私立東海大學資訊工程學系研究所

# 碩士論文

指導教授：林祝興 博士

Dr. Chu-Hsing Lin

共同指導教授：劉榮春 博士

Dr. Jung-Chun Liu

# 行動裝置多媒體視訊串流播放之電力分析

# Energy Analysis of Multimedia Video Streaming on Mobile Devices

研究生：楊宗哲

(Tsung-Che Yang)

中 華 民 國 九 十 九 年 六 月

# Abstract

Nowadays widespread applications can be found on a variety of mobile devices with embedded systems inside, which operates on batteries with limited power capacity. Among these applications, multimedia replay is most power consuming. In view of this, we systematically analyze encoding methods by measuring power consumption of replay of multimedia. Various usage modes according to user's preference are included in a series of experiments: playing back of videos preloaded on a single mobile device, video streaming via multimedia broadcasting by servers set up in the laboratory or by existing network multimedia servers. By observing effect on power consumption due to common video encoding and playing modes, we are able to decide suitable video encoding methods to produce video streaming that has good picture quality and is energy efficient for playing on the mobile devices.

Based on our experimental results, the difference of power consumption of the worst and the best combinations in this experimental of choices of codecs and file formats is about 30%.

Keywords: codec, energy consumption, mobile device, video streaming.

# 中文摘要

由於各類嵌入行動裝置被廣泛應用，而其中以多媒體的播放與傳輸，最為消耗電能；我們希望透過有系統分析編碼方式，以及一般使用者應用嵌入式裝置的習性去分析，藉由一系列實驗作驗證。實驗情境包括單機播放、自行架設之多媒體伺服器串流播放、現行一般網路多媒體伺服器串流播放等。我們找出利用現今經常使用的編方式，產生的視訊，在嵌入行動裝置撥放時，哪些較為節能，且視覺效果亦達到適切的要求；同時我們也針對影片場景變化快慢模式對電力消耗的影響作觀察,找出適合各種影片場景模式的編碼方式。依據我們實驗結果,採用不同編碼、檔案格式組合的播放方式，在本實驗中最好與最差組合，相比較可節省約有 30%電量消耗。

關鍵字：編碼、電力消耗、行動裝置、串流播放

# 致謝

本篇論文能順利完成，要感謝的人很多，首先要感謝指導老師　林祝興教授及劉榮春教授兩位恩師，在這兩年來細心指導，使我獲益良多。也要感謝口試委員特別撥冗指正論文中錯誤，使論文更加完整，並提供未來可再精進的方向。

在資安實驗試期間也要感謝學長姐懋樺、嘉仁、倫元、建廷、美君、志捷、嘉瀚、亞太影印店的惠娟給我許多建議與幫助；實驗室同學蓉蓉、浩天、煒程、慶儒我們一起在 ST 大樓等待東方太陽緩緩升起；感謝學弟妹羅琪、逸竹、哲維、信雁、本加、信斌；感謝你們給予陪伴與鼓勵；謝謝你們。

研究生期間讓我成長許多，最要感謝的是我家人的精神支持，僅將此論文獻給我最敬愛的阿爸、阿母、阿兄、阿嫂、妹妹，與所有關心我的親友；以及永遠給我最大支持與陪伴的老婆家君、兒子冠霖。今日終於完成碩士學位。

在此，致上我最大的感謝與祝福所有師長、家人、朋友與學弟妹們，平安喜樂! 願目前本實驗能對未來研究者有所裨益。

# Contents

# Figure List

# Table List

# Chapter 1
# Introduction

Mobile devices with embedded systems have wide-ranging and pervasive applications in the modern era, for example, they are frequently used to play back music, pictures, and films for entertainment purpose. We can't help but understand the accumulative amount of energy consumption from intensive and frequent use of mobile devices be significant and not to be neglected. One way to cope with this problem, as most manufacturers do, is to reduce energy consumption spent on hardware computation, the other way, as we investigate in this paper, is to assist the electronic consumers in selecting suitable and energy efficient encoding methods for multimedia streaming.

In recent year, people like to store and share a variety of multimedia data, such as photos, audio, and video in network platforms. Applications through the Internet bloom, and load on it is multiplied in a geometrical way. Multimedia files are shared with and replayed by not only friends and family members, but also for anyone known or unknown in the whole world. Among multimedia, video data is most noteworthy, since it uses vivid forms to express and communicate messages among people. There are variety of replaying systems for video data, such as DVD players, personal computers, and mobile electronics such as the PDA and smart phones. Mobile embedded electronic devices are constrained by batteries, for which recent technology still does not find solutions to offer enough electric capacity for sustainable use. To replay video on the mobile devices, we need be aware of the video code efficiency in regard with power consumption. Therefore, we think in other angles to relax power consumption in the user side. By investigating effect of upload video formats on the power consumption and visual quality, we try to find solutions based on existing technology to alleviate power consumption immediately as well as to maintain acceptable visual quality.

Rapid development of wired and wireless networks and popular use of mobile devices have great influence on user's life style. Video streaming technology has become important since it is indispensible in information broadcasting organization, personal website, and Blogs. According to mesurement of power consumption of video codes of various file type, file format, and adjustible parmaters of codecs, we try to discover which common video codecs are more energy efficient. We also investigate power consumption of replaying of video streaming in a play after download mode.

In order to obtain a realistic experimental data, we plan to investigate power consumption of video replay in a platform set up in the laboratory environment as well as in a platform combined with the real Internet network.

# Chapter 2
# Background

We introduce the relevant video encoders, the file conversion software, and the stream server in our experiment in this chapter.

## 2.1 Video codec standards

The video codec standards used in the experiment are as follows:

WMV

Windows Media Video (WMV) is a generic name for the set of video codec technologies developed by Microsoft. WMV competes with other codec such as RealVideo, DivX, Xvid, and H.264, etc. Generally speaking, common usage of the term WMV refers to this codec only. WMV supports variable bit rate, average bit rate, and constant bit rate encoding. Although WMV is generally packed into an Advanced Systems Format (ASF) container format, it can also be put into AVI or Matroska containers. When encapsulated in ASF file format, WMV can be supported and managed tools with several rights that are use for protecting the intellectual property right [14]. In the experiment, we adopt WMV 7, WMV 8 and WMV 9.

MPEG-4

MPEG-4 is a compression encoding standard for audio and video data. It is released by Moving Picture Experts Group (MPEG) of ISO and IEC. Its first version was released on October 1998, and its second, on December 1999. The major application of MPEG-4 format is network streaming, DVD dispense, Audio transmission (Internet phone), and TV broadcasting.

MPEG-4 includes most functions and merits of MPEG-1, MPEG-2, and other formats. It also adds or expands supports for Virtual Reality Modeling Language (VRML), object oriented combined files (including audio, video and VRML objects), Digital Right Management (DRM), and other interactive features.

Most functions of MPEG-4 are left for users to decide whether or not to use them, which means that all features of it are not necessary included in some files, so, this format has so called profiles and levels to define sets for some features of MPEG-4 specific applications.

Divx

Divx technically belongs to the level of MPEG-4 Part2. DivX codec is very popular since it can compress long videos into smaller pieces and maintain relatively high visual quality. DIVX7 was release in 2009, in which it improves limit of the older formats by adding supports for H.264 video, AAC audio format, and Matroska container.

### Xvid

Xvid (formally called as XviD) is an open source video codec for MPEG-4, written based on OpenDivX. Xvid is developed by a group of volunteer developers for OpenDivX that ended development since July 2001. Xvid supports various encoding modes, including quantization methods, range control, motion detection and curve balance distribution, dynamic I-frame interval, psycho-visual brightness modification, cast select item, externally definition control, motion vector acceleration (Hinted ME) encoding, screen optimization decoding techniques, etc. and becomes a very powerful tool for users.

### H.263

H.263, a video codec, is a low rate video encoding standard made by ITU-T. Initially, H.263 was designed for communication of H.324 based systems (i.e. to realize video conference and telephoning for public switch telephone based networks and other circuit switch based networks). But, later on it is found that H.263 can be successfully applied on H.323 (a videoconferencing protocol on the RTP/IP network), H.320(a videoconferencing protocol on the ISDN network), RTSP(Real Time Streaming Protocol) and SIP (Session Initiation Protocol).

H.264

H.264, or called as the tenth part of MPEG-4, is a highly compressed digital video codec proposed by the Joint Video Team (JVT) of Video Coding Experts Group (VCEG) of ITU-T and Moving Picture Experts Group (MPEG) of ISO/IEC.

FLV

Flash Video (shorted as FLV), a popular container file format used to deliver video over the Internet, has become very widespread along with the prosperity of video websites. The FLV media streaming format is a new video format, and its appearance has effectively taken care of problems such as inefficient use on the network of large volume SWF files, which result from introducing Flash into video files.

In general, FLV files are wrapped in the shell of SWF PLAYER. FLV can protect well the original address from being easily loaded and so it reaches the goal of copyright protection. By now, most video sharing websites, such as YouTube, NICONICO Douga, Google Video, Yahoo! Video, and MySpace, has used this format.

## 2.2 Video conversion software

We use Super© to encode the video files in the experiment. Super© is a free software for file conversion of various multimedia video and audio formats as shown in Figuer 1. Many features of conversion are provided, such as Container (refer to Figuer 2), Video Codec (refer to Figuer 3), Audio Codec, Video Scale Size, Aspect, Frame rate, Bit rate (refer to Figuer 4), Logo Watermark, Turn Right, and Turn left.

