasize Flash in favour of HTML5.”* (LaForge, 2016).

Google chrome has officially decided to replaced Adobe Flash with HTML5 by January 2017.

“In December, “Four months ago we announced that we’d be moving to HTML5 By Default to offer a safer, more power-efficient experience. As a reminder, this change disables Adobe Flash Player unless there’s a user indication that they want Flash content on specific sites, and eventually all websites will require the user’s permission to run Flash. To ensure a smooth transition, not all users and sites will be affected immediately. HTML5 by Default and the associated user prompts will be introduced gradually as follows.” (Chromium Blog, 2016) .

According to (Richer, 2016) The percentage of websites using Adobe Flash has reduced from about 50% in 2010 to less than 10% in 2016, and this percentage is to continue sinking.

An online organisation (Occupy Flash) is actively fighting a campaign to kill Adobe Flash with the goal to get the world to uninstall the Flash Player plugin from their desktop browsers. This is to show how serious Adobe Flash’s death is near.

By the end of 2020, Microsoft edge will remove the ability to run Adobe Flash in Microsoft Edge and Internet Explorer on all versions of Windows. Users will no longer be able to run any Flash (Microsoft Edge Team, 2017).

And to sum it up, Adobe is planning to end the life of Flash in 2020. *“Adobe is planning to end-of-life Flash. Specifically, we will stop updating and distributing the Flash Player at the end of 2020 and encourage content creators to migrate any existing Flash content to these new open formats”* (Adobe Blog, 2017).

The date 2010 is to permit businesses that have built upon the Adobe Flash technology to gradually migrate to other technologies.

In this chapter, I have provided some statistics on the tendencies on the internet about how Adobe Flash is losing ground to HTML5. More importantly, Adobe has decided to go on its own and by so doing providing a gradual transition, to give time for dependents to adapt and find alternative solutions.

7. Conclusion

As stated in the title page, the goal of this thesis is to present the possibilities of displaying moving images (videos) on webpages, including the advantages and limitations. To do that, it was essential to subdivide that task into logical chapters. Each chapter built from the other by adding chunks of information to the stack.

In the first chapter, I have given some definitions to the basic terms that would re-occur in throughout the thesis. In chapter 2, I talked about video formats, what they are, which are the most common video formats available over the internet and how these formats can be played (which video players are required to play them), followed by the different methods to display a video on a webpage. Chapter 3 discussed Adobe Flash and the Adobe Flash method to embed videos on a webpage, followed by some advantages and limitations of this method. Chapter 4 discussed HTML5, what is HTML5, how to embed videos using HTML5. Chapter 5 discussed the advantages and limitations of HTML5 method over the Adobe Flash method of embedding videos. In chapter 6, I discussed some trends over the internet about the percentage usage of web sites still using Adobe Flash, and how the tendency will grow.

Based on the findings in this thesis, it is undoubted that HTML5 video content is superior to Flash video content in terms battery life in mobile devices and in overall performance (Danny Winokur, 2011). Even though it is unarguable that Adobe animations are better than HTML5 animations. To use a mix of both technologies would be an optimal choice, HTML5 for videos and use Adobe Flash for complex animations, given that both technologies are not mutually exclusive. However, due to increasing complaints and criticisms about the instability, performance and security issues of Flash, Adobe has decided to discontinue its Adobe Flash platform by 2020 (Adobe Blog, 2017), allowing users to gradually switch to HTML5 in a peaceful transition. As stated by (Jobs, 2010) Adobe Flash was created in the era of desktop computers with mice, and not for tablets and mobile phones with touchpads, this era is now behind.

8. List of Figures

Figure 1 : Digital Image	12
Figure 2: Video Frames	13
Figure 3: Frame Clipping	14
Figure 4: Displaying a Video using html embed tag	23
Figure 5: Embed a Youtube Video	24
Figure 6: Embedding a Facebook Video in to the website.....	26
Figure 7: Output of video embedded using Apple's QuickTime.....	28
Figure 8: Embedding a Video using the Windows Media Player Plugin.....	29
Figure 9: Embedding a Video using the HTML5 method.....	31

9. Bibliography

(1&1 Internet, 2016) 1&1 Internet Ink. (2016). *How to embed a video with HTML*. [online] Available at: <https://www.1and1.com/digitalguide/websites/website-creation/how-to-embed-a-video-with-html/> [Accessed 15 Dec. 2017].

