

中華民國第 55 屆中小學科學展覽會

作品說明書

高中組 地球科學科

040503

螺旋？棒旋？傻傻分不清楚

學校名稱：國立馬祖高級中學

作者： 高一 蔡聿翔 高一 葉品妍 高一 曹雅涵	指導老師： 卜文強 鄭景文
---	-----------------------------

關鍵詞：螺旋星系、棒旋星系、CLEAN algorithm

摘要

我們在從眾多星系中選取了 M81 與 M109 兩個具有相似的傾斜角、且 M81 的核球占其盤面的比例與 M109 之棒狀結構占其盤面的比例相近的星系來進行研究。接著我們利用 Mathematica 程式將 CLEAN 演算法寫成程式碼，進而使用這個程式來將 M81 與 M109 放至不同的距離來觀察其影像的變化。我們發現把一個如 M109 大小(直徑約 15 萬光年)之星系置於約 1340Mpc 遠時其棒狀結構將會難以辨識而易誤判為單純的螺旋星系。

壹、研究動機

某次上課，學校請了對天文領域有所研究的老師來任課，課程內容使我們對天文產生了濃厚的興趣，便向老師請教了一些有關天文方面的問題，老師提供了我們 APOD 這個有關天文方面的網站。有天偶然瀏覽到 M31 這個星系，覺得它無比美豔，又向老師詢問了有關星系的問題，經老師一番解釋後，我們便對星系中的螺旋星系與棒旋星系，特別感興趣，所以我們自己去維基百科中查詢棒旋星系的資訊，在那裡面中我們看到一段文字寫道：棒旋星系與螺旋星系的比值隨著時間的增加而增加，且棒旋星系與螺旋星系會互相的演化。但是為什麼棒旋星系與螺旋星系的比值會持續增加還是個目前還無法解釋的問題，而有些論文中則提到分辨棒旋星系、螺旋星系的方法是以肉眼來分辨，探討兩者間互相演化的問題。所以我們思考其中的星系是否有些會因為距離太過遙遠、影像的解析度、星系盤的視角、棒狀結構太短、所觀測的波長、甚至是個人的觀點等諸多問題，進而導致誤判，進而對整體上的結果有所影響，我們便針對了其中一個因子－星系離我們的距離。這個因子所造成分辨不清的問題來做探討。

於是我們討論了假設性的問題：要是有一棒旋星系的棒狀結構比例與另一個螺旋星系的核球比例相似，且兩者離我們夠近，結構易分，傾斜角也相近時，將兩個星系放至多遠便會混淆兩者？

相關教材：

龍騰 基礎地球科學(下) 6-1 觀測星空

貳、研究目的

我們想研究將這對星系放至多遠便會產生視覺上的混淆，例如把棒旋星系中的棒狀結構誤判為核球而將其認定為螺旋星系。

所以我們選一螺旋星系(稱 A 星系)及一棒旋星系(稱 B 星系)，而此二星系需具有以下特性：

- 一、該螺旋星系(A 星系)的核球比例與棒旋星系(B 星系)的棒狀結構比例相似
- 二、此二星系離我們夠近使其結構可以清楚分辨
- 三、傾斜角相近。

參、研究設備及器材

一、資料彙整

本研究藉由 SIMBAD Astronomical Database(以下簡稱：SIMBAD)所提供梅西爾天體(Messier object，簡稱：M)、星雲和星團新總表(New General Catalogue of Nebulae and Clusters of Stars，簡稱：NGC)資料進行彙整；並由 Sloan Digital Sky Survey(SDSS)資料庫取得星系影像。

二、紀錄軟體

(一)Microsoft Excel：彙整數據

(二)Microsoft PowerPoint(以下簡稱：PPT)：量測棒狀結構與核球結構占盤面半徑的比例

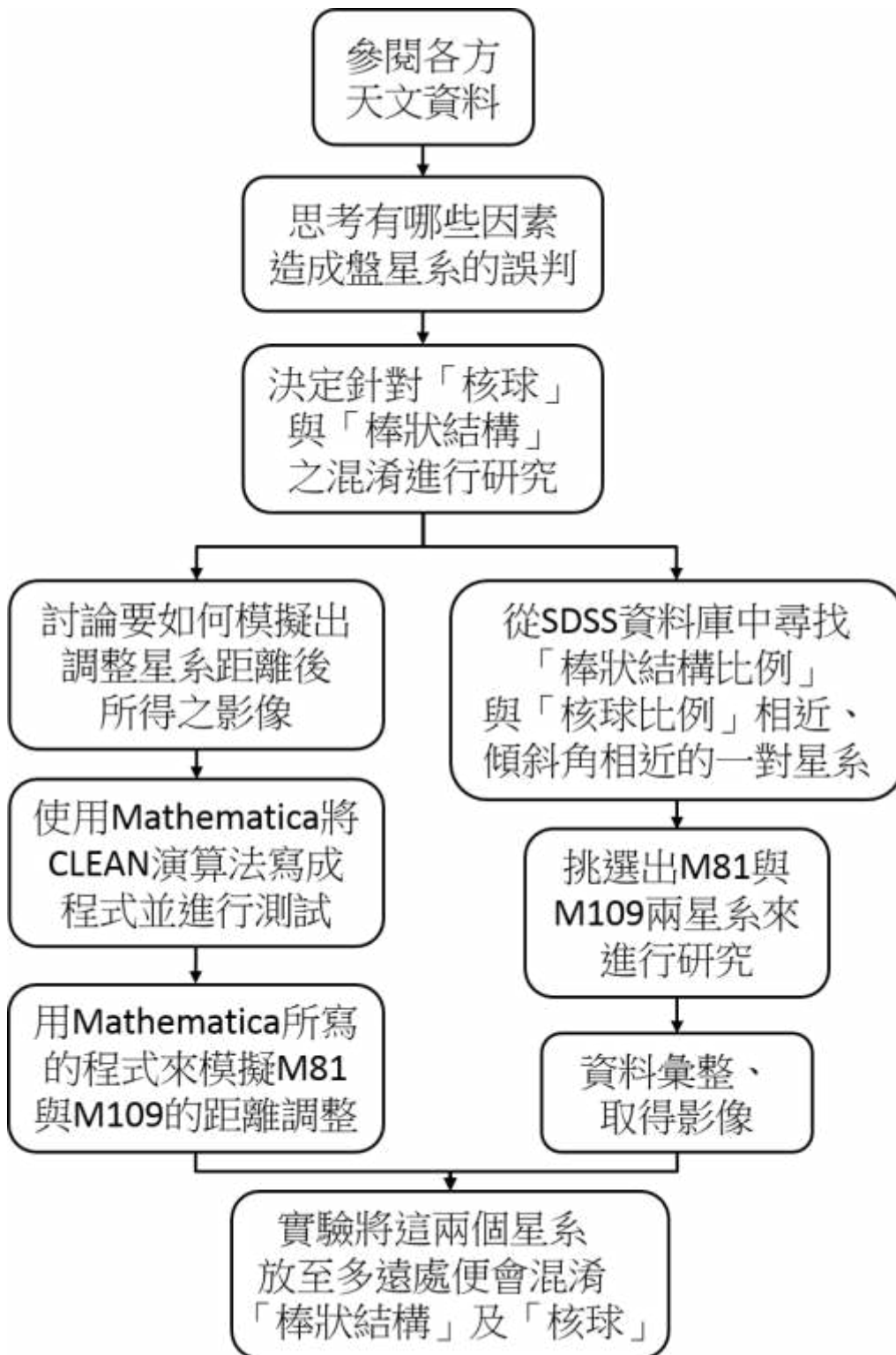
(三)Microsoft Word：進行紀錄

三、繪圖及數據處理

Mathematica 10.0.1：De-Convolution (利用 CLEAN 演算法[2])，再次 Convolution，調整星系距離所之影像皆使用此程式。