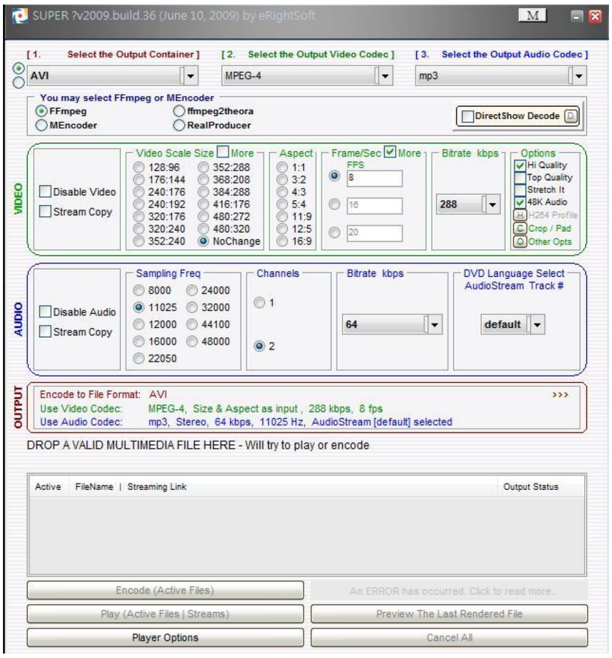


Figure 1. Super soft

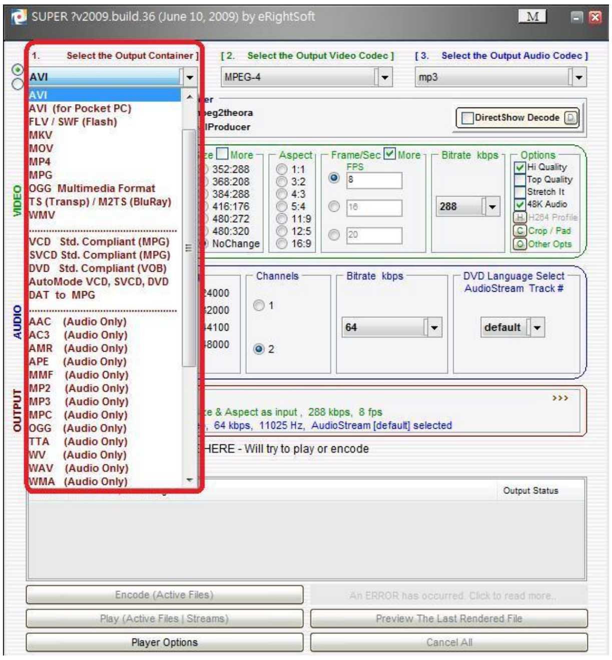


Figure 2. Super container

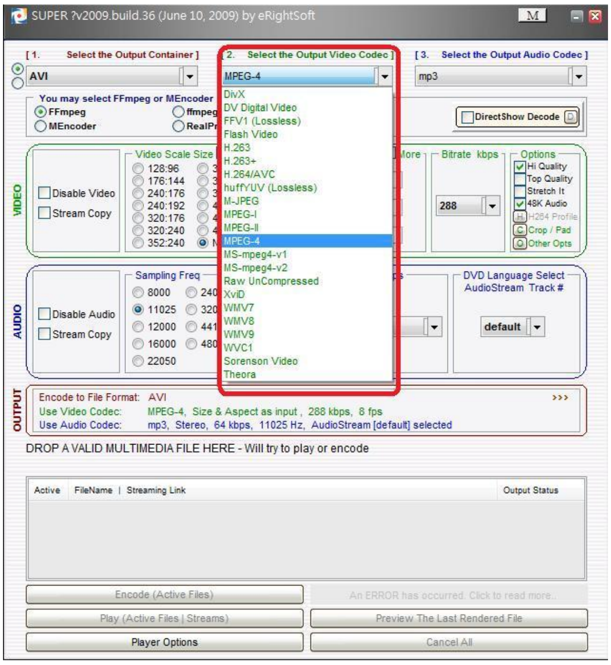


Figure 3. Super Video Codec

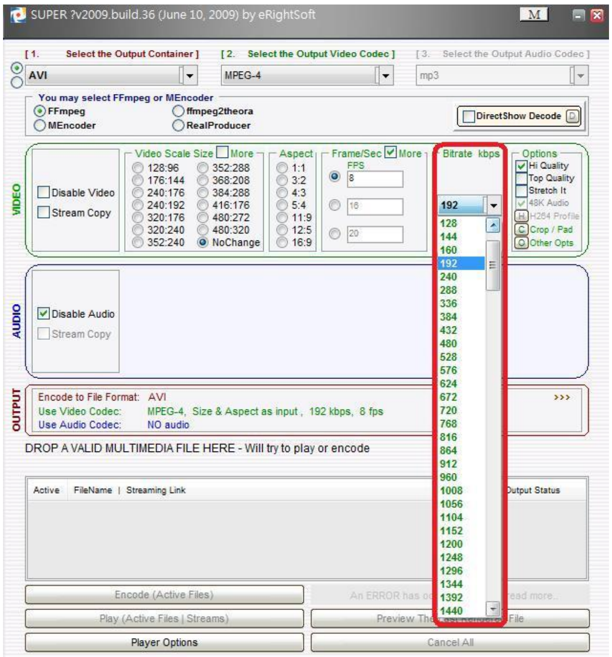

Figure 4. Super Bit rate kbps

## 2.3 Video player

TCPMP

The Core Pocket Media Player (TCPMP) is a free and open multimedia player (refer to Figure 5), which can run on Windows, and Windows CE／Windows Mobile／Palm OS operating systems. It supports many file formats such as ASF, ASX, AVI, DIVX, M2V, M3U, MKV, MP3, MP2, MPEG, MPG, mpeg4, OGG, OGM, WAV, WMV.


Figure 5. TCPMP

## 2.4 Related application software and equations

LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench） is a compiling platform for graphic programming developed by National Instrument Inc. (refer to Figure 6). It first released in 1986 in Apple Computer. LabVIEW was specially designed for automatic control of electronic instruments in the early time, but it has become well-developed high level programming language. It uses graphics for programming. The difference between graphic programming and conventional programming languages is that it uses concept of data flows to separate from traditional thinking ways and so it enables programmers complete writing of program codes as soon as they finishing construction of flow charts. LabVIEW was first to introduce special concept of virtual instrument to let users, through interface of man and machine, directly control instruments developed by the users. Besides, rich function library is provided by LabVIEW, includes: signal capturing, signal analysis, machine vision, numerical computation, logical operation, voice vibration analysis, data storage, etc. Now it supports operation systems of Windows, UNIX, Linux, Mac OS, etc.

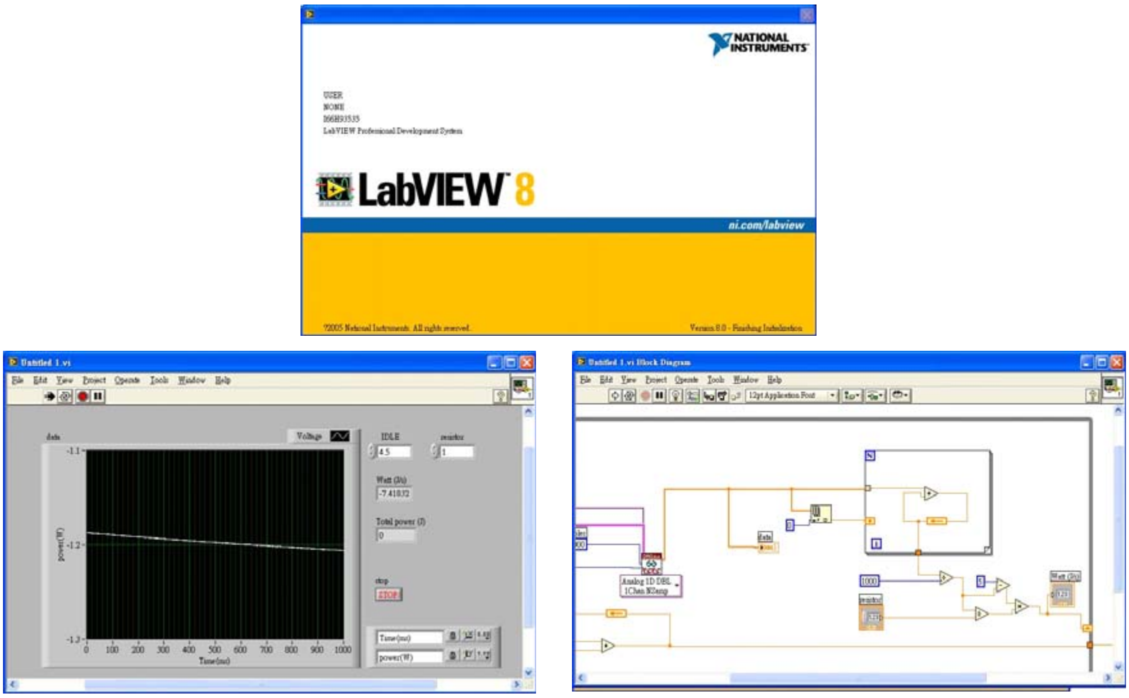


Figure 6. LabVIEW

MMS

Microsoft's streaming server Microsoft Media Services (previously called NetShow Services) uses the Microsoft Media Server (MMS) protocol to transfer unicast data. MMS can be transported via UDP or TCP. The MMS default port is UDP/TCP 1755.

MATLAB

MATLAB is commercial mathematic software developed by MathWorks, a programming language and a numerical computing environment, includes mainly the two major parts: MATLAB and Simulink. MATLAB is based on matrix operation and its full name: MATrix LABoratory is derived from this. It plays the leader role in numeric computation of mathematic and scientific application software. MATLAB can carry out matrix operations, plots of functions and data, implementation of algorithms, construction of user interface, and sequence for connecting with other languages. Its main applications are in the areas of engineering computation, control design, signal processing and communications, image processing, signal detection, financial modeling design and analysis, etc.

