國立交通大學

多媒體工程研究所

碩 士 論 文

利用類神經網路的手部影像掌心方向及關鍵點抽取系統設計

The Design of Palm Orientation and Keypoints Extraction System in Hand Images using Deep Learning CNN

研 究 生：廖哲珽

指導教授：蔡淳仁　教授

中 華 民 國　108 年　11　月

利用類神經網路的手部影像掌心方向及關鍵點抽取系統設計

The Design of Palm Orientation and Keypoints Extraction System in Hand Images using Deep Learning CNN

研 究 生：廖哲珽　　Student：Zhe-Ting Liao

指導教授：蔡淳仁　　Advisor：Chun-Jen Tsai

國 立 交 通 大 學

多 媒 體 工 程 研 究 所

碩 士 論 文

A Thesis

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 2019

Hsinchu, Taiwan, Republic of China

中華民國 108 年

摘要

近年來虛擬實境的應用日漸增加，因此人機介面溝通的相關技術開始受到大家的重視，手勢辨識也成為一項熱門的研究。

本論文以單攝影機第一人視角的頭戴式系統為使用平台，結合深度學習類神經網路與手部模型，建立一個具實務應用價值的手部姿勢估計系統，可以在任意動態影像中估算出手部姿勢的完整系統。在手部姿勢估計方面，我們利用神經網路估計 2D 手部關鍵點，再以一個 3D 手部模型來比對 2D 關鍵點估計結果作逆向運動學估算，藉此得到使用者手部的 3D 姿勢，包括掌心的方向角以及手指關節的位置。

在神經網路架構方面，本論文對於神經網路架構的選擇以及組合的效益，進行多項實驗與分析。我們評估了多種神經網路架構在不同資料集中的表現，藉此找到最適合的網路架構。在逆向運動學方面，我們利用神經網路推論手掌朝向的方向，並以此做為迭代估測法的初始值，改善收斂的狀況。並以大量手勢經 PCA 拆解後的樣本向量空間，作為迭代估測的解空間，使最終估測出來的手勢更加自然。

Abstract

In recent years, applications of Virtual Reality grow rapidly. Therefore, 3D hand gesture estimation that is crucial for human-computer interaction has become a hot research topic today.

This thesis aims to design a 3D hand pose estimation system for virtual reality applications using a single image as input. The proposed system uses deep learning neural networks for estimating 2D hand keypoints, and inverse-kinematic inferences of the 3D hand pose, including the orientation of the palm and the 3D angles of the finger joints, using a 3D hand model.

In terms of the neural network architecture, a series of experiments and analyses on various choices of neural network architecture and their efficiency to form an integrated system were conducted during design exploration. We evaluate the performance of different neural networks in multiple data sets, to find the best neural network architecture. For inverse kinematics estimation, we have improved the convergence of each iteration by using the orientation information of the palm, which was inferred by a neural network, as the initial values of iterative optimizations. Furthermore, we have used the Principal Component Analysis (PCA) technique to create a hand gesture vector space using a lot of real hand images. The PCA vector space is then used as the confined solution space of the iterative hand pose estimation process. As a result, the final estimated hand pose looks more natural.

誌謝

本篇論文的完成，首先要感謝我的指導老師蔡淳仁教授這兩年來的教導，在研究上遇到困難時，老師都能給予有用建議，並找出問題的關鍵點，使我的研究過程更加順利。此外老師也不吝於分享自己在工作上的各種經驗，來教導我應有的做事方法與態度，相信在未來這些經驗都會讓我受用無窮。