肆、研究過程或方法

一、研究過程



二、文獻回顧

(一) 星系概要

星系為構成宇宙的基本單位，由多個恆星系統(star system)、恆星(star)、星團(star cluster)、雲氣(gas)、塵埃(dust)所組成，且受重力束縛並持續繞其質量中心運轉的大質量系統。

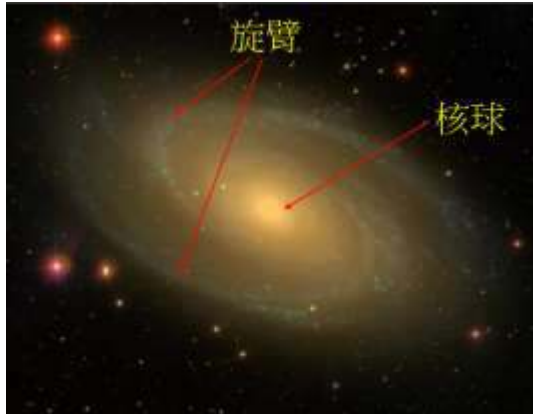


圖 1A 螺旋星系範例 Messier 81

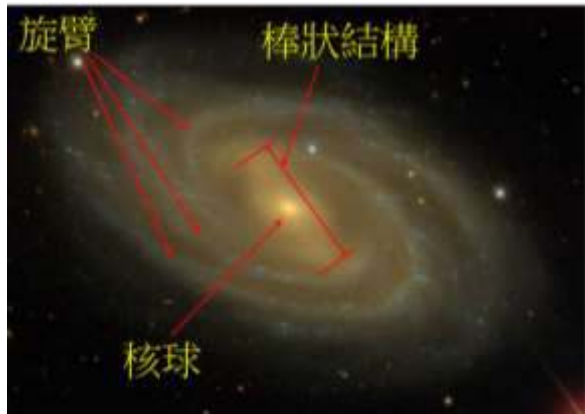


圖 1B 棒旋星系範例 Messier 109

依視覺上的形狀可將星系概分成四種：橢圓星系(elliptical galaxy)、螺旋星系(spiral galaxy)、棒旋星系(barred spiral galaxy)、不規則星系(irregular galaxy)，由於本研究主要針對螺旋星系及棒旋星系進行討論，故以下便不對另外兩種星系多做敘述。

1. 螺旋星系[3] (Spiral Galaxy): 於星系盤中有明顯的旋臂結構(spiral arm structure)，而星系盤中心有核球結構(bulge)，如圖 1A 中所呈現，名為 M81 的星系即為一例。
2. 棒旋星系[4](barred spiral Galaxy)：似螺旋星系，與之不同處在於旋臂結構與核球結構間，有長條形的棒狀結構(bar structure)，其為恆星密集組成的結構，圖 1B 中所呈現的是名為 M109 的星系，即是一棒旋星系。

(二) M81 與 M109 的基本特性表

	視角(角分)	距離(Mpc) ^{*註}	星系類型
M81 [5]	16.52 × 8.45	3.61 ± 0.06	螺旋星系
M109 [6]	6.75 × 3.78	26.8 ± 0.2	棒旋星系

表 1

註：Mpc 為 Mega-parsec(百萬秒差距)的縮寫，是星系天文上常用的一種單位。

(三) 如何調整星系的距離？

由於本研究中我們想做的是觀察兩星系調整至多遠時便會開始有結構上的

混淆，因此要如何合理的模擬出星系搬移至其他距離所應拍攝到的影像為本研究的關鍵之一。

在生活中，當我們遠離一個物體時，看起來就像在等比例縮放它，因此，直接縮小影像尺寸或許行得通。但這個做法在模擬星系影像時是不可行的，以下我們用星點的影像做說明。若直接縮小影像尺寸會造成圖 1A 變成圖 1B，而實際上拉遠星系距離後應該是圖 1A 變成圖 1C。可以看到直接縮放和理論值有很明顯的差距。

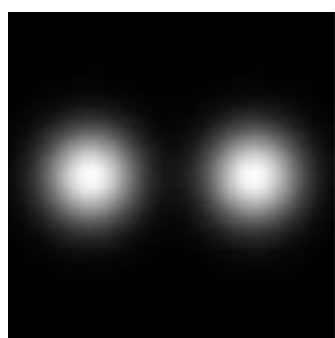


圖1A 原始影像



圖1B 直接縮放



圖1C 理論值

為什麼圖 1A 會變成圖 1C 呢？為了方便呈現，我們取圖 1A 的切面如圖 2 的高斯函數(藍)。因為兩個高斯圖形(藍)可還原成兩星點(藍)，將星點距離拉遠後(紅)，故經透鏡後應得到兩個間距更近但寬度相同的高斯函數(紅)。

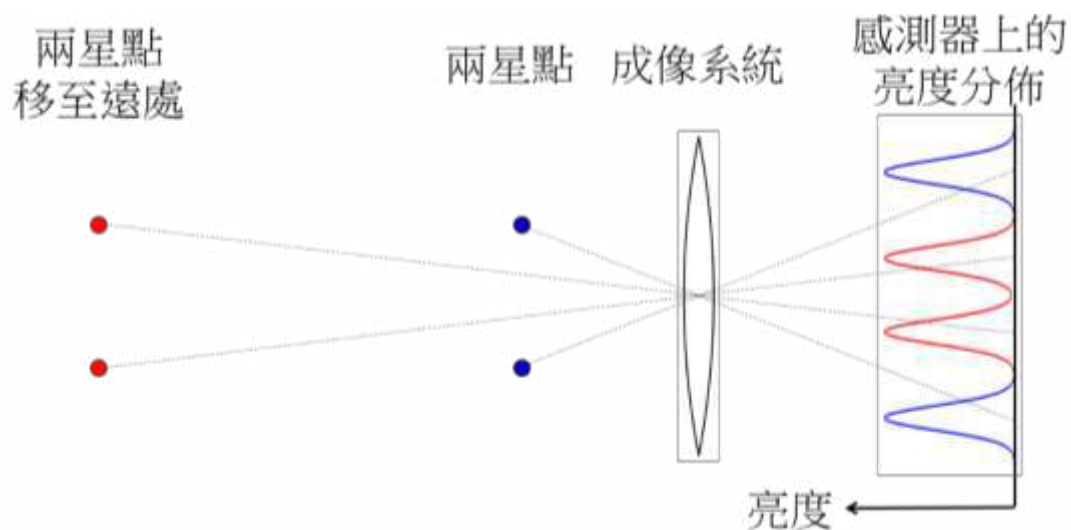


圖 2

而為什麼星點和高斯函數會有以上的對應關係呢？在這裡我們要先解釋一個專有名詞—點擴散函數。

一個成像系統的点擴散函數[7]就是點光源進入該系統後所成的像(如圖 2 中所示)。而這個像的擴散程度可以反映出一個系統的好壞。但實際上，影響點擴散函數的因素非常的多，例如光在感測原件中的擴散、太空望遠鏡的追蹤誤差等，而對於地面的望遠鏡而言，大氣擾動則是影響點擴散函數的主要原因。由於本研究中所使用的影像數據是來自 SDSS 資料庫，而 SDSS 資料庫是地面望遠鏡所拍攝出的數據，故這些影像中拍攝的星點是會受到大氣擾動的，經擾動後的像則近似於一個高斯函數(Gaussian function)圖形(圖 3)。

