

國立交通大學

資訊科學與工程研究所

碩士論文

H.264/AVC-SVC 分層視訊串流動態提取技術



Dynamic Layer Extraction in H.264AVC
Scalable Extension

研究生：楊植竣

指導教授：蕭旭峰、李素瑛

中華民國九十五年十月

H.264/AVC-SVC 分層視訊串流動態提取技術
Dynamic Layer Extraction in H.264AVC Scalable Extension

研 究 生：楊植竣

Student : Chih-Chun Yang

指導教授：蕭旭峰

Advisor : Hsu-Feng Hsiao

李素瑛

Suh-Yin Lee

國立交通大學
資訊科學與工程研究所
碩士論文



A Thesis

Submitted to Institute of Multimedia Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

October 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年十月

Dynamic Layer Extraction for H.264/AVC Scalable Extension (H.264/AVC-SVC 分層視訊串流動態提取技術)

研究生：楊植竣

指導教授：蕭旭峰、李素瑛 教授

國立交通大學

資訊科學與工程研究所碩士班

Abstract

目前 JVT 正在制訂包括於 AVC (或稱 H.264) 裡之可調式視訊編碼。其精神是在以 AVC 為基底來進行可調式視訊編碼，使視訊串流不僅具有 AVC 的高壓縮品質，且具備時間、空間、噪訊比等可調性，使系統不需經過重新轉碼即可因應不同使用者的軟/硬體及網路環境條件。

本論文首先介紹，在可調式視訊串流服務模組裡，為達成目標傳輸速率進行可調式視訊編碼 (H.264/AVC Scalable Video Coding) 視訊串流分析，以建構所需刪節或取用各可調性分層的提取技術。在介紹此可調式視訊提取技術演算法的同時，一併說明其程式模組及使用方法。

Table of Content

1.	Introduction	5
1.1.	Introduction to H.264/AVC Scalable Video Coding	5
2.	Previous Work on H.264/AVC SVC.....	6
2.1.	H.264/AVC SVC System Architecture.....	6
2.1.1.	Joint Scalable Video Model (JSVM).....	6
2.2.	Bitstream Information of H.264/AVC Scalable Extension.....	7
2.3.	Scalability of H.264/AVC Scalable Video Coding	10
2.3.1.	Spatial Scalability	10
2.3.2.	Temporal Scalability	10
2.3.3.	SNR Scalability	10
3.	Dynamic Layer Extraction	11
3.1.	Defect of Joint Scalable Video Model.....	11
3.2.	Syntax Elements	13
3.2.1.	Syntax Elements of NAL Unit Header	13
3.2.2.	Syntax Elements of Sequence Parameter Sets	14
3.2.3.	Syntax Elements of Picture Parameter Sets	14
3.2.4.	Syntax Elements of Slice Header	15
3.3.	Layer Extraction Algorithm.....	15
3.3.1.	Intra Key Picture and IDR Picture Converison	15
3.3.2.	Video Layer Extraction	18
4.	Experimental Results.....	23
4.1.	Dynamic Adaptation of H.264/AVC Scalable Video Coding.....	23
5.	Future Work.....	25
6.	REFERENCES	26

1. Introduction

1.1. Introduction to H.264/AVC Scalable Video Coding

視訊多媒體服務為目前極熱門的網路多媒體應用。尤其近來隨著手機運算能力的成長，透過手機觀賞數位電視已不再是夢想，行動影音服務的品質逐漸成為備受重視的新領域。但因為網路環境及封包傳輸行為的不同，必須依據其特性分別加以處理，如何能夠隨著現有頻寬狀況快速調整分層視訊串流可調性，以增進系統效能與影音服務品質便成為重要的課題。

因此本系統配合 JVT 的 SVC 制訂，發展以 AVC/H.264 為基底之可調式視訊編碼 (Scalable Video Coding) 作為串流服務之來源編碼，使視訊串流不僅具有 H.264/AVC 的高壓縮品質，而且具備時間、空間、噪訊比等可調性，不需經過重新轉碼 (transcoding) 即可因應不同使用者的軟/硬體及網路環境條件。雖然 JVT 已提供更動 H.264/AVC-SVC 視訊串流可調性組合之工具，但此工具僅能對完整之檔案進行操作，而且也必須限制視訊壓縮檔之可調性組合從頭至尾為一致，並不符合在網路上依端點至端點動態資源 (包含頻寬及計算能力) 分佈而調整 H.264/AVC-SVC 視訊串流可調性組合之要求。本文將逐步說明如何根據 JVT 目前所制定 SVC 之標準，依循時間軸上可調性組合的變化，動態分離出 H.264/AVC 之可調式視訊編碼 (Scalable Video Coding) 各分層視訊串流，以達成在異質網路上傳送同時性多速率視訊串流的需求，使本系統可即時提供適合的視訊資料品質及解析度，並符合邊接收邊播放的隨選視訊要求。

在第二章我們首先簡介 H.264/AVC SVC 的背景知識及系統架構；第三章說明分層動態提取系統的演算法，以及目前 JSVM 的限制；第四章列出相關實驗結果，第五章則為結語及後續發展；第六章為研究相關文件。

2. Previous Work on H.264/AVC SVC

2.1. H.264/AVC SVC System Architecture

為了在異質網路上傳送動態可變速率視訊串流的需求，並依據各獨立解壓縮單元至伺服器之間即時的網路狀況，藉由調整時間、空間、訊噪比等幾項可調性，以產生新的分層視訊串流組合，本文介紹透過修改 H.264/AVC-SVC 之 JSVM (Joint Scalable Video Model) 來建立適合的實驗環境以作為串流服務所需的來源編碼方式。

2.1.1. Joint Scalable Video Model (JSVM)

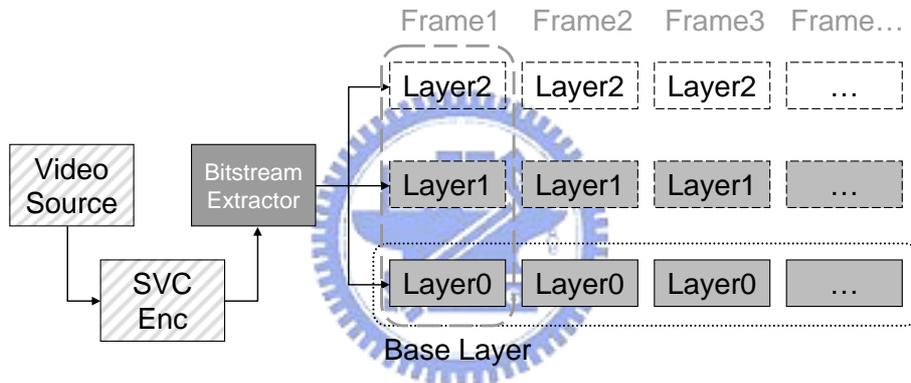


圖 2-1 H.264 /AVC SVC 可調式視訊之壓縮模式

圖 2-1 所示為 H.264/AVC-SVC 可調式視訊編碼之原始壓縮模式，乃是根據各影片之分層視訊串流預先設定好之分層數量與屬性進行壓縮，並包裝成 NAL Units 成為視訊串流。以一個隨選視訊的系統而言，只需要將這些 NAL Units 配合 RTP / UDP / IP 的協定包裝成網路封包，再將這些封包傳遞給客戶端的解碼器。