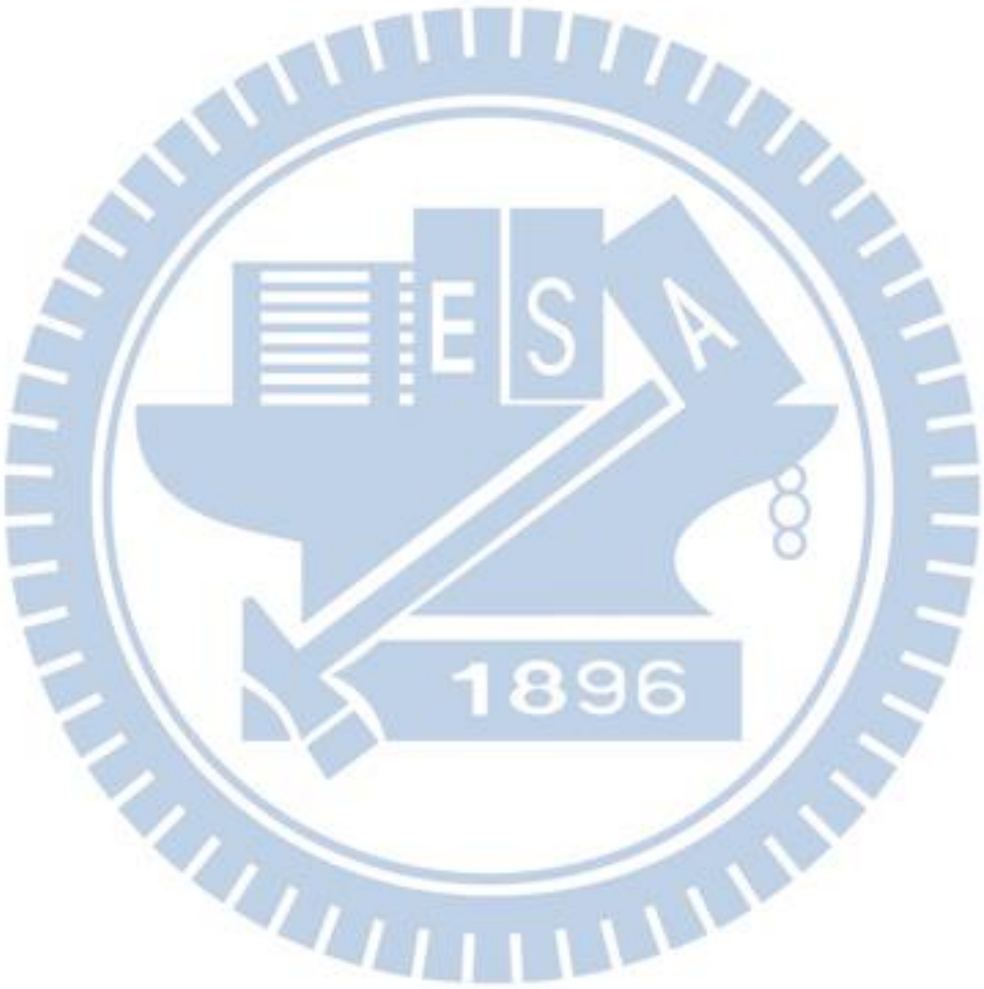
也感謝胡毓志教授與張添烜教授在繁忙之中願意撥空擔任我的口試委員，並對本論文提出許多意見以及改善方向，讓我知道一些研究上的盲點，以及許多能改進的地方。

接著也要感謝實驗室裡學長的幫助，感謝呂芳鎮學長在我進入實驗室後對我細心的指導，讓我能快速進入狀況並開始進行研究。也感謝實驗室的其他同學，在我遇到研究上的問題時，能和我討論並解決。同時也感謝李沿樞同學在亞利桑那的幫助，進行研究與書寫論文時有良好的壓力抒發管道。

最後要感謝家人願意支持我完成碩士學業，以及親朋好友的鼓勵，在我完成碩士學位的路上的陪伴，謝謝你們。

目錄

圖目錄



表目錄

第一章、前言

1.1 研究動機

近年來虛擬實境(VR)及擴增實境(AR)的技術蓬勃發展，人機互動的方式上也更加多元且人性化，與此相關之應用、遊戲與日俱增。然而若要能精確的操縱虛擬實境中的各種物件，仍然需要依靠手把、手套等相關設備，大大降低使用上的便利程度以及使用者體驗，因此如何不借助任何特殊設備，僅從單一攝影機所錄製的影像中，獲取手部的相關資訊(手勢、動作...等)，一直以來都是一個重要的課題。