PSNR

In general, a PSNR value of 27 or 28 is acceptable for integrity. It works alright for images that do not need high precision or complexity. Any image with a PSNR value more than 30 has high integrity and very good picture quality, no deviation from the original copy is perceivable by the general eyes. When the PSNT value is above 34 or 35, it has hardly any loss of integrity. In usual lossy compression, one needs to trade off some compression rate to have such high quality image. If image has PSNR value of 40 and above can be treat as the same of the original copy. [12]

For visual perception, the PSNR value is not very objective as it suggest, sometimes we need help of human eyes to decide image quality.

## 2.5 Measurement of power consumption

We used a HTC Touch Diamond2 Microsoft® Windows Mobile™ 6.1 Professional (ROM: 512 MB, RAM: 288 MB, 2G SDRAM) and an Acer n300PDA, which has a 400 MHz Samsung S3C2440 processor (with 64 MBROM and 64 MB SDRAM), and its operating system is Microsoft® Windows Mobile™ Version 5.0 [14]. In order to measure the power consumption, we removed the battery from the Acer n300 PDA and placed a 1 Ω resistor in series with a 5 V power supply [17-20]. To follow up the current advances in mobile device technology, we use a smart phone: HTC Touch Diamond2 as well as the PDA: Acer n300 to do experiments.

The processes are video encoding and replaying is shown in Figure 7. We load the original DVD or video files in Super© and encode video files in various formats. Then we use video players such as TCPMP to replay the encoded video files. Since other activities in the same network may affect the video streaming behavior, the experiments are conducted in the time and location that will have smallest interference from other uncontrollable networking factors.

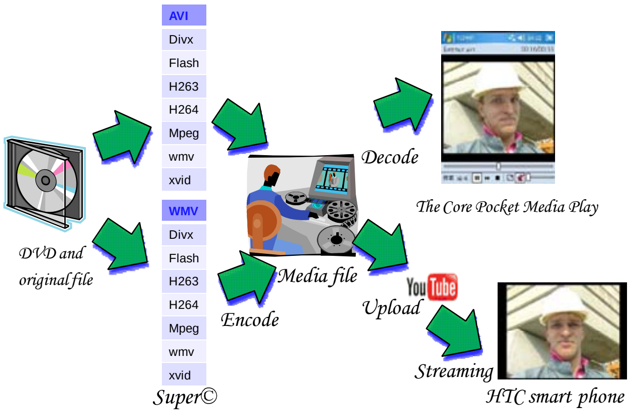


Figure 7. The process of video encoding and replaying

We used a National Instruments PCI DAQ data acquisition board to sample the voltage drop across the resistor to calculate the current at 1000samples/s. The energy measurement was performed in LabVIEW 8, a GUI-based data acquisition, measurement analysis, and presentation software [21].

To find the actual energy consumption due to replaying of video stream on mobile devices, we first record the power consumed by the PDA and smart phone in the idle state. Then we start playing the video and record total power consumption from start to end of video playing. We then calculate the actual power consumption due to replaying of the video by subtracting the idle power consumption from the total power consumption. Finally, the actual energy consumption (in Joules) due to of replaying of the video stream is computed by summing up all the actual power consumption multiplied by the time.

We calculated the instantaneous power consumption corresponding to each sample and the total energy by Eqs. (1) and (2):

$$P_{Inst} = \frac{V_R}{R} \times V_{PDA} \quad \text{.........................................................(1)}$$

$$E = \sum P_{Inst} \times T \quad \text{.........................................................(2)}$$

In general, human eyes are more sensitive to images of low frequency than of high frequency. In fact, human eyes are more sensitive to change of brightness than of colors. In the experiments, we use full color images, consisting of three components of colors: Y, Cb, and Cr. Y, the luma component, takes up about 80% to 95% of energy of the full color image, while the other two components: Cb and Cr, the blue-difference and red-difference chroma components, take up the rest 5% to 20% of energy (This ratio is close to the ratio of volumes of rods to cones). Light sensing organizations of the retina consists of two major classes: first, rods for sensing of the light; second, cones for sensing of the color. Since the number of rods is larger than that of cones, their sensing abilities are stronger. As a result, Human eyes cannot distinguish minuscule change of colors as they do in distinguishing lightness from darkness. In other words, data of Y component is the most important.

To calculate PSNR values, we had tried to use RGB values of pictures for comparison of code compressed video data and original ones. But, since RGB colors do not uniformly distribute for every frame of pictures, the calculated PSNR values deviate a lot. We also had the original picture transform into black and white or grey scale and obtained reasonable PSNR values close to observation by the eyes. But, since significant amount of energy is lost when transforming original color pictures into black and white or grey scale, we do not use this approach, either.

To have the same resolution as the encoded video picture, we reduced the original video picture by optimal quantization method, and then we converted their RGB values into YCbCr values (refer to Figure 8), and used the Y components to calculate PSNR values for comparison.

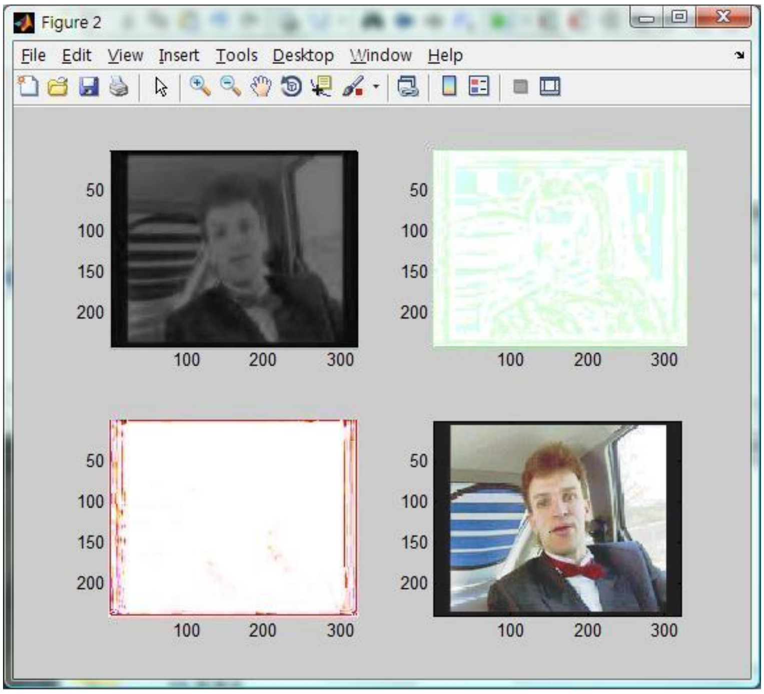


Figure 8. Exploded view of YCbCr

Table 1. Video types used in the experiment

| | |
|---|---|
| Hardly moving or still pictures from scene to scene | The Simpsons |
| Standing still and then slowly moving pictures from scene to scene | Foreman |
| Fast moving pictures among scenes | Carphone |

Video clips of foreman and carphone are common and standard films used for video experiments (Refer to http://trace.eas.asu.edu).

# To calculate PSNR values:

$$PSNR = 20 \cdot \log_{10}\left(\frac{MAX}{\sqrt{MSE}}\right)$$

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j) - K(i,j)\right]^2$$

$$PSNR = 20 \cdot \log_{10}\left(\frac{255}{\sqrt{\frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j) - K(i,j)\right]^2}}\right)$$

I: Y values of the original image

K: Y values of replaying of encoded image

m: length of images

n: width of images

MSE: mean square error

# Chapter 3 Experiment Setting

In this chapter, we give detailed description of the experiment setup for the series of experiments to investigate power consumption for replaying video files encoded by various codecs on mobile devices.

## 3.1 Experiment 1 Setting

We investigate the effect of the bit rate on the power consumption when replaying video files on the HTC smart phone. Also, we compute corresponding PSNR values to assess video quality when replaying the video files. Since previous experimental results shows that the frame rate of video does not affect much of the power consumption, we will not investigate the effect of the frame rate on the video stream in this thesis.

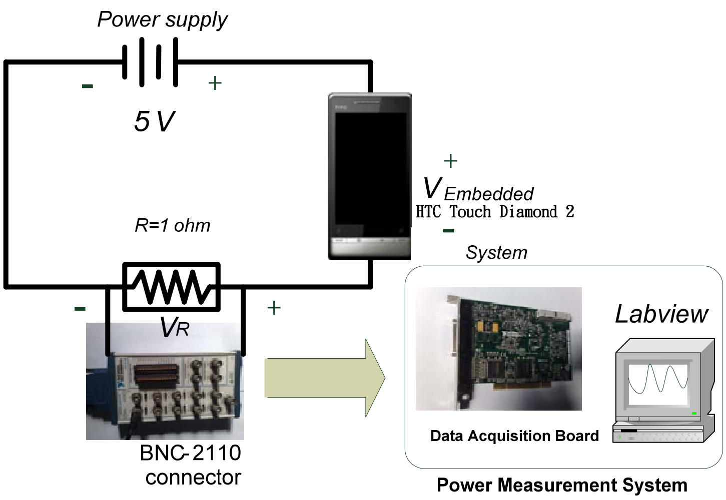

Figure 9. Experimental 1. Setup

## 3.2 Experiment 2 Setting

In this experiment, the video files are replayed separately on the PDA and Smart phone. The video files are stored beforehand in the mobile devices.

