

# 第一章、緒論

近年來，數位多媒體內容 (digital multimedia content) 的相關技術及創作內容已快速蓬勃地發展，各種不同形式的影音特效是造就多媒體內容如此受大眾歡迎的主因，正由於多媒體資料是如此的豐富且複雜，相關的多媒體資料壓縮技術也已經成為重要的研究方向，其中，區塊壓縮 (block-based) 搭配移動補償 (motion compensation) 的方法已被許多國際壓縮標準所採納且行之多年，例如：JPEG、MPEG1/2/4、H. 264/AVC。雖然區塊壓縮可達到不錯的壓縮效率，但它同時也造成了視訊影像的失真，其中最明顯的就是方塊效應 (blocking effect)，為了減低此效應的影響，各個壓縮標準也分別制定其去方塊效應濾波器 (deblocking filter)，以求失真程度降到最低。如何利用後處理 (post-processing) 來降低此令人討厭的方塊效應就成了多媒體壓縮技術的一大考量了。

## 1.1. 研究動機

如今，全球大量地製造產生影像資料，來滿足人們對感官需求的增加，卻造成資訊量一直急速地爆增，也成為了傳輸媒介與儲存媒體的沉重負擔；在加上網路通訊的發達，更需要作有效的影像壓縮，不論是運用在影像本身或是影像傳輸…等，各種應用領域，此壓縮的技術扮演相當重要的角色。

而在現在最常用的多媒體壓縮技術中，由於是採用區塊壓縮的方式，所以其擾人的方塊效應會伴隨著壓縮技術而出現。如何去偵測出正確的方塊效應且使用適當的方式去消除方塊效應是一個在現在多媒體壓縮技術中存在的一個問題。

## 1.2. 研究目的

餘弦變換是經典的頻譜分析工具，它考察的是整個時域過程的頻域特徵或整個頻域過程的時域特徵。在JPEG中，離散餘弦變換(DCT)將圖像壓縮為8x8 的小塊，然後依次放入文件中，這種算法靠丟棄頻率信息實現壓縮，因而圖像的壓縮率越高，頻率信息被丟棄的越多。在極端情況下，JPEG圖像只保留了反映圖像外貌的基本信息，精細的圖像細節都損失了，這也就會造成所謂方塊效應[1]。

近年來，小波分析成為研究熱潮。它在信號處理、圖像壓縮、語音編碼、模式識別、地震勘探以及許多非線性科學領域內獲得了巨大的突破。小波 (Wavelet) 是以數學基礎為發展背景的一種訊號處理技術，而且不同的小波類型有其不同的波形型態，當然所產生的功能也就不一樣，應用的範圍也不同，例如影像處理、訊號分析等等工程問題。小波轉換的基底是由一個原型 ( prototype ) 函數的擴張 ( dilation ) 與平移 ( translation ) 所形成的[10]。這些基底函數具有短持續時間且高頻率，以及長持續時間且低頻率的特性，因此相當適合表達高頻的突發暫態信號或是長時間緩慢變化的信號。相較於傅立葉系列所採用的弦式波基底而言，小波轉換的基底有較短的持續時間 ( time duration )，因而得名。

因為方塊(block)出現的地方是固定的，而且一定是跟附近像素的值有極大的落差，也就是其斜率(slop)會很大，這表示其有很大的高頻訊號出現。因為小波變換是現代譜分析工具，它既能考察局部時域過程的頻域特徵，又能考察局部頻域過程的時域特徵。它能將圖像變換為一系列小波系數，這些系數可以被高效壓縮和存儲，也可以對特定階層的小波係數做處理，達到我們對影像所想要的要求，所以在對方塊效應上，小波也應該會有很大的空間可以發展[7][17][18]。

近年來，離散小波變換在圖像處理與圖像分析的各個領域中得到了廣泛的應用。這主要因為小波的時頻局域化使它在信號分析中具有很大的優勢，而且由於它對高頻成分採用由粗到細的漸進取樣間隔，從而可以放大任意細節。因此，小波分析被譽為數學上的顯微鏡，是構造圖像多分辨率表示的有力工具。

### 1.3. 現有方法

在1983年，在Reeve 和Lim的研究中[1],應用的時域上的濾波器來對方塊效應做處理,但因為是對全部的方塊(subblock)邊界(boundary)做低通濾波器的處理,所以會對影像本身造成模糊化的後果, Jarske et al 提出的去方塊效應[2]則會去判斷每個方塊邊界的特性,是否有邊緣(edge)存在,來施與不同的濾波器.但是因為要有閾值(threshold)來決定方塊邊界的邊緣判斷,會造成不同的影像要用不同的閾值來做調整,才會達到去方塊效應的作用,所以在實務上是比較難達成的. 在Ramamurthi 和 Gersho的研究中[3],也認為去方塊效應不可以只用一種濾波器,所以先判斷方塊內是不是有邊界存在,如果有的話,要判斷邊界的方向,也就是要依邊界的方向用不同的方向來對方塊做濾波器的處理,如果是平滑的方塊,則用2維(2D)濾波器來做處理,可以看出Ramamurthi 的目的不只處理方塊邊界地方,會連方塊內部都會做後處理.而缺點也就是要因為每個影像的特性,是否比較多邊緣(edge)或是平滑度來做不同閾值或者是再切為更小的方塊來做判斷,但這對實際的情況是不符的. 在Lee等人[4]的研究中利用Sobel算子來區分出邊界和邊界的區域,再利用適當的濾波器來做去方塊效應.另外就是利用凸集合投影法(POCS, [23][24][24])設定適當的條件,對重構(reconstructed)影像在這些條件中來做遞迴投影,最後希望會收斂到所有設定條件的交集. POCS的缺點也就是其只能在軟體實踐,且要遞迴運算,設定的條件愈多,去方塊的結果會愈好,但相對的運算量會更大。另外因為小波的特性,利用小波的多重解析

度來對重構影像做小波分解,在Kim 等人[7]中利用1維小波分解後,判斷出方塊的位置,要利用過完備小波(overcomplete wavelet)對第一級和第二級小波細節係數(detail coefficient)來做去方塊效應,在[15]也是利用(overcomplete wavelet)中利用第一級小波細節係數和第二級小波細節係數的相乘結果來判斷是不是影像的邊緣,要將不是邊緣的小波細節係數都清為0,來達到去方塊效應的結果。

## 1.4. 論文貢獻

因為方塊出現的地方可以視做是一個高頻雜訊,而小波由於其多尺度(multi-scale)且多解析度(multi-resolution)的特性[11],可以找出適當的小波來偵測方塊出現之處,再用除噪(de-noise)的方式來消除特定高頻訊號,將可達到解決方塊效應的問題。本文利用過一維完備小波轉換的小波係數,來偵測出出現在影像中的每個方塊的邊界是屬於方塊效應還是影像的邊緣,或是影像的平滑(smooth)區域,並對平滑和方塊效應的點做不同的處理,利用Matlab的結果,去方塊效應的效果很好,且為了硬體的實現度,對演算法做了適當的折衷,當然前提是不可以失去去方塊效應的效果,且可以符合我們的硬體要求:時脈為100MHz,每秒可以處理D1大小(720x480)的資料量之下,利用verilog來實踐且透過FPGA來做實際的測試。

## 1.5. 章節介紹

本文的章節為第二章主要是針對現有的去方塊效應的方法和過完備小波及在小波去去噪(de-noising)的方法做一個說明,2.1 節是說明為何會有方塊效應,2.2 節是就現有的一些去方塊效應的方法做一個說明,2.3 則是說明在小波中去噪的方法,2.4 是解釋過完備小波的方法和其好處。

第三章則是說明本文利用過完備小波和小波中去噪的方法來做去方塊效應的方法。3.1 節是對原來 Kim 等人[7]所用的方塊效應判斷方法做一個說明, 3.2 節則是對[7]所用的去方塊效應的方法說明, 3.3 則是本文行用小波去噪來對[7]所判斷出的方塊效應來做改善。

第四章是利用 Matlab 來對 5 種不同的去方塊效應演算法來和本文的去方塊效應演算法做一個主觀和量化的比較。

第五章則是說明在硬體實踐上本文所用的方式和所要符合的要求。

最後附錄 A 則是說明小波快速轉換的原理和方法。



## 第二章、方塊效應與小波去噪

### 2.1. 方塊效應簡介

在影像編碼系統中，轉換編碼和移動估測對減少影像的大小是十分有幫助的，為了減少時域上(spatial)的資料大小，將時域上的資料轉換成頻域上的資料是因為要減少像素(pixel)在時域上的相關性，而在頻域上，便可以選擇性的壓縮想要的係數。

