國立臺灣師範大學
資訊工程研究所碩士論文

指導教授： 黃　文　吉　博士

自動對焦 3D 數位全像重建硬體系統

3D Digital Holography Hardware Reconstruction System with Autofocusing

研究生： 黃 元 品　撰

中華民國　一百零四　年　六　月

中文摘要

本論文旨在 FPGA(Field Programmable Gate Array)平台上設計可自動對焦之 3D 數位全像圖重建硬體系統。設計此類系統的一大困難點，在於需要考慮數位全像圖的正確焦距。特別是在類似全像圖顯微鏡等應用中，極有可能時常更換樣本，如此一來便無法手動設定焦距，導致無法重建出清晰的影像觀察樣本。因此，自動對焦功能在 3D 數位全像重建系統確實有其重要性。

本論文提出以菲涅耳轉換與細胞自動機為基礎的全像重建流程，並以重建影像的 Normalized Variance 數值評估影像清晰度作為對焦參考，以最高清晰度的焦距作為正確焦距，進行最後還原，達到自動對焦的目的。此外，本論文所提出的系統也能還原更大尺寸的全像圖，大幅擴展了系統的可應用範圍。本論文並帶入數種 I/O 加速技術，旨在減少調整焦距時影像重複使用以及傳送的時間，進而提升系統效能。

本論文最後的成果與效能評估，可證明本系統能正確還原大尺寸之數位全像圖。我們以本電路系統於 FPGA 上與 MATLAB 軟體於個人電腦上的還原程式作比較，可發現本系統的還原速度遠快於軟體運算速度，並且電路在 FPGA 上運作的功耗也遠低於軟體運作於 CPU 上的功耗。

關鍵字：FPGA、全像重建、自動對焦

目錄

表目錄

圖目錄

第一章 緒論

本章節先探討本論文的研究動機、研究目的與研究方法，並大略說明接下來各章節的概要。

1.1 研究背景

數位全像圖(Digital Holography)[1]是一種利用感光耦合元件 (Charged-Coupled Devices, CCDs)紀錄物體在光波繞射於其表面的相位與振幅資訊的技術，經過光學的繞射計算與相位還原後即可將裝置所紀錄的全像圖(Hologram)的光波資訊重建成物體原始 3D 影像。由於數位全像術具有可將影像以數位形式儲存的優點，因此這種技術目前大量投入應用於工業測量與生物醫學工程的生物細胞檢驗等領域。在這些數位全像應用中，數位全像顯微鏡(Digital Holographic Microscopy)即為一種非侵入性的測量工具，可以重建微、奈米等級物體的原始 3D 影像，在如今網路與行動裝置的發展下，全像應用也越來越多元。

為了將全像重建成原始 3D 影像，需要對全像圖進行繞射計算，再進行相位還原展開即可得到原始影像。繞射計算可使用菲涅耳轉換、摺積法與角頻譜繞射法等法則[2]，計算完成後會得到一個相位數值壓縮在 $-\pi \sim \pi$ 之間的不連續相角圖(wrapped phase)，此時再利用相位展開法則還原物體真實相位。在本論文中使用菲涅耳轉換進行繞射計算以及細胞自動機執行相位展開來重建全像

圖。菲涅耳轉換僅需一次快速傅立葉(Fast Fourier Transform)即可加速完成全像圖重建，相較於需要三次快速傅立葉的摺積法能降低更多計算複雜度。

除了降低計算複雜度與功率消耗困難之外，繞射計算過程需要全像圖與重建影像之間的焦距，否則影像將會重建失敗。傳統方式是人工變更焦距並判斷重建的影像是否在準確焦距內，但此種方式不但缺乏效率，且也無法應用於即時全像重建。因此自動對焦在即時全像重建是必要的，為了取代人工判斷影像，需要一個法則來評估影像是否成功重建。一般而言以正確焦距重建的影像比起失焦的影像有更大的高頻能量，也就是說清晰影像的物體邊界與相鄰相素灰階值的差異值或梯度值較大。而目前常用於評估影像清晰度的方法大概分為影像梯度值(gradient)計算與影像差異值(differential)計算兩種。梯度值評估如Laplacian、梯度總合計算等[3][4]；差異值法如 Tamura coefficient[5]、Normalized Variance、影像強度差異(amplitude)計算等。在此本論文選用Normalized Variance 作為評估法則，是因其在資源與誤差範圍可容許情況下，沒有使用任何的平方根運算，如此一來可減少計算複雜度以及平方根電路的硬體消耗。

1.2　動機與目的

數位全像顯微鏡為了重建微米級觀測物體的原始影像，並達到即時呈現的目的，除了上述硬體效能與功耗問題以外，全像顯微鏡所觀察的樣本極有可能

需要時常更換樣本，這種情形下未必能事先知道正確焦距，若數位全像顯微鏡不能自動對焦重建影像，則觀測者就無法順利觀察樣本。為了達成自動對焦，系統需要在一個焦距範圍內重覆進行全像重建，並評估其清晰度，再根據清晰度最高的焦距點進行重建影像。本論文採用 FPGA 作為實現平台，此架構對比軟體的優勢如表 1.1 所示。

表 1.1 FPGA 與軟體架構之特性比較

	FPGA	軟體
可攜帶性	高	低
功率消耗	低	高
運算速度	高	低

由表 1.1 可看出硬體架構比起軟體有下列優勢：高速運算、高可攜帶性、低功率消耗。在其他現存的自動對焦數位全像重建系統[10]，一樣採用 FPGA 作為實現平台，但是其架構在調整焦距時，需浪費大量時間等待全像圖重新載入全像重建系統中，且可還原尺寸較小。相位還原法則[9]僅使用在全像還原，本論文將其架構整合至自動對焦系統來協助還原影像。此外，該架構使用定點數進行運算，精確度較低。本論文架構使用浮點數運算，有效提高運算精確度。

本論文目的是設計一個減少自動對焦等待資料重新載入的時間以及可還原更大尺寸的自動對焦全像重建硬體系統。此種架構更適合用於嵌入式數位全像顯微鏡等數位全像應用，提供使用者一個即時的自動對焦全像重建硬體系統。

1.3 研究方法

本論文主要設計是以菲涅耳轉換進行繞射計算，並對計算後結果進行相位展開重建影像，最後對重建影像進行 Normalized Variance 評估分析。如此反覆數次調整焦距後，再根據評估最高值的焦距進行影像重建。

菲涅耳轉換分為三個部分，前轉換單元(Pre-transform Unit)、快速傅立葉單元(FFT Unit)、後轉換單元(Post-transform Unit)，在本論文架構中前轉換單元與後轉換單元利用查表來簡化電路運算以減少計算複雜度。而此三個單元皆以 IEEE754 單精度浮點數格式進行運算，相較於定點數(Fixed-point)來說，IEEE754 可表示範圍更大，精確度也更高，對於快速傅立葉轉換等數值變動差異大的計算是較適合的格式。

相位展開是採用細胞自動機(Cellular Automata)[6]為基礎的硬體架構，此法則可以還原出大部分的圖形，並且有低功率消耗的優點。

自動對焦的實現分為兩步驟，第一步驟先決定一個重建焦距範圍，再對每個焦距點進行全像重建。第二步驟對每個重建影像進行影像評估，最後根據最高評估值進行全像重建。在本論文中是使用 Normalized Variance 數值來進行清晰度評估。

本論文所使用的測試樣本為國立臺灣師範大學光電科技研究所資訊光電實

驗室提供，此樣本為已知曲率(curvature)與半徑(radius)的微透鏡(microlens)，由資訊光電實驗室的數位全像顯微鏡拍攝，產生的全像圖作為本論文的輸入影像。將上述電路先實作於 FPGA 平台上，我們使用可程式化系統晶片(System on Programmable Chip; SoPC)來實現，SoPC 為 Altera 公司提供的 Qsys 平台所設計，為一種 NoC(Network on Chip)架構，在本論文中此 SoPC 系統由 NIOS Processor、本論文提出的硬體架構以及其他開發版上的周邊元件組成。借由此系統架構我們可以不斷修改硬體電路，並利用 NIOS software 來進行電路驗證，使研究過程具有修改彈性以及方便驗證。

效能考量項目包涵全像重建影像清晰度、系統運算時間以及電路資源消耗(Area cost)。資源消耗的測量工具使用 Altera Quartus II，包涵上述 Qsys 系統一併測量，影像清晰度和系統運算時間則由 NIOS Processor 作為 CPU 進行測量，系統運算時間包括菲涅耳轉換、相位展開、評估運算與全像圖載入時間。軟體驗證則使用 MATLAB 來進行，將電路最終輸出結果與 MATLAB 相互驗證計算正確性，並利用 MATLAB 呈現可變更視角的 3D 影像，以便觀察還原後的 3D 影像。

1.4 全文架構

本論文共分為五個章節，以下為各章節內容概要：

第一章 緒論

說明研究背景、研究動機與目的、研究方法以及本文架構。

第二章 基礎理論及技術背景

簡介本論文使用的理論、演算法則、技術背景。

第三章 系統架構

介紹系統硬體電路架構，並說明系統運作流程。

第四章 實驗數據與效能比較

說明實驗環境、實驗數據分析以及軟硬體效能比較。

第五章 結論

對於本論文提出的硬體架構以及實驗結果進行總結

第二章 基礎理論及技術背景

本章節將介紹本論文所使用的基礎理論以及技術背景介紹。第一節討論到本架構使用的繞射計算：菲涅耳轉換，第二節討論本架構搭配使用的相位展開法則：細胞自動機(Cellular automata)。第三節討論實現自動對焦系統的基礎理論，以 Normalized variance 計算來評估影像清晰度。最後，在第四節介紹本論文實作在 FPGA 以及 SoPC 系統整合設計。提供整體系統的設定及說明。

2.1 菲涅耳轉換

菲涅耳轉換(Fresnel transform)[8]，在本論文是用於數位全像重建的繞射計算上，全像圖為數位全像顯微鏡透過 CCDs 裝置拍攝，令此全像圖資訊為 η ，焦距為 z 之重建影像以 B 表示，B 可由菲涅耳轉換計算得到，如公式(2.1.1)，

$$B(r,s)=\frac{-j}{\lambda z}e^{j\frac{2\pi}{\lambda}z}e^{j\frac{\pi}{\lambda z}\left(r^2+s^2\right)}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\eta(p,q)e^{j\frac{\pi}{\lambda z}\left(p^2+q^2\right)}\,e^{-j\frac{2\pi}{\lambda z}(pr+qs)}dpdq \quad (2.1.1)$$

其中 λ 代表其光源的波長，(p,q)及(r,s)分別為全像圖與影像平面上的座標。

由於數位全像顯微鏡拍攝的全像圖是以離散形式表示，因此為了計算將菲涅耳轉換式表達成離散形式，假設顯微鏡在x,y軸方向皆以採樣週期Δ_f，共採樣了$N\times N$個樣點，如公式(2.1.2)。

$$B_{u,v}=\frac{-j}{\lambda z}e^{j\frac{2\pi}{\lambda}z}e^{j\frac{\pi}{\lambda z}\Delta_g^2\left(u^2+v^2\right)}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left[\eta_{x,y}e^{j\frac{\pi}{\lambda z}\Delta_f^2\left(x^2+y^2\right)}\right]e^{-j2\pi\left(\frac{xu}{N}+\frac{yv}{N}\right)} \quad (2.1.2)$$