在此原始壓縮模式中，由於 JVT JSVM 之限制，此視訊串流之可調性組合從頭至尾需為一致，而無法達成隨著播放的進行動態改變視訊影片可調性組合之要求。雖然在壓縮出初步的視訊壓縮檔後，JSVM 另外包含了 Bitstream Extractor 工具，可以擷取出較少的可調性組合，但仍舊受到只能對檔案整體作處理，以及可調性組合從頭至尾需為一致的限制。

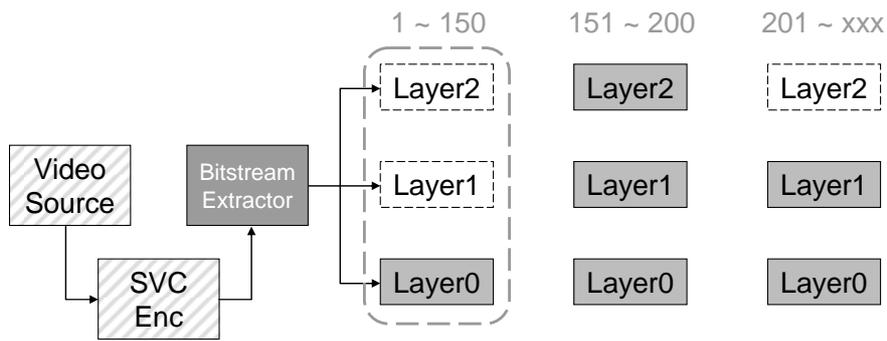


圖 2-2 隨可調性組合調整分層視訊串流數量

圖 2-2 說明可依據網路的情況,動態調整各張影像所傳送的視訊串流分層數量之示意圖。此例子說明在影片第 1 ~ 150 秒裡都只傳送單一的分層視訊串流;在 151 ~ 200 秒因網路狀況改善或系統自行調整,可以同時傳送三層視訊串流,增加影音服務品質;在第 200 秒後網路頻寬可能又開始轉壞,於是系統主動降低傳送的資料量。本文所發展之動態提取技術,便是要將各分段分離成可獨立解壓縮的單位,符合目前 H.264 / AVC-SVC 所規定之標準語法。

若要正確的分離出 H.264/AVC-SVC 之可調式視訊編碼的各分層,必須對於封裝 H.264/AVC-SVC 視訊串流之 NAL Units 加以分析,才能夠將各種情況下所需要的分層視訊串流可調性組合分離出來。JVT 所訂定之 H.264/AVC-SVC 視訊串流標準的相關背景知識,將於下一節中配合規劃的研究思維加以介紹。

2.2. Bitstream Information of H.264/AVC Scalable Extension

在建構上述系統架構時,我們參照正在發展中的 H.264 / AVC-SVC JSVM Software 作為實驗的輔助工具,藉以產生包含時間、空間、訊噪比等各種可調性的分層視訊串流。

圖 2-3 為用來封裝 H.264/AVC SVC 視訊串流之 NAL unit 結構,如圖所示可知若該 NAL unit payload 所裝填之資訊為底層之外的可調式視訊分層,則可從該 NAL unit header 得知其可調性組合,例如 dependency id, temporal level, quality level 等。因此本系統可以根據此資訊,動態提取出適合的可調性組合分層視訊串流。

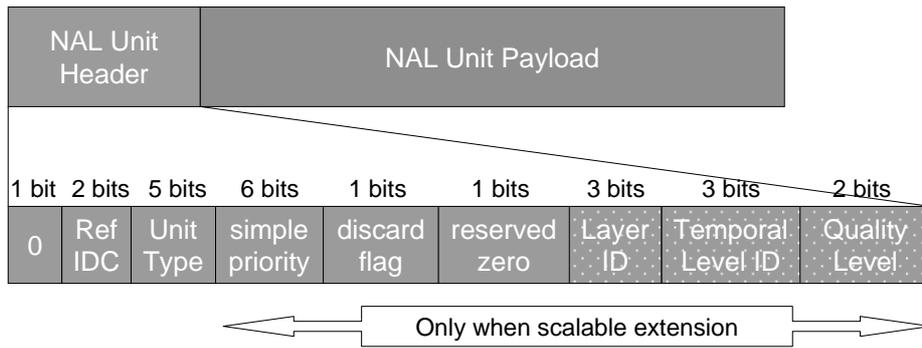


圖 2-3 H264/AVC SVC bitstream NAL unit

圖 2-4 進一步說明由 NAL unit 所包裝之 H.264 / AVC SVC 視訊位元流範例，此圖省略開頭之 SEI Header。圖中所標示之 00 00 00 01 即為各 NAL unit 的 start code。我們可以由第一個 byte 後五個 bits 的 NAL Unit Type 得知其 Payload 所裝填之資訊為 SEI, SPS, PPS, 或是各可調性分層視訊串流, 例如若 NAL Unit Type 為 7 即為 sequence parameter set ; 若 NAL Unit Type 為 5 即為 coded slice IDR picture , 詳情請參照 [6] Table 7-1 NAL Unit Type codes。

