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

以FPGA 實現可自動對焦之 3D 數位全像重建系統

The Implementation of Autofocusing System by FPGA for 3D Digital Holography Reconstruction

研究生: 林烝祺撰

中華民國 一百零三 年 七 月

中文摘要

本論文主要目的為在 FPGA(Field Programmable Gate Array)平台提出一個硬 體電路架構以實現全像圖的 3D 影像重建,此架構提供 3D 影像還原其相位所需的 繞射計算以及相位展開功能,並且能夠對影像還原的焦距進行自動校正。此架構 具有兩大優勢,其一為解決現存以 GPU 為主的 3D 影像還原系統功率消耗較高的 缺點。其二為提供大多 3D 影像重建系統缺少的自動焦距校正功能。此架構中利 用硬體進行菲涅耳轉換(Fresnel Transform)以執行繞射計算,自動焦距校正的部分 則使用 Normalized Variance 數值來評估不同焦距下的影像清晰度。

在論文中我們將以FPGA 來實現上述之硬體架構,並實際量測此架構執行之 效能與功耗,根據實驗結果,本論文提出的架構較 GPU 或其他實現方式更具有 可攜性、低功率消耗以及高速計算的優點。對於嵌入式數位全像顯微鏡(Embedded Digital Holographic Microscopy)等相關應用下是相當適合的設計架構。

關鍵字: FPGA、菲涅耳轉換、自動焦距校正

誌謝

終於從兩年的碩士生涯中畢業,這篇論文以及相關實驗得以順利完成,仰賴 許多人的幫助與鼓勵,在此表達最深的謝意。

首先非常感謝指導教授黃文吉博士,在這兩年中提供我所需的研究資源,對 於相關的知識以及經驗也毫不吝嗇的傾囊相授,在研究開發遇到瓶頸時,教授總 能提供確切的建議和改進步驟,培養我們獨自解決問題的能力,而當我們對研究 的態度產生偏差,教授也會立即給予嚴厲的指正。這些收穫於將來的人生道路上, 都會是我最寶貴的經驗、優勢。

在此感謝特地前來參與我口試,並且擔任口試委員的林群富博士、楊士萱博 士、熊博安博士,在口試階段找出我研究論文中忽略的重點,並且提供我豐富的 建議,這對於此論文的完成是相當重要的關鍵,在口試結束時也不忘鼓勵我,平 靜我緊張的情緒。也感謝吳榮根博士抽空參與實驗室的 meeting,針對我們的研 究提供獨到的見解。

當然,沒有實驗室的學長姐提供完善的開發環境、介面和部分成果,此論文 的實作與延伸無法如此順利,在此感謝台灣師範大學資訊工程研究所多媒體通訊 暨系統晶片實驗室學長姐的貢獻,以及臺灣師範大學光電科技研究所鄭超仁教授 以及賴信吉學長提供研究所需的樣本及相關理論、數據。特別感謝陳建廷學長, 持續給我許多協助、程式撰寫的經驗。感謝這兩年一同奮鬥的好友:皓棠、思淮、 一修、奇恩、光耀、焕元、雅姿。感謝一年級的學弟:映綸、柏佑、建旻、信豪、 元品、承祐,在口試時給予我許多幫忙。

最後要感謝我的家人,父母在我忙於研究時,盡全力的支持我,在最困難的 時候是我最堅強且溫柔的後盾。感謝默默陪著我的佳穎,給我前往目標的動力及 信心。還有許多我曾接受過幫助的人,每個人都在我這個階段的成功中扮演重要 的角色,我將銘記於心,僅以此誌謝表達至上的感謝。

誌謝

目錄

中文摘要	i
目錄	iii
表目錄	iv
圖目錄	v
	1
	····· ·
1.1 研究背景	1
1.2 動機與目的	
1.3 研究方法	5
1.4 全文架構	7
第二章 基礎理論及技術背景介紹	
2.1 菲涅耳轉換	
2.2 相位展開之基礎理論	
2.3 自動對焦系統之基礎理論	
2.4 討論菲涅耳轉換及自動對焦於數位全像顯微鏡之應用	15
	10
2.5 FPGA 系統設計	
2.5 FPGA 系統設計 第三章 系統架構	
 2.5 FPGA 系統設計 第三章 系統架構 3.1 菲涅耳轉換 	
 2.5 FPGA 系統設計 第三章 系統架構 3.1 菲涅耳轉換	
 2.5 FPGA 系統設計 第三章 系統架構	
 2.5 FPGA 系統設計	
 2.5 FPGA 系統設計 第三章 系統架構	
 2.5 FPGA 系統設計	
 2.5 FPGA 系統設計	

表目錄

表 1.1 FPGA 與	GPU 架構之特性比較	4
表 4.1 Altera Stra	atix IV EP4SGX230KF-40C2	
表 4.2 菲涅耳轉	率换與相位展開各單元資源複雜度比較	
表 4.3 搜尋與評	² 估系統中各單元資源複雜度比較	
表 4.4 菲涅耳轉	率换與相位展開電路各單元之 ALMs 資源消耗表	
表 4.5 搜尋與評	² 估系統各單元之 ALMs 資源消耗表	
表 4.6 菲涅耳轉	p換與相位展開電路各單元之 Memory bits 資源消耗表	
表 4.7 搜尋與評	₽估系統各單元之 Memory bits 資源消耗表	
表 4.8 菲涅耳轉	p換與相位展開電路各單元之 DSP blocks 資源消耗表	
表 4.9 自動對焦	系統各單元之 DSP blocks 資源消耗表	
表 4.10 整體電話	路在 DE4 開發板之資源校耗表	
表 4.11 自動對;	焦系統在頻率 500 MHz 下各單元之執行時間(ms)	45
表 4.12 自動對	焦系統在 3.4GHz 環境下搜尋 500 點之執行時間(s)	
表 4.13 FFT 單方	元在頻率 500MHz 下之功率消耗(W)	

圖目錄

圖 2.1 全像圖之重建流程16
圖 2.1 NIOS 系統架構圖19
圖 3.1 自動對焦系統架構圖
圖 3.2 菲涅耳轉換硬體架構圖
圖 3.3 前轉換單元電路架構圖
圖 3.4 1D FFT 模組之運作圖
圖 3.5 2D FFT 之列運算
圖 3.6 FFT 單元架構圖: (a) FFT 列運算模式, (b) FFT 行運算模式
圖 3.7 後轉換單元電路架構圖
圖 3.8 整體菲涅耳轉換運算
圖 3.9 搜尋單元架構圖
圖 3.10 表格建置模組架構圖
圖 3.11 表格建置模組配合菲涅耳轉換之運作流程
圖 3.12 評估單元架構圖
圖 4.1 Altera Stratix IV EP4SGX230KF-40C2 開發版外觀
圖 4.2 Microlens 自動對焦結果
圖 4.3 Off-focus image (microlens): (a) 焦距 5 mm, (b) 焦距 500 mm
圖 4.4 In-focus image (microlens) : (a) wrapped phase, (b) unwrapped phase
圖 4.5 Microlens array 自動對焦結果49
圖 4.6 Off-focus image (microlens array): (a) 焦距 5 mm, (b) 焦距 500 mm50
圖 4.7 In-focus image (microlens array) : (a) wrapped phase, (b) unwrapped phase51

第一章 緒論

本章節主要在探討本論文的研究背景與動機、研究目的與方法,並大略說明 各章節的主要內容與重要特性。

1.1 研究背景

數 位 全 像 術 (Digital Holography) [1] 是 一 種 利 用 感 光 耦 合 元 件 (Charged-Coupled Devices, CCDs)有效紀錄物體光波绕射於物體表面的相位與振 幅資訊的技術,經過光學的绕射計算後可將此裝置紀錄之全像圖(Hologram)的光 波資訊重建為原始的 3D 物體影像。由於數位全像術具有可將影像以數位的形式 儲存的優點,因此這種技術現今被大量投入運用在工業測量、生物醫學工程的生 物細胞檢驗等領域。在這些數位全像的應用之中,數位全像顯微鏡(Digital Holographic Microscopy)即為其中一種非侵入性質的測量工具,可以重建細微物體 的原始 3D 影像。在如今無線網路傳輸與行動裝置的快速發展下,全像術的應用 也將越來越多元且方便快速。

