

第五章 實驗結果與討論

本章我們將介紹系統實作的結果，並且介紹整個系統的實驗環境，以及實驗的方法與其數據的討論。本實驗的討論主要針對上一章提出四種 Level 的 DropFrame 機制進行數據分析與討論。

5.1 實驗環境介紹

本論文的遠端監控系統測試環境，選擇在台灣師範大學的校本部與台灣師範大學分部進行遠端點對點的測量。我們將本論文提出的 SIP-Cam 架設在台師大本部綜合大樓 10 樓，從綜合大樓 10 樓的窗口監控大樓樓下的道路車流量的情況。接著，我們在師大分部應用科技大樓使用本論文所提到的 PDA 進行影像的監測。PDA 採用無線網路進行測試，並利用 Kaise 電子公司提供的 802.11b 無線網卡作為網路連線的媒介。

台師大本部與台師大分部實體距離約為 2.4 公里。而網路環境方面，從台師大分部連線到台師大本部的 SIP-Cam，總共經過了 8 台機器 (switch 或 router) (如下頁圖 5.1)。整體的實驗環境架構就如下圖 5.2 所示。

```
[root@ruihong root]# traceroute 140.122.87.108
traceroute to 140.122.87.108 (140.122.87.108), 30 hops max, 38 byte packets
 1  140.122.184.254 (140.122.184.254)  7.709 ms  1.607 ms  0.633 ms
 2  140.122.134.1 (140.122.134.1)    0.679 ms  6.391 ms  0.652 ms
 3  140.122.128.254 (140.122.128.254) 1.833 ms  0.372 ms  0.345 ms
 4  140.122.128.28 (140.122.128.28)  1.139 ms  1.009 ms  0.886 ms
 5  140.122.128.1 (140.122.128.1)   1.473 ms  1.295 ms  1.260 ms
 6  140.122.6.209 (140.122.6.209)    1.014 ms  1.025 ms  0.949 ms
 7  140.122.6.58 (140.122.6.58)     1.274 ms  1.464 ms  1.290 ms
 8  140.122.87.108 (140.122.87.108)  4.406 ms  3.895 ms  3.935 ms
[root@ruihong root]#
```

圖 5.1

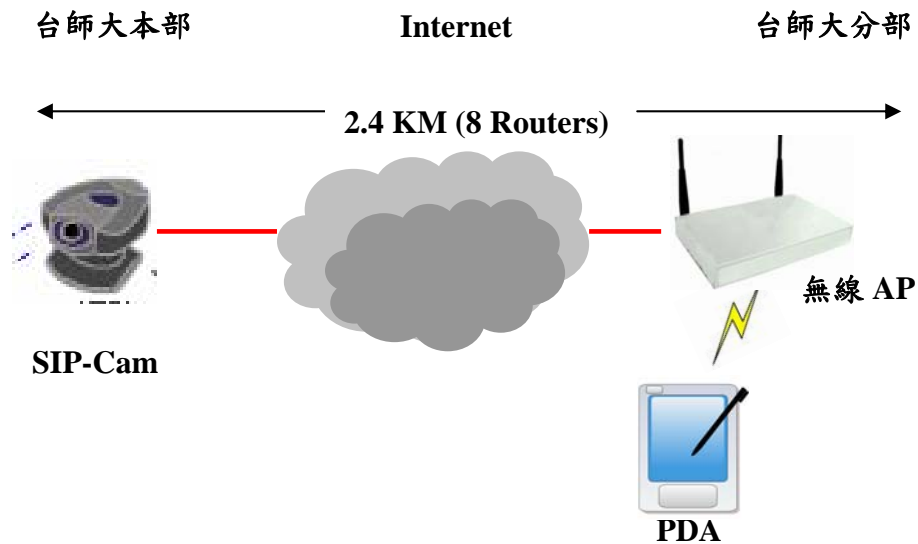


圖 5.2

本論文第 3 章曾提到 SIP-Cam 的影像輸出格式有四種，分別為 D1(720*480)、half-D1(720*240)、CIF(352*240)以及 QCIF(176*144)。本章節的實驗中(除了 5.5 節外)皆採用 D1 的影像格式，並且設定每秒 30 張影像畫面(30 frames/sec)。

