

行動視訊播放與管理系統

葉耀明 鄭皓中

國立臺灣師範大學資訊工程所

ymyeh@csie.ntnu.edu.tw rock3300178@hotmail.com

摘要

本研究主要以 HTML5、JSP、XML、eXist、NFC 等技術實現行動視訊播放與管理系統。利用 OpenCV 來分析影片內容，找出關鍵畫面並做為影片的分段點，自動為整部影片分割成多個片段，系統依照影片類別，要求使用者輸入和此類別相關的片段資料並以 XML 的格式儲存相關資訊，並以原生 XML 資料庫 eXist 儲存 XML 文件，並且利用 XQuery 查詢語言查詢資料庫中影片及片段內容，讓使用者能夠快速地找到想要瀏覽的影片內容，例如運動影片的精彩畫面或是英語教學影片中的特定授課內容、角色對話，增加使用者瀏覽影片的效率。我們使用 HTML5、CSS、JavaScript 設計出影片的瀏覽頁面，在不需額外安裝任何的附加元件情況下就能夠正常的播放影片內容，能夠提供使用者無論是行動裝置或是個人電腦都有一個良好的瀏覽環境。另外本系統透過與 NFC 技術結合，感應不同的場所以及不同功能的 tag，產生出更多樣化的用途。

(關鍵詞: HTML5、XML、eXist、JSP、智慧型手機、NFC)

Abstract

This research uses the technologies including HTML5, JSP, XML, eXist, and NFC to implement a mobile video playback and management system. Our work uses OpenCV to analyze the video content and identify the key frame to index the cutting point of the video so that it can provide the automatic clipping function for the video to be viewed according to the indexed point. Our system can store the uploaded video according to the provided classification and use the XML format as the basic unit of the eXist Database System. XQuery can be used to provide the query function for the users to navigate the required video. We use HTML5, CSS, and Javascript to design the mobile webpages for the video, which can playback video without extra plugin module. In addition, our system also include the NFC tag for the purpose of video searching which can provide an new interface and experiences for users.

Keywords: HTML5, XML, eXist, JSP, smart phone, NFC

1. 簡介

近年來隨著資訊科技快速的發展，使得網路和電腦提供了人們各式各樣方便、快速的功能，使得

現在大多數人們的日常生活已經和電腦、手機等 3C 產品息息相關，電腦在人們的各個領域，例如工作、學習、休閒...等已經成為不可或缺的工具，網路分享資訊很方便，大多數使用者的頻寬也很足夠，線上教學成為了各方面資訊學習非常熱門的方式，在網路上找的到許多的教學影片，例如語言學習、電腦相關學習、甚至是部份補習班也有線上觀看教學影片的服務。由於多媒體技術的蓬勃發展以及網路儲存空間日漸增加，因此線上分享影音資訊已經變得非常容易，網路上就有許多影音分享的平台提供人們進行瀏覽、分享，根據 Alexa 的排名，全球最大的影音分享平台 YouTube 已經成為全球第三大網站，僅次於 Google 和 Facebook[2]，代表瀏覽、分享影音資訊已經成為大多數人們使用網路的一部分了。

最近幾年智慧型手機也是發展非常快速的一項 3C 產品，到這兩年已經是幾乎人手一隻，功能也越來越強大，結合了傳統手機和個人電腦的功能，因此也能夠隨時隨地的連結網際網路從事各種活動，例如瀏覽網頁、玩遊戲、收發郵件...等等，當然也可以連結到影音分享平台進行瀏覽、分享。HTML5 是 HTML(HyperText Markup Language, 超文件標示語言)的最新版本，其中有一項最重要的新增語法<video>，可以讓 HTML5 格式的網頁在不增加任何附加元件的情況下播放影片，這對於目前市面上有不只一種熱門作業系統的智慧型手機來說，HTML5 是非常適合用來做為影音分享平台的網頁撰寫格式。