(Adobe Blog, 2017) Adobe Blog. (2017). *Flash & The Future of Interactive Content | Adobe Blog*. [online] Available at: <https://theblog.adobe.com/adobe-flash-update/> [Accessed 15 Dec. 2017]

(Adobe.com, 2017) Adobe.com. (2017). *What is Adobe Flash Player | FAQ | Adobe Flash Player*. [online] Available at: <http://www.adobe.com/products/flashplayer/faq.html> [Accessed 15 Dec. 2017].

(Apple Support, 2016) Support.apple.com. (2016). *QuickTime: Embedding QuickTime for web delivery*. [online] Available at: https://support.apple.com/kb/ta26485?locale=en_US [Accessed 15 Dec. 2017].

(Chromium Blog, 2016) Chromium Blog. (2016). *Roll-out plan for HTML5 by Default*. [online] Available at: <https://blog.chromium.org/2016/12/roll-out-plan-for-html5-by-default.html> [Accessed 15 Dec. 2017].

(Danny Winokur, 2011) Danny Winokur, I. (2011). *Flash to Focus on PC Browsing and Mobile Apps; Adobe to More Aggressively Contribute to HTML5*. [online] Latest company news & updates | Adobe Conversations Blog. Available at: <https://blogs.adobe.com/conversations/2011/11/flash-focus.html> [Accessed 15 Dec. 2017].

(Encoding.com, 2016) Encoding.com. (2016). *Different Video Formats | Encoding.com*. [online] Available at: <https://www.encoding.com/formats> [Accessed 15 Dec. 2017].

(Facebook Developers) Facebook for Developers. *Embedded Videos - Social Plugins - Documentation - Facebook for Developers*. [online] Available at: <https://developers.facebook.com/docs/plugins/embedded-video-player> [Accessed 15 Dec. 2017].

(Fast Company, 2015) Fast Company. (2015). *The Agonizingly Slow Decline Of Adobe Flash Player*. [online] Available at: <https://www.fastcompany.com/3049920/the-agonizingly-slow-decline-of-adobe-flash-player> [Accessed 15 Dec. 2017].

(Freemake.com, 2017) Freemake.com. (2017). *Embed Video on Website: 5 Easiest Ways - Freemake*. [online] Available at: <http://www.freemake.com/blog/embed-video/> [Accessed 15 Dec. 2017].

(File, 2016) File, A. (2016). *FLV File Extension - What is an .flv file and how do I open it?*. [online] Fileinfo.com. Available at: <https://fileinfo.com/extension/flv> [Accessed 15 Dec. 2017].

(File, 2016) File, M. (2016). *MP4 File Extension - What is an .mp4 file and how do I open it?*. [online] Fileinfo.com. Available at: <https://fileinfo.com/extension/mp4> [Accessed 15 Dec. 2017].

(FileInfo, 2016) Fileinfo.com. (2016). *Browse File Types*. [online] Available at: <https://fileinfo.com/browse/> [Accessed 15 Dec. 2017].

(Forums.adobe.com, 2015) Forums.adobe.com. (2015). *Flash player randomly crashes computer / Adobe Community*. [online] Available at: <https://forums.adobe.com/thread/643581> [Accessed 15 Dec. 2017].

(Future Releases, 2017) Future Releases. (2017). *Firefox Roadmap for Flash End-of-Life*. [online] Available at: <https://blog.mozilla.org/futurereleases/2017/07/25/firefox-roadmap-flash-end-life/> [Accessed 15 Dec. 2017].