下圖 5.3 為我們使用本論文提出的 PDA 進行實驗的情況，圖 5.3 的右上角為 PDA 所使用的無線網卡，右下角則是輸出顯示的 LCD 觸控式螢幕。

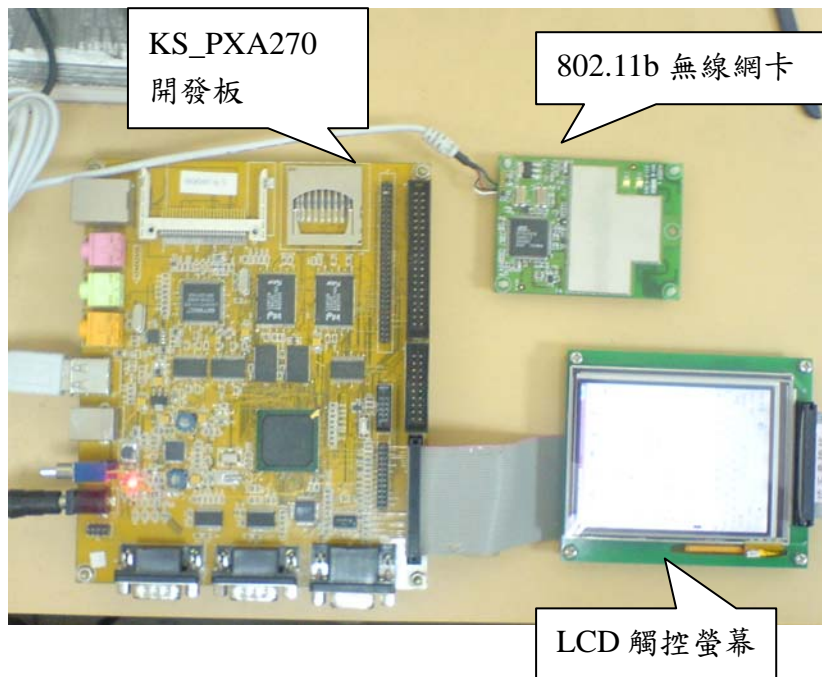


圖 5.3

下圖 5.4 是透過 PDA 擷取 SIP-Cam 即時影像的快照圖，圖中所顯示的即為台師大本部綜合大樓樓下的交通流量情形。



圖 5.4

5.2 網路使用頻寬的比較

網路使用頻寬的比較實驗中，我們基於上述圖 5.2 的實驗架構，在 PDA 端擷取位於師大本部的 SIP-Cam 影像。我們分別以四種不同的模式進行實驗，在固定的時間內抓取影像資料，再計算出平均的頻寬使用量進行比較。

這個實驗進行的方式，我們利用 wget 這個指令，分別針對四種不同模式擷取 SIP-Cam 的影像，其擷取時間固定為 5 分鐘，再計算其頻寬使用率。

```
[root@ruihong html]#  
[root@ruihong html]#  
[root@ruihong html]# wget http://140.122.87.108/vdata.v  
--11:30:02-- http://140.122.87.108/vdata.v  
=> `vdata.v'  
Connecting to 140.122.87.108:80... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: unspecified [text/plain]  
  
[ <=> ] 206,105 104.40K/s
```

圖 5.5

經過四種不同模式的實驗後，我們可以得到下圖 5.6 的數據。從 5.6 的圖表可得知，網路頻寬使用最高的是 Level1，最低的為 Level4。而且從 Level1 到 Level4 可以很明顯看出網路所使用的頻寬穩定的下降，尤其 Level3 及 Level4 與完全不丟棄封包的 Level1 相比，頻寬使用率至少降低了一半以上。這可以顯示當 PDA 選擇使用 Level3 或 Level4 時，PDA 所需要解碼的封包大為減少，對於 PDA 的運算負擔也相對減輕許多。

