

國立交通大學

多媒體工程研究所

碩士論文

果蠅腦 warping 及標準腦建立之研究



Brain warping and building template brain for Drosophila

研究生：陳耿郁

指導教授：荊宇泰、林文杰 教授

中華民國九十八年十一月

果蠅腦 warping 及標準腦建立之研究
Brain warping and building template brain for Drosophila

研究生：陳耿郁

Student：Keng-Yu Chen

指導教授：林文杰

Advisor：Wen-Chieh Lin

國立交通大學

多媒體工程研究所

碩士論文



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

November 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年十一月

果蠅腦 warping 與標準腦建立之研究

研究生：陳耿郁

指導教授：林文杰 博士

國立交通大學

多媒體工程研究所

摘要



我們的研究目的是把一組果蠅腦形變到另一組果蠅腦中。在此論文裡，首先對果蠅腦定義一個座標系統。基於此座標系統，我們提出兩種不同方法的內插，根據此內插法來實作形變。接著，利用形變的方法來定義一組標準果蠅腦的建立流程，並且帶入神經組織的形變至標準腦。

Brain Warping and building template brain for *Drosophila*

Student: Keng-Yu Chen

Advisor: Dr. Wen-Chieh Lin

Institute of Multimedia Engineering
National Chiao Tung University



Our goal is warping one *Drosophila* brain into another one. In this thesis, we first define a coordinate system for brain of *Drosophila*. Base on this coordinate system, we demonstrate two different original interpolation methods , and use these methods for warping . Then we define a procedure of building a template brain of *Drosophila* by warping and introduce the warping of neuron to template brain.

誌 謝

本論文能得以完成，首先要感謝我的指導老師荊宇泰老師，在這近一年多的時間內細心的指導也耐心的傳授專業知識，在我的學業上有不少的進步，才能順利完成此論文。也由於平常會關心大家的生活，因此使得不只在學業上的增進，在其它部份也因老師的言談中略有所得不少東西。也同時感謝參與口試的委員江安世教授、林文杰教授、陳永昇教授在百忙之中抽空參加，建議我在論文及報告上的不足及知識，使得本論文加以完整。

在此也感謝實驗室的秉璋、昌杰學長。在我面對問題有困難時，得以幫助我在這方面的不足，提供我新的觀點、想法及技術上的分析，收獲良多。

感謝過去 503 的所有成員們：大喬、小喬、佳昇、黑鬼、進錕、mog、彥龍。那一年是大家所懷念的回憶。如果少了大家在課業、生活、感情、聯誼等等的不及備載的共同鼓勵。我相信我在交大的生活是枯燥乏味的。

感謝一起打拼畢業的細菌人、淳哥、宗澤、家樺、懷元、書豪、還有賊晨。雖然沒有一起畢業，但大家一起努力的過程，我們都知道那是辛苦的，我們也做到了。

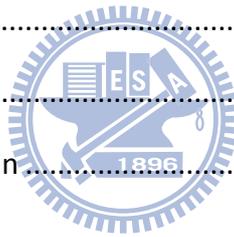
感謝實驗室的所有學弟妹們，多謝大家害我在請客要花很多錢在你們身上。雖然沒有學長陪你們聊天解悶，至少那段時間是大家共同創造的，也很珍惜。

最後感謝我的家人，感謝女友，感謝過去有支持我的人。在我沒有後顧之憂的情況下得以順利完成學業。

TABLE OF CONTENTS

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
圖目錄	vi
CHAPTER 1 INTRODUCTION	1
1.1 Introduction	1
CHAPTER 2 COORDINATE SYSTEM FOR DROSOPHILA.....	2
2.1 Moments.....	2
2.2 Principle Axes	3
2.3 Bounding Box	5
2.4 Find ellipsoid body center.....	6
CHAPTER 3 MORPHING.....	8
3.1 3D Morphing	8
3.1.1 Initialization.....	8
3.1.2 Triangulation	9
Delaunay Triangulation	9
3.1.3 Warping.....	11
3.2 Introduction interpolation method.....	13
3.3 Interpolation 1	14
Flow chart	18
3.4 Interpolation 2	19
Flow chart	27

3D Morphing flow chart.....	28
3.5 Result.....	29
3.5.1 Interpolation 1	29
3.5.2 Interpolation 2	30
3.5.3 Bad result	31
CHAPTER 4 TEMPLATE BRAIN	32
4.1 Introduction of affine transformation.....	32
4.2 Introduction of nonrigid transformation	33
4.3 Template brain	35
Template brain flow chart.....	37
4.4 Neuron warp	38
4.5 Result.....	39
4.5.1 Affine.....	39
4.5.2 Template brain.....	41
4.5.3 Neuron warp	44
CHAPTER 5 USER INTERFACE FOR QT	48
5.1 QT introduction.....	48
5.2 User interface for QT.....	49
CHAPTER 6 CONCLUSIONS	53
REFERENCE.....	54



LIST OF FIGURES

圖 2-1	物體 centroid 與軸分佈	3
圖 2-2	三維空間的物體主軸	4
圖 2-3	物體旋轉後的新座標系統	4
圖 2-4	沿著 P1 找兩側的邊界	5
圖 2-5	沿著 P3 找兩側的邊界	5
圖 2-6	Bounding box 與 Bounding points	6
圖 2-7	最小封閉的環狀體	7
圖 3-1	破碎、不連續的果蠅腦	9
圖 3-2	Delaunay Triangulation 示意圖 1	9
圖 3-3	Delaunay Triangulation 示意圖 2	10
圖 3-4	Delaunay Triangulation 示意圖 3	10
圖 3-5	Delaunay Triangulation 後 warping 的果蠅腦	10
圖 3-6	Reference 與 target 的對應關係	11
圖 3-7	Reference 與 target 對應關係	14
圖 3-8	從 target 填值至 reference	14
圖 3-9	三角錐的分割	15
圖 3-10	選取其中一角錐計算體積	15
圖 3-11	第一組 cell 的對應	16
圖 3-12	第二組 cell 的對應	16
圖 3-13	第三組 cell 的對應	17
圖 3-14	第四組 cell 的對應	17
圖 3-15	四組選出一組 cell 繼續執行下一次疊代	18
圖 3-16	Interpolation 1 method 流程圖	18

圖 3-17	Reference 與 target 對應關係	19
圖 3-18	從 target 填值至 reference	19
圖 3-19	三角錐的分割	20
圖 3-20	選取其中一角錐計算體積	20
圖 3-21	第一組 cell 的對應	21
圖 3-22	第二組 cell 的對應	21
圖 3-23	第三組 cell 的對應	22
圖 3-24	第四組 cell 的對應	22
圖 3-25	第五組 cell 的對應	23
圖 3-26	第六組 cell 的對應	23
圖 3-27	第七組 cell 的對應	24
圖 3-28	第八組 cell 的對應	24
圖 3-29	第九組 cell 的對應	25
圖 3-30	第十組 cell 的對應	25
圖 3-31	第十一組 cell 的對應	26
圖 3-32	第十二組 cell 的對應	26
圖 3-33	十二組選出一組 cell 繼續執行下一次疊代	27
圖 3-34	Interpolation 2 method 流程圖	27
圖 3-35	3D Morphing flow chart	28
圖 3-36	檔案名稱為 1200 與 1500 的果蠅腦資訊	29
圖 3-37	1500 warp 至 1200 與 1200 warp 至 1500	29
圖 3-38	檔案名稱為 12001 與 12002 的果蠅腦資訊	30
圖 3-39	12002 warp 至 12001 與 12001 warp 至 12002	30
圖 3-40	Bad image1	31
圖 3-41	Bad image2	31
圖 4-1	Reference file 與 target file	35



圖 4-2	一次 affine 與二次 affine.....	35
圖 4-3	原始檔與 warp 後的結果.....	36
圖 4-4	標準腦的建立流程.....	37
圖 4-5	原始神經與 warp 後的結果.....	38
圖 4-6	檔案編號 12 與 6.....	39
圖 4-7	檔案編號 6 warp 至 12 與 12 warp 至 6.....	39
圖 4-8	檔案編號 34 與 41.....	40
圖 4-9	檔案編號 41 warp 至 34 與 34 warp 至 41.....	40
圖 4-10	三十四組果蠅腦的資訊.....	42
圖 4-11	標準果蠅腦的資訊.....	43
圖 4-12	Template brain.....	43
圖 4-13	檔案編號 1200 warp 至 template.....	44
圖 4-14	檔案編號 1500 warp 至 template.....	45
圖 4-15	檔案編號 12001 warp 至 template.....	45
圖 4-16	檔案編號 12002warp 至 template.....	46
圖 4-17	Four neuron and template brain.....	46
圖 4-18	Four neuron and template brain.....	47
圖 5-1	QT 開發架構.....	48
圖 5-2	讀取副檔名為 .ipt、.hdr、.img 的檔案.....	49
圖 5-3	處理 warp 的相關檔案.....	50
圖 5-4	處理 affine 與 mask 的部份.....	51
圖 5-5	處理 nonrigid 的部份.....	52

第一章 Introduction

● 1.1 : Introduction

腦神經科學的研究範疇，長久以來是科學家們不停探索的奧妙領域，人類為萬物之靈，人類的大腦更是一個極為複雜的構造，能夠深層探索大腦的能力並且開發其無窮的潛力一般相信會有足以改變人類發展的力量。由於人類腦部龐大的繁複結構且因道德、效率與成本眾多因素，不易研究。因此尋找一個好的且能比擬人類的模式生物來研究有其必要性。

在眾多種類的生物裡，做為研究的諸如：老鼠、果蠅(*Drosophila*)、兔子、線蟲、大腸桿菌等。其中以果蠅最符合經濟效應，因果蠅容易繁殖且腦部結構易於研究，和人類一樣具有學習、記憶等能力。在生命週期方面，果蠅的世代交替約為十日，因此觀察果蠅生命較有效率。此外，人類腦神經方面疾病的基因也可在果蠅中發現，如阿茲海默症等。綜觀所有因素，利用果蠅來當作模式生物培養和觀察，對於解開大腦奧秘的腦神經科學上有其相當的優勢。

在觀察果蠅的腦神經時，通常會透過各種染色的方式顯示欲觀察的區域。通常是每次實驗犧牲一隻果蠅，觀察一組神經。我們會希望所有染色的神經都可以放在一個果蠅腦之中以供觀察。由於果蠅腦部的大小差異不一，再加上人工顯微鏡取像時所受的外在因素影響，所以將多組的果蠅腦神經影像放進同一組果蠅腦中，是一個高難度的 warping 問題。本文將提出果蠅腦 warping 的方法。

接著為了取得不同果蠅腦的神經組織解讀並相互比較，我們需要先建立一個標準的果蠅，其中包含了標準的腦外廓，標準腦神經組織的形狀與定位，但沒有任意兩個果蠅腦是完全相同的，由單一果蠅腦當作我們的標準腦並不客觀，因此如何定義一個具有參考意義的標準腦成為我們值得研究的課題。

第二章 Coordinate system for Drosophila

● 2.1 : Moments

果蠅原始檔任何的處理前，約略有45°不等的傾斜，為了方便處理及觀察，必須對原始檔作適度的Rotation，使得果蠅腦旋轉至標準方向的新座標系統上。

首先Moments是一種測量物體本身質量如何在空間中的分佈。主要是用來測量voxels的分佈及描繪在空間中物體的外型及特徵[1][2][3]。在統計學上，多個維度物體的外型、形狀常利用它的特徵來達到具體描繪，如principal axes，因此利用它來描述果蠅腦在3D空間裡的影像外型。

(i, j, k) - moments是3D影像並定義成：

$$m_{ijk} = \sum_{x,y,z} x^i y^j z^k f(x, y, z)$$

其中 $f(x, y, z)$ 為灰階值 $= \begin{cases} 1, & \text{if } (x, y, z) \text{ is foreground voxel} \\ 0, & \text{otherwise} \end{cases}$

而 Order of moment m_{ijk} ，則定義成 $sum(i + j + k)$ ，如此就可以用 $(0,0,0)$ -moments 來描述物體：

$$m_{000} = \sum_{x,y,z} f(x, y, z)$$

當在計算物體的 moments 時，會因為物體的平移位置不同而有所不同，為了避免此問題，針對物體本身的 centroid 為中心開始計算。

$g = (g_x, g_y, g_z)$ 影像中心，first - order moments 的定義如下：

$$g_x = m_{100}/m_{000}$$

$$g_y = m_{010}/m_{000}$$

$$g_z = m_{001}/m_{000}$$

(i, j, k) -moments M_{ijk} of Image 將被定義成：

$$M_{ijk} = \sum_{x,y,z} (x - g_x)^i (y - g_y)^j (z - g_z)^k f(x, y, z)$$

如上計算 central moments，物體的質量中心就會被當作影像的原點。

● 2.2 : Principle axes

接著將果蠅腦的centroid(灰點)當作座標軸的原點，我們希望把果蠅腦的座標軸方向標準化如下圖所示：

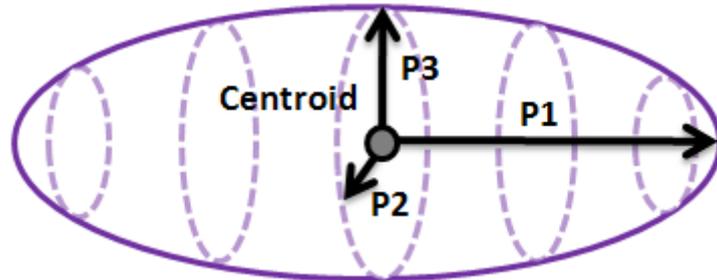


圖 2-1：物體centroid與軸分佈

其中P1為此物體最長軸，P3為最短軸，而P2為介於兩者長度之間的軸，此三軸相互兩兩正交。接著利用second – order central moments[1][2]的定義：

$$\mu_{ijk} = M_{ijk} = \sum_{x,y,z} (x - g_x)^i (y - g_y)^j (z - g_z)^k f(x, y, z)$$

可將此橢圓物體用一個慣性矩陣(inertia matrix)來表示，因此影像Image定義如下：

$$Image = \begin{bmatrix} \mu_{020} + \mu_{002} & -\mu_{110} & -\mu_{101} \\ -\mu_{110} & \mu_{200} + \mu_{002} & -\mu_{011} \\ -\mu_{101} & -\mu_{011} & \mu_{200} + \mu_{020} \end{bmatrix}$$

上述矩陣等價如下：

$$Image = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}$$

其中對角元素以慣性矩稱之，非對角元素以慣性積稱之

Image 矩陣是一個具有 symmetric 的性質，其 eigenvalue 必定是實數，並且可求得其對應的 eigenvector。再經由矩陣對角化之後，可以推導出一個矩陣 R 滿足下列方程式：(假定 $\lambda_1 > \lambda_2 > \lambda_3$)

$$R^T Image R = D = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

其中 R 為 eigenvector 排列，D 為 eigenvalue 排列

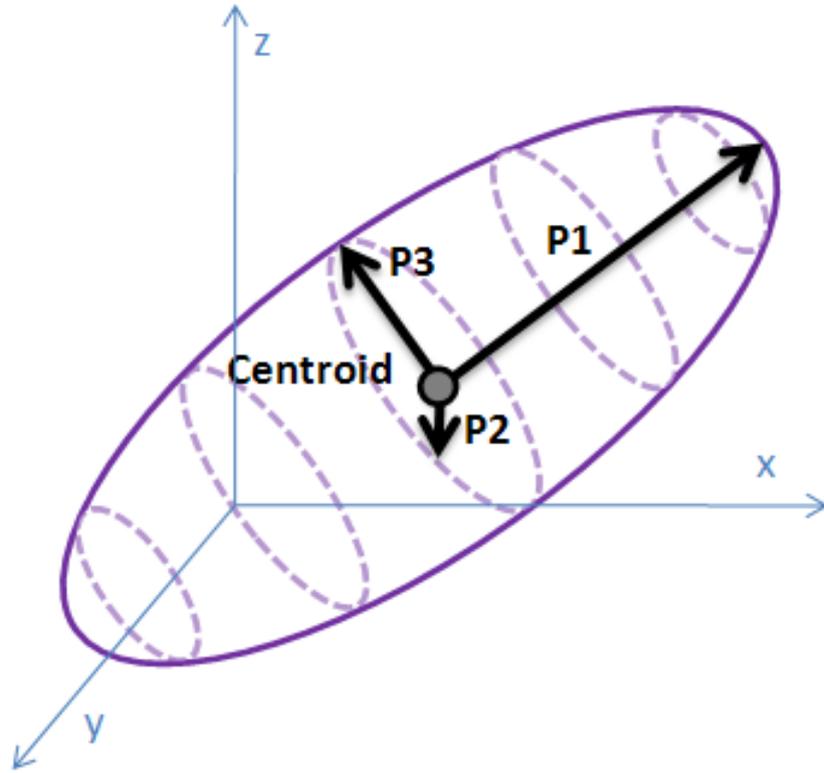


圖 2-2：三維空間的物體主軸

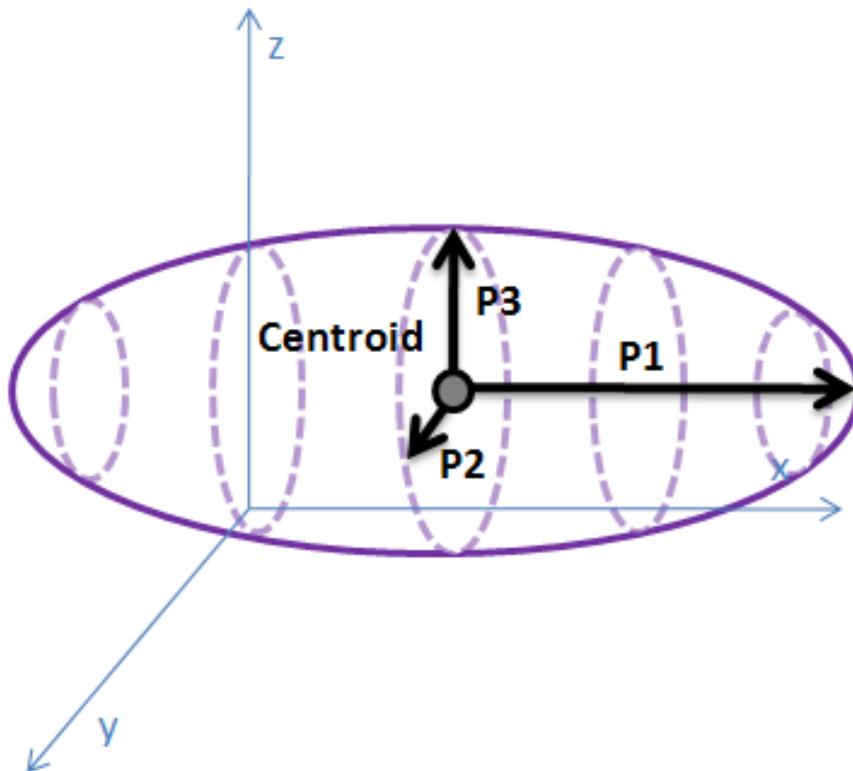


圖 2-3：物體旋轉後的新座標系統

● 2.3 : Bounding box

由於果蠅腦在3D影像主要的分佈於影像的中心點周圍，靠近影像邊緣的區域鮮少有資訊量存在。因此我們可以定義一組bounding box，其內容主要包覆整個果蠅腦大小的資訊量，並且記錄bounding point[4]的座標。

首先，利用second – order central moments找到的新主軸，分別沿著三個主軸，以主軸為法向量的切平面尋找3D影像是否有voxel值存在，並記錄存在最後一個voxel切平面的深度。如下兩張圖所示。

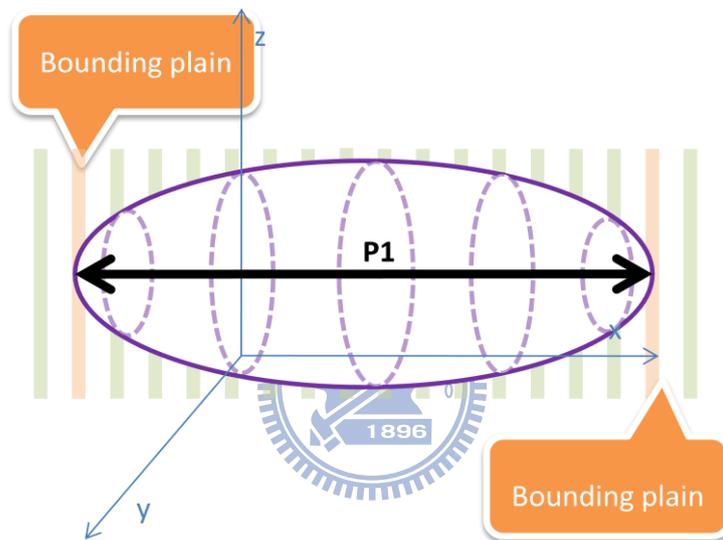


圖 2-4：沿著P1找兩側的邊界

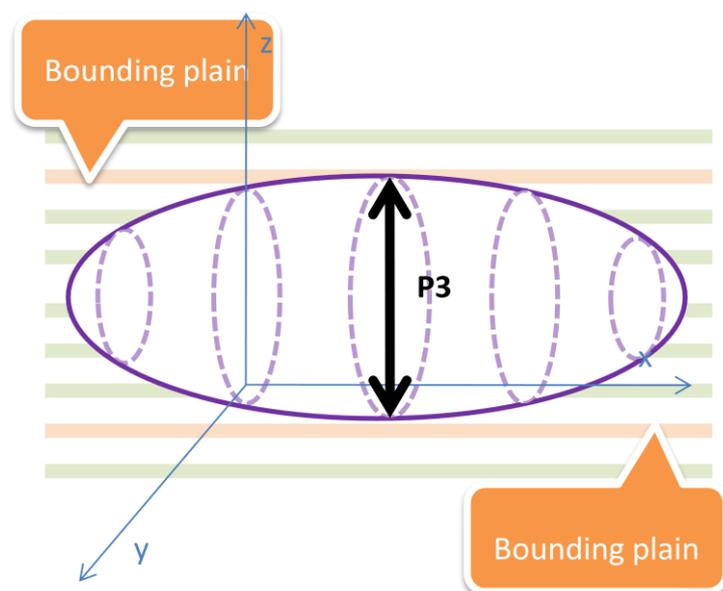


圖 2-5：沿著P3找兩側的邊界

以此類推，我們可以找到三個主軸皆有兩個切平面的資訊。最後會得到六個邊界平面，六個平面形成一個包覆果蠅腦的邊界盒，此邊界盒有八個頂點。利用左前方頂點與右後方頂點(紅點)表示此bounding box的資訊。Bounding point均內含三個維度的座標。如下圖所示。

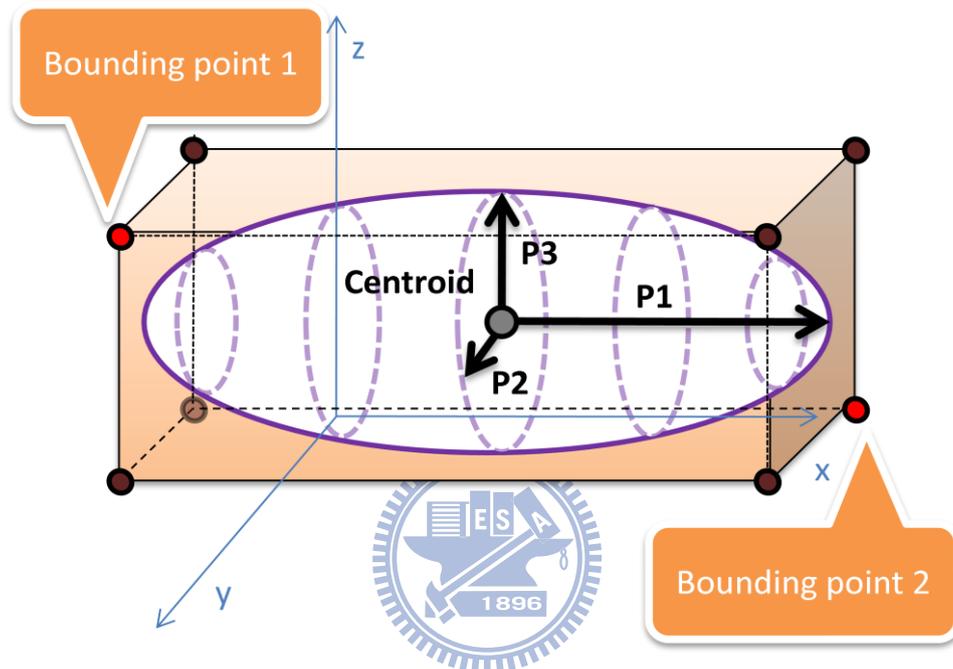


圖 2-6：Bounding box與Bounding points

● 2.4：Find ellipsoid body center

接著我們需尋找一個具代表性的中心點，果蠅腦裡的ellipsoid body，除了本身是球體，且具有左右半邊器官對稱的代表意義。由於ellipsoid body形體接近同球心的兩個球體，因此需尋找兩球體的球心位置，當作ellipsoid body的座標。