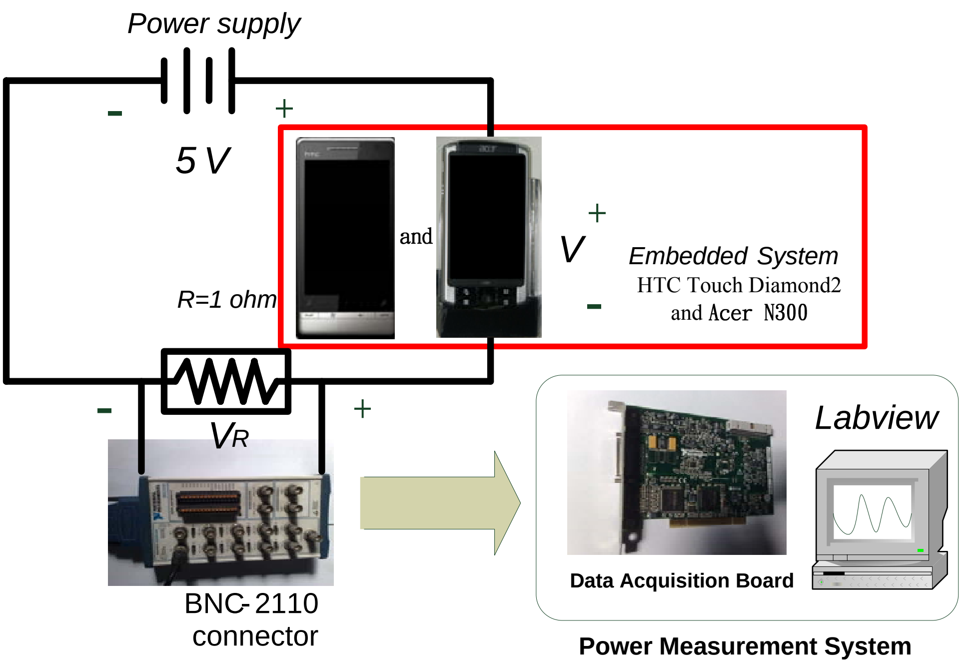


Figure 10. Experimental 2. Setup

## 3.3 Setting for Experiment 3

In this experiment, we construct Microsoft Media Services on the Microsoft's streaming server to broadcast the video streaming, which is wirelessly uploaded to the Smart phone.

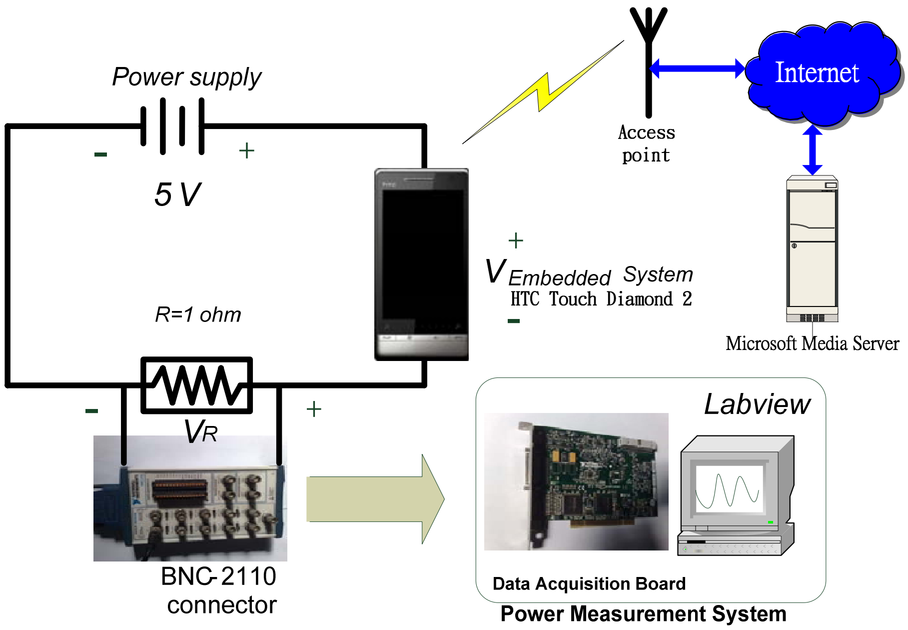


Figure 11. Setting for Experiment 3

## 3.4 Experiment 4 Setting

In this experiment, we upload video files in You Tube, and then we load the video files via the Internet into the Smart phone.

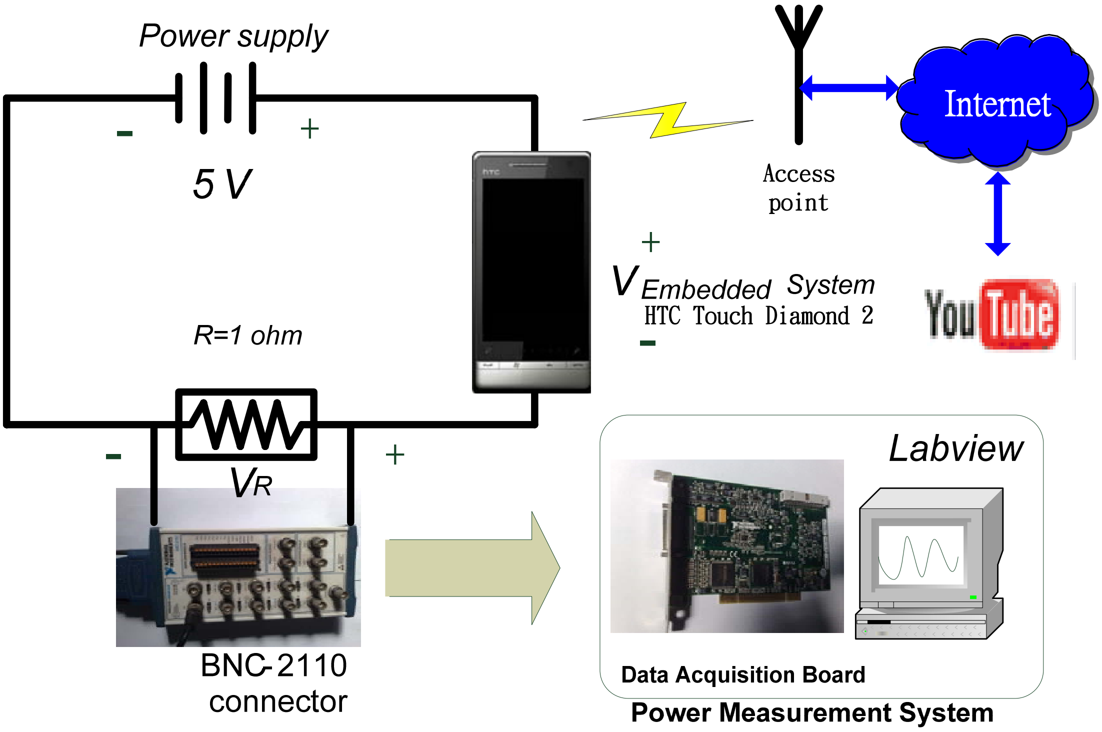


Figure 12. Setup of Experiment 4.

# Chapter 4
# Experiments and Results

In chapter 4, we show plots and tables of experiment results of power consumption and related PSNR. From trend of curves in the plot, we are able to find the optimal combination of video encoding codec and parameters to have energy efficient video codes for replaying on the embedded mobile devices.

Experiment 1:

Table 2. Power consumption (in Joules/sec) and PSNR values of replaying the Simpsons encoded with various bitrates on the smart phone (resolution: 320×240, frame rate: 25 frames/sec, video format: wmv_wmv9)

| *Bitrates( unit: kbps)* | *144* | *288* | *432* | *576* | *720* | *864* | *1008* | *1152* |
|---|---|---|---|---|---|---|---|---|
| *The Simpsons* | 0.0385 | 0.0473 | 0.0484 | 0.0558 | 0.0737 | 0.0764 | 0.0803 | 0.0811 |
| *Growth rate* | null | 22.3% | 2.3% | 15% | 32% | 4% | 5% | 3.8% |
| *PSNR* | 20.3 | 23.32 | 24.43 | 26.31 | 27.2 | 27.88 | 28.51 | 28.9 |

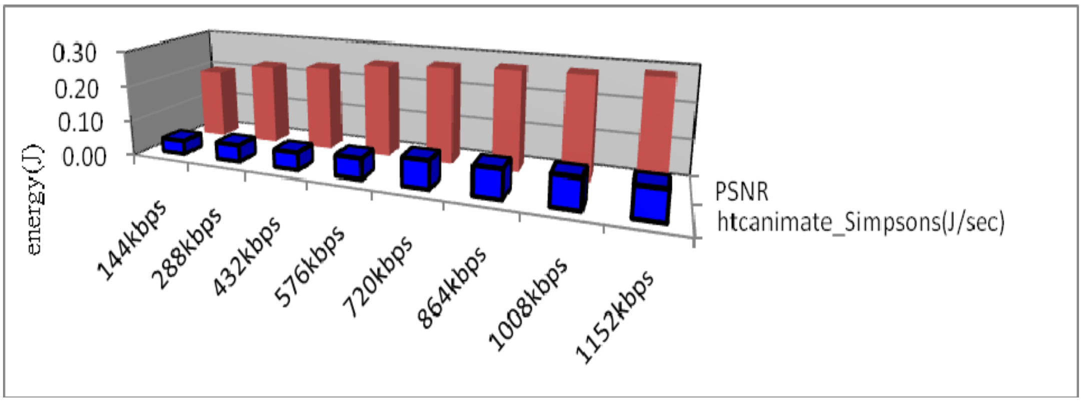


Figure 13. Power consumption (in Joules/sec) and PSNR values of replaying the Simpsons encoded with various bit rates on the smart phone (resolution: 320×240, frame rate: 25 frames/sec, video format: wmv_wmv9)

From Figure 13 and Table 2, we observe that we obtain good picture quality films encoded with bit rates of 576 kbps or more, since the PSNR values of them are large enough (26.31~28.9). From here we can set the PSNR value of 26.3 for the critical value. When seeing by the eyes, nearly no variance of picture quality is perceived for films encoded with bit rates of 576 kbps or more. A similar trend is observed in our previous paper, and since the smart phone has more resources than the PDA, Acer n300, used in the previous paper, we obtain a more complete picture of power consumption due to encoding bit rates of films in embedded devices.