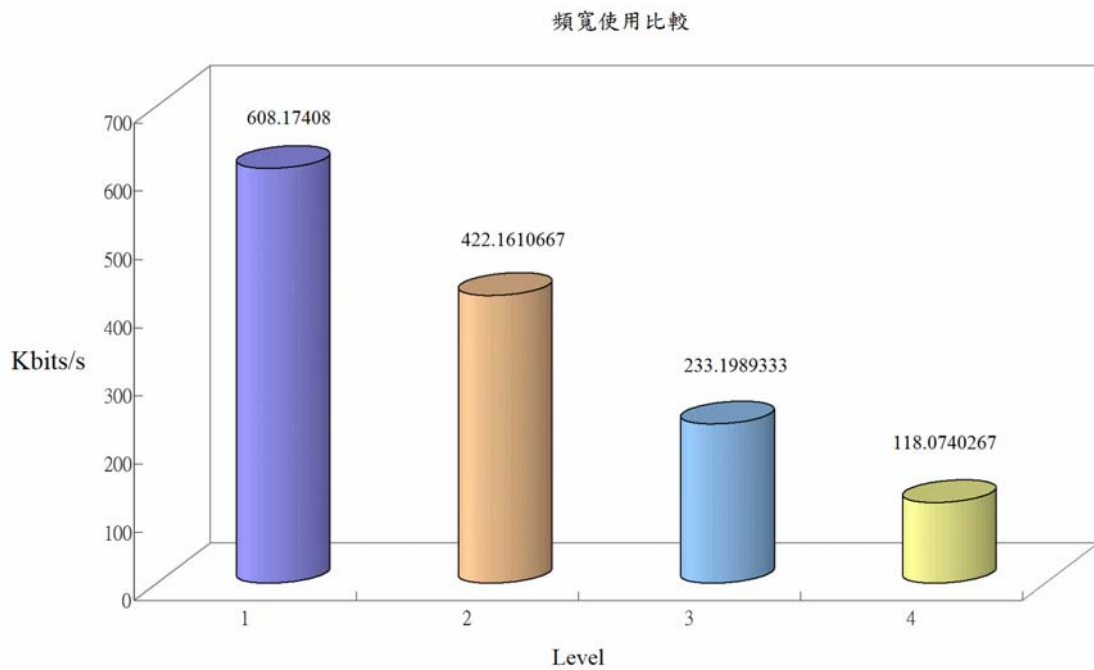


圖 5.6

5.3 TCP Window Size 的比較

TCP 通訊協定中,Header 的欄位有一項為 Window Size,如圖 5.7。TCP Window Size 代表接收端允許傳送端傳送的資料量。所以當接收端的 TCP Window Size 越大,也就代表接收端所能接受的資料量越多。反之,若接收端的 TCP Window Size 越小,就代表接收端能接受的資料量越少。換句話說,假設 TCP Window Size 由大變小,則代表接收端的處理能力無法負荷目前的傳輸量,所以必須藉著調整 TCP Window Size 來告知傳送端降低傳送量,以避免接收端無法負荷的情形。

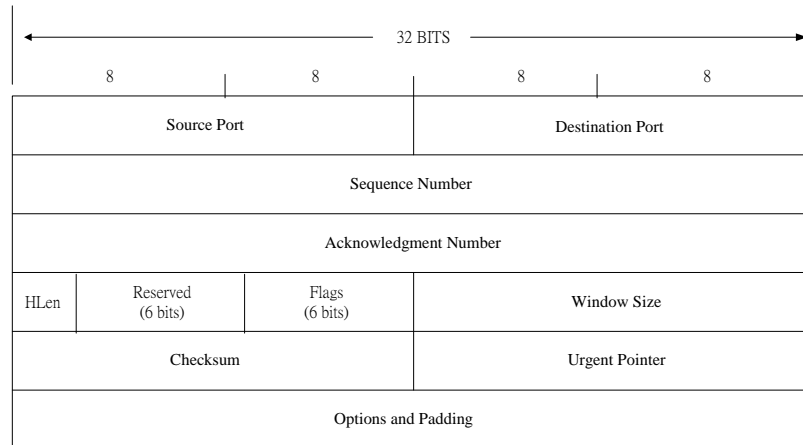


圖 5.7

這個實驗我們基於上述圖 5.2 的架構，並且使用每秒 30 張 D1 的影像畫面作為實驗的依據，再利用 Etherdump 這套開放原始碼的封包監聽程式進行網路封包的擷取。我們在 PDA 所使用的無線 AP 內安裝 Etherdump 這套軟體，藉由 AP 來擷取 PDA 回傳給 SIP-Cam 的封包，其架構示意圖如圖 5.8。最後，我們利用 Etherdump 擷取的封包資訊進行 TCP Window Size 的分析。