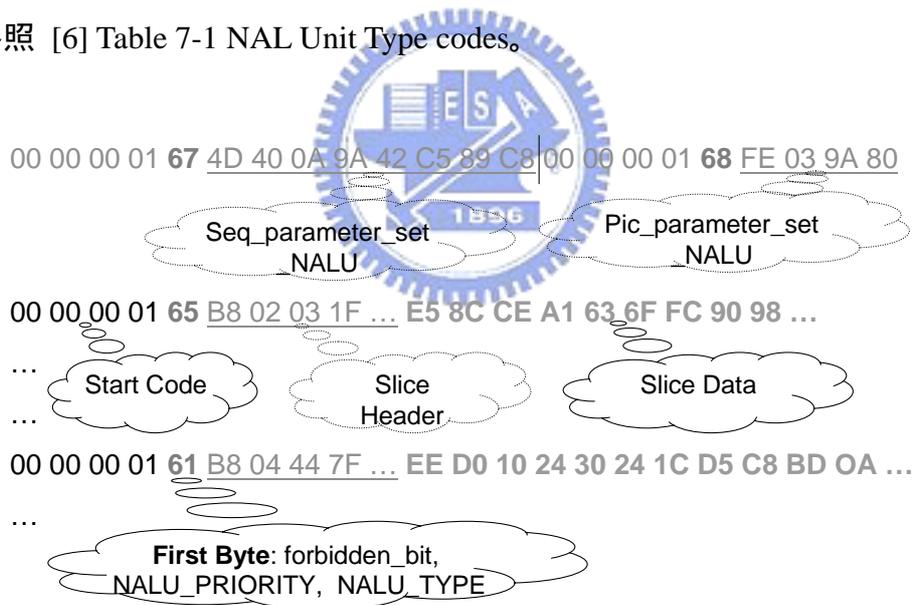


圖 2-4 H264/AVC SVC bitstream binary view

圖 2-5 則用來說明 H264/AVC SVC 視訊串流的內容組成示意圖，其開頭即為記錄各類 scalability 相關資訊的 SEI Stream Header (SCALABLE_SEI)，其所包含的 syntax elements 資訊請見 [6] Annex F；另外各主要 key frame NAL unit 之前，都會有一個記錄 temporal 資訊的 SEI SUB_SEQ_INFO。各類 SEI 的資訊並不會影響解碼

器的解壓縮過程，只是單純用來分析視訊串流的輔助工具，所以只有在不重新轉碼 (transcoding) 而要分離出底層之可調式視訊串流時，才需要分析其資訊。

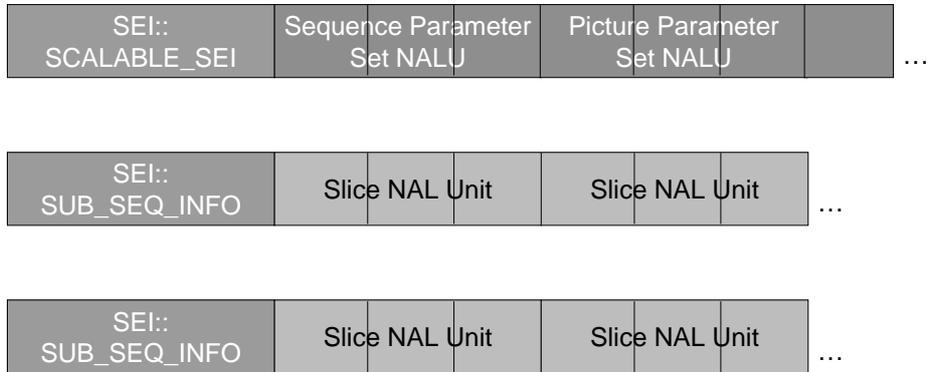


圖 2-5 H264/AVC SVC 視訊串流分析

圖中所示之 Sequence Parameter Sets 及 Picture Parameter Sets，以具有三層 spatial or CGS layers 之視訊串流為例，會各自對應於該 NAL unit 的 dependency id (spatial or CGS layer id)，其資訊會影響到後面各張影片可否正確被解壓縮；各影像之播放順序，也是由此處之資訊配合 slice header 的資訊所產生，和其壓縮順序並不相同，詳情見圖 2-6。接續於 Parameter Sets 其後之 Slice NAL Unit 的內容又各自包含有 slice header 及 slice data 兩部份，各可調性視訊分層在解碼過程中需要的參數大部份是記錄在此處。

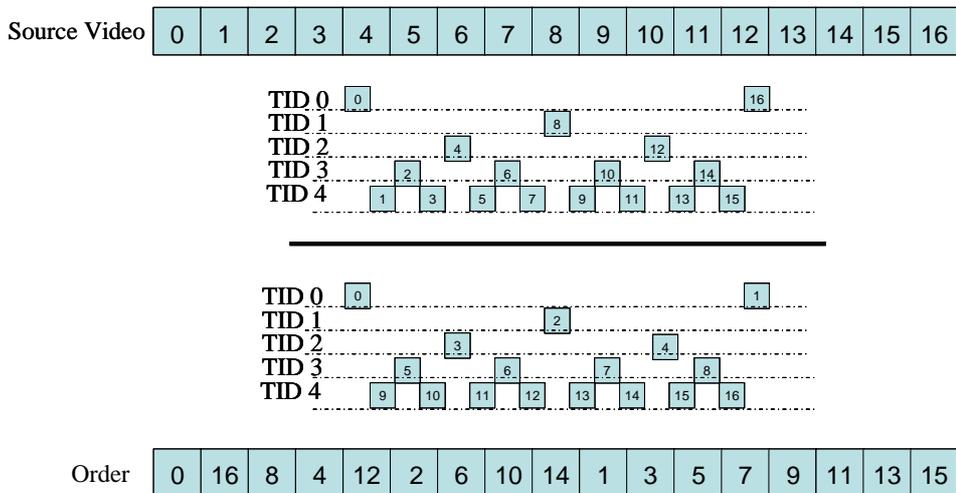


圖 2-6 原始視訊播放順序及壓縮順序之對應

2.3. Scalability of H.264/AVC Scalable Video Coding

H.264/AVC Scalable Extension 是由 ISO/IEC MPEG 和 ITU-T VCEG 組成之 JVT (Joint Video Team) 所發展的新一代分層視訊串流，其底層視訊串流相容於目前壓縮效率即高的 H.264/AVC。在即將於 2007 年 1 月發佈的 Final Draft International Standard 中，H.264/AVC Scalable Extension 包含有 Spatial Scalability, Temporal Scalability, SNR scalability 等可調性，以下將逐一說明。

2.3.1. Spatial Scalability

近似於 MPEG-2/4 的觀念，Spatial Scalability 藉由拆解原有影片成為 spatial pyramid 結構所組成，各個 spatial layer 的 motion, temporal prediction 只和相同 spatial layer 的影片相關，因此可以獨立壓縮，不受其它 spatial layer 影片的影響。

2.3.2. Temporal Scalability

Temporal Scalability 基本上是由 hierarchical B-pictures 這項功能所提供，進行 closed-loop 壓縮，使得屬於較底層視訊串流的影片可以先被壓縮，如此才能提供較高層的視訊串流影片壓縮時的參照，並藉以取代原有之 open-loop MCTF。

2.3.3. SNR Scalability

SNR Scalability 包含有 Coarse Grain Scalability (CGS), Fine Grain Scalability (FGS) 兩種，其中 CGS 提供底層的 transform coefficients 壓縮，motion information 及 temporal prediction 都必須單獨處理，無法和其它 CGS layer 共用資訊；FGS 則是用來配合 Bit Rate 的控制，可以在任一時間點被捨棄或僅保留部份，而此種功能是因為 FGS 壓縮時分別經過 significant pass 及 refinement pass，使其 transform coefficients 可以 embedded bit stream 的型態組成，因此可以在任一時間點作刪節。