傳統上的影像壓縮技術，像是H.263, MPEG-1/2 和MPEG4 等音壓縮標準是採用方塊(8x8 pixel)為單位的離散餘絃轉換(block based discrete cosine transform, BDCT)，離散餘絃轉換(DCT)的公式如式(2-1):

$$F_{x,y} = \frac{C(x)C(y)}{4} \sum_{i=0}^7 \sum_{j=0}^7 f_{i,j} \cos\left(\frac{(2i+1)x\pi}{16}\right) \cos\left(\frac{(2j+1)y\pi}{16}\right) \quad (2.1)$$

其中  $f_{i,j}$  是時域上的8x8=64個資料，

$F_{x,y}$  是經DCT轉換後頻域上的64個資料，

$$C(n) = \begin{cases} \frac{1}{\sqrt{2}}, n=0 \\ 1, n \neq 0 \end{cases} \quad (2.2)$$

實際上，BDCT並不會達到壓縮的效果，BDCT的好處是因為它是近似完美的能量集中特性(energy compaction property)、極好的頻率區分特性和在硬體實踐上的快速。而要達到壓縮的效果，主要是要將BDCT的資料產生的係數做適當的量化(quantization)，來減化其係數的作用。量化的主要作用是將不重要的資料(如較高頻的資料)用較大的權重的量化因子(quantization factor)來做量化，也就是其量化步階(quantization step)較大，而比較重要的資料用較小的量化因子來做量化。經此量化後，其實已經破壞了原來

的訊號, 因為當影像在重構回來時, 會將量化的係數再乘上當初量化的因子, 如此重構回來的係數已經失去了原來的一些資料了, 可以看做量化其實就是在做離散化(discrete)的動作. 在低壓縮率之下, 失去的資料量很小, 所以重構回來的影像人眼是看不出來的, 然而在高壓縮率之下, 將會出現一塊一塊容易被看出來的方塊效應. 式(2-3)說明了量化的方式和反量化:

$$F_{i,j}^Q(u,v) = \text{round}\left(\frac{F_{i,j}(u,v)}{Q(u,v)}\right) \dots \text{Quantization} \quad (2.3)$$

$$F_{i,j}^R(u,v) = F_{i,j}^Q(u,v) \cdot Q(u,v) \dots \text{reconstruction}$$

方塊效應是重建影像時, 因為每個方塊之間的振幅, 或是彼此間的平滑度已經被破壞, 造成不連續的現象. 由於方塊效應是人為的結果, 且造成的每個方塊的不連續平滑會吸引人的目光, 更加強化了人對方塊效應的不舒服感. 最明顯的就是在每個方塊的相接之處(boundary). 圖1顯示了經過高度壓縮之後的影像和原來的影像的差別, 可以看出經過壓縮之後的影像有明顯的方塊效應.

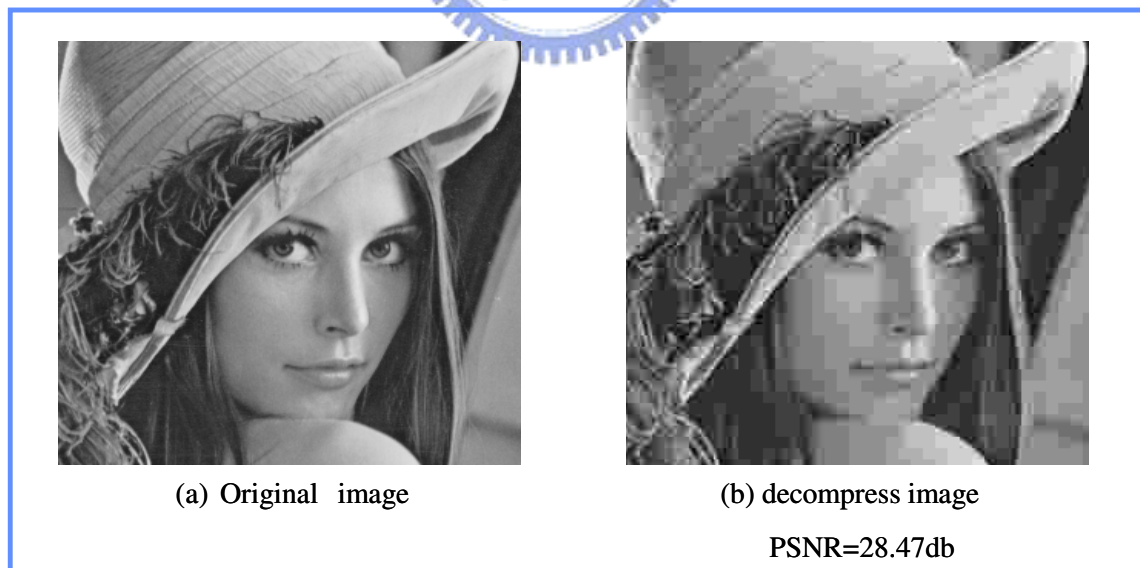


圖1: 方塊效果範例

## 2.2. 一般時域去方塊效應演算法

這一節會說明傳統在時域上如何來去除方塊效應。

- I. 在 Reeve 和 Lim 的研究中[1],認為方塊效應是因為在做 BDCT 時每個 8x8 的方塊彼此並沒有相重疊的原因,再加上量化的影響,在 low bit rate 之下,會造成每個方塊重構時因為缺乏了其他方塊的資訊,所以會在影像上造成明顯的方塊效果,所以 Reeve 等人提出了二種方法來改善方塊效應,第一種是重疊法(overlay),希望可以改善方塊效應應用的時域上的濾波器來做,以每個方塊是 3x3 為例,每個方塊和相鄰的方塊重疊了一個像素(pixel)的距離,如圖 2,

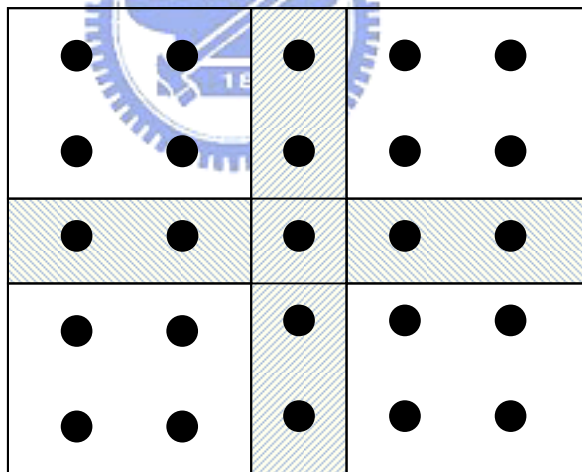


圖 2:[1]的重疊法例子

由圖 2 可以看出,因為每個 3x3 的方塊和相鄰的方塊各重疊了一個像素,所以在接下來做 BDCT 時,會因為有相鄰方塊的資訊,而在重構時不會有跟相鄰方塊有不相連的情況發生,也就是方塊效應會被消除。缺點是這和一般的影像編碼所用的流程是不一樣的,所以在解碼時也是和標準的影像解碼不同,這會造成不相容的情況發生。且在資料量方



面也是會有很大的增加。

第二種是濾波器方法,圖 3 是[1]所用的時域上的濾波器:

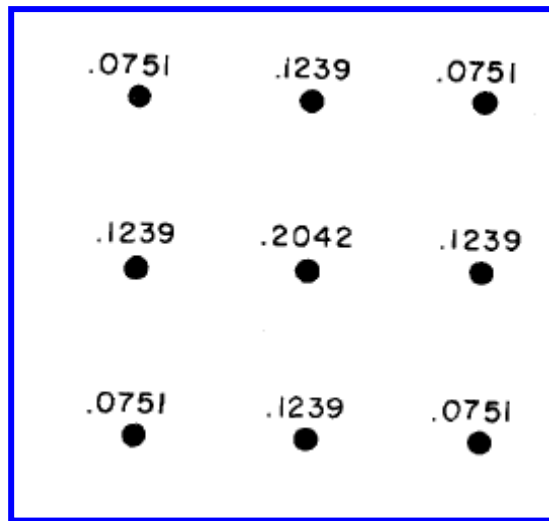


圖 3:[1]所用的濾波器係數

,因為方塊效應是可以看做在兩兩8x8方塊之間有不相連的形狀,所以如果用一個低通濾波器來做處理,應該會有很大的改善,如圖 3 所示的係數,而這個濾波器主要是只用在每個 8x8 方塊相鄰的點.但缺點就是會因為都用低通濾波器,所以影像的有些邊緣剛剛好在每個 8x8 方塊的邊界的話,會被過度模糊化,如果影像是很多邊緣的話,那反而會因為使用低通濾波器,而產生過於模糊的效果。

II. 而在 Jarske 等人提出的去方塊效應[2]為 Averaging Threshold Filter (ATF) 和 Gaussian Low Pass Filtering ,分述如下:

i. Averaging Threshold Filter (ATF):

每個 8x8 的方塊中,因為在方塊的邊界處和方塊內部是不同的,不能用同一個濾波器來做,所以[2]利用一個閾值(threshold)來決定不同的濾波器,其中判斷的方法為:

$$AVE\{x_{k,l} \mid (|x_{k,l} - x_{i,j}| < T)\} \quad (2.4)$$

其中  $x_{i,j}$  是方塊中心的像素值,  $x_{k,l}$  是方塊其他地方的像素值.而  $T1=10, T2=30$ .