$\{B_{u,v}, 0 \leq u, v \leq N-1\}$為重建影像之數位形式，$\eta_{x,y} = \eta(x\Delta_f, y\Delta_f)$即代表全像圖$\eta$的第$(x, y)$個 sample，其中$0 \leq x, y \leq N-1$，$\Delta_g$則是以$\lambda z/N$為比例的$\Delta_f$之倒數，如公式(2.1.3)。

$$\Delta_g = \frac{\lambda z}{N\Delta_f} \tag{2.1.3}$$

在公式(2.1.2)中，與$\eta_{x,y}$相乘之指數函數可以依指數加法律整理得公式(2.1.4)，並將其重新定義為μ，如公式(2.1.5)，

$$e^{j\frac{\pi}{\lambda z}\Delta_f^2(x^2+y^2)} = e^{j\frac{\pi}{\lambda z}\Delta_f^2(x^2)} \times e^{j\frac{\pi}{\lambda z}\Delta_f^2(y^2)} \tag{2.1.4}$$

$$\mu_x = e^{j\frac{\pi}{\lambda z}\Delta_f^2(x^2)},\ \mu_y = e^{j\frac{\pi}{\lambda z}\Delta_f^2(y^2)} \tag{2.1.5}$$

公式(2.1.5)與全像圖資訊$\eta_{x,y}$相乘後得到函數$\rho_{x,y}$，化簡如公式(2.1.6)。$\rho_{x,y}$為一週期函數，因此可以將其做傅立葉轉換後得到函數$\tau_{u,v}$，如公式(2.1.7)。本論文中以快速傅立葉轉換的架構來計算$\tau_{u,v}$，相關的計算電路將於第三章系統架構中詳細討論。

$$\rho_{x,y} = \eta_{x,y} \times \mu_x \times \mu_y \tag{2.1.6}$$

$$\tau_{u,v} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\rho_{x,y}e^{-j2\pi\left(\frac{xu}{N}+\frac{yn}{N}\right)} \tag{2.1.7}$$

由於菲涅耳轉換式中λ為固定數值、z也假設為已知焦距，因此可定義一定值α，將其化簡為公式(2.1.8)，

$$\alpha = \frac{-j}{\lambda z} e^{j\frac{2\pi}{\lambda}z} \tag{2.1.8}$$

另外，與$\tau_{u,v}$相乘之指數函數也可仿公式(2.1.5)定義為ω_u及ω_v，如公式(2.1.9)所示，

$$\omega_u = e^{j\frac{\pi}{\lambda z}\Delta_g^2(u^2)},\ \omega_v = e^{j\frac{\pi}{\lambda z}\Delta_g^2(v^2)} \tag{2.1.9}$$

接著利用公式(2.1.9)將菲涅耳公式化簡，可得到函數解$B_{u,v}$，如公式(2.1.10)。

$$B_{u,v} = \alpha \times \omega_u \times \omega_v \times \tau_{u,v} \tag{2.1.10}$$

到此部分為止，菲涅耳公式計算完成，為了進行後續相位展開法則運算，必須將複數$B_{u,v}$的虛部$Im(B_{u,v})$除以實部$Re(B_{u,v})$後進行反正切函數(arctangent)運算來取得相位$\phi_{u,v}$，如公式(2.1.11)所示，

$$\phi_{u,v} = tan^{-1}\frac{Im(B_{u,v})}{Re(B_{u,v})} \tag{2.1.11}$$

令$Im(B_{u,v}) = Q$，$Re(B_{u,v}) = I$，也就是將$B_{u,v}$表示成$B_{u,v} = I + Qi$，本論文中的反正切函數運算使用 arctangent approximation 公式[7]來求得近似解，如公式(2.1.12)，

$$tan^{-1}\left(\frac{Q}{I}\right) = \frac{\frac{Q}{I}}{1+0.28086\left(\frac{Q}{I}\right)^2}, -1 \le \frac{Q}{I} \le 1 \tag{2.1.12}$$

另外，由於計算出的$(\frac{Q}{I})$不一定符合公式(2.1.12)的條件$-1 \le \frac{Q}{I} \le 1$，為了解決這個問題，本論文的反正切運算將會以公式(2.1.12)搭配公式(2.1.13)及公式(2.1.14)使用。

$$tan^{-1}\left(\frac{Q}{I}\right)=\frac{\pi}{2}-tan^{-1}\left(\frac{I}{Q}\right),\frac{Q}{I}>0 \tag{2.1.13}$$

$$tan^{-1}\left(\frac{Q}{I}\right)=-\frac{\pi}{2}-tan^{-1}\left(\frac{I}{Q}\right),\frac{Q}{I}>0 \tag{2.1.14}$$

整理以上的公式，此章節的菲涅耳轉換公式以及 arctangent 公式流程如以下步驟：

步驟一：給予輸入數位全像圖函數$\eta_{x,y}$，$0\leq x,y<N$。

步驟二：利用公式(2.1.6)，計算函數$\rho_{x,y}$。

步驟三：利用公式(2.1.7)對$\rho_{x,y}$進行 2D 傅立葉轉換得到$\tau_{u,v}$，在本論文架構中是以兩次 1D 傅立葉轉換來實現，首先對$\rho_{x,y}$中每一列x執行快速傅立葉轉換或離散傅立葉轉換，可得到一運算結果陣列$\beta_{x,v}$，如公式(2.1.15)所示，

$$\beta_{x,v}=\sum_{y=0}^{N-1}\left(\rho_{x,y}cos\frac{2\pi}{N}yv+j\rho_{x,y}sin\frac{2\pi}{N}yv\right),0\leq x\leq N-1 \tag{2.1.15}$$

接著再對$\beta_{x,v}$每一行v進行一次傅立葉轉換，最後得到 2D 傅立葉轉換結果$\tau_{u,v}$如公式(2.1.16)。

$$\tau_{u,v}=\sum_{x=0}^{N-1}\left(\beta_{x,v}cos\frac{2\pi}{N}xu+j\beta_{x,v}sin\frac{2\pi}{N}xu\right),0\leq y\leq N-1 \tag{2.1.16}$$

步驟四：使用公式(2.1.10)對$\tau_{u,v}$做菲涅耳轉換最後的計算，求出函數$B_{u,v}$。

步驟五：使用公式(2.1.12)、(2.1.13)、(2.1.14)對函數$B_{u,v}$進行反正切運算求得相位值$\phi_{u,v}$。

2.2 細胞自動機

3D 數位全像的重建需要執行相位展開才能完整還原觀測物體，相位展開法則在本論文中使用細胞自動機(Cellular Automata)，細胞自動機能還原不連續相位的全像圖，但是無法保證運算能達成收斂，因此在本論文中我們使用的法則為廣域型相位展開法則(Regional (7 × 7))[9]。

細胞自動機原理為將目前處理的 Current pixel 進行運算，運算需參考相鄰區域。令$B_i[m,n]$為$x_i[m,n]$的相鄰區域，i為迭代的次數，如公式(2.2.1)

$$B_i[m,n] = \{x_i[p,q], m-k \le p \le m+k, n-k \le q \le n+k\} \qquad (2.2.1)$$

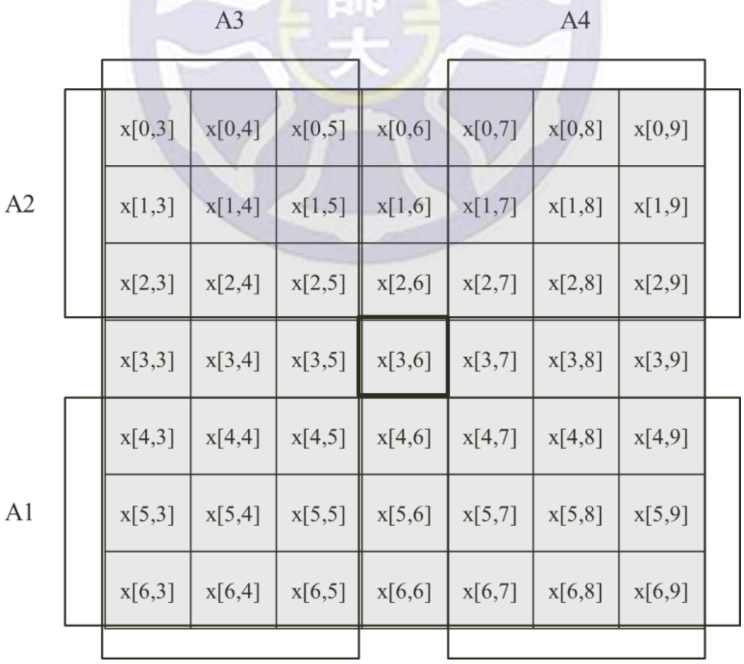


圖 2.1 細胞自動機$B_i[m,n]$示意圖與其相鄰區域

$A_j, j = 1,2,3,4 \text{ for } \mathrm{i} = 0, \mathrm{m} = 3, \mathrm{n} = 6 \text{ and } \mathrm{K} = 7 (\mathrm{i.e.}, \mathrm{k} = 3)$

由於$x_i[m,n]$會出現在影像邊緣，$B_i[m,n]$會包含到影像外的 Pixel，將這些 Pixel 設為 0。接下來把$B_i[m,n]$分割成四塊區域$A_1 \sim A_4$，定義如下：

$$A_1 = \{x_i[p,q], m+1 \le p \le m+k, n-k \le q \le n+k\},$$

$$A_2 = \{x_i[p,q], m-k \le p \le m-1, n-k \le q \le n+k\},$$

$$A_3 = \{x_i[p,q], m-k \le p \le m+k, n+1 \le q \le n+k\},$$

$$A_4 = \{x_i[p,q], m-k \le p \le m+k, n-k \le q \le n-1\} \tag{2.2.2}$$

分出四塊區域後，我們即可定義出P_j，P_j為一區域包含所有A_j區域中的 Neighbors 與 Current Pixel 相減後大於π的 Pixel，如下所示：

$$P_j = \left\{x_i[p,q]: x_i[p,q] \in A_j, x_i[p,q] - x_i[m,n] > \pi\right\} \tag{2.2.3}$$

如果P_j的數量大於$\frac{kK}{2}$，$h_i[m,n]$就設為 1。

$$h_i[m,n] = \begin{cases} 1, \left|P_j\right| > \frac{kK}{2}\ for\ any\ j = 1,2,3,4 \\ 0,\ otherwise \end{cases} \tag{2.2.4}$$