3. Dynamic Layer Extraction

3.1. Defect of Joint Scalable Video Model

若遵循現階段 JVT 制訂之 JSVM 結構，一旦更動了某張影片的可調性組合，會因為其與其它影片所擁有之可調性組合不同，而可能不將該影片輸出；或是找不到對應之可調性視訊串流影像，而造成完全無法解壓縮等幾種狀況。

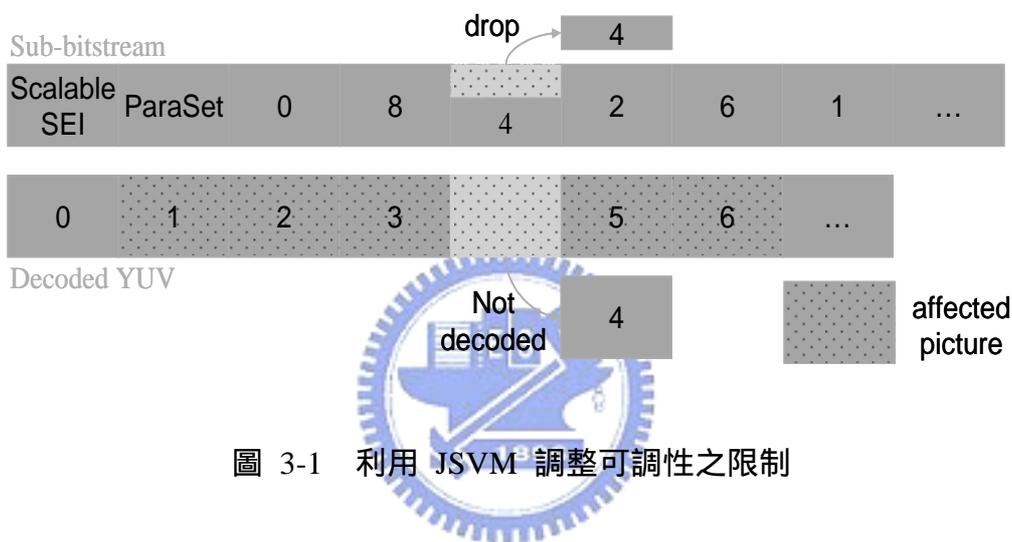


圖 3-1 利用 JSVM 調整可調性之限制

圖 3-1 說明一個標準的 H.264/AVC-SVC 視訊串流 NAL units 簡化圖，封裝各張影像可調性組合之 NAL units，在此圖中以相同標記的區塊來代表。一個標準的 H.264/AVC-SVC 視訊位元流，開頭有記錄各類 scalability SEI 資訊的 SEI Stream Header。這些資訊僅作為提供分析可調性組合之用，不會影響到解壓縮之過程。

圖中所示 Parameter Sets (ParaSet) 的部份，緊接於 SEI Stream Header 之後，可分為 Sequence Parameter Sets (SPS) 及 Picture Parameter Sets (PPS) 兩種，其數量約略對應於 spatial or CGS (coarse grain scalability) 可調性的層數。在擷取出較少的可調性組合時，僅需提取與 spatial or CGS layer 相對應的 parameter sets，而無需更動其內容，關於 SPS 及 PPS 各欄位的介紹請見 [6] chapter 7。

在壓縮或解壓縮 H.264/AVC-SVC 各分層視訊串流的過程中，需要參考對應於該分層視訊串流 spatial or CGS layer 的 Parameter Sets 資訊，但因非本文重點在此不細加闡述。在圖中 Parameter Sets 後所標示之區塊號碼代表播放時的影像次序號碼，一

個區塊代表了所有由該張影片所產生之 NAL units，每一個 NAL units 都具有特定之可調性組合。如圖 3-1 所示，若僅去除第四張影像之較高層可調性視訊位元分層，使其與相鄰影像之組成不同時，經過解壓縮後則此第四張影像會因為和此獨立單元的另一張影像可調性組成分層不同，被 JSVM 解壓器判別為解壓縮不完全而直接捨棄。

依據實驗結果，一個 H.264/AVC-SVC 編碼的視訊無法被 JSVM 正常解壓縮的狀況包含下列二種：1) 當該張影像的 layer id 發生中斷的時候，亦即當該張影像失去中間分層時，如圖 3-2，JSVM 因無法找到正確的參照影像而無法解壓縮，並因此導致解壓縮程序中斷，2) 當該張影像由最高分層開始，連續失去其下之數個分層時，亦即當該張影片的可調性組成少於第一張 IDR picture (instantaneous decoding refresh)，如圖 3-3，JSVM 將其判別為解壓縮不完全而直接捨棄該張影片，轉而處理後續的 NAL units。我們為了解決在前面提到 JSVM 的兩個問題，對於每張影像，必須在開始對 NAL unit 做解壓縮之前，檢查該 NAL unit 和前一個 NAL unit 的 layer id 是否連續。若不為連續，則表示該張影像只能以解壓縮到上一個 NAL unit 為止的圖片被輸出，即使後續收到同一張影像的其他 NAL unit 也不予解壓縮。

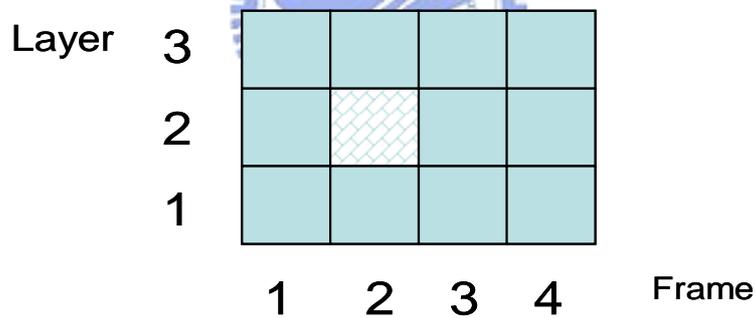


圖 3-2 影格 2 之 layer id 不連續

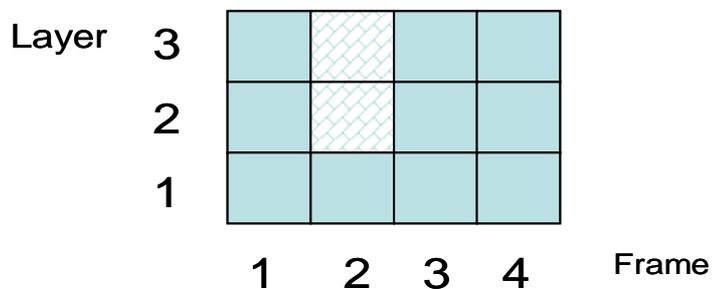


圖 3-3 影格 2 失去 Layer 2 和 3 的 NAL units

3.2. Syntax Elements

在動態調整 H.264/AVC SVC 的 Spatial, Temporal, SNR Scalability 時，我們需要判別目前的 NAL Unit 為 Sequence, Picture Parameter Sets, Coded Slice，或是僅提供輔助資訊的 Supplemental Enhancement Information 來作對應的處理，NAL Unit Header 就記錄了這些資訊，詳情請見 3.2.1 節；又為了動態刪節 Spatial, Temporal, SNR Scalability 等可調性，並考量到 hierarchical B-pictures 的特點，我們以 GOP 為區塊單位作處理的方法，可以省去過於複雜的邏輯判斷，因此需要由 Parameter Sets 及 Slice Header 所提供的資訊，計算出各 Coded Slice 屬於那一個 GOP 的那一張 Frame，相關的 Syntax Element 詳述於 3.2.2 節、3.2.3 節。