過去我們實驗室已經開發出可達 96.18%正確率的即時動態背景手部語意切割系統[31]，以及在單色背景下可辨勢 36 種不同手勢、具有 96.71%正確率的 2D 手勢辨識系統[27]，結合兩個研究成果，已經可以藉由手勢達成基本的人機互動，然而對於在虛擬實境中移動或拿取物品等，需要較為精細的 3D 操作時，仍需從影像中標記出手指關節及手掌心方向等關鍵點。因此，本次研究主要目標為手部方向角度的估計及關鍵點的標記。除此之外，目前此類問題大多以卷積神經網路解決，由於神經網路是一種資料驅動(data-driven)的方法，需要大量資料進行訓練，但 3D 手部姿勢不容易產生標註好的練資料，通常需借助特殊裝置才能辦到。

因此本論文研究主要目的在於設計一個不須太過依賴大量具有 3D 關鍵點標註的資料，就可以訓練出的可靠手部姿勢估計系統，同時也將我們的系統與本實驗室之前的研究[31]進行整合。

1.2 研究成果及貢獻

本論文以一般攝影機做為使用平台，結合神經網路與手部模型，建立手部姿勢估計系統，以多任務學習的方式與手部語意切割系統做整合。並以藉由手掌作為初始值

與修改迭代演算法來改善逆向運動學的收斂狀況。圖 1 為本論文所提出系統的實際估測結果。整體來說，本論文的主要貢獻有二：

一、設計實作一個關鍵點估計網路，並藉由多任務學習同時提升關鍵點估計以及語意切割兩者的準確度。

二、提出以神經網路估計手掌方向來做為初始值，改善迭代的結果。

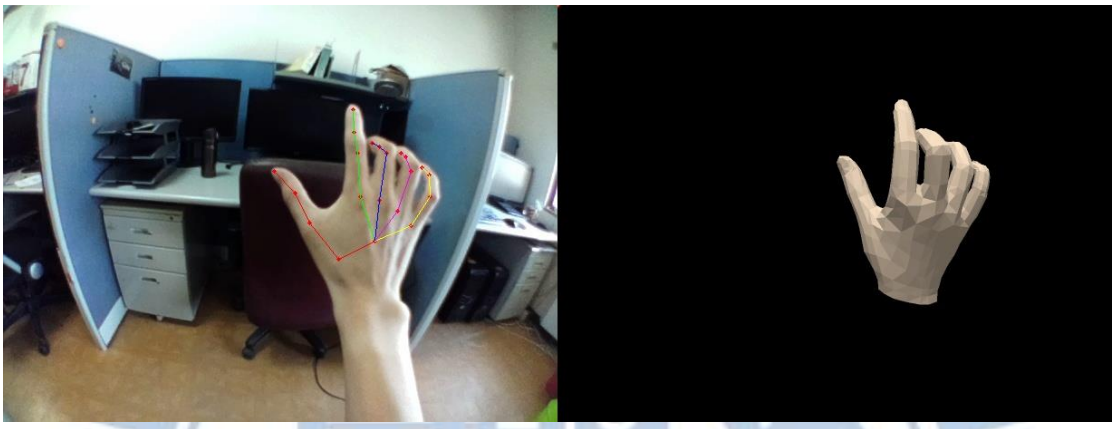


圖 1、實際估測結果。左邊為輸入影像、標上估測出來的手指關鍵點，右邊為根據估測出來的參數所繪製的 3D 手模型。

1.3 論文架構

本論文分成六章，第一章介紹動機與成果。第二章介紹手部姿勢與關鍵點抽取的相關研究。第三章介紹本論文所提出的系統架構所使用到的技術描述。第四章介紹本論文系統所針對的問題的明確定義以及如何修改、整合第三章所提到的技術成為最終的系統架構。第五章描述實驗方式以及結果。第六章則為論文的結論以及未來方向。

第二章、手部姿勢及關鍵點抽取相關研究

手部姿勢估計一直以來都是電腦視覺領域中重要的研究方向之一，目前在此類問題上，主流方法為藉由神經網路估計關鍵點位置，再以關鍵點資訊推論手部姿勢，因此本章主要介紹手部姿勢以及關鍵點估計的相關研究。而在關鍵點估計的部分，由於網路架構設計的相關研究大多都以人體關鍵點做為研究對象，且手部與人體的關鍵點估計並無太大差異，因此此處以多人體關鍵點估計的相關研究為介紹對象。此外，本論文最終是利用 3D 手部模型做為手部姿勢估測的主要限制條件 (Model-constrained estimation method)，因此本章最後會介紹手部模型的相關研究。