則下一輪的$x_{i+1}[m,n]$值即為

$$x_{i+1}[m,n] = x_i[m,n] + h_i[m,n]2\pi \tag{2.2.5}$$

逐步掃描並完成計算直到沒有任何 Pixel 需要更新，即完成相位展開。

2.3 自動對焦之基礎理論

本論文使用自動對焦理論是針對不同焦距下重建的影像進行清晰度評估，而評估值最高的影像代表其擁有最佳焦距。由上一節公式(2.1.2)中可知繞射計算與焦距 z 有很大的關係，在μ_x, μ_y中要根據焦距重新計算，所以本論文的自動對焦過程即為不斷變換焦距，並且重建影像、進行評估。在本論文第一章節研究背景中提到成功對焦的影像有較大的高頻能量，在這個部份，我們使用 Normalized Variance[10]量測相位展開之重建影像P的差異值，Normalized Variance 可視為 Tamura measure 的衍伸型，令 Tamura measure 對P的計算結果為$T(P)$，可定義為：

$$T(P) = \sqrt{\frac{\sigma(P)}{\gamma(P)}} \tag{2.3.1}$$

其中$\sigma(P)$代表重建影像P的標準差，而$\gamma(P)$則為影像P的平均值。

$\gamma(P)$與$\sigma(P)$的計算式為

$$\gamma(P) = \frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} P_{u,v} \tag{2.3.2}$$

$$\sigma(P) = \sqrt{\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[P_{u,v} - \gamma(P)\right]^2} \tag{2.3.3}$$

我們將本論文所用的 Normalized Variance 定義為$V(P)$，其計算內容如公式(2.3.4)，

$$V(P) = \frac{\sigma(P)^2}{\gamma(P)} \tag{2.3.4}$$

經過上述公式化簡[10]後我們可以得到公式(2.3.5)

$$V(P)=\frac{1}{\gamma(P)}\left[\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}{P_{u,v}}^2-\gamma(P)^2\right]$$

$$=\frac{1}{\gamma(P)}\left[\frac{1}{N^2}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}{P_{u,v}}^2\right]-\gamma(P) \tag{2.3.5}$$

由上述公式便可以計算出影像的清晰度，不同焦距下對影像進行繞射計算後還原的清晰度也不相同，將這些數值$V(P)$相互比較後，找到具有最高$V(P)$的重建影像就代表其為重建最完整的結果。

對焦範圍本論文使用 depth-of-focus[11]法則，此法則是針對全像顯微鏡的相關係數進行運算得到DOF，而成像會落在 M^2DOF 的數倍上。計算公式如下：

$$\mathrm{DOF}=\lambda n_m/NA^2 \tag{2.3.6}$$

其中 λ 為紀錄全像使用的波長，n_m為介質傳送的速率，NA 是透鏡的 Numerical Aperture(數值孔徑)。M 則是全像顯微鏡的放大倍率。經過運算得到M^2DOF，將其設為自動對焦的 Search Step，在數倍的 Search Step 範圍內進行全像重建，便可以得到正確成像焦距。

有了前幾節的公式我們可以整理出自動對焦流程，如圖 2.2 所示。先將 Hologram $\eta_{x,y}$經過繞射計算得到$B_{u,v}$，之後再將其結果進行 Arctangent 運算得到相位圖$\phi_{u,v}$，最後經過相位展開取得全像重建圖$P_{u,v}$，透過評估公式計算得到影像清晰度。這樣便完成一次的重建運算，改變焦距再次進行上述流程，反覆數次即可得知最佳還原焦距。

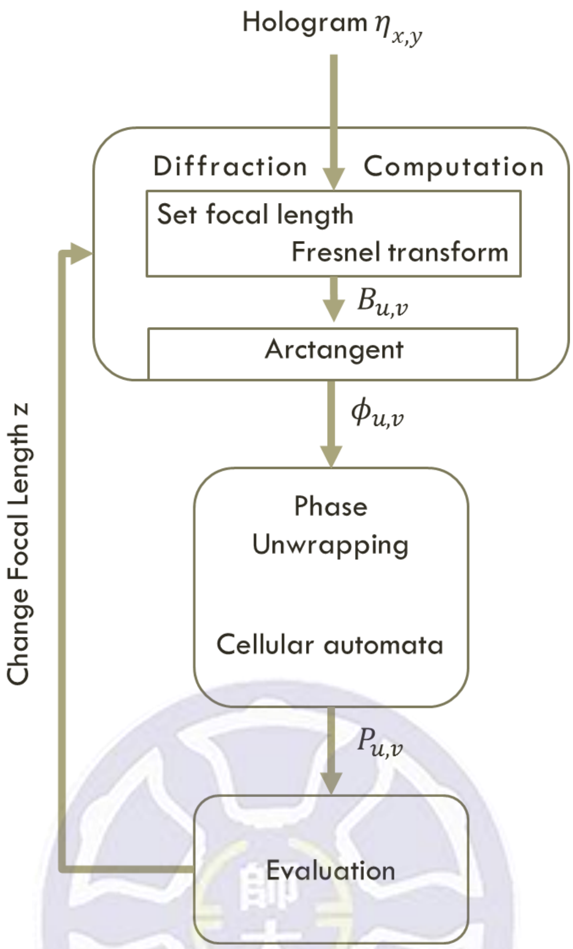

Hologram $\eta_{x,y}$
Diffraction Computation
Set focal length
Fresnel transform
$B_{u,v}$
Arctangent
$\phi_{u,v}$
Phase
Unwrapping
Cellular automata
$P_{u,v}$
Evaluation
Change Focal Length z

圖 2.2 自動對焦流程

2.4 實作於 FPGA 與 SoPC 系統整合設計

本論文使用硬體描述語言(Hardware Description Language, HDL)搭配可重覆程式設計的 FPGA 晶片設計電路，縮短研發時間而且更有彈性，經過系統提供的合成與繞線佈局，可快速燒錄到 FPGA 板上進行測試，且不需要負擔實際合成電路失敗的高成本風險，是目前研究 IC 設計的技術主流。

Altera 在 NIOS 系統中提供了一套專門給 NIOS II Processor 使用的匯流排(Avalon bus)，由 Qsys 平台自動產生，透過硬體描述語言設計硬體電路，經由 Qsys 所建立的 SoPC 系統在 FPGA 上程式化後，即可使用 NIOS 開發工具以 C 語言撰寫的程式執行電路，並觀察電路結果。

為了減少重複使用全像圖所消耗的時間，除了將全像圖載入 SDRAM 以外，本論文也使用 Altera 的 SGDMA Controller 來更進一步加速傳輸。

第三章 系統架構

本章節將介紹本論文自動對焦系統的硬體架構，包含菲涅耳轉換、相位展開以及評估單元。第一節先討論整個系統的硬體架構，接著在第二節討論菲涅耳轉換與表格建置模組。第三節討論整合後的電路架構是如何進行自動對焦，影像資料在各單元之間運算以及作業流程。

3.1 整體系統硬體架構

本節討論自動對焦系統的各硬體單元，在 2.3 節討論了整個系統架構需要的計算公式以及流程。根據其流程，本論文系統架構如圖 3.1 所示，全像重建系統含有菲涅耳轉換單元、相位展開單元、評估單元以及表格建置單元。SDRAM 用來存放原始全像圖以及使資料能更快載入 On-Chip RAM，本論文使用 SGDMA Controller 來傳輸資料。

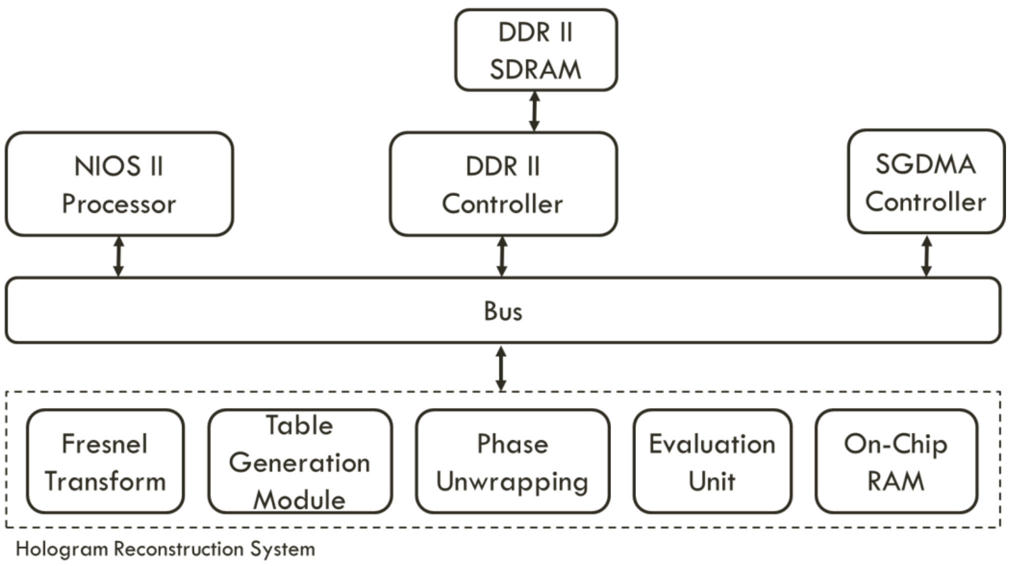


圖 3.1 自動對焦 3D 數位全像重建系統架構圖

為了獲得在不同焦距下的不連續相角圖$\phi_{u,v}$，菲涅耳運算中的焦距z，成為自動對焦過程重要部分，在菲涅耳轉換中部分使用焦距的運算，事先將其計算完成，並儲存在表格提供給前轉換運算以及後轉換運算使用。相位展開單元使用的架構引用自 2.2 節提到的相位展開法則Regional (7×7)，其架構可參考文獻[9]，評估單元依照公式(2.3.5)設計，電路中包含一組控制器、兩組累加器、三個除法器以及乘法器、減法器各一組。其電路架構可參考文獻[10]。

3.2 菲涅耳轉換以及表格建置單元

菲涅耳轉換主要實現公式(2.1.2)計算，分為三個部分，分別為前轉換單元、快速傅立葉轉換單元、後轉換單元與儲存參數的表格。如圖 3.2 所示。

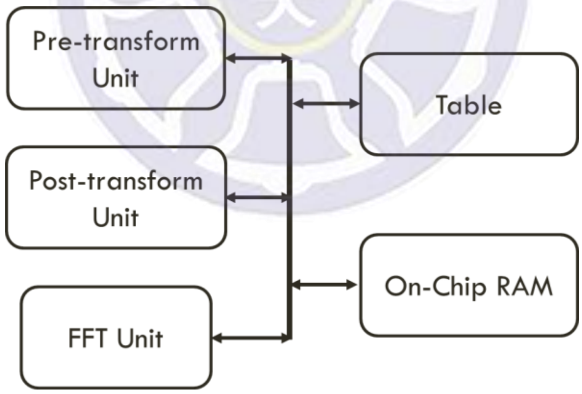


圖 3.2 菲涅耳轉換單元

3.2.a 前轉換單元

前轉換單元是根據公式(2.1.6)設計，根據x, y從表格讀取參數μ_x, μ_y後，與全像資訊$\eta_{x,y}$進行複數計算。架構如圖 3.3 所示。包含一個地址產生器(Address Generation Unit,AGU)、存放參數的表格、兩個複數乘法器。