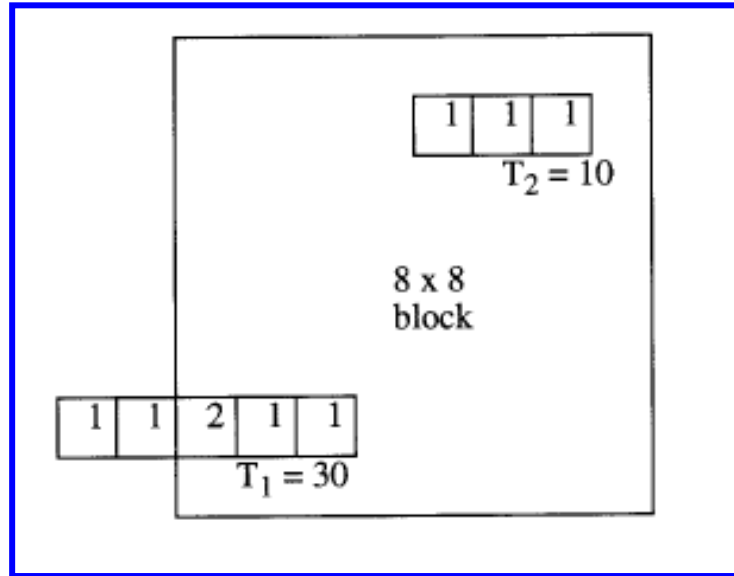


圖 4:[2]所用到的 ATF 示意圖

ii. Gaussian Low Pass Filtering:

利用 Gaussian Low Pass Filter 的低頻濾波器的方法來去除每個方塊邊界 (boundary) 的方塊效應。

其 Gaussian Low Pass Filter 的係數為:

$$h(n) = \{-0.031, 0.29, 0.48, 0.29, -0.031\} \quad (2.5)$$

III. 在 Ramamurthi 和 Gersho 的研究中[3],認為要做好去方塊效應,不可以只用一個濾波器來做全面的處理,而且因為一般會用低通濾波器(Low pass filter)來做,所以用造成本來影像邊緣(edge)會被模糊掉,反而造成另一種影像失敗的效果,所以在[3]中,希望可以判斷出影像的邊緣(edge)或是平滑(monotone)區域,對不同的區域做不同的處理.而在邊緣(edge)的判斷中,又會去判斷不同的邊緣(edge),像是 45 度, 0 度, 90 度, 135 度的不同邊緣(edge),按照不同的邊緣(edge)角度,來沿著邊緣(edge)來做濾波.在平滑區域用的濾波器的係數為:

$$h^{2d}(i, j) = h(i)h(j), \quad (2.6)$$

$$h(i) = \{0.227, 0.547, 0.227\}$$

而沿著邊緣的一維濾波器係數為{0.036,0.282,0.363,0.282,0.036}

判斷邊緣是[3]的特點,利用每個方塊(8x8,E),在方塊 E 內分別對水平的像素兩兩相減,即:

$$\begin{aligned}
 \Delta x_{i,j} &= g_{i+1,j} - g_{i,j} \\
 T_{i,j} &= T / 2 \times \{g_{i+1,j} + g_{i,j}\} \\
 k &\leftarrow k + 1 \text{ if } \Delta x_{i,j} > T_{i,j} \\
 k &\leftarrow k - 1 \text{ if } \Delta x_{i,j} < -T_{i,j}
 \end{aligned} \tag{2-7}$$

其中 T 是一個定值,在[3]是設為 0.15

同時也對垂直方向來做,來得到一個變數 L,

再用 K,L 來判斷是平滑區域或是邊緣.

$$\begin{aligned}
 1) & \textit{monotone}, \textit{if } |K| < m \textit{ and } |L| < m \\
 2) & \textit{edge}, 0^\circ, \textit{if } |K| < m \textit{ and } |L| > m \\
 3) & \textit{edge}, 45^\circ, \textit{if } |K| > m \textit{ and } |L| > m, \textit{and } \textit{sgn}(K) = \textit{sgn}(L) \\
 4) & \textit{edge}, 90^\circ, \textit{if } |K| > m \textit{ and } |L| < m \\
 5) & \textit{edge}, 135^\circ, \textit{if } |K| > m \textit{ and } |L| > m \textit{ and } \textit{sgn}(K) = -\textit{sgn}(L)
 \end{aligned} \tag{2-8}$$

其中 m 是一個定值,在[3]是設為 3

IV. Lee 等人 [4]利用 Sobel 算子來區分是平滑或是邊緣區域,用 Soble 算子來區分平

滑或是邊緣比在[4]的區分方式是比較好的,但缺點就是運算量較大.

式(2-8)列出 Sobel 算子的區分方式:

$$\begin{aligned}
 \textit{Horizontal gradient} : G_h(x, y) &= S(x, y) * I_h(x, y) \\
 \textit{Vertical gradient} : G_v(x, y) &= S(x, y) * I_v(x, y) \\
 \textit{Absolute of the gradient vector} : M(x, y) &= |G_h(x, y)| + |G_v(x, y)| \\
 \textit{Direction of the gradient vector} : \theta(x, y) &= \tan^{-1} \left( \frac{G_v(x, y)}{G_h(x, y)} \right)
 \end{aligned} \tag{2-9}$$

利用一個選擇的閾值(T),來決定是不是邊緣,

決定了平滑或是邊緣區域,再用不同的濾波器來做去方塊效應且保留原來的邊緣.

## V. 使用凸集合投影法(POCS, [23][24][24]):

所謂凸集合定義為:

若一集合 $D$ 內的任兩點, $x_1, x_2 \in D$ 且 $\alpha$ 是比例實係數, $0 \leq \alpha \leq 1$ ,滿足下式得到的

$x_3 \in D$ 的話,此集合 $D$ 為凸集合.

$$x_3 = \alpha x_1 + (1 - \alpha)x_2$$

若集合 $C$ 為凸集合,則點 $z$ 用基本投影法只會在集合 $C$ 上產生唯一投影點(圖5),若 $C$ 不是凸集合,則點 $z$ 基本投影在集合 $C$ 上,投影點不只一個,這也就是為什麼要用凸集合投影法.

若我們想回復,設計或計算一個未知的量 $f$ ,可以將未知的量 $f$ 所擁有的所有已知特性,將其當作為限制(constraint)條件的方程式,再利用這些式子來計算推得原有未知的量 $f$ . 如果有 $m$ 個限制集合(constraint set) $C_1, C_2, \dots, C_m$ ,且 $C_0$ 是此 $m$ 個限制集合的交會集合(intersection set),則 $f$ 一定會位於其交會合內, $f \in C_0$ .

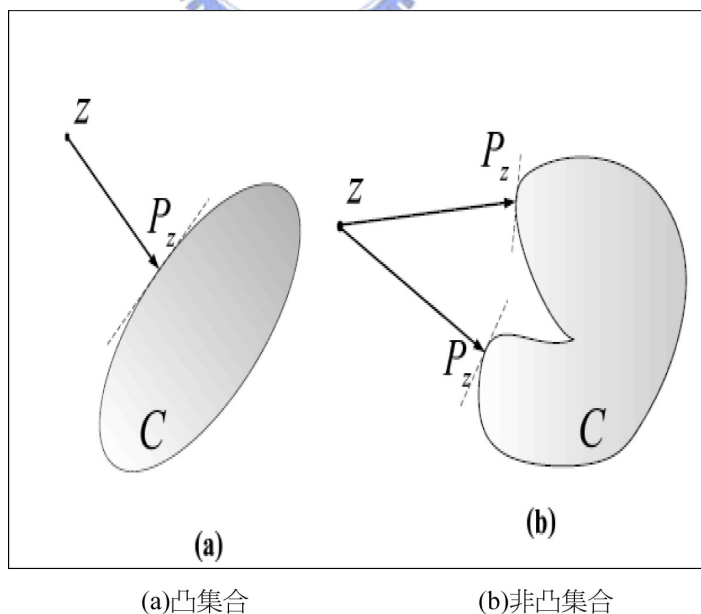


圖 5: 凸集合 vs 非凸集合

當所有的限制集合 $C_i$ 為凸集合時,從上述介紹可以知道投影點只有唯一,所以我們可以在此 $m$ 個凸集合間,以遞迴方式依序投影(圖6),逼近於交會集何 $C_0$ 內的未知量 $f$ . 這