2.1 3D 手部姿勢估計

3D 手部姿勢估計方法主要可以分成三類，配戴感測手套的方法(Gloved-Based Methods)、以深度影像作為估測資料的方法(Depth-Based Methods)、和直接以視覺影像作為估測資料的方法(Vision-Based Method)。以下分別對三者做介紹。

2.1.1 Gloved-Based Methods

此方法以配戴具有感測元件[18]或是顏色經過特殊設計[19]的手套(如圖 2、圖 3)來獲取資訊，相較於其他方法，由於擁有較直接且有用的資訊，故能取得更加精確的結果，但相對的在使用上也有一定的不便，舉例來說手套可能會影像到手指關節的靈活度。

圖 2、具感測元件的手套，圖片出自[18]

圖 3、顏色為特殊設計的手套，圖片出自[19]

2.1.2 Depth-Based Methods

在過去基於深度影像進行手部姿勢估測方法中，大多數研究皆搭配手部模型進行，透過尋找手部模型參數，使手部模型的深度資訊與深度圖像相似，通常透過迭代最佳化演算法求解，如 PSO[20]。而近期的研究，則多以神經網路估計 3D 關鍵點位置[23][24]。Wan 等人提出讓卷積神經網路同時估計 2D、3D 關鍵點以及單位向量場(指向手指頭)，藉此提升精確度的方法[23]。Ge 等人則提出使用 3D 卷積神經網路來估計 3D 關鍵點位置的機制[24]。

2.1.3 Vision-Based Method

使用擬人視覺技術的方法是以一般光學影像資料作為輸入，只需以相機拍攝，相較於前兩者最為便利，但同時也最為困難。在近期神經網路相關技術及硬體發展逐漸成熟後，開始有大幅進展。

此類方法依據架構設計，可再細分為 one-stage[13]以及 two-stage[14][15] [16]兩

種。前者直接估計關鍵點 3D 位置，如 Spurr 等人藉由 VAE 模型學習潛在空間與不同資訊間的關係[13]。後者則是先估計 2D 手部關鍵點位置，再利用 2D 的資訊估計 3D 關鍵點的位置，Zimmermann 和 Brox 最早提出以神經網路估計單視角 RGB 圖片的手部關鍵點 3D 位置的論文[14]，藉由神經網路從 2D 關鍵點的 heatmaps 來估計 3D 關鍵點位置，圖 4 為該論文提出的系統架構圖，目前許多研究的架構設計皆從此論文修改。其中，Panteleris 等人利用手部模型對 2D 關鍵點做逆向運動學(inverse kinematics)得到 3D 關鍵點位置[15]。Cai 等人則是設計弱監督式學習(weakly supervised learning)的網路架構，只需要 3D 渲染的 3D 關鍵點位置以及真實影像的深度資訊就可以訓練網路，無需真實影像資料的 3D 關鍵點位置[16]。

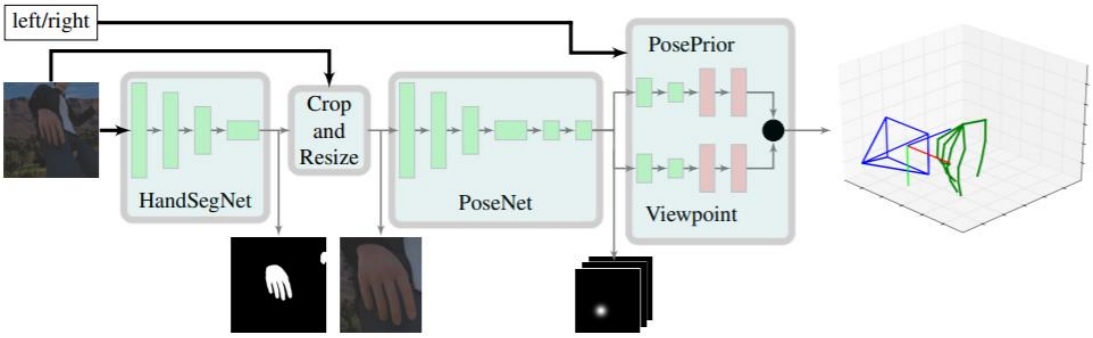


圖 4、Zimmermann 和 Brox 所提出的整體架構圖[14]