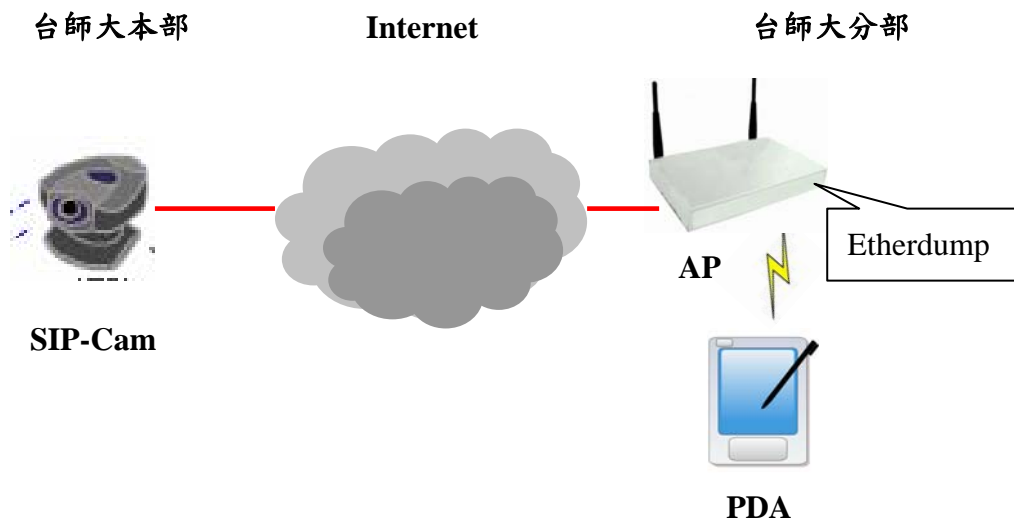


圖 5.8

下圖 5.9 是我們針對四種不同的 Level 進行 TCP Window Size 的比較。圖表中的 y 軸為 TCP Window Size 的大小，x 軸則為時間。其中的每個點代表 20 秒內 TCP Window Size 的平均大小。藉由 20 秒內的平均值，我們可以清楚地觀察到 TCP Window Size 的變化趨勢。