一種有系統的數值演算法方式被稱為凸集合投影法。

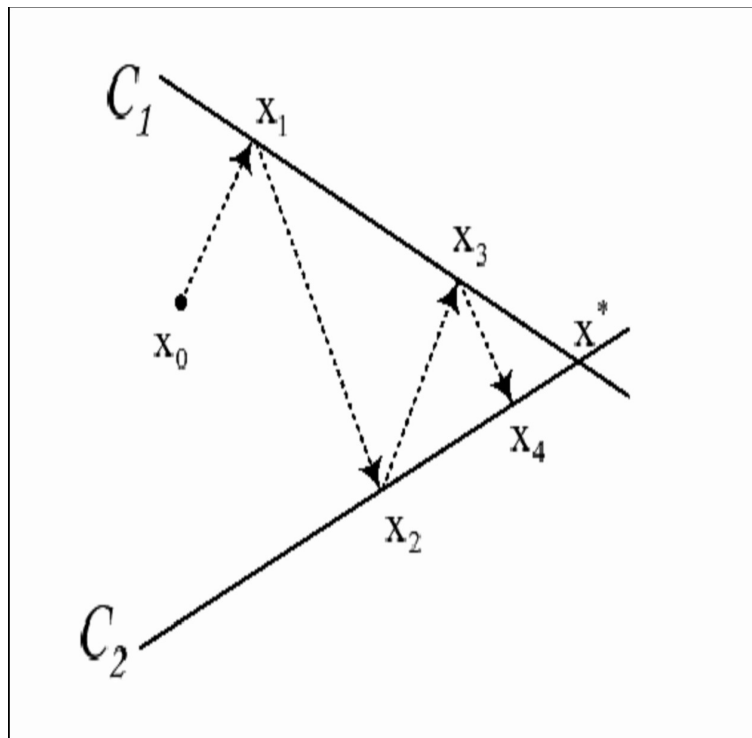


圖 6:POCS 遞迴投影

凸集合投影法是所有去方塊效應演算法中最佳解，不會有過度模糊且對去方塊效應又有絕佳的效果[20][7][19]. 但因為其使用遞迴方式來做，在計算量是十分大的，且有其複雜度。

#### VI. 二維過完備小波去方塊效應演算法[18]:

在Xiong 等人使用的是二維(2-D)過完備小波轉換來分解影像[18], ,其概念是利用過完備小波中的特性, 即頻域與時域的一致性, 來判斷影像的邊緣(edge). 其利用第一級和第二級小波細節係數的相乘, 由過完備小波的特性, 如果是影像的邊緣(edge), 則其相對應點的小波細節係數不論是在第一或第二級皆會很大, 所以如果第一級和第二級小波細節係數相乘的結果很大的話, 那就代表這個點就是影像的邊緣(式2-10)[18].

$$\begin{aligned}
E^H(m,n) &= W_1^H(m,n) \bullet W_2^H(m,n) \\
E^V(m,n) &= W_1^V(m,n) \bullet W_2^V(m,n) \\
E(m,n) &= \begin{cases} 1, & \text{if } E^H(m,n) + E^V(m,n) > T \\ 0, & \text{otherwise} \end{cases}
\end{aligned} \tag{2-10}$$

為了決定判斷邊緣, 須設定一個閾值T.

$$T = \frac{1}{MN} \left\{ \sum_{m=0}^{M-1} \sum_{n=1}^{\frac{N}{8}-1} [f(m,8n) - f(m,8n-1)]^2 + \sum_{m=1}^{\frac{M}{8}-1} \sum_{n=0}^{N-1} [f(8m,n) - f(8m-1,n)]^2 \right\} \tag{2-11}$$

其中f(m,n)是時域上的訊號, N, M為f的大小.

判斷出E(m,n)之後, 把不屬於邊緣的小波細節係數皆設為0, 只留下是邊緣的小波細節係數.

$$W_j^{O \in \{H,V\}}(m,n) = \begin{cases} W_j^{O \in \{H,V\}}(m,n), & \text{if } E(m,n)=1 \\ 0, & \text{if } E(m,n)=0 \end{cases} \tag{2-12}$$

而且把小波近似係數也依E(m,n)的判斷來做下列的處理.

$$S_2(m,n) = \begin{cases} \frac{1}{3}(S_2(m,n-1) + S_2(m,n) + S_2(m,n+1)), & \text{if } E(m,n) = 0 \text{ and } n=0(\text{mod } 8) \\ \frac{1}{3}(S_2(m,n-1) + S_2(m,n) + S_2(m,n+1)), & \text{if } E(m,n) = 0 \text{ and } m=0(\text{mod } 8) \\ S_2(m,n), & \text{otherwise} \end{cases} \tag{2-13}$$

由上式可知, 以2維來做過完備小波轉換, 其運算量會十分大, 因為過完備小波轉換的缺點就是其轉換後的細節係數和近似係數和原來訊號大小相同, 以2維來看, 原來是MxN大小的影像, 經過2級的過完備小波轉換後, 其全部須處理的資料量為6xMxN, 即隨影像大小有2J(J為小波級數)的倍數成長, 所以如果設計為硬體須有極大的負擔.

再加上其須對第一級和第二級小波細節訊號相乘, 這也是不利硬體設計的. 更不用

就因為硬體頻寬的不足,會使的處理所須的fps(Frame Per Second, 幀/秒)低於標準的 30fps.

除了上述的 6 種,還有[4][5]中也都提出了重疊(overlap) 的方式,也就是在做 DCT 之前,各種編碼都會以方塊為基礎,而 overlap 的方法就是每個方塊的切割方法是有部分和其他的方塊重疊的,再去做 DCT 的頻域轉換,希望能在 DCT 的頻域轉換的結果會和其他相鄰的方塊有某些關聯,如此一來,就可以大大降低方塊效應的發生,但這個方法會影響到影像編碼的標準,所以實踐上和標準不太相合.

而因為方塊效應是因為頻域上轉換的關係,所以在[6]中利用重構回來的影像,每個方塊和相鄰的方塊之間的像素的差異,來反推回去修改在頻域上的係數,再 IDCT 重構一次影像來達到去方塊效應的方法.

其實可以看出,要解方塊效應應該不可只用單純的濾波器方式來做,要考慮影像的邊緣和平滑的區域才行,而在時域上,能有效的分辨出邊緣和平滑的區域,主要用的除了 Sobel 算子還有 canny 算子,但其運算量都很大,對硬體而言是一大負擔.

近年來,小波因為其時頻域的特性,所以在影像處理上漸漸佔有一席之地,而應用在去方塊效應的地方也愈來愈多,所以本文利用小波的特性[7][10],來做去方塊效應的方法.

有關小波的演進,請參考附錄 A,而下一節將對小波的一些特別應用來做說明.

### 2.3. 小波去噪和 Donoho 演算法

由於 blocking effect 可以看做是一種噪音加在原本的訊號上,而且因為 blocking effect 是高頻訊號,所以如果我們可以利用小波的多重解析度的概念且能找出應是 blocking effect 的點,那我們是否可以利用去噪的方式來做 deblocking 的處

理?所以接下來我們要先介紹小波去噪的方式,接下來再介紹過完備小波。

在小波去噪的方法中,Donoho 和 Johnstone(1994) [8][9]提出了一個方法由一個有噪音的訊號去重建一個未知道的訊號,他們的假設如下:

假如我們想去由一個有噪的函數中去復原一個未知道且介於 $[0, 1]$ 的函數  $f$ , 則:

$$d_i = f(t_i) + \sigma z_i, i = 0, \dots, n-1 \quad (2-10)$$

其中  $t_i = i/n$ ,  $z_i$  是一個標準的高斯白噪音(Gaussian white noise)且是相互獨立且具有相同分配(independent and identically distributed (i. i. d)); 定義為  $z_i \stackrel{iid}{\sim} N(0,1)$ .

且  $\sigma$  是噪音的振幅. 我們想要去噪(de-noisine)是想要去得到  $\hat{f}$  且滿足 mean-squared error:

$$n^{-1} E \left\| \hat{f} - f \right\|_{l_n^2}^2 = n^{-1} \sum_{i=0}^{n-1} E(\hat{f}(i/n) - f(i/n))^2 \quad (2-11)$$

同時要滿足下列條件:

$$\hat{f} \text{ 要和 } f \text{ 一樣的平滑(smooth)} \quad (2-12)$$

為什麼要滿足式(2-12), 因為很多統計上的技術只有簡單的優化 mean-squared error, 但是在訊號的 bias 和變異數(variance)並沒有和原來的訊號一致. 因為如果只是單純優化(optimal) 均方誤差(mean square error)將會產生很多噪音所導致的振盪的情況(如:ripples, blips...), 所以才有條件(2-12)的出現. 而這些由噪音所引起的振盪(oscillations)在某些應用上(比方說天文和地球物理學)是很去辨別是真的訊號或是噪音引起.