最早於1991年提出這樣概念於計算不同圓心的最小半徑與最小面積[5]。而後1999年提出尋找最小寬度的環狀球體[6]，如下圖所示：

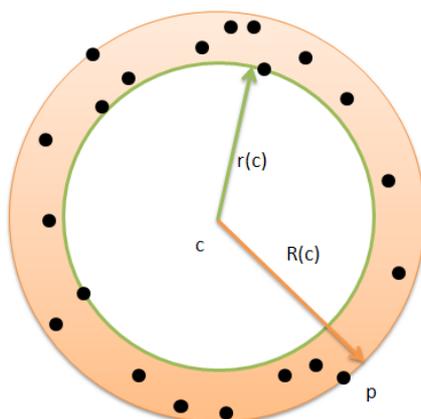


圖 2-7：最小封閉的環狀體

假定 S 是 n 個點的集合分佈在 d 個維度的歐式空間裡。 S 環狀球體的大小 (roundness) 的計算，相當於求得最小的球面體(或環狀面積於 R^2) 且包含 S 集合。換句話說，求 S 集合的一點 p 到同心圓 c 的最大距離 R 和最小的半徑 r ，計算公式如下：

$$\min_{c \in \mathbb{R}^d, r \in \mathbb{R}} \max_{p \in S} |d(p, c) - r|$$

for $c \in \mathbb{R}^d$ and $r, R \in \mathbb{R}$ with $0 \leq r \leq R$ ，定義環狀的球體為 $\mathcal{A}(c, r, R)$ 是一個封閉區間介於兩個同心球球心 c 半徑為 r 與 R 的球體間。而 $\mathcal{A}(c, r, R)$ 的寬度是 $R - r$ ，測量 S 點集環狀球體大小的問題，等價於計算一個外殼， $\mathcal{A}^*(S)$ 包含 S 點集的最小寬度。之後於 2000 年提出尋找球心 c 相當於求得 $R^2 - r^2$ 的最小值[7]。

首先給定點集 P 其中 $ma(P)$ 是最小環狀體並包含所有 P 的點集。例如， $ma(P)$ 為可退化集合， $ma(P) = \emptyset$ if $P = \emptyset$ and $ma(P) = \{p\}$ if $P = \{p\}$ ， S 是包含於 P 的最小子集，其中 $ma(S) = ma(P)$ 稱 S 為支援集 (support set)， S 裡的所有點為支援點，且都落在 $ma(S)$ 的邊界上，因而 S 並非要唯一。最終的最小環的結束條件為下：

- Min annulus 包含所有定義的點集 P
- Min annulus 是最小 annulus 包含它的 support set S
- Support set S 為最小 i.e. no support point is redundant

利用 `Class(CGAL::Min_annulus_d<Traits>)` 可找到最小環狀體的同心圓，即為 ellipsoid body center。

第三章 Morphing

● 3.1 : 3D Morphing

在[8][9][10]這篇論文所提到的方法，是對針對3D果蠅腦影像有興趣的部份，給定特徵點，接著再利用這些特徵點，定義出鄰近點的幾何關係來實作Warp。其結果在果蠅腦外殼的部份對應的較不準確，其原因在於只給定特徵點去作Morphing，原始的兩組果蠅腦本身的方向與大小就有些差距，所以變形後的結果，位於特徵點鄰近的部份準確度較高，遠離特徵點較遠的點差距就相對的大。

有鑑於此方法的缺點，3D 果蠅腦影像在經過下列的方法處理後，再做特徵點對應的 Morphing 結果會改善許多，不僅利用特徵點對應讓有興趣的部份對得很精確，外殼的對應能夠維持準確度。主要作法分成下列三個部份：

● 3.1.1 : Initialization



首先，我們選擇兩組果蠅腦，一組當作 reference、一組當作 target。接著分別對兩組果蠅腦設定特徵點，且需要一一對應。特徵點的選擇由人工決定，一般來說選擇的方式會挑選具有明顯的特徵、器官或者較具代表性的位置。最後選取內部特徵點的數量約為 15~19 個，外部特徵點數量即為 3D 影像的八個頂點。

● 3.1.2 : Triangulation

接著要利用這些特徵點來產生多組的三角錐(Tetrahedron，本文後皆總稱 cell)，此 3D 影像即被這些 cell 所劃分成多個區域(如：特徵點 23 個， cell 數 85~105 個不等)。產生的 cell 就可以作 reference cell 與 target cell 的對應。

而 cell 產生的形體即會影響 reference cell 與 target cell 對應結果的好壞。一般 cell 若是呈現明顯的銳角，在之後 cell 間的對應的結果會產生不連續、破碎的影像，我們稱 bad triangle。如下圖所示。

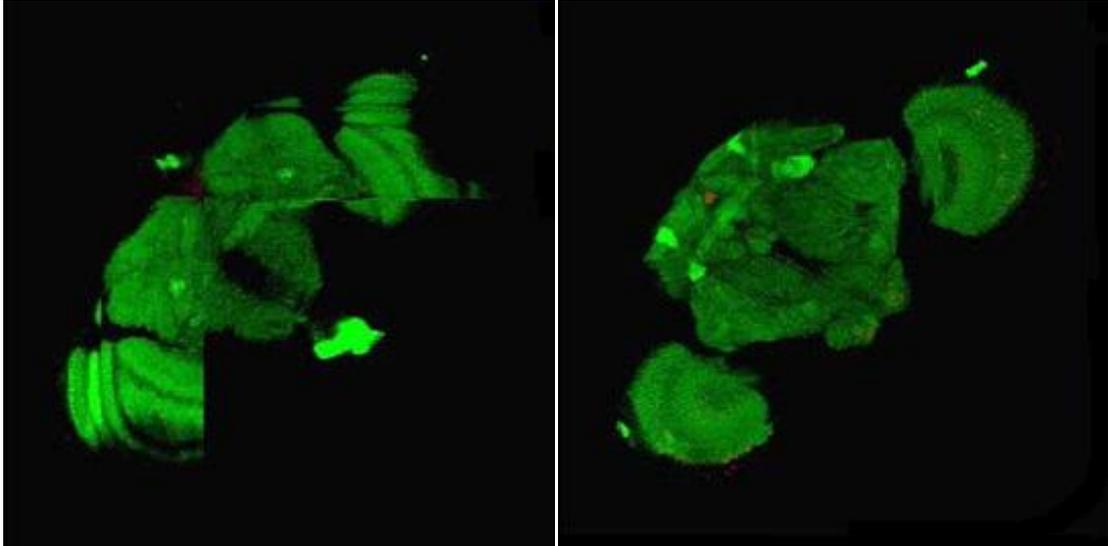


圖 3-1：破碎、不連續的果蠅腦

因此，為了解決bad triangle的問題，採用Delaunay Triangulation這個演算法來處理上述的問題。Delaunay Triangulation的優點在於：

一、使最小角為最大化，結果最接近於 regular triangulation。

二、具有唯一性(若任意四點不共圓)。

此方法是檢查每一個三角形的外接圓是否有包含其它的點。當包含其它點時，則破壞此三角形的結構。若不含其它點時，則停止檢查此三角形。最後，所有的三角形都滿足上述條件時，即完成。

Delaunay Triangulation：

一、依序將特徵點插入已經完成Delaunay Triangulation裡，開始進行外接圓的檢測，若三角形的外接圓內包含其它點，則必需破壞三角形。如下圖所示。



圖 3-2：Delaunay Triangulation 示意圖1

二、把包含紅點的外接圓三角形移除，接著記錄不需移除與需移除的點與邊。如此一來就可以得到點與邊需要重新三角化，而不至於破壞已經完成的Delaunay Triangle。如下圖所示。

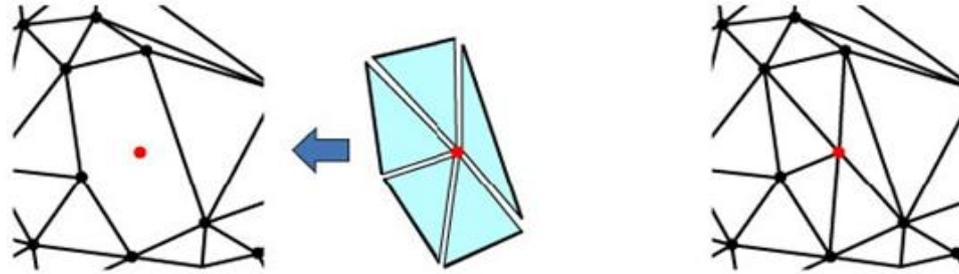


圖 3-3：Delaunay Triangulation 示意圖2

三、接著把所得到的點與紅點繼續作Delaunay Triangulation，產生新的三角形，因此可維持Delaunay Triangulation的特性，完成即停止檢查。如下圖所示。

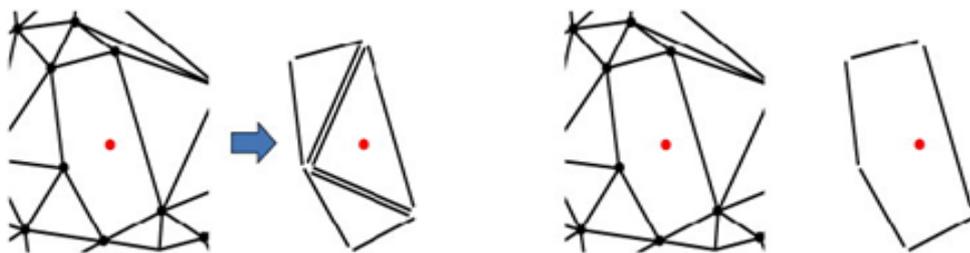


圖 3-4：Delaunay Triangulation 示意圖3

上述的動作完成後，可以去除含有明顯銳角的cell，使得cell都擁有regular triangulation性質。對於之後的warping，可以得到最佳的3D影像。如下圖所示。

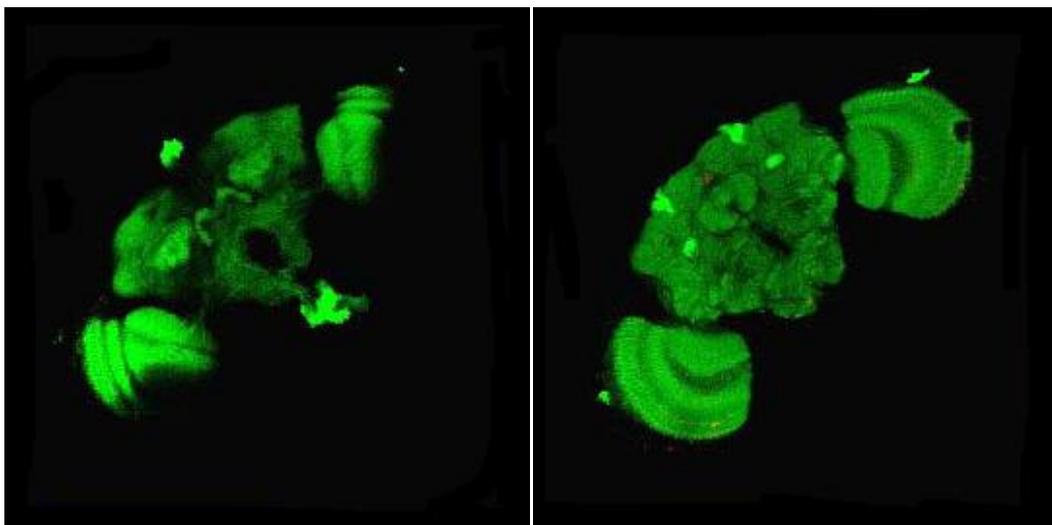


圖 3-5：Delaunay Triangulation後warping的果蠅腦

● 3.1.3 : Warping

在得到所有 cell 之後，由於先前有特徵點一對一對應，因此可以從 reference cell 找到相對應的 target cell。接著再定義出 reference cell 與 target cell 兩者間的對應關係 (Transfer function)，最後將 target cell 的值填入 reference cell 裡，即完成填值的動作。

◇ 採用 Reverse warping 是預防在進行填顏色值的時候，某些點會因為沒有對應的關係而造成此點沒有任何的顏色值，變成破碎。

Transfer function :

一、定義 cell 間的對應關係

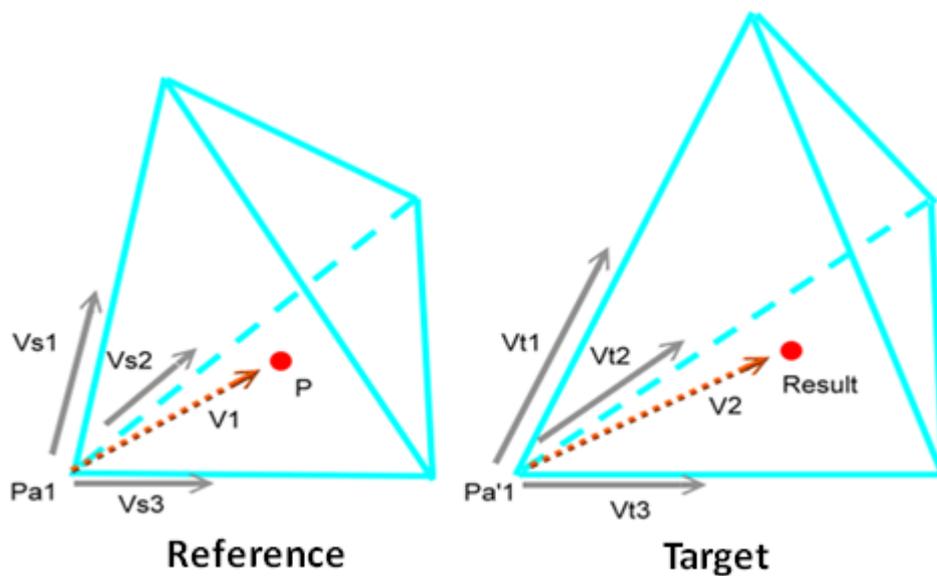


圖 3-6 : Reference 與 target 的對應關係

在 reference cell，需描述點 P 在 reference cell 裡的幾何關係，因此利用其中任一頂點及頂點鄰近的三個邊來描述。頂點 Pa1 與點 P 的向量 $\vec{V}_1 = (x_s, y_s, z_s)$ 係由三個向量 \vec{V}_{s1} 、 \vec{V}_{s2} 、 \vec{V}_{s3} 所構成的。利用此關係可以描述如下：

$$\vec{V}_1 = a \vec{V}_{s1} + b \vec{V}_{s2} + c \vec{V}_{s3} \quad \text{其中} \quad \begin{cases} \vec{V}_{s1} = (x_{s1}, y_{s1}, z_{s1}) \\ \vec{V}_{s2} = (x_{s2}, y_{s2}, z_{s2}) \\ \vec{V}_{s3} = (x_{s3}, y_{s3}, z_{s3}) \end{cases}$$

整理上式，得到一組線性方程式，並且求得未知係數 a 、 b 、 c 。

$$\begin{cases} x_s = a x_{s1} + b x_{s2} + c x_{s3} \\ y_s = a y_{s1} + b y_{s2} + c y_{s3} \\ z_s = a z_{s1} + b z_{s2} + c z_{s3} \end{cases}$$

在target cell，相同的原理下可以描述點Result在target cell的幾何關係。因此頂點Pa'1與點Result的向量 $\overline{V}_2 = (x_t, y_t, z_t)$ 係由三個向量 \overline{V}_{t1} 、 \overline{V}_{t2} 、 \overline{V}_{t3} 所構成的。利用此關係可以描述如下：

$$\overline{V}_2 = a \overline{V}_{t1} + b \overline{V}_{t2} + c \overline{V}_{t3} \quad \text{其中} \quad \begin{cases} \overline{V}_{t1} = (x_{t1}, y_{t1}, z_{t1}) \\ \overline{V}_{t2} = (x_{t2}, y_{t2}, z_{t2}) \\ \overline{V}_{t3} = (x_{t3}, y_{t3}, z_{t3}) \end{cases}$$

整理上式，得到一組線性方程式，並且求得 x_t 、 y_t 、 z_t ，即向量值。

$$\begin{cases} x_t = a x_{t1} + b x_{t2} + c x_{t3} \\ y_t = a y_{t1} + b y_{t2} + c y_{t3} \\ z_t = a z_{t1} + b z_{t2} + c z_{t3} \end{cases}$$

計算出 \overline{V}_2 後，即可以求得點Result在target cell的相對關係。接著每一個cell都有四個頂點，因此再次使用上述的方法，會得到四組Result的空間座標，分別為 $Result_1$ 、 $Result_2$ 、 $Result_3$ 、 $Result_4$ 。



二、定義權重值

因每個頂點都是特徵點，這些頂點的周遭變動必須是最少的，也就是頂點鄰近的點位，其權重值為是最大。根據這樣的原則，可定義出權重值：

$$weight_i = 1/(1 + d_i^2)$$

其中 d_i 表示點P到reference cell第i個頂點的距離， $i = 1、2、3、4$

因此最後對應的點Result：

$$Result = \sum_{i=1}^4 Result_i * weight_i / weight \quad \text{其中} \quad weight = \sum_{i=1}^4 weight_i$$

接著採用reverse warping，從上述方法中reference cell的任一點找出對應target cell的點，然後再把找到對應點的顏色值填入reference cell的點。因此，3D影像上的每一點(voxel)逐一進行填值的動作，即完成最後的處理。

● 3.2 : Introduction interpolation method

由於3D影像的Morphing成像好壞在於三角化所使用的方法及Warping所對應的Transfer function。先前三角化的方法所產生的三角錐因有明顯的銳角，使得Warping後的影像不連續、破碎，導致失真而不適用。

因而有Delaunay Triangulation的演算法來解決銳角三角錐的問題，使得所有的三角錐皆變成regular triangle，來幫助Warping後的影像不會失真。

因此，相對的Warping定義的Transfer function變成3D影像的Morphing最主要考量的部份。Warping的好壞在於是否能夠將影像準確的轉移至另一影像，及內部細節的精確度。能夠使得兩組影像完全的重疊，甚至使人眼無法辨別，這是我們所設想達到的目標。

Transfer function :

本節在於Transfer function的重新定義reference與target相對應的關係。

首先，選擇reference cell內部的任一點(本文採用四個頂點座標的中心)或多點。同樣的，target cell選擇的方式如前。因頂點是特徵點，所以reference cell與target cell本身的頂點已經一一相對應，接著把對應好的頂點與前述選擇的點顏色值從target cell填入至reference cell。

而後連起點與頂點之間的線段，在三角錐的內部形成數個小三角錐，從中選取任一個小三角錐，繼續進行上述所描述的步驟。反覆使用遞迴的方式，因此必須設定一個停止條件，判斷的依據是計算任一三角錐的體積是否小於threshold，如是則停止，如否則繼續進行三角錐的切割。

最後為了符合reverse warping，需判斷reference cell的體積是否小於threshold。若採用判斷target cell的體積，會造成某些點(voxel)沒有相對應的關係而無法填入顏色值。

如上敘述，我們提出兩種類似概念的方法。接著介紹這兩種方法的方式。

● 3.3 : Interpolation 1

一、首先，我們擁有reference cell與target cell的特徵點(灰點)對應，接著在兩組cell內部皆選擇一點，此點為cell的四個頂點的中點座標(紅點)。如下圖所示。

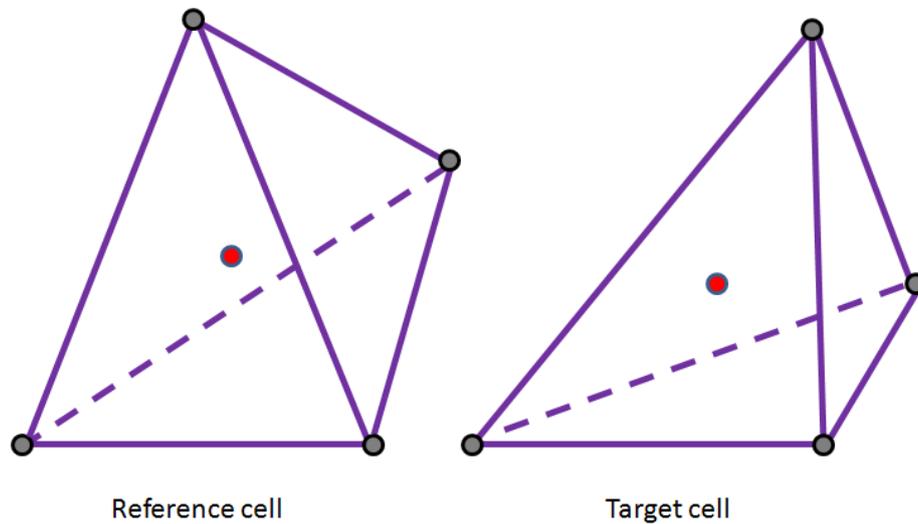


圖 3-7 : Reference與target的對應關係

二、有特徵點(灰點)與中心點(紅點)的對應，接著可以把target cell點的颜色值填到reference cell的點，因此一次可以填入5個點。如下圖所示。

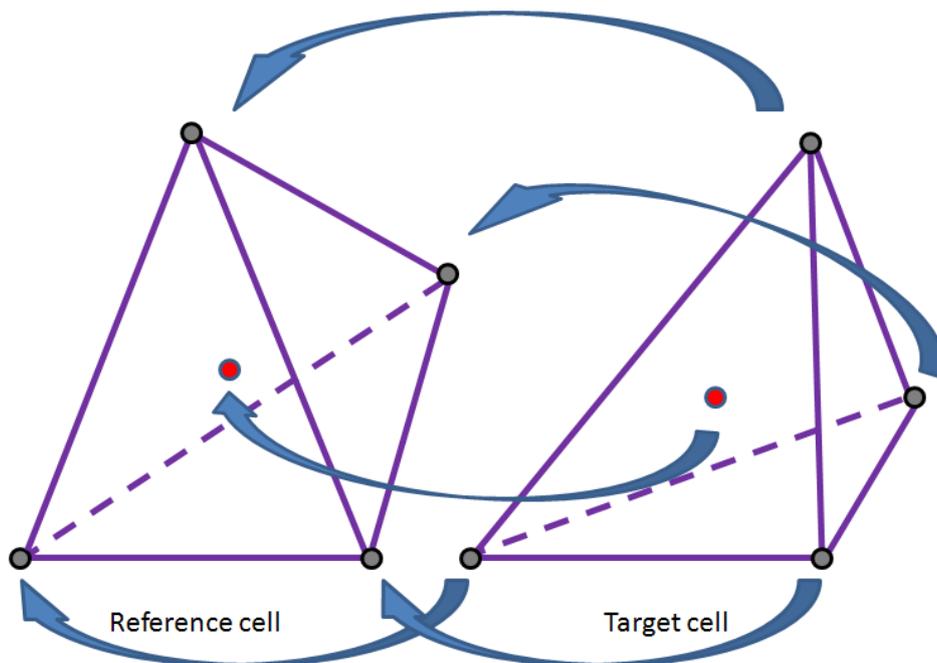


圖 3-8 : 從target填值至reference

三、把中心點(紅點)與cell的四個頂點(灰點)連接成線段。如下圖所示。

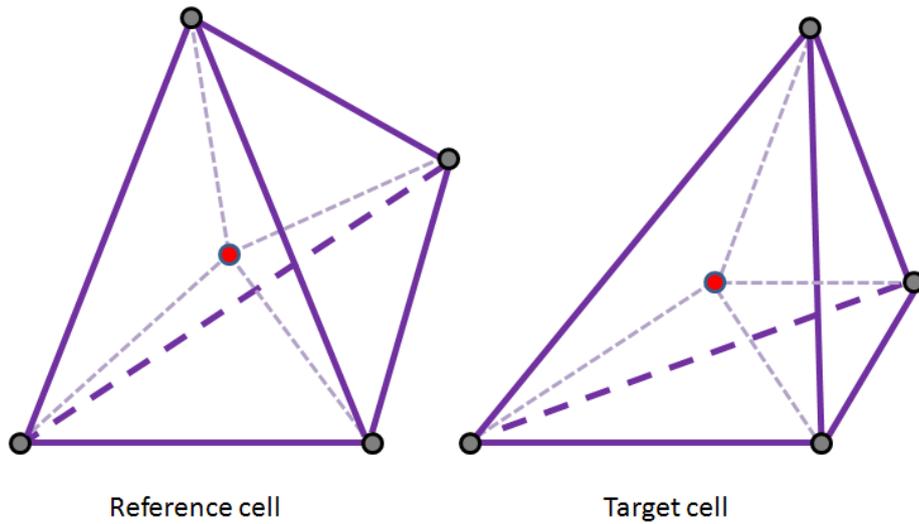


圖 3-9：三角錐的分割

四、任意三頂點與中心點的組合可以再產生四個cell，從reference cell選一個cell且先判斷體積。假定cell的四個頂點為 p_1 、 p_2 、 p_3 、 p_4 ，體積計算如下：

$$\begin{aligned} \vec{V}_1 &= p_2 - p_1 & \vec{V}_2 &= p_3 - p_1 \\ \vec{V}_3 &= p_4 - p_1 & \vec{V}_4 &= \vec{V}_2 \wedge \vec{V}_3 \end{aligned} \quad \begin{array}{l} \text{Operator } \wedge \text{ is cross} \\ \text{Operator } \circ \text{ is dot} \end{array}$$