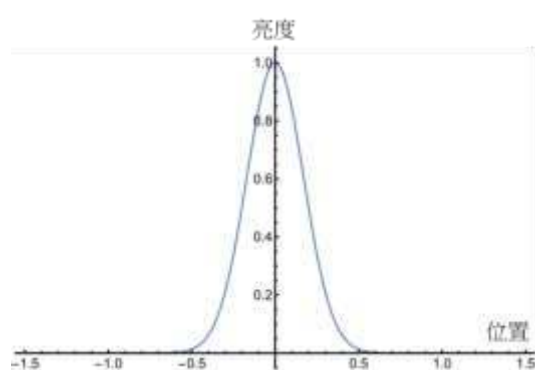


圖 3A 高斯函數的一維亮度圖

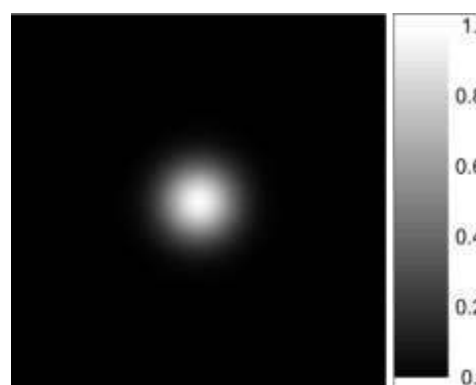


圖 3B 高斯函數的二維灰階亮度圖

既然不能直接縮小影像尺寸，那該怎麼做才能將星系移到遠處呢？

由上述中我們發現要拉遠星系的距離就必須移動星點的位置，所以關鍵在於取得原始的星點分佈，因此，在本研究中，我們先使用反摺積(decovolution)還原出原始的星點分布，將星點拉遠後，再以摺積得到新影像。不過何謂摺積和反摺積呢？下文中我們先從摺積(covolution)做說明。

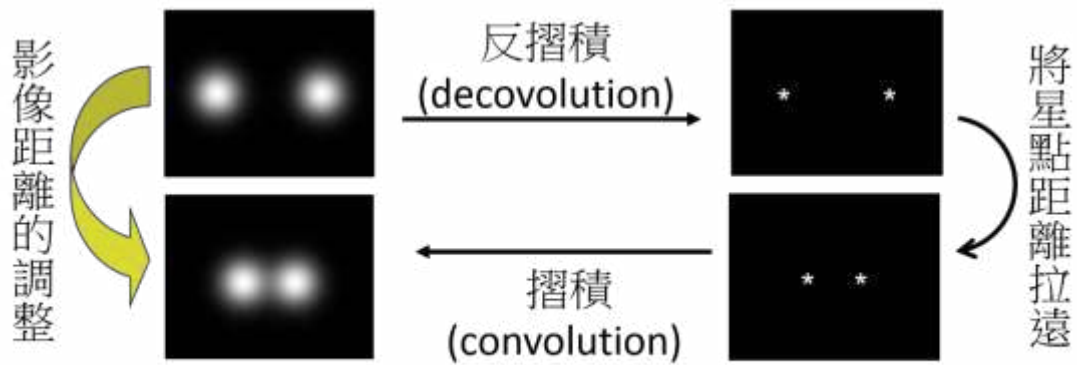


圖4 調整影像距離的基本理論示意圖

(四)摺積與反摺積

當我們同時拍兩顆星點時，兩星點所發出的光在其成像時並不會互相影響，因此，如果我們分別拍兩個星點最後將兩影像合在一起，結果會與同時拍攝兩星點相同，基於這個(不互相影響的)特性，兩個星點的成像是可以互相相加的(如圖 5)。

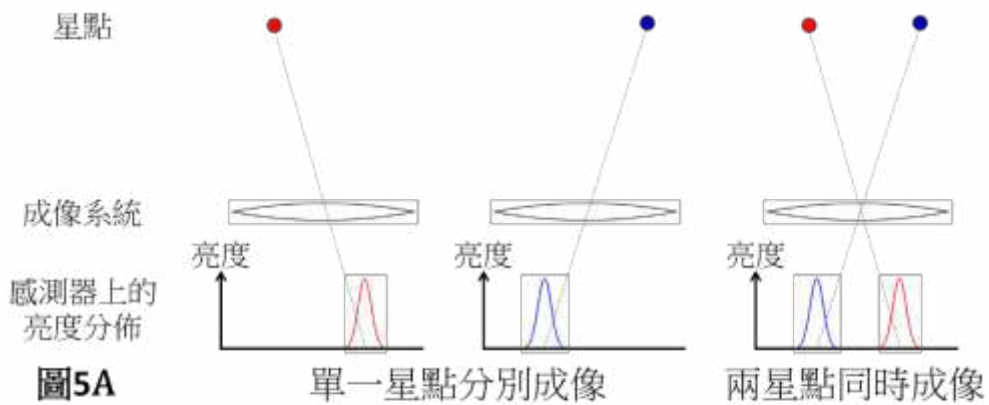


圖5A

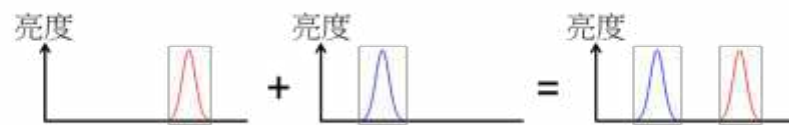
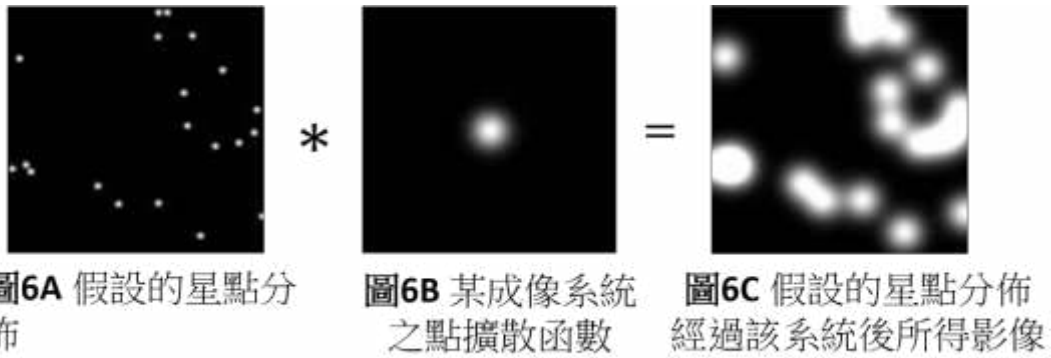
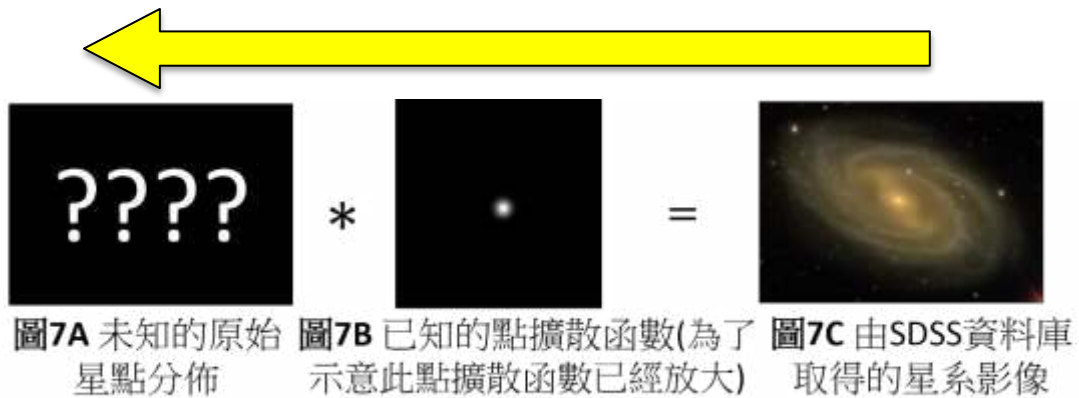