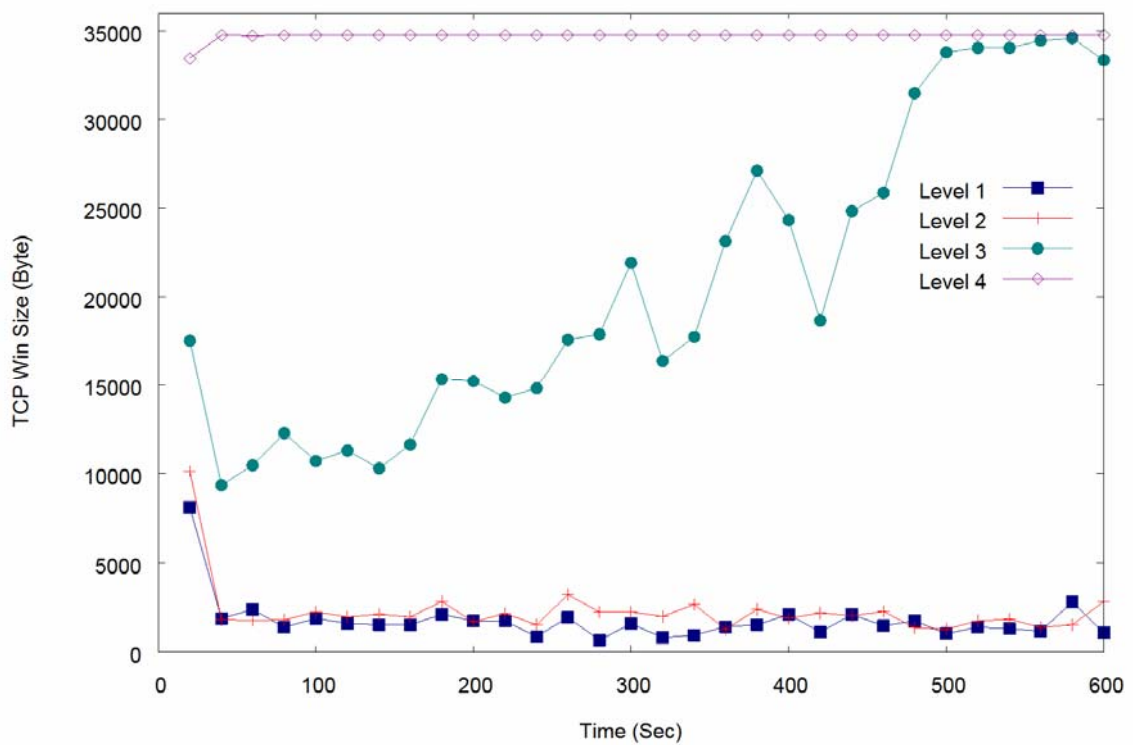


圖 5.9

由圖 5.9 可知，Level1 沒有丟棄任何封包，其 TCP Window Size 一開始就由 10K 左右劇降至 1~2K，接著一直維持在 1~2K 的位置，代表沒有進行封包丟棄時，

PDA 明顯有處理速度跟不上的趨勢，導致 TCP Window Size 劇降後維持在偏低的位置。Level2 丟棄了一半的 P-Frame，其 TCP Window Size 的情況和 Level1 相當類似，剛開始也是劇降後維持在 1~2K 的位置，然而 Level 2 的曲線又比 Level 1 稍微高一點點。

Level3 和 Level4 的 TCP Window Size 則有比較明顯的提升。Level3 丟棄了所有的 P-Frame，TCP Window Size 則呈現逐步上升的情況，代表使用 Level3 時，PDA 的處理速度有明顯改善的跡象，所以能夠接收封包的程度變大，而提高了 TCP Window Size 的大小。Level4 丟棄了所有的 P-Frame 以及一半的 I-Frame，TCP Window Size 穩定的維持在 35K 左右，代表 PDA 經過 Level4 的封包丟棄後，處理速度維持在很順暢的狀態，以致於 TCP Window Size 可以穩定的維持在較高的位置。

從圖 5.9 的數據比較來看，Level3 與 Level4 的 TCP Window Size 有明顯的提升，也代表我們所採用的 DropFrame 機制中，Level3 與 Level4 有效的減輕了 PDA 在解碼時的負擔。

5.4 解碼使用時間的比較

本論文 5.4 節將討論在四種不同 Level 的情況下，PDA 解碼所使用時間的比較，藉由此實驗觀察 DropFrame 的機制是否能有效的降低 PDA 解碼所承受的負擔。

這個實驗中，我們設定 SIP-Cam 的動態影像模式為每秒 30 張 D1 影像，接著我們分別針對 Level1、Level2、Level3 以及 Level4 各自錄製 5 分鐘的監控影像，再將錄好的影像置於 Web Server 上，最後利用 PDA 播放位於 Web Server 上的影片，並且統計 PDA 對於四種不同 Level 的影像其解碼所花費的時間。

實驗過程中，我們錄製 SIP-Cam 影像所使用的平台為 Linux 的 PC，其作業系統為 Fedora 4。我們使用 Linux 系統中的 wget 指令擷取 SIP-Cam 的資料，並且設定擷取時間為 5 分鐘，最後將擷取的資料放置於此 Linux PC 的 Web Server 上。PDA 再透過此 Web Server 進行影像的播放，並且統計不同 Level 解碼所使用的時間。

下表 5.1 為上述實驗所得到的統計數據表。即 PDA 播放預先錄製好的 5 分鐘影像，其四種 Level 分別所需要花費的解碼時間。圖 5.10 則是利用表 5.1 的數據