3D影像重建的繞射計算可使用菲涅耳轉換、摺積法與角頻譜繞射法等等[2], 然而這些法則都有相當高的計算複雜度,再加上繞射計算完成後所得到之相位數 值為壓縮在-π~π之間的不連續相角(wrapped phase),造成相位圖呈現非線性分布, 一般而言利用相位展開法則還原原始物體的真實相位是最常用的解決方式。在本 論文中使用菲涅耳轉換進行繞射計算搭配相位展開法重建全像圖,菲涅耳轉換僅 需一次快速傳立葉(Fast Fourier Transform)即可加速完成全像圖的數值重建,相較 於需要進行三次傅立葉轉換的摺積法能降低更多計算複雜度。儘管如此,以一般 的軟體欲實現即時 3D 數位影像重建仍然相當困難。為了克服此瓶頸,針對各種 法則的加速硬體架構相繼被提出,包括使用 GPU 計算為基礎以實現菲涅耳轉換 法則的硬體架構[3,4,5],同樣使用 GPU 架構實現摺積法則的架構[3,6,7]和 GPU 計 算角頻譜的架構[8],這些架構的共通點為使用 GPU 的平行運算優勢提高執行速 度,然而這些架構的功率消耗也隨著效能提高而增加,這些負擔對於普遍的行動 運算裝置或是嵌入式系統是難以負荷的。由於此限制,不少以 FPGA 設計來減少 功率消耗的全像圖重建系統開始被重視,包含以 FPGA 實現傅立葉全像圖[9]、摺 積法[10,11]和角頻譜法則[12,13]的硬體架構。

除了計算複雜度和功率消耗的困難之外,绕射計算的過程中需要鎖定全像圖 與呈相影像間的焦距,否則影像重建將會失敗,傳統的對焦方法為人工方式判斷 還原影像是否在準確焦距內,但此種方法缺乏效率,亦無法應用於即時的影像重 建上。因此,自動對焦功能對於即時的 3D 影像呈現是必須的,一般而言,成功 對焦的影像相較於失焦的影像有更大的高頻能量,也就是說清晰影像中物體邊界 與相鄰像素灰階值的差異值或梯度值較大。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)計算雨種。梯度值法如 Laplacian、梯度總和計算等[14,15];差異值法如 Tamura coefficient[16]、Normalized Variance、影像強度差異(amplitude)計算等,這些廣泛使用於自動對焦實作上。

1.2 動機與目的

以數位全像顯微鏡來說,為了完整重建微奈米級觀測物體的原始 3D 影像並 達到即時呈現的目的,上述硬體效能與功率消耗的問題顯然相當重要。此外光學 顯微鏡在生物學方面的應用常常遇到兩個問題,其一為顯微鏡所觀測的樣本極有 可能需要時常更換或者對象為會活動的微生物,這種情形下我們未必能事先知道 正確的焦距,若數位全像顯微鏡不能進行自動對焦以重建還原,則觀測者就無法 順利重建影像並且觀測樣本。其二為這類微小生物的樣本在環境因素變動下很容 易遭到破壞,往往無法在樣本採取後完整運送到實驗室進行觀察,或者在保存樣 本上花費大量成本與時間,所以針對數位全像顯微鏡的行動式硬體裝置就相對重 要,如此一來,可攜式數位全像顯微鏡在距離實驗室遙遠的地區取得觀測物的全 像圖並且當下對其影像相位重建後,只要透過筆記型電腦或是其他顯示裝置將完 整的 3D 影像顯示出來,觀測者就可以對物體的各種角度進行觀察和數據紀錄。 為了解決這些問題,我們採用 FPGA 為實現平台,此架構對比 GPU 平台的優勢 與弱點如表 1.1 所示。

3

表 1.1 FPGA 與 GPU 架構之特性比較

	FPGA 為主之架構	GPU 為主之架構			
可攜帶性	百	低			
功率消耗	低	百同			
運算速度	低	高			

本論文所提出的硬體架構具有下列優勢:

- 1. 高速繞射計算
- 2. 高可攜帶性
- 3. 低功率消耗
- 4. 自動焦距校正功能

這些優點將達成此論文架構的主要目的,也就是提出一種基於 FPGA 實現的硬體 架構,此架構相較於 CPU 計算、GPU 硬體電路甚至現存的其他 FPGA 架構更適 合使用於嵌入式數位全像顯微鏡等行動裝置的數位全像術應用,提供使用者或研 究人員更即時的影像資訊以及便利性和更低的資源消耗。

1.3 研究方法

本論文中的主要設計是以菲涅耳轉換進行繞射計算,並對其結果執行相位展 開還原完整影像,最後對此還原影像進行 Normalized Variance 清晰度評估與自動 焦距校正。

本論文之電路設計中,菲涅耳轉換法可分為三部分,前轉換單元(Pre-transform Unit)、快速傳立葉轉換單元(FFT Unit)、後轉換單元(Post-transform Unit),菲涅耳 轉換法只需一次快速傳立葉轉換運算,儘管在前轉換單元與後轉換單元仍需要進 行大量計算,在本論文架構中可利用查表法簡化電路以減少計算複雜度,因此大 部分計算時間都集中於影像的載入和快速傳立葉轉換上,在這三個單元中電路皆 以IEEE 754 單精度福點數格式進行計算,相較於定點數(Fixed-point)的計算來說 浮點數可表示的範圍更大、精確度較高、也不需要在不同硬體單元間調整,對於 快速傳立葉轉換等數值變動差異大且精確度需求高的硬體架構而言是相當適合 的表示格式。

相位展開法是採用細胞自動機(Cellular Automata)[17]為基礎的硬體架構,此法則可以還原出較複雜的圖形,並且具有高速運算及低消耗的優點。

自動焦距校正的實現可分為兩步驟,首先決定重建焦距的範圍與調整單位, 以每個焦距逐一對全像影像進行菲涅耳轉換計算與相位展開,每次重建完成後評 估影像的完整性和清晰度,第二步驟為比較其評估結果,其中最高清晰度影像的 焦距即為理想焦距。 本論文所使用的測試樣本為國立台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),其由 資訊光電實驗室的數位全像顯微鏡拍攝、產生全像圖作為本論文的實驗輸入影像。 這些電路整合在 FPGA 平台來實際執行計算以及測量運算效能,我們使用可程式 化系統晶片(System on Programmable Chip; SoPC)中的硬體加速器來實現,此SoPC 系統為 Altera 公司提供的 Qsys 平台所設計(屬於 Network on Chip, NoC 架構),在 本論文設計中,此SoPC 系統由一 NIOS processor、本論文提出的電路架構以及其 他所需的周邊元件組成。藉由這種系統架構我們可以不斷地修改及驗證設計中的 電路,使研究的過程具有設計彈性與便利性。

效能的考量項目包含 3D 影像還原的清晰度、系統運算時間以及硬體電路資源消耗(area cost)。資源消耗的測量方式即為以 FPGA 實現本論文架構時所消耗的 資源,測量工具使用針對 FPGA 設計的 VLSI 開發軟體—Altera Quartus II[18],包 含本硬體架構以及 Qsys 系統都將一併測量。影像清晰度比較和電路執行時間則 由 NIOS processor 作為 CPU 進行量測,此執行時間包括菲涅耳轉換、相位展開法 和自動對焦的花費時間,與模擬時間不同,此為貼近 3D 影像還原之嵌入式系統 實作的實際執行時間,所得數據較具有實用價值。軟體驗證方面,利用 MATLAB 強大的數學計算能力和繪圖處理能力,最終電路的輸出結果將與 MATLAB 軟體 相互驗證計算正確性與精確率,並且利用 MATLAB 進行可變更視角的 3D 呈像。

6

1.4 全文架構

本篇論文共分為五個章節,以下為各章節內容概述:

【第一章】緒論

說明本論文的研究背景、動機、目的、方法及本文架構。

【第二章】基礎理論及技術背景介紹

簡介本論文所使用之演算法、理論基礎、技術背景及本論文在數位全像顯微鏡中的應用。

【第三章】系統架構

介紹所提出的菲涅耳轉換以及搜尋與評估電路的設計架構,並提供各電路內部的討論與說明。

【第四章】實驗數據與效能比較

包含了系統環境的說明、相關實驗數據分析以及軟硬體效能比較。

【第五章】結論

對於所提出之硬體架構以及實驗的結果進行總結。

第二章 基礎理論及技術背景介紹

本章節將介紹本論文所使用的基礎理論與技術背景。首先,本章的第一節將 介紹本架構執行繞射計算所使用之菲涅耳轉換(Fresnel Transform)法則的基本原 理及運作流程,第二節討論實現本論文架構所搭配參考的的相位展開法,第三節 則介紹自動對焦系統的基礎理論,評估影像清晰度採用 Normalized variance 計算。 第四節針對上述之法則討論其在數位全像顯微鏡中的應用與步驟。最後,第五節 介紹 FPGA 技術與 NoC 的系統整合設計,提供整體系統架構的實驗設定說明。

2.1 菲涅耳轉換

菲涅耳轉換法又可以稱為菲涅耳近似式(Fresnel approximation),在本論文是 用於數位全像顯微鏡應用中影像重建的繞射計算上,數位全像顯微鏡透過 CCDs 裝置拍攝並以數位形式儲存全像圖,令此全像圖資訊為η,平行於此全像圖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),

