

行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※

※

※ 無線傳輸之訊源及通道編碼 (3/3) ※

※ Source and Channel Coding for ※

※ Wireless Transmission (3/3) ※

※

※※※※※※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：個別型計畫 整合型計畫

計畫編號：NSC 90-2219-E009-003

執行期間：90年8月1日至91年7月31日

計畫主持人：杭學鳴

計畫參與人員：林敬雄，蔡家揚，蔡尚軒，陳繼大，李岳凌

本成果報告包括以下應繳交之附件：

- 赴國外出差或研習心得報告一份
- 赴大陸地區出差或研習心得報告一份
- 出席國際學術會議心得報告及發表之論文各一份
- 國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學電子工程學系

中華民國 91年 7月 31日

行政院國家科學委員會專題研究計畫成果報告

無線傳輸之訊源及通道編碼 (3/3)

Source and Channel Coding for Wireless Transmission (3/3)

計畫編號: NSC 90-2219-E009-003

執行期限: 90 年 8 月 1 日至 91 年 7 月 31 日

主持人: 杭學鳴 國立交通大學電子工程學系教授

計畫參與人員: 林敬雄, 蔡家揚, 蔡尚軒, 陳繼大, 李岳凌

國立交通大學電子研究所

中文摘要

我們的目標是發展與實作適合於無線通訊所使用的訊源及通道編碼。第三年達成的三項課題為：(1)「適應性多重位元率」語音編碼之深入研究，(2) H.263+視訊編解碼器之改進，與(3)通道編碼之研究、模擬，並以數位訊號處理器實現。並且將此三項與其他子計畫連結整合，建立一個在 DSP 上實現的完整系統。

關鍵詞：AMR, H.263+, 通道編碼

Abstract

The goal of this project is to develop and implement video and speech coding schemes that are suitable for wireless telecommunication systems. We achieve the following goals during the third year. (1) Simulate the "Adaptive Multi Rate" (AMR) speech coding standard. (2) Improve the DSP-based implementation of H.263+ video coder. (3) Simulate and implement the 3GPP convolutional codes on DSP. Furthermore, we integrate these three items together with other sub-projects to complete a real-time wireless simulation system on TI DSP chips.

Keywords: AMR, H.263, Channel Coding

第一部份 語音編碼

在第一年及第二年中，已將 G.723.1 之編解碼器於 DSP 上之實現完成，因此在第三年主要是針對適應性多重位元率再做深入之研究。

A.背景

歐洲通信標準委員會在 1997 年開始發展此一新一代的語音壓縮技術，稱為「適應性多重位元率」(Adaptive Multi Rate, AMR)，由「第三代無線通訊合作計畫」(3rd Generation Partnership Project) 所接受。其特色之一就是將傳輸通道的狀況考慮在內。可以用在視訊電話、視訊會議、無線多媒體傳輸等應用當中。

B.研究步驟

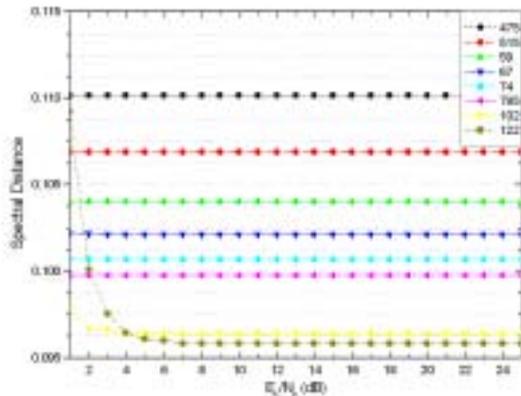
「適應性多重位元率」(AMR)的目的在保持有效率的頻譜使用的情況下，能夠達到更高的語音品質。整個系統可以針對不同的傳輸通道流量狀況，對通道模式 (channel mode) 和編碼模式 (codec mode) 做各種調整。當網路狀況不佳時，語音編碼器產生較低的位元數和較差的聲音品質，同時允許提供較多的保護碼來對抗通道失真。反之，在網路狀況較佳的時候，語音編碼器提高其所產生的位元數，而保護編碼也較不需要，這也是這個語音壓縮技術命名的由來。