所畫出來的立體直條分析圖。

表 5.1 D1 影像解碼時間比較

	Level1	Level2	Level3	Level4
解碼時間(秒)	1197.0715	712.2281	183.2120	91.1388

D1 影像解碼時間比較圖

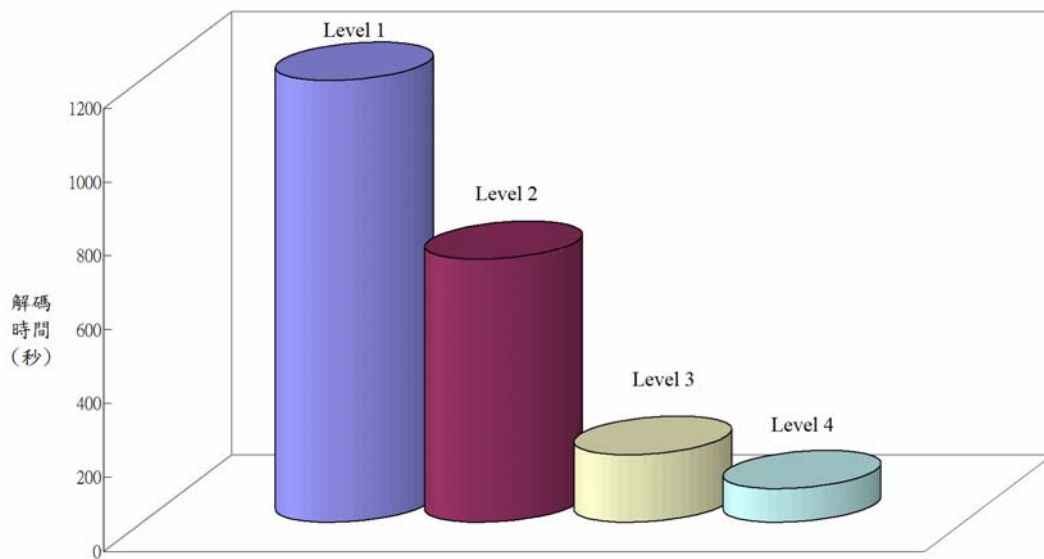


圖 5.10

由表 5.1 可得知，Level1 解碼所需要花費的時間約為 1197 秒(約 20 分鐘)。一段錄製 5 分鐘的影片，PDA 解碼卻需要花費 20 分鐘左右，很明顯可以看出 PDA 使用 Level1 時，其解碼的速度是過於緩慢的。而使用 Level2 時，解碼所需花費

的時間為 712 秒(約 12 分鐘)。雖然丟棄部分的 P-Frame 使得 PDA 解碼時間減少了，但解碼時間仍然超過 5 分鐘，也代表此時(使用每秒 30 張 D1 影像的狀況下)Level2 對於 PDA 解碼的負擔改善程度相當有限。

相較之下，從表 5.1 可得知 Level3 花費於解碼的時間大幅減少至 183 秒(約 3 分鐘)，代表丟棄所有的 P-Frame 以後，能有效的改善 PDA 解碼的負擔。而 Level4 所需要的解碼時間更只剩下 91 秒(約 1 分半鐘)，代表丟棄所有 P-Frame 與部分 I-Frame 以後，PDA 對於解碼的負擔大為減輕。

圖 5.10 可以清楚看出四種 Level 對於解碼時間的關聯，也很瞭解知道 Level3 與 Level4 對解碼時間的改善。由圖 5.10 可大概得知 Level2 對解碼時間稍有提升，但其解碼時間仍超過 5 分鐘，無法有效的改善解碼的負擔。然而由圖 5.10 可知，Level3 的解碼時間約只為 Level2 的四分之一，而 Level4 更只需要 level3 的二分之一左右，顯然 Level3 與 Level4 對於 PDA 的解碼有相當大的幫助。

由這些數據與圖表，我們可以知道，如果 SIP-Cam 選擇使用較高解析度的影像模式(如每秒 30 張 D1 影像)，PDA 適合選擇 Level3 或者 Level4 的封包丟棄機制，以避免 PDA 無法負荷 SIP-Cam 壓縮影像的速度。

5.5 QCIF 與 D1 的比較

本論文 5.5 節的實驗是針對 SIP-Cam 使用每秒 30 張 QCIF 與 D1 兩種不同影像播放模式進行數據的比較與分析。在此實驗中，我們想要比較相同監控時間內 (5 分鐘)，D1 與 QCIF 兩種不同的動態影像模式其解碼所需的時間。

此實驗的環境及實驗的方法與 5.4 節相同，我們先預先錄製 5 分鐘的 SIP-Cam 影像，再使用 PDA 進行播放，以比較 D1 與 QCIF 兩種影像模式的解碼時間。