體積 $V = |\vec{V}_1 \circ \vec{V}_4| / 6$

選定 $threshold$ ，判斷體積 V $\left\{ \begin{array}{ll} \text{繼續步驟六} & \text{if } V > threshold \\ \text{結束} & \text{if } V < threshold \end{array} \right.$

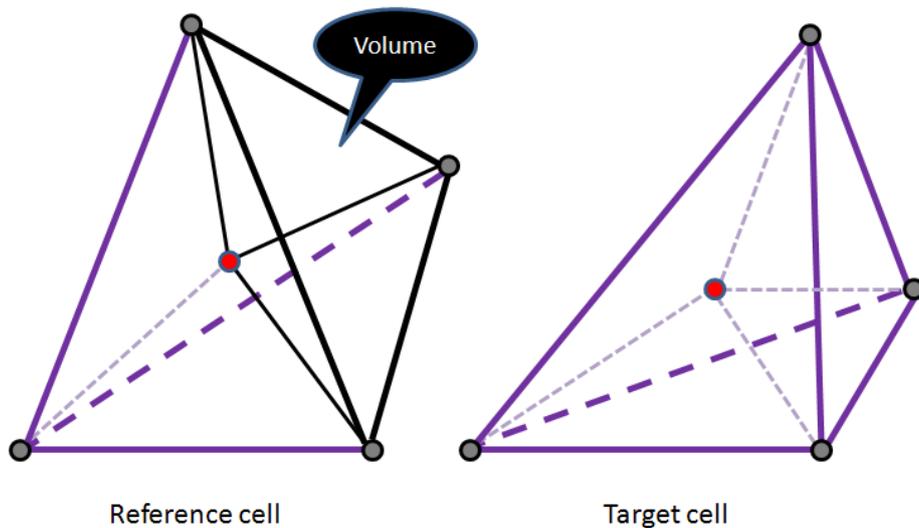


圖 3-10：選取其中一角錐計算體積

五、從上述可以得到四組新的cell，為了方便描述，下圖利用黑色線來描述產生的新cell：

第一組為中心點與右方三個頂點構成的cell。

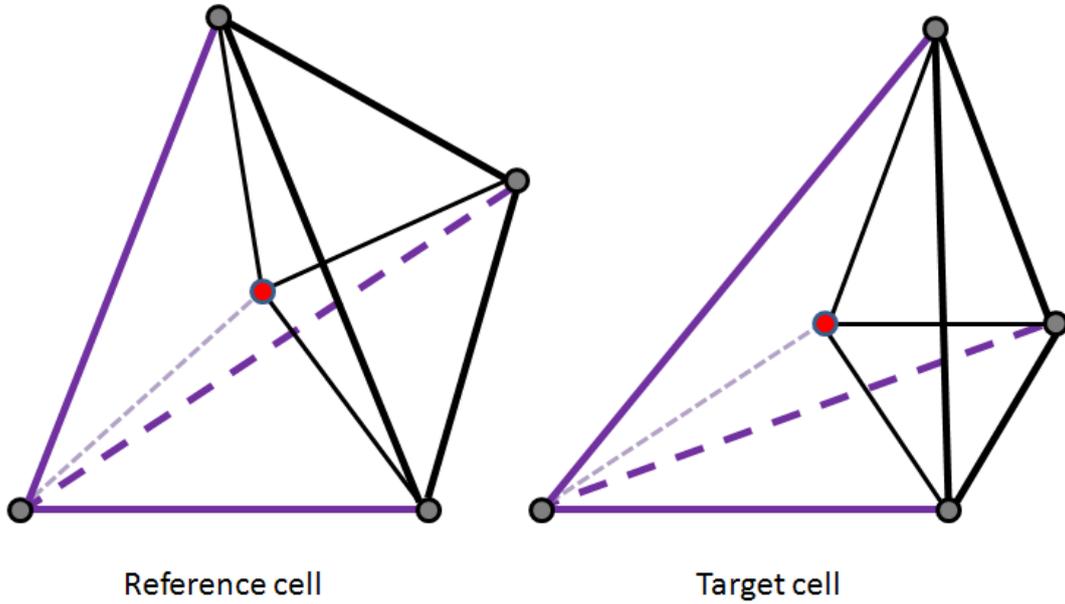


圖 3-11：第一組cell的對應

第二組為中心點與下方三個頂點構成的cell。

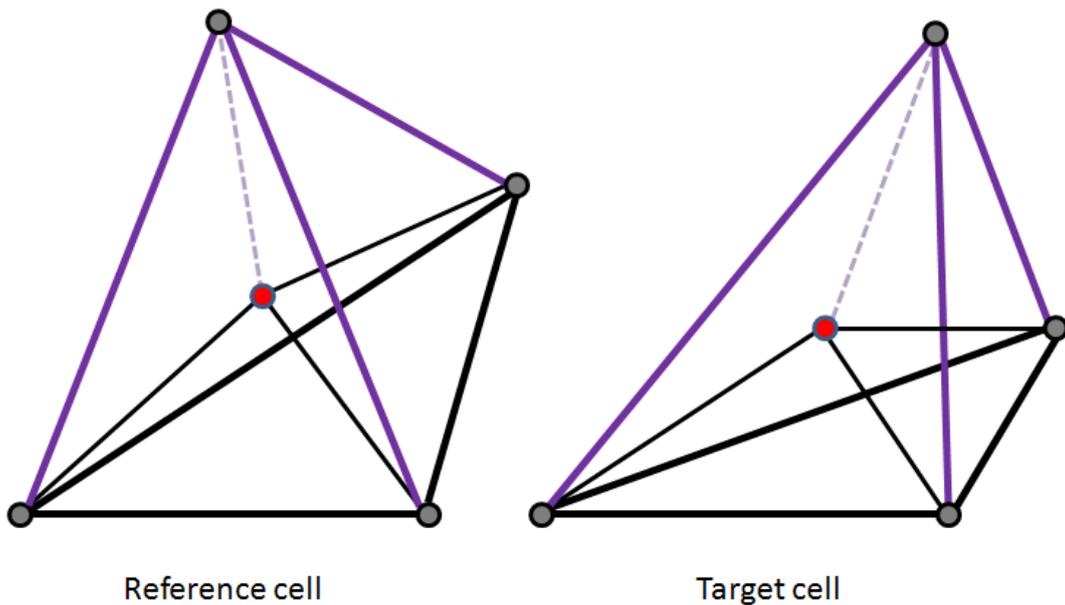


圖 3-12：第二組cell的對應

第三組為中心點與後方三個頂點構成的cell。

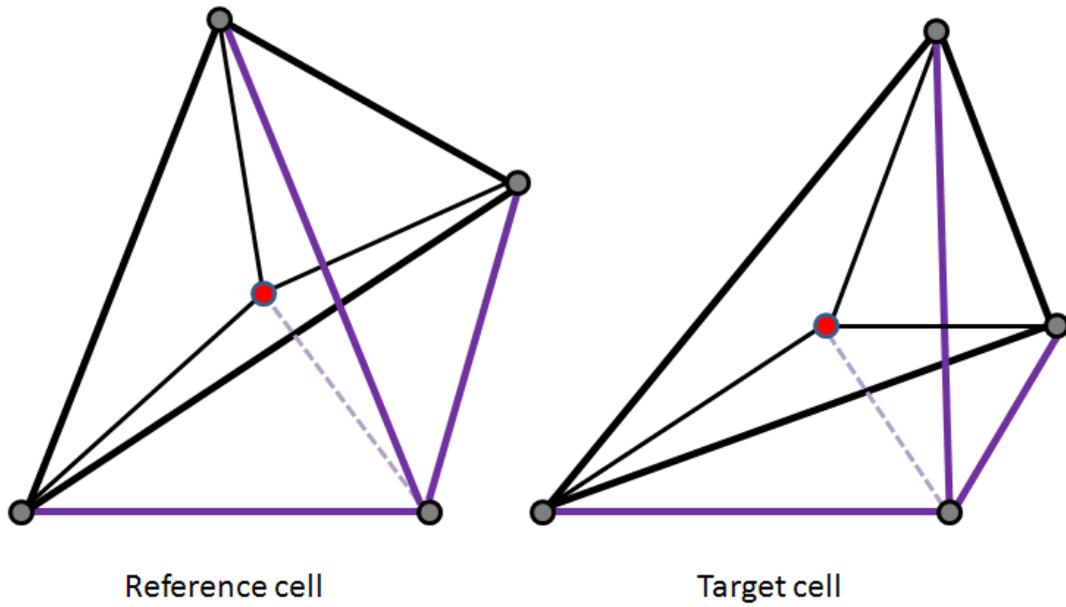


圖 3-13：第三組cell的對應



第四組為中心點與前方三個頂點構成的cell。

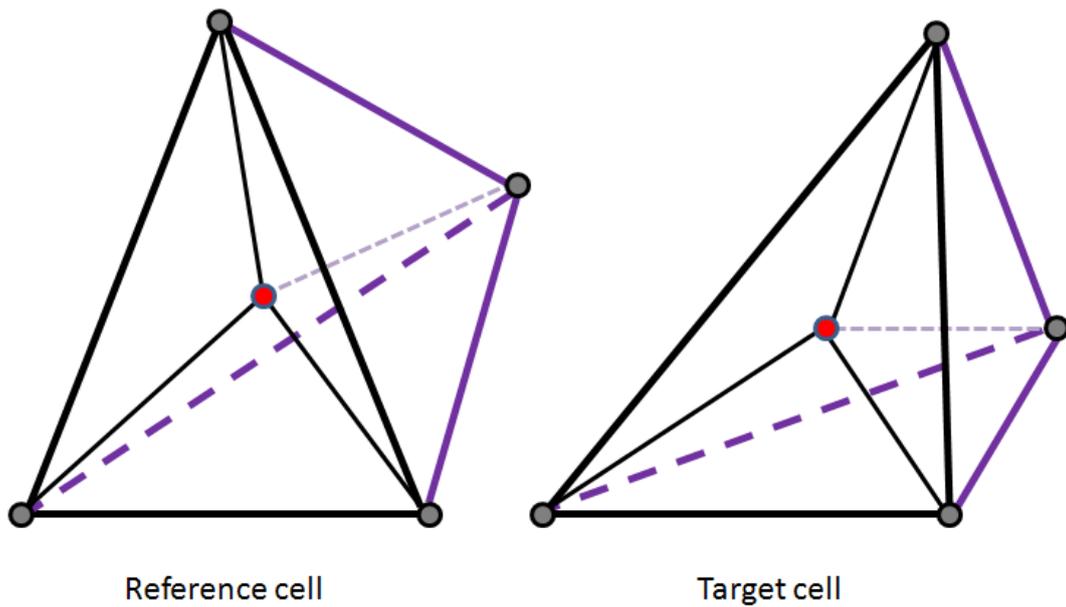


圖 3-14：第四組cell的對應

六、從四個cell中，選出第一組cell重新執行第一個步驟，如下圖所示，直到完成停止條件。而後再選出第二組cell重新執行第一個步驟...依序下去。等所有cell達到停止條件，即完成最後的工作。

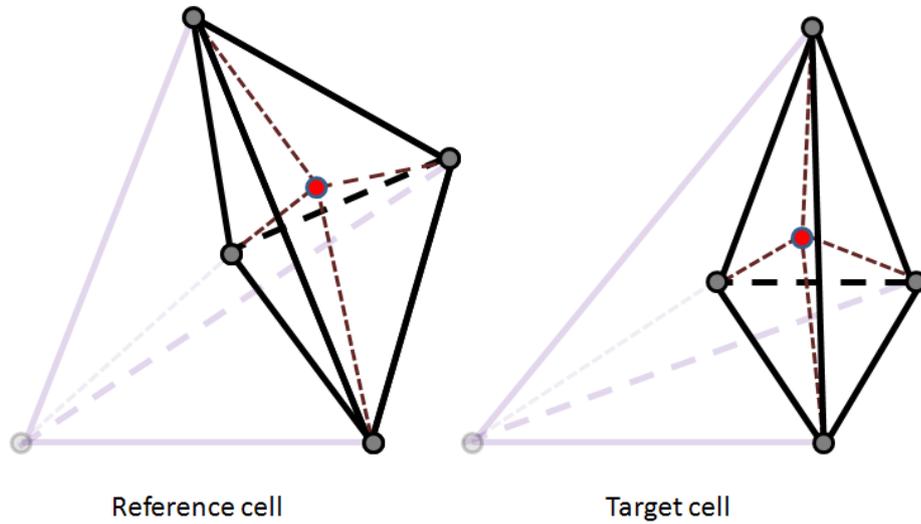


圖 3-15：四組選出一組cell繼續執行下一次疊代

Flow chart :

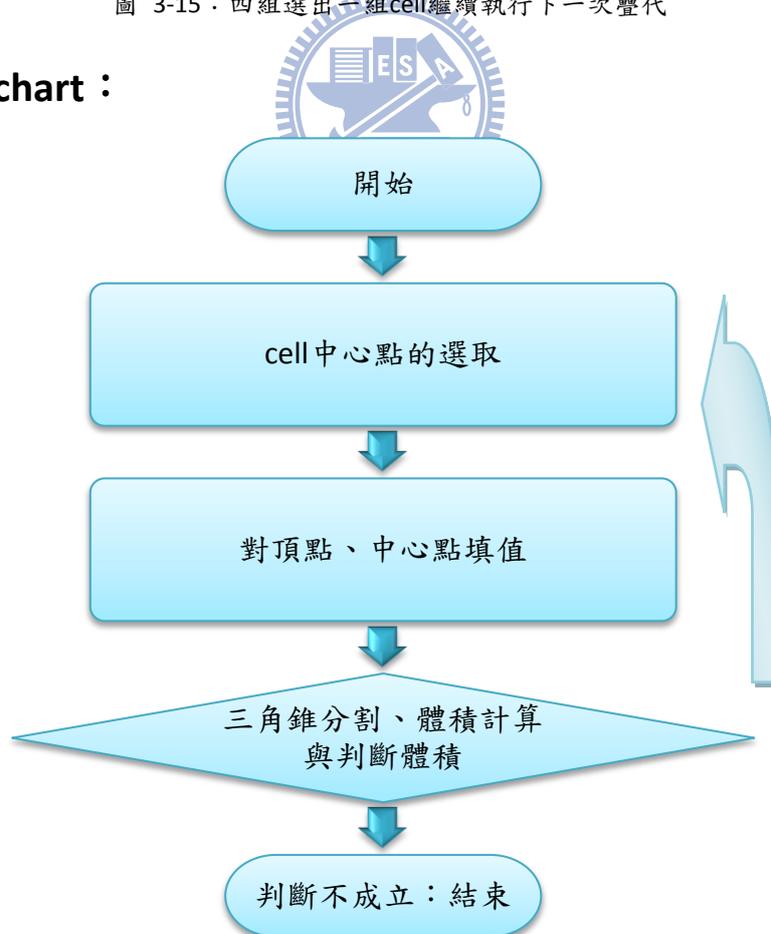


圖 3-16：Interpolation 1 method 流程圖

● 3.4 : Interpolation 2

一、首先，reference cell與target cell有特徵點(灰點)的對應，接著在兩組cell內部皆選擇一點，此點為cell的四個頂點的中點座標(紅點)，以及選取cell六個邊的中點(淺灰點)。如下圖所示。

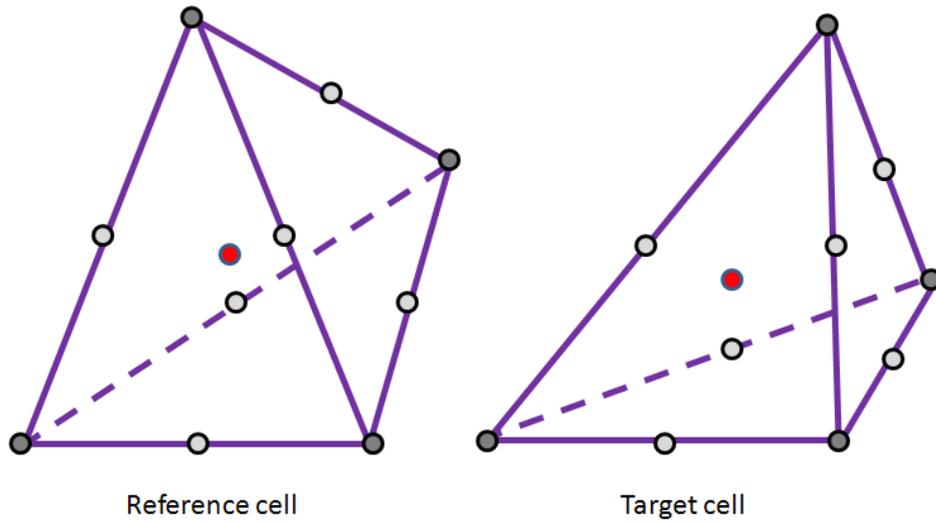


圖 3-17：Reference與target的對應關係

二、有特徵點(灰點)、中心點(紅點)及邊中點(淺灰點)的對應，接著把target cell 點的颜色值填到reference cell的點，因此一次可以填入11個點。如下圖所示。

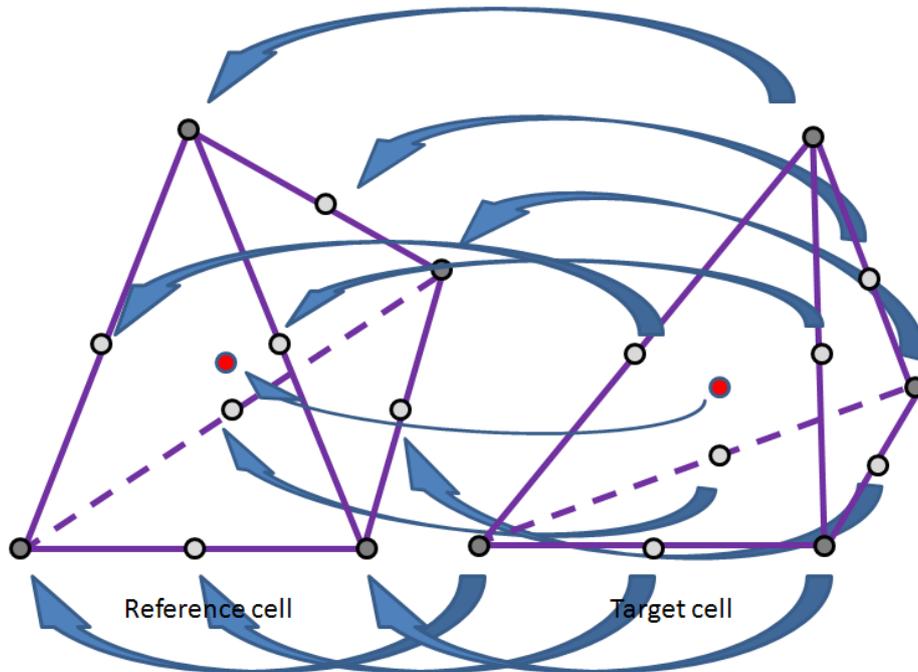


圖 3-18：從target填值至reference

三、把中心點(紅點)與cell的六個邊長中點(淺灰點)及各個邊長中點相互連接成線段。如下圖的短虛線即是，內部總共有19條線段。

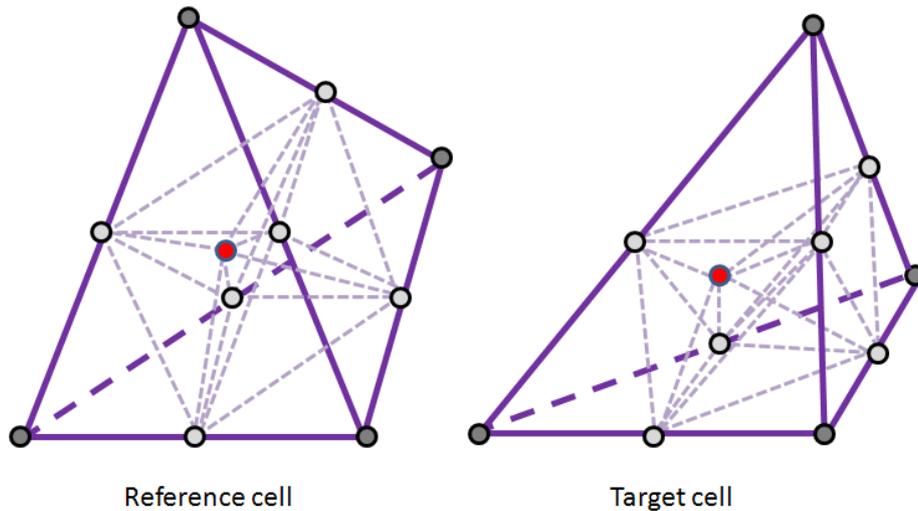


圖 3-19：三角錐的分割

四、上述組合可以再產生12個cell，從reference cell選定一個cell且先判斷體積。

假定cell的四個頂點分別為 p_1 、 p_2 、 p_3 、 p_4 ，體積計算如下：

$$\begin{aligned} \vec{V}_1 &= p_2 - p_1 & \vec{V}_2 &= p_3 - p_1 \\ \vec{V}_3 &= p_4 - p_1 & \vec{V}_4 &= \vec{V}_2 \wedge \vec{V}_3 \end{aligned}$$

Operator \wedge is cross
Operator \circ is dot

體積 $V = |\vec{V}_1 \circ \vec{V}_4| / 6$

選定 *threshold*，判斷體積 V $\left\{ \begin{array}{ll} \text{繼續步驟六} & \text{if } V > \textit{threshold} \\ \text{結束} & \text{if } V < \textit{threshold} \end{array} \right.$