由於目前影音分享技術的蓬勃發展，任何人在網路上瀏覽影片都變得非常方便，但是這只限於完整影片的搜尋，在影音分享平台或是線上教學的單位，到處可見數十分鐘長的影片，如果只想觀看影片的特定片段，或是從中間某個場景開始瀏覽，就要花上不少時間在尋找正確的時間點上，尤其是教學影片，時間通常都有蠻長的時間，但是學生可能一次不想看完全部，或者是只想觀看某個部份的教學時，會顯得比較不方便，如果可以分析整部影片，找出影片內的各個主題，為影片分段，並且擷取關鍵畫面作為縮圖，最後為片段建立詳細的描述，就可以輕鬆的為整部教學影片分割成多個適當的章節，學生只需要根據關鍵字搜尋，就可以快速的瀏覽教學影片的內容，減少尋找欲觀看內容的時間，大大的增加學生學習以及一般使用者觀看影片的效率。

2. 研究架構與方法

2.1 系統架構

如圖 1，本系統主要由兩個部分組成，第一個是由許多個 JSP 網頁所組合成的整個系統的入口，使用者可以在此進行使用者登入、影片上傳、影片查詢、片段查詢、影片清單以及影片管理的動作。使用者在上傳影片之後，系統會將上傳時輸入的相關資訊集成一個 XML 檔案，此 XML 檔案的內容是利用 Dublin Core 的格式來對影片內容做描述。之後系統會自動將影片依照影片長度切割成數個片段，每個片段都會產生一個 XML 檔案，檔案中會有數個標籤來描述片段的內容，依照影片的分類，會有不同的標籤來描述不同種類的影片，所有的 XML 檔案都會儲存在 XML 原生資料庫:eXist [3] 中，使用者可以利用系統中影片查詢和片段查詢的功能快速的找到想要觀看的影片或片段。查詢完畢後會將結果傳送至 HTML5 的頁面做影片的瀏覽，在瀏覽頁面中，使用者可以對片段資訊做新增、修改、刪除的動作，來增加影片的查詢效率和準確度。

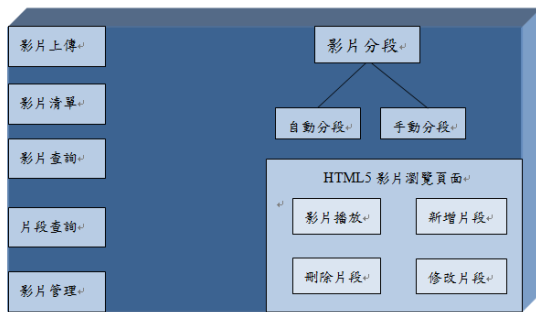


圖 1

■ 影片上傳相關功能

本系統使用 jspSmartUpload API 完成影片上傳之工作。上傳過程中，使用者選擇欲上傳影片路徑、輸入影片相關資訊後，系統會將使用者輸入的影片相關資訊，製作成一份利用 Dublin Core[4]格式描述影片內容的 XML 檔案。所用到的 Dublin Core 資源描述元素有影片上傳者、影片的主題類別、影片的相關描述、影片的上傳日期、資源所屬的類別、影片的格式、影片的唯一識別、影片中使用的語言、影片的來源，表 1 為影片描述檔案範例、表 2 為每個 XML 元素所對應的 Dublin Core 資源描述元素。

影片描述檔案	
<code><?xml version="1.0"?></code>	
<code><rdf:RDF</code>	
<code>xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"</code>	
<code>xmlns:dc="http://purl.org/dc/elements/1.1/"></code>	
<code><rdf:Description</code>	
<code>rdf:about="http://www.w3school.com.cn"></code>	
<code><dc:title>basketball</dc:title></code>	
<code><dc:identifier>b77e5d14-cabb-4641-8ec4-5bda85e72d19</code>	
<code></dc:identifier></code>	
<code><dc:creator>me</dc:creator></code>	
<code><dc:language>en</dc:language></code>	
<code><dc:date>2013/06/13</dc:date></code>	
<code><dc:source>NBA</dc:source></code>	
<code><dc:rights>NBA</dc:rights></code>	
<code><dc:format>mp4</dc:format></code>	
<code><dc:type>video</dc:type></code>	
<code><dc:subject>高畫質專區.體育.籃球</dc:subject></code>	
<code><dc:description>basketball</dc:description></code>	
<code></rdf:Description></code>	
<code></rdf:RDF></code>	