下表 5.2 為 PDA 分別對於 D1 與 QCIF 兩種影像播放模式，播放預先錄製的 5 分鐘影像，而得到的解碼時間比較數據。

表 5.2 QCIF 影像解碼時間比較

	Level1	Level2	Level3	Level4
QCIF 解碼時間(秒)	228.4057	129.9446	30.6619	15.3838
D1 解碼時間(秒)	1197.0715	712.2281	183.2120	91.1388

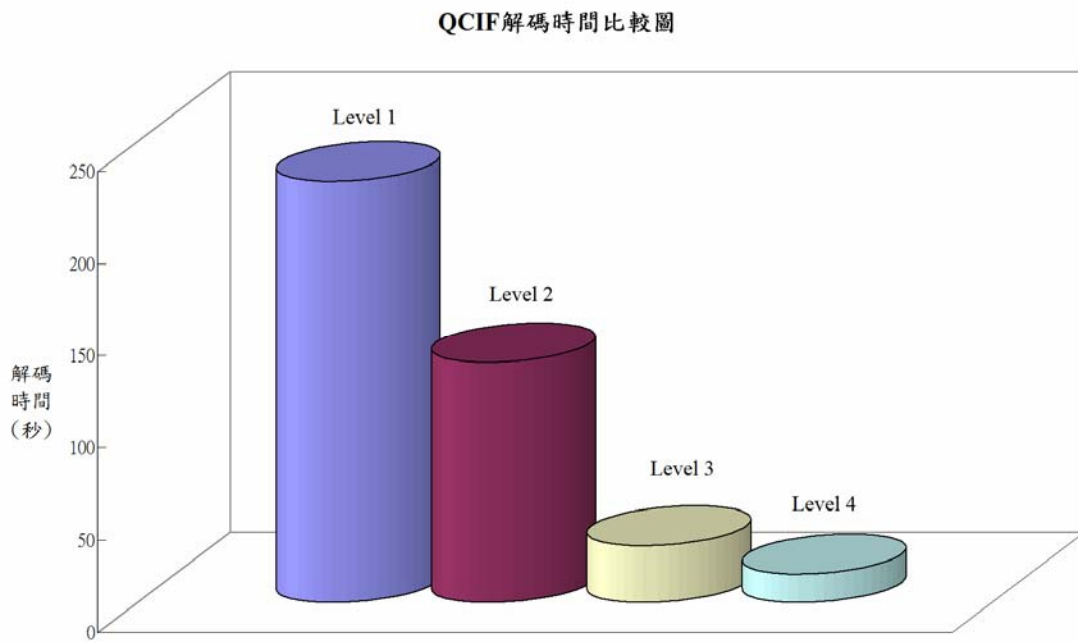


圖 5.11

上圖 5.11 則是利用表 5.2 的數據，將 PDA 對於 QCIF 影像解碼花費的時間所繪製的立體直條分析圖。圖 5.11 中，PDA 對於四種不同 Level 的解碼時間其呈現的比例與圖 5.10 中所呈現的比例差不多。然而比較大的差別在於 Y 軸的解碼時間，下圖 5.12 則是針對 QCIF 與 D1 兩種模式的比較圖。

由圖 5.12 可以知道，如果同樣選擇 Level1 的封包丟棄模式，當 SIP-Cam 使用解析度較小的影像模式(如 QCIF)與解析度較大的影像模式(如 D1)時，PDA 花費在 D1 影像模式的解碼時間約為 20 分鐘，而花費在 QCIF 影像模的解碼時間只