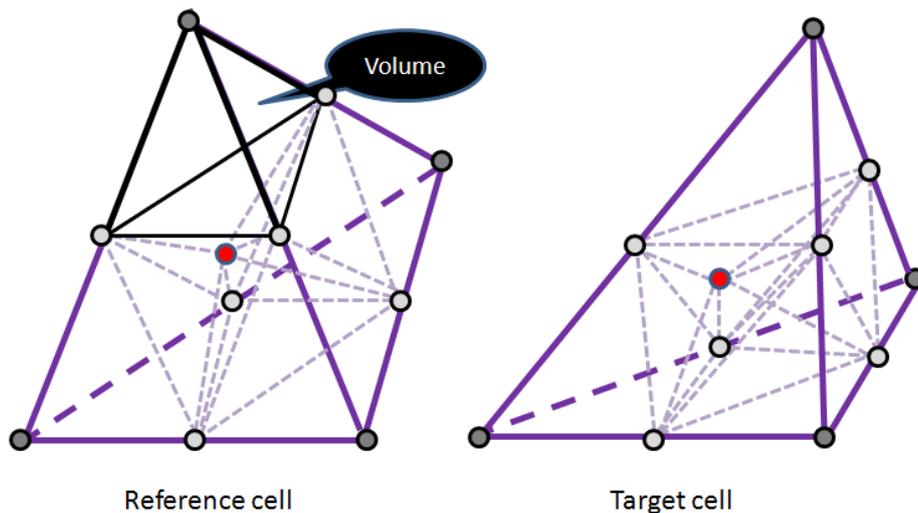


圖 3-20：選取其中一角錐計算體積

五、從上述可以得到十二個新的cell，為了方便描述，下圖利用黑色線來描述產生的新cell：

第一組為上方頂點與三個邊長中點構成的cell。

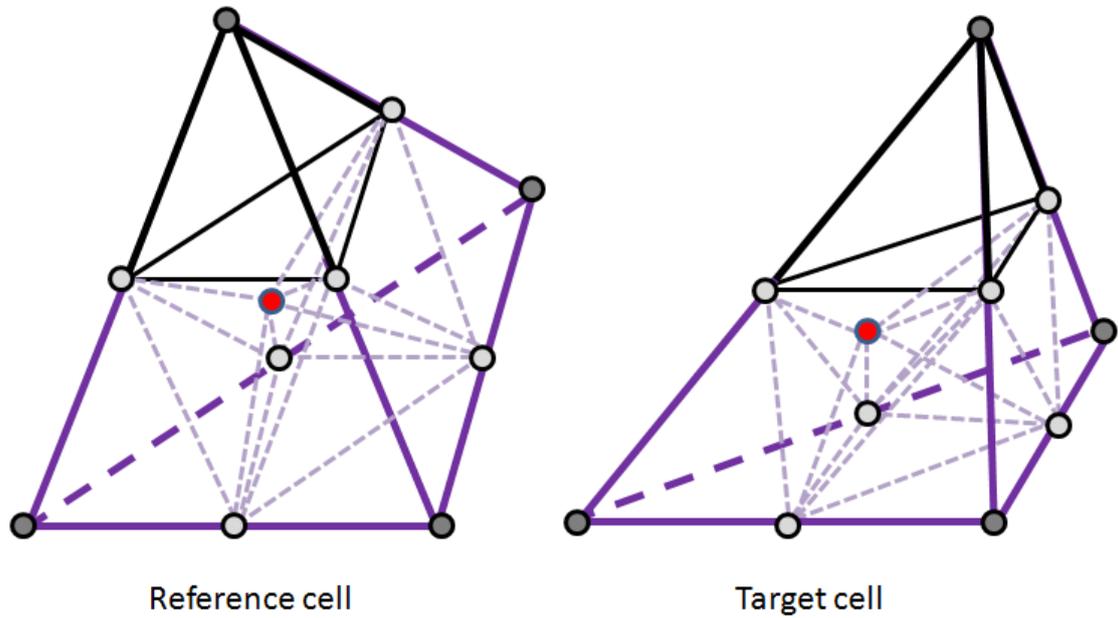


圖 3-21：第一組cell的對應

第二組為左方頂點與三個邊長中點構成的cell。

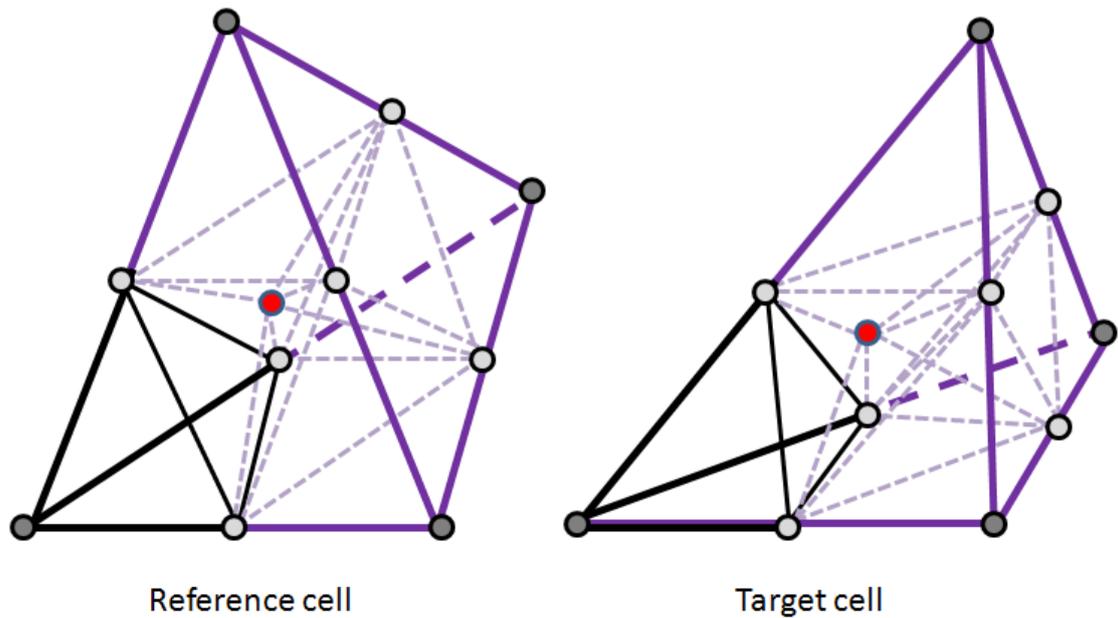


圖 3-22：第二組cell的對應

第三組為下方頂點與三個邊長中點構成的cell。

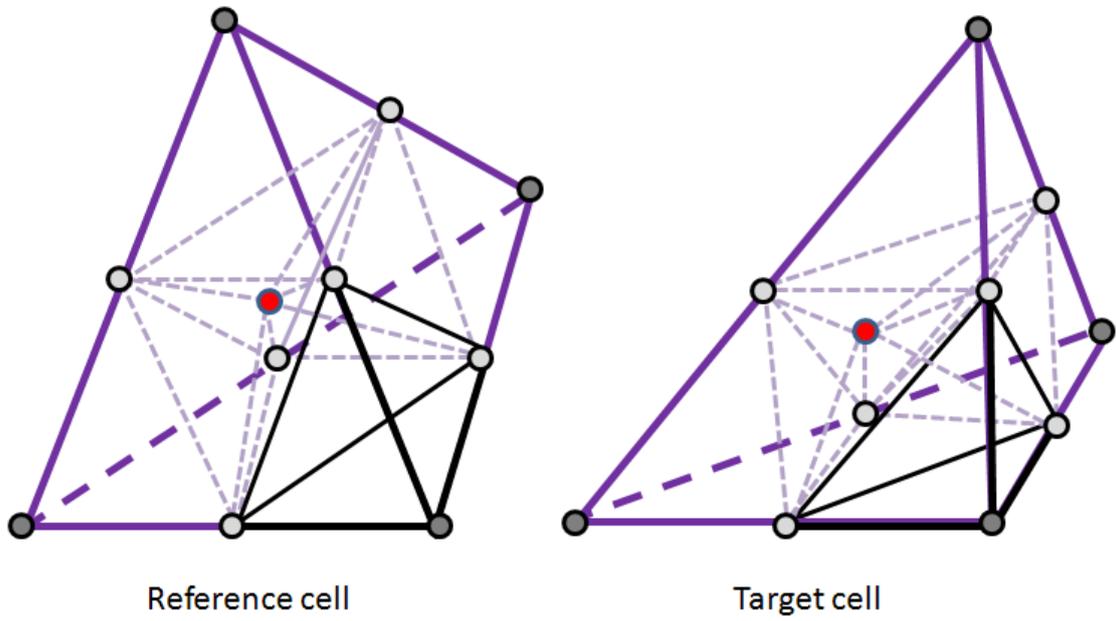


圖 3-23：第三組cell的對應

第四組為右方頂點與三個邊長中點構成的cell。

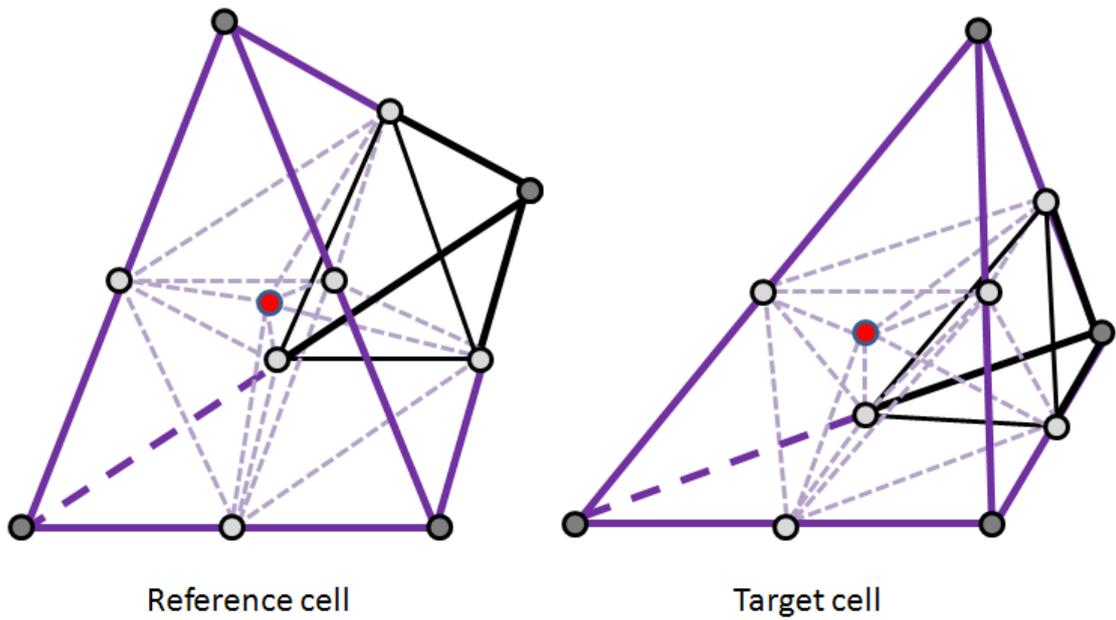


圖 3-24：第四組cell的對應

第五組為中心點與上方三個邊長中點構成的cell。

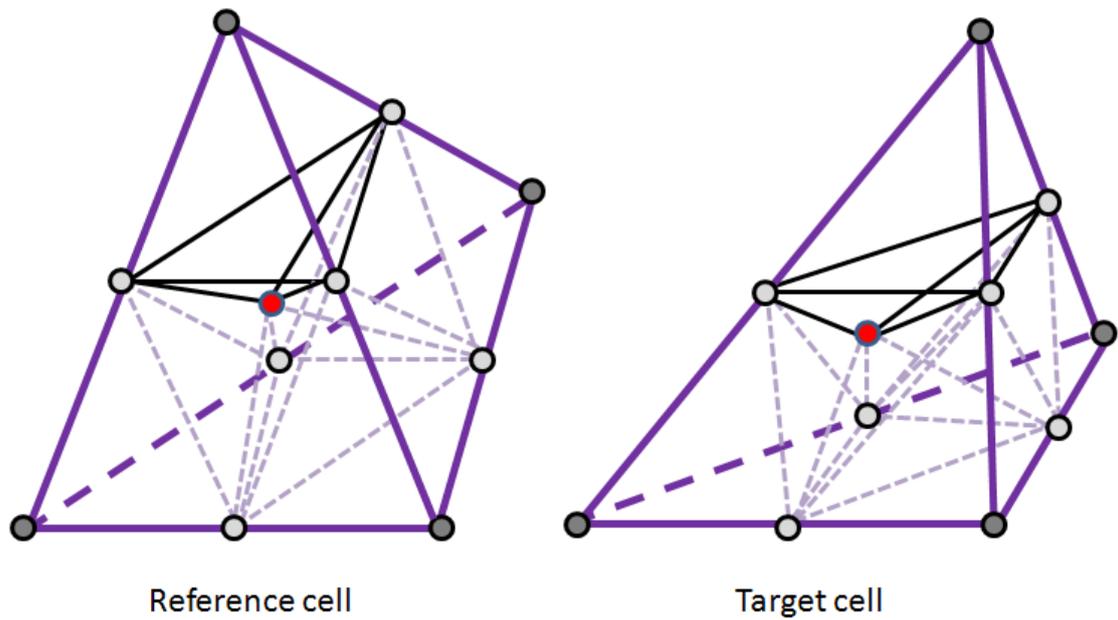


圖 3-25：第五組cell的對應



第六組為中心點與右方三個邊長中點構成的cell。

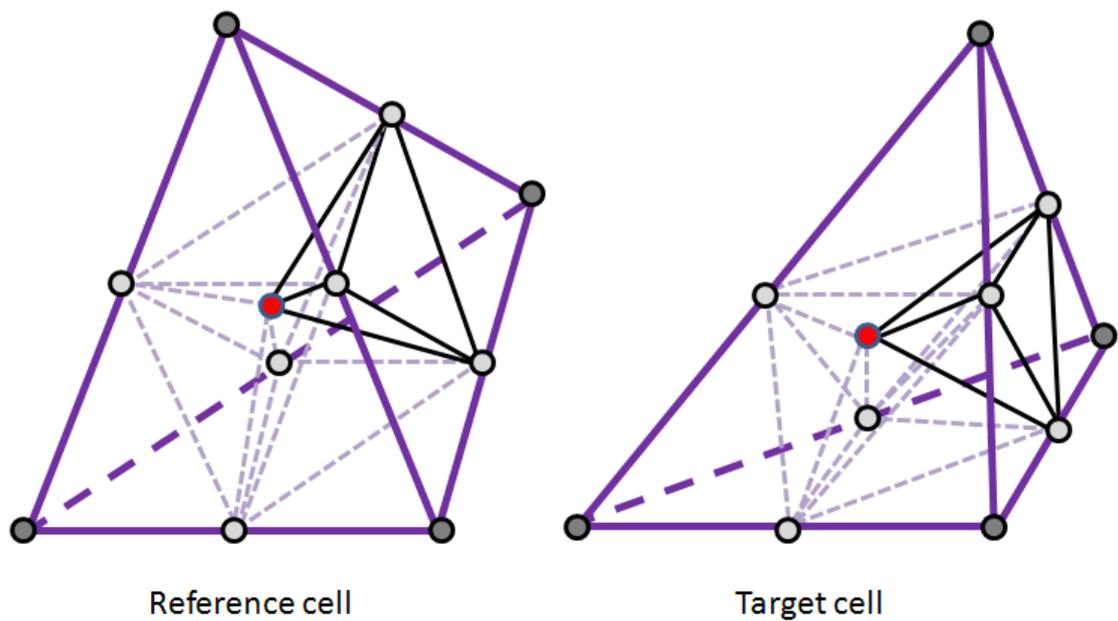


圖 3-26：第六組cell的對應

第七組為中心點與前方三個邊長中點構成的cell。

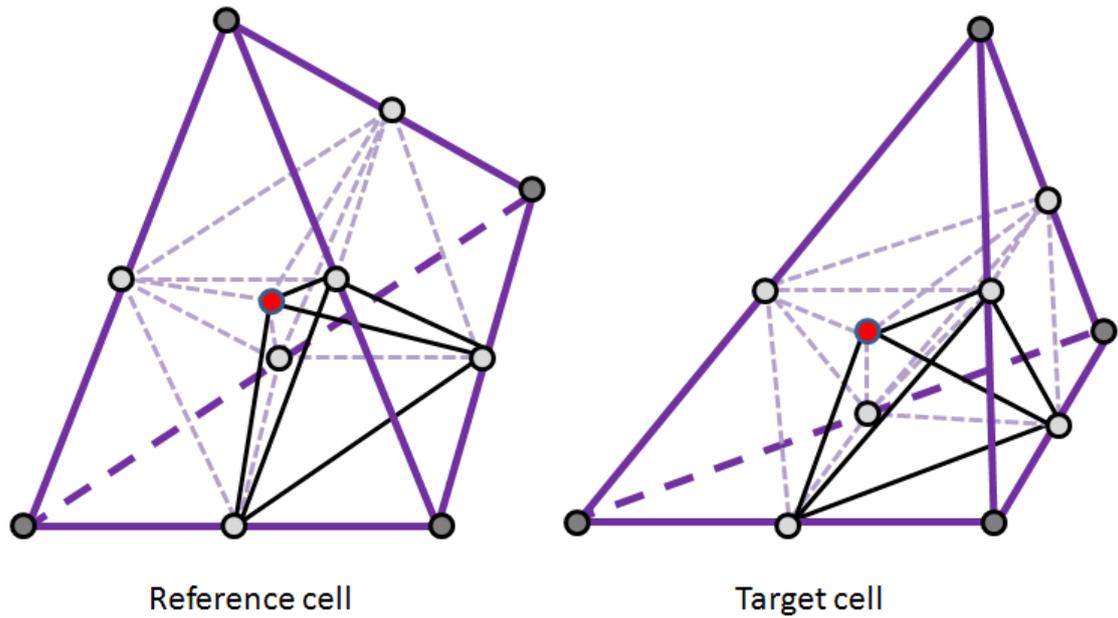


圖 3-27：第七組cell的對應



第八組為中心點與左前方三個邊長中點構成的cell。

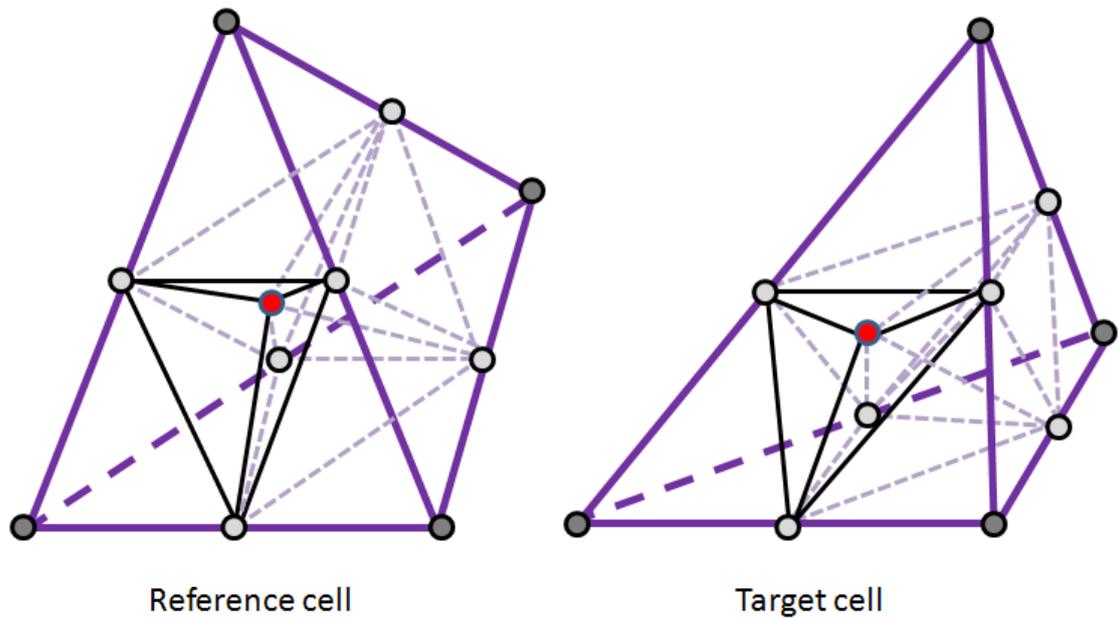


圖 3-28：第八組cell的對應

第九組為中心點與後方三個邊長中點構成的cell。

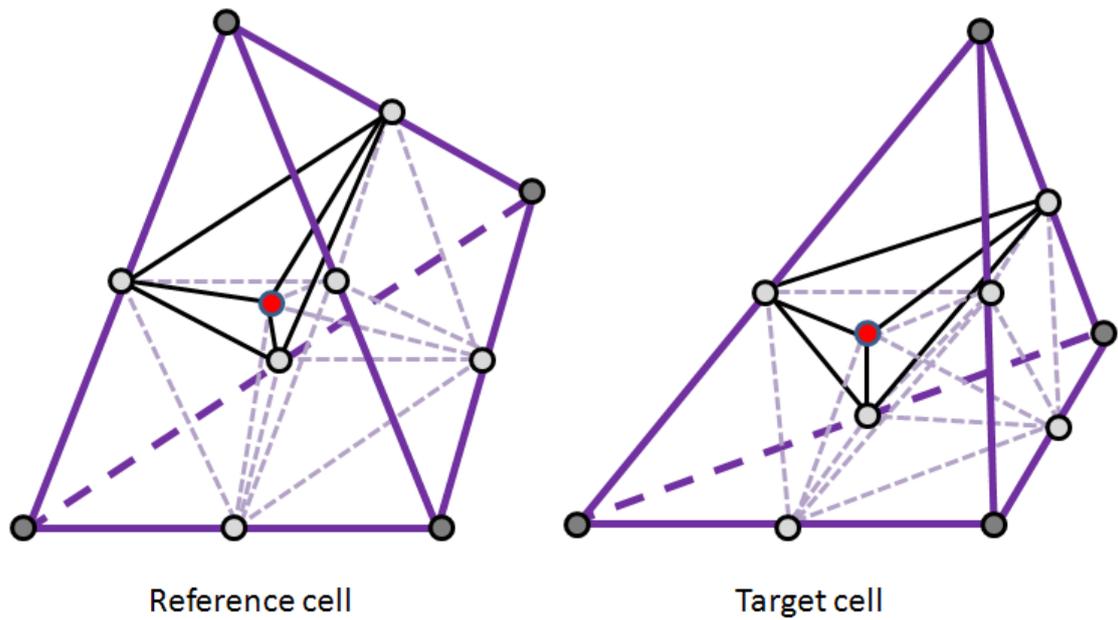
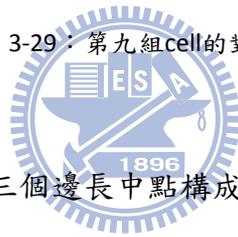


圖 3-29：第九組cell的對應



第十組為中心點與右後方三個邊長中點構成的cell。

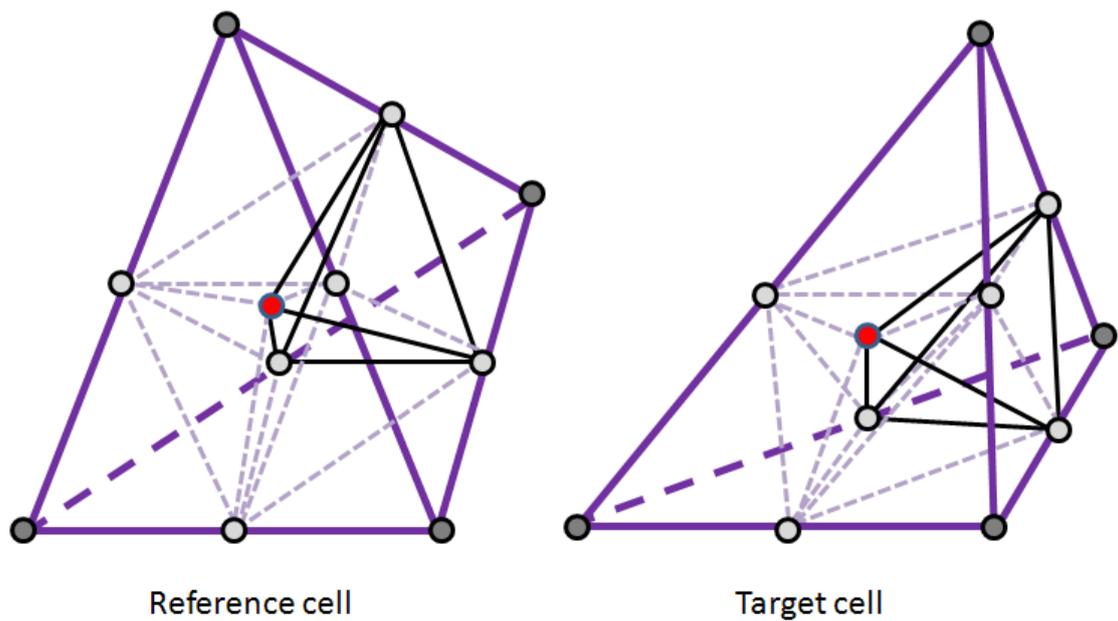


圖 3-30：第十組cell的對應

第十一組為中心點與下方三個邊長中點構成的cell。

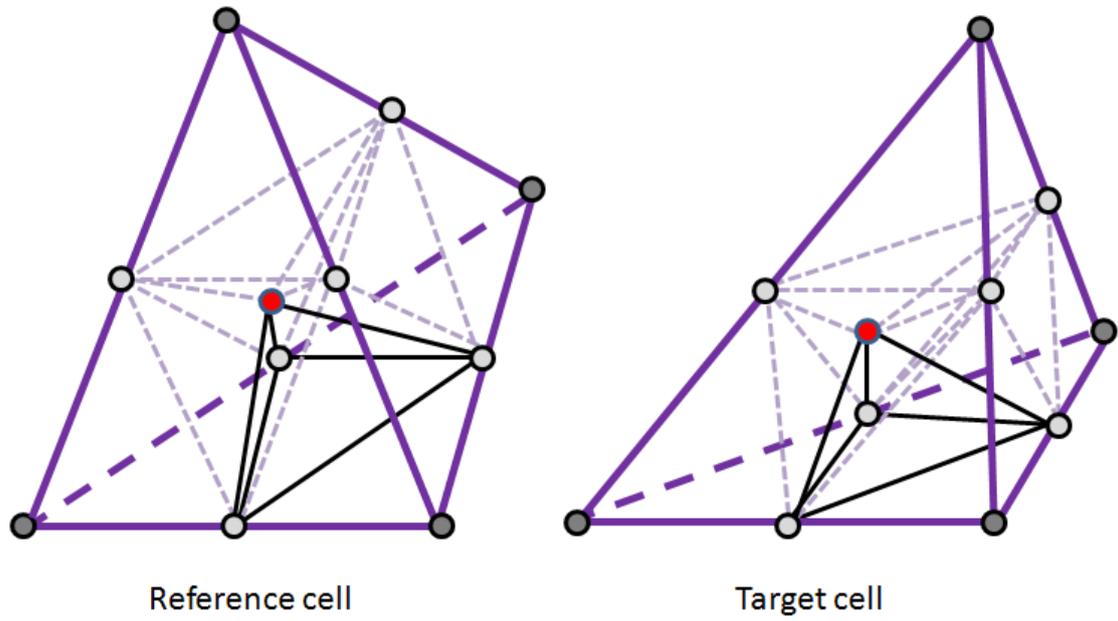
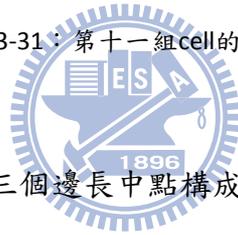