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

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。

由於全像圖 η 在 CCD 中是以離散形式表示的,這對於數位全像顯微鏡的菲涅 耳轉換也是必要的,假設數位全像顯微鏡在x,y軸方向皆以採樣週期 Δ_f 共採樣了 $N \times N$ 個樣點(sample),其離散形式的菲涅耳轉換法如公式(2.1.2),

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

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

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

在公式(2.1.2)中,與η_{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),

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

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

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

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

$$\alpha = \frac{-j}{\lambda z} e^{j\frac{2\pi}{\lambda}z}$$
(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 $\omega_u \mathcal{D} \omega_v$,如公式(2.1.9)所示,

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

$$\phi_{u,v} = tan^{-1} \frac{Im(B_{u,v})}{Re(B_{u,v})}$$
(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[19]來求得近似解, 如公式 (2.1.12),

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

另外,由於計算出的(Q/I)不一定符合公式(2.1.12)的條件-1≤Q/I≤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,詳細使用方法將於第三章做說明。

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

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

$$(2.1.14)$$

整理以上的公式,此章節所描述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 $\eta_{x,v}$, $0 \le x < N \ge 0 \le y < N$ 。

步驟二:利用公式(2.1.6),計算函數px,v。

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,

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

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

步驟四:使用公式(2.1.10)對二維傳立葉轉換計算出的*tu,v*做菲涅耳轉換最後的計算,求出函數*Bu,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 相位展開之基礎理論

由於 3D 數位全像的重建結果需要執行相位展開才能完整還原觀測物體的影像,因此本論文採用以細胞自動機(Cellular Automata)為基礎的相位展開法則來實現,在電路實作中,我們使用了基於此法則在數位全像顯微鏡上的高速相位展開硬體架構[20]。

細胞自動機最早由 Von Neumann 於 1950 年代以細胞的繁殖及消滅等概念所 提出,其為一離散模型,可被視為由許多格子所組成的架構,每個格子皆為有限 狀態,而這些狀態決定於前一時間點各鄰居格子的狀態。

細胞自動機的技術原理為將目前處理的 current pixel 透過運算單元(cell)進行 運算,同時 current pixel 會參考鄰近領域的 pixel 進行運算,如公式(2.2.1),

 $Current Pixel = \begin{cases} Current Pixel + 2\pi, if Pixel - Current Pixel > \pi \\ Current Pixel, otherwise \end{cases}$ (2.2.1) 此相位展開法則每個參考領域皆只有一個相位點,當任一參考領域的相位點較 Current Pixel 之相位值大於π時,則 Pixel 之相位值增加2π,以此逐步掃描並修正 整個影像,遞迴式地將相位展開。

2.3 自動對焦系統之基礎理論

本論文中使用的自動對焦機制是針對不同焦距下重建完成的影像進行清晰 度評估,其中評估數值最高的影像即代表最佳焦距範圍內的重建影像,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中要根據焦距變化重 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 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:

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

其中σ(P)代表重建影像P的標準差,而γ(P)則為影像P的平均值。

$\gamma(P)$ 與 $\sigma(P)$ 的計算式為

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

$$\sigma(P) = \sqrt{\frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{\nu=0}^{N-1} \left[P_{u,\nu} - \gamma(P) \right]^2}$$
(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(2.2.4),

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

公式(2.2.4)中的σ(P)²可以由公式(2.2.3)平方得到,

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

由於每個像素的總和可以表示為: $\sum_{u=0}^{N-1} \sum_{v=0}^{N-1} P_{u,v} = N^2 \gamma(P)$,所以公式(2.3.5)可以

展開並且化簡為公式(2.2.6),

$$\sigma(P)^{2} = \frac{1}{N^{2}} \left[\sum_{u=0}^{N-1} \sum_{\nu=0}^{N-1} P_{u,\nu}^{2} - 2\gamma(P) [N^{2}\gamma(P)] + N^{2}\gamma(P)^{2} \right]$$
$$= \frac{1}{N^{2}} \sum_{u=0}^{N-1} \sum_{\nu=0}^{N-1} P_{u,\nu}^{2} - \gamma(P)^{2}$$
(2.3.6)

如此化簡的目的是為了配合硬體設計中必須將影像每個像素值讀取並且計算過 一遍才能得到平均值γ(P),若要計算每個像素與平均值之間的差異則需要讀取影 像兩次,這對於影像處理來說是效能上很大的負擔,將算式整理過後便可以只讀 取一次影像同時計算出P_{u,v}²以及γ(P)²。因此透過公式(2.3.5)與(2.3.6)可以推導出

$$V(P) = \frac{1}{\gamma(P)} \left[\frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} P_{u,v}^2 - \gamma(P)^2 \right]$$
$$= \frac{1}{\gamma(P)} \left[\frac{1}{N^2} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} P_{u,v}^2 \right] - \gamma(P)$$
(2.3.7)

在本論文中使用 Normalized Variance 而非 Tamura measure 的原因是在資源與誤差範圍可容許的情況下這個方法沒有使用任何的平方根運算(square root),如此一來, 本論文的系統就能減少相當多的計算複雜度以及平方根電路的硬體資源。

經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。完整自動對焦系統流程為下列步驟: 步驟一:決定一與全像圖平行之焦距範圍,並且選擇數個焦距點z。 步驟二:選擇一焦距z進行繞射計算並且相位展開完成影像重建。

步驟三:對步驟二之重建影像進行清晰度評估。

步驟四:變更焦距Z並重複步驟二與三,最後比較出最大V(P)值。

2.4 討論菲涅耳轉換及自動對焦於數位全像顯微鏡之應用

全像術的主要概念就是紀錄並且還原影像,傳統全像圖通常由底片拍攝而成, 於 1967 年 Goodman、Laurence[21]提出利用電腦紀錄及重建全像圖的方法,儘管 當時的科技還不成熟,使這種需要大量運算的影像處理系統無法充分的表現還原 結果,當然這樣的問題在這幾十年有了很大的突破,隨著電腦科技的進步已逐漸 發展成使用高效率且快速的電子式 CCD/CMOS 影像感測器來記錄全像圖資訊, 而非使用傳統的底片紀錄,影像重建的部分也以快速的數值計算還原出原始物體 的完整振幅與相位。

我們將這種全像術的技術應用到光學顯微鏡上,目的是還原出微生物或超小 型物體的 3D 影像,並且放大展現於人們眼前,數位全像顯微鏡同樣也是記錄全 像圖再進行繞射計算以重建影像,所記錄的全像圖看似一張灰白條紋相間的圖片, 這些條紋與原始物體的影像沒有任何幾何上的相似性,也就是說,對人類而言這 樣的全像圖片是難以辨識的,所以需要透過重建將影像還原出來,本論文即透過 菲涅耳轉換取得相位值,在此計算過程中我們將不斷調整焦距以確定我們所觀察 的樣本或微生物能完整還原,而這個過程需要搭配相位展開來實現。



圖 2.1 全像圖之重建流程

圖 2.1 為本論文重建全像圖同時進行自動焦距校正的流程,取得全像圖的複 數振幅資訊η_{x,y}之後,將其作為菲涅耳轉換的輸入資料,也就是將之進行繞射計 算,藉由計算菲涅耳轉換公式獲得原始物體的複數振幅資訊B_{u,v},並利用反正切 函數取得相位值φ_{u,v},此時的相位數值會被壓縮在-π~π之間,形成 wrapped phase, 因此可以再利用相位展開法則重建出真實的相位P_{u,v},取得完整相位之後就能對 其進行清晰度評估,以不同焦距重複進行菲涅耳轉換、相位展開以及清晰度評估 後,即可找出最佳的重建影像和適當的焦距。 將這些流程應用到數位全像顯微鏡上,我們可以達到將細微的物體 3D 呈現 在眼前的目的,最重要的是,有了物體的相位資訊,完成重建的 3D 影像可以任 意的放大縮小,大如螞蟻般的昆蟲、血管裡的紅血球、小至人類的神經細胞等等, 都可以放大到肉眼所見的比例,甚至遠大於人類比例的巨大影像也能做到,此影 像還能 360 度的旋轉,讓觀測者觀察物體任何一表面的細節,對於光學測量和生 物醫學都是相當有幫助的技術。