Experiment 2:

Table 3. Power consumption of Acer n300 (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

| *PDA* | *ms_mpeg4 v1* | | *ms_mpeg4v2* | | *wmv7* | | *wmv8* | | *wmv9_M* | | *wmv9_S* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *The Simpsons* | 1.2269 | 27.4 | 1.2328 | 29.3 | 1.2207 | 27 | 1.2396 | 27.8 | 1.2247 | 29.3 | 1.2421 | 28.8 |
| *foreman* | 2.3630 | 27 | 2.4108 | 29.5 | 2.4142 | 26.7 | 2.4206 | 27.6 | 2.4299 | 29.5 | 2.4630 | 27.3 |
| *carphone* | 2.3153 | 26.7 | 2.3530 | 29.1 | 2.4329 | 26.8 | 2.3704 | 27.3 | 2.4647 | 29.6 | 2.4962 | 27 |

*J/S : Joules/Sec

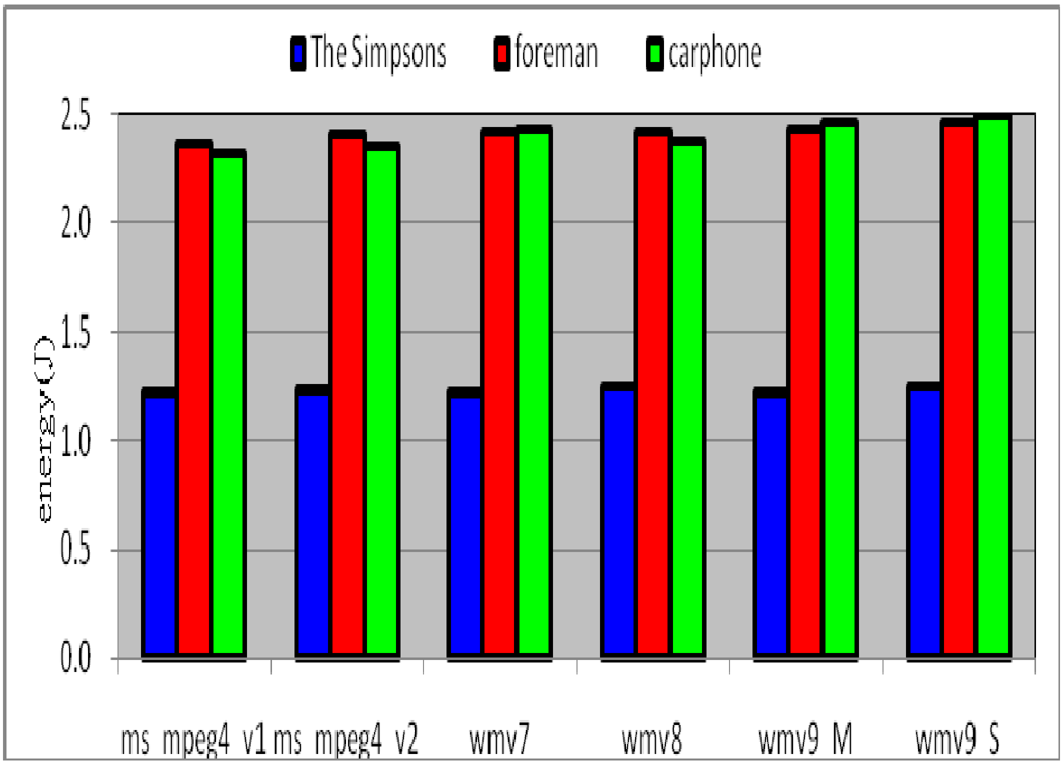


Figure 14. Power consumption of Acer n300 (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

In Figure 14 and Table 3, effects of power consumption and picture quality from the scene change are investigated. Films of three kinds of scene change characteristics are used and they are encoded with various codecs. When replaying the Simpsons, a nearly still scene film, ms_mpeg4_v2 gives the best picture quality but consumes most energy, and wmv9_M gives the best picture quality and consumes less energy than ms_mpeg4_v2. (ms_mpeg4_v1: Video for Windows-based codec, Non-standard MPEG-4 codec incompatible with the later standardized version of MPEG-4 Part 2. ms_mpeg4_v2: VfW-based codec, Non-compliant with finalized MPEG-4 part 2 standard). When replaying the foreman, films of still to slows scene change, ms_mpeg4_v2 gives the best picture quality and does not consume most energy. When replaying the carphone, films of fast scene change, wmv9_M brings the best quality but consumes most energy.

Table 4. Power consumption of HTC (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

| *HTC* | *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *wmv7* | | *wmv8* | | *wmv9_M* | | *wmv9_S* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *The Simpsons* | 0.0825 | 27.3 | 0.0836 | 29.5 | 0.0833 | 28 | 0.0843 | 27.8 | 0.0832 | 29.1 | 0.0846 | 29 |
| *foreman* | 0.2325 | 27.7 | 0.2371 | 29.9 | 0.2374 | 28.5 | 0.2381 | 28.3 | 0.2397 | 29.4 | 0.2410 | 29.2 |
| *carphone* | 0.2048 | 27.1 | 0.2088 | 29.3 | 0.2222 | 28.8 | 0.2174 | 28.7 | 0.2297 | 29.2 | 0.2327 | 29.1 |

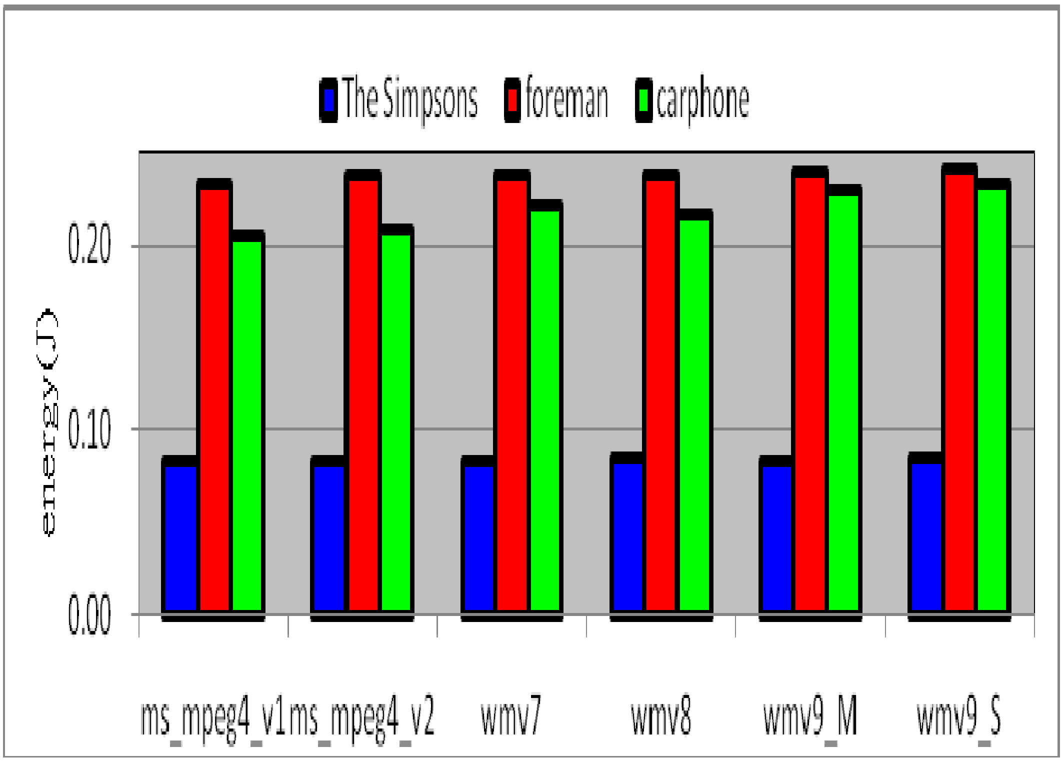


Figure 15. Power consumption of HTC (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

In Figure 15 and Table 4, effects of power consumption and picture quality from the scene change are investigated in HTC, the smart phone. Films of three kinds of scene change characteristics are used and they are encoded with various codecs. When replaying the Simpsons, a nearly still scene film, ms_mpeg4_v2 gives the best picture quality but consumes most energy. When replaying the foreman, films of still to slows scene change, ms_mpeg4_v2 gives the best picture quality and does not consume most energy. The same result is found when replaying the carphone. Because ms_mpeg4_v2 has the dynamic prediction feature, for fast scene changing films, it is able to have best picture quality and be energy efficient.