表 1

元素名稱	定義
Title	影片名稱
Creator	影片上傳者
Subject	影片的主題類別
Description	影片的相關描述
Date	影片的上傳日期
Type	資源所屬的類別
Format	影片的格式
Identifier	影片的唯一識別(UUID)
Language	影片中使用的語言
Source	影片的來源
Rights	影片的版權

表 2

■ 影片管理相關功能

因為隨著時間的增加本系統中的影片資源會越來越多，因此利用網路本體語言(OWL)技術為系統資料庫做管理的工作，將每部影片依類別分類，對於影片資源的管理來說更加的方便和整齊。由於每份 XML 影片描述檔，都是獨立無相關性的，本系統為了提供完整且方便的查詢功能，利用 OWL

技術，並將所有影片依照所屬的類別，分類管理，如圖 2 所示。

本系統使用 Protégé API[5] 來建立 RDF/OWL 格式的 XML 檔案。Protégé 為一個免費、公開程式碼的本體語言編輯工具，其提供了一套完整的 OWL API，供開發人員使用。

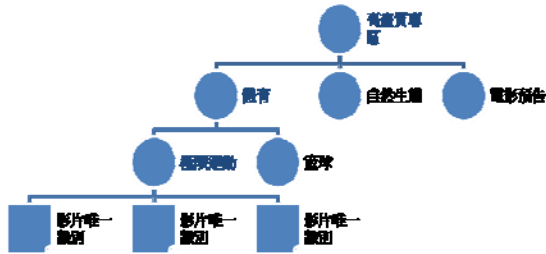


圖 2

■ 影片查詢相關功能

查詢功能為影片資料庫中相當重要的功能，系統依使用者輸入的查詢關鍵字，在資料庫中找出符合的影片資源。本系統中主要提供使用者三種查詢：(1)依影片類別(2)依影片名稱(3)依影片相關描述。

● Query RDF/OWL format file

如圖 4，當使用者選取影片類別後，系統會利用 Protégé OWL API，依照影片類別尋找所有符合的影片。系統將 RDF/OWL 文件載入後，搜尋檔案之樹狀結構。由較大的分類往下搜尋至樹狀結構的最底層，將影片 ID 送至 HTML5 影片瀏覽頁面呈現。



圖 2

● Query eXist Database

如圖 5，當使用者依影片名稱或影片相關描述來做查詢動作時，系統會利用 XQuery[6]查詢語言，在儲存影片相關資訊的 XML 檔案的 eXist 資料庫中尋找該影片，並將查詢結果傳送至 HTML5 影片瀏覽頁面進行瀏覽。

eXist 資料庫支援 XQuery 查詢語言，XQuery 是由 W3C 的 XML Query 工作小組所開發和定義的語言，為專門查詢 XML 檔案的查詢語言。XQuery 語言是一種路徑運算式(Path Expressions)、FLWOR 運算式(類似 SQL 語法)、條件運算式和 XQuery 函數所集合而成的。本研究統合 eXist 提供的 API 與 XQuery 查詢語言，輸入影片關鍵字，即會回傳所有查詢結果，大幅提高資料庫的搜尋效率。



圖 3

■ 影片分段相關功能

當使用者上傳影片之後，系統會利用自動分段的功能將影片依照長度切割成數個片段，長度越長，就會切割越多片段，切割完畢之後，使用者需要前往瀏覽系統中利用片段修改的功能，將切割出來的片段依照影片類別的不同編輯一些相關的資訊，讓其他使用者能夠更快速的找到這些片段，假如自動分段的結果使用者並不滿意，也可以在瀏覽系統中手動進行片段的新增或刪除。自動分段的功能是利用 OpenCV[7](Open Source Computer Vision) 函式庫對影片進行處理，判斷出場景轉換的時間點，對影片進行切割的動作。

OpenCV 是一個由 Intel 公司所開發出來的 C/C++ 語言的 Open Source 圖形演算法的函式庫。包含了 300 多個 C 語言函數的跨平台的中高層 API，可以製做圖片、視訊、矩陣運算、統計、圖論... 等等的 C/C++ 語言程式設計，不依賴其他外部的函式庫，其中大部份的函數用來解決以下領域中的問題，例如人臉辨識、動作分析、物體追蹤、傅立葉轉換和圖像分割等，主要應用在靜態圖片(BMP、JPG、TIF、PNG)和動態 Webcam 的影像處理。