從壓縮技術來看，AMR 這個語音壓縮標準是以「電碼激發線性估計編碼」(ACELP) 的技術為設計基礎，並有八種編碼率，分別介於 4.75 到 12.2 kbit/s 之間，並針對各種不同的編碼率提供不同比率的「迴旋編碼」(Convolutional Code)，因此在不同的無線網路狀況下，可以從中選擇最合適的編碼率得到最佳的品質。

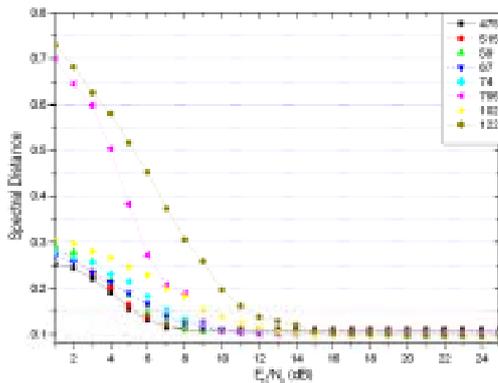
實驗中我們引進兩種通道雜訊模型，分別

是「雷利衰減通道」(Rayleigh fading channel, Gilbert model)和「加性白色高斯雜訊」(additive white Gaussian noise, AWGN)。我們引用以上兩種模型在不同噪音情況下測試上述各種模式，以比較其優劣。

C. 實驗與結果



(a)



(b)

圖 1. (a)與(b)分別為各種模式在 AWGN 及 Rayleigh fading channel 之表現，圖中標示 475、515、59、67、74、795、102、122 分別表示 4.75kbps、5.15kbps、5.9kbps、6.7kbps、7.4kbps、7.95kbps、10.2kbps、12.2kbps。

我們用原信號與還原信號之頻譜誤差 (Spectrum distance) 量測其傳輸之最終品質，此誤差為壓縮誤差與傳輸錯誤共同合成之結果。圖 1 為各種模式在加性白色高斯雜訊通道與雷利衰減通道下之頻譜誤差。我們可觀察發現，在相同的 E_b/N_0 之下，不同模式之最後結果相當接近。例外是，12.2kbps 與 7.95kbps 兩者對於通道所造成的錯誤較為敏感。

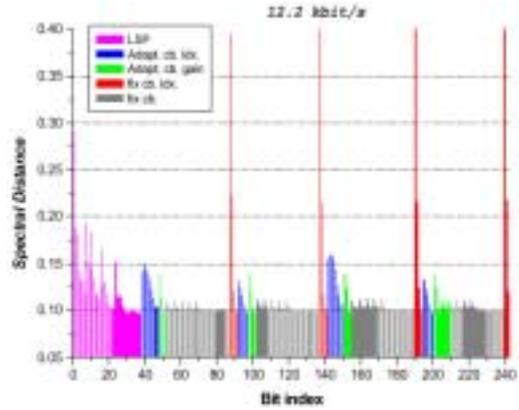


圖 2. 12.2kbps 編碼模式各位元之錯誤敏感度分析。

圖 2 是 12.2kbps 編碼模式各位元之錯誤時所造成之頻譜誤差，證實其具有特別敏感之位元。

第二部份 視訊編碼

視訊編碼之研究，在第二年已經將視訊編碼器與解碼器在 DSP 上實現，第三年主要將其 DSP 之執行效率加以提升，並且加上碼率控制(Rate control)。

A. 背景

在視訊編碼部分，我們將 ITU 所制訂的視訊標準 H.263+，利用德州儀器的 TMS320C6201SP，實現一個即時編解碼系統。在之前的成果中，我們引進了 Fast Motion Vector Search 和 DIF DCT 來提升編碼速率。在第三年，我們則是利用 Code Composer Studio 的 Compiler 特性以及 DSP 架構，並且調整調整記憶體使用之分配，使整體效能提升。另外，我們將 H.263 TMN5 所定義的碼率控制(Rate control)機制加入我們原來的編碼器中並且在 DSP 上實現。

B. 研究步驟

一. 利用 DSP 特性加速編碼效率:

在第二年我們已完成 H.263+ 編解碼器的實現，在今年由於進行所有子計畫的系統整合，會經過許多不同的界面，經過越多層的介面所造成的延遲(delay)會越嚴重，而編碼速度的考量就變得相對重要。因此我們將編碼器中的 DCT 部份再度置換為定點 DIF DCT，並且針對以下三點，對編碼器做細部的調整。