圖5B 兩單一星點分別的影像相加等於同時拍攝兩星點之影像

同理可知，若我們有一個複雜的點光源分佈(如圖 6A)，此分佈在經過透鏡後每一個點光源都會分別形成一個(不互相影響的)點擴散函數，當把這些全部加總時會得到該物體的影像(如圖 6C)，此過程等同於取此複雜物體與點擴散函數的摺積[9](Convolution)(符號以「*」代表)。



而反摺積就是摺積的相反：將拍攝到的星系影像還原為原始的亮度分佈(星點分佈)，如圖7所示。



三、程式寫作與檢測

(一)思考各種還原原始結構(進行反摺積)的方法

1. 方法1：直觀的影像分解

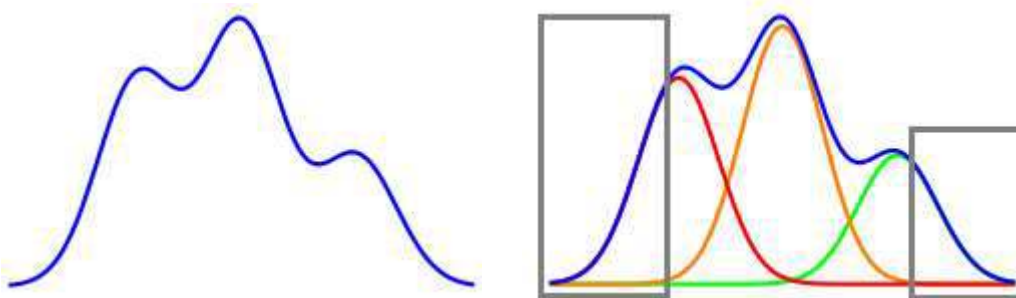


圖8 藍色線為我們所看到之影像(一維亮度切面示意圖) 圖9 灰色框框部分為幾乎不受一旁高斯干擾的區塊

在思考要如何還原出一個星系影像可能的原始星點分佈之初，我們考慮了如圖9所示的方法：假設圖形兩側皆為不受干擾區(圖中灰色框)，用不受干擾區畫出完整的點擴散函數(我們假設為一高斯函數圖形，以下將點擴散函數簡稱高斯圖形(如圖9中綠色與紅色高斯)，接著再把這兩個繪出的圖形移除掉，此時又有

不受干擾區浮現出來，可以在繼續循環、再進行移除，依此類推。但是此法問題在於高斯的高度和中心位置會受到旁邊高斯的影響而有些微的改變，且影像最左及最右的形狀可能是多個臨近的高斯所疊加而出，要如何猜測將會非常主觀。

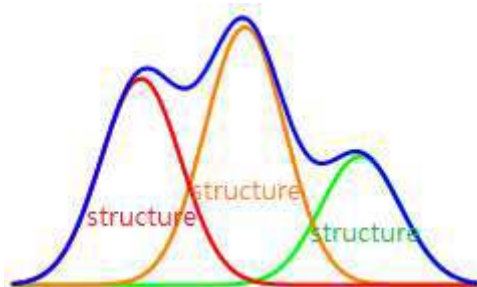


圖 10 從影像分離出多個高斯函數

2. 方法 2：用影像的點擴散函數疊出個最符合的圖形

在這個方法中我們考慮如圖 11 所示的將一張影像(粗深藍線)分解為多個點擴散函數(細、淺藍色線)所構成，這個方法的優點在於因為使用了此影像的點擴散函數，所以寬度是固定的，但對於任何一個張影像而言，會有無數多種方式可以用其點擴散函數來排列出來，因此這個方法也是相當受制於我們主觀的猜測。

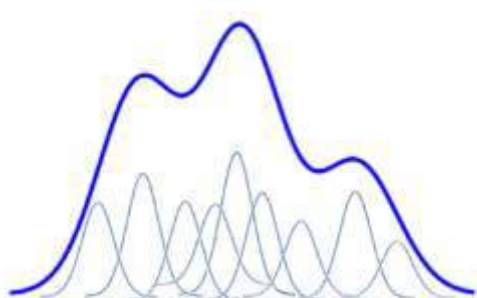


圖 11

3. 方法 3：CLEAN 演算法

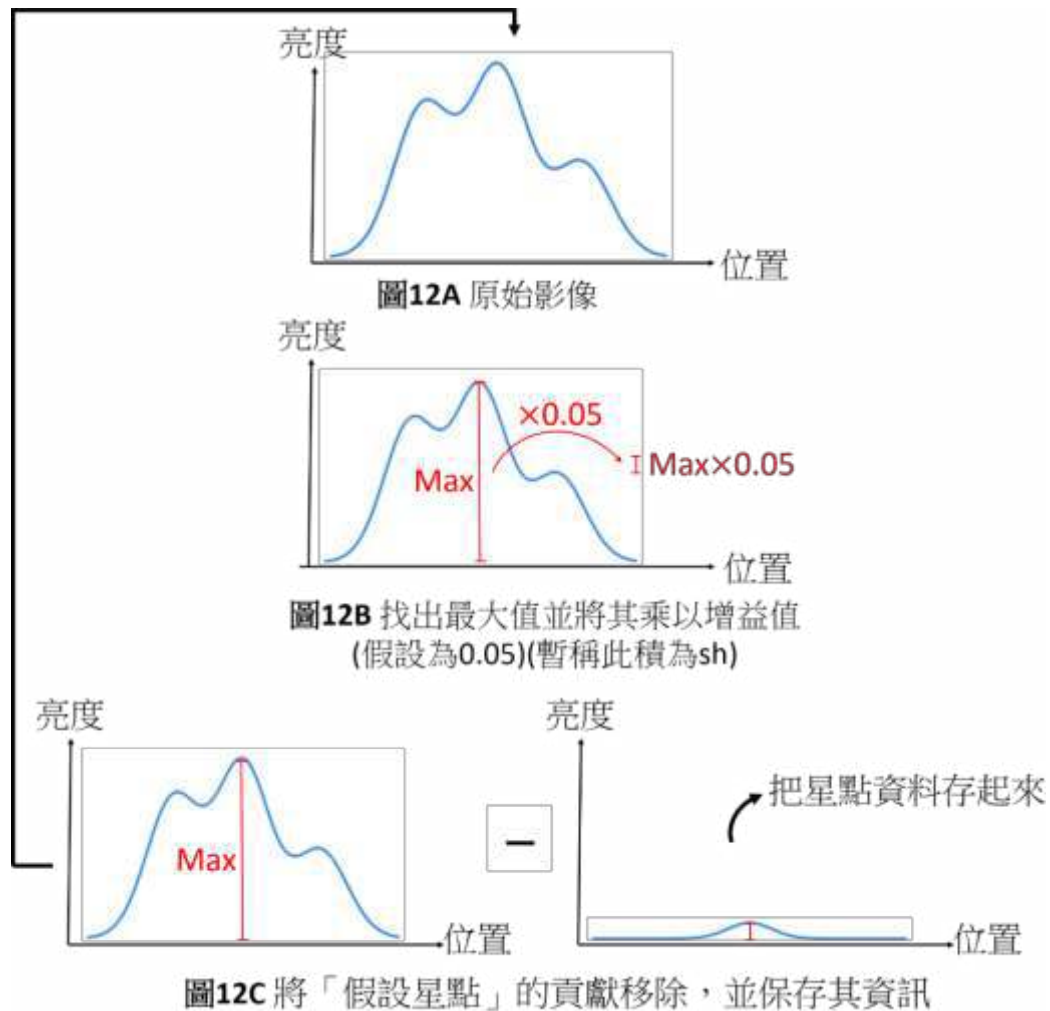
從上述方法 1 與方法 2 中，我們發現這兩種方法所產生的結果都容易是極度主觀的，因此我們思考要如何在諸多未知數中進行取捨：構成一張影像所需要的點擴散函數(高斯)個數、每個高斯的亮度、位置，都需要有所取捨。最後我們採用的是一個名為“CLEAN”的演算法。