(Gholamreza 2015) Gholamreza A. (2015). *Digital Image Processing*. [online]. Sisu.ut.ee. Available at: <https://sisu.ut.ee/imageprocessing/book/1> [Accessed 23 Oct.2017]

(Hachman, 2015) Hachman, M. (2015). *Tested: How Flash destroys your browser's performance*. [online] PCWorld. Available at: <https://www.pcworld.com/article/2960741/browsers/tested-how-flash-destroys-your-browsers-performance.html> [Accessed 15 Dec. 2017].

(Help.adobe.com) Help.adobe.com. *Adobe Flash Platform * Embedding SWF content in HTML*. [online] Available at: https://help.adobe.com/en_US/as3/dev/WS4B441C24-BAE3-4110-91FD-A4E5EEFB2467.html [Accessed 15 Dec. 2017].

(Helpx.adobe.com, 2017) Helpx.adobe.com. (2017). *Flash Player Help*. [online] Available at: <https://helpx.adobe.com/flash-player.html> [Accessed 15 Dec. 2017].

(Help Center, 2016) Help Center. (2016). *Embedding videos overview*. [online] Available at: <https://help.vimeo.com/hc/en-us/articles/224969968-Embedding-videos-overview> [Accessed 19 Dec. 2017].

(Imagen, 2015) Imagen. (2015). *Top 10 video formats - Imagen*. [online] Available at: <https://imagenevp.com/blog/top-10-video-formats/> [Accessed 15 Dec. 2017].

(Jobs, 2010) Apple.com. (2010). *Thoughts on Flash - Apple*. [online] Available at: <https://www.apple.com/hotnews/thoughts-on-flash/> [Accessed 15 Dec. 2017].

(Jennifer, 2017) Jennifer, K. (2017). *How to Use HTML 5 to Display Video in Modern Browsers*. [online]. Available at: <https://www.thoughtco.com/how-to-use-html-5-to-display-video-in-modern-browsers-3469944> [Accessed 25 Oct. 2017]

(krill, 2016) krill, P. (2016). *Flash vs. HTML5: The last stand*. [online]. InfoWorld. Available at: <https://www.infoworld.com/article/3117746/web-development/flash-vs-html5-the-last-stand.html> [Accessed 14 Dec. 2017]

(LaForge, 2016) LaForge, A. (2016). *Flash and Chrome*. [online]. Google. Available at: <https://blog.google/products/chrome/flash-and-chrome/> [Accessed 8 Dec. 2017]

(Mediacollege.com) Mediacollege.com. *How to Embed Windows Media Files*. [online] Available at: <http://www.mediacollege.com/video/format/windows-media/streaming/embed.html> [Accessed 19 Dec. 2017].

(MDN web docs, 2017) MDN web docs. (2017). *Media formats for HTML audio and video*. [online]. Mozilla Developer Network. Available at: https://developer.mozilla.org/en-US/docs/Web/HTML/Supported_media_formats [Accessed 21 Nov. 2017]

(Movie, 2017) Movie, A. (2017). *MOV File Extension - What is a .mov file and how do I open it?*. [online] Fileinfo.com. Available at: <https://fileinfo.com/extension/mov> [Accessed 2 Jan. 2018].

(Mozilla Developer Network, 2017) Mozilla Developer Network. (2017). *Basic animations*. [online] Available at: https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial/Basic_animations [Accessed 15 Dec. 2017].

(Microsoft Edge Team, 2017) Microsoft Edge Team. (2017). *The End of an Era – Next Steps for Adobe Flash*. [blog]. Microsoft Edge Dev Blog. Available at: <https://blogs.windows.com/msedgedev/2017/07/25/flash-on-windows-timeline/> [Accessed 14 Dec. 2017]

(Mozilla Developer Network, 2017) Mozilla Developer Network. (2017). *Using Web Workers*. [online] Available at: https://developer.mozilla.org/en-US/docs/Web/API/Web_Workers_API [Accessed 15 Dec. 2017].