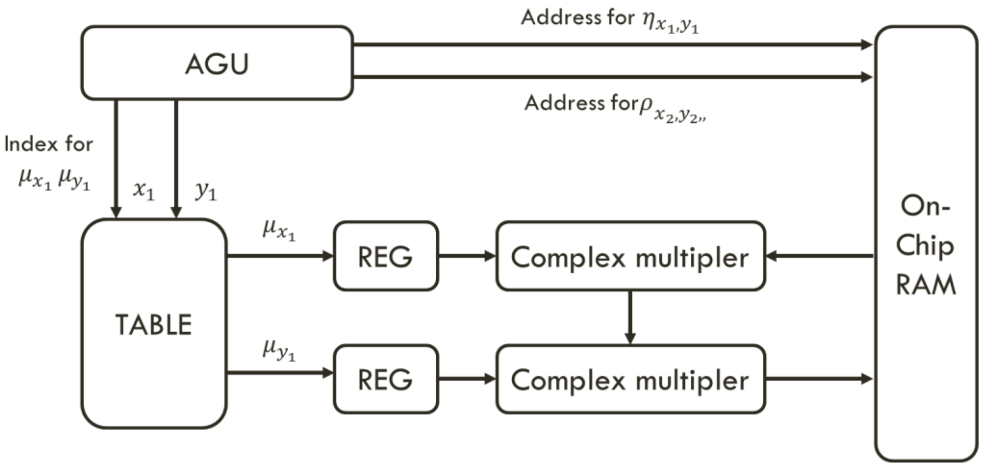
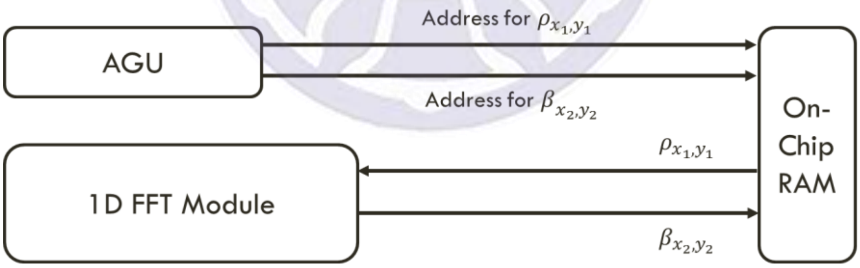


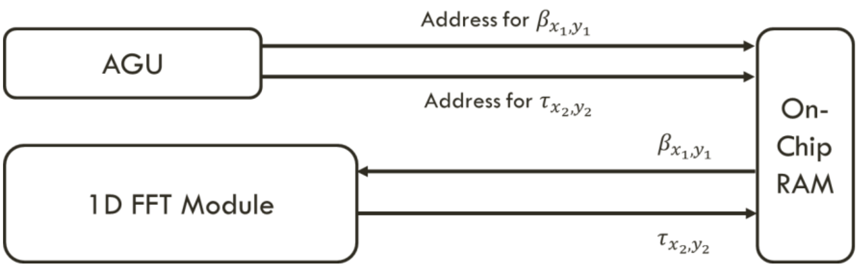
圖 3.3 前轉換單元

3.2.b 快速傅立葉轉換單元

快速傅立葉轉換單元是計算公式(2.1.7)的$\tau_{u,v}$，此單元是由一個地址產生器(AGU)以及 FFT 模組組成。如公式(2.1.15)以及公式(2.1.16)，本論文是使用 1D FFT[12]來實現 2D FFT 計算。



(a)



(b)

圖 3.4 快速傅立葉轉換單元架構: (a) FFT 列運算, (b) FFT 行運算

3.2.c 後轉換單元

後轉換單元實現公式(2.1.10)以及公式(2.1.11)，如圖 3.7 所示。與前轉換單元類似，同樣含有計算參數表格以及複數乘法器，最後在執行 Arctangent 即完成後轉換運算。

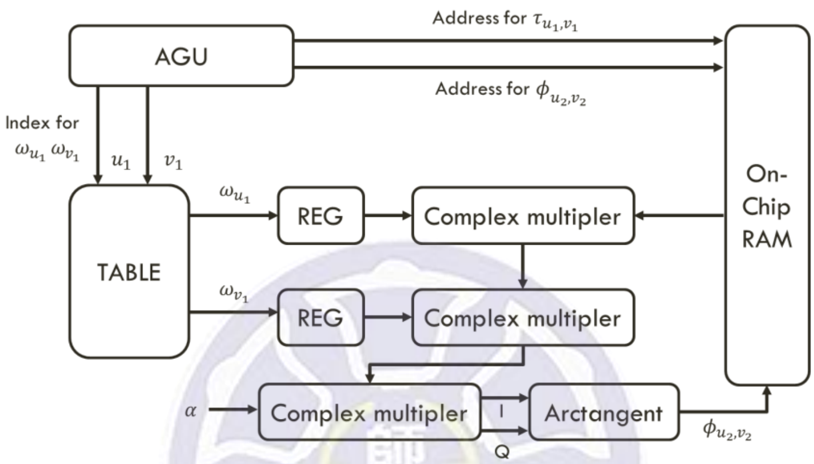


圖 3.5 後轉換單元

3.2.d 表格建置單元

在菲涅耳轉換中[8]，為了更快速的計算，先將部分參數計算後存放於表格中，運算的時候提供使用。這些參數運算也使用到焦距。而上述架構是針對固定焦距進行運算，表格內容並不會更動，但在自動對焦過程中，必須讓電路可以隨著焦距動態更新表格。

表格建置模組在前轉換與後轉換運算的同時產生新的表格，從公式(2.1.5)得知μ_x和μ_y可以共用一張表格，同樣的根據公式(2.1.9)可以得知ω_u和ω_v也共用一張表格，因此我們只需要兩張 N 個 entries 的表格即可。為了進行運算我們

將公式(2.1.5)以及公式(2.1.9)展開成sin與cos的函數，如公式(3.1)與公式(3.2)。

$$\begin{cases} \mu_x = cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_f^2 x^2\right] + jsin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_f^2 x^2\right] \\ \mu_y = cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_f^2 y^2\right] + jsin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_f^2 y^2\right] \end{cases} \tag{3.1}$$

$$\begin{cases} \omega_u = cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_g^2 u^2\right] + jsin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_g^2 u^2\right] \\ \omega_v = cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_g^2 v^2\right] + jsin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_g^2 v^2\right] \end{cases} \tag{3.2}$$

其中因為$\Delta_g = \frac{\lambda z}{N\Delta_f}$，所以公式(3.2)可以推導出公式(3.3)

$$\begin{cases} \omega_u = cos\left[\left(\frac{\pi\lambda z}{N^2\Delta_f^2}\right)u^2\right] + jsin\left[\left(\frac{\pi\lambda z}{N^2\Delta_f^2}\right)u^2\right] \\ \omega_v = cos\left[\left(\frac{\pi\lambda z}{N^2\Delta_f^2}\right)v^2\right] + jsin\left[\left(\frac{\pi\lambda z}{N^2\Delta_f^2}\right)v^2\right] \end{cases} \tag{3.3}$$

有了公式(3.1)以及公式(3.3)我們即可根據焦距z與索引值x, y, u, v進行表格建置，公式中$\Delta_f^2 x^2$與u^2/Δ_f^2是常數與索引值，圖 3.6 為表格建置模組的架構圖。

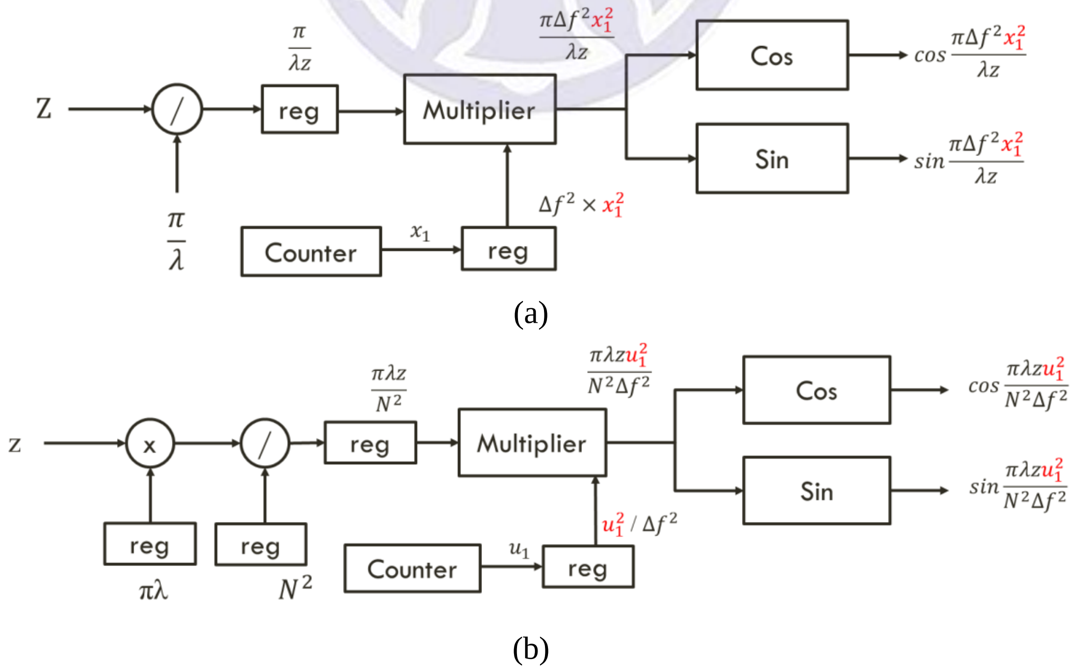


圖 3.6 表格建置模組架構

3.3 自動對焦資料流程

本節討論在此系統架構下完整執行自動對焦的流程。如圖 3.7 所示，SGDMA Controller 會控制 DDR II Controller 將全像圖從 SDRAM 傳輸至全像重建系統的 On-Chip RAM，之後如圖 3.8，將目前的重建焦距以及下一輪的重建焦距寫入全像重建系統，並啟動菲涅耳轉換單元，同時菲涅耳轉換單元內的控制器也會啟動表格建置單元，進行表格建置運算。如此一來在菲涅耳轉換結束後下一輪的表格也已經產生完成。

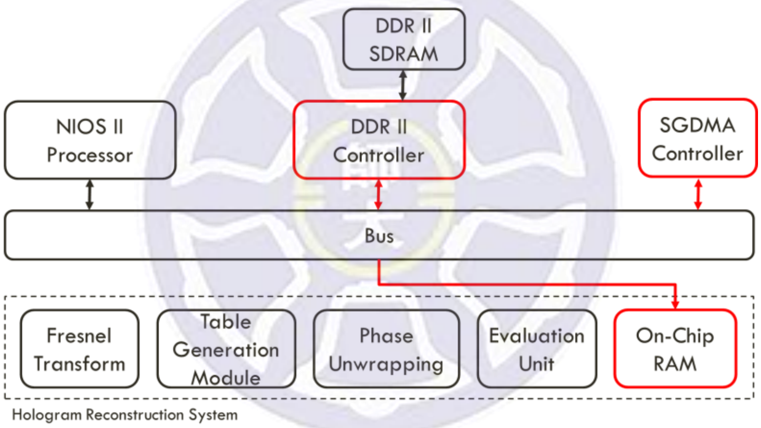


圖 3.7 由 SGDMA 傳輸資料至 On-Chip RAM

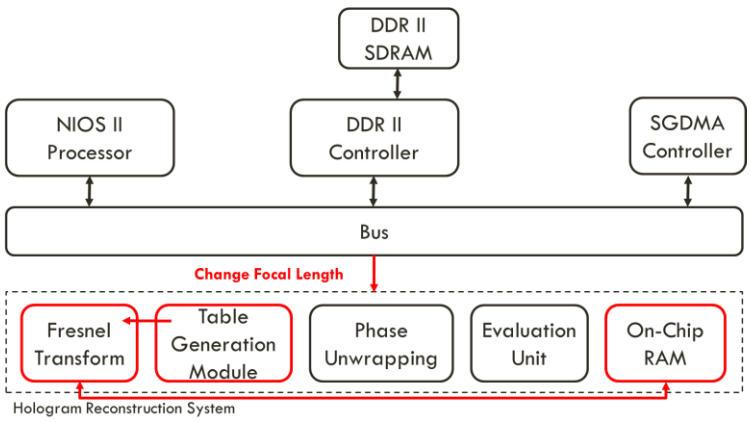


圖 3.8 設定焦距並執行菲涅耳轉換與表格建置

菲涅耳轉換結束後，如圖 3.9 執行相位展開取得重建影像，接下來如圖 3.10 執行評估單元運算取得評估值。如此便完成一次全像還原。反覆執行圖 3.7~圖 3.10 的流程，等到所有焦距點都還原以及評估完成。便會得到最高清晰度的焦距點，再以此焦距進行最後一次還原，即完成自動對焦。

