# 相關資訊：

1. [樹莓派架設 RTMP 串流（Streaming）伺服器，傳送即時攝影機影像](https://blog.gtwang.org/iot/raspberry-pi-nginx-rtmp-server-live-streaming/)
2. [使用opencv建立即時串流server](http://www.takobear.tw/2014/01/12/bear-diy-webcam/)
3. [Android 建立即時串流Socket範例](http://xxs4129.pixnet.net/blog/post/164402593-android-socket%E7%AF%84%E4%BE%8B)
4. [架設串流服務應用讓你的webcam-升級成ipcam](http://me1237guy.pixnet.net/blog/post/63678997-%E6%9E%B6%E8%A8%AD%E4%B8%B2%E6%B5%81%E6%9C%8D%E5%8B%99%E6%87%89%E7%94%A8%3A-%E8%AE%93%E4%BD%A0%E7%9A%84webcam-%E5%8D%87%E7%B4%9A%E6%88%90ipcam)
5. [VLC : Webcam 擷取變串流](http://gsyan888.blogspot.tw/2013/04/vlc-webcam.html)
6. [專案GitHub網址](https://github.com/andy6804tw/webCam)

# 整體架構：

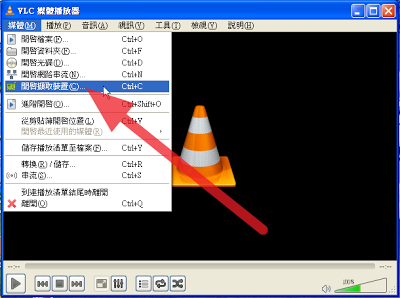
無人機影像串流以P2P方式提供一個ip stream供手機即時存取，手機部分使用webview即時顯示網頁的ip串流。

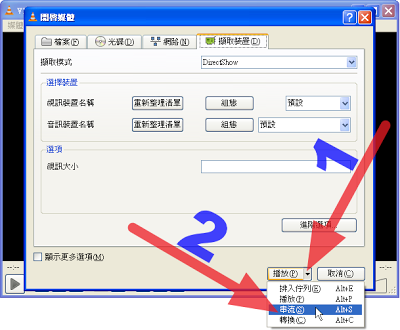
本實作利用電腦ip webcam方式提供一個local host網址供行動裝置端存取

# 實作

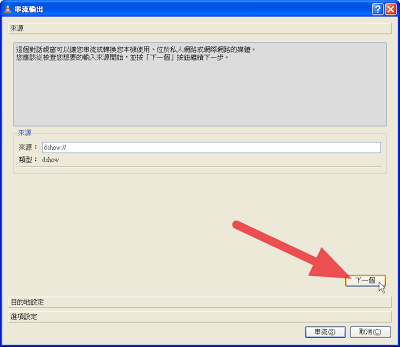
1. webcam建立ip影像串流

**Step 1 :**  
  
當我們把 Webcam 接好，並點選上方選單中的 [媒體] → [擷取裝罝] 就可以出現目前可以擷取畫面裝置的頁面：

[](http://1.bp.blogspot.com/-TEFpOEzpkiY/UXS33RxzCzI/AAAAAAAAJKI/cwKZ50U-ABA/s1600/vlc-webcam-stream-1.png)

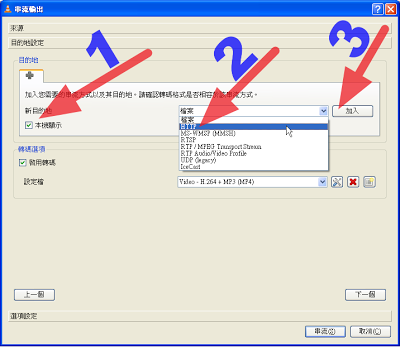
[](http://4.bp.blogspot.com/-GNSmAJPLNfo/UXS329fIjlI/AAAAAAAAJKA/MT3c9ykw1Sg/s1600/vlc-webcam-stream-2.png)