需要 4 分鐘不到，可以明顯看出兩者差距甚大，而且 QCIF 的四種 Level 都能在 5 分鐘以內完成解碼的動作。

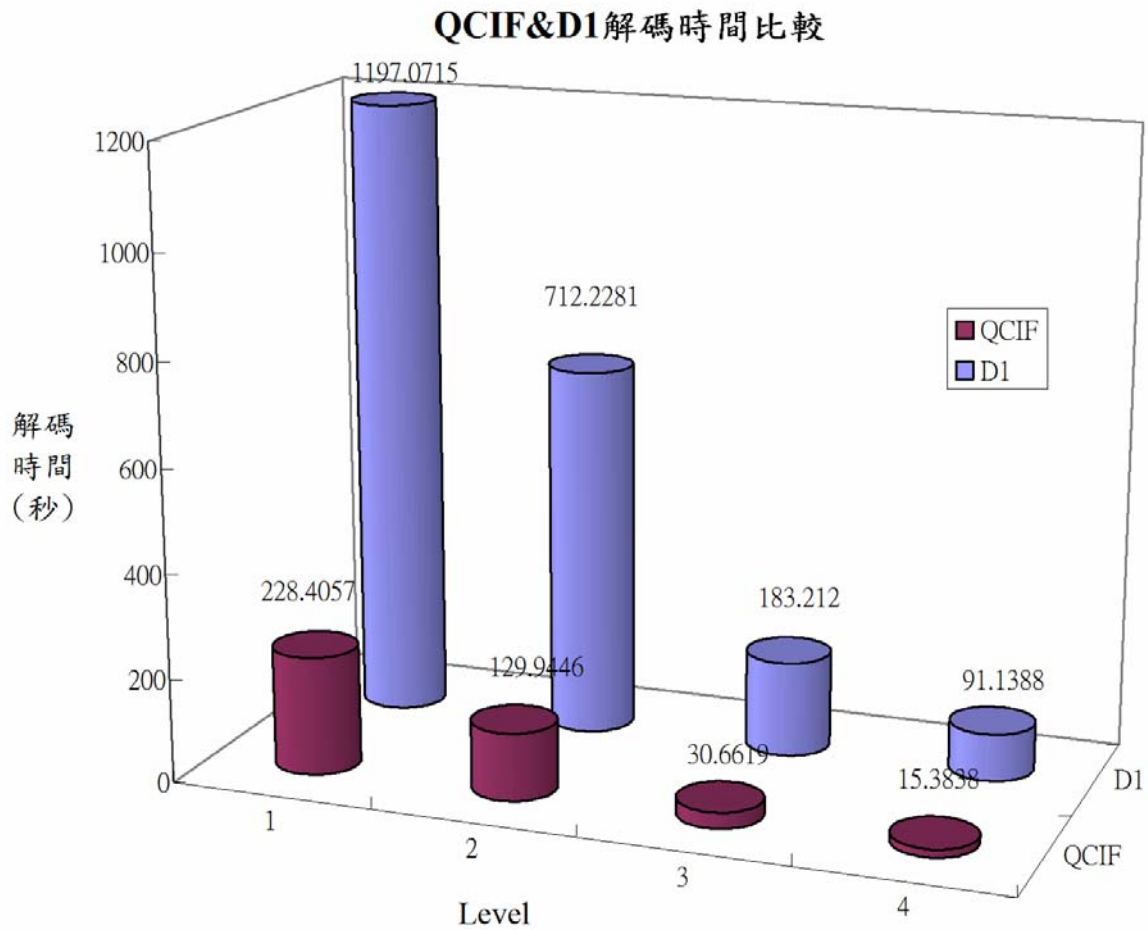


圖 5.12

換句話說，如果 SIP-Cam 設定的影像模式解析度較小時(如 QCIF)，PDA 端即使選擇丟棄較少封包的 Level1 或 Level2，也不至於有負荷不了的情況發生。

再者，我們可以從圖 5.12 中發現，QCIF 的 Level1 解碼所需時間其實只比 D1 的 Level3 多一點，也就是說如果 SIP-Cam 使用較小解析度的影像模式(如 QCIF)，PDA 選擇 Level1 的封包丟棄模式，其解碼的效能就相當於 SIP-Cam 使用較大解析度的影像模式(如 D1)，而 PDA 選擇 Level3 封包丟棄模式的情況。

5.6 結論

從以上幾個實驗來看，本論文提出的遠端監控系統使用 DropFrame 的機制時，可以有效的減輕 PDA 的負擔，也能使 PDA 順暢地解碼影像。

當 SIP-Cam 傳送的影像解析度較大時(例如 D1)，PDA 端就比較適合採用 Level3 或者 Level4 的機制，適當的丟棄部分封包以減輕 PDA 解碼的負擔。反觀，如果 SIP-Cam 傳送較小解析度的影像(例如 QCIF)時，PDA 選擇 Level2 甚至 Level1，PDA 都足以應付 SIP-Cam 壓縮影像的速度。