本論文的實驗流程即是應用於數位全像顯微鏡上,首先由國立台灣師範大學 光電科技研究所資訊光電實驗室利用 CCDs 感測器記錄物體的振幅資訊,照射光 源為一固定波長的雷射光,樣本則為本論文第一章節所描述的微透鏡,此物體是 光學應用上一種常見的測試元件,表面形狀為一曲面、中空的球柱狀結構,顏色 趨近透明。數位全像顯微鏡將此透鏡拍攝成肉眼難以辨識、灰白條紋的全像圖, 但也因此得以數位的形式儲存在硬體電路中,我們對此全像圖執行菲涅耳轉換、 反正切函數運算和相位展開後,就能對重建影像評估其清晰度,當範圍內的焦距 點z都進行過重建以及評估後,我們就能找到最清晰的還原影像所需要的焦距為 何,接下來只要以此焦距進行最後的重建,所還原出來的影像即為完整的影像。

17

2.5 FPGA 系統設計

近年來,積體電路(Integrated Circuit, IC)的發展已經越來越成熟,隨著製程技術提高,卻造成其生命週期日益下降、功率需求提升,所要求的功能複雜度也提高許多,為了滿足這些應用的嚴格要求與挑戰,特別是隨著設計專案規模越來越 大時,硬體電路的設計師們便需要一個易於使用、靈活且彈性的設計環境來探索 不同的設計實現,以達到其成本、功率消耗和性能指標,因此FPGA越來越常使 用在對成本敏感、功耗敏感的大量應用上。

本論文使用硬體描述語言(Hardware Description Language, HDL)搭配可重複 程式設計的FPGA晶片設計數位電路,可縮短研發時間且更有彈性,經過系統提 供的自動合成繞線佈局,可快速地燒錄至FPGA上進行測試,最重要的是不需要 負擔實際合成電路卻失敗的高成本風險,是現代IC設計的技術主流。目前FPGA 系統的開發,允許軟體與硬體共同設計,來達成系統化的IC設計,具有開發時間 短及系統可修改的優點。

Altera公司根據不同使用者的需求,提供許多不同系列的FPGA開發板,本論 文是使用NIOS development kit中的Stratix IV EP4SGX230KF-40C2系統開發板來 實現菲涅耳轉換法則以及自動對焦系統電路。

18



圖 2.1 NIOS 系統架構圖

圖2.2為一個簡單的NIOS範例結構,在NIOS系統中提供了一套專門給NIOS II processor使用的匯流排(Avalon switch fabric或Avalon bus),它由Qsys平台自動產生, 有自己的資料交換協定,將系統中的NIOS II processor與其他設備端口連接起來。 設計師所撰寫出的硬體電路被視為一個客製化邏輯電路(custom logic circuit),透 過此匯流排,將客製化電路掛在Avalon bus上與整個系統溝通。

使用 NIOS II 軟體開發工具設計系統的優點為可以透過C語言指令及函式庫 來控制電路,電路輸出的結果也能以軟體得到有效的驗證。整體系統的設計步驟 是將由 qsys 所建立的 SoPC 系統在 FPGA 上程式化後,在 NIOS 開發工具上以C 語言撰寫的程式執行並觀察其電路的輸出結果即可。

第三章 系統架構

於此章節將詳細介紹本論文自動對焦系統所使用的菲涅耳轉換、搜尋與評估 系統(Search and Evaluation System)之硬體電路架構,分別在第一節將討論菲涅耳 轉換法則的電路架構,接著在第二節討論搜尋與評估系統的電路架構並整理菲涅 耳轉換與此系統搭配相位展開法在整體自動對焦系統中的關係與作業流程。

整體系統架構如圖 3.1 表示,此系統包含重建影像所需的菲涅耳轉換架構、 細胞自動機為基礎的相位展開法以及支援自動焦距校正功能的硬體電路,三者必 需互相搭配運作,才能達成在未知焦距下仍然能完整重建 3D 影像的目的。



Search and Evaluation system

圖 3.1 自動對焦系統架構圖

3.1 菲涅耳轉換

這個架構的主要目標是由 FPGA 實現公式(2.1.2)的計算,此架構將菲涅耳轉 換公式分為三大部分,如圖 3.2 所示,分別為前轉換單元(Pre-transform Unit)、快 速傳立葉轉換單元(FFT Unit)、後轉換單元(Post-transform),此電路包含一組嵌入 式記憶體(On-chip RAM)供這三大單元存取以減少記憶體的存取時間。

在菲涅耳轉換電路中,前轉換單元負責計算公式(2.1.6),從 On-chip RAM 中 讀取數位全像圖資訊η_{x,y},並根據公式(2.1.6)進行計算得到ρ_{x,y},完成後即寫入 On-chip RAM。接著快速傳立葉轉換單元同樣從 On-chip RAM 讀取前轉換單元的 計算結果,根據公式(2.1.7)計算得到τ_{u,v}並且寫入結果。最後,後轉換單元由 FFT 執行結果根據公式(2.1.10)計算出B_{u,v},此即為物體的數位重建影像資訊,經過反 正切函數運算便可以得到相位φ_{u,v},到此部分為止菲涅耳轉換計算完畢,此電路 的輸出結果可經由相位展開電路展開其非線性分布的相位值,並完成原始物體的 影像還原。



圖 3.2 菲涅耳轉換硬體架構圖

3.1a 前轉換單元(Pre-transform Unit)

前轉換單元的運算是根據公式(*)來進行的,也就是說此單元最重要的部分是 計算 μ_x, μ_y 和其與全像圖影像資訊 $\eta_{x,y}$ 的乘法運算,在本架構中為了提升計算速度 而將 μ_x 和 μ_y 的數值事先透過軟體計算儲存在硬體表格中,因為 μ_x 和 μ_y 的位址索引 x, y都是整數,而且 $0 \le x, y \le N - 1$,當 λ, z 和 Δ_f 皆為已知參數時, μ_x 和 μ_y 只有N種 不同結果,因此 μ_x 和 μ_y 各自的表格都將只有N個 entries。

圖(*)為前轉換單元架構圖,包含一個地址產生器(Address Generation Unit, AGU)、事先儲存的表格兩張、兩個複數乘法器以及暫存器,地址產生器的用途 是產生索引(indices)和地址, indices 即為x, y的數值,作為讀取μ_x和μ_y表格的輸入 值。產生的地址則送入 On-chip RAM 中以讀取全像圖η_{x,v},經過電路中複數乘法 器計算完成Px,y後再寫入 On-chip RAM,作為下一單元的資料來源。

由於本架構的乘法器必須支援浮點數的複數乘法計算,要於單一時脈週期中 完成在實現上是有困難的,因此在此電路的設計裡所有乘法器都以多時脈週期來 實現,令 L_1 表示前轉換單元電路中每個像素計算的延遲時間,例如從 $\eta_{x,y}$ 第一筆 資料輸入電路到計算輸出第一筆 $\rho_{x,y}$ 的執行時間即為 L_1 ,因此如圖 3.3 所示, η_{x_1,y_1} 和 ρ_{x_2,y_2} 分別為前轉換單元的輸入與輸出值,依照電路執行順序可知其遵循 $x_1 \ge x_2$,並且 $(x_1N + y_1) - (x_2N + y_2) = L_1$,前轉換單元用於計算 $\{\eta_{x,y}, 0 \le x, y \le$ N - 1}所花費的總時間則為 $N^2 + L_1$ 。



圖 3.3 前轉換單元電路架構圖

3.1b 快速傅立葉轉換單元(FFT Unit)

快速傅立葉轉換單元的目的是計算公式(2.1.7)的τ_{u,v},此單元是由一個地址產 生器和 FFT (1D-FFT) 模組組成,如同公式(2.1.15)、(2.1.16)所描述的,本論文利 用此 1D-FFT 模組來實現 2D-FFT。此硬體單元啟動時,我們每次從 On-chip RAM 以一個列的 array 為單位讀取影像,並且每個 array 各自進行一次 1D-FFT 計算後, 再將其計算結果寫入 On-chip RAM 中的同一個列位址裡,當 FFT 的列運算完成 後,就得到公式(2.1.15)中的β_{x,v}了,接著行運算便以相同的方式開始運作,每次 以一個行的 array 為單位進行 1D-FFT,待所有的行運算完成,並且寫入相對的 On-chip RAM 位址中,2D-FFT 的結果τ_{u,v}就計算完成了。

本單元中的 1D FFT 模組使用 Altera 的 FFT MegaCore function[22]來實現, 由於 1D-FFT 一次只執行單一行或單一列的計算,所以 FFT 的轉換長度為N。 1D-FFT 模組資料輸入與輸出都是單一的,並且為管線化(pipeline)的設計,此外, FFT 模組輸入與輸出的 dataflow 以 streaming 的模式進行操作,所以我們可以進 行持續不間斷的資料處理及運算,同時輸出結果也是連續的 stream 資料。圖 3.4 為 1D FFT 模組運算情形,1D FFT 的計算可分成三個步驟,第一個步驟為從 On-chip RAM 讀取N筆輸入資料,所以在此步驟需要花費N週期來完成,第二個 步驟為執行 FFT 計算,這個步驟會在N筆資料全部輸入後才開始運作,我們此步 驟所需的計算時間記為L₂,最後第三個步驟將計算完成的N筆資料寫入 On-chip RAM,由於第一筆資料輸出時即可寫入記憶體,因此我們可以推算第一筆輸入