Table 5. Power consumption of Smart phone and PDA (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI)

| *Smart phone & PDA* | | *divx* | | *h263* | | *h263+* | | *h264* | | *mpeg1* | | *mpeg4* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *The Simpsons* | *Smart phone* | 0.0826 | 28.8 | 0.0807 | 28.2 | 0.0807 | 28.4 | 0.0920 | 29.1 | 0.0809 | 28.1 | 0.0786 | 29.3 |
| | *PDA* | 1.3211 | 28.1 | 1.2681 | 27.7 | 1.2688 | 27.8 | 1.6878 | 28.2 | 1.2700 | 28 | 1.2348 | 28.6 |

(Continued)

| *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *mpeg4 for pocket* | | *wmv7* | | *wmv8* | | *xvid* | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| J/S | PSNR | J/S | PSNR | J/S | PSNR | J/S | PSNR | J/S | PSNR | J/S | PSNR |
| 0.0814 | 28.7 | 0.0815 | 29.4 | 0.0835 | 30.1 | 0.0815 | 28.3 | 0.0836 | 28.1 | 0.0830 | 29.2 |
| 1.3023 | 28.1 | 1.3191 | 28.7 | 1.3776 | 29.1 | 1.3163 | 28.1 | 1.3380 | 28.1 | 1.3284 | 29.0 |

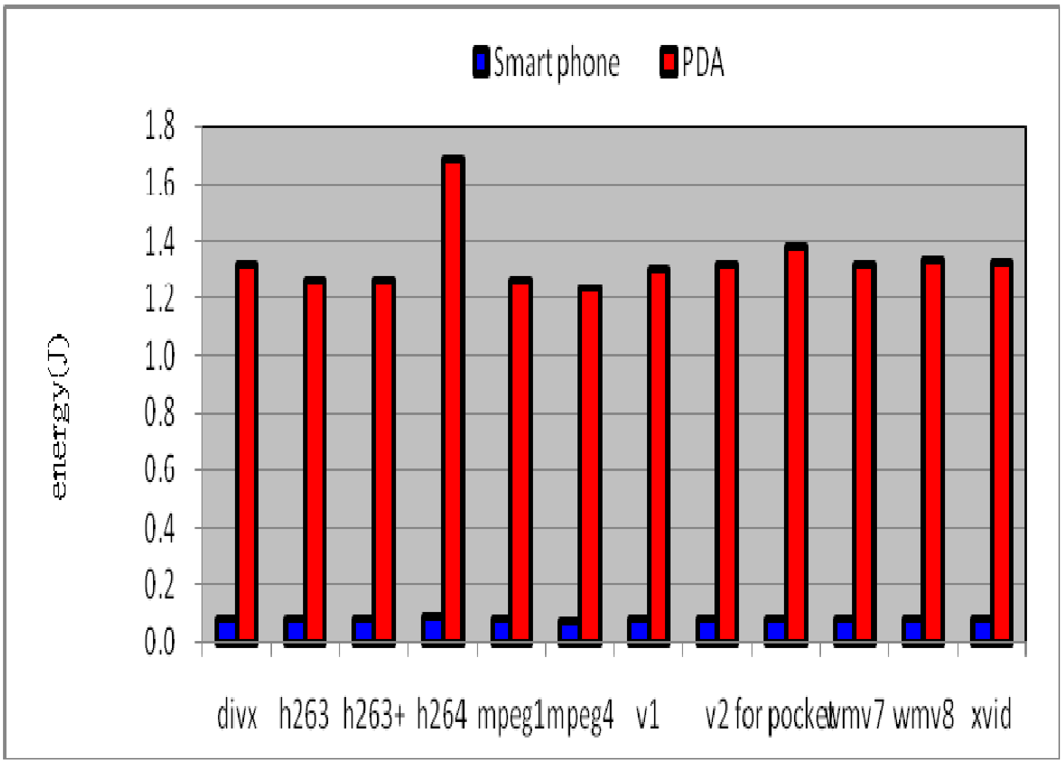


Figure 16. Power consumption of Smart phone and PDA (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI)

From Figure 16 and Table 5, we find films replaying on the smart phone have larger PSNR values than those of the same films replaying on the PDA. It is because that the smart phone has better hardware resource than the PDA, and so it improves encoding efficiency of all the codecs. Films encoded with mpeg4 for pocket has best quality picture, but they consume most energy, while films encoded with ms_mpeg4_v2 has second best and consume less energy.

Table 6. Power consumption of HTC (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI)

| *HTC* | *divx* | | *h263* | | *h263+* | | *h264* | | *mpeg1* | | *mpeg4* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *The Simpsons* | 0.0826 | 28.8 | 0.0807 | 28.2 | 0.0807 | 28.4 | 0.0920 | 29.1 | 0.0809 | 28.1 | 0.0786 | 29.3 |
| *foreman* | 0.2355 | 28.7 | 0.2471 | 28 | 0.2475 | 28.3 | 0.2548 | 29 | 0.2397 | 27.7 | 0.2389 | 29.1 |
| *carphone* | 0.2539 | 28.5 | 0.2567 | 27.7 | 0.2603 | 28.5 | 0.2700 | 28.8 | 0.2616 | 27.6 | 0.2509 | 29 |

(Continued)

| *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *mpeg4 for pocket* | | *wmv7* | | *wmv8* | | *xvid* | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| 0.0814 | 28.7 | 0.0815 | 29.4 | 0.0835 | 30.1 | 0.0815 | 28.3 | 0.0836 | 28.1 | 0.0830 | 28.7 |
| 0.2398 | 28.8 | 0.2500 | 29.6 | 0.2444 | 30.3 | 0.2466 | 28.3 | 0.2475 | 28.7 | 0.2457 | 28.5 |
| 0.2664 | 28.1 | 0.2723 | 29.0 | 0.2623 | 29.8 | 0.2756 | 27.3 | 0.2691 | 28.3 | 0.2763 | 28.7 |

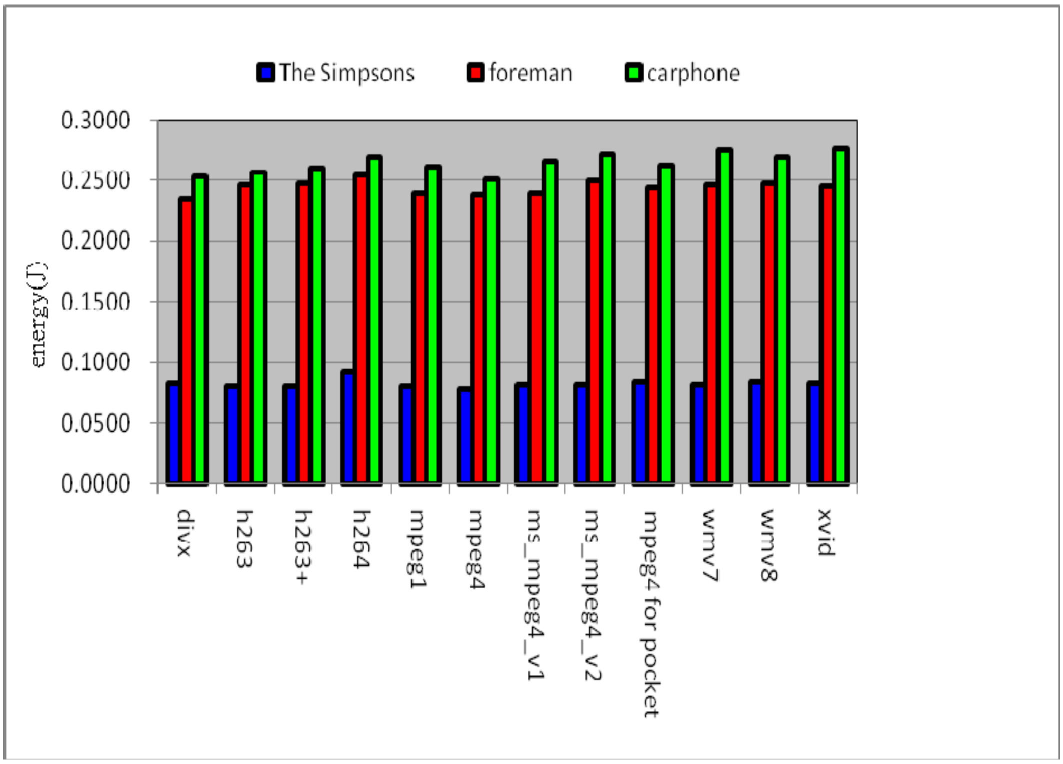


Figure 17. Power consumption of HTC (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI

From Figure 17. and Table 6. films encoded by ms_mpeg4_v2 have best picture quality but consume most power. Regardless of scene change character, films encoded by mpeg4 have rather good picture quality and consume less power.