它有兩個基本的假設：

- (1)假設影像上的所有亮度皆為點光源所造成。
- (2)假設在最大值的位置中有一個亮度為最大值乘以某個百分比(稱之為增益值(gain))的星點的存在。

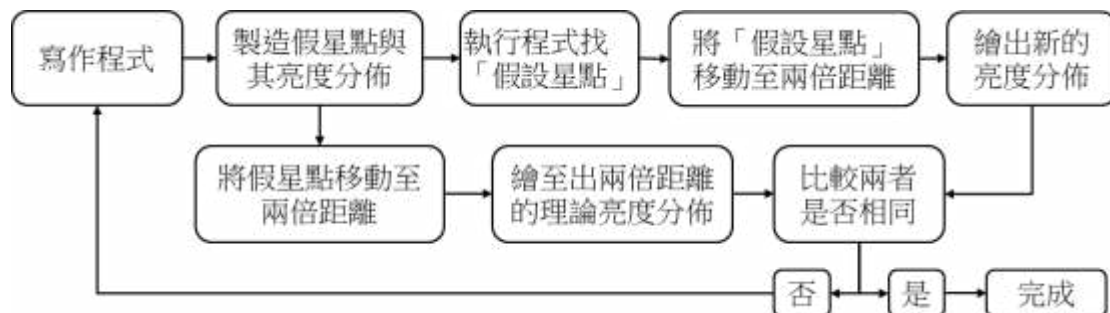
CLEAN 演算法的步驟如下：先取出亮度最大值再乘以增益值做為高斯的高度，也就是假設影像中存在有一個「最大值乘以增益值」這樣的亮度的星點的貢獻，因此我們接下來將這個「假設星點」的貢獻給移除掉，同時將此「假設星點」

的資訊保存下來，只要持續這樣操作便可以將可能的星點都找出來，這樣就同時解決了高度(亮度)與位置未知的問題，使全部的條件都成了固定值，此方式相對於前兩方法便客觀許多，只要設定好相同的增益值則都應該得到相同的結果，我們在圖 12 中以圖示的方式呈現此演算法的運作。



CLEAN 演算法的示意圖

(二)一維程式寫作與檢測



為了檢測我們所撰寫的程式是否正確，我們先在一維的狀況下進行基本測試。我們先如圖 13 製造兩個「假星點」，其中心位置分別位於-6 與+6 的座標值、高度分別為 0.7 與 1.0、點擴散函數假設為一高斯函數，標準差為 $\sqrt{3}$ 。我們預期當設定增益值(gain)為 0.05 時，程式會在前面幾個步驟中連續從位於+6 的假星點移除一些亮度，接著，當兩個假星點殘留亮度相近時，便會開始從位於-6 假星點減去。