2.2 多人關鍵點估計

多人關鍵點估計主要有兩種方式，分別為 top-down[8][10]以及 bottom-up[9][11]。前者首先找出圖片中人物的位置並裁切，再做單人的關鍵點估計。後者則是先估計出圖片中所有關鍵點位置，再區別哪些關鍵點屬於同一個人。本論文採用 top-down 的方式，因此下面針對單人關鍵點估計的方法進行討論。

在過去此類問題大多先建構各個身體部位之間的關係，組成人體的 pictorial structures 模型(如圖 5 所示)，去尋找圖片中與人相似的形狀[1][2][3][4]，然而這種方法較無法處理被遮擋的部位，而神經網路可大幅改善此狀況[5]。

有別於過去直接估計關鍵點在圖片上的位置[6]，近期的研究則改為估計各個關鍵

點的 heatmaps[5][7][8]。神經網路架構的部分，Wei 等人建立一個多階段估計的網路架構，以利學習關鍵點之間的關係[5]。Newell 等人則在以 encoder-decoder 架構作為多階段估計中的組件，有效的融合不同尺度的特徵[7]。圖 6 為他們所提出的網路架構圖。在近期研究中多以這個架構為基礎做改良，如 Li 等人所提出的跨階段特徵結合(cross stage feature aggregation)的方法[8]，可以改善 Netwell 等人的架構中，多階段估計特徵丟失的問題。

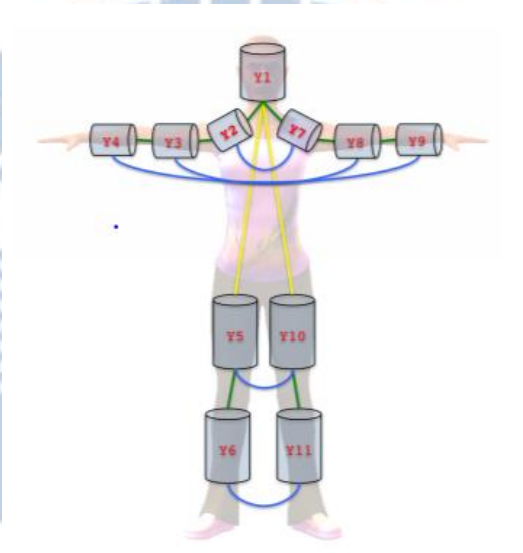


圖 5、pictorial structures 模型示意圖，圖片出自[3]

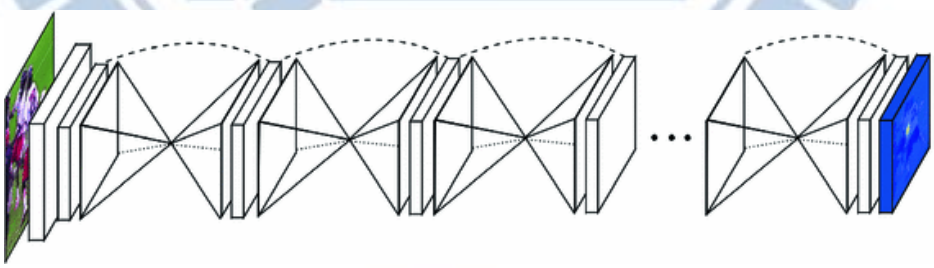


圖 6、多階段 encoder-decoder 架構示意圖，圖片出自[7]

2.3 手部模型

本節討論手部模型的選擇，包括了 Hsu [28]、Libhand [29]、和 MANO[30]等三種手部模型介紹。這三者各有其優點，Hsu 利用 Blender 所設計的模型，在手部關節轉動上符合真實人類的手部狀況，而 Libhand 以真實手部影像做渲染，視覺上較接近真實影像，MANO 則是有形狀參數，可以改變手部關節比例。考量未來的發展性，本論文最終選擇以 MANO 作為手部模型，並結合 Hsu 的手部模型中靜態關節角度限制。本論文使用 Hasson 等人所提供的 MANO 手部模型的 PyTorch 版本[35]，渲染的軟體則是使用 Kato 等人所開發的可微分渲染程式庫[36]。