圖 3-31：第十一組cell的對應



第十二組為中心點與左方三個邊長中點構成的cell。

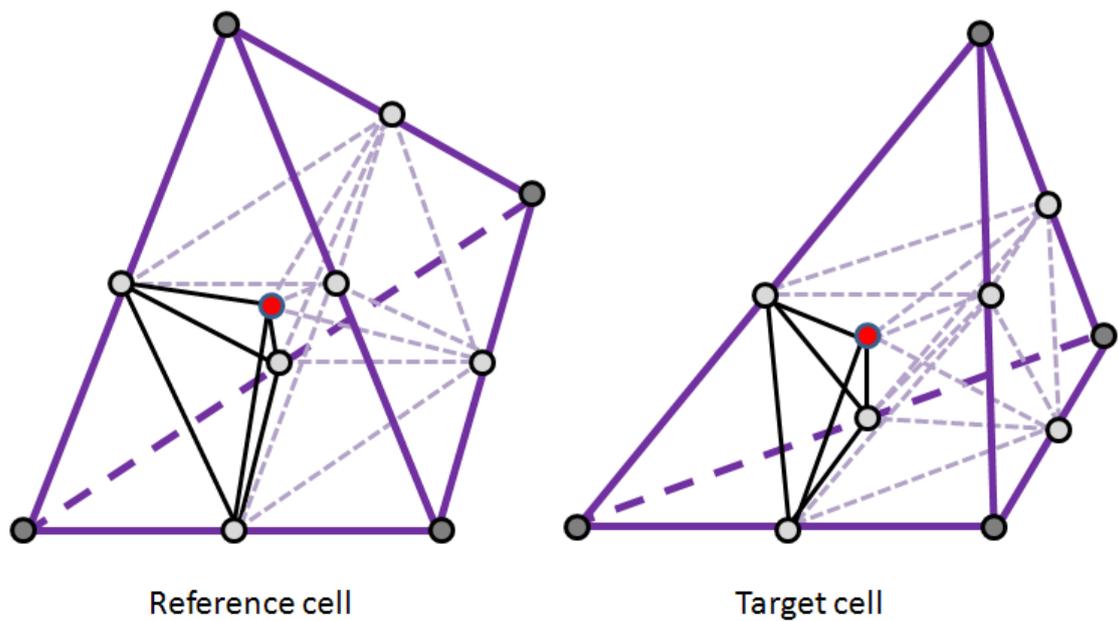


圖 3-32：第十二組cell的對應

六、從十二組cell中，選出第一組cell重新執行第一個步驟，如下圖所示，直到完成停止條件。而後再選出第二組cell重新執行第一個步驟...依序下去。

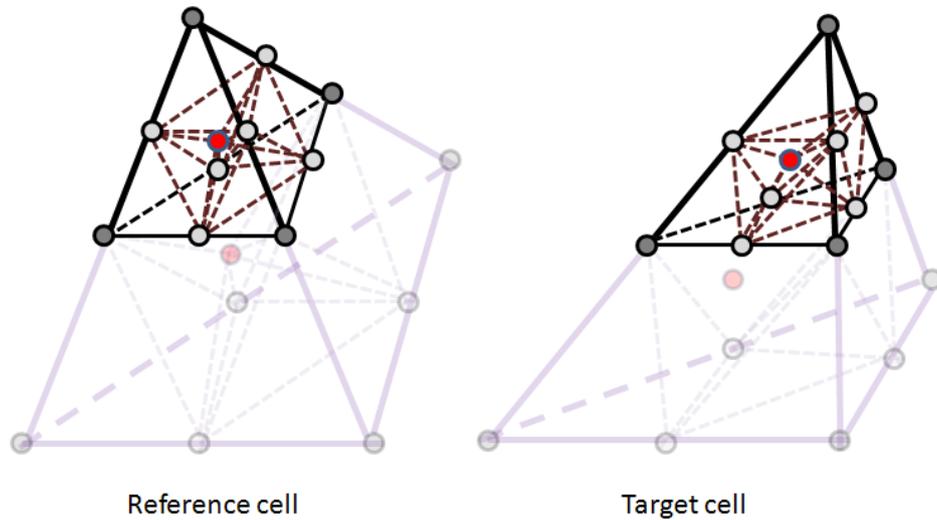


圖 3-33：從十二組選出一組cell繼續執行下一次疊代

Flow chart :

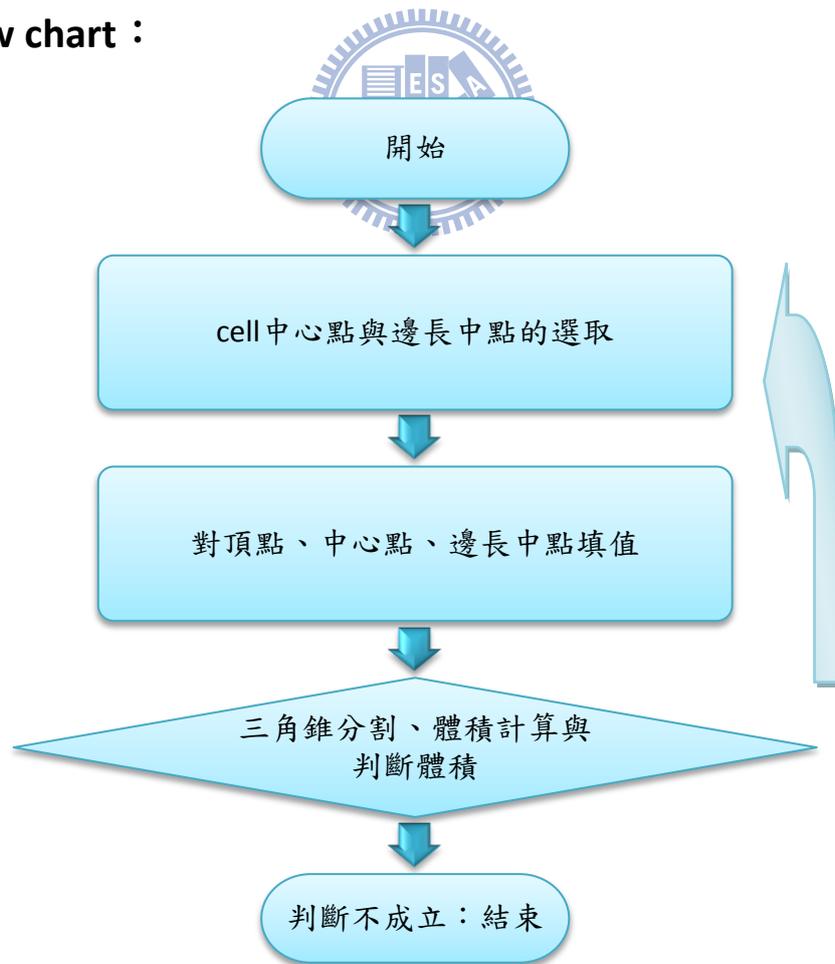


圖 3-34：Interpolation 2 method 流程圖

3D Morphing flow chart :

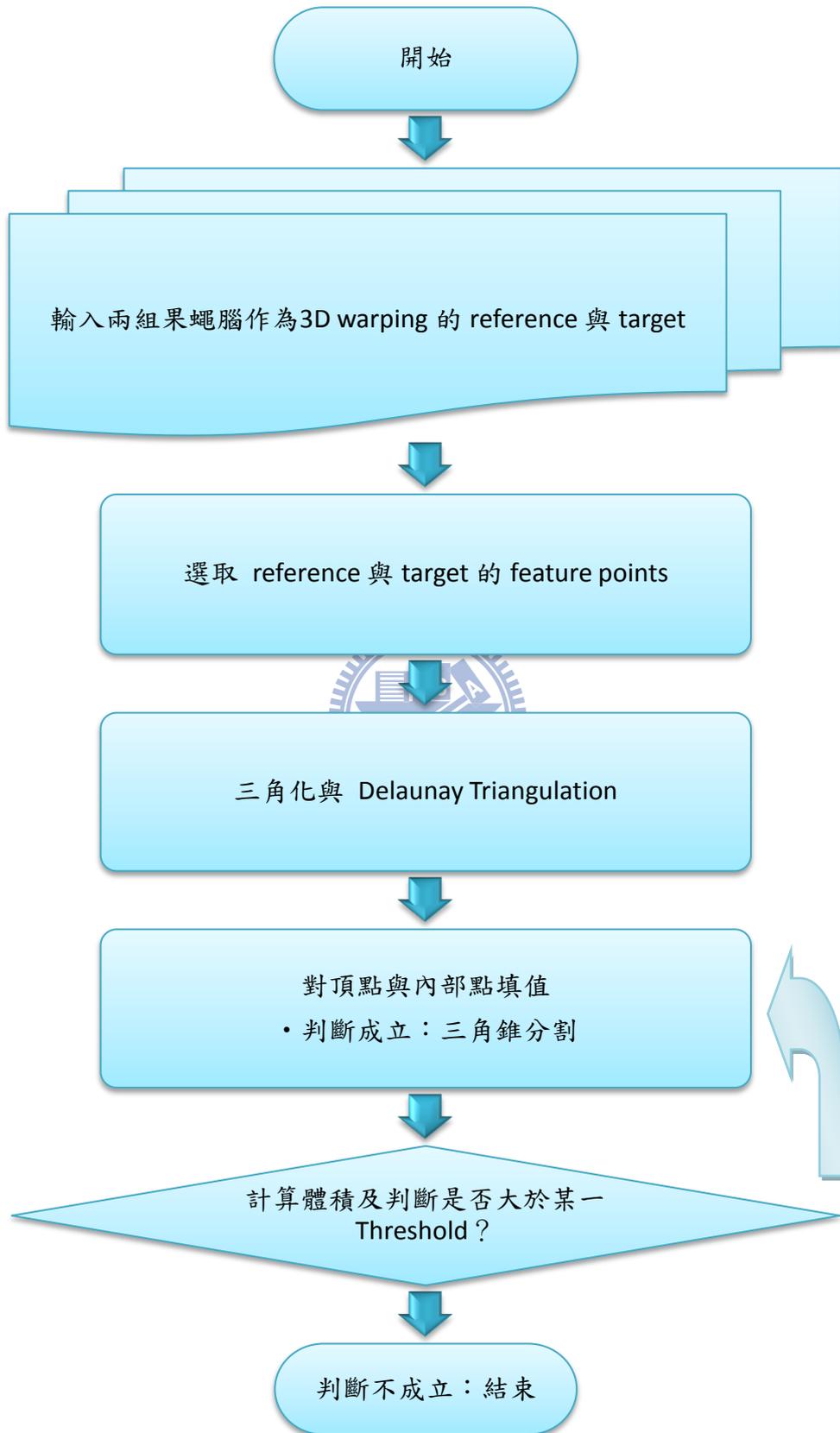


圖 3-35 : 3D Morphing flow chart

● 3.5 : Result

● 3.5.1 : Interpolation 1

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdlg1200	512x512x84	(16 , 105 , 5)	(508 , 392 , 73)	(258 , 216 , 43)
Or83bdlg1500	512x512x67	(19 , 136 , 4)	(486 , 369 , 63)	(259 , 216 , 26)

下圖實驗為選定Threshold：0.3，執行時間500秒~600秒。

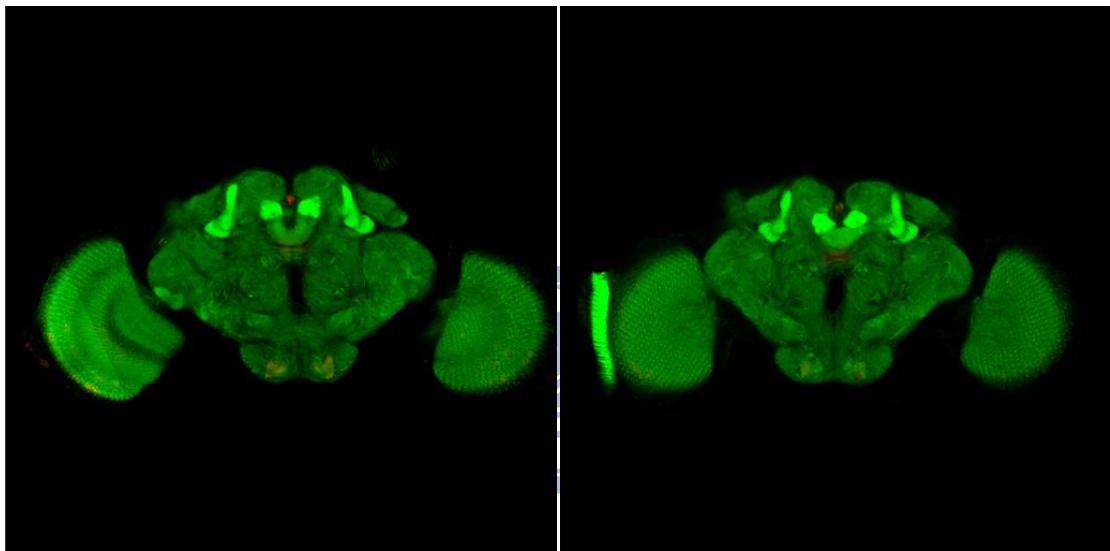


圖 3-36：檔案名稱為 1200 與 1500 的果蠅腦資訊

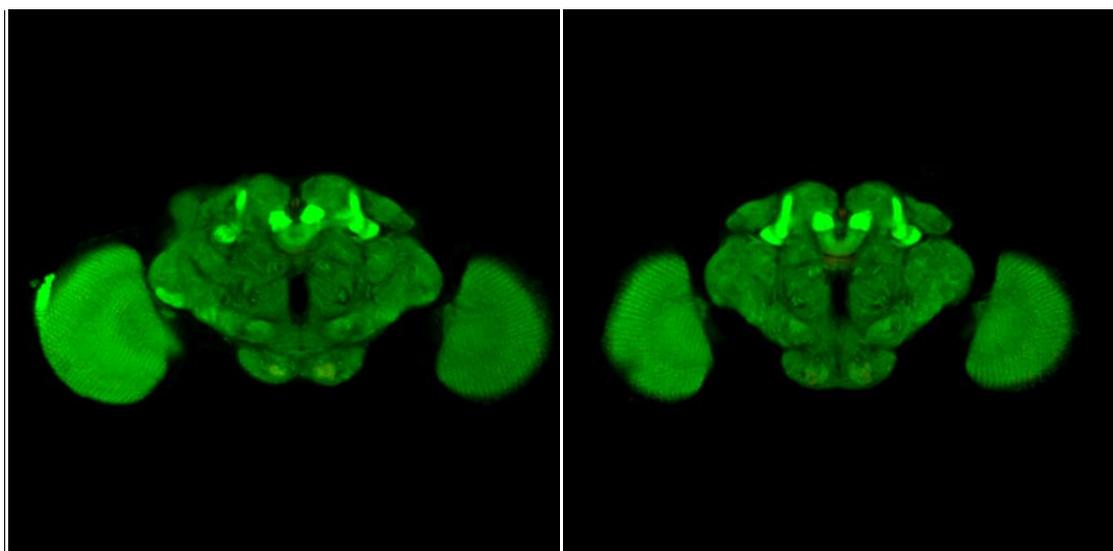


圖 3-37：1500 warp 至 1200 與 1200 warp 至 1500

● 3.5.2 : Interpolation 2

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdlg12001	512x512x87	(36 , 128 , 5)	(485 , 368 , 76)	(253 , 208 , 40)
Or83bdlg12002	512x512x81	(31 , 145 , 3)	(487 , 366 , 70)	(259 , 206 , 36)

下圖實驗為選定Threshold：0.3，執行時間1000秒~1200秒。

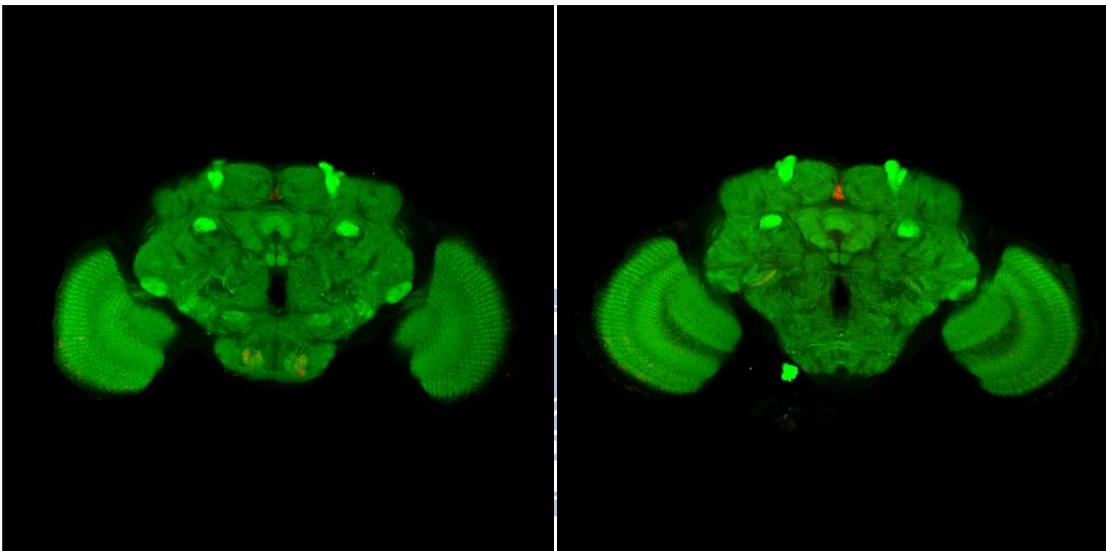


圖 3-38：檔案名稱為 12001 與 12002 的果蠅腦資訊

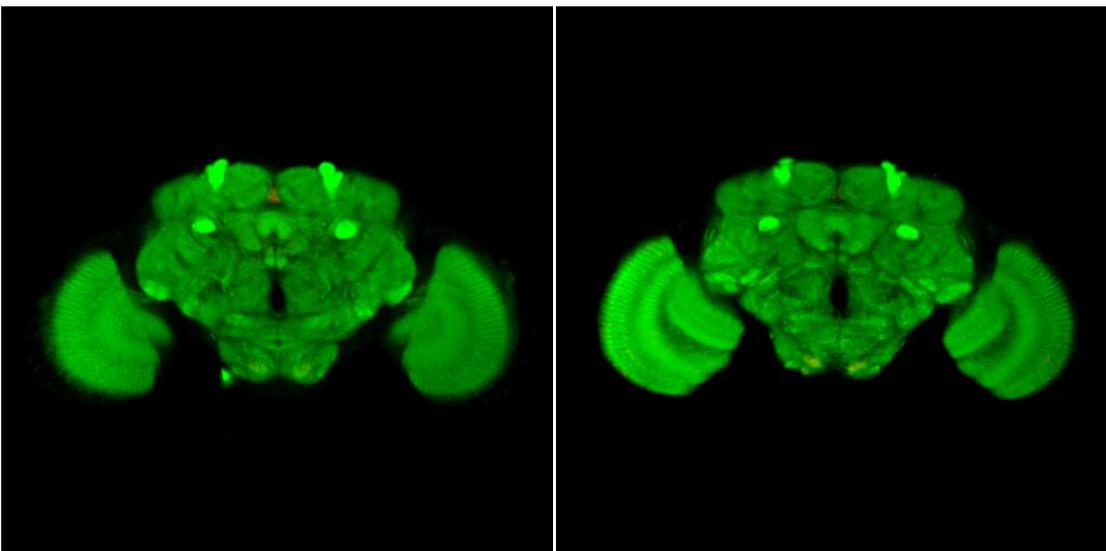


圖 3-39：12002 warp 至 12001 與 12001 warp 至 12002

● 3.5.3 : Bad result

下列圖為使用interpolation的方法。在特定的區域中，影像會有浮動、被擠壓、不自然的變形，如圖中紅色方框內即為此。

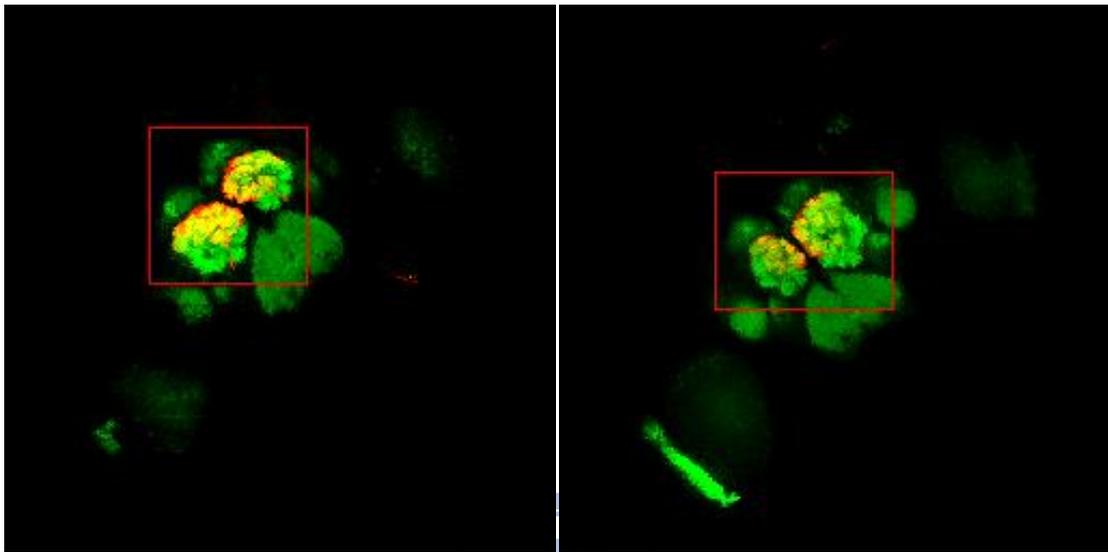


圖 3-40 : Bad image1

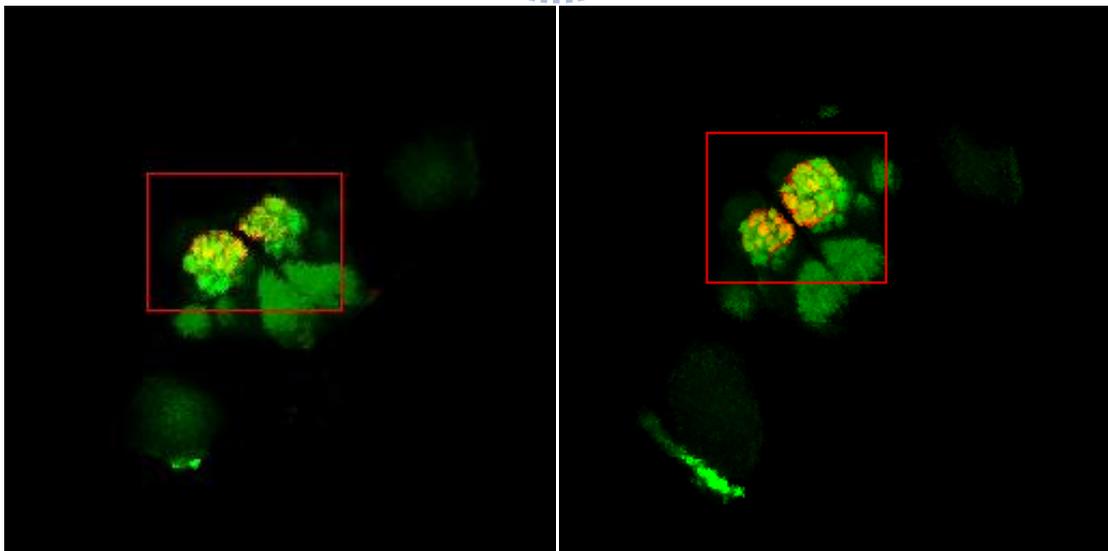


圖 3-41 : Bad image2

第四章 Template brain

由於上述所使用的interpolation的方法所產生的結果不如我們預期的理想，因此Template brain建立主要是利用陳永昇老師的實作來應用。

我們參考建立標準腦的制定流程[11]，制訂出一套標準果蠅腦的實作流程。在統計學上，我們所需的果蠅腦樣本數至少需30個以上，才具公信力。扣除一些實作後結果不好的樣本，所有被參考的總樣本數為34組。