Experiment 3:

Table 7. Power consumption of download and play after download; Windows Media Service (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

| *HTC* | *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *wmv7* | | *wmv8* | | *wmv9* | | *wvc1* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *foreman* | 0.1621 | 27.8 | 0.1638 | 28.3 | 0.1472 | 26.2 | 0.1885 | 27.2 | 0.2451 | 28.4 | 0.1555 | 28 |
| *carphone* | 0.1607 | 27.1 | 0.1638 | 27.7 | 0.1608 | 27 | 0.1816 | 26.8 | 0.1921 | 28.2 | 0.1523 | 27.6 |

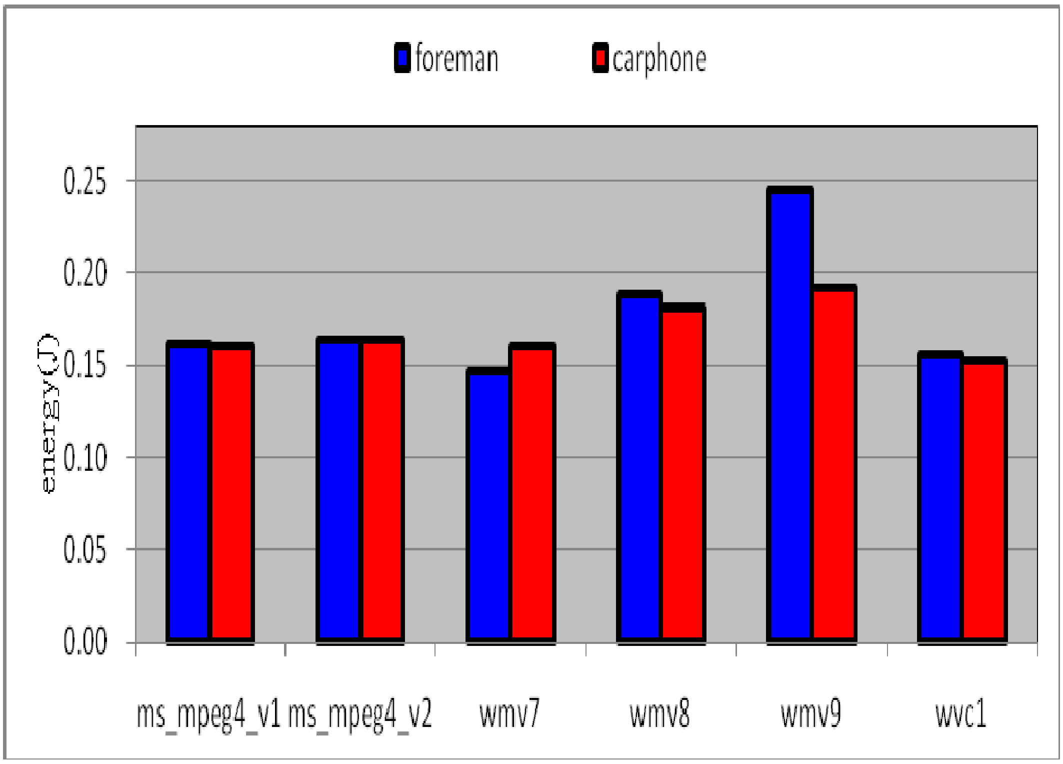


Figure 18. Power consumption of download and play after download; Windows Media Service (resolution: 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV)

From Figure 18. and Table 7. films encoded by WMV9 have best picture quality, while films encoded by ms_mpeg4_v2 have good picture quality and are relatively energy efficient.

Table 8. Power consumption of Windows Media Service (resolution: 320×240, Bit rate: 1152kbps, frame rate:25 frame/sec, video format: ASF)

| *HTC* | *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *wmv7* | | *wmv8* | | *wmv9* | | *wvc1* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *foreman* | 0.1351 | 26.3 | 0.1331 | 26.8 | 0.1206 | 25.3 | 0.1598 | 26.6 | 0.1442 | 27.1 | 0.1307 | 25 |
| *carphone* | 0.1398 | 26.2 | 0.1388 | 26.7 | 0.1237 | 24 | 0.1430 | 25.8 | 0.1561 | 27.2 | 0.1209 | 24.6 |

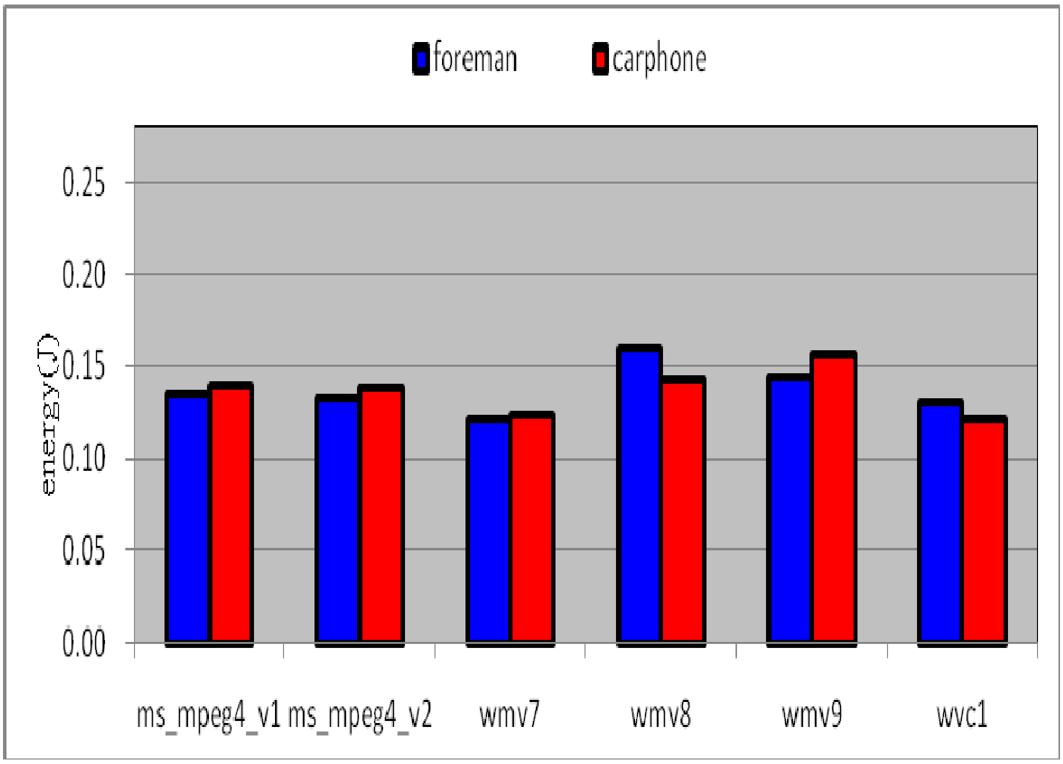


Figure 19. Power consumption of Windows Media Service (resolution: 320×240, Bit rate: 1152kbps, frame rate:25 frame/sec, video format: ASF)

From Figuer 19. and Table 8. among containers of WMV, ms_mpeg4_v2 brings best picture quality and is relatively energy efficient, while WMV7 gives least picture quality and consumes least power. Energy consumptions of Foreman films are lower than those of carphone films.

Experiment 4:

Table 9. Power consumption of You Tube (resolution of 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV )

| *HTC* | *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *wmv7* | | *wmv8* | | *wmv9* | | *wvc1* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *foreman* | 0.1670 | 26.3 | 0.1687 | 26.4 | 0.1516 | 25.6 | 0.1942 | 26 | 0.2526 | 26.6 | 0.1602 | 26.4 |
| *carphone* | 0.1659 | 26 | 0.1691 | 26.2 | 0.1660 | 25.3 | 0.1874 | 25.3 | 0.1982 | 26.2 | 0.1572 | 25.8 |

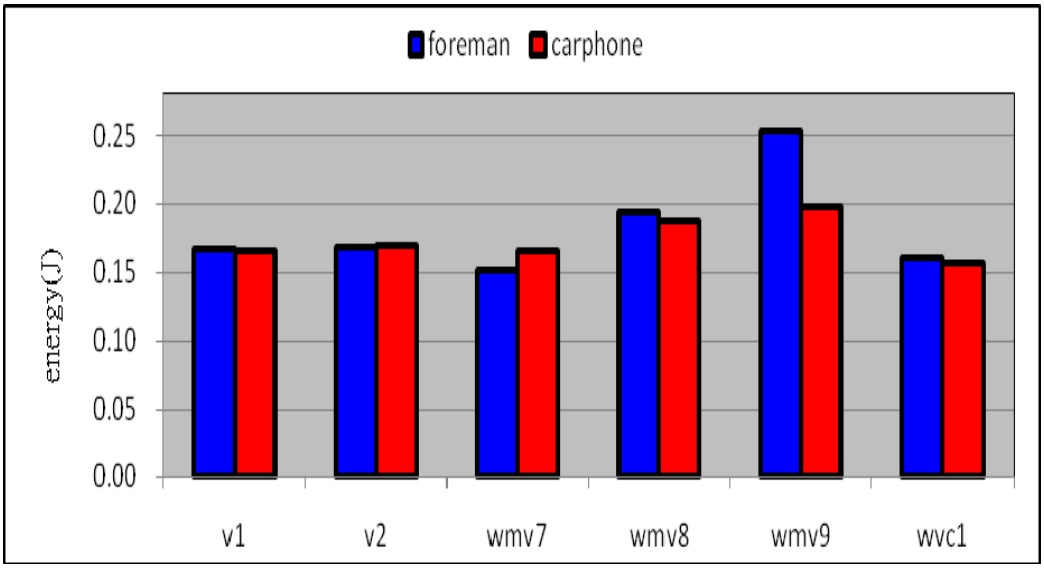


Figure 20. Power consumption of You Tube (resolution of 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: WMV )

From Figure 20. and Table 9. among containers of WMV, ms_mpeg4_v1 and ms_mpeg4_v2 bring best picture quality and are relatively energy efficient, while WMV7 gives least picture quality and consumes least power. No apparent trend is found regarding comparison of energy consumptions of Foreman films with those of carphone films.