其中的「擷取模式」中，包括了「DirectShow」、「數位電視」和「桌面」，這三種都可以再選取最下面的「排入佇列」、「播放」、「串流」和﹞「轉換」來執行。Webcam 選用的「擷取模式」是「DirectShow」。選完還可以指定「視訊裝置名稱」及「音訊裝置名稱」，這裡我們先忽略不選取，都用「預設」即可。  
  
**Step 2 :**  
  
接著選取執行的動作，我們先點選圖中 1 的小箭頭展開選項，然後再點選「串流」，準備進行串流服務的相關設定。  
  
看看畫面，「來源」已自動填入「dshow://」

[](http://3.bp.blogspot.com/-PVVi0Fl61sw/UXS325lgF3I/AAAAAAAAJJ8/stnRGL-u7xo/s1600/vlc-webcam-stream-3.png)

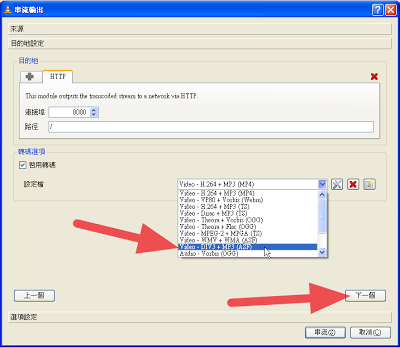
按「下一個」的按鈕，準備設定「目的地」。

**Step 3 :**

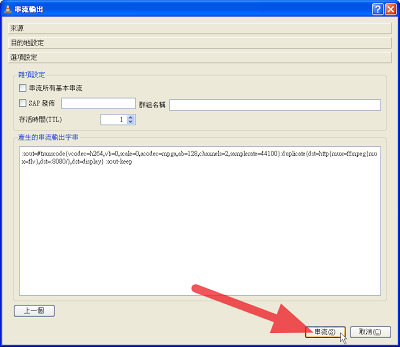
[](http://2.bp.blogspot.com/-TqEiK7bSjss/UXS33xovNrI/AAAAAAAAJKU/BODZBrYPdKg/s1600/vlc-webcam-stream-4.png)

這邊可以讓我們選取要將影音輸出成哪一種串流服務，我們先試試用 HTTP 的方式：

1. 勾選「本機顯示」，讓我們的 VLC 可以監看輸出的結果。
2. 點選「HTTP」。
3. 按「加入」鈕，把 HTTP 服務加入。

[](http://1.bp.blogspot.com/-CRhr66ItnG0/UXS34CYdIeI/AAAAAAAAJKg/SaSVId9X_L4/s1600/vlc-webcam-stream-5.png)

如果加入成功，會出現「連接埠」和「路徑」兩選項，保持預設值即可。  
  
  
**Step 4 :**  
  
「轉碼選項」的部份，如果遠端也是用 VLC 來觀看，使用預設的「Video - H.264 + MP3 (MP4)」的設定檔即可。如果遠端是用 MS Media Player 來觀看，則可以改用「Video - Div3 + MP3 (ASF)」的設定檔。如果是想使用網頁瀏覽器來觀看，則必須使用「MJPEG」才行(這部份請參考最後面，直接貼語法比較快)。  
  
選好轉碼的「設定檔」後，就可以按「下一個」來繼續最後的步驟。  
  
  
**Step 5 :**

[](http://1.bp.blogspot.com/-yBMvuJnXt-0/UXS34SnTDcI/AAAAAAAAJKc/2zoOmHyAFuk/s1600/vlc-webcam-stream-6.png)

最後，VLC 會把依據我們前面設定所產生的語法，顯示在「選項設定」的頁面上，已經設定好了，所以直接按「串流」的按鈕，正常的話，我們可以在自己的 VLC 中看到 Webcam 的畫面，而串流服務也開啟，可以在另一臺電腦開啟播放軟體來看串流了。