● 影片自動分段

影片上傳之後，系統會自動找出影片中場景轉換的地方，並且設為一個分段點，產生一個片段。優點在於不需要自己花時間為影片分段，系統會自動完成。缺點就是準確度可能會比不上使用者手動的分段，所以使用者有可能要做一些細微的修改，並且片段產生後，XML 檔案中的相關資訊還是需要使用者自己來完成，自動分段流程如下所示：

1. 將影片切割成許多畫面
2. 擷取相鄰畫面
3. 將畫面灰階化
4. 比對相同位置像素
5. 統計差異度大的像素個數
6. 依照影片長度選出數個分段點，切割成片段

● 影片手動分段

使用者在 HTML5 影片瀏覽頁面中可以為影片新增片段，只要輸入片段的起始和結束時間，以及依照影片類別的不同，輸入片段相關的描述和資訊，系統就會產生片段和存有相關描述和資訊的 XML 檔案。優點在於可以很清楚的找到影片中各個主題或者是最精彩的部分，缺點在於使用者必須完整的瀏覽過影片一次，才能切割出所有的片段，

如果影片時間比較長，所花費的時間或精力就會比較大。

2.2 HTML5 影片瀏覽系統

在一個影音分享平台中，最重要的就是在於提供使用者一個良好的影片瀏覽環境。本系統使用 HTML5 網頁技術來設計整個影片瀏覽系統，近年來智慧型手機和平板電腦這些行動裝置的普及率越來越高，功能也越來越強，扣除掉一些較吃硬體效能的功能(例如玩畫面較精緻的遊戲)，這些行動裝置已經可以做到大部分一般人使用電腦時所做的事情。因為目前主流行動裝置上預設的瀏覽器都能夠支援 HTML5，並且 iOS 和 Android 這兩大行動裝置作業系統也宣布未來不會再支援 Flash，因此本研究選擇使用 HTML5 配合 JavaScript 以及 CSS 來設計整個影片瀏覽系統。

如圖 6 所示，影片瀏覽頁面主要分成數個功能及區塊：影片瀏覽視窗、搜尋結果區塊、影片資訊區塊以及新增至清單、新增片段和片段播放三個功能。系統使用流程如下：

1. 在 JSP 頁面搜尋到的結果傳到影片瀏覽頁面後，使用者可以在搜尋結果區塊點擊想要觀賞的影片截圖，點選之後，上方的影片瀏覽視窗就會將影片載入，使用者就可以點擊播放鍵播放影片，另外下方的影片資訊區塊也會顯示出該影片的相關資訊。
2. 使用者如果想要將影片記錄下來供日後瀏覽，點擊新增至清單的按鈕，系統就會將影片加入到目前所登入的帳號下的影片清單，之後只要在 JSP 頁面使用影片清單的功能，就會看到所記錄的全部影片。
3. 使用者可以在此頁面進行片段的新增，只要點擊新增片段的按鈕，頁面會跳出多個輸入欄位，依照影片類別的不同，每個輸入欄位要輸入的資訊也會不同，使用者輸入完畢之後，系統會將所輸入的片段資訊存成 XML 檔案，並存入 eXist 資料庫中。
4. 如果想要瀏覽或修改目前播放影片的所有片段，可以透過點擊片段播放的按鈕，頁面便會跳至片段播放頁面，如圖 7 所示，即可進行片段的瀏覽或修改。



圖 6

片段瀏覽頁面格式和影片瀏覽頁面非常相似，主要分成數個功能及區塊：片段瀏覽視窗、搜尋結果區塊、片段資訊區塊以及播放片段、修改片段、刪除片段和回主畫面四個功能。系統使用流程如下：