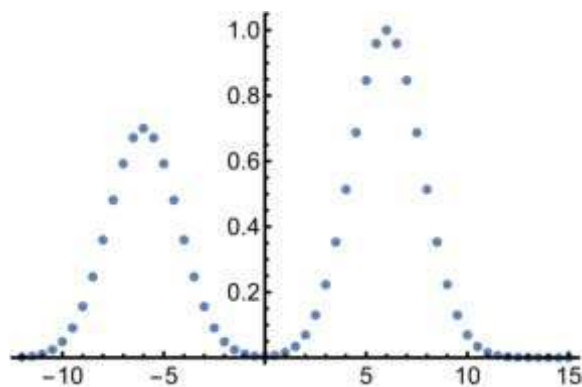


圖 13 進程式檢測用的假星點亮度分佈

以下是擷取我們的程式運算過程中幾個步驟，藍色點為被減去前的亮度分佈、橘色點為減去用的亮度分佈、綠色點為減去後的亮度分佈：

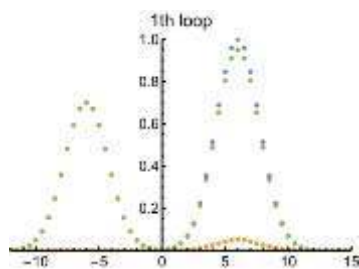


圖 14A 初始狀態

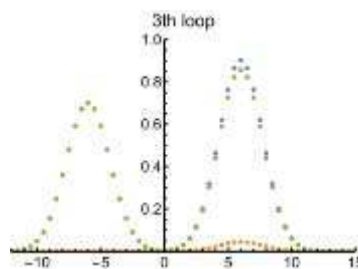


圖 14B 位於+6 的星點持續被減去

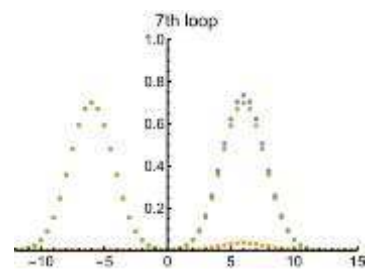


圖 14C 兩假星點之殘留亮度相近

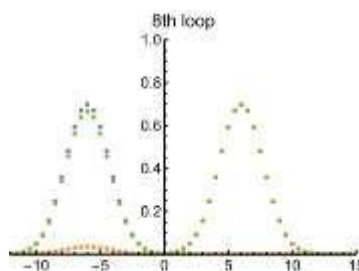


圖 14D 位於-6 的星點開始被減去

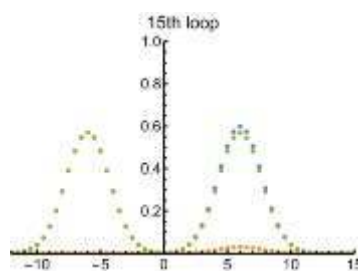


圖 14E 位於+6 與-6 的星點依序被減去

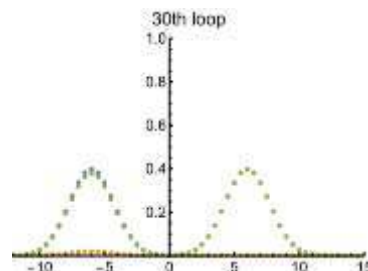


圖 14F 亮度持續下降

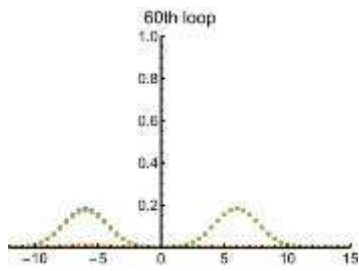


圖 14G 亮度持續下降

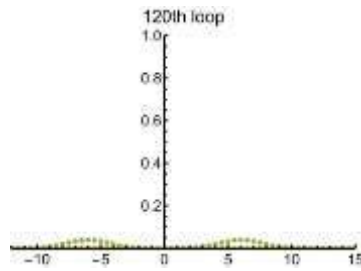


圖 14G 圖亮度持續下降

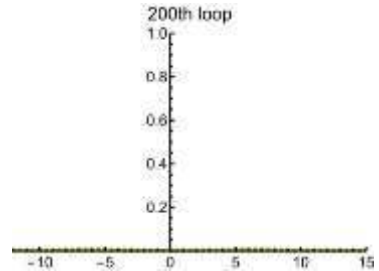


圖 14H 圖亮度降至接近零(結束)

接著，我們將所蒐集到的「假設星點」移動至兩倍距離遠，也就是將所有「假設星點」的座標值除以二，再將其與原點擴散函數(標準差為 $\sqrt{3}$ 之高斯函數)作摺積，得兩倍距離遠之「影像」。另外，由於我們已知「假星點」的初始座標，因此可以計算出將這兩個假星點移至兩倍距離遠時的理論圖形。我們將繪製出來的「影像」與理論圖形進行比較如圖 15

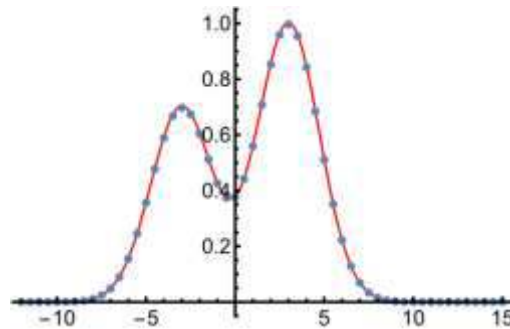


圖 15 藍色點為透過 CLEAN 過程、移動至兩倍距離、再摺積所得之影像，紅色線為理論上所得的影像，可以看到他們非常相近。

接著我們進行比較實際的測試，也就是直接用二維影像來進行以上的步驟，其流程與本章節所示流程圖相同。

(三)二維影像程式寫作與檢測

1.使用七個「假星點」來作檢測

(1)產生一幅大小 50×50 的影像，並分別在座標(47,35)、(15,27)、(2,13)、(30,47)、(10,15)、(40,5)、(25,25)處置入亮度為 1、0.5、0.8、0.9、0.2、0.85、0.7 的假星點，影像如圖 16A

(2)設定增益值(gain)為 1，並讓迴圈進行七次，七次的過程如下(箭頭指向將減去的星點)：

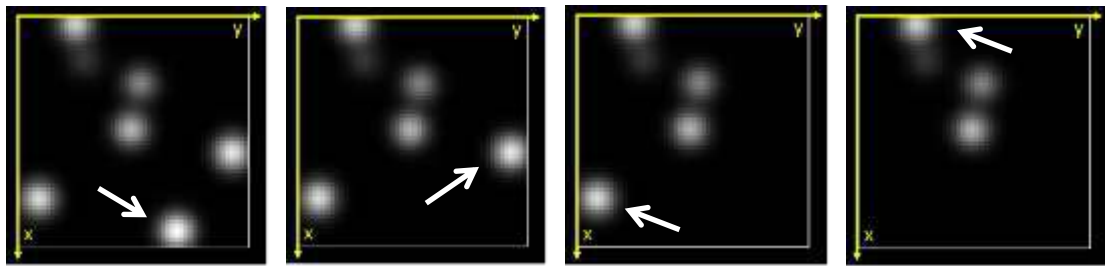


圖 16A
原始影像

圖 16B
1 個迴圈後

圖 16C
2 個迴圈後

圖 16D
3 個迴圈後

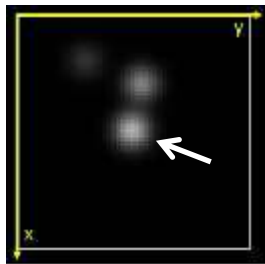


圖 16E
4 個迴圈後

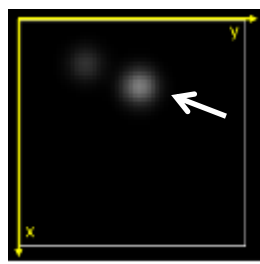


圖 16F
5 個迴圈後

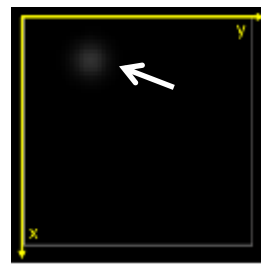


圖 16G
6 個迴圈後

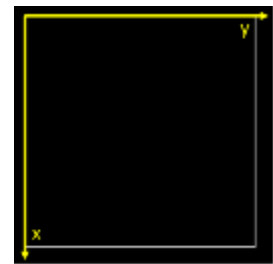


圖 16H
7 個迴圈後

- (3)將所得之星點分佈調整至 3 倍距離
- (4)輸出新影像(圖 17A)
- (5)產生理論上應該看到的影像(圖 17B)
- (6)比較(4)與(5)的結果(圖 17C)

由下圖可知，理論影像與我們所繪製出的新影像在視覺上並沒有太大的差異，由於本研究主要是透過視覺來辨識星系的型態，以及其可能造成的誤判，因此理論影像與新影像在視覺上的不可區分性便映證了我們的方法在本研究目的上的適用性。



圖 17A 新影像



圖 17B 理論影像



圖 17C 差影像

四、M81 與 M109 星系距離調整

(一)由 SDSS 取得 M81 與 M109 之影像、點擴散函數資訊(M81 的核球比例與 M109 的棒狀結構比例皆約在 0.3~0.4 之間)

	原始影像尺寸	綠光波段點擴散函數 半高腰寬(角秒)	單一像素所對應 天上的視角(角秒)
M81	2048 × 2048	1.68	0.396

M109	2048 × 2048	1.46	0.396
------	-------------	------	-------

表 2

- (二)將影像裁切至適合大小來縮短程式計算時間
- (三)利用程式將兩張影像分離成可能的「假設星點」
- (四)將兩星系移動至各種不同距離並輸出影像

伍、研究結果與討論



圖 18 M81 之影像(與下方不成比例)



圖 19A 20 倍



圖 19A 50 倍



圖 19A 100 倍



圖 19A 200 倍



圖 20 M109 之影像(與下方不成比例)