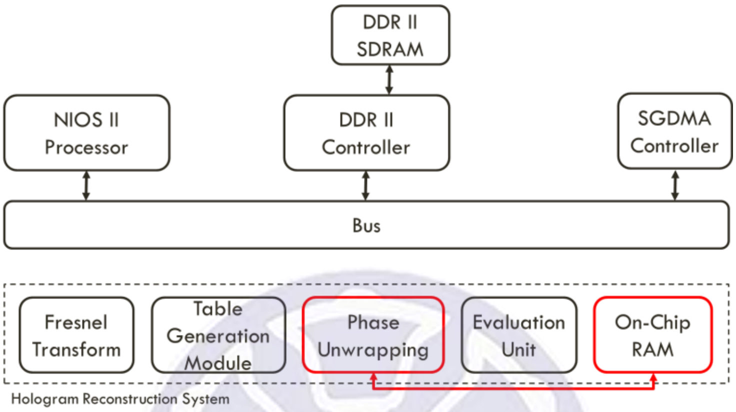


圖 3.9 執行相位展開運算

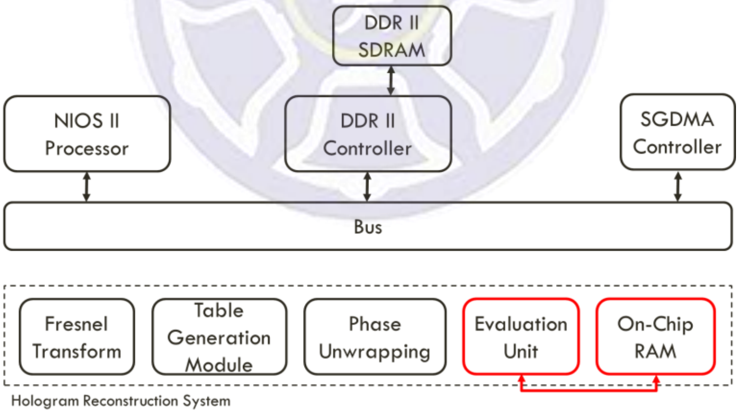


圖 3.10 執行評估單元運算

第四章 實驗數據與效能比較

本章節將整理論文提出架構的整合效果、實際效能測量與比較，以及實驗環境介紹。

4.1 開發平台與實驗環境設定

本論文提出的硬體架構是以 Altera Stratix IV GX EP4SGX530 FPGA 開發板作為主要的實現平台，外觀如圖 4.1 所示。表 4.1 則整理出開發版的詳細規格。FPGA 開發版可用來實現與驗證本論文電路架構，其擁有可程式化系統晶片(SoPC)，易於重複修改電路架構的特性，在硬體設計的驗證上相當適合，因此選擇此開發版作為本論文實現硬體設計的平台。

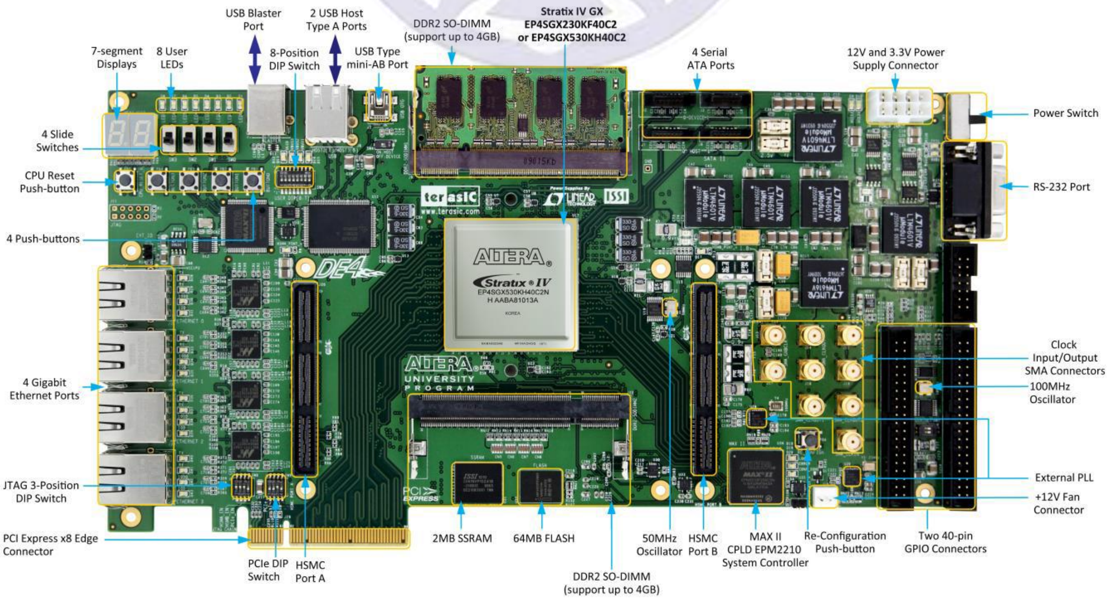


圖 4.1 Altera Stratix IV GX EP4SGX530 FPGA (Top view)

表 4.1 Altera Stratix IV GX EP4SGX530 詳細規格

Feature	Stratix® IV GX FPGA
Device	**EP4SGX530**
Adaptive Logic Modules (ALMs)	**212,480**
Logic Elements (LEs)	**531,200**
M9K Memory Blocks	**1,280**
M144K Memory Blocks	**64**
Total memory Kbits	**27,376**
DSP block 18-bit elements	**1,024**
Total PLLs	**8**
User I/Os	**744**

本論文使用 Altera Quartus II 14.0 作為 Verilog 硬體描述語言的開發平台，Quartus 提供給 FPGA 開發版使用者一個基礎的開發環境，並且可以藉由 RTL-level 與 Gate-level 模擬訊號波型圖驗證電路正確性。當系統架構設計完成後，便可將系統掛載在 SoPC 系統上，在 Quartus II 14.0 中使用的是 Qsys 平台，包含本論文使用到的 SDRAM、SGDMA 等元件。將系統燒錄於開發板後即可利用 NIOS II 觀察電路輸出。

硬體實驗環境
Device : Altera Stratix IV GX EP4SGX530 FPGA
CPU : NIOS II Processor 150 MHz

軟體實驗環境
CPU : Intel(R) Core(TM) i7-3770 CPU @ 1.60GHz
RAM: 8.0GB
Matlab R2014a

4.2 實驗數據呈現與討論

本節展示自動對焦 3D 數位全像重建硬體系統的實驗結果與效能分析，首先觀察本論文架構中的菲涅耳轉換之資源消耗，表 4.2 是針對前轉換單元、快速傅立葉轉換單元、後轉換單元與嵌入式記憶體之資源消耗複雜度分析。假設菲涅耳轉換輸入影像邊長為N，輸入的影像大小為$N \times N$，加法器、乘法器、除法器在不同運算單元中分析結果類似，不會隨著影像大小而改變，而暫存器數量則會隨著影像大小變化。

在前轉換單元(Pre-transform)的部分，使用了兩個複數乘法器，複數乘法器使用到 4 個乘法器與 2 個加法器，影像大小並不會改變此兩者數量，所以複雜度為$O(1)$。前轉換運算沒有使用到除法器，因此複雜度為 0。在暫存器的部分，前轉換單元包含了記載計算參數的 Table，Table 為大小N的 On-Chip RAM，因此複雜度為$O(N)$。在快速傅立葉單元中同樣擁有固定數量的加法器與

表 4.2 菲涅耳轉換與嵌入式記憶體資源複雜度分析

	Pre-transform Unit	FFT Unit	Post-transform Unit	On-Chip RAM
Adders	$\boldsymbol{O}(\mathbf{1})$	$\boldsymbol{O}(\mathbf{1})$	$\boldsymbol{O}(\mathbf{1})$	**0**
Multipliers	$\boldsymbol{O}(\mathbf{1})$	$\boldsymbol{O}(\mathbf{1})$	$\boldsymbol{O}(\mathbf{1})$	**0**
Dividers	**0**	**0**	$\boldsymbol{O}(\mathbf{1})$	**0**
Registers	$\boldsymbol{O}(\boldsymbol{N})$	$\boldsymbol{O}(\boldsymbol{N})$	$\boldsymbol{O}(\boldsymbol{N})$	$\boldsymbol{O}(\boldsymbol{N}^{\mathbf{2}})$

乘法器，並且沒有使用到除法器，因此加法器與乘法器複雜度皆為$O(1)$而除法器則為 0。後轉換單元的架構與前轉換單元十分相似，因此加法器、乘法器皆為$O(1)$，而後轉換單元同樣包含大小為N的 Table，暫存器複雜度為$O(N)$，不同的地方是，後轉換單元需要反正切函數電路計算，此電路包含不隨影像大小改變的除法器數量，因此複雜度為$O(1)$。而嵌入式記憶體作為各單元輸入與輸出影像資料儲存器，其必須能存放$N \times N$的影像資料，所以暫存器複雜度為$O(N^2)$，其餘皆為 0。

接下來以相同方法分析討論相位展開電路、搜尋單元與評估單元的資源複雜度。由於 3D 數位全像重建與搜尋單元中的焦距產生是借由 NIOS 處理器產生，因此另外分析 NIOS 處理器資源複雜度。如表 4.3，相位展開電路的計算為自身$+2\pi$後與鄰近區域進行比較，需要數量$O(1)$的加法器與乘法器，且為了進行比較，我們需要長度為$N \times K$的蛇狀暫存器來存放影像資料，K為相位展開電路中的還原法則中，代表進行比較的鄰近區域邊長，因此暫存器複雜度為$O(N \times K)$。

表 4.3 搜尋單元與評估單元資源複雜度分析

	Phase Unwrapping	Table Generation Module	Evaluation Unit	NIOS II Processor
Adders	$\boldsymbol{O(1)}$	**0**	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$
Multipliers	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$
Dividers	**0**	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$
Registers	$\boldsymbol{O(N \times K)}$	$\boldsymbol{O(N)}$	$\boldsymbol{O(1)}$	$\boldsymbol{O(1)}$