● 4.1 : Introduction of affine transformation

Affine主要包含了rotation、translation、scaling與shearing這些矩陣的轉換，因此定義如下[12]：

$$s' = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = Ms = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

其中： $p = [p_1 p_2 \dots p_{12}]^T$ ， p 代入 M 即為affine transformation

s' 為經過轉換後的點。(target model)

s 為原始的點。(original target model)

M 即為上述轉換矩陣後的乘積。接著我們可以定義target model與reference model的距離：

$$D = \sum_{i=1}^N d^2(s_i)$$

其中： d^2 為 s' 與reference model的距離。

N 為target model點的個數。

觀察function $D(p)$ ，我們要找到一組 p 並且使得 $D(p)$ 為最小。代入 M 即為轉換的affine transformation。

● 4.2 : Introduction of nonrigid transformation

主要是利用RBF(Radius basis function)，分佈於d個維度，對應的function為 $T(\vec{x})$ ，其中 $k = 1, \dots, d$ ，表示如下：

$$T(\vec{x}) = [f_1(\vec{x}), \dots, f_k(\vec{x}), \dots, f_d(\vec{x})]$$

RBF的function主要由兩個部份組合而成：

- Global：根據control point影響的是全域的影像。其數學式如下：

$$P_{mk}(\vec{x})$$

- Local：根據control point影響的是區域的影像。其數學式如下：

$$\sum_{i=1}^N w_{ik} g(r_i)$$

因此可以將RBF的function寫成：

$$f_k(\vec{x}) = P_{mk}(\vec{x}) + \sum_{i=1}^N w_{ik} g(r_i) \quad x \in R^d$$

其中： P_{mk} 為一多項式，最高次方為 m 。

w_{ik} 為第 i 個權重值。

$r_i = \|\vec{x} - \vec{x}_i\|$ 為點 x 與點 x_i 的距離(Euclidean norm)。

g 為basis function

下列是一些常用的radial basis function，其中 $r = \|x - c_i\|$ ：

- Gaussian : $g(r) = e^{-r^2/2\sigma^2}$
- Multiquadric : $g(r) = (r^2 + c^2)^{1/2}$
- Inverse Multiquadrics : $g(r) = (r^2 + c^2)^{-1/2}$
- Thin plate spline : $g(r) = r^2 \ln(r)$
- LCD : $g(r) = (1 - r^2)^3$
- Wendland : $g(r) = \begin{cases} p(r) & 0 \leq r \leq 1 \\ 0 & r \geq 1 \end{cases}$

在此將RBF用於三維($d = 3$)的生物醫學影像，因此 P_{mk} 多項式為線性($m = 1$)，多項式表示為：

$$P_{mk}(\vec{x}) = a_0 + a_1x + a_2y + a_3z$$

分別從target與reference image上選擇相對應的control point $\vec{x}_j = (x, y, z)$ 與 $\vec{u}_j = (x_i, y_i, z_i)$ ，且需滿足：

$$f_k(\vec{x}_j) = \vec{u}_{jk}, \quad j = 1, 2, \dots, n$$

上述表示 \vec{x}_j 為target image的第 j 個control point可經由function f_k 找到相對應 \vec{u}_j 在reference image的第 j 個control point。

此聯立方程式有 $n+4$ 個變數，由於上述只有 n 個線性方程式，因此需要加入額外的限制條件(constraints)，如下：

$$\sum_{i=1}^n w_i = \sum_{i=1}^n w_i x_i = \sum_{i=1}^n w_i y_i = \sum_{i=1}^n w_i z_i = 0$$

接著列出所有方程式表示如下：

$$\begin{bmatrix} g(r_{11}) & \cdots & g(r_{1n}) & \mathbf{1} & x_1 & y_1 & z_1 \\ \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g(r_{n1}) & \cdots & g(r_{nn}) & \mathbf{1} & x_n & y_n & z_n \\ \mathbf{1} & \cdots & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ x_1 & \cdots & x_n & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ y_1 & \cdots & y_n & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ z_1 & \cdots & z_n & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}_k \begin{bmatrix} w_1 \\ \vdots \\ w_n \\ a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}_k = \begin{bmatrix} u_1 \\ \vdots \\ u_n \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}_k$$

可解得未知係數 w_1, \dots, w_n 與 a_0, a_1, a_2, a_3 ，且可得知每一個control point 其對應的control point之間所移動的距離。

● 4.3 : Template brain

首先，利用 affine、mask、nonrigid 操作一組檔案的 warp 動作。

在 affine 的部份，先選擇 reference file(R)與 target file(T)，下圖所選取的檔案。

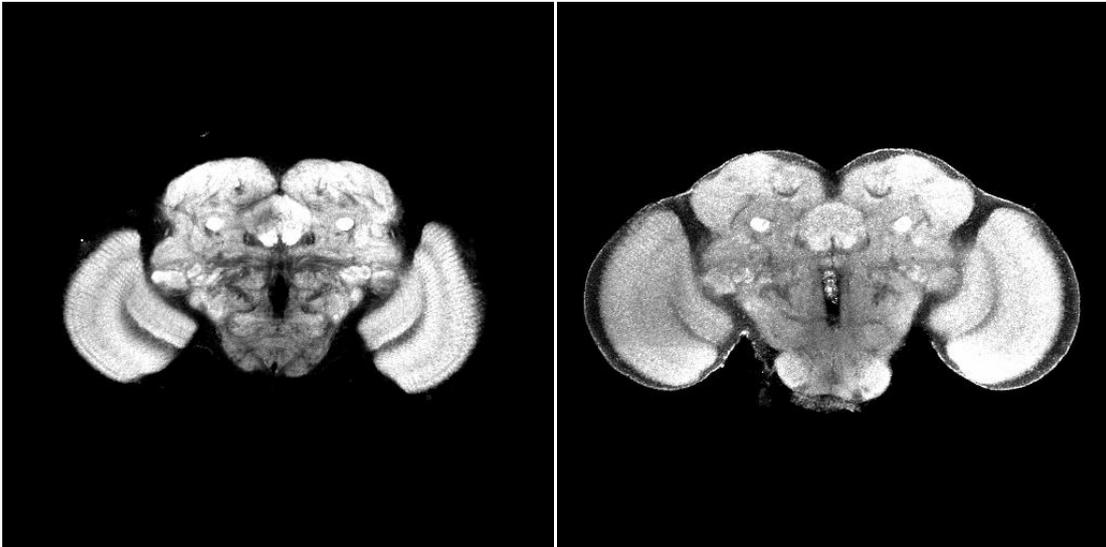


圖 4-1：reference file與target file

- 利用 affine，選擇檔案(T)與(R)並且設定 Degree of freedom(DOF)為 7，產生的結果為(aT)。
- 利用 affine，選擇檔案(aT)與(R)並且設定 Degree of freedom(DOF)為 12，產生的結果為(aaT)。如下圖所示。

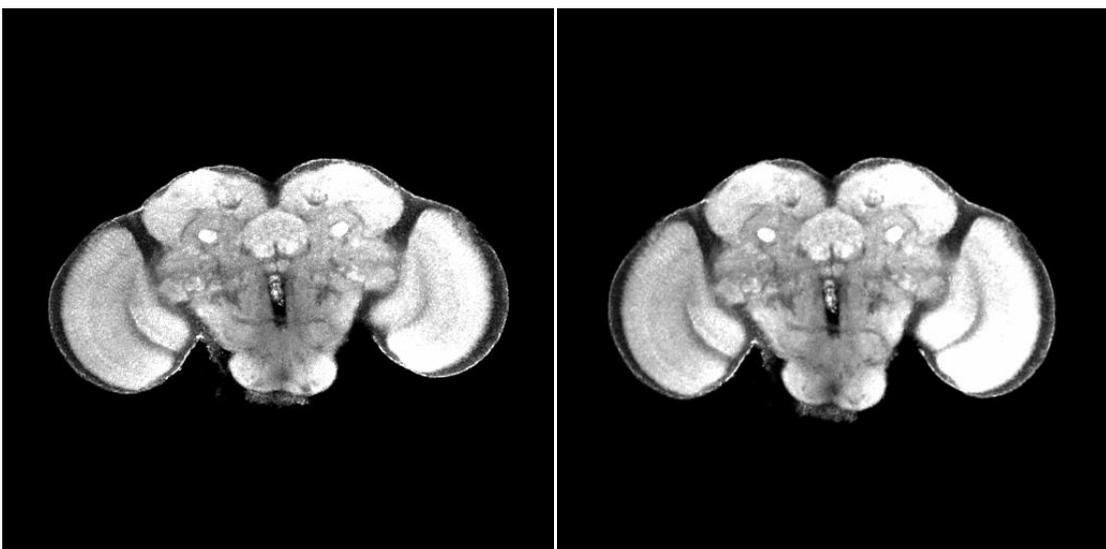


圖 4-2：一次affine與二次affine

- 接著把reference(**R**)取mask，得到feature mask(**fR**)。
- 最後nonrigid的設定：
 - 一、 Test image ：二次affine後的結果(**aaT**)。
 - 二、 Reference image ：reference file(**R**)。
 - 三、 Feature mask ：mask後的結果(**fR**)。

執行後，下圖為最後warp的結果。

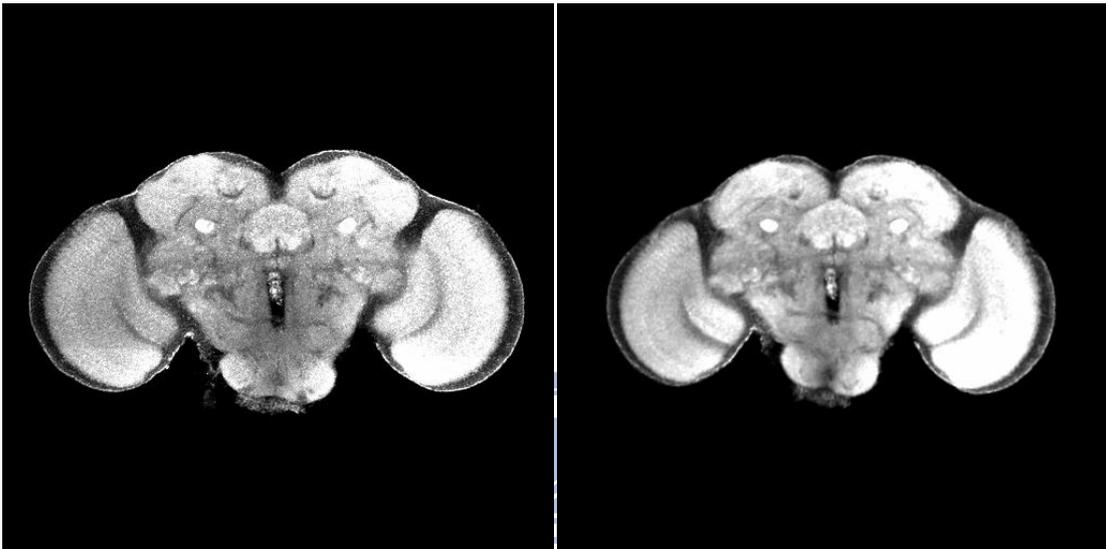


圖 4-3：原始檔與warp後的結果

上述即為一組 warp 的過程。接著將 42 組果蠅腦相互 warp 後，會得到總共 $42 \times 41 = 1722$ 組的結果。檢查產生後的結果，去除不理想的影像及原始果蠅腦有不完整的影像，有 8 組。因此被我們所參考的果蠅腦總共有 34 組。接著考慮這 34 組相互 warp 後產生的 x、y、z 方向的 deformation field。每一組腦都有其它 33 組 warp 後的 deformation 並算出位移量將其累加，即為總位移量的值。接著為了使得 warp 後位移量為最小，因而取出這 34 組裡最小的總位移量，我們把這個腦當作暫時的標準腦。

而後，再把其它 33 組腦分別 warp 到暫時的標準腦，因此會產生 33 組的結果，最後我們再把這 33 組作平均，即為最後的標準腦(template brain)。

Template brain flow chart :

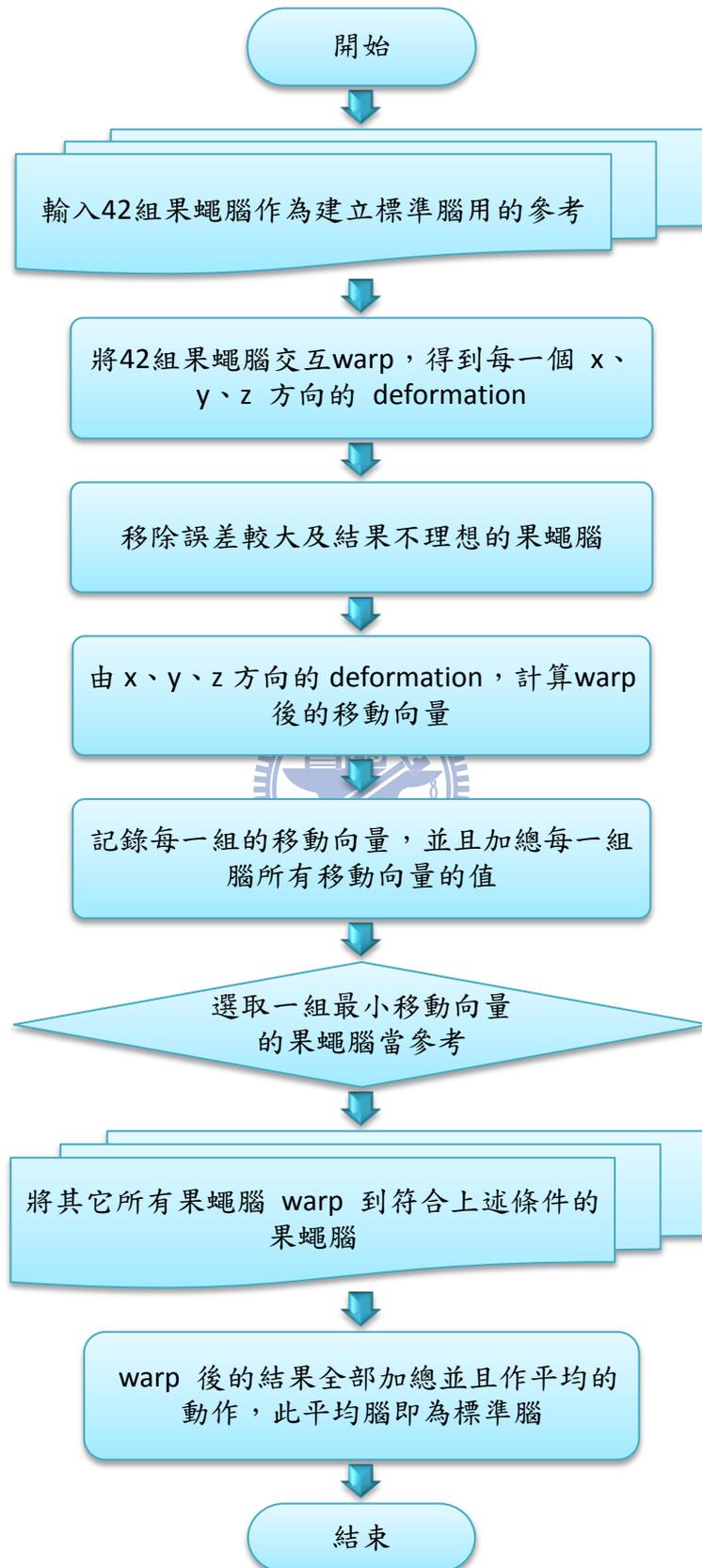


圖 4-4：標準腦的建立流程

● 4.4 : Neuron warp

接著我們需將神經 warp 到標準腦中。在 warp 過程中，由於一次無法處理兩個 channel 也無法單獨處理神經的 warp，因此需要對果蠅腦做前處理。

首先，對其中任意擁有兩個 channel 的果蠅腦分成兩個各包含一個 channel 的檔案，接著我們把結構(T)warp 到標準腦中。在 warp 過程中，affine 會產生一個轉換的矩陣(M)，二次 affine 也會產生一個轉換矩陣(aM)，最後處理 nonrigid 也會產生一個 x、y、z 方向的 deformation(xd、yd、zd)。

再來，神經(N)的 warp 即利用這些轉換的矩陣來操作。

- 利用 applyaff，神經經由轉換矩陣(M)產生第一次 affine 後的結果(aN)。
- 利用 applyaff，第一次的結果(aN)經由轉換矩陣(aM)產生第二次 affine 的結果(aaN)。
- 利用 applydef，把第二次的結果(aaN)由 x、y、z 方向的 deformation(xd、yd、zd)產生結果，即為最後 warp 的處理。

如下圖所示，紅色方格即為果蠅腦大小。



圖 4-5：原始神經與warp後的結果

● 4.5 : Result

● 4.5.1 : Affine

下列為其中兩組果蠅腦相互 warp 的結果。

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
F_antidlg_20mb_12	512x512x60	(42 , 155 , 2)	(455 , 369 , 57)	(255 , 214 , 24)
F_antidlg_20mb_06	512x512x60	(26 , 113 , 1)	(503 , 366 , 52)	(261 , 217 , 25)

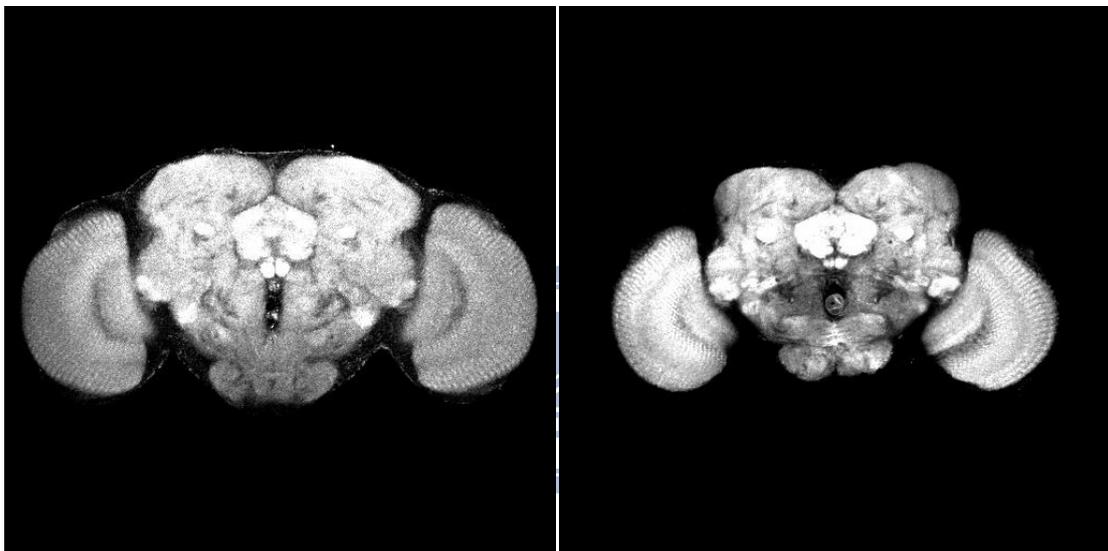


圖 4-6 檔案編號 12 與 6

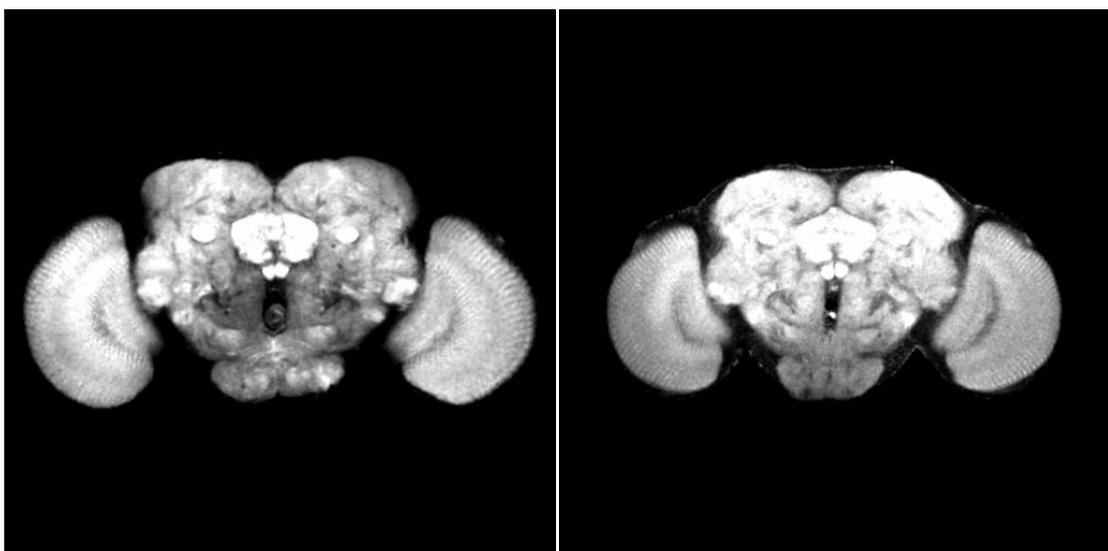


圖 4-7 檔案編號 6 warp 至 12 與 12 warp 至 6

下列為其中兩組果蠅腦相互 warp 的結果。

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
F_antidlg_20mb_34	512x512x69	(62 , 128 , 5)	(462 , 347 , 62)	(254 , 208 , 28)
F_antidlg_20mb_41	512x512x64	(9 , 122 , 4)	(493 , 376 , 60)	(246 , 209 , 25)