圖 3.5 2D FFT 之列運算

資料到第一筆資料輸出之間的計算時間如圖 3.4 所示為N+L2。

為了使用 1D FFT 達成 2D FFT 的運算,AGU 在此扮演重要的角色,FFT 單 元中的 AGU 負責產生地址用以讀取和寫入資料到 On-chip RAM,因為 FFT 同時 間只會有單一的資料輸入與輸出,所以每個週期 AGU 都將產生兩個地址,分別 用於讀取資料和寫入計算結果。另外,因為 1D FFT 模組是採用 streaming 的資料 模式,我們可以將連續列與行的資料不間斷的輸入至 FFT 電路,如圖 3.5 表示,可以得知 2D FFT 中列運算的總計算時間為N₂ + L₂ + N個週期,同樣的在行運算中也需要花費同樣的時間,所以完成 2D FFT 的計算共需要2(N₂ + L₂ + N)個週期。

圖 3.6 為 FFT 的架構圖,圖 3.6(a)展示 FFT 列運算的流程,輸入資料為前轉 換單元的輸出表示為 ρ_{x_1,y_1} ,輸出值表示為 β_{x_2,v_2} ,由圖 3.4 中已知 1D FFT 模組的 延遲時間為 $N + L_2$,所以 $(x_1N + y_1) - (x_2N + v_2) = N + L_2$ 。圖 3.6(b)則為 FFT 行運算的流程,輸入與輸出資料表示為 β_{x_1,v_1} 及 τ_{u_2,v_2} ,且 $(x_1 + v_1N) - (u_2 + v_2N) = N + L_2$ 。



(a)



圖 3.6 FFT 單元架構圖: (a) FFT 列運算模式, (b) FFT 行運算模式

3.1c 後轉換單元(Post-transform Unit)

後轉換單元負責利用公式(2.1.10)之計算重建物體影像Bu,v,如圖 3.7 所示, 後轉換單元的架構與前轉換單元相似,同樣包含 AGU、事先計算好的表格、兩個 複數乘法器以及暫存器,不同的是在後轉換單元為了後續相位展開的計算而增加 了反正切函數的電路。



圖 3.7 後轉換單元電路架構圖

在後轉換單元中預先計算好 $\omega_u n \omega_v \pm L$ 儲存於表格中,做法與前轉換單元中的 $\mu_x \mathcal{R}\mu_y$ 相同,因為 $0 \le u, v \le N - 1$, $\omega_u n \omega_v$ 各自的表格都有N個 entries。後轉換單元的 AGU 也與前轉換單元的部分相同,每個週期將產生查詢表格的索引值(u,v)以及自 On-chip RAM 讀取 $\tau_{u,v}$ 的地址。乘法器計算的結果 $B_{u,v}$ 會被送進反正切函數單元計算出相位 $\phi_{u,v}$,並將此結果儲存在 On-chip RAM,作為後續相位展開的輸入資料。

令 L_3 表示後轉換單元電路中每個像素計算的延遲時間,計算方式為輸入 $\tau_{u,v}$ 到輸出 $\phi_{u,v}$ 需要的時間,因此如圖 3.7 所示, τ_{u_1,v_1} 和 ϕ_{u_2,v_2} 分別為後轉換單元的輸入與輸出值,依照電路執行順序可知其遵循 $u_1 \ge u_2$,並且 $(u_1N + v_1) - (u_2N + v_2) = L_3$,後轉換單元所花費的總時間則為 $N^2 + L_3$ 。

3.1d 菲涅耳轉換架構之運作

在上面三小節中我們討論了本論文所提出架構的運作流程,三個單元是依序 執行的,由前轉換單元先開始計算,產生ρ_{x,y}後寫回記憶體,當記憶體的寫入完 成後才啟動 FFT 單元,同樣後轉換單元的啟動也必須等待 FFT 單元將輸出值τ_{u,v} 全部寫入記憶體後才開始。因此圖 3.8 整理了整個系統依序執行的流程、以及菲 涅耳轉換所需的時間。

Pre-transform FFT		FFT Computation (Rows)	FFT Computation (Columns)	Post-transform
	$\leftarrow N^2 + L_1 \rightarrow$	$\leftarrow N^2 + L_2 + N \rightarrow$	$\leftarrow N^2 + L_2 + N \rightarrow$	$\leftarrow N^2 + L_3 \rightarrow$

圖 3.8 整體菲涅耳轉換運算

3.2 搜尋與評估系統架構

此架構的目標是提供以菲涅耳轉換與相位展開為主的 3D 影像重建系統自動 焦距校正的功能,此架構可以分為三個單元,分別是搜尋單元、3D 重建單元和 評估單元,由這三個單元負責繞射計算的焦距變化處理,以及相位展開後的影像 清晰度評估。

搜尋單元在重建影像與全像圖之間不斷找出最佳的焦距值Z,與菲涅耳轉換協同運作,由搜尋單元所產生的焦距Z將提供給菲涅耳轉換中前轉換單元與後轉 換單元需要使用到的表格μ_x,μ_v,ω_u,ω_v和數值α。

3D 重建單元為一個控制整體自動對焦系統運作的控制器,用以協調整體 3D 數位全像重建系統中菲涅耳轉換、相位展開、搜尋與評估系統之間的運作。

當影像經過相位展開重建完成後,評估電路將對此影像進行 Normalized Variance 清晰度評估,評估完成後將會計算出其影像的差異值,交付搜尋單元比較後就能找出最佳的焦距值。

3.2a 搜尋單元(Search Unit)

圖 3.9 表示搜尋單元的架構圖,包含焦距點生成模組(Point Generation Module) 與表格建置模組(Table Generation Module),焦距點生成模組所產生的每個焦距z將 用於表格建置模組產生表格μ_x,μ_y,ω_u以及ω_v,這些表格在菲涅耳轉換中的前轉換 單元與後轉換單元將會被使用到。

焦距點生成模組的焦距產生方式是在影像重建之前先決定一範圍,在此範圍 中必定存在一點能從數位全像圖完整還原物體影像,重建影像從此範圍最遠處漸 漸靠近全像圖,每次移動產生的焦距點便成為菲涅耳轉換及表格建置模組所需的 參數值。而此模組在本論文中是使用 NIOS 處理器進行焦距點產生的實現。



圖 3.9 搜尋單元架構圖

表格建置模組在自動對焦過程中隨著焦距改變也產生新的表格,儘管這些表 格可以事先透過軟體計算並且儲存起來,但是當產生的焦距點較多時,儲存這些 表格將會消耗大量的硬體資源,並且大大提高電路面積,這對於本論文設計可攜 帶性架構的目的是衝突的,所以我們由電路動態的根據焦距z產生表格。此外, 根據公式(2.1.5)得知μ_x和μ_y可以共用一張表格,同樣ω_u和ω_v也共用一張表格,因 此在表格建置模組中實際上只需要產生兩張N個 entries 的表格,在公式(2.1.8)中α 也需要跟著焦距z改變,但是α只需要一個暫存器儲存數值即可。

圖 3.10 為表格建置模組的架構圖,從圖中我們可以看到此架構的輸入值為焦 距z和前轉換單元與後轉換單元各自讀取表格的索引值x₁和u₁,圖 3.10(a)和圖 3.10(b)分別為μ_x和μ_y的表格建置模組,此架構是根據公式(2.1.5)及公式(2.1.9)所設 計,這兩個公式可以展開成sin與cos的函數,如公式(3.1)與公式(3.2),

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

$$\omega_{u} = \cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_{g}^{2}u^{2}\right] + j\sin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_{g}^{2}u^{2}\right]$$

$$\omega_{v} = \cos\left[\left(\frac{\pi}{\lambda z}\right)\Delta_{g}^{2}v^{2}\right] + j\sin\left[\left(\frac{\pi}{\lambda z}\right)\Delta_{g}^{2}v^{2}\right]$$
(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)

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

由公式(3.1)與公式(3.3)可知μ,ω中只要給予焦距z與索引值x,y,u,υ外就能在前轉 換單元與後轉換單元執行前先計算出表格,公式中Δ²_fx²與u²/Δ²_f是常數與索引值 的計算結果,也可以事先計算儲存在表格中,依照此表格建置的架構就能分別見 算出μ,ω表格中cos與sin的部分。