**連線的網路是什麼？**  
  
網址我們可以先打開 Windows 附屬應用程式的「命令提示字元」，然後以鍵盤輸入底下的指令後按 Enter 鍵來查詢：

*ipconfig*

將查到的 IP 和我們在 Step 3 中的埠號結合起來，假設電腦的 IP 為 192.168.1.100，而 Step 4 中用的是預設的埠號 8080 ，網址就是：

*http://192.168.1.100:8080*

只要將這串網址在串流播放器中輸入並開啟，應該就可以看到 Webcam 所拍攝的影像了。  
  
  
**如何使用瀏灠器就可以觀看影像**  
  
如果我們採用的是 M-JPEG 格式的串流，這樣，我們就可以跨平臺的在像 Chrome、Firefox、Safari ...... 等瀏覽器中觀看 Webcam 的串流。  
  
最快的方法是直接貼語法，先照前的方法選取「DirectShow」的「擷取模式」後按「串流」，然後直接點選最下方的「選項設定」，跳前我們前面提過的 Step 5，將語法貼入「產生的串流輸出字串」中：

*:sout=#transcode{****vcodec=MJPG****,vb=800,fps=15,scale=1,width=800,height=600,acodec=none}:duplicate{dst=http{mux=mpjpeg,dst=:8080/,****access="http{mime=multipart/x-mixed-replace;boundary=--7b3cc56e5f51db803f790dad720ed50a}"****},dst=display} :sout-keep*

裡面的 ：

* width=800, height=600 可以換成自己想要的畫面大小。
* fps=15 可以換成自己想要的每秒影格數(FPS)。
* vb=800 是 Video Bit Rate ，愈大畫質愈好。

這裡和我們利用圖形界面來設定的方法來比較，串流輸出字串裡，最大的不同是：

* vcodec=MJPG
* access="http{mime=multipart/x-mixed-replace;boundary=--7b3cc56e5f51db803f790dad720ed50a}"