3.2.1. Syntax Elements of NAL Unit Header

nal_unit(NumBytesInNALunit) {	C	Descriptor
forbidden_zero_bit	All	f(1)
nal_ref_idc	All	u(2)
nal_unit_type	All	u(5)
nalUnitHeaderBytes = 1		
if(nal_unit_type == 20 nal_unit_type == 21) {		
nal_unit_header_svc_extension() /* specified in Annex F */		
}		
...		

表 3-1 NAL unit syntax

如表 3-1 所示為一般 NAL Unit Header 的部份內容，其中 *forbidden_zero_bit* 用來檢查 byte alignment，或是否要解碼器解壓縮；*nal_ref_idc* 在該 NAL Unit 為 non-reference 時會被設為零，若為 *nal_unit_type* = 5 (IDR), 7 (sequence parameter set), 8 (picture parameter set) 等，或是為 reference 特性者，則被設定為非零值 (2 bits)；*nal_unit_type* 則代表此 NAL Unit Payload 所封裝之型別，詳情見 [6] chapter 7。

nal_unit_header_svc_extension() {	C	Descriptor
simple_priority_id	All	u(6)
discardable_flag	All	u(1)
reserved_zero_bit	All	u(1)
temporal_level	All	u(3)
dependency_id	All	u(3)
quality_level	All	u(2)
nalUnitHeaderBytes += 2		
}		

表 3-2 NAL unit header SVC extension syntax

如表 3-2 所示, *simple_priority_id* 共有 6 bits, 用來存放 SPS 中記錄的 Spatial, Temporal, SNR Scalability; *discardable_flag* 可用來增進程式速度, 確認此 NAL Unit 是否還會被 enhancement layer 使用到, 若不再需要就先釋放記憶體空間; *reserved_zero_bit* (extension_flag) 目前為保留字元; *temporal_level* 即為 temporal scalability, 用於 reference lists 的解壓縮過程; *dependency_id* 即為 Spatial scalability, 用來指明 spatial or CGS layer 的代碼; *quality_level* 即為 SNR scalability, 每個 *quality_level* 代表一個 FGS layer, 會對應於所屬的 CGS layer *dependency_id*。

3.2.2. Syntax Elements of Sequence Parameter Sets

僅節錄比較重要的 Sequence Parameter Sets 的 syntax elements:

- (1) *seq_parameter_set_id* ($ue(v)$): 目前 Sequence Parameter Sets 的編號。
- (2) *pic_order_cnt_type* ($ue(v)$): 決定 Picture Order 怎麼計算。
- (3) *log2_max_frame_num_minus4* ($ue(v)$): Temporal frame_num 的最大容量。
- (4) *log2_max_pic_order_cnt_lsb_minus4* ($ue(v)$): Picture order 的最大容量。

3.2.3. Syntax Elements of Picture Parameter Sets

僅節錄比較重要的 Picture Parameter Sets 的 syntax elements:

- (1) *pic_parameter_set_id* ($ue(v)$): 目前 Picture Parameter Sets 的編號。
- (2) *seq_parameter_set_id* ($ue(v)$): 此 PPS 對應於那一個 SPS。

3.2.4. Syntax Elements of Slice Header

僅節錄比較重要的 Slice Header 的 syntax elements :

- (1) pic_parameter_set_id ($ue(v)$) : 此 Slice 對應於那一個 PPS。
- (2) frame_num ($u(v)$) : 記錄連續 access units 的 reference fields。
- (3) idr_pic_id ($ue(v)$) : 目前壓縮到第幾個 IDR picture。
- (4) long_term_reference_flag ($u(1)$) : 記錄 reference pictures 的 long term reference。
- (5) pic_order_cnt_lsb ($u(v)$) : 此 Slice 的 picture order, 即一般認為之 frame num。
- (6) delta_pic_order_cnt ($se(v)$) : 記錄 field coding 的 picture order count difference。

3.3. Layer Extraction Algorithm

在經過本文第二章詳加探討 H.264/AVC SVC 標準及其結構組成, 並說明各影像各分層之資訊可由何處取得後, 我們可以根據這些資訊, 隨著需求動態改變各張影像播放時的可調性組合。在詳細說明本文所提之視訊串流分層提取技術演算法, 使本系統擷取各分層視訊串流的流程, 能夠符合邊接收邊播放的隨選視訊要求之前, 先來回顧一下使用 H.264/AVC JSVM (reference software) Bitstream Extractor 的一些限制。

由 JSVM 提供的提取工具 Bitstream Extractor 僅能對檔案操作, 而且必須處理完整的檔案後才可進行檔案的其他使用; 該工具也必須限制其視訊串流之可調性組合從頭至尾需為一致。由實驗結果得知, 若恣意更動任何一張影像之可調性組合, 會因為其組合多於或少於第一張影像之可調性組合, 造成系統因無法找到正確的參照影片而無法解壓縮, 或將其判別為解壓縮不完全而直接捨棄。這種僅能對整體檔案操作, 而且必須處理完整個檔案才可開始傳送資料, 並不符合邊接收邊播放的隨選視訊要求; 無法恣意更動任何一張影片之可調性組合, 並無法滿足動態調整 H.264/AVC SVC 視訊串流分層組合之要求。

3.3.1. Intra Key Picture and IDR Picture Converison

為了能符合 H.264/AVC SVC Decoder 的標準, 第一張 slice 需為 IDR Picture 的要求, 首先將 Sequence Parameter Sets, Picture Parameter Sets 的資訊都存起來, 待接收到要重新分段的訊息時, 也要將下一個分段開頭的 intra key picture 更改為 IDR picture 後, 依據下一分段可調性組成的設定先備份起來。以下為 sheudo code :

```

if ( eNalUnitType == NAL_UNIT_SPS ) {
    pcBinDataSPS[cPacketDescription.SPSid].operator=( *pcBinData);
    maxSPSid = cPacketDescription.SPSid;
}

if ( eNalUnitType == NAL_UNIT_PPS ) {
    pcBinDataPPS[cPacketDescription.PPSid].operator=( *pcBinData);
    maxPPSid = cPacketDescription.PPSid;
}

if (BackupIDR && uiLayer <= uiUserLayer && uiFGSLayer <= uiUserFGSLayer
    && uiTempLevel <= uiUserTempLevel) { // 依據下一分段的可調性組合
    cPacketDescription.BackupIDR = true; // Flag (將 intra picture 轉型成 IDR picture)
    RNOK( m_pCH264AVCPacketAnalyzer->process( pcBinData, cPacketDescription,
        pcScalableSEIMessage ) );
    pcBinDataSlice[maxSliceId++].operator=( *pcBinData);
}