在搜尋單元中，表格建置模組中使用了固定數量的乘法器與除法器，而Δ_f與Δ_g參數和索引值計算也是先大小為N的 Table 儲存，因此暫存器的資源複雜度為$O(N)$。在評估單元中計算$N \times N$大小的影像資料，但是由於資料是 Serial 交給單元計算，因此資源消耗不會隨著影像大小改變，所以複雜度皆為$O(1)$，而 NIOS Processor 資源消耗並不會隨著影像大小改變，只會根據設定改變。因此資源消耗複雜度皆為$O(1)$。將菲涅耳轉換、相位展開單元、搜尋單元以及評估單元的電路資源複雜度整理後，可得到整體系統的資源複雜度在加法器、乘法器以及除法器皆為$O(1)$，而暫存器的資源複雜度則是$O(N^2 + N)$。

資源消耗分析完成後，接下來討論實作此硬體架構所需資源。本論文以影像尺寸 256×256 ($N = 256$) 以及 512×512 ($N = 512$) 的全像圖作為實作樣本。在開始討論 Altera FPGA 開發版資源消耗量測時，是根據 Altera 官方的計算方式，使用三種單位作為評估資源消耗的依據，分別是 Adaptive Logic Modules(ALMs)、Embedded memory bits 和 DSP block，其中 ALMs 被用來實現運算單元及暫存器；Embedded memory bits 主要用於暫存器及記憶體元件；而 DSP block 則被用來實現數值運算的元件如快速傅立葉模組中的乘法或後轉換單元中的除法器等。

表 4.4 表示了菲涅耳轉換各單元以及嵌入式記憶體單元的 ALMs 消耗，大部分的資源消耗集中在菲涅耳轉換的三個單元，原因是 FFT 的複雜計算消耗了大量 ALMs，且隨著N的增加，FFT 也會消耗更多的 ALMs，而前轉換以及後轉

換單元中的計算需求並不會隨著影像大小增加而改變，所以 ALMs 並沒有顯著的增加。

表 4.4 菲涅耳轉換各單元與嵌入式記憶體之 ALMs 資源消耗表

Size	Pre-transform Unit	FFT Unit	Post-transform Unit	On-Chip RAM
N = 256	**3,255**	**9,809**	**7,153**	**149**
N = 512	**3,631**	**14,483**	**7,650**	**614**

表 4.5 為相位展開單元、搜尋單元與評估單元在不同法則下的 ALMs 資源消耗，在本實驗中相位還原法則為 Regional (7 × 7) 的相位還原法則，因此消耗的暫存器數量相較於另外其他單元多，但 ALMs 並不會隨著影像大小增加。表格建置單元、評估單元在計算的部分不會隨著影像大小改變而增加，因此 ALMs 也沒有顯著增加。

表 4.5 相位展開單元、搜尋單元與評估單元之 ALMs 資源消耗表

Size	Phase Unwrapping	Table Generation Module	Evaluation Unit	NIOS II Processor
N = 256	**16,174**	**6,549**	**2,019**	**1,174**
N = 512	**17,057**	**7,900**	**2,244**	**1,414**

表 4.6 以及表 4.7 分別為菲涅耳轉換與嵌入式記憶體以及相位展開單元、搜尋單元與評估單元的 Memory bits 消耗，從表中可觀察到前轉換單元以及後轉換單元因包含兩張記載參數的 Table，所以資源消耗隨著影像大小提升，與表 4.2 分

析相同，而嵌入式記憶體因其暫存器複雜度為 $O(N^2)$，所以 Memory bits 消耗是最多的。

相位展開單元則因為蛇狀暫存器需要複雜度為 $O(N \times K)$ 的暫存器，所以會隨著影像大小以及法則比較鄰近區域範圍提升而增加其對 Memory bits 的消耗。而搜尋單元中的建置表格模組則需要與影像大小相同的表格，因此也會隨著影像大小而增加。Memory bits 資源消耗與表 4.3 的分析符合。評估單元與 NIOS Processor 在 Memory bits 的消耗上，並不會隨著影像大小增加。

表 4.6 菲涅耳轉換各單元之 Memory bits 資源消耗表

Size	Pre-transform Unit	FFT Unit	Post-transform Unit	On-Chip RAM
N = 256	**16,600**	**51,064**	**25,600**	**4,194,304**
N = 512	**32,768**	**168,188**	**41,984**	**16,777,216**

表 4.7 相位展開單元、搜尋單元與評估單元之 Memory bits 資源消耗表

Size	Phase Unwrapping	Table Generation Module	Evaluation Unit	NIOS II Processor
N = 256	**56,851**	**23,994**	**18,627**	**63,680**
N = 512	**114,867**	**37,376**	**18,432**	**65,024**

表 4.8 與 4.9 是各單元的 DSP Blocks 消耗，菲涅耳轉換各單元、相位展開還原與表格建置單元使用的複數計算是 DSP Blocks 的主要消耗來源，表格建置單元、後轉換單元與評估單元因含有除法計算，DSP Blocks 消耗相較於其他單

元更高。而這些運算單元數量並不會因為影像大小 N 而改變。

表 4.10 是整體電路在 DE4 開發版的資源消耗，從表中觀察可以得知，在影像大小 N 增加的情況，ALMs 增加幅度不大，而 Memory bits 則是以 $O(N^2)$ 的複雜度增加。DSP Block 主要用於計算，所以整體也沒有太大變化。整體系統在影像大小 N 增加的情況，以 Memory bits 的資源消耗為主要。

表 4.8 菲涅耳轉換各單元之 DSP blocks 資源消耗表

Size	Pre-transform Unit	FFT Unit	Post-transform Unit	On-Chip RAM
N = 256	**32**	**48**	**88**	**0**
N = 512	**32**	**64**	**88**	**0**

表 4.9 相位展開單元、搜尋單元與評估單元之 DSP blocks 資源消耗表

Size	Phase Unwrapping	Table Generation Module	Evaluation Unit	NIOS II Processor
N = 256	**0**	**108**	**68**	**4**
N = 512	**0**	**108**	**68**	**4**

表 4.10 整體電路在 DE4 開發版之資源消耗表

Size	ALMs	Memory bits	DSP Block
N = 256	**47,935 (22 %)**	**6,499,954 (30 %)**	**348 (33 %)**
N = 512	**65,418 (30 %)**	**17,409,807 (82 %)**	**364 (36 %)**

表 4.11 為菲涅耳轉換、相位展開電路、搜尋單元與評估單元的執行時間分析，在本次實驗中以 150MHz 執行自動對焦 3D 數位全像重建硬體系統，由於表格建置單元與菲涅耳轉換電路可以平行運算，且表格建置單元的執行時間為 $O(N)$ 遠小於菲涅耳轉換中前轉換與後轉換執行時間，因此在這邊忽略討論表格建置單元的時間，首先觀察菲涅耳轉換中前轉換單元、快速傅立葉單元與後轉換單元的執行時間。可觀察出三個單元的執行時間都相當接近，因為其執行時間取決於影像大小，從讀取第一筆資料後便開始運算，運算後即輸出結果，執行時間相當於影像大小 N^2 執行時間結果符合期待。而評估單元也同樣需要讀取 N^2 的影像資料，因為浮點數的累加性質，評估單元的執行時間相對菲涅耳轉換來的高。相位展開還原則是花費相對大量的時間，是因展開過程中需要不斷重複與鄰近區域比較，重覆迭代後才能完成還原。

獲得各單元執行速度後，可算出整個 3D 數位全像重建硬體系統的執行時間為 235(ms)。其中菲涅耳轉換電路的計算時間為 10.91(ms)，產出量為(Throughput)為 24.02Mpixels/sec，而在 Matlab 軟體執行的狀況下花費計算時間為 0.5548(s)，產出量為 0.472Mpixels/sec，相較於軟體執行，本論文的菲涅耳轉換以更快速計算出結果。

表 4.11 自動對焦數位全像重建硬體系統($N = 512$)在 150MHz 下之執行時間(ms)

Pre-transform Unit	FFT Unit	Post-transform Unit	Phase Unwrapping	Evaluation Unit
3.52	**3.70**	**3.69**	**209**	**14.1**

表 4.12 為自動對焦 3D 數位全像重建硬體系統採取不同方式讀取原始影像的執行時間。在本論文中，自動對焦還原系統需先針對不同焦距進行表格建置以及全像還原運算並且進行評估。在此過程中，需不斷重新讀取原始影像，重覆數次還原以及評估。最後根據評估值，取得最佳焦距，進行最後一次還原更新。從表中觀察可得知將原始影像資料放到 DRAM 上，再傳輸到 On-Chip RAM，需消耗 100ms，若是透過 SGDMA 可將傳輸時間再減少至 21ms，有效減少每次對焦重新使用原始全像圖的傳輸時間。

表 4.12 自動對焦 3D 數位全像重建硬體系統在不同方式下讀取影像時間(ms)

DRAM to On-Chip RAM using SGDMA	DRAM to On-Chip RAM using IOWR
21	**100**

接下來對本論文提出的自動對焦系統進行效能及正確性分析，本論文的全像圖是在波長= 658nm、數值孔徑(Numerical aperture, NA) = 0.4、放大倍率 M = 13x 以及介質速率n_m= 1 的環境下進行拍攝，以這些參數計算M^2DOF(公式 2.3.6)，計算結果為 0.695mm。以此為 Search Step 從全像圖相距 0.695mm 處開始進行繞射計算與相位展開，每次與全像圖拉遠距離再次計算，硬體實驗中 step size 設定為 0.695mm，共採樣了 70 個焦距點，分別對相同全像圖進行重建，相位還原展開迭代次數超過 150 回則判定其無法完成還原，再對還原結果進行清晰度(Normalized Variance)評估。圖 4.2 是以 micro lens 為樣本進行對焦的結果，橫軸為 0.695mm~48.65mm 的焦距，縱軸為 70 個焦距點全像重建影像的評估值，可以觀察到距離全像 37.53mm 所重建的影像具有最高評估值。我們選擇失焦的影像與成功對焦的影像比較，圖 4.3(a)為焦距 5.56mm 所重建的影像，圖 4.3(b)為焦距 21.55mm 所重建的影像，圖 4.4 則是 37.53m 成功對焦的重建影像。圖 4.4(a)為後轉換計算完成後的相位俯視圖，以灰階圖顯示，圖 4.4(b)為相位展開之影像，可以發現失焦還原的影像都不夠完整。