由於此架構採用單精度浮點數的乘法器與三角函數元件的設計,所以從輸入 焦距與索引值到輸出並儲存至表格需要花費數個週期,令此延遲時間為 L_4 ,以計 $算\mu_x 表格為例, 可知x_1 \ge x_2, 並且x_1 - x_2 = L_4, 因此產生一個表格所需的時間$ $即為<math>L_4 + N$ 。



(a)



(b)

圖 3.10 表格建置模組架構圖

雖然產生一個表格的時間為L₄ + N但是在數位全像重建的過程中並不需要多 花兩倍的時間來產生兩個表格,在本論文架構中,表格建置模組是可以與菲涅耳 轉換共同進行的,在前轉換單元運作時表格ω是沒有被使用到的,此時便可以進 行ω的表格建置,同樣在後轉換單元運作時我們也可以同時進行表格μ的建置,其 運作流程如圖 3.11 表示,在菲涅耳轉換過程中同時更換新的表格,達到調整繞射 計算中焦距的目的。

Pre-transform	FFT unit	Post-transform
ω		μ
$N + L_4$	Ν	$U + L_4$

 $N^2 + L_1$ $2(N^2 + L_2 + N)$ $N^2 + L_3$

Table Generating flow with Fresnel Transform

圖 3.11 表格建置模組配合菲涅耳轉換之運作流程

3.2b 評估單元(Evaluation Unit)

此單元用於評估重建影像的清晰度,電路的輸出值作為評斷影像重建是否失 焦的依據,圖 3.12 為實現 Normalized Variance 差異值評估的硬體架構圖,本架構 是依照公式(2.3.7)設計的,電路中包含一組控制器、兩組累加器(Accumulator)、 除法器三個以及乘法、減法器各一組。評估單元的輸入資料為相位展開後的影像 和讀取影像的地址,此地址由控制器產生,其也用於計算公式(2.3.7)所使用的N², 讀取影像中每個像素時控制器的計數器同時在進行累加,因此當影像讀取完成後, 影像的面積也已經計算完成,接著便能計算出影像的平均值和標準差的部分計算 了。評估單元的輸出資料即為此還原影像的對焦評估值,也就是重建後的影像清 晰度,此評估值會被輸入 3D 重建單元進行比較,給予菲涅耳轉換的焦距Z評估得 到最高V(P)者就是最佳焦距,而P為對焦結果最清晰的影像。令L₅表示評估單元 每個像素的執行時間,因此我們可以將評估單元的整體運算週期記為N² + L₅。



圖 3.12 評估單元架構圖

3.2c 3D 重建單元(3D reconstruction Unit)

3D 重建單元是一個協調菲涅耳轉換、相位展開架構以及搜尋與評估系統之 間的控制器,3D 重建單元的運作可以分為三個步驟,第一步驟為啟動菲涅耳轉 換以及搜尋與評估系統中的搜尋單元以提供菲涅耳轉換所需的焦距。第二步驟, 啟動相位展開架構對菲涅耳轉換的結果進行相位展開的處理。最後,第三步驟啟 動搜尋與評估系統架構中的評估單元來判斷相位展開後的影像清晰度,而評估結 果將會保留在搜尋與評估系統中,此流程會不斷重複,直到所有的焦距點都被菲 涅耳轉換重建一次,並由本單元比較出最佳的焦距值。

在本論文中此單元是由 NIOS 處理器產生數個焦距點,將這些焦距點依序交 由菲涅耳轉換進行繞射計算,並在相位展開後輸入搜尋與評估系統的評估單元評 估影像清晰度,整理所有的評估結果,經過比較後得到最高的評估值,以其相對 的焦距再次執行 3D 數位全像的重建流程就能成功重建原始物體的立體影像。

第四章 實驗數據與效能比較

本章節將展示本論文所提出的菲涅耳轉換架構及搜尋與評估系統整合之成 果、實際效能測量與比較,以及實驗環境的介紹。

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 的 Stratix IV EP4SGX230KF-40C2 FPGA 開發版為主要的實現平台,如圖 4.1 所示。表格 4.1 整理出了此開發版的詳細規 格資訊。選擇以 FPGA 開發版來實現與驗證電路的理由是因為其所擁有的可程式 化系統晶片(System on a Programmable Chip, SoPC)具有可重複修改電路的彈性, 對於硬體設計的驗證相當合適,因此作為本論文實現硬體設計的平台。



圖 4.1 Altera Stratix IV EP4SGX230KF-40C2 開發版外觀

Feature	Stratix IV
Device	EP4SGX230KF-40C2
Adaptive Logic Modules (ALMs)	91,200
Logic Elements (LEs)	228,000
M9K Memory Blocks	1,235
M144K Memory Blocks	22
Total Memory Kbits	17,133
DSP Blocks	1288
Total PLLs	8
Total DLLs	4
User I/Os	744

表 4.1 Altera Stratix IV EP4SGX230KF-40C2

本論文使用 Altera Quartus II 13.0 作為 Verilog 硬體描述語言的撰寫平台, Quartus 提供基礎的語法解析與偵錯、時序分析、初步的邏輯元件繞線布局、產生 規劃檔案以及電路合成等強大功能,可以快速建立系統架構,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 13.0 中使用的是 Qsys 平台,系統 中也包含許多元件如 CPU、客製化電路(菲涅耳轉換、相位展開及搜尋與評估電 路)、記憶體等等,接下來將完成的 SoPC 系統燒錄於 FPGA 開發板上,便可以進 行實質的訊號輸入與輸出驗證系統架構的效能及正確性。Altera Quartus II 13.0 的 執行環境為 Intel® CoreTM i7-3770 CPU @ 3.40GHz、16G DDRIII 記憶體。

為了將硬體電路的資料輸出與軟體互相驗證,Altera 公司提供了採用 Eclipse 為基礎的 NIOS II 軟體作為控制及測試電路的用途。NIOS II 系統提供設計人員在 FPGA 中開發完整的 NIOS II 嵌入式系統所需的軟體工具、應用程式、函式庫及 驅動程式,方便使用者利用此軟體來與 FPGA 開發板溝通互動,令系統的開發更 有效率且具有豐富的功能性。

本論文除了使用 Verilog 硬體描述語言來設計所提出的硬體電路架構,軟體方面也使用 MATLAB R2011b 為平台來設計本系統的演算法則和硬體電路架構的執行結果相互比較。

※ 硬體實現環境

Device : Altera Stratix IV EP4SGX230KF-40C2

CPU: NIOS II 500MHz

Memory: 64-MB DDR2 SDRAM

※ 軟體實現環境

CPU: Intel® Core™ i7-3770 CPU @ 3.40GHz

Memory : DDRIII 16.0 G

MATLAB R2010b

4.2 實驗數據呈現與討論

本節展示 3D 數位全像圖重建的實驗結果與效能分析,首先觀察本論文架構 的菲涅耳轉換及相位展開之資源消耗複雜度,表 4.2 針對前轉換單元、快速傳立 葉轉換單元、後轉換單元、相位展開單元以及嵌入式記憶體五個單元進行分析比 較,加法器、乘法器、除法器和暫存器是本架構的主要構成元素,因此表 4.2 歸 納出五個單元在各元素方面的需求複雜度,假設菲涅耳轉換輸入影像的邊長為N, 輸入影像大小為N×N,可以看出加法器、乘法器、除法器在不同運算單元中的 分析結果類似,不會因為影像的大小而受到影響,隨著影像大小而改變的只有暫 存器數量。

在前轉換單元中使用了兩個複數乘法器,複數乘法器雖然使用到乘法器與加 法器,然而兩者的數量並不會隨著影像輸入的大小而改變,所以複雜度皆為O(1), 除法器則因為在此單元沒有使用所以複雜度為O,另外本單元有使用了 entry 數量 表 4.2 菲涅耳轉換與相位展開各單元資源複雜度比較

	Pre-transform Unit	FFT Unit	Post-transform Unit	Phase Unwrapping	On-Chip RAM
Adders	0(1)	0(1)	0(1)	0(1)	0
Multipliers	0(1)	0(1)	0(1)	0(1)	0
Dividers	0	0	0(1)	0	0
Registers	O(N)	O(N)	O(N)	O(N)	$O(N^2)$

39

為N的 Lookup table,所以暫存器的複雜度為O(N)。快速傳立葉轉換單元中 同樣擁有固定數量的加法器及乘法器,並且沒有使用除法器,所以複雜度與前轉 換單元相同,皆為O(1),而快速傳立葉轉換單元在不同維度時需要不同大小的資 料暫存空間,所以也需要複雜度為O(N)的暫存器資源。後轉換單元的架構與前轉 換單元非常相似,因此加法器與乘法器的複雜度並沒有改變,entry 數量為N的 Lookup table 也同樣產生了複雜度O(N)的暫存器,唯一的不同處是在後轉換單元 的最後增加了反正切函數電路,此電路包含了不隨影像大小改變的除法器,所以 複雜度為O(1),相位展開電路的計算內容為不斷與鄰近區域進行比較,做+2π的 運算,需要O(1)的加法器與乘法器複雜度,並且需要存放與影像長度相同的暫存 器數量,所以暫存器複雜度為O(N)。最後,記憶體作為各單元輸入與輸出的資料 儲存器,其必需存放N×N的影像資料,所以暫存器複雜度為O(N²),其餘皆為O。