1. 在 JSP 頁面進行關鍵字的搜尋或在影片瀏覽頁面使用片段播放功能，頁面都會跳至片段瀏覽頁面，使用者可以在搜尋結果區塊點擊想要觀賞的片段截圖，點選之後，上方的片段瀏覽視窗就會將影片載入，使用者只要點擊播放片段按鈕就可以進行片段的瀏覽，另外下方的影片資訊區塊也會顯示出該片段的相關資訊。
2. 使用者如果認為片段的相關資訊和片段內容有出入需要修改，只要點擊修改片段的按鈕，頁面會跳出和新增片段時相同的數個輸入欄位，使用者只需要輸入正確的片段資訊並點擊送出按鈕，系統便會將修改過的片段相關資訊，傳至 JSP 修改頁面，然後在 eXist 資料庫中找到此片段相關資訊之 XML 檔案，並且進行修改。
3. 使用者可以在此頁面進行片段的刪除，只要點擊刪除片段的按鈕，頁面會將片段名稱傳至 JSP 刪除頁面，然後在 eXist 資料庫中找到此片段相關資訊之 XML 檔案，並且進行刪除，如此就可以將使用者認為比較多餘或者不重要的片段刪除。
4. 下方有一個回主畫面的按鈕，使用者點擊之後頁面便會跳至目前播放片段的影片瀏覽頁面，進行完整影片的瀏覽。



圖 7

3. 系統實作

本研究結合 XML、RDF/OWL、eXist、HTML5、NFC 等多項技術實做行動視訊播放與管理系統，並利用。本章針對系統內部規劃做詳細說明，首先以「新舊系統比較圖」[1]，如圖 8，呈現本研究新增的功能以及相關說明。

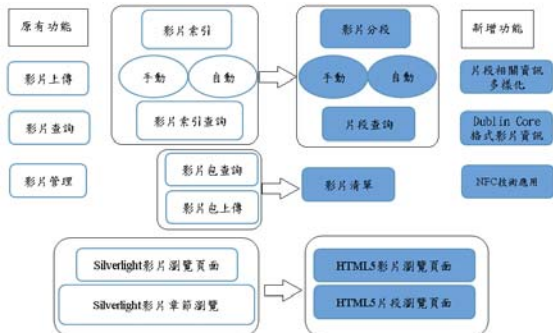


圖 8

■ 片段資訊多樣化

使用者在對影片新增片段時，無論是手動或是自動新增片段，系統都會要求使用者輸入片段的相關資訊，此時依照原影片的類別，會有不同種類的相關資訊來讓使用者輸入，藉此來讓這些片段的相關資訊能夠更清楚的對片段進行描述。

■ Dublin Core 格式影片資訊

每部上傳至影片資料庫的影片都會有一個 XML 檔案來儲存影片相關資訊，影片資源的描述格式是利用 Dublin Core 資源描述格式來對影片進行描述，Dublin Core 總共有 15 種元素可以對資源進行描述，在本研究中會用到其中 11 種，如表 2 所示，透過這 11 種元素就可以對使用者所上傳的影片進行非常詳細的描述。

■ NFC 技術應用

本系統利用與 NFC 的結合，展現出更多樣化的用途，例如在不同地方感應該處的 tag，例如體育館、運動場的 tag，會呈現所有和運動相關的影片，

或者是在英語教學的教室、場所的 tag，則是會呈現所有英語教學的影片，可以讓使用者瀏覽和所在位置相關的影片。另外透過感應不同的 tag，可以針對不同的 Dublin Core 元素進行影片查詢，讓使用者可以有更多的搜尋方式可以使用。最後則是在像博物館、展覽館之類的場所，感應在各個展覽物品、區域的 tag，就能夠播放該物品、區域的介紹、導覽影片，對於使用者在參觀上有一定的幫助。

4. 系統使用分析

圖 9 為本系統的使用案例圖，系統總共有 7 大功能，分別為：影片上傳、新增片段、影片查詢、片段查詢、影片清單、影片瀏覽、影片管理。其中使用者可以使用的功能為前六項，而系統管理者除了前六項外，還可以使用影片管理的功能，各功能的說明如下：