而在 Donoho 和 Johnstone(1992) [8][9]提出的一個非常簡單的閾值處理來復原有噪的訊號. 且滿足兩個一般想要的考量:

**[Smooth] :**  $\hat{f}$  要和  $f$  一樣的平滑(smooth).



[Adapt] :  $\hat{f}$  要滿足極小化最大均方誤差(minimax mean square error)<sup>1</sup>. 即

$$\inf_{\hat{f}} \sup_{F_C} n^{-1} E \|\hat{f} - f\|_{L_n^2}^2$$

$F[0,1]$  is a function space (Triebel or Besov spaces)

$F_C$  : the ball of functions  $\{f : \|f\|_F \in C\}$

為了得到一個方法來做去噪, Donoho and Johnstone[8][9]去得到了一個小波收縮 (wavelet shrinkage)的方法來做, 簡述如下:

假設 data  $y = (y_i)_{i=1}^n$  with  $n = 2^{J+1}$ , 且令  $n \times n$  小波轉換正交陣列(orthogonal matrix)  $W$ , 則我們可得小波細節係數為:

$$w = Wy \tag{2-13}$$

因為小波轉換陣列是正交的, 所以我們有反小波轉換為:

$$y = W^T w \tag{2-14}$$

而小波細節係數  $w$  是一個  $2^{J+1} \times 1$  的向量, 而我們可以定義每一級的小波細節係數為:

$$w_{j,k} : j = 0, \dots, J; k = 0, \dots, 2^j - 1, \tag{2-15}$$

且  $W_{jk} : (j,k)$ -th row of  $W$

所以其反小波轉換為:

$$y = W^T w \rightarrow y_i = \sum_{j,k} w_{j,k} W_{jk}(i) \tag{2-16}$$

而我們想要重建的訊號中, 我們只要選擇性的小波細節係數來重建, 所以

$$T_{SW}(y, \delta) = \hat{f} = \sum_{(j,k) \in \delta} w_{j,k} W_{jk} \tag{2-17}$$

<sup>1</sup> **sup** S == S 之上確界(supremum); **inf** S == S 之下確界(infimum)

$\delta$ : 我們想要的特定條件的小波細節係數集合.

我們要假設下列幾項:

(1) 噪音(noise,  $e$ )是分佈在所有的小波細節係數的. 所以  $e = (e_i)$  可以看做是白噪音 (white noise).

(2) 噪音( $e$ )的正交轉換( $z = We$ )也是一個白噪音.

(3) 對於消噪, 我們只是要在某些特定條件的小波細節係數集合內動手腳(most of the action).

所以經驗上(empirical)小波細節係數可以看成下式:

$$w_{j,k} = \theta_{j,k} + z_{j,k} \quad (2-18)$$

其中  $\theta = Wf$ ,  $f$  is noiseless data  $f = (f(t_i))_{i=0}^{n-1}$

我們想要去發現一個估算子(estimator)去得到小的均方根差(mean-squared error)

$E \|\hat{\theta} - \theta\|_{\ell_n^2}^2$  並滿足其平滑的條件

$$|\hat{\theta}_I| \leq |\theta_I|, \forall I \in I_n$$

先提出一個簡單的抽象模型,

$$y_I = \theta_I + \delta \cdot u_I, I \in I \quad (2-19)$$

其中,  $\delta > 0$  是噪音振幅,  $u_I$  是一個噪音項, 且滿足  $|u_I| \leq 1, \forall I \in I$ .

則我們想要得到最差的結果:

$$E_\delta(\hat{\theta}, \theta) = \sup_{|u_I| \leq 1} \|\hat{\theta}(y) - \theta\|_{\ell_n^2}^2 \quad (2-20)$$

我們想要式(2-20)最小, 所以要保證均勻收縮條件(uniform shrinkage condition):

$$|\hat{\theta}_I| \leq |\theta_I|, I \in I. \quad (2-21)$$

而在[8]證明軟閥值演算法可以滿足均勻收縮條件.

$$\eta_t(y, \lambda) = \text{sgn}(y)(|y| > t)_+ \quad (2-22)$$

根據式(2-20)和式(2-22)可以改寫成：

$$\begin{aligned} \hat{\theta}^{(\delta)}(y) &= \eta_t(y_t), I \in I \\ E_\delta(\hat{\theta}^{(\delta)}, \theta) &= \sum \min(\theta_t^2, 4\delta^2). \end{aligned} \quad (2-23)$$

也可以得到軟閾值演算法是近似極小化最大均方誤差(near-minimax error)[9].

而有了估算子--軟閾值演算法之後,就要來決定閾值,因為訊號上的雜訊不是已知的,所以有各種的閾值選擇法被一一提出,在說明各種閾值的選擇之前,先說明除了軟閾值演算法可以用在做小波係數的去噪,也有所謂的硬閾值演算法,簡述硬閾值演算法和軟閾值演算法如下:

(a) 硬閾值演算法(hard-thresholding):

簡單來說就是將所有低於臨界值的係數都設為零

$$y = \begin{cases} x & |x| \geq c \\ 0 & |x| < c \end{cases} \quad (2-24)$$

式中, c是閾值. 如圖7.

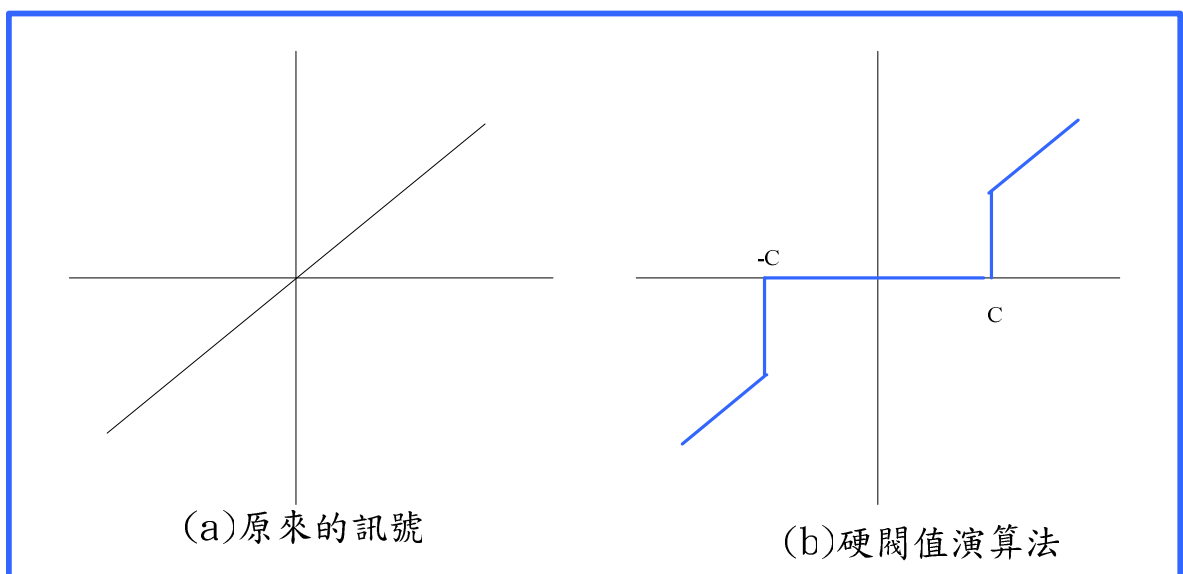


圖 7: 硬閾值演算法

(b). 軟閾值演算法(soft-thresholding):

是硬式臨界值演算法的延伸，它仍然是將所有低於臨界值的係數設為零，但是它同時縮減了其餘所有的係數，如圖8:

$$y = \begin{cases} \text{sgn}(x)f(|x|-c) & |x| \geq c \\ 0 & |x| < c \end{cases}, \quad (2-25)$$

硬閾值演算法的計算最為簡單，而軟閾值演算法有較好的數學特性且仍維持影像中的邊緣特徵[11]. 經過實驗比較後，也證明了軟閾值有較佳的效果. 所以本文是採用軟閾值來做去方塊效應的方法之一. 而決定了軟閾值演算法次後，最重要的就是定出要過濾掉的雜訊的閾值大小的選取。太大的閾值可能會造成信號的失真，而太小的閾值又無法有效的去除雜訊，依據 Donoho and Johnstone 提出了下列 3 種閾值選擇的方法：