這兩部份。尤其 access 指定了資料在編碼時的格式，如果少了這個，當我們用瀏覽器開網址時，只會出現「下載」，而看不到畫面。

設定好了，就直接按 Step 5 畫面中的「串流」按鈕。

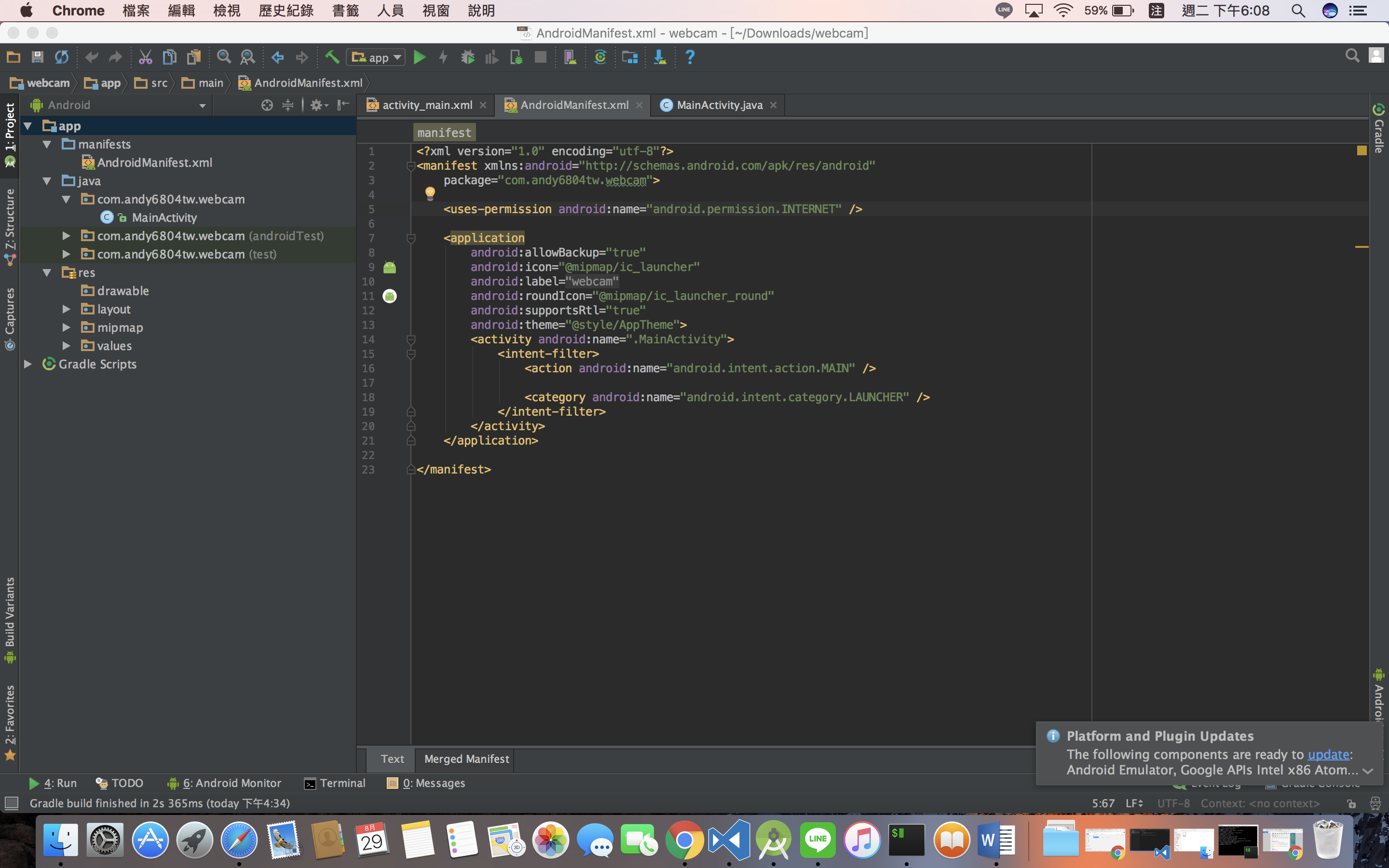
接著就可以打開瀏覽器，並在網址列中輸入類似 http://192.168.1.100:8080 這樣的網址來觀看了。

1. 手機呈現

這裡利用Android來實作，首先建立新專案在layout加入webview的元件，webview的功用是能夠在Android上快速執行http的網頁資訊，相對的影像串流也是可行的但必須在建立於同一個網域下才能正常執行。

**Step 1 :**

**在manifest.xml中加入網際網路的權限**

<uses-permission android:name="android.permission.INTERNET" />

**Step 2 :**

**在main\_activity.xml加入webview元件**

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.andy6804tw.webcam.MainActivity">  
 <WebView  
 android:id="@+id/webView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginTop="45dp"  
 tools:layout\_editor\_absoluteX="8dp"  
 tools:layout\_editor\_absoluteY="8dp" />  
  
</android.support.constraint.ConstraintLayout>

**Step 3 :**

**在MainActivity.java中撰寫程式提供串流來源即時顯示影像**

public class MainActivity extends AppCompatActivity {  
  
 private WebView mWebView = null;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 mWebView = (WebView)findViewById(R.id.*webView*);  
 mWebView.setWebViewClient(mWebViewClient);  
 mWebView.setInitialScale(1);  
 mWebView.getSettings().setLoadWithOverviewMode(true);  
 mWebView.getSettings().setUseWideViewPort(true);  
 mWebView.getSettings().setJavaScriptEnabled(true);  
 mWebView.loadUrl("http://1010code.tk/webCam/");  
 }  
  
 WebViewClient mWebViewClient = new WebViewClient() {  
 @Override  
 public boolean shouldOverrideUrlLoading(WebView view, String url) {  
 view.loadUrl(url);  
 return true;  
 }  
 };  
}

# 實驗成果