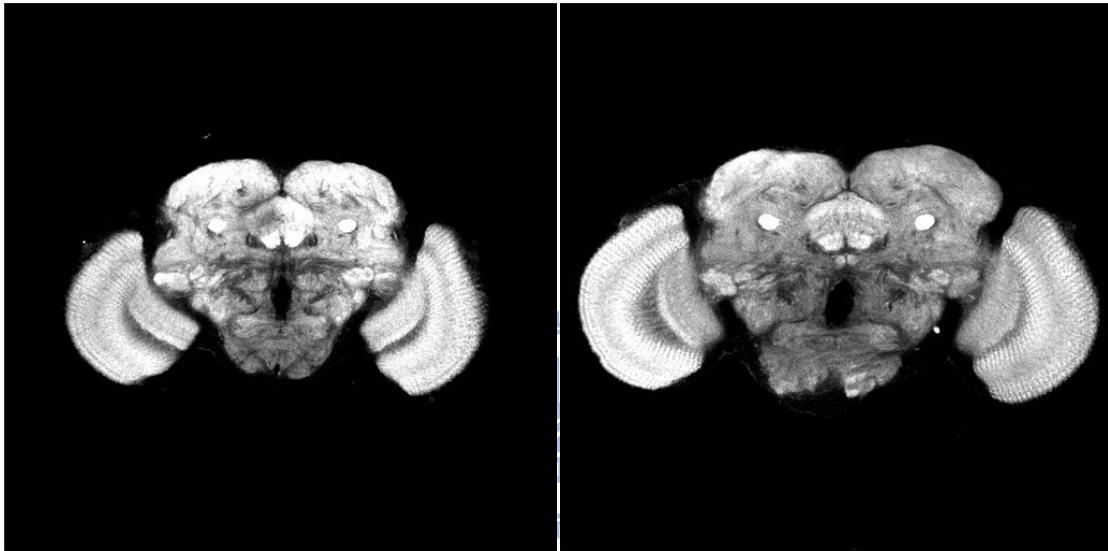


圖 4-8 檔案編號 34 與 41

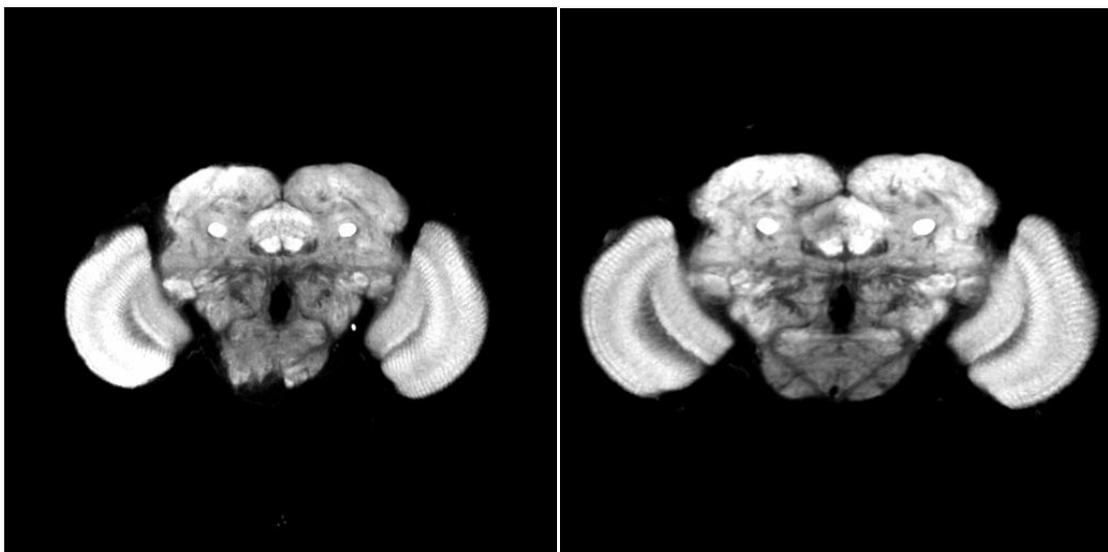


圖 4-9 檔案編號41 warp至34與34 warp至41

● 4.5.2 : Template brain

要建立標準腦，做為往後所有果蠅腦的參考，因此我們需要參考的果蠅腦資訊量不能過少。下表為 34 組果蠅腦的資訊。

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
F_antidlg_20mb_06	512x512x60	(26 , 113 , 1)	(503 , 366 , 52)	(261 , 217 , 25)
F_antidlg_20mb_08	512x512x55	(23 , 143 , 1)	(508 , 384 , 54)	(257 , 211 , 24)
F_antidlg_20mb_09	512x512x60	(16 , 128 , 1)	(477 , 352 , 60)	(257 , 210 , 23)
F_antidlg_20mb_11	512x512x65	(19 , 163 , 5)	(475 , 377 , 62)	(256 , 208 , 26)
F_antidlg_20mb_12	512x512x60	(42 , 155 , 2)	(455 , 369 , 57)	(255 , 214 , 24)
F_antidlg_20mb_13	512x512x61	(45 , 142 , 2)	(463 , 347 , 59)	(254 , 212 , 25)
F_antidlg_20mb_14	512x512x67	(52 , 141 , 6)	(435 , 348 , 61)	(259 , 214 , 24)
F_antidlg_20mb_15	512x512x70	(44 , 158 , 6)	(473 , 369 , 59)	(263 , 219 , 27)
F_antidlg_20mb_16	512x512x68	(0 , 136 , 3)	(494 , 371 , 64)	(262 , 211 , 25)
F_antidlg_20mb_18	512x512x65	(10 , 145 , 1)	(489 , 374 , 62)	(261 , 208 , 27)
F_antidlg_20mb_19	512x512x70	(27 , 126 , 6)	(485 , 372 , 65)	(255 , 217 , 27)
F_antidlg_20mb_20	512x512x66	(28 , 128 , 3)	(489 , 355 , 62)	(256 , 222 , 25)
F_antidlg_20mb_21	512x512x77	(21 , 139 , 3)	(483 , 349 , 67)	(259 , 218 , 29)
F_antidlg_20mb_22	512x512x75	(26 , 132 , 7)	(485 , 480 , 38)	(255 , 204 , 33)
F_antidlg_20mb_23	512x512x70	(45 , 120 , 7)	(456 , 404 , 65)	(252 , 204 , 29)
F_antidlg_20mb_24	512x512x66	(23 , 112 , 6)	(505 , 359 , 64)	(262 , 207 , 26)
F_antidlg_20mb_25	512x512x67	(17 , 105 , 5)	(493 , 345 , 62)	(260 , 206 , 25)

F_antidlg_20mb_26	512x512x63	(14 , 137 , 4)	(431 , 354 , 60)	(261 , 212 , 24)
F_antidlg_20mb_27	512x512x69	(3 , 110 , 6)	(462 , 348 , 65)	(258 , 208 , 29)
F_antidlg_20mb_28	512x512x70	(36 , 115 , 7)	(479 , 360 , 66)	(258 , 206 , 26)
F_antidlg_20mb_29	512x512x65	(14 , 128 , 3)	(479 , 367 , 62)	(262 , 208 , 27)
F_antidlg_20mb_30	512x512x72	(14 , 116 , 6)	(495 , 364 , 66)	(265 , 207 , 27)
F_antidlg_20mb_31	512x512x70	(39 , 109 , 3)	(480 , 1344 , 62)	(263 , 209 , 26)
F_antidlg_20mb_32	512x512x64	(34 , 128 , 2)	(459 , 362 , 57)	(253 , 203 , 24)
F_antidlg_20mb_33	512x512x65	(27 , 125 , 7)	(470 , 367 , 61)	(253 , 206 , 25)
F_antidlg_20mb_34	512x512x69	(62 , 128 , 5)	(462 , 347 , 62)	(254 , 208 , 28)
F_antidlg_20mb_35	512x512x70	(21 , 117 , 6)	(481 , 345 , 65)	(253 , 219 , 25)
F_antidlg_20mb_36	512x512x68	(51 , 148 , 6)	(472 , 382 , 62)	(252 , 213 , 27)
F_antidlg_20mb_37	512x512x62	(49 , 120 , 5)	(494 , 344 , 58)	(255 , 207 , 25)
F_antidlg_20mb_38	512x512x61	(10 , 148 , 3)	(503 , 371 , 57)	(254 , 217 , 23)
F_antidlg_20mb_39	512x512x64	(29 , 137 , 4)	(466 , 363 , 58)	(247 , 207 , 26)
F_antidlg_20mb_40	512x512x61	(0 , 113 , 3)	(508 , 376 , 58)	(257 , 205 , 25)
F_antidlg_20mb_41	512x512x64	(9 , 122 , 4)	(493 , 376 , 60)	(246 , 209 , 25)
F_antidlg_20mb_42	512x512x65	(0 , 118 , 4)	(506 , 366 , 61)	(258 , 208 , 26)

圖 4-10：三十四組果蠅腦的資訊

在 Template brain 的實驗中，我們需要找出一個最小 deformation field 的果蠅腦。實驗結果得到編號十四號的腦符合上述條件，而後會將其它的腦 warp 到十四號腦接著取平均變成 Template brain，下列為此 Template brain 的資訊。

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Template Brain	512x512x67	(57 , 139 , 8)	(459 , 368 , 66)	(260 , 216 , 26)

圖 4-11 標準果蠅腦的資訊

下圖為 Template brain 的其中四張影像：

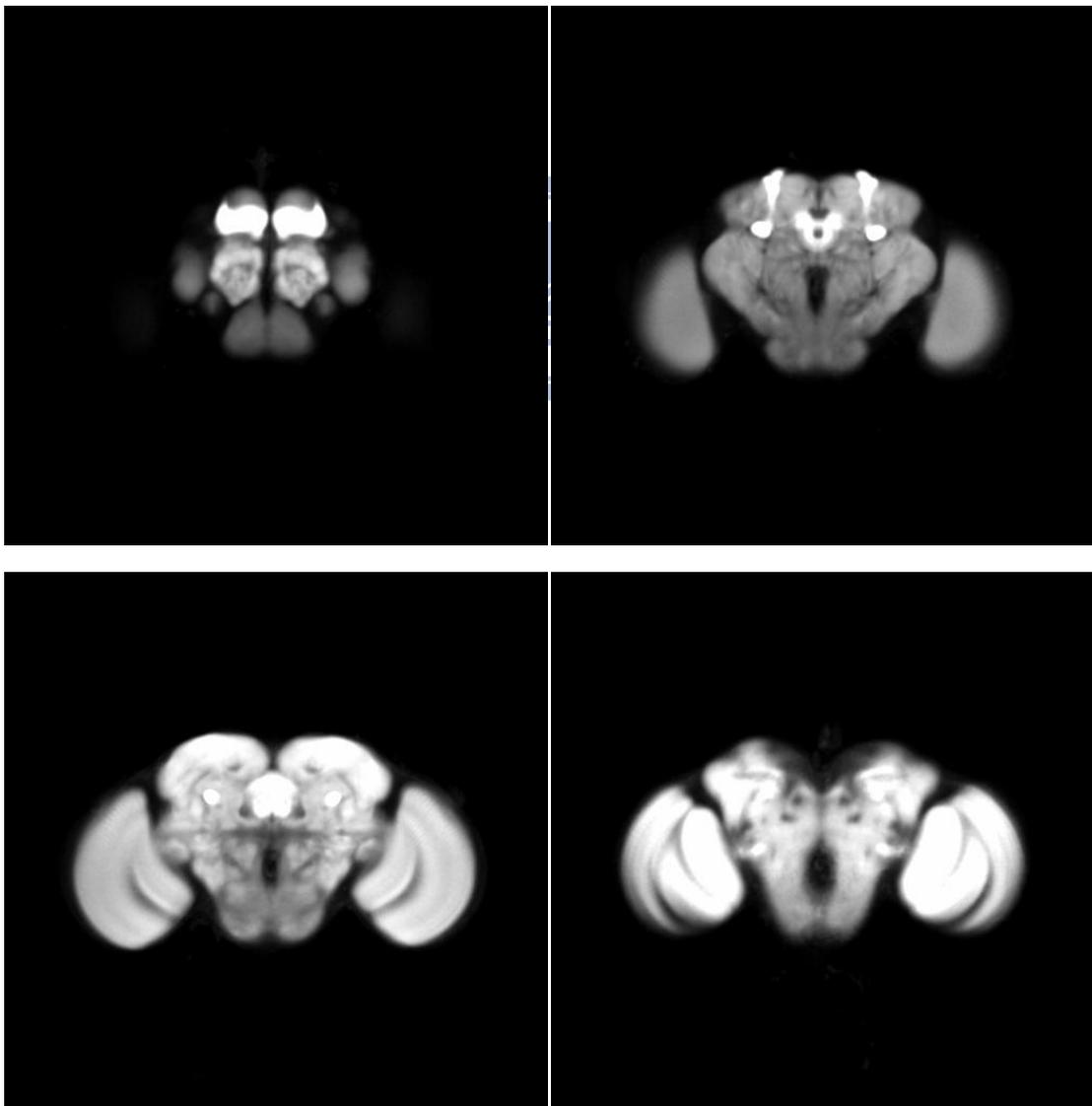


圖 4-12：Template brain

● 4.5.3 : Neuron warp

此實驗資料的原始果蠅腦為兩個 channel 檔案，在此我們需把兩個 channel 的資料分成兩個各一個 channel 的檔案。其中取出果蠅腦神經的 channel 來做 neuron warp，主要目地是要將 neuron warp 到果蠅的標準腦裡。下表為四組包含果蠅腦組織與神經的資訊。

由於神經的部份在影像區域較小且不明顯，因此經由染色後較方便觀看，下列為四組果蠅腦神經組織 warp 至 template 中，紅色外框表示其 bounding box 大小：

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdlg1200	512x512x84	(16 , 105 , 5)	(508 , 392 , 73)	(258 , 216 , 43)
Template Brain	512x512x67	(57 , 139 , 8)	(459 , 368 , 66)	(260 , 216 , 26)

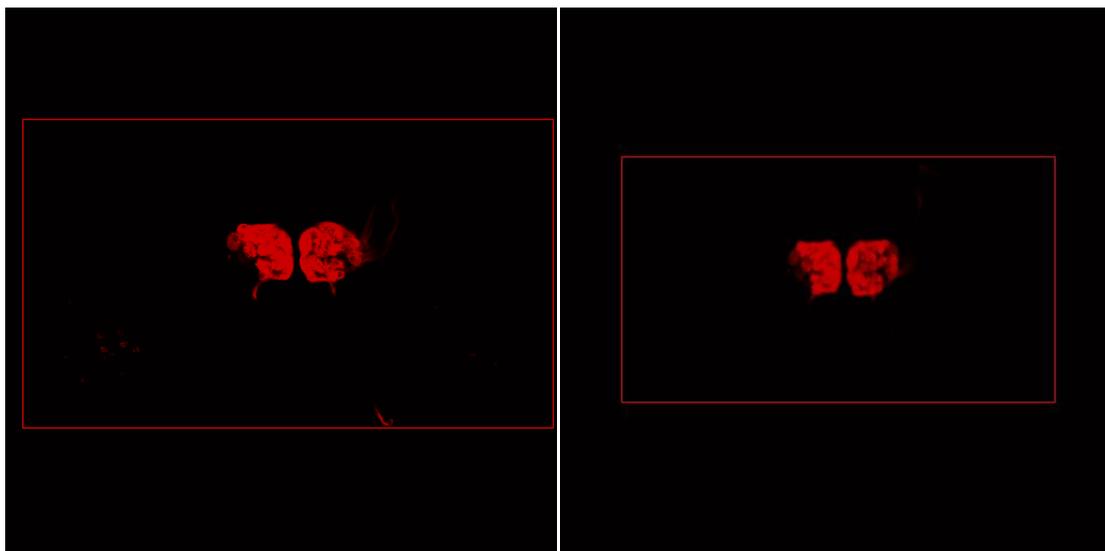


圖 4-13：檔案編號 1200 warp 至 template

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdIlg1500	512x512x67	(19 , 136 , 4)	(486 , 369 , 63)	(259 , 216 , 26)
Template Brain	512x512x67	(57 , 139 , 8)	(459 , 368 , 66)	(260 , 216 , 26)

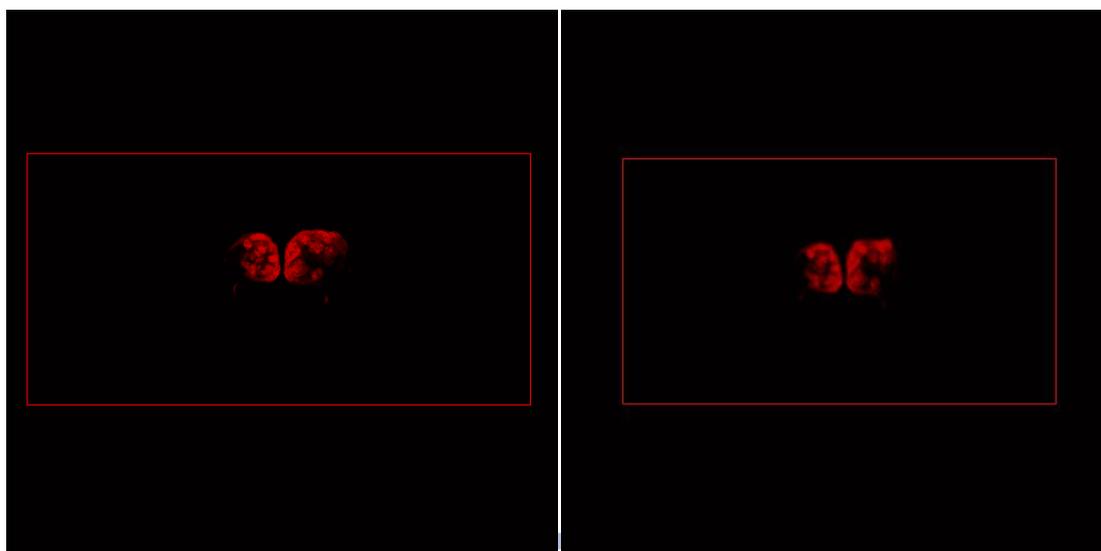


圖 4-14：檔案編號 1500 warp 至 template

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdIlg12001	512x512x87	(36 , 128 , 5)	(485 , 368 , 76)	(253 , 208 , 40)
Template Brain	512x512x67	(57 , 139 , 8)	(459 , 368 , 66)	(260 , 216 , 26)

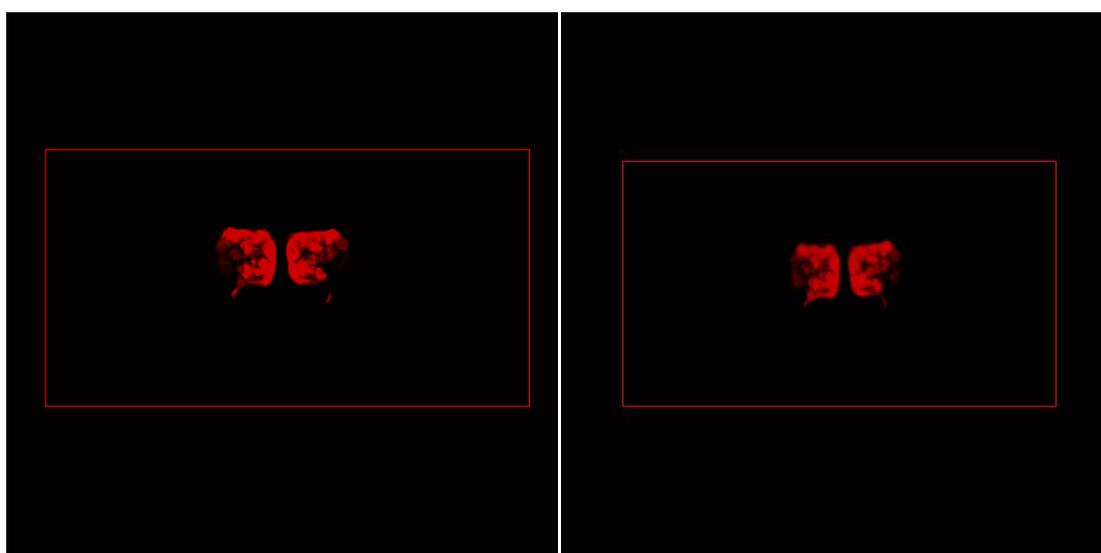


圖 4-15：檔案編號 12001 warp 至 template

檔案名稱	維度大小	Bounding Point1	Bounding Point2	Ellipsoid Body
Or83bdlg12002	512x512x81	(31 , 145 , 3)	(487 , 366 , 70)	(259 , 206 , 36)
Template Brain	512x512x67	(57 , 139 , 8)	(459 , 368 , 66)	(260 , 216 , 26)

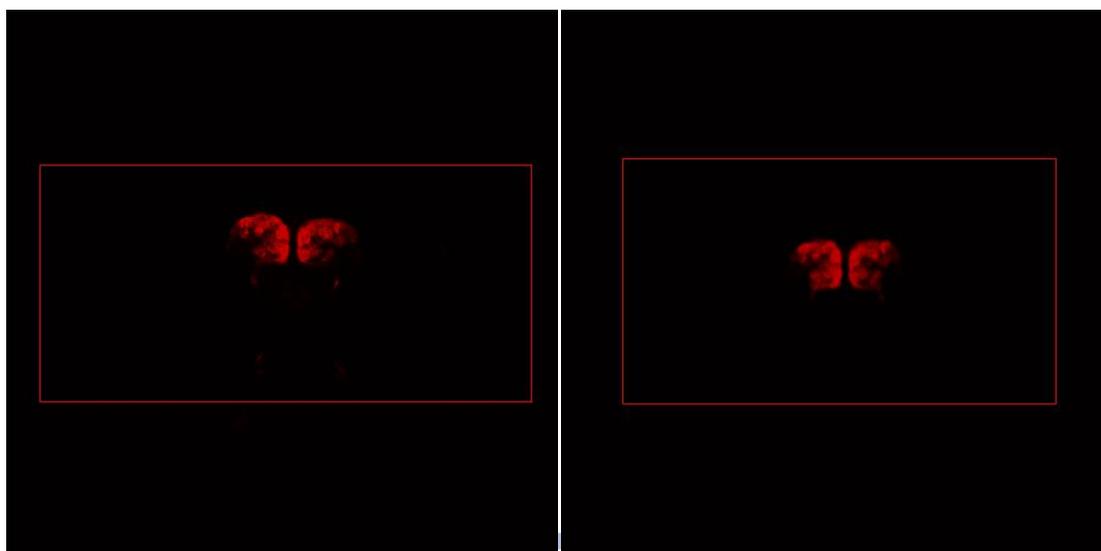


圖 4-16：檔案編號 12002 warp 至 template

下列為其中四組果蠅腦神經同時放入 template，包含五個 channel：

- 紅色部份，即為 template brain。
- 其於部份，即為四組 neuron。

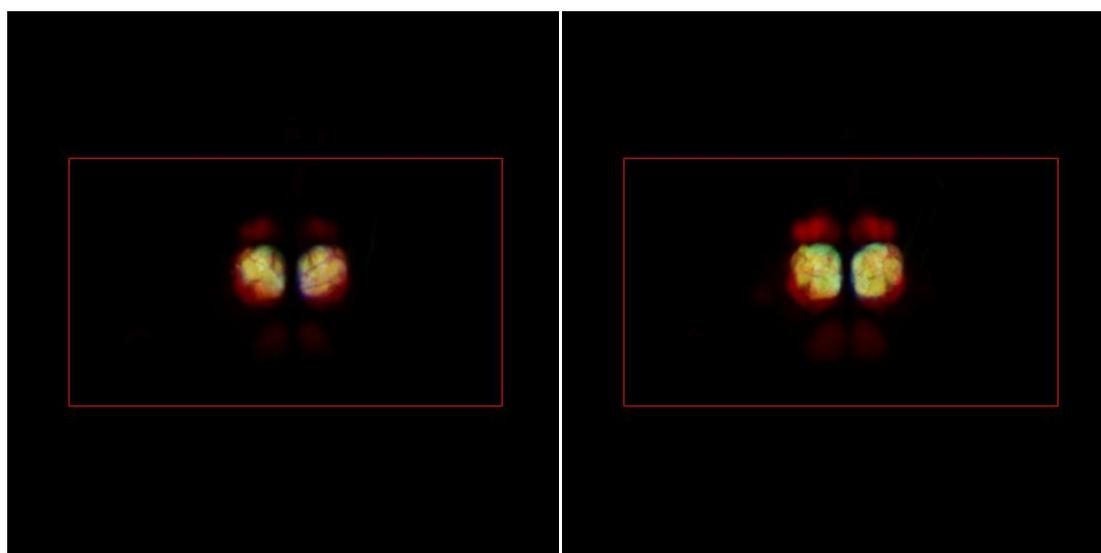


圖 4-17：Four neuron and template brain

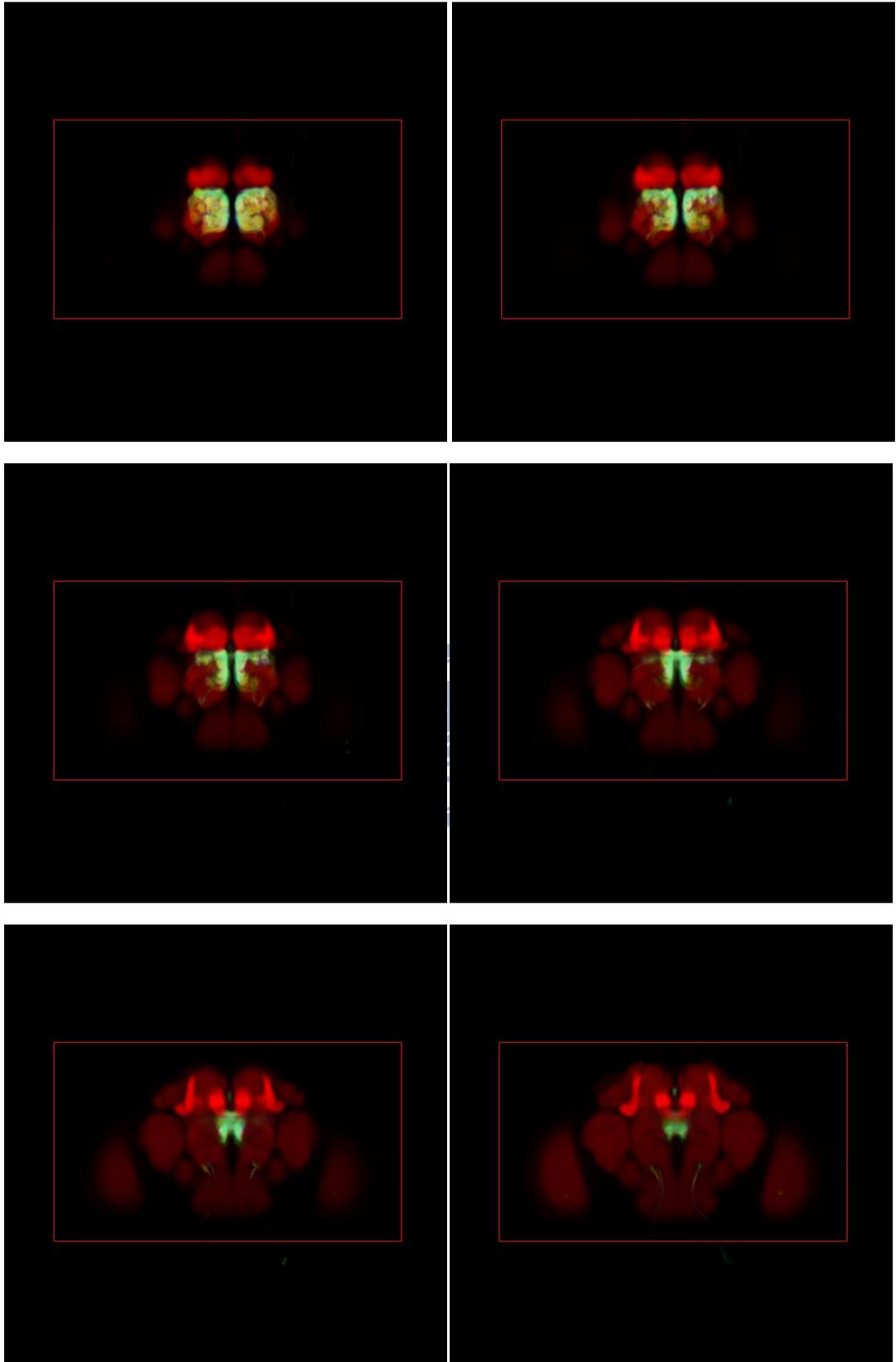


圖 4-18 : Four neuron and template brain

第五章 User interface for QT

● 5.1 : QT introduction

QT 是一個跨平台的應用程式架構，由 Havard Nord 和 Eirik Chambe-Engm 於 1994 年開發完成並創立 QT 軟體公司，為完全物件導向設計，易於擴充並且允許元件開發，可以開發一般應用程式以及使用者介面(GUI)，並且能夠很快速地部署於各種不同的桌面與嵌入式作業系統中，如圖 2.14，可以發現 QT 支援 C++與 Java，並且於跨平台支援包含以下作業系統：Microsoft Windows, Mac OS, Linux/X11, Embedded Linux 等。