```

接著當伺服器端，將目前段落的全部資訊傳送出去後，依據下一分段的可調性組合，對網路傳送之前儲存的 Sequence Parameter Sets, Picture Parameter Sets；以及之前已將 intra key picture 轉型成 IDR picture，並已經下一分段的可調性組合篩選過之 Coded Slice 傳送給使用端。以下為產生新分段的 sheudo code：

```

if ( bNewSegment ) { // 如果要產生新分段
    newSegFrameNum = ( (m_iPoc/GOPSize)+2) * GOPSize; // 算出下一分段第一張
    bNewSegment = false; // 關閉產生新分段的邏輯判斷
    maxSliceId = 0; // 將 Coded Slice 暫存區清空
    uiUserLayer, uiUserFGSLayer, uiUserTempLevel // 設定下一分段的可調性組合
}

if ( newSegFrameNum && m_iPoc == newSegFrameNum ) // 備份下一分段第一張
    BackupIDR = true;

```

```

If ( newSegFrameNum &&.m_iPoc == (newSegFrameNum+userGOPSize)
    && cPacketDescription.Layer == 0 ) { 若為下一分段的第二張，插入備份資料
    if( bRequired ) printf("write SPS BinData:%d\n", index);
    if( bRequired ) printf("write PPS BinData:%d\n", index);
    printf("write Slice BinData:%d\n", index);
}

```

當 Decoder 接收到第一個 Sequence Parameter Set 資料時，便會進行重新啟動 Decoder 的動作，使各獨立分段可以獨立解壓縮。以下為 pseudo code：

```

Bool bNewSegment = false; // 是否收到新分段 ?!
Bool bCompleteSeg = false; // 是否完成新分段 ?!

if ( eNalUnitType == NAL_UNIT_SPS && ! bCompleteSeg )
{ // 發現第一個 SPS，產生新分段 (同時用 bCompleteSeg 略過高層的 SPS)
    bNewSegment = true;
    bCompleteSeg = true;
    ...
}

if ( bNewSegment ) { // 接收到新分段
    bNewSegment = false; // 關閉邏輯
    if (oldDecoder) delete oldDecoder; // 關閉舊 decoder
    new newDecoder; // 產生新的 decoder
    ...
}

if ( eNalUnitType == NAL_UNIT_CODED_SLICE_IDR ) {
    bCompleteSeg = false; // 接收到 SPS，又接收到 IDR 後，產生新分段
    ... // 同時準備好下一次需要重新分段時的 flag
}

```



3.3.2. Video Layer Extraction

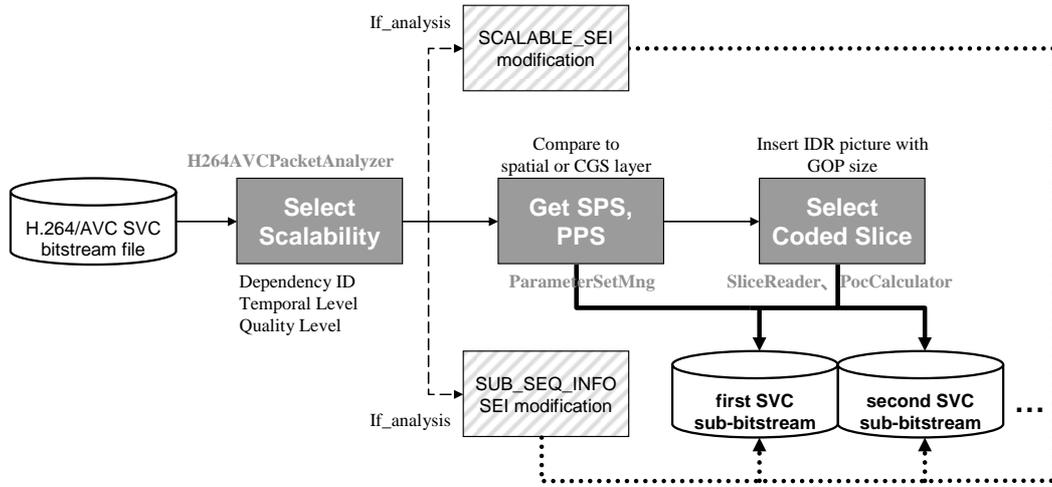


圖 3-4 分層視訊串流提取技術演算法流程

圖 3-4 所示即為本計畫之分層視訊串流提取技術演算法流程,其判定方法及組成各分段隨選視訊串流之過程,將詳加說明於後。圖 3-5 則從 bitstream 的角度,由 NAL Unit Header, Sequence Parameter Set, Picture Parameter Sets, Slice Header 的相關 syntax elements 來說明此提取技術演算法相關的訊號控制及流向。

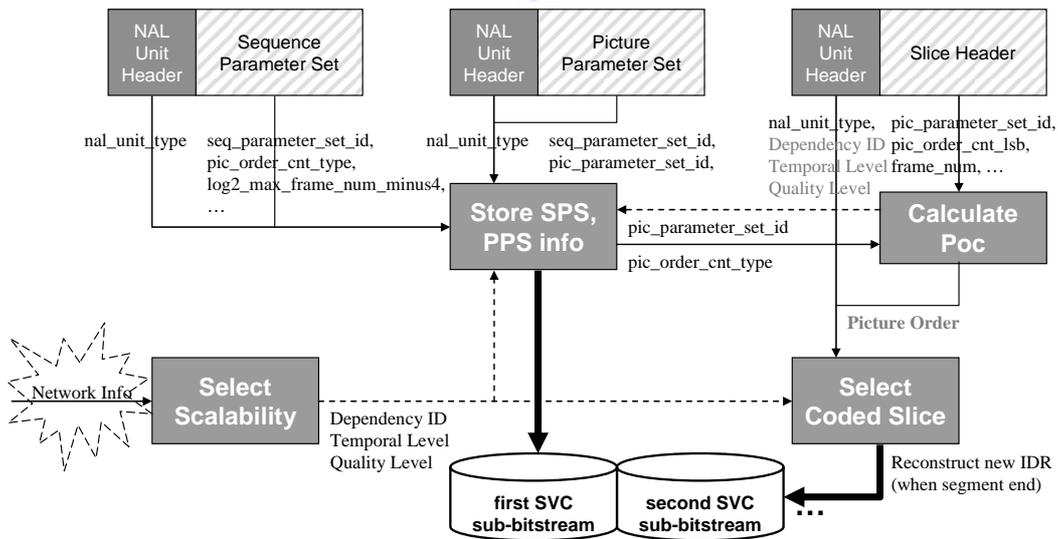


圖 3-5 分層視訊串流提取技術內部訊號圖

1) 決定可調性組合：

在壓縮出初步的視訊影片後，首先決定要擷取的可調性組合，包含空間、時間、訊噪比等；此部份在串流服務系統中可以配合用戶端網路頻寬偵測模組回傳之資訊，調整其可調性組合。