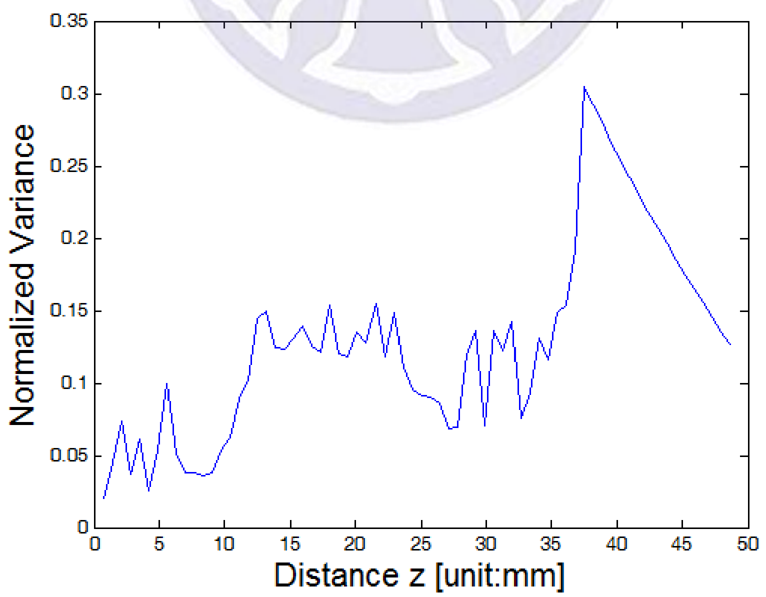
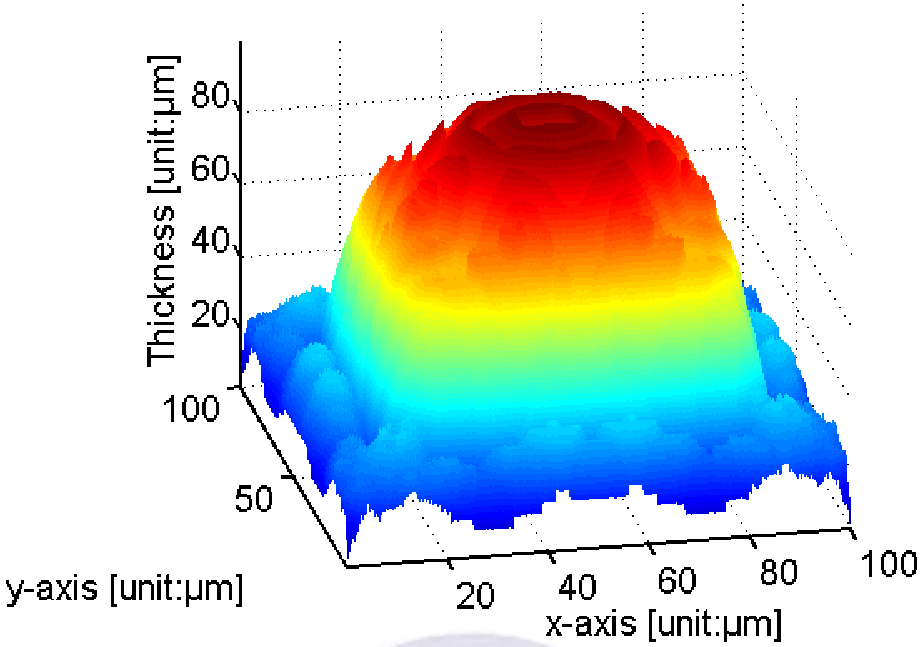
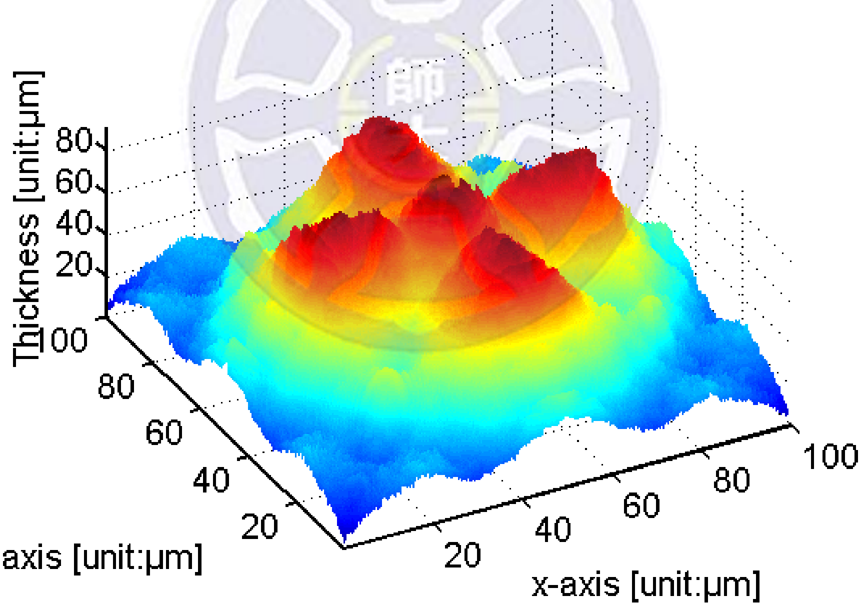


圖 4.2 micro lens 自動對焦評估結果

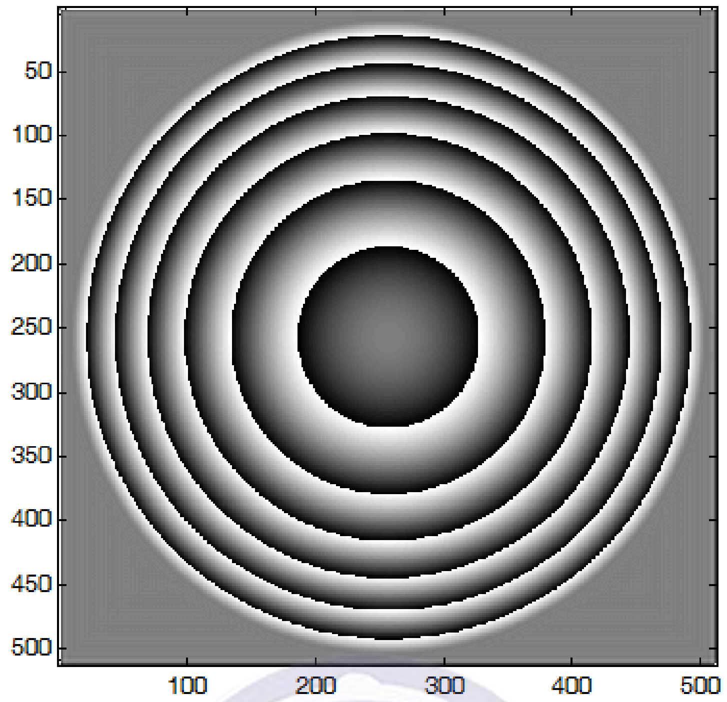


(a)

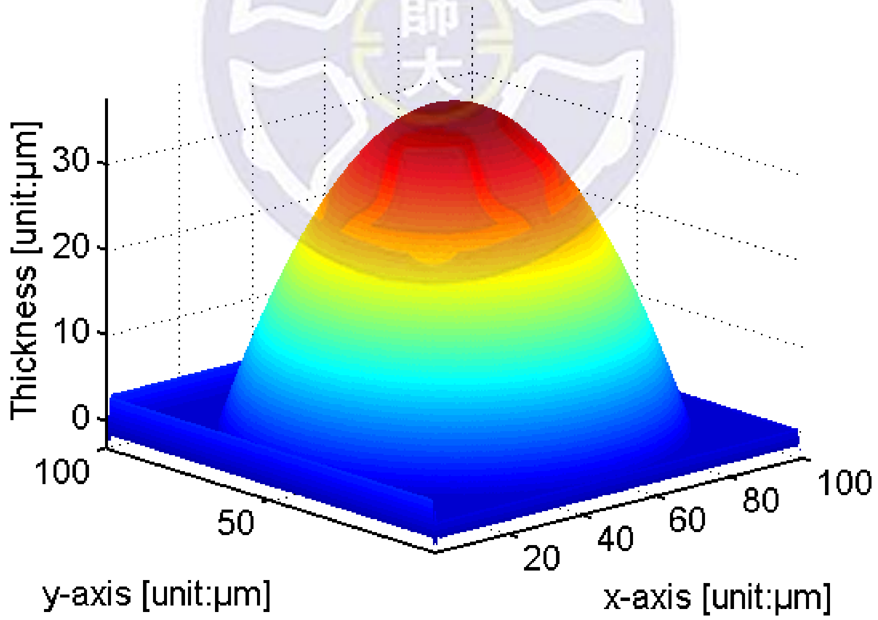


(b)

圖 4.3 Off-focus image (micro lens)：(a) 焦距 5.56mm (b) 焦距 21.55mm



(a)



(b)

圖 4.4 In-focus image (micro lens)：(a) wrapped phase (b) unwrapped phase

本實驗第二個樣本是多個透鏡陣列(micro lens array)，由多個 micro lens 整齊排列而成的樣本，本論文使用的相位還原法則為 Regional (7×7) ，可正確地相位展開。圖 4.5 為以 micro lens array 為樣本進行自動對焦的結果，橫軸為 0.695mm~48.65mm 的重建焦距，縱軸為針對 70 個焦距重建影像的評估值，從圖中可得知，最高的評估值坐落在 38.92mm 處，圖 4.6 一樣舉出兩個失焦的例子，圖 4.6(a)為焦距 13.9mm 所重建的影像，圖 4.6(b)則是焦距 45.87mm 所重建的影像。圖 4.7 是成功對焦的重建影像，圖 4.7(a)是後轉換計算之後的相位俯視圖，一樣以灰階呈現。圖 4.7(b)是相位展開還原後的影像，可以觀察到失焦與對焦成功的影像差異很大，在經過評估分析後，取得最佳焦距便能完整還原全像圖資訊。這兩個樣本自動對焦的結果顯示本論文提出的架構可以正確找到理相的重建焦距。