	Table Generation Module	Evaluation Unit	NIOS II Processor
Adders	0	0(1)	0(1)
Multipliers	0(1)	0(1)	0(1)
Dividers	0(1)	0(1)	0(1)
Registers	O(N)	0(1)	0(1)

表 4.3 搜尋與評估系統中各單元資源複雜度比較

接下來以相同的分析方法討論搜尋與評估系統中搜尋單元及評估單元的硬 體資源需求複雜度,由於 3D 重建單元與搜尋單元中的焦距點生成模組在本論文 中使用 NIOS 處理器來實現,因此將另外分析 NIOS 處理器的資源需求複雜度, 整理出來的資源需求如表 4.3。我們可以發現在表格建置模組中使用了固定數量 的乘法器與加法器,但是在 Δ_f 參數和索引值的相關計算中也是先以N個 entries 的 表格儲存起來,因此暫存器的資源複雜度為O(N)。在評估單元架構中每個元素都 有使用到,但是的資源並不會隨著影像大小而增加,所以複雜度皆為O(1),NIOS II processor 只用做產生焦距點和整體自動對焦系統的控制器,因此複雜度也是 O(1)。我們將菲涅耳轉換以及搜尋與評估系統的電路資源複雜度進行整理後,可以得到整體系統的資源複雜度在乘法器、加法器、除法器方面皆為<math>O(1),而暫存 器的資源複雜度則為 $O(N^2 + N)$ 。

各單元的資源需求分析完成後我們即將討論實作此硬體架構實際所需要資源消耗,在本論文中是以256×256(也就是N = 256)尺寸的全像圖作為實作樣本。 在討論 Altera FPGA 開發板的資源消耗測量時,我們根據 Altera 官方的計算方式 使用三種單位作為評估資源消耗(area cost)的依據,分別是 Adaptive logic modules (ALMs)、Embedded memory bits 和 DSP block,其中 ALMs 被用來實現暫存器以 及運算單元;Embedded memory bits 主要用於實現暫存器或記憶體元件;而 DSP block 則被用來實現數值運算的元件如快速傳立葉轉換模組中的乘法器等。

41

圖 4.4 表示了菲涅耳轉換、相位展開電路中各單元的 ALMs 資源消耗表,大 部分的資源集中在菲涅耳轉換的三個單元,原因是前轉換與後轉換單元使用了由 ALMs 組成的 Lookup table,這些資源消耗也將隨著影像邊長N提高而增加,而快 速傳立葉轉換則是因為使用了大量的數學運算,然而這種消耗並不受影像大小的 影響。圖 4.5 為搜尋與評估系統中各單元的 ALMs 消耗,表格建置模組同樣使用 了 Lookup table,因此有較多的 ALMs 消耗。

Size	Pre-transform	FFT	Post-transform	Phase	On-Chip
	Unit	Unit	Unit	Unwrapping	RAM
<i>N</i> = 256	3402	11988	6772	281	456

表 4.4 菲涅耳轉換與相位展開電路各單元之 ALMs 資源消耗表

表 4.5 搜尋與評估系統各單元之 ALMs 資源消耗表

Size	Table Generation Module	Evaluation Unit	NIOS II Processor
<i>N</i> = 256	7943	1549	1307

表 4.6 及表 4.7 分別為菲涅耳轉換、相位展開電路、On-chip RAM 以及搜尋 與評估系統各單元的 Memory bits 消耗,從表中可以看出 On-chip RAM 使用了最 多的資源,這與 On-chip RAM 的組成有關,並且根據影像邊長提高, Memory bits 消耗越是可觀,相對於記憶體單元的資源消耗,其他單元的需求則低了需多。

Size	Pre-transform	FFT	Post-transform	Phase	On-Chip
	Unit	Unit	Unit	Unwrapping	RAM
N = 256	0	52228	9398	8675	4227136

表 4.6 菲涅耳轉換與相位展開電路各單元之 Memory bits 資源消耗表

表 4.7 搜尋與評估系統各單元之 Memory bits 資源消耗表

Size	Table Generation Module	Evaluation Unit	NIOS II Processor	
<i>N</i> = 256	66169	0	67776	

表 4.8 與表 4.9 分別為菲涅耳轉換、相位展開電路以及自動對焦系統各單元的 DSP blocks 消耗,在前轉換單元、快速傳立葉單元、後轉換單元與表格建置模組中使 用的複數乘法器成為各自資源消耗的主要來源,另外,後轉換單元與表格建置模 組還包含了反正切函數電路中的除法器,除法器的 DSP blocks 消耗更是佔了後轉 換單元整體消耗的一半,因此造成兩者的差異,相位展開、記憶體和評估單元中 由於未使用到複數乘法器或除法器的運算,幾乎不使用 DSP blocks。這些運算單 元不會隨著影像大小而改變,所以我們可以推斷若將維度增加,DSP blocks 的消 耗仍然會維持在一定值。

表 4.10 整理出了整體 3D 數位全像重建系統在本實驗裡用的 DE4 開發板中消 耗資源的情況,我們可以發現影像來源對於記憶體的龐大需求仍然是影響最大的 問題,在影像尺寸提高的硬體延伸發展下,將會消耗更多的 Memory bits。

43

Size	Pre-transform	FFT	Post-transform	Phase	On-Chip
	Unit	Unit	Unit	Unwrapping	RAM
<i>N</i> = 256	32	48	88	0	0

表 4.8 菲涅耳轉換與相位展開電路各單元之 DSP blocks 資源消耗表

表 4.9 自動對焦系統各單元之 DSP blocks 資源消耗表

Size	Table Generation Module	Evaluation Unit	NIOS II Processor
<i>N</i> = 256	108	4	4

表 4.10 整體電路在 DE4 開發板之資源校耗表

Size	ALMs	Block Memory Bits	DSP Block
$N = 2\Gamma C$	33698	4431382	284
N = 256	(37.0%)	(25.2%)	(22.0%)

表 4.11 為菲涅耳轉換電路、相位展開電路以及搜尋與評估系統中各單元的執 行時間分析,在本次實驗中皆以 500MHz 執行數位全像重建系統,由於表格建置 單元可以與菲涅耳轉換電路平行運作,且表格建置單元的執行時間為N + L4遠小 於菲涅耳電路中前轉換及後轉換的執行時間,所以在此將忽略其花費的時間。首 先觀察菲涅耳轉換中三個單元的執行時間,可以觀察出三個單元的時間十分相近, 並且從記憶體中讀取影像的時間佔了主要部分,這個特點符合第三章所整理的菲 涅耳轉換執行流程,各單元在讀取第一筆資料時便開始運算並且準備輸出資料, 執行時間相當於各自讀取影像花費的N²週期。在評估單元也同樣需要讀取N²的

44

影像資料,並從讀取第一筆資料後便開始運算,因此所花費時間與菲涅耳轉換各單元非常相近。相位展開的部分則花費相對大量的時間,其原因是相位展開過程 中需要不斷參考相鄰區域的資訊,重複疊代數次才能完整還原。

獲得了各單元的執行速度後,接著可以計算出整體數位全像重建系統的計算時間為 16.65ms。其中菲涅耳轉換電路的計算時間為 0.789ms,產出量(Throughput) 為 83.06 Mpixels/sec,而在 MATLAB 軟體執行情況下花費的計算時間為 0.2708(s), 產出量為 0.242Mpixels/sec,相較於軟體執行,本論文的菲涅耳轉換電路能更高速 的計算出結果。

Pre-transformFFTPost-transformPhaseEvaluationUnitUnitUnitUnwrappingUnit0.26210.26440.262215.60.261

表 4.11 自動對焦系統在頻率 500 MHz 下各單元之執行時間(ms)

接下來對於本論文提出的自動對焦系統進行效能及正確性分析,首先在與全 像圖相距 500mm 處對其進行繞射計算與相位展開,每次與全像圖拉近距離再次 計算,實驗中 step size 設定為 1mm, 共採樣了 500 個焦距點,分別對相同全像圖 進行重建,根據多次實驗統計,在實驗中相位展開疊代次數超過 300 回則判定其 無法完成展開,直接進行清晰度(Normalized Variance 值)評估。圖 4.2 是以 microlens 為樣本進行自動對焦的結果,橫軸為 1mm~500mm 的焦距,縱軸則為對 500 個焦 距點重建影像評估的數值,可以觀察到距離全像圖 50mm 處所重建的影像具有最 高的評估值。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 5mm 所重建的影像,圖 4.3(b)為焦距 500mm 所重建的影像,而圖 4.4 則為成功對焦的 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,可以發現焦距太近或太遠都會造成影像重建的失敗。



圖 4.2 Microlens 自動對焦結果



(a)



(b)

圖 4.3 Off-focus image (microlens): (a) 焦距 5 mm, (b) 焦距 500 mm.