(Occupy Flash) Occupy Flash. *The movement to rid the world of the Flash Player plugin*. [online]. Available at: <http://occupyflash.org/> [Accessed 19 Nov. 2017]

(Raymond.cc, 2016) Raymond.cc, (2016), *5 Ways to Extract Video Frames and Save to Images*. [online]. Available at: <https://www.raymond.cc/blog/extract-video-frames-to-images-using-vlc-media-player/> [Accessed: 15 Oct. 2017]

(Richer, 2016) Richer, F. (2016). *The Web Is Turning Its Back on Flash*. [online]. Statista Infographics. Available at: <https://www.statista.com/chart/3796/websites-using-flash/> [Accessed 30.11.2017]

(Ronald, P, 2012) Ronald, P. (2012). *Embedding Video*. [online] Available at: http://www.ist.rit.edu/~rpv/local/tutorials/embedding_video/ [Accessed 15 Dec. 2017].

(Sharpened Productions, 2010) Sharpened Productions. (2016). *Avi File Extension* [online]. Fileinfo.com. Available at: <https://fileinfo.com/extension/avi> [Accessed 28 Oct. 2017]

(Stellnberger) Stellnberger, E. *Videos for the Web with HTML 5*. [Online]. Elstel.org Available at: <https://www.elstel.org/html5video/FlashVersusHtml5Video.html.en> [Accessed 10 Oct. 2017]

(Video.js) Video.js. *HTML5 Browser Support by Codec*. [online]. Available at: <http://videojs.com/html5-video-support/> [Accessed 26 Oct. 2017]

(Videomaker.com, 2010) Videomaker.com. (2010). *What is a CODEC?*. [online] Available at: <https://www.videomaker.com/article/f6/14743-what-is-a-codec> [Accessed 15 Dec. 2017].

(W3.org, 2017) W3C. (2017). *HTML 5.1 2nd Edition*. [online]. W3.org. Available at: <https://www.w3.org/TR/html51/> [Accessed 21 Nov. 2017]

(W3schools.com) W3schools.com. *CSS Animations*. [online] Available at: https://www.w3schools.com/css/css3_animations.asp [Accessed 15 Dec. 2017].

(W3Schools.com) W3schools.com. *What is new in HTML5 W3C - What is New in HTML5*. [online]. W3schools.com. Available at: https://www.w3schools.com/html/html5_intro.asp [Accessed 23 Nov. 2017]

(w3schools.com) W3schools.com. *HTML Video*. [online]. W3schools.com. Available at: https://www.w3schools.com/html/html5_video.asp [Accessed 25 Nov. 2017]

(www.tutorialspoint.com) www.tutorialspoint.com. *Digital Image Processing Introduction*. [online] Available at:

https://www.tutorialspoint.com//dip/image_processing_introduction.htm [Accessed 15 Dec. 2017]

(Yakura) Yakura, M. *HTML5 & CSS3 READINESS*. [online]. Html5readiness.com. Available at: <https://html5readiness.com> [Accessed 21 Nov. 2017]

10. List of Abbreviations

API:	Application Programming Interface
ASF:	Advanced System Format
AVI:	Audio Video Interleave
CPU:	Central Processing Unit
CSS:	Cascading Style Sheets
FLA:	Flash
FLV:	Flash Video Format
FPS:	Frame Per Second
GIF:	Graphics Interchange Format
HTML:	Hypertext Mark-up Language
HTML5:	Hypertext Mark-up Language Version 5
MPEG:	Moving Pictures Expert Group
PDF:	Portable Document Format
SVG:	Scalable Vector Graphics
SWF:	Small Web Format
VLC:	Video Lan Client
W3C:	World Wide Web Consortium
WMV:	Windows Media Video

11. Attachments

11.1 The Thesis DVD

As attachment to this thesis is a DVD containing:

- Copies of this thesis as pdf and word formats
- A website to display the various examples stated in the thesis.

To access this website, right click on the file name “index” or “index.html” and choose “open with”, then select your favourite web browser. I recommend Internet Explorer because I tested the website on internet explorer, and it functioned without requiring any additional installation.