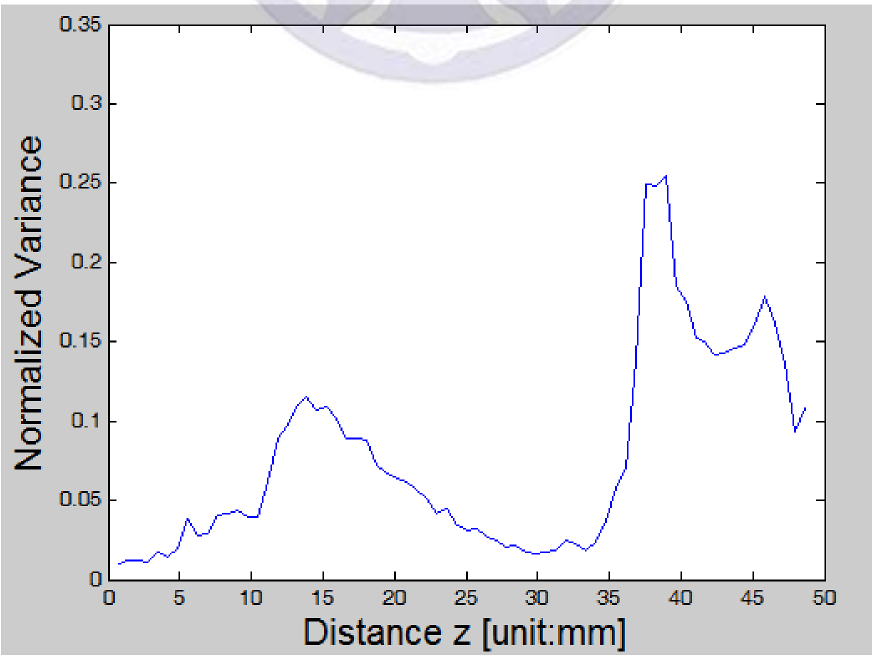
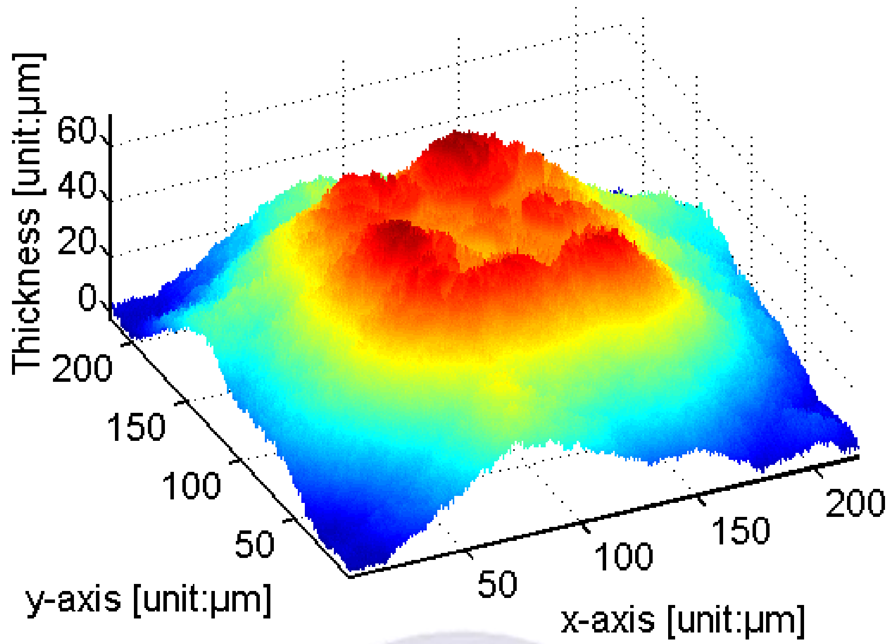
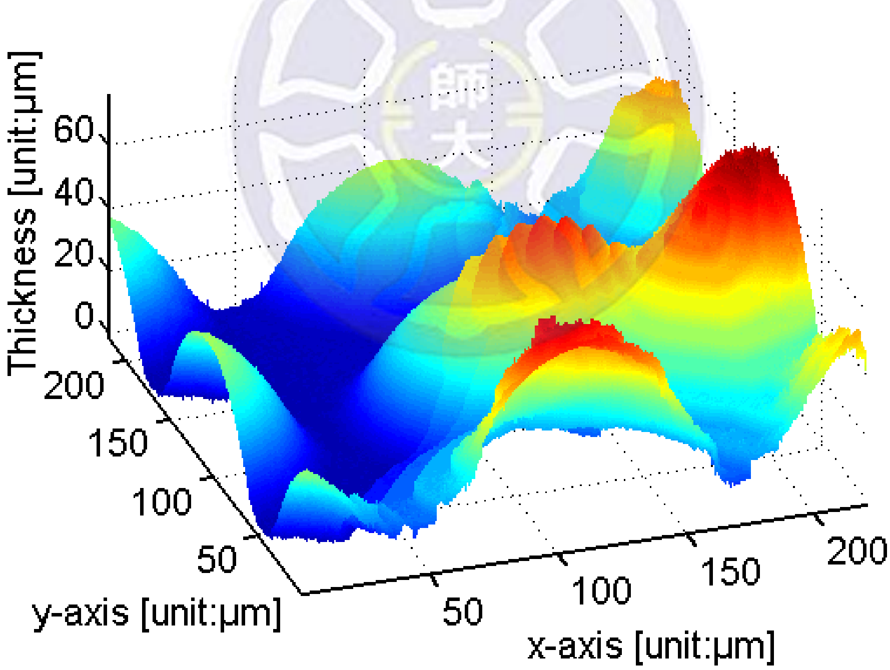


圖 4.5 micro lens array 自動對焦評估結果

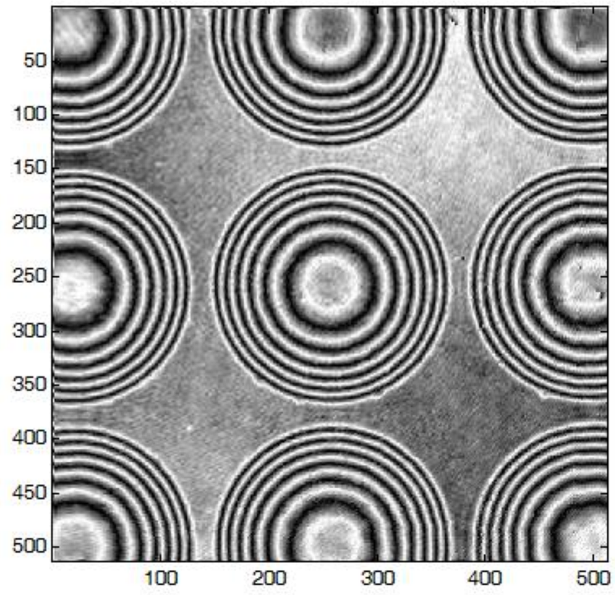


(a)

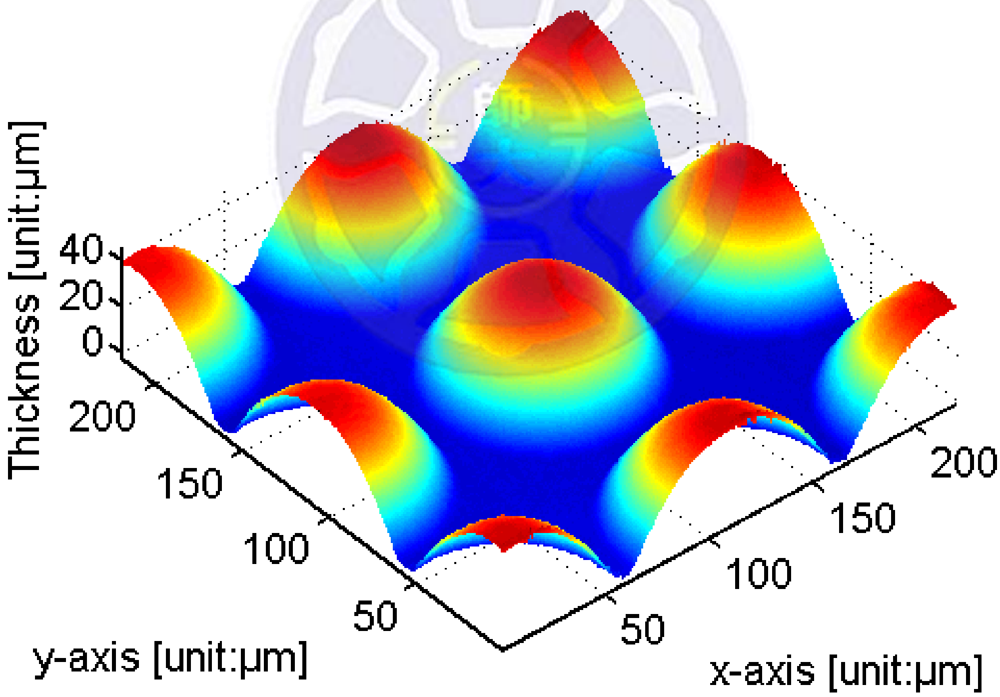


(b)

圖 4.6 Off-focus image (micro lens array)：(a) 焦距 13.9mm (b) 焦距 45.87mm



(a)



(b)

圖 4.7 In-focus image (micro lens array)：(a) wrapped phase (b) unwrapped phase

表 4.13 為硬體與軟體 Matlab 執行自動對焦的執行時間比較結果，測量結果自動對焦 3D 數位全像重建硬體系統對 micro lens 樣本進行重建搜尋 70 個焦距點所花費的時間為 18.304(s)，而 Matlab 需要花費 723.683(s)；micro lens array 樣本進行重建搜尋 70 個焦距點所花費的時間為 21.957(s)，而 Matlab 需要花費 749.865(s)；

表 4.13 自動對焦系統之執行時間(s)

	Hardware in FPGA (150MHz)	Software in Matlab (1.6GHz)
Micro lens	**18.304**	**723.683**
Micro lens array	**21.957**	**749.865**

第五章 結論

本論文實現適用於數位全像顯微鏡應用的自動對焦 3D 數位全像重建硬體系統，並且擁有大尺寸全像重建與解決調整焦距時原始全像圖重覆使用問題。經過軟體驗證，硬體實際運算結果與軟體執行結果相互驗證後確認能完整重建。本論文提出的硬體架構對於可攜式數位全像還原具有下列優勢：高速運算、低功率消耗、自動對焦、大尺寸全像重建。本架構使用 IEEE754 浮點數運算，有效提高精準度。且各運算單元內部採用管線化設計提高運算速度。本硬體架構相較於 GPU 為主的架構或軟體而言，不但具有可攜式與功耗低的優點，且運算速度因管線化設計而相對較快。此外本架構的自動對焦功能比部分現存自動對焦系統更好，進一步的減少重新載入原始全像圖的時間，對於即時運算的嵌入式數位全像顯微鏡應用，本論文提出的架構是更合適的。

參考文獻

[1] U. Schnars and W.P. Jueptner, Digital Holography, Springer-Verlag, 2005.

[2] M. Kim, L. Yu and C. Mann, Interference techniques in digital holography, J. Opt. A: Pure Appl. Opt., Vol. 8, pp. S518-S523, 2006.

[3] Y. Sun, S. Duthaler, and B. J. Nelson, Autofocusing in Computer Microscopy: Selecting the Optimal Focus Algorithm, Microscopy Research and Technique, Vol.65, pp.139–149, 2004.

[4] S. Lee, J. Y. Lee, W. Yang, and D. Y. Kim, Autofocusing and edge detection schemes in cell volume measurements with quantitative phase microscopy, Opt. Express, Vol. 17, pp.6476-6486, 2009.

[5] Y. Sun, S. Duthaler, and B. J. Nelson, Autofocusing Algorithm Selection in Computer Microscopy, Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.419-425, 2005.

[6] Dennis C. Ghiglia, Gary A. Mastin, Louis A. Romero, Cellular-automata method for phase unwrapping, Sandia National Laboratories, Albuquerque, New Mexico 87185, 1984.

[7] S. Rajan, S. Wang, R. Inkol, A. Joyal, “Efficient Approximations for the Arctangent Function,” IEEE Signal Processing Magazine, vol. 23 pp. 108-111, 2006.

[8] 陳建廷, 適用於高速菲涅耳轉換之積體電路架構設計, 國立臺灣師範大學資訊工程研究所, 2013.

[9] 陳煥元, 數位全像顯微鏡之高速相位展開法則電路於FPGA上之實現, 國立臺灣師範大學資訊工程研究所, 2014.

[10] 林烝祺, 以 FPGA 實現可自動對焦之 3D 數位全像重建系統, 國立臺灣師範大學資訊工程研究所, 2014.

[11] Tristan Colomb, Nicolas Pavillon, Jonas Kühn, Etienne Cuche, Christian

Depeursinge, and Yves Emery, Extended depth-of-focus by digital holographic microscopy, 2010

[12] Altera Corporation, FFT MegaCore Function User Guide, 2011.