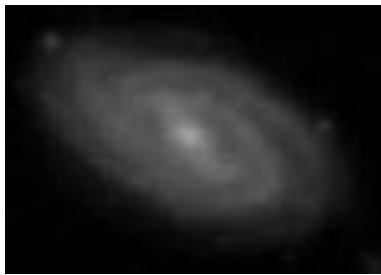


圖 21A 20 倍

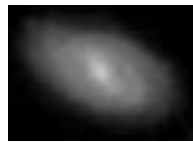


圖 21A 50 倍



圖 21A 100 倍



圖 21A 200 倍

圖 19 與圖 21 顯示了我們將 M81 與 M109 兩星系放至不同距離時所得到的影像，我們發現將 M109 放至 50 倍距離遠時就容易將其棒狀結構看成核球結構(由圖 28E 可知)，同時也發現將 M109 放至 50 倍距離的影像與將 M81 放至 100 倍距離遠的影像頗為相像。這意味著，對於一個如 M109 一般直徑約 15 萬光年的星系、傾斜角(盤面與我們視線的夾角)約 30 度、棒狀結構占整體半徑約 0.3~0.4 的話，將這樣一個星系放在約 1340Mpc(M109 原始距離的 50 倍)遠時即會難以分辨其真面目究竟是棒旋星系還是螺旋星系。

因 M109 移動至 50 倍距離後大約距離我們有 1340Mpc 之遠，其所對應的大約是宇宙年齡八十億年之時(距今約 50 億年)，因此可得知：如果我們用 SDSS 的

資料庫來研究比較早期宇宙中的早期的星系的話，到了約 50 億年前的宇宙中的星系，差不多即為不可分辨棒旋星系及螺旋星系的距離，故我們估計用 SDSS 資料庫裡的資料來討論約宇宙年齡 $2/3$ 前的星系並非良策。

我們要是想討論比 M109 更遙遠的星系，首先需要的就是更高解析度的影像資料，由於 SDSS 的望遠鏡是放在地面上的 2.5 公尺望遠鏡，易受大氣擾動，所得的影像解析度較差，在綠色波段下其點擴散函數的半高腰寬約 1.5 角秒。如果用哈伯望遠鏡的話，一樣是直徑為 2.5 公尺的望遠鏡拍攝，但因其處於太空中故不受大氣擾動的影響，因此其解析度較 SDSS 的數據來說有 10 倍以上的增進，可以達到 0.06 角秒，這樣比較適合用來研究宇宙早期的星系。

陸、總結

一、我們由眾多梅西爾(Messier)星系中選取出了 M81 與 M109 這對星系來進行研究，M81 是個螺旋星系而 M109 則是一個棒旋星系，M81 的核球比例約 0.3~0.4，而 M109 的棒旋結構比例亦約 0.3~0.4。

二、我們使用數學軟體 Mathematica 來將 CLEAN 演算法、星點距離調整寫成程式並進行測試。

三、我們將 M81 與 M109 之影像放入 Mathematica 中並匯出不同距離下所模擬出來的影像，這讓我們發現當一個大小約 15 萬光年大小的星系(如 M109)移動至 1340 百萬秒差距(Mpc)遠時便會難以辨識出它的棒狀結構。

柒、未來展望

在本研究的過程中，我們檢測了固定傾斜角、固定比例的一對星系，而我們發現其時可已造成誤判因子其實相當的多元，例如距離太過遙遠、星系的傾斜角、棒狀結構太短等問題，這些問題都可以用這麼模擬方法進行研究，並作更深入的探討。

另外，我們也因參考了文獻[10]，得知棒旋星系的的比例較早期宇宙增長了不少，也想了解為什麼棒旋星系的的比例會隨宇宙年齡的增加而增長，而透過我們的方法，可以估計出判斷一張遙遠星系的影像為某一種結構是否是有意義的，因而作出更精確的棒旋比例判斷。

捌、參考資料

- [1] Ripples in a Galactic Pond. (October, 2005) Scientific American
- [2] Högbom, J.A. (1974). Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines. *Astronomy and Astrophysics Supplement*, 15, 417-426.
- [3] Spiral galaxy. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Spiral_galaxy
- [4] Barred spiral galaxy. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Barred_spiral_galaxy
- [5] Messier 81. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Messier_81
- [6] Messier 109. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Messier_109
- [7] Point Spread function. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Point_spread_function
- [8] Airy Disc. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from http://en.wikipedia.org/wiki/Airy_disk
- [9] Convolution. (n.d.). In *Wikipedia*. Retrieved May 19, 2015, from <http://en.wikipedia.org/wiki/Convolution>
- [10] Sheth, K. (2008). Evolution of the Bar Fraction in COSMOS: Quantifying the Assembly of the Hubble Sequence. *The Astrophysical Journal*, 675, 1141-1155.

玖、附錄

附錄一、二維影像用程式碼改良過程概述

附錄二、改良前程式碼

附錄三、改良後程式碼

附錄一、

一開始用 Code 寫 CLEAN 演算法時，我們是最「直接」的方式，將其轉換為程式看得懂的語言，也就照著論文裡所說的内容逐字的「翻譯」出來，但後來我們發現以這樣的方式來計算一個影像約 400×300 的星系之反摺積就要花上將近 7 天的時間，於是我們開始改良它，測試各種方法，使時間能壓縮到最短，而不會改變 CLEAN 演算法的本質。

整體上程式碼的大略過程為：

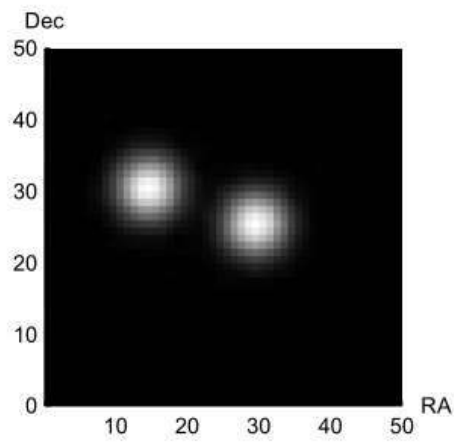
1. 取出最大值。
2. 最大值乘上一個百分比(我們是取 5%)，作為減掉用的高斯的高度。
3. 從影像中減掉該比例的高斯、將減去用的高斯之資訊存下來。

以下是我們的改良過程：

一開始我們是直接產生一個維度(ex:一張影像為“ 400×300 ”此為維度)與原圖相同的高斯(高斯是一個中心亮度有特定寬度而過了中心亮度的特定寬以外，四周的數值是無限延伸的，而此數值小於 10 的-12 次方的函數)再相減，照這樣跑一迴圈要花上 3 秒，所以我們只取出中心位置向外延伸 5~7 個標準差的範圍(將以外的值都假設為零)，再選取出要在原圖上減掉的範圍去相減，另外，為了不要使它每次需要剪掉高斯時都要製造一個新的高斯(只有高度不同，而標準差相同)而浪費時間，所以我們在前面是先製造了一個高斯的模板，要更改高度時只要在乘上最高點高度的百分比、加上它應該在原圖上的哪個位置減去就能完成這個操作。