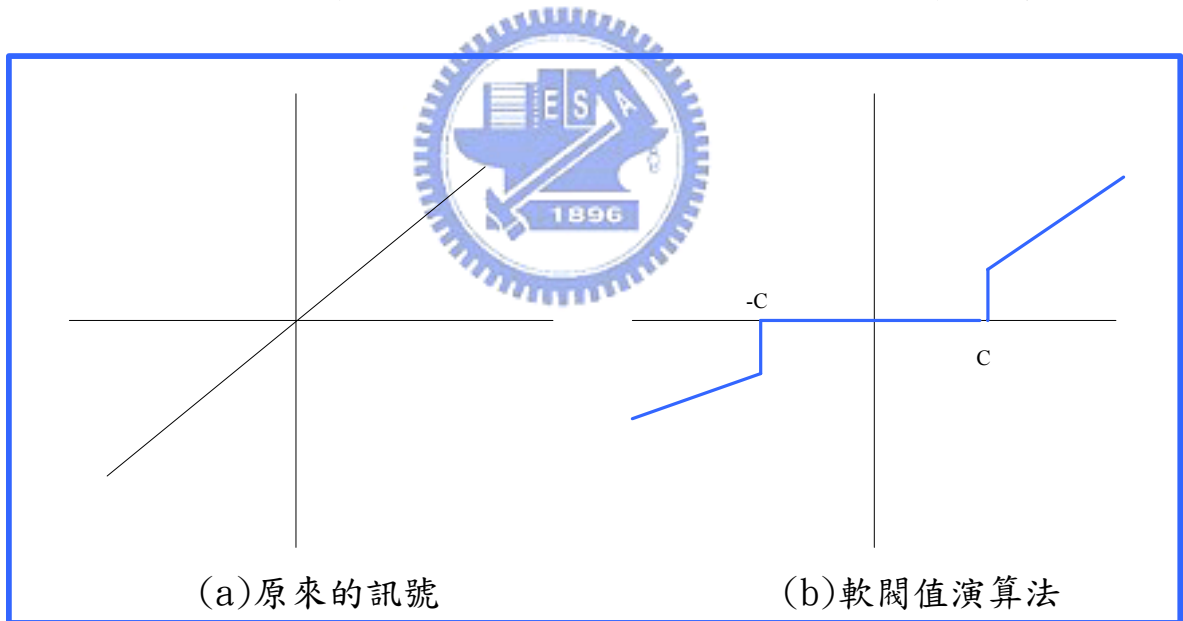


圖 8: 軟閾值演算法

### 2.3.1 通用閾值(Universal threshold, VisuShrink):

Donoho 和 Johnstone(1994) 提出了閾值(VisuShrink)的選擇，且故名思意，這個閾值是對各層的小波細節係數都是一樣，也就是一個全域的閾值(global):

$$\hat{\theta}^v : \text{estimator in the wavelet domain.} \quad (2-26)$$

$$\hat{\theta}^v = \eta_t(w_{j,k}, \sigma\sqrt{2\log n})$$

$$t = \varepsilon_1\sqrt{2\log n} = \sigma\sqrt{2\log n} \quad (2-27)$$

綜合上述, 可以得到流程圖(圖 9)和式(2-28):

- (a) 先對含噪的訊號做小波分解.
- (b) 將選擇的閾值求出, 並用軟閾值來對所有小波細節係數中做去噪的處理.
- (c) 重建去噪的小波細節係數和小波近似係數, 來求得去噪後的訊號.

$$\hat{f}_n^v \equiv W^T \circ \hat{\theta}^v \circ W \quad (2-28)$$

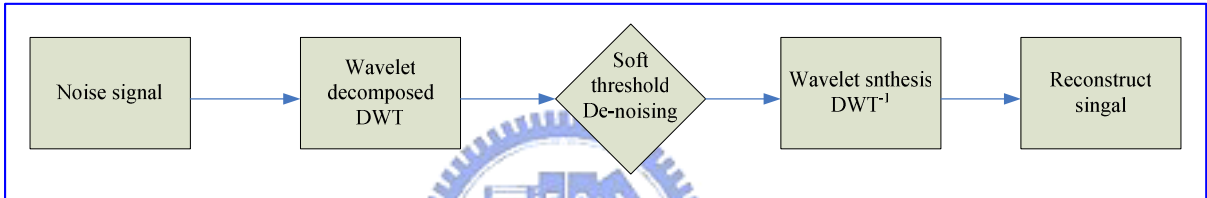


圖 9: donoho de-noise flow

因為  $\sigma$  為噪音的變異數, 為理想之下已知的條件, 並不符合實際的情況, 所以在 Donoho 和 Johnstone (1994)[8][9] 提出了一個由小波細節係數來求出的快速  $\sigma$  求法, 估算噪音準位為由  $\sigma$  改為小波細節係數的中位絕對值偏差量 (median absolute deviation, MAD) 在最細階 (finest scale) 的小波轉換的小波細節係數 [8][9] 再除以 0.6745. 為什麼選最細階 (finest scale) 的小波轉換的小波細節係數? 因為在經驗上, 最細階 (finest scale) 的小波轉換的小波細節係數實際上大部份是噪音 [8][9].

所以式(2-27)可以改為 MAD 型式:

$$t = \frac{\text{med}(|w_{j,k}|)}{0.6745} \sqrt{2\log n} \quad (2-29)$$

### 2.3.2 MSE 最低估測法則 (Minimax-performance threshold)

## selection Rule)

此法則目的為產生最小的 MSE(mean square error)值, 主要是設計一個統計上的估測因子(estimator), 而去雜訊後的訊號可以由估測因子的回歸函數(regression function)來近似[8][9].

MSE 最低估測法則的定義如下:

$$t = 0.3936 + 0.1829 \times (\log(n) / \log(2)) \quad (2-30)$$

其中  $n$  為訊號長度.

由上式可以看出, MSE 最低估測法則也是個全域(global)的閾值.

### 2.3.3 史汀無偏差 Risk 估測臨界值(SURE):

史汀無偏差 Risk 估測臨界值假設訊號  $f$  是隨機變數[8][9], 且並對每一層的小波細節係數所產生的閾值皆不同, 也就是 SURE 是一個局部(local)的閾值. 其想法為做估測後產生的均方誤差可由下式表示:

$$\begin{aligned} E\|f + g(f) - \theta\|^2 &= E\{trV + \|g(f)\|^2 + 2tr[V\hat{g}_k(f)]\} \\ &= E\{U(t; f)\} \end{aligned} \quad (2-31)$$

其中  $\hat{g}_k(f)$  可以用下式表示:

$$\hat{g}_k(f) = \begin{cases} -t & \text{for } f_k > t \\ -f_k & \text{for } |f_k| \leq t \\ +t & \text{for } f_k < -t \end{cases} \quad (2-32)$$

我們對  $U(t; f)$  做無偏差 risk 估測可得:

$$\hat{U}(t) = \sigma^2 n + \sum_k (f_k^2 \wedge \lambda^2) - 2\sigma^2 I\{|f_k| \leq t\} \quad (2-33)$$

我們可以從  $\hat{U}(t)$  中估測得一新的估測值  $\hat{t}(f)$

$$\hat{t}(f) = \arg \min_{0 \leq t \leq \sqrt{2 \log(n)}} \hat{U}(t) \quad (2-34)$$

而後把  $f = \frac{b}{\sigma}$  代入  $\hat{t}(f)$ , 再乘上  $\sigma$  可得:

$$\lambda = \sigma \hat{t}(b/\sigma) \quad (2-35)$$

其中  $b$  為訊號的小波轉換.

由 2.3.1, 2.3.2 和 2.3.3 節可以看出, 通用閾值計算簡單, 故得到很廣泛的應用, 但是其會趨向於過度丟棄小波係數, 會導致較大的重建誤差, 而 MSE 最低估測法則也會有這個問題, 最好的是 SURE 方法, 但缺點是計算太過複雜, 時至今日, 對閾值的選擇方法的研究仍在進行當中, 仍有新的閾值公式不斷被提出, 低通常閾值是根據實際應用的須要, 通過確定合適的準則, 並通過對可能的閾值來進行最好的選擇. 本文因為硬體實踐上, 通用閾值是比較容易實踐, 且效果也不錯, 所以採用通用閾值的方法來做去方塊效應的應用. 下圖(圖 10)表示了三種不同的閾值, 使用軟閾值演算法對一維訊號去噪的結果. 由圖 10 可以看出, 這三種閾值去噪都能有效降低原來含噪的訊號.