圖 4.4 In-focus image (microlens) : (a) wrapped phase, (b) unwrapped phase.

本實驗的第二個樣本是多維的透鏡陣列(microlens array),由多個microlens整 齊排列而成的樣本,由於上個實驗的相位展開法對此圖形的計算無法達到收斂, 因此我們需要參考文獻[20]將每個pixel的鄰近參考區域擴大以正確展開相位,但 區域的擴大同時造成速度提升及資源需求增加。圖4.5為以microlens array為樣本 進行自動對焦的結果,橫軸同樣為1mm~500mm的重建焦距,縱軸則為對其500個 焦距點重建影像進行評估的數值,從圖中可以看到最理想的焦距也在50mm處, 圖4.6舉出兩個失焦重建影像的例子,圖4.6(a)為焦距5mm所重建的影像,圖4.6(b) 為焦距500mm所重建的影像。圖4.7則為成功對焦的重建影像,圖4.7(a)為後轉換 計算完成後的相位俯視圖,圖4.7(b)為相位展開之影像,我們可以觀察到失焦與對 焦成功的影像差異很大,在理想的焦距下便能夠完整重建全像圖資訊。這兩個樣 本的自動對焦結果顯示了本論文提出的架構能正確找到理想的重建焦距。



圖 4.5 Microlens array 自動對焦結果

49



(b)

圖 4.6 Off-focus image (microlens array): (a) 焦距 5 mm, (b) 焦距 500 mm.



(a)



(b)

圖 4.7 In-focus image (microlens array) : (a) wrapped phase, (b) unwrapped phase.

表 4.12 為硬體與 MATLAB 執行自動對焦的執行時間比較結果,為了與軟體 執行的環境設定一致,將電路的工作頻率提升至 3.4GHz,測量結果自動對焦系統 對於 microlens 樣本進行重建搜尋 500 個焦距點所花費的時間為 1.224(s),而 MATLAB 需要花費 92.632(s);若系統以 microlens array 為樣本,並搜尋 500 個焦 距點進行重建所花費的時間為 0.79(s),MATLAB 則花費了 99.498(s),在兩種樣 本的情況下速度皆有顯著的提升。

表 4.12 自動對焦系統在 3.4GHz 環境下搜尋 500 點之執行時間(s) Hardware in FPGA Software in MATLAB Microlens 1.224 92.632 Microlens array 0.790 99.498

Auchitesture	Power	
Arcmtecture	Consumption	
FFT Unit of	5 10	
Proposed Architecture	5.10	
[23]	270	

表 4.13 FFT 單元在頻率 500MHz 下之功率消耗(W)

最後我們將討論功率消耗的部分,由於可自動對焦的數位全像還原系統並不 多見,可互相比較的參考文獻也很有限,並且使用的平台與法則都不盡相同,因 此只討論快速傳立葉單元的功率消耗,在本論文中使用 Altera 公司提供的 PowerPlay Power Analyzer Tool[18]為測量工具,表 4.13 量測了 2D 快速傳立葉轉 換在工作頻率 500MHz 下的功率消耗,並與 GPU 為主的架構[23]進行比較,其架 構對 400MHz 到 700MHz 頻率進行 2D FFT 測量,然而功率量測結果都大於 220W, 當和本論文的架構同樣使用 500MHz 時功率更高達 270W,因此以 FPGA 實現 FFT 運算所消耗的功率相對小了很多,本論文所提出的系統架構在 FFT 單元具有明確 的低功率消耗,對於功率消耗需求嚴格的應用具有相當的優勢。

第五章 結論

本章節實現適用於嵌入式數位全像顯微鏡應用的 3D 影像重建系統,並且支 援自動焦距校正的功能,經過軟體模擬結果與硬體實際運算的結果相互驗證後確 認能完整還原。本論文提出的硬體架構對於可攜式的數位全像還原系統具有三個 優勢,分別為高速運算、低功率消耗以及可自動對焦,本架構的繞射計算使用浮 點數運算,有效提高精確度。各運算單元內部採用管線化設計提高運算速度,使 資料輸入的同時電路也能夠進行運算。所以本硬體架構相較於 GPU 為主的架構 而言,不但具有低功率消耗與可攜帶性的特點,在運算速度上也因為設計方法不 同而大幅提升了計算效能,此外本論文提出的架構比現存大部分的系統增加了自 動焦距校正的功能,對於即時運算的嵌入式數位全像顯微鏡系統之應用上,本論 文提出的系統架構是相當適合的。

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] L. Ahrenberg, A. J. Page, B. M. Hennelly, J. B. McDonald, and T. J. Naughton, Using Commodity Graphics Hardware for Real-Time Digital Hologram View-Reconstruction, IEEE Journal of Display Technology, Vol. 5, pp.111-119, 2009.
- [4] N. Pandey, D. P. Kellya, T. J. Naughtona and B. M. Hennellya, Speed up of Fresnel transforms for digital holography using precomputed chirp and GPU processing, Proc. SPIE, Vol. 7442, 2009.
- [5] Z. Zhu, M. Sun, H. Ding, S. Feng and S. Nie, Fast numerical reconstruction of digital holography based on graphic processing unit, Proc. IEEE Pacific Rim Conference on Lasers and Electro-Optics, 2009.
- [6] T. Shimobaba, Y. Sato, J. Miura, M. Takenouchi and T. Ito, Real-time digital holographic microscopy using the graphic processing unit, Opt. Express, Vol. 16, pp.11776-11781, 2008.
- [7] T. Nishitsuji, T. Shimobaba, T. Sakurai, N. Takada, N. Masuda, and T. Ito, Fast calculation of Fresnel diffraction calculation using AMD GPU and OpenCL, OSA Technical Digest, 2011.
- [8] M. Dogar, H. A. Ilhan, and M. Ozcan, Real-time, auto-focusing digital holographic microscope using graphics processors, Review of Scientific Instruments, Vol. 84, 2013.

- [9] A. C. Atoche, M. P. Cortes, J. V. Castillo, R. A. Ensenat, FFT Implementation for Electronic Holograms using Field Programmable Gate Array, Proceedings of the 6th International Caribbean Conference on Devices, Circuits and Systems, Mexico, 2006.
- [10] N. Masuda, T. Ito, K. Kayama, H. Kono, S. Satake, T. Kunugi and K. Sato, Special purpose computer for digital holographic particle tracking velocimetry, Opt. Express, Vol. 14, pp.587-592, 2006.
- [11] Y. Abe, N. Masuda, H. Wakabayashi, Y. Kazo, T. Ito, S. Satake, T. Kunugi, and K. Sato, Special purpose computer system for flow visualization using holography technology, Opt. Express, Vol. 16, pp.7686-7692, 2008.
- [12] R. Veitch, D. C. Hendry, and J. Watson, Reconfigurable Hardware Applied to Holographic Reconstruction, Proc. IEEE OCEANS, Aberdeen, Scotland, 2007.
- [13] T. Lenart, M. Gustafsson and V. Owall, A Hardware Acceleration Platform for Digital Holographic Imaging, Journal of Signal Process Syst., Vol. 52, pp.297-311, 2008.
- [14] 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.
- [15] 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.
- [16] 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.

- [17] Dennis C. Ghiglia, Gary A. Mastin, Louis A. Romero, Cellular-automata method for phase unwrapping, Sandia National Laboratories, Albuquerque, New Mexico 87185, 1984.
- [18] Altera Corporation. Quartus II Handbook Ver 13.0, 2013; Available online.
- [19] S. Rajan, S. Wang, R. Inkol, A. Joyal, "Efficient Approximations for the Arctangent Function," IEEE Signal Processing Magazine, vol. 23 pp. 108-111, 2006.
- [20] 陳焕元,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.
- [21] J. W. Goodman and R. W. Lawrence, Digital image formation from electronically detected holograms, Appl. Phys. Lett. vol. 11, pp. 77–79, 1967.
- [22] Altera Corporation, FFT MegaCore Function User Guide, 2011.
- [23] Y. Jiao, H. Lin, P. Balaji, W. Feng, Power and Performance Characterization of Computational Kernels on the GPU, Proc. IEEE/ACM Int'l Conference on Green Computing and Communications, pp.221-228, 2010.