附錄二、改良前

```
(*測試用假星點數據*)  
{xmin, xmax, dx, ymin, ymax, dy} = {1, 50, 1, 1, 50, 1};  
 $\sigma = 3$ ;  
data = Table[{x, y,  $e^{-\left(\frac{(x-25)^2}{2\sigma^2} + \frac{(y-30)^2}{2\sigma^2}\right)}$  +  $e^{-\left(\frac{(x-20)^2}{2\sigma^2} + \frac{(y-15)^2}{2\sigma^2}\right)}$ },  
  {x, xmin, xmax, dx}, {y, ymin, ymax, dy}];  
Show[Image[data[[All, All, 3]]] (*only plot the z part*),  
  Axes → True, AxesLabel → {"RA", "Dec"}, ImageSize → Small]
```



```

residual = data; (*把上面的資料讀下來*)
stars = {}; (*放星點的框*)
gain = 0.05; (*百分比*)
ti = AbsoluteTime[]; (*計算所花的時間*)
(*For迴圈*)
For[i = 1, i ≤ 2, i++,
  Print["Loop ", i]; (*顯示:Loop→文字,i的值*)
  h = Max[residual[[All, All, 3]]];
  (*找出資料(residual)中第三層(z軸,亮度)的最大值*)
  Print["最大值 = ", h];
  maxpos = Position[residual[[All, All, 3]], h][[1]];
  (*找出h在陣列中的位置,叫做"maxpos"*)
  Print["最大值h出現在 ", maxpos, " 位置"]; (*最大值出現在maxpos格子點位置*)
  {xmaxpos, ymaxpos} = residual[[maxpos[[1]], maxpos[[2]]][[1 ;; 2]];
  (*個別取出最大值的x,y的值*)
  Print["最大值h出現在座標值 ", {xmaxpos, ymaxpos}, " 處"];
  sh = gain · h; (*製造等等要拿來減原圖的高斯高度*)
  Print["小高斯的高度 = ", sh];
  sg =
    Table[{x, y, sh × e-((x-xmaxpos)2/2 - (y-ymaxpos)2/2)}, {x, xmin, xmax, dx}, {y, ymin, ymax, dy}];
  Print["產生小高斯"];
  residual = Transpose[{residual[[All, All, 1]], residual[[All, All, 2]],
    residual[[All, All, 3]] - sg[[All, All, 3]]}, {3, 1, 2}];
  (*把原圖與製造出的高斯相減*)
  Print["減去完成"];
  AppendTo[stars, {xmaxpos, ymaxpos, sh, σ}];
  (*把上面減掉的星點裝進"stars"這個框*)
  Print["將星點加入列表"];
  Print[Show[Image[residual[[All, All, 3]]] (*只畫z部分*),
    Axes → True, AxesLabel → {"RA", "Dec"}, ImageSize → Small]];
  (*畫已減過高斯後的圖*)
]

finimg = Table[{x, y,
  Sum[stars[[j, 3]] · e-((x-stars[[j, 1]])2/2stars[[j, 4]]2 + (y-stars[[j, 2]])2/2stars[[j, 4]]2), {j, 1, Dimensions[stars][[1]]}],
  {x, xmin, xmax, dx}, {y, ymin, ymax, dy}]; (*把被剪下的"星點"全部加起來*)

finimg[[All, All, 3]] + residual[[All, All, 3]] == data[[All, All, 3]] (*驗證*)
True

```

附錄三、改良後程式碼

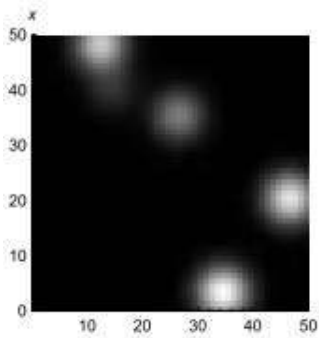
```

(*測試用數據*)
{xmin, xmax, dx, ymin, ymax, dy} = {1, 50, 1, 1, 50, 1};
σ = 3.2;

test2 = Table[{{x, y, e-((x-47)2 + (y-35)2 / (2σ2 + 2σ2)) + 0.5 e-((x-15)2 + (y-27)2 / (2σ2 + 2σ2)) + 0.8 e-((x-2)2 + (y-15)2 / (2σ2 + 2σ2)) +
0.9 e-((x-30)2 + (y-47)2 / (2σ2 + 2σ2)) + 0.2 e-((x-10)2 + (y-15)2 / (2σ2 + 2σ2))}}, {x, xmin, xmax, dx}, {y, ymin, ymax, dy}];

data = test2;
Dimensions[data]
Show[Image[data[[All, All, 3]]] (*only plot the z part*),
  Axes → True, AxesLabel → {y, x}, ImageSize → Small]
{50, 50, 3}

```



```

(*****
*****)

residual(*殘留, just a name*) = data; (*把上面的資料讀下來*)
stars = {}; (*放星點的框*)
gain = 0.5; (*取的百分比*)
lim = 2; (*多少個σ*)

ti = AbsoluteTime[];
(*製造模板σσ*)
width = Ceiling[lim σ]; (*一個σ是3.5格, lim=寬度有幾格*)
fullwidth = 2 · width + 1; (*模板寬度*)
center = width + 1; (*從右邊算過來的中心位置*)
master = Table[e-((x-center)2 + (y-center)2 / (2σ2 + 2σ2)), {x, 1, fullwidth, dx}, {y, 1, fullwidth, dy}];
(*製造高斯模板*)
PrintTemporary[Row[{ProgressIndicator[Dynamic[i / loops]],
  " ", Dynamic[N[(100 i) / loops, 3] (*顯示三位數*)]}, "%"]];
(*看它跑的進度"條"*)

loops = 13; (*迴圈數*)
For[i = 1, i ≤ loops, i++,
  Print["Loop ", i];
  h = Max[residual[[All, All, 3]]]; (*原圖中z軸(亮度)的最大值, h*)
  Print["max value值 = ", h];
  {xmaxpos, ymaxpos} = Position[residual[[All, All, 3]], h][[1]];

```

(*最大值的座標*)

```
Print["max value h appears at physical position ",
      {xmaxpos, ymaxpos}, " within image"];
sh = gain · h; (*要拿去減的小高斯的高度*)
Print["小高斯高度 = ", sh];
(*設條件,如果"撞牆"時x寬度要等於xmaxpos.....*)
xwidth = width; ywidth = width; widthx = width; widthy = width;
If[xmaxpos < xwidth+1, Print["撞上"];
   xwidth = xmaxpos - 1]
(*"-1"是減掉maxposition原本佔的那一格,因為我們要算的是"到邊界的寬度"*);
Print["xwidth ", xwidth];
If[ymaxpos < ywidth+1, Print["撞左"];
   ywidth = ymaxpos - 1];
Print["ywidth ", ywidth];
If[xmax - xmaxpos < widthx, Print["撞下"];
   widthx = xmax - xmaxpos]
(*這邊maxposition原本佔的那一格已經減掉了所以就不用再減了*);
Print["widthx ", widthx];
If[ymax - ymaxpos < widthy, Print["撞右"];
   widthy = ymax - ymaxpos];
Print["widthy ", widthy];
(*模板上各個方向width的數值,不越界*)

residual[[ xmaxpos - xwidth ;; xmaxpos + widthx,
            ymaxpos - ywidth ;; ymaxpos + widthy, 3]] -= sh ·
  master[[ center - xwidth ;; center + widthx, center - ywidth ;; center + widthy]];
(*residual - sh · master = residual(剪完後直接改變residual數值)*)
Print[Dimensions[ residual[[ All, All, 3]]]];
(*residual維度,驗證用*)
AppendTo[ stars, {xmaxpos, ymaxpos, sh,  $\sigma$ }; (*裝星點*)
Print["Add Stars to list"];
Print[ Image[ residual[[ All, All, 3]]]]; (*畫出剪完後的圖*)
]
tf = AbsoluteTime[];
Print["Computation done in ", tf - ti, " seconds"];
```

【評語】 040503

分辨螺旋與棒旋星系型態具學術價值，建議主題與使用方法能更相關結合。