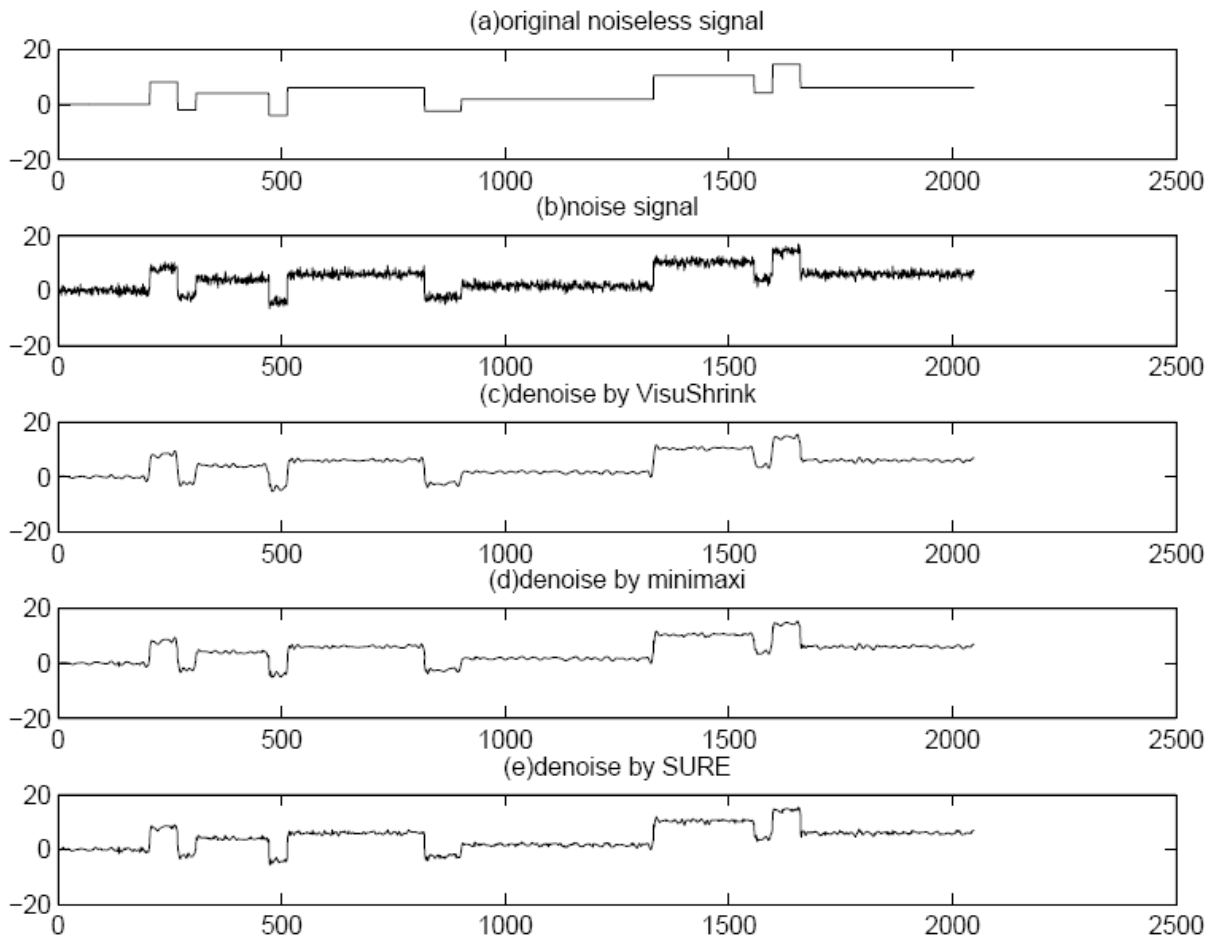


圖 10:3 種去噪處理的結果

## 2.4 過完備小波(Overcomplete Wavelet)

附錄A中的小波皆會對下一層的結果做次取樣(sub-sampling),其缺點即是上述的小波是非時移不變的.即改變原來訊號進入的時間,會產生不同的結果,不同的小波係數. DWT會是十分好用的主要是Mallat 快速演算法是一個對硬體實現非常適用的方法.對於有些算子,使用基波基底可以使相對應的矩陣成為稀疏矩陣(sparse matrix),從而使電腦計算加快。而且所有的小波係數總合是和原來的訊號大小一樣,這做運算量大大的降低,且對於資料壓縮和快速計算都是很有利的。

而最大的潛在問題就是DWT不是時移不變的[12] [13],它的次取樣的好處是在



降低資料量但使的時域和頻域產生不一致的現象. 即DWT比較強調在頻域上的分析而非時域上. 但這也導致了DWT在對方塊效應上的問題.

傳統的小波轉換的雜訊濾除存在著許多的缺點, 如在不連續點或頻率變動較大處會有Gibbs的現象發生, 那也是因為小波基底缺乏時移不變(translation invariance)的特性.

常用的快速離散小波轉換算法, 如Mallat快速算法, 採用的是二進塔式方案(Dyadic pyramid scheme)。這種方案的小波轉換輸出結果隨分析數據起點位置的移動而變。除非移動偶數個樣品位置, 否則信號移動後的小波轉換係數, 不等於信號未移動時的小波轉換係數移動相同位置後的結果. 這對於提取事件特徵或對事件進行定量分析而言是一大缺陷。為此, 開發出了許多小波轉換的輸出與信號移動無關的小波轉換移動不變算法來克服這一點, 便發展出了所謂非抽樣小波轉換(undecimated wavelet transforms)和小波循環旋轉.

### 2.4.1 小波循環旋轉

Coifman 提出的小波循環旋轉方法[46]是一種將所有位移的小波收縮結果取平均的方法, 由圖 11 可知道, 小波收縮循環旋轉是將訊號分別做順向  $k$  次的環形位移, 將這  $k$  次的位移結果做小波收縮處理, 然後, 我們再將上述小波收縮結果, 分別做逆向  $k$  次的環形位移, 最後我們平均這  $k$  次相加的結果, 因此可以用式子表示:

$$\bar{T}(x; (S_h)_{h \in H}) = Ave_{h \in H} [S_{-h}(T(S_h(x)))] \quad (2-36)$$

其中  $H$  是位移的範圍,  $T$  表示小波收縮演算法,  $S_h(x)$  是指對訊號做順向  $H$  次的環形位移,  $S_{-h}(x)$  是指對訊號做逆向  $H$  次的環形位移,  $Ave_{h \in H}$  是指平均這  $H$  個訊號.

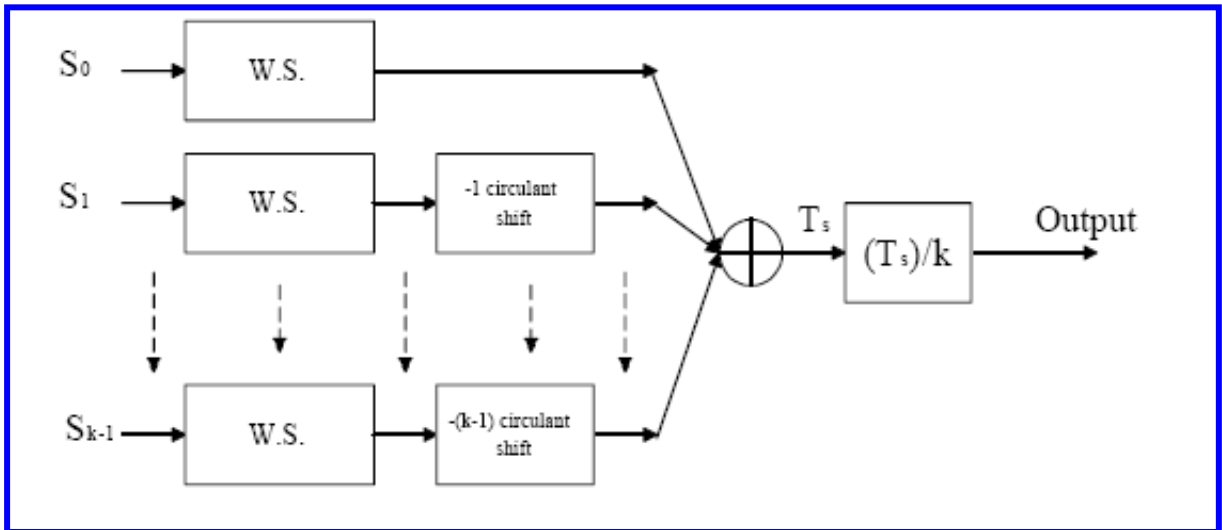


圖 11：小波循環旋轉

小波循環旋轉可以克服快速小波轉換不是時移不變的問題,但是其硬體複雜度是很高的,跟訊號  $N \times 1$  和環形位移的次數  $H$  呈  $O(HN)$  高度複雜度.

#### 2.4.2 過完備小波的分解和重構

基於Mallat快速小波算法的影像融合,由於存在抽取和插值運算,其結果存在一定的相位失真. 過完備小波算法通過有限濾波器內插近似,實現對影像數據的無抽取離散小波變換,很好地解決了上述問題.