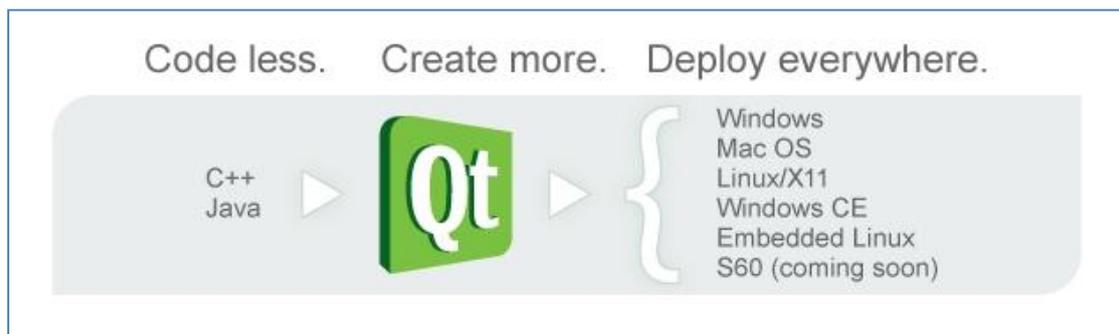


圖 5-1 : QT 開發架構

在使用者介面(GUI)程式開發上，QT 利用訊號與槽(signal/slot)機制取代傳統的 callback 來連結物件之間的溝通。所以當操作事件發生時，物件會發出一個訊號，而槽(slot)為一個函式(function)即可以接受特定訊號並且執行本身設定的動作，而訊號與槽則透過靜態的方式來連結(connect)。

為了使得制定果蠅標準腦的操作使用容易，因此我們利用QT設定出一個簡易使用者介面(GUI)，主要的內容包含：

- Affine：處理全域性的影像轉換
- Fmask：處理對果蠅腦取 mask
- Nonrigid：處理 RBFs 的區域性部份
- Applyaff：對一果蠅腦進行任意一矩陣的乘積
- Applydef：對一果蠅腦進行所有 voxel 的 deformation

● 5.2 : User interface for QT

本節為利用 QT 開發環境的使用者介面(UI)。

下圖表示讀取副檔名為.ipt、.hdr、.img 的影像及影像部份 detail 的資訊量。

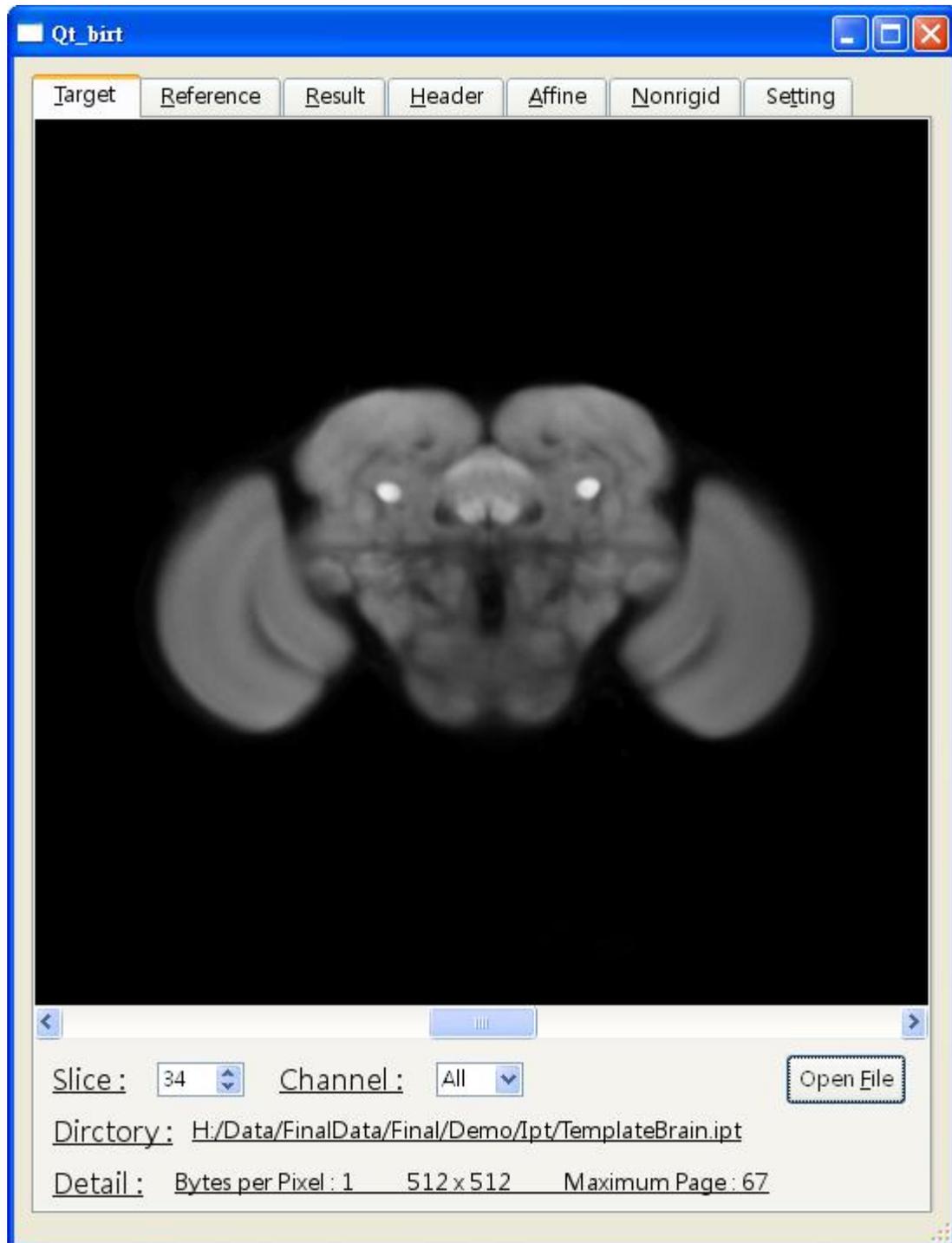


圖 5-2 : 讀取副檔名為.ipt、.hdr、.img 的檔案

下圖為處理 applyaff，applydef 等部份。

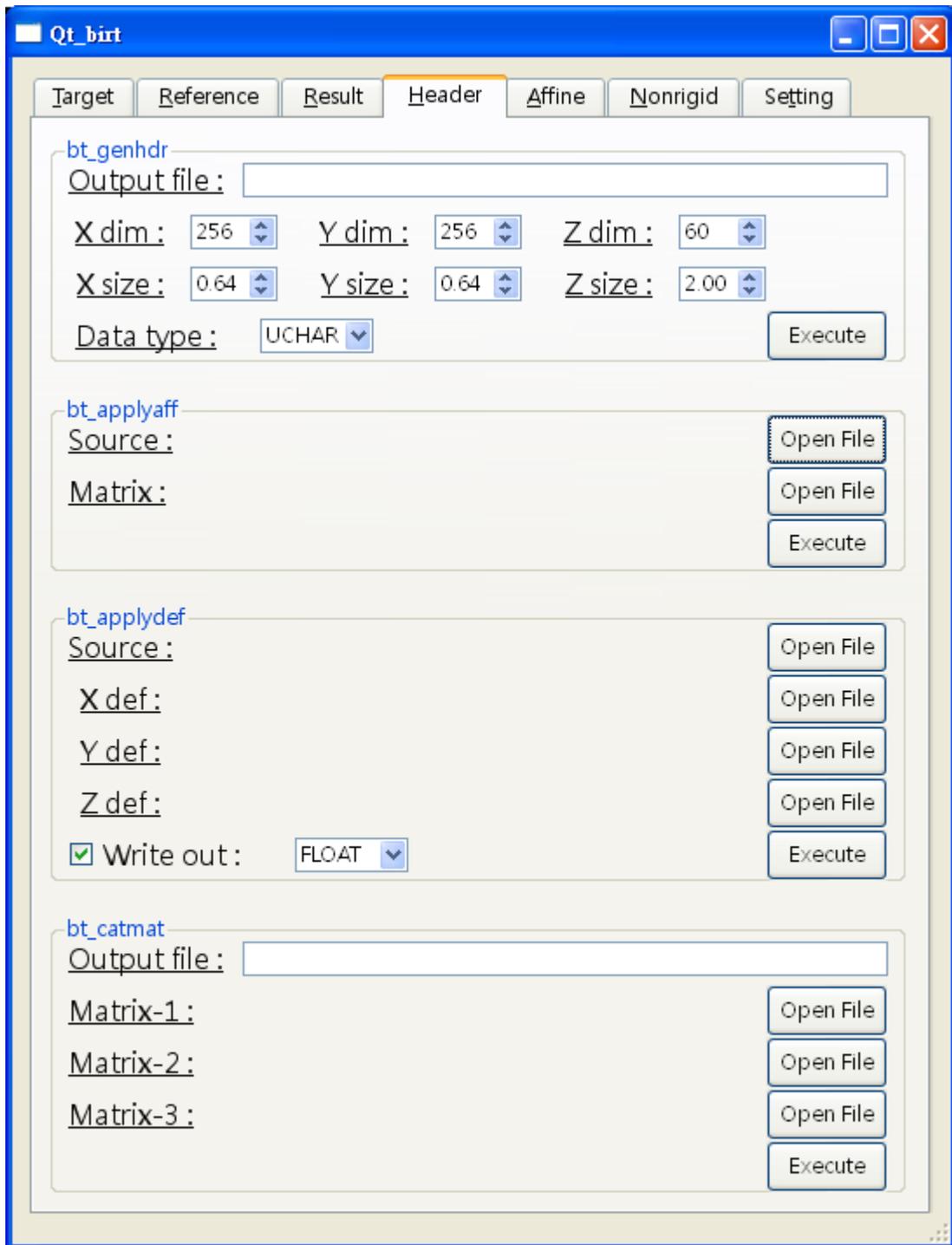


圖 5-3：處理 warp 的相關檔案

下圖為處理 affine 與 mask 的部份。Affine 的部份內容包含 reference 與 target 檔案的選取、參數的設定與選項的勾選。Mask 的部份包含 reference 檔案的選取、mask 是否選定與參數的設定。

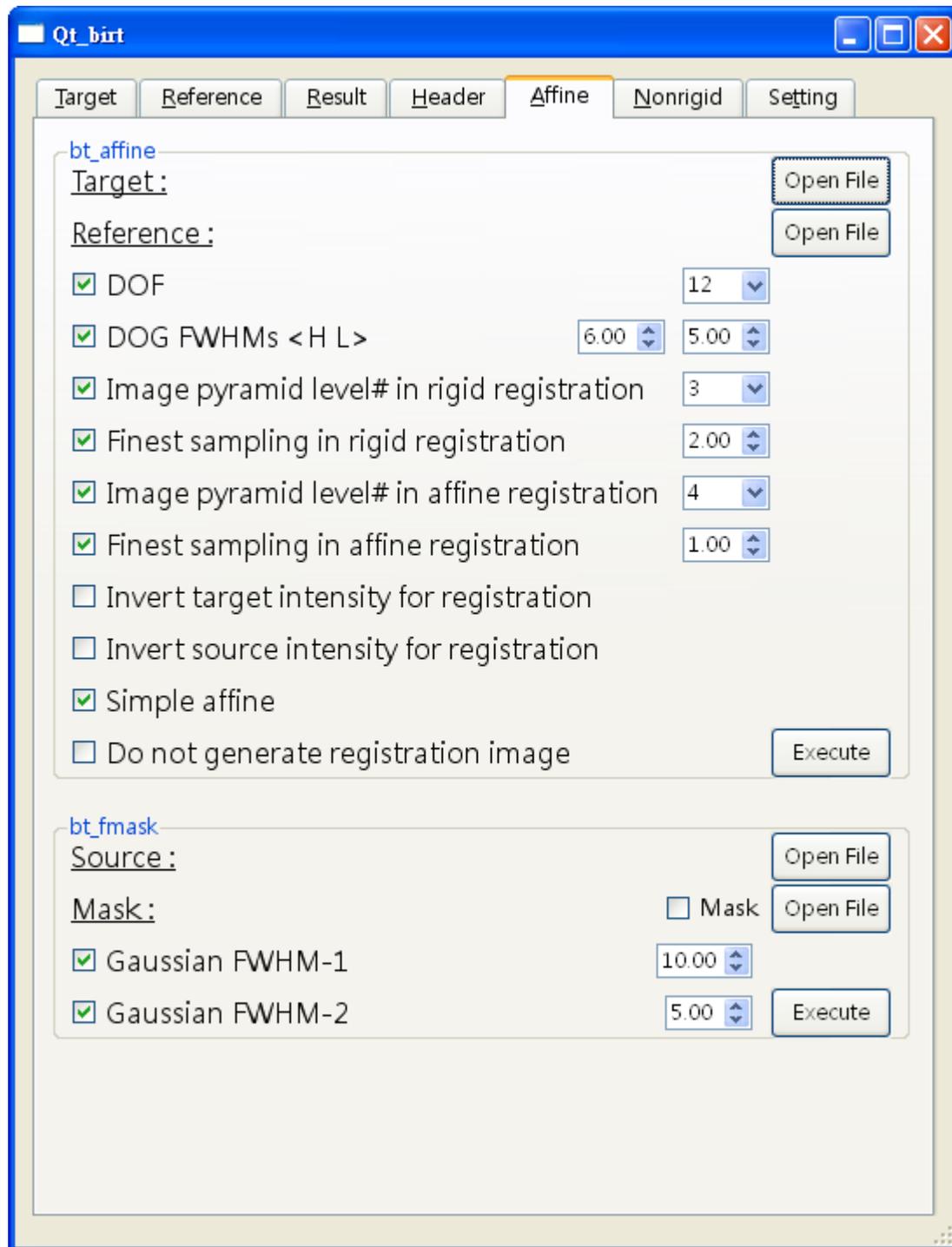


圖 5-4：處理 affine 與 mask 的部份

下列為處理 nonrigid 的部份，包含經由 affine 後 reference 檔案的選取、target 檔案的選取、由 mask 後 mask 檔案的選取、參數的設定、寫出檔案的型式、大小，與最後選項的勾選。

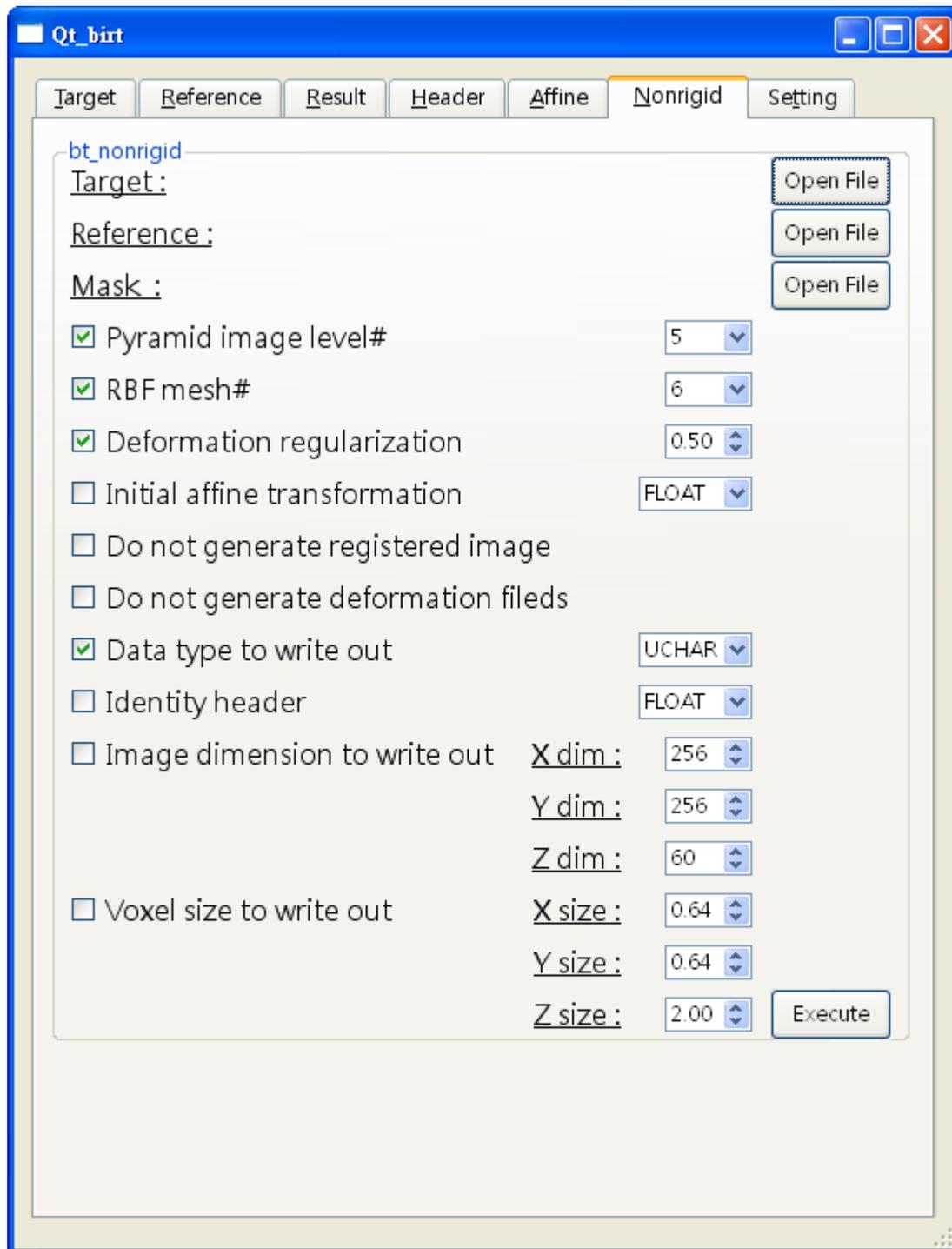


圖 5-5：處理 nonrigid 的部份

第六章 Conclusion

下列為實驗過程中所使用的環境：

- 作業系統：Microsoft windows XP3
- 開發環境：Visual studio 2005 + QT + CGAL
- CPU 時脈：Intel(R) Core(TM)2 Duo CPU E8400 3.0 GHz
- 顯示卡：nVIDIA GeForce 9800GTX
- 記憶體：3GB

在我們的實驗中，利用 interpolation 的方法進行 warp，其優勢在於不管影像檔包含有多少個 channel 都可以一次的 warp 完成，並且除了影像中的某特定區域外，其它部份的對位都很精準。而其缺點排除人為主觀的設定特徵點與三角化的過程之外，對於 warp 後 ellipsoid body 與 antennal lobe 的部份，其附近影像有浮動的感覺，使結果不盡理想。又因處理過程使用到遞迴，處理時間些微壟長。目前除了瞭解問題所在並且可待解決。

接著我們利用眾多的果蠅腦建立一組標準的果蠅腦(template brain)，除了作為往後所有果蠅腦的標準參考之外，而後可以將神經的部份建立於標準腦內，使得未來神經的 trace、尋找、與資料庫的建立等便利性，大大提升未來工作的方便性與準確性。

為了 warp 使用上的便利，我們建立一組簡單的 user interface(UI)，其中包含處理 warp 的所有功能。技術純熟後，未來可逐漸地將此介面擴增其功能。

此外，若能把 warp 的結果、標準腦與 volume rendering 結合，就可以 3D 的結構形式來做比較觀察，讓研究者更能清楚的看出 image 間的差異與細節的部份。在網路的應用上，將此系統加上資料庫與網路應用，建立 Web Server 供使用者能夠線上查詢、觀看，以 Volume Rendering 展現結果，讓系統應用更廣更完善。

Reference

- [1] L. Gabriele, Volumetric Image Analysis, chapter 3 ,pp. 53-59, Wiley, May 1998.
- [2] F. A. Sadjadi, E. L Hall, “Three-dimensional moment invariants”, IEEE Transactions on Visualization and Computer Graphics, Vol.2, no.2, pp. 127-136, March 1980.
- [3] T. L. Faber, E. M. Stokely, “Orientation of 3-D Structures in Medical Images”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.10, no.5, pp. 626-632, September 1988.
- [4] 劉佳昇,「Coordinate System and Warp for Drosophila Brain」, 國立交通大學, 多媒體工程研究所碩士論文, 民國 97 年 9 月
- [5] V. B. Le, D. T. Lee, “Out-of-Roundness Problem Revisited”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.13, no.3, pp. 217-223, March 1991.
- [6] P. K. Agarwal, B. Aronov, S. Har-Peled, M. Sharir, “Approximation and Exact Algorithms for Minimum-Width Annuli and Shells”, Annual Symposium on Computational Geometry. Proceedings of the fifteenth annual symposium on Computational geometry, pp. 380-389, Miami Beach, Florida, United States, 1999
- [7] B. Gartner, S. Schonherr, “An Efficient, Exact, and Generic Quadratic Programming Solver for Geometric Optimization”, Annual Symposium on Computational Geometry. Proceedings of the sixteenth annual symposium on Computational geometry, pp. 110-118, Clear Water Bay, Kowloon, Hong Kong, 2000
- [8] 楊博鈞,「Deform Brain Data using Feature-Based Metamorphosis」, 國立交通大學, 多媒體工程研究所碩士論文, 民國 96 年 8 月

- [9] T. Beier, S. Neely, "Feature-base image metamorphosis", Proceedings of the ACM SIGGRAPH92 Computer Graphics, Vol.26, no.2, pp. 35-42, New York, NY, USA, July 1992.
- [10] A. Leros, C. D. Garfinkle, M. Levoy, "Feature-Based Volume Metamorphosis", International Conference on Computer Graphics and Interactive Techniques. Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. pp. 449-456, 1995.
- [11] Jörn Diedrichsen · "A spatially unbiased atlas template of the human cerebellum" · Laboratory for Computational Motor Control, Department for Biomedical Engineering, Johns Hopkins University School of Medicine, Baltimore, MD 21205, USA · August 2006.
- [12] Jia-Xiu Liu, Yong-Sheng Chen, and Li-Fen Chen. Affine and nonlinear spatial normalization techniques using derivatives of brain magnetic resonance images. In Annual Meeting of the Organization for Human Brain Mapping, Austria, 2008.