Table 10. Power consumption of You Tube (resolution of 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI)

| *HTC* | *divx* | | *flash_vido* | | *h263* | | *h263+* | | *h264* | | *mpeg1* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| *foreman* | 0.1524 | 28.7 | 0.1663 | 28.8 | 0.1699 | 27.7 | 0.1769 | 28.5 | 0.1798 | 29.4 | 0.1741 | 27.0 |
| *carphone* | 0.1365 | 28.6 | 0.1660 | 28.4 | 0.1591 | 27.3 | 0.1757 | 28.5 | 0.1643 | 28.8 | 0.1636 | 26.7 |

(Continued)

| *mpeg2* | | *mpeg4* | | *ms_mpeg4 v1* | | *ms_mpeg4 v2* | | *mpeg4 for pocket* | |
|---|---|---|---|---|---|---|---|---|---|
| *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| 0.1979 | 27.3 | 0.1677 | 29.3 | 0.1584 | 29.3 | 0.2001 | 29.5 | 0.2610 | 29.8 |
| 0.1848 | 26.7 | 0.1584 | 29.1 | 0.1280 | 29.3 | 0.1383 | 29.4 | 0.2063 | 29.7 |

(Continued)

| *wmv7* | | *wmv8* | | *wmv9* | | *wmvc1* | | *xvid* | |
|---|---|---|---|---|---|---|---|---|---|
| *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* | *J/S* | *PSNR* |
| 0.1445 | 26.1 | 0.1603 | 26.3 | 0.1646 | 28.0 | 0.1612 | 28.2 | 0.1723 | 28.3 |
| 0.1464 | 26 | 0.1528 | 26.2 | 0.1562 | 27.5 | 0.1542 | 28 | 0.1357 | 28 |

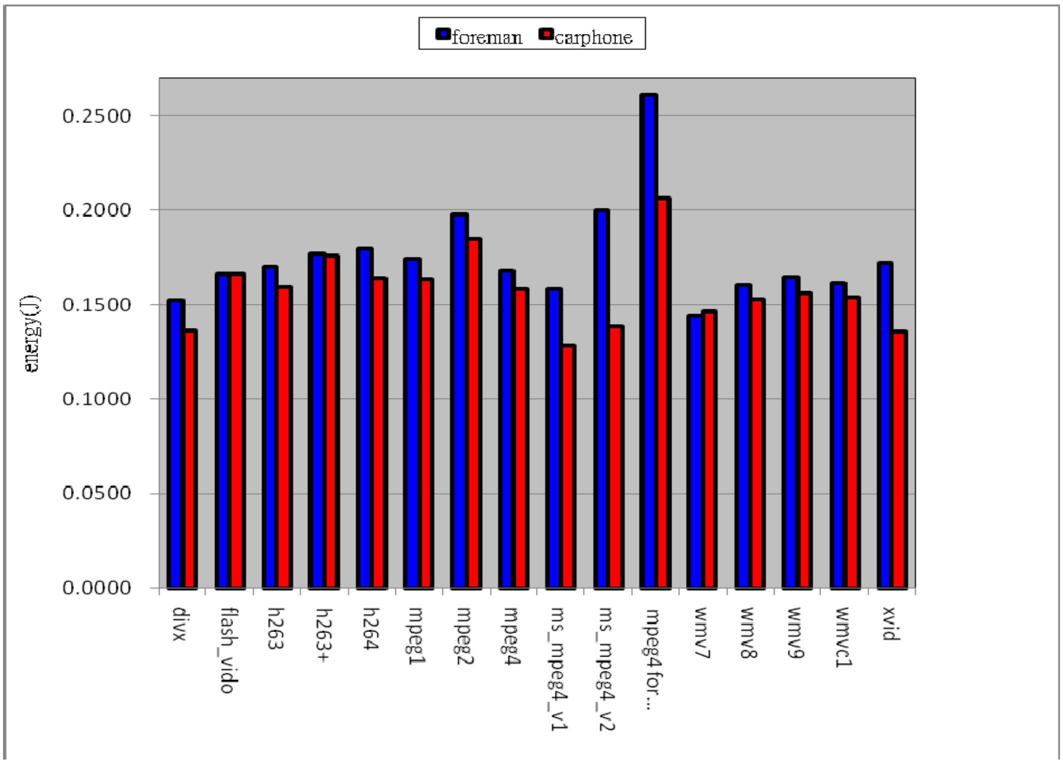


Figure 21. Power consumption of You Tube (resolution of 320×240, bit rate: 1152kbps, frame rate: 25 frame/sec, video format: AVI)

From Figure 21 and Table 10, flash_ video has a format closest to the real broadcasting format, and it behaves averagely in power consumption and picture quality. The mpeg4 series and h264 have best picture quality and mpeg4 consumes relatively less energy. In view of effect of scene change types, Energy consumptions of Foreman films are greater than those of carphone films.

# Chapter 5
# Conclusions

From the experimental results, we conclude that

- We use Y component of YCbCr to compute PSNR values for comparison of picture quality and find that the PSNR values agree with what we see by the eyes, so, we can use this simple approach to decide picture quality.
- When replaying on the smart phone or PDA, videos encoded with bit rates of 576 kbps or more have good picture quality (with PSNR values more than 26.31, which is acceptable for most people).
- In view of scene changes, still or nearly motionless films consume far less power than fast changing scene films.
- When streaming by adopting Windows Media Service, WMV9 videos show best picture quality; however, when compromising picture quality with power consumption, ms_mpeg4_v2 is the best choice.
- When playing video streaming via You Tube, films encoded by mpeg4 series codecs and H264 have best picture quality, and films encoded by mpeg4 codec consume least power.

In the future work, we plan to formulate an algorithm to automatically analyze content of video and decide the most suitable codecs and parameters of encoding to produce video codes that have suitable quality and energy efficient when replayed on the mobile devices.

# Bibliography


[1] G. Cote, B. Erol, M. Gallant, F. Kossentini, H.263+: video coding at low bit rates, IEEE Trans. on Circuits and Systems for Video Technology, Nov. 1998, pp. 849–866.

[2] T.B. Atwater, P.J. Cygan, F.C. Leung, Man portable power needs of the 21st century. I. Applications for the dismounted soldier. II. Enhanced capabilities through the use of hybrid power sources, Journal of Power Sources 91 (1) (2000) 27–36.

[3] W. Ashmawi, R. Guerin, S. Wolf, M. Pinson, On the impact of policing and rate guarantees in DiffServ networks: a video streaming application perspective, Proceedings of the 2001 ConferenceOn Applications, Technologies, Architectures,And Protocols For Computer Communications, California, United States, Aug. 2001, pp. 83–95.

[4] Tong-Xiao Chen,, Zhen-Cheng Zhang, Guo-Feng Huang ,Digital image processing, Unalis Corporation，2001

[5] K. Lahiri, A. Raghunathan, S. Dey, D. Panigrahi, Battery driven system design: a new frontier in low power design, Proc. ASP-DAC/VLSI Design 2002, Bangalore, India, January 2002, pp. 261–267.

[6] N.R. Potlapally, S.R., A. Raghunathan, N.K. Jha, Analyzing the power consumption of security protocols, Proceedings of the 2003 International Symposium on Low Power Electronics and Design, Seoul, Korea, 2003, pp. 30–35.

[7] Applications and requirements for scalable video coding, Mar. 2003

MPEGdocument ISO/IEC JTC1/SC29/WG11 N5540.

[8] T.K. Tan, A. Raghunathan, N.K. Jha, A simulation framework for energy-consumption analysis of OS-driven embedded applications, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, Sept. 2003, pp. 1284–1294.

[9] Zhenhua Jiang, Lijun Gao, M.J. Blackwelder, R.A. Dougal, Design and experimental tests of control strategies for active hybrid fuel cell/battery power sources, Journal of Power Sources 130 (2004) 163–171.

[10] Kejariwal, S. Gupta, A. Nicolau, N. Dutt, R. Gupta, Energy efficient watermarking on mobile devices using proxy-based partitioning, IEEE Trans. on Very Large Scale Integration (VLSI) Systems, June 2006, pp. 625–636.

[11] Chu-Hsing Lin, Jung-Chun Liu, Chun-Wei Liao, Energy analysis of multimedia video decoding on mobile handheld devices, 2007 International Conference on Multimedia and Ubiquitous Engineering, Seoul, Korea, April 26–28, 2007, pp. 120–125.

[12] Chu-Hsing Lin, Jung-Chun Liu, Chun-Wei Liao, Power consumption analysis of audio applications on mobile handheld devices, The IEEE TENCON 2007, Oct 30–Nov 2, 2007.

[13] C. Stone, Fuel cell technologies powering portable electronic devices, Fuel CellsBulletin 2007 (10) (October, 2007) 12–15.

[14] T.P. Brock, S.R. Smith, Using digital videos displayed on personal digital assistants(PDAs) to enhance patient education in clinical settings, International Journal of Medical Informatics 76, Issue 11–12

(November–December, 2007) 829–835.

[15] Chu-Hsing Lin, Jung-Chun Liu, and Mao-Hua Cheng, “Energy analysis of multimedia video decoding on embedded systems,” The 22nd edition of the International Conference on Information Networking, Jan. 25–Jan 28, 2008.

[16] Chu-Hsing Lin, Jung-Chun Liu, Mao-Hua Cheng, Tsung-Che Yang and Mei-Chun Chou, Energy Analysis of Multimedia Video Streaming on Mobile Devices, Springer Berlin / Heidelberg ,Lecture Notes in Computer Science ,June,2009,pp. 347–357

[17] Chu-Hsing Lin, Jung-Chun Liu, Chun-Wei Liao, Energy analysis of multimedia video decoding on mobile handheld devices, Computer Standards & Interfaces 32 ,2010,pp. 10–17

[18] National Instruments Corp. http://www.ni.com.

[19] Overview of the MPEG-4 Standard: http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm. June, 2010.

[20] DivXNetworks, Inc. [Online]. Available: http://www.divx.com/divx/.

[21] XviD Software Package [Online]. Available: http://www.xvid.org/.

[22] SUPER©: http://www.erightsoft.com/SUPER.html.

[23] TCPMP: http://tcpmp.corecodec.org/.

[24] Acer n300: http://www.acer.com.tw/PRODUCTS/pda/n300.htm.

[25] HTC Touch Diamond 2: http://www.htc.com/www/product/touchdiamond2/overview.html

[26] http://trace.eas.asu.edu