過完備小波是由Mallat提出,主要是藉由小波轉換模極大化(wavelet transform modulus maxima, WTMM)的方式來分析訊號,同時發現其區域最大值(local maxima)的組合正好和Canny 邊緣偵測(Canny edge detector)相同[14]. 且使用過完備小波,在消噪(denoising)的效能上比使用次取樣的DWT有更好的效果[28].

由上述的優點,可以知道過完備小波是一個很適合來解決方塊效應的方法. 而缺點就是對於一個  $N \times N$  的資訊,其複雜度會是  $O(N \log N)$  [14]。

所謂過完備小波是因為其經過高通/低通濾波器後,並沒有對產生的訊號作次取樣的動作,也就是其細節空間(Detail Space)與近似空間(Approximation Space)的大小和

原來的訊號大小一樣. 也就是如此, 其方塊位置(block boundary)在每一層都是不變的, 且和時域的位置一樣. 這也就是為什麼過完備小波在消除方塊效應上有很大的優勢 [15].

而在過完備小波中最常用到的實現演算法是所謂多孔( $\hat{a}$  trous)算法, 它藉由修改標準的離散小波轉換(Discrete Wavelet Transform, DWT)在對於低通/高通濾波器的方式來達成. 它的實現是藉由濾波器的插值取樣(up-sample)來取代原本離散小波轉換對細節空間(Detail Space)與近似空間(Approximation Space)的次取樣. 而插值取樣是藉由在每一層的低通/高通濾波器的每兩點之間插入一個0來達成. (在法文中, trous 是孔的意思). 因為不對每一層的結果做次取樣的動作, 所以每一層的結果大小和原來的訊號一樣, 所以過完備小波也叫非抽取小波轉換(Undecimated Wavelet Transforms, UWT). 實際上, 由於它保留了每一個近似空間的係數, 且把近似空間的係數用來做下一層的輸入, 所以可以保留所有的高頻係數, 即相鄰的高頻振幅也可以表現出來, 這來消噪是一個很好的條件.

在[14]中, Mallat提出一個新的wavelet演算法, 也就是過完備演算法, 且給出了一組濾波器係數, 由[14]中, 可得其一維(1-D)分解(decompose)公式:

$$\begin{aligned} & \text{while}(j < J) \\ & w_{2^{j+1}}^d f = S_{2^j}^d f * G_j \\ & S_{2^{j+1}}^d f = S_{2^j}^d f * H_j \\ & j = j + 1 \\ & \text{end} \end{aligned} \tag{2-37}$$

其中,  $j$  開始是由0向 $J$ 分解,  $J \leq \log_2 N$ ,  $N$ : 訊號的長度.

而  $w_{2^j}^d$  表第 $j$ 級分解所得的小波細節係數.

$S_{2^j}^d$  表第 $j$ 級分解所得的小波近似係數.

$H$  : 第0級時的小波低通分解濾波器係數, 其轉換函數(transfer function)

為  $H(\omega)$ .

$G$  : 第0級時的小波高通分解濾波器係數, 其轉換函數(transfer function)

為  $G(\omega)$ .

$H_j$  : 第j級時的小波低通分解濾波器係數, 其為放入  $2^j - 1$  個0在每兩個H濾波

器係數之間. 其轉換函數(transfer function)為  $H(2^j \omega)$ .

$G_j$  : 第j級時的小波高通分解濾波器係數, 其為放入  $2^j - 1$  個0在每兩個G濾波

器係數之間. 其轉換函數(transfer function)為  $G(2^j \omega)$ .

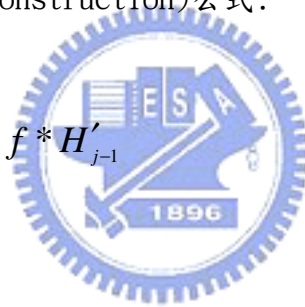
而其一維(1-D)重構(reconstruction)公式:

$while(j > 0)$

$$S_{2^j}^d f = w_{2^j}^d f * K_{j-1} + S_{2^j}^d f * H'_{j-1} \quad (2-38)$$

$j = j - 1$

$end$



$H'_{j-1}$  : 第j-1級時的小波低通重構濾波器係數, 為  $H_{j-1}$  的反時間序列.

$K_{j-1}$  : 第j-1級時的小波高通重構濾波器係數.

而第0級的濾波器係數[14]分別為表1, 其分解方塊圖(圖12)和其重構方塊圖(圖

13):

n	H	G	K
-3			0.0078125
-2			0.0054685
-1	0.125		0.171875
0	0.375	-2	-0.171875
1	0.375	2	-0.0054685
2	0.125		-0.0078125

表格 1: 過完備小波係數

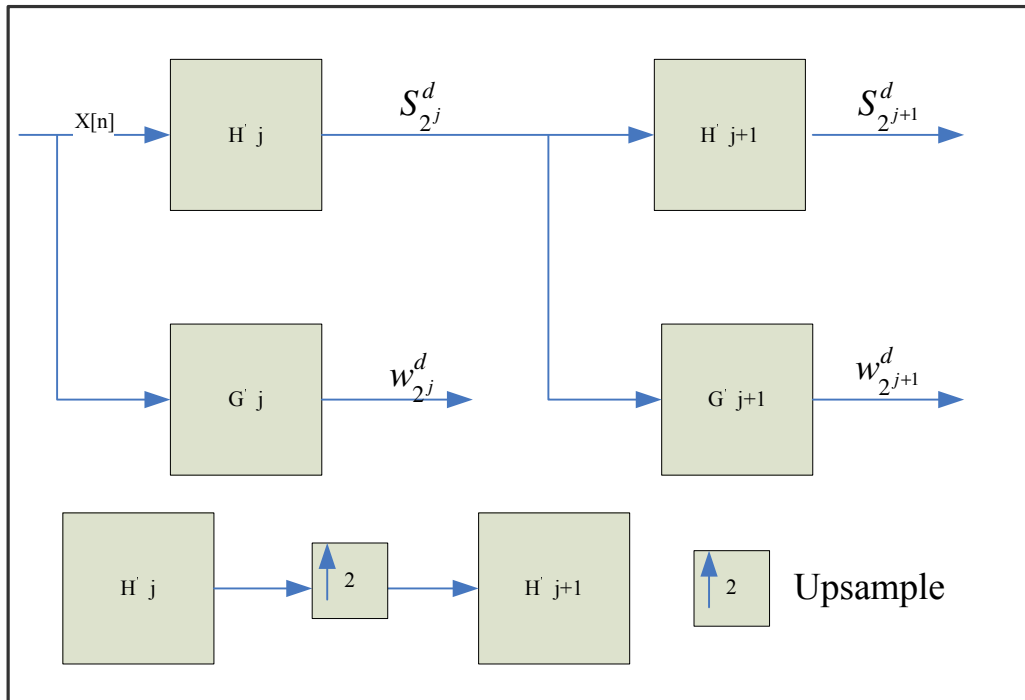


圖 12: 過完備小波的分解

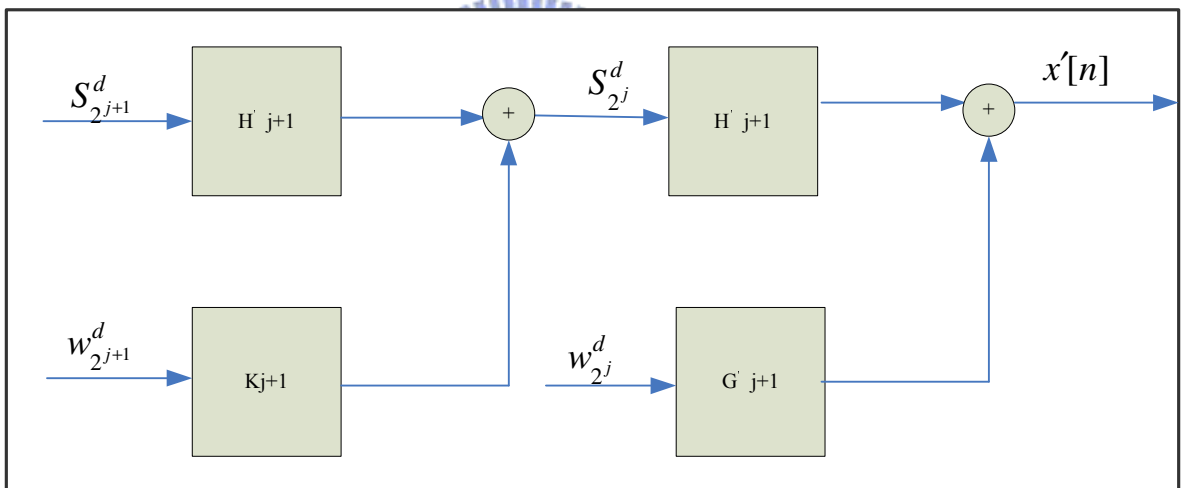


圖 13: 過完備小波的合成