(1)利用 TI Code Composer Studio 編譯器 (compiler)所提供的內部運算子(intrinsic operator)替換程式中的運算部份,可以縮減程式記憶體(program memory)使用量,並且可以對該運算加速。

(2)為增加 DSP 對資料處理(throughput)能力,我們利用 32 bits 的一次載入(load)兩筆 16 bits 的資料,這樣可以降低 DSP 對記憶體載入與儲存(store)的次數,並且搭配內部運算子的使用,加速編碼速度。

(3)內部程式記憶體的速度較快,但是僅有 64kbytes。我們調整程式中資料宣告,將常用的程式區快使用內部記憶體,而將其他資料使用外部記憶體(external SDRAM),因此可加快整體讀取資料速度。

二. 輸出碼率控制(Rate Control):

為了維持一定的編碼輸出速率,並且考量加入碼率控制機制後,會對 DSP 造成額外的運算負擔與記憶體使用量,我們選用了一個運算量較低的方法,此方法為 H.263 TMN5 文件中定義。此方法主要利用兩種方式調整輸出碼率:

(1)利用輸出緩衝區(output buffer)判斷是否對下一張 frame 編碼,若是緩衝區內容量大於最大臨界點,則選擇不做編碼,避免輸出碼率過高。

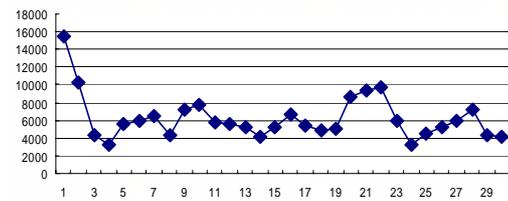
(2)在 Macroblock 層級調整量化參數(quantization parameter)。在對某一個 Macroblock 進行編碼之前,可以算出此 Macroblock 可用的預算(budge),若是利用目前的量化參數得到此 Macroblock 編碼碼率大於此預算,則調大接下來下一個 Macroblock 的量化參數,使下一個 Macroblock 的編碼碼率接進預算,反之亦然。

在經過實驗模擬之後,我們並且將此機制加入 H.263+的 DSP 實現中。

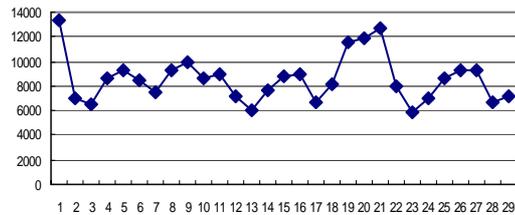
C. 實驗與結果

圖 3 為 TMN5 之實驗結果,在此圖中所用的目標碼率(target bitrate)為 53kbps,輸出緩衝區的臨界點設為 8kbps,I 畫面每十張更新一次。由於 I 畫面所需的碼率較高,因此在有碼率控制的情形下,會造成大概每十張 PSNR 會略為下降的情況。由圖中

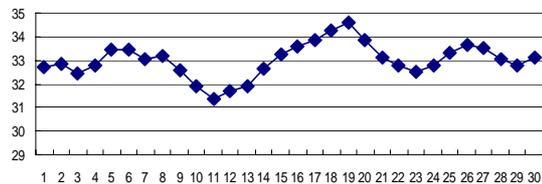
可以觀察到此現象。



(a)



(b)



(c)

■ 3. QCIF foreman sequence with TMN5 rate control (a) 每張 frame 之 bits 使用量, (b) 編碼輸出緩衝區情況, 及 (c) 每張 frame 之 PSNR.

另外,在經過最佳化處理之後的編碼器,I 畫面由原先的 67.48MIPS 加速至 2.89MIPS,P 畫面則由原先的 229.52MIPS 加速至 6.44MIPS;平均來說編碼速率約為每秒 20 張 QCIF,解碼為 26 張 QCIF。而編碼器記憶體使用量為 58kbytes,解碼器(decoder)記憶體使用量為 27kbytes。

D. 結論

在經過 DIF DCT 與 Diamond search 的快速演算法,及針對 DSP 特性對程式碼做最佳化後,最後加上碼率控制,已可在 DSP 上達成平均為 20 張 QCIF 的編碼器實現。