2) 修改 SCALABLE_SEI (stream header) :

配合先前決定之可調性組合，調整 SEI stream header 的內容，這部份僅供分析視訊串流，或進一步由此分段擷取出更少之可調性組合才會參照其資訊。

3) 擷取 SPS, PPS :

因為 sequence parameter sets 和 picture parameter sets 的資訊會影響到後面各張影片可否正確被解壓縮，所以必須擷取出和 spatial or CGS 相對應的 SPS, PPS，該分段之影片才能被獨立解壓縮。

4) 修改 SEI SUB_SEQ_INFO :

在 H.264/AVC SVC 的標準中，SEI SUB_SEQ_INFO 僅會出現在各 key frame 之前，僅供分析視訊串流，或需要由此次擷取出之分段，進一步擷取出更少之可調性組合才會參照其資訊。

5) 擷取分層視訊串流 :

依據先前決定的可調性組合，以 GOP 為單位，僅擷取出符合條件的 H.264/AVC SVC 可調性分層視訊串流 (可由該 NAL unit 之 syntax element 判別)。為避免各 GOP 單位因更動可調性組合，而造成無法解壓縮或解壓縮不完全之情況發生，視需要加入原本需參照的 IDR picture，並更改其 slice header syntax element 資訊。

在此一流程中，關於擷取視訊串流分層的部份，有幾點需要多加留意：1) 因為各影像之播放順序，和其壓縮順序並不相同，詳情見圖 3-7，在判斷各張影像何時已不再需要被其它影像參照，難以迅速切出可邊接收邊播放的獨立可解碼單元。為了符合隨選視訊之要求，本系統乃以 GOP 作為分段的單位，即可清楚明瞭的切出各可獨立解壓縮的分段。

2) 由於 JSVM 解碼器在開始解壓縮各分段影片，會優先檢查第一張影片是否為 IDR picture，因此當本系統以 GOP 為單位時，除了需設定 intra period 的間隔和 GOP 的大小相同外，還必須在該分段的第一張影片前安置原本需參照的前一分段最後一張影像 (I picture)，並調整該影片之 slice header，除去不屬於 IDR picture 的 syntax

elements , 並填入如 `idr_pic_id`, `no_output_of_prior_pics_flag`, `long_term_reference_flag` 等 , 將該參照的 I picture 轉型成為 IDR picture。

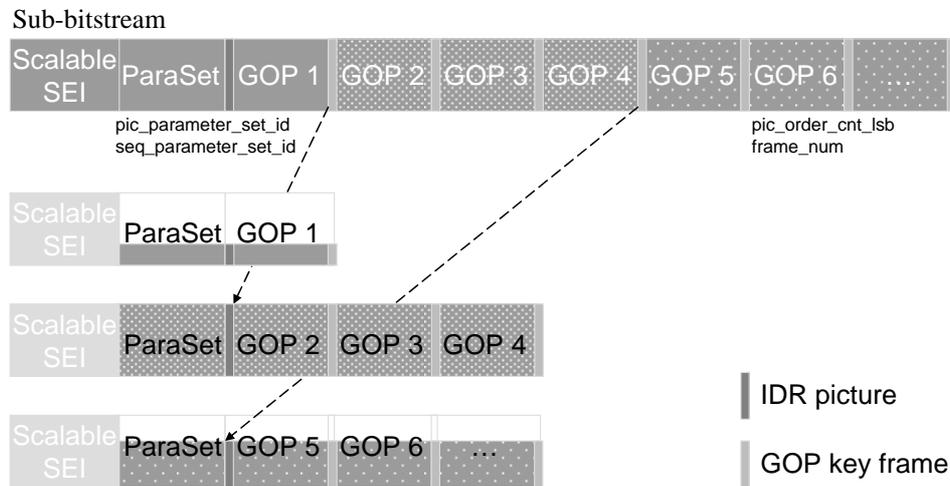


圖 3-6 分層視訊串流提取技術演算法實例說明

圖 3-6 舉例說明此分層視訊串流提取技術演算法, 如圖所示最上面之視訊串流為由 encoder 壓縮而得, 在不需要重新轉碼的情況下, 本系統在切割出隨選視訊各分段時, 將原有之 key frame 替換為 IDR picture, 使 decoder 端能順利解壓縮具有不同可調性組成的隨選視訊分段;

另外由於 SEI 資訊僅作為輔助分析之用, 因此在網路上傳送視訊資料時, 可選擇不傳送 SEI 的資訊, 在圖中以淡灰色表示。本演算法之主要程式模組如下所述, 包含有 Extractor、H264AVCPacketAnalyzer、ParameterSetMng、SliceReader、PocCalculator 等。

模組名稱	Extractor
模組功能	圖 3-4 即為結合 SliceReader、PocCalculator、ParameterSetMng、H264AVCPacketAnalyzer 等模組之功能, 分析 H.264/AVC-SVC 視訊串流之內容。由 NAL unit header 之 <code>nal_unit_type</code> 得知該 NAL unit payload 所包含之資料為 parameter sets、coded slices 或只是輔助用的 SEI 資訊; 由 PocCalculator 配合 slice header 中之 <code>pic_order_cnt_lsb</code> 及 <code>frame_num</code> 計算出 coded slices 的影片

	<p>播放順序；可由該 coded slices 之 NAL unit extension 得知其可調性組成，並依此擷取出其所對應的 parameter sets 資訊等。此一分層提取技術演算法模組，未來可配合接收端之使用者資訊，使其在網路上傳輸時，能夠依據當時網路頻寬狀況動態分離出所需要的影片品質，Extractor 所包含的相關模組功能將逐項說明之。</p>
--	--

模組名稱	H264AVCPacketAnalyzer
模組功能	<p>利用 nal unit header 之 nal_unit_type 資訊，對於 SPS, PPS, SEI, coded slice 等不同型能的 NAL unit 資料作相對應之分析與資料維護；當處理到 coded slice 時利用 SliceReader 模組讀出該 coded slice 的 slice header 資訊，例如其屬於那一個 GOP 分段等，提供 Extractor 模組進行隨選視訊影片分段時，所需要的重要訊息。</p>
模組名稱	ParameterSetMng
模組功能	<p>記錄並維護 sequence parameter sets 及 picture parameter sets 之資訊，其中 picture parameter sets 的 syntax element 有記錄其 pic_parameter_set_id 與 seq_parameter_set_id 的對應關係，執行隨選視訊影片分段時，所需要的 SPS, PPS 資訊可以由此擷取，其中決定 coded slice 播放順序計算方法的 pic_order_cnt_type 也是 SPS 的資訊之一。</p>