1. 影片瀏覽:使用者可在此功能下觀看查詢結果的所有影片內容。
2. 影片上傳:使用者可以將想要分享的影片上傳至影片資料庫中，提供其他使用者或自己觀賞，也可以對自己所上傳的影片作新增片段等等的處理。
3. 新增片段:使用者可以在瀏覽頁面中手動新增影片片段，依照影片類別的不同，輸入不同的片段資訊，接著將這些資訊利用 JDOM 的 API 建立一個 XML 檔案，並上傳至資料庫。
4. 影片查詢:使用者可選擇依照影片類別、影片名稱、影片描述進行查詢，透過 XQuery 語法查詢資料庫中的 XML 檔案，將符合條件的影片結果列於 HTML5 瀏覽頁面中。
5. 片段查詢:使用者輸入關鍵字在資料庫中依照片段描述搜尋所有影片片段，接著將所有結果列於 HTML5 瀏覽頁面中。
6. 影片清單:使用者可藉由此功能建立自己帳號中的影片清單，可以將想要收藏的影片加入清單中，之後便可以在系統首頁中點選影片清單的功能，在 HTML5 瀏覽頁面中便會列出所有此帳號收藏的影片，讓使用者點選播放。
7. 影片管理:管理者才可使用的系統功能，可針對影片類別做新增或是刪除的動作。

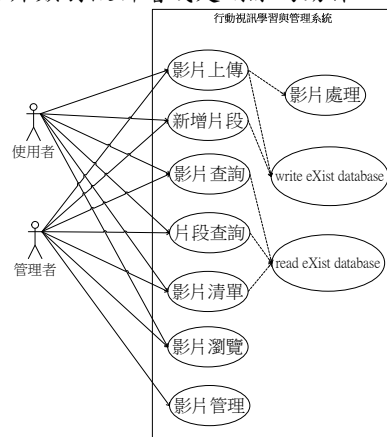


圖 9

5. 結論與未來發展

5.1 結論

在現今的網路環境中有著非常豐富的資源，人們在網路上可以找到幾乎是你想找到的任何東西，近年來由於影音分享平台的蓬勃發展，在網路上能夠找到的影片資源也是越來越多。本系統提供一個良好的影音分享平台，透過功能豐富的使用者介面，使用者可以上傳影片與他人分享，而使用者也可透過搜尋的功能來瀏覽資料庫中的影片資源。但是由於影片長度較長以及影片資源過於豐富等等原因，使用者可能會不容易找到想要瀏覽的影片內容，因此本系統提供了自動分段和手動分段的功能，將影片切割，並且依照影片類別輸入不同的片段相關描述，提供使用者進行搜尋，讓使用者能夠更輕鬆的找到想要瀏覽的內容，例如影片中的精彩畫面和重點場景或是教學影片中的特定主題。並且搭配影片字幕、影片清單的功能，影片字幕可以為影片增加文字，例如英語影片的翻譯、運動相關影片的解說、教學影片的輔助等等，影片清單的功能則是將想要收藏的影片集合起來，類似書籤的功能，以便於日後想要重新瀏覽該影片。綜合以上各項功能，利用本系統可以提升影片的瀏覽效率以及學生瀏覽教學影片時的學習效率。並且結合 NFC 的功能，可以在特定的場所中讀取 NFC tag，獲得相關的影片資源，為本系統展現出更多樣化的用途。

5.2 未來發展

■ 增加與 NFC 技術結合之應用

目前本系統中與 NFC 技術結合的功能大部分是屬於搜尋方面的功能，但是還有許多 NFC 技術的特色沒有利用在本系統中，未來可新增更多除了搜尋相關功能的應用，能夠為本系統在使用上能夠更多用途、多樣化。

■ 直播頻道

HLS (HTTP Live Streaming) 是一個由 Apple 公司提出的基於 HTTP 的直播技術，HTML5 的多媒體相關元素能夠利用此技術進行影片直播的功能，目前大部分是應用在 Apple 公司的相關產品及系統中，未來預計會成為 HTML5 多媒體相關應用中線上直播的主要技術。

6. 參考文獻

- [1]. 栢慶瑞 語言學習之影音索引技術研究. 國立臺灣師範大學資訊工程研究所碩士論文.(2011)
- [2]. Alexa Top 500 Global Sites
<http://www.alexa.com/topsites>
- [3]. eXist-db Open Source Native XML Database.
<http://exist.sourceforge.net/>
- [4]. DCMI Home: Dublin Core® Metadata Initiative (DCMI)
<http://dublincore.org/>
- [5]. The Protégé Ontology Editor and Knowledge Acquisition System

<http://protege.stanford.edu/>

[6]. W3C Recommendation (2007, Jan). XQuery 1.0: An XML Query Language. Retrieved from

<http://www.w3.org/TR/xquery/>

[7]. OpenCV | OpenCV <http://opencv.org/>