第三部份 通道編碼

此部份主要為通道編碼在 DSP 上的實現。

A. 背景

在 3GPP 所制訂的第三代無線通訊標準中,採用了渦輪編碼(Turbo Coding)及迴

旋編碼(Convolutional Coding) 等兩種通道編碼來保護傳送的資料。3GPP 的技術報告並指出，迴旋編碼較適合運用在即時系統上，因此我們選擇迴旋編碼來進行 DSP 即時實現。一般迴旋編碼的解碼都是利用 Viterbi 解碼方式，它的基本觀念是建立樹狀圖結構，並從其中找到最佳路徑而把原始的資料重建回來。

B. 研究步驟

一般而言，Viterbi 解碼方式必須等到全部的資料都接收之後才能開始做解碼的動作，但是如此一來就無法達成即時(Real-time)處理，因此我們採用一種稱為「削去式 Viterbi」解碼方式。當解碼器收集滿 128 筆資料之後，即決定前 64 筆資料輸出，因此最多只有 64 筆資料的時間延遲，並且可以節省記憶體的使用。

此外，我們發展了一套 ANSI C 的程式來確認整個通道編碼系統的正确性，然後將這個程式移植到德州儀器生產的 TMS320C6201 DSP 上面，而直接將此 ANSI C 之程式在 DSP 上實現無法達到即時系統的要求，因此我們針對以下幾點對此 Viterbi 解碼器做最佳化：

(1) 改變資料型態之宣告:

在 DSP 中處理 long 宣告格式為 40 bits，必須分為 32 bits 與 8bits 做處理，會降低處理速度，因此我們將程式中所有使用 long 宣告之資料改為 int 宣告，可改善運算速度。另外，常用的資料區塊則宣告為 Register 格式，可加快 DSP 接觸(access) 此資料的速度。

(2) 去除程式碼前後相關性(dependency):

在相鄰的程式碼易出現資料的相關性，當編譯器編譯(compile)時，必須解決因程式前後相關性所造成的 data hazard，造成許多不必要的運算。因此我們將相鄰程式碼間的相關性儘量排除，可得到相當大幅度的運算速度改善。

(3) 軟體管線(software pipeline)及迴圈展開(loop unrolled):

軟體管線的概念來自對迴圈的處理。在迴圈中，下一個反覆(iteration)的指令必須等到這個反覆的指令全部做完之後才會開始做，加入軟體管線的觀念之

後，每一個反覆中沒有相關的指令可以放在一起做，如此一來便可以加快迴圈的執行速度。

(4) 增加每次資料輸入量(throughput):

方法同此報告之前提過第二部份 B. 中第(2)點描述。

C. 實驗與結果

表 1 列出了程式原始版本和改良後的不同版本的效能比較，可以看出在處理速度上有極大的改善，最後一個版本的速度比原始版本快了 20 多倍，在本子計畫中，最低傳送語音及視訊資料所需之 bit-rate 約為 60Kbps，因此足以達成 real-time 之編解碼效能。

最佳化步驟	Total cycle	Code size	Proc. rate (kbit/s)
Original	43,664,891	0x1720	6.5
long type	20,124,171	0x1540	14.2
int type			
DMEM	16,454,127	0x1500	17.4
Register			
I, i/2	3,000,674	0x1720	95.5
I, j (i/2)			
Copy	2,054,134	0x19c0	139.5
Unrolled			
Char (pp)	1,863,381	0x1900	153.8
Int (pp)			

表 1. 經過程式各個最佳化步驟後，運算速度逐步改進之比較，最佳化步驟由上而下順序分別為在 B. 研究步驟所題之 4 種最佳化方法。

D. 結論

透過一些 DSP 最佳化之技巧，可使解碼執行速度由 6.5kbps 加速至 153.8kbps，總共加快 2550%，使其達到 Real-time 系統的要求。

Publications:

- (1) C.-H. Yang and H.-M. Hang, "An investigation on low bit-rate speech coding over noisy channels," 2002 National Symp. on Telecomm., Puli, Nantou, Taiwan, Dec. 2002. (accepted)
- (2) K.-T. Shih, C.-Y. Tsai, and H.-M. Hang, "Real-time implementation of H.263+ using TI TMS320C6201 Digital Signal Processor," *ibid.*
- (3) J.-S. Lin and H.-M. Hang, "AMR coding performance analysis," *ibid.*