模組名稱	SliceReader
模組功能	<p>分析 coded slice nal unit，將其 nal unit payload 中屬於 slice header 的資訊讀出，並儲存於 slice header 資料結構當中；其中 slice_type 可知道其是否為 FGS frame, I frame 等；可依據 pic_parameter_set_id 找到相對應之 picture parameter set 資訊；若為 IDR picture，則會有獨一無二的 idr_pic_id；要計算該 coded slice 之播放順序 picture order 時則需要 frame_num、pic_order_cnt_lsb 等資訊，其它重要資訊如下：</p>

	<p>first_mb_in_slice、</p> <p>slice_type、</p> <p>pic_parameter_set_id、</p> <p>frame_num、</p> <p>idr_pic_id、</p> <p>pic_order_cnt_lsb、</p> <p>delta_pic_order_cnt、</p> <p>...</p>
--	---

模組名稱	PocCalculator
模組功能	<p>用於計算各張影片之播放順序，需配合下列參數，並依據 pic_order_cnt_type，而主要分為三種運算模式，再配合其配合其 reference frame 的選擇得出，詳情請見 [6] 8.2.1 decoding process for picture order count.</p> <p>(1) sequence parameter sets 中的 syntax elements</p> <p>pic_order_cnt_type、</p> <p>offset_for_ref_frame、</p> <p>offset_for_non_ref_pic、</p> <p>log2_max_frame_num_minus4、</p> <p>log2_max_pic_order_cnt_lsb_minus4、</p> <p>num_ref_frames_in_pic_order_cnt_cycle</p> <p>(2) slice header 中的 syntax elements</p> <p>pic_parameter_set_id、</p> <p>pic_order_cnt_lsb、</p> <p>delta_pic_order_cnt、</p> <p>frame_num</p> <p>(3) nal unit header 中的 syntax elements</p> <p>nal_ref_idc、</p> <p>nal_unit_type</p>

4. Experimental Results

4.1. Dynamic Adaptation of H.264/AVC Scalable Video Coding

以下簡述實驗結果，本實驗的條件為設定三層 spatial/CGS layers，亦即會有三層的 Dependency IDs。如前所述將原始壓縮影片的 GOP size 及 intra period 設定為相同值 16，以利於需要擷取較少可調性組合分段時，可在不用重新轉碼的情況下，生成各分段第一張不需參照其它影片的 IDR picture，使本系統能夠順利分離出各段獨立可解壓縮影片，並可進一步觀察其可調性分層組合與視訊畫面品質。

此三層 spatial/CGS layer 的解析度為 176x144, 176x144, 352x288；所設定的 frame rate 依序為 15, 30, 30；QP 則分別為 40, 36, 30，其中第二層的 spatial/CGS layer 具有額外兩層的 FGS layers，以求實驗結果的完整性。我們使用依此設定之 SVC 壓縮 Stefan sequence 200 個 frames，利用本系統具有之邊擷取串流分段邊輸出之隨選視訊特性，將 0~80 frame 切成第一分段，請見圖 4-3；同時將第 80~196 frame 切給第二段，請見圖 4-4，其中第二段之第一張影像即為根據 80th frame 所產生之 IDR picture。如此才能達成各分段可以自行調整可調性，而且都可供獨立解壓縮的目標。將來才能應用到網路上，依據使用者的頻寬狀況，動態調整各張影片所需傳送的分層視訊串流數量。



圖 4-3 stefan_cif.264 Frame #19 (第一分段)

調整第一分段 (包含 0~80 frame) 之 spatial or CGS layer 為僅有 base layer 及一層 enhancement layer。由於第一層的 enhancement layer 具有額外兩層 FGS layer 視訊串流的增強，由圖 4-3 實驗結果可知其畫面仍舊非常接近原有視訊品質。



圖 4-4 stefan_cif.264 Frame #12 (第二分段)

調整第二分段 (包含 80~196 frame) 之 spatial or CGS layer 僅留下 base layer，由於底層所使用之 QP 較大，其畫面品質自然較原始畫面品質為差。

5. Future Work

依據實驗結果得知，若遵循現階段 JVT 所訂定之 SVC bitstream 結構以及現階段之 JSVM，一旦更動了某張影像的可調性組合，會因為其與第一張影像所擁有之可調性組合不同，而可能產生解壓縮不完全，或是找不到對應的可調性視訊串流影片，而造成完全無法解壓縮等幾種情況，詳情及相關解決方法請見本文第三章的探討。

在本文裡，我們修改 H.264/AVC SVC JSVM 架構，賦予系統可隨時依據目前可用頻寬，動態調整視訊串流傳輸數量的能力後，接下來的目標便是使其能夠透過網路環境模擬各種可能的狀況，並探討衍生出的相關問題，例如在有限的通道邊碼保護下，於不可靠的網路環境中傳輸大量的可靠多媒體資料以及提高系統的效能及穩定度，其中包含有無線網路環境訊號偵測模型、適應性前向糾錯編碼分層視訊串流、整合系統與展示等，均是我們感興趣的研究方向，



6. REFERENCES

- [1] ISO/IEC JTC 1/SC 29/WG 11/N6359, Advanced Video Coding (second edition), ISO/IEC 14496-10:2004, ISO/IEC, Mar. 2004.
- [2] ISO/IEC JTC 1/SC 29/WG 11/N6360, *Advanced Video Coding Amendment 1 Fidelity Range Extensions*, ISO/IEC 14496-10:2003/FPDAM 1, ISO/IEC, Mar. 2004.
- [3] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264 / AVC Video Coding Standard," *IEEE Trans. on CSVT*, 13(7), pp. 560- 576, July 2003.
- [4] ISO/IEC JTC 1/SC 29/WG 11/N6540, Advanced Video Coding (pre-third edition), ISO/IEC 14496-10:2004, ISO/IEC, July. 2004.
- [5] "Independent parsing of spatial and CGS layers," H. Schwarz, D. Marpe, and T. Wiegand, Doc. JVT-S069, April 2006.
- [6] "Scalable Video Coding Standard Joint Draft 6," T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien, Doc. JVT-S201, April 2006.
- [7] "Subband Extension of H.264/AVC," H. Schwarz, D. Marpe, and T. Wiegand, Joint Video Team of ISO/MPEG and ITU-T VCEG, March 2004.
- [8] "Combined Scability Support for the Scalable Extension of H.264/AVC," H. Schwarz, D. Marpe, and T. Wiegand, IEEE International Conference on Multimedia and Expo (ICME'05), July 2005.