2.3.1 Hsu 的 Blender 手部模型

Hsu [28]的手部模型為本實驗室研究生過去以 Blender 所製作的手部模型，擁有 26 個自由度，分別為每根手指的 ADB、MP、PIP、DIP 的關節角度，以及控制掌心位置及方向的 6 維的全域自由度，同時此手部模型考量了靜態、動態的關節角度限制，因此擺出的姿勢，相當接近真實人類的狀況。表一為關節的角度限制，圖 7 為 Blender 手部模型圖。

表 1、靜態關節限制，表格出自[28]

	Thumb	Index	Middle	Ring	Little
ADB	-40~40	-30~30	-30~30	-30~30	-30~30
MP	-51.7~0	-66.3~0	-77.3~0	-88.4~0	-105.8~0
PIP	-51.7~0	-95.7~0	-109.2~0	-114~0	-102.9~0
DIP	-66.3~0	-70.6~0	-89.7~0	-76.4~0	-90~0

圖 7、blender 手部模型，圖片出自[28]

2.3.2 Libhand

Libhand 是以 OGRE 為基礎所製作的三維手部模型[29]，用於渲染的皮膚材質是由真實手部影像掃描製作，在視覺上與真實圖像較為相似。Libhand 目的在於提供直觀且容易的介面，讓使用者能輕鬆的操縱手部模型取得各種手部姿勢的圖片，便於研究與分析。圖 8 為 Libhand 模型圖。

圖 8、Libhand 手部模型[29]

2.3.3 MANO hand

MANO 手模型是在 2017 年提出的開源碼手部模型[30]。以姿勢(pose)、形狀(shape)兩種參數類型來描述模型狀態，由姿勢參數控制關節的轉動，形狀參數則影響

骨架比例與手的形狀。

MANO 不同於上面兩個手部模型使用 LBS(Linear Blend Skinning)來描述骨架與網格之間的關係，而是從手的三維掃描(3D scan)中學習姿勢、形狀兩種參數和骨架、網格之間的關係，為此 Romero 等人提出了一個包含 51 種手勢的三維掃描資料集。圖 9 為 MANO 模型圖。這是本論文所採用的 3D 手部模型。

圖 9、MANO 手部模型，圖片出自[30]

第三章、本論文所採用的技術介紹

本論文所提出的手部姿勢估測系統，是參考了許多現有已發表的技術，進行實驗分析後，抽取出各技術的長處以組合出最後的系統。本章針對我們在設計實作系統架構的過程中所實際採用到的技術進行討論。至於我們提出的系統架構細節會在第四章描述。

3.1 Multi-Stage Networks for Human Pose Estimation

Multi-Stage Pose Network (MSPN) [8]為 Microsoft COCO 2018 [17]關鍵點估計競賽的冠軍。對於分類問題，神經網路越深效果應該越好，因此多階段估計網路架構表現應優於單階段估計，然而當時在 MS COCO 資料集上，卻是單階段估計架構表現較突出。因此 Li 等人以當時表現較優的 Netwell 等人[7]多階段網路架構為例，提出三個多階段估計網路架構的問題，並加以改進，實驗結果請參考圖 11。

首先，Netwell 等人提出的用來組成多階段網路架構的 encoder-decoder 模組設計並不好，在降採樣(down sampling)時，特徵的通道(channels)數量仍然保持相同，導致大量特徵資訊流失。

第二，由於重複降低採樣率(down sampling)及提升採樣率(up sampling)的過程，無法保證能保留前面階段有用特徵，換句話說網路模型將不易最佳化。因此 Li 等人提出跨階段特徵結合(cross-stage feature aggregation)，將降低採樣率所得到的特徵與前一階段的特徵相加，以保留有用的特徵。

第三，在多階段估計中，在越後面的階段中，神經網路所估計之關鍵點位置會越精確，因此在訓練上採用傳統方法上常用的 coarse-to-fine 技巧，不同於[7]在所有階段皆以相同目標做訓練，而是在越後面的階段使用越精確的 heatmaps 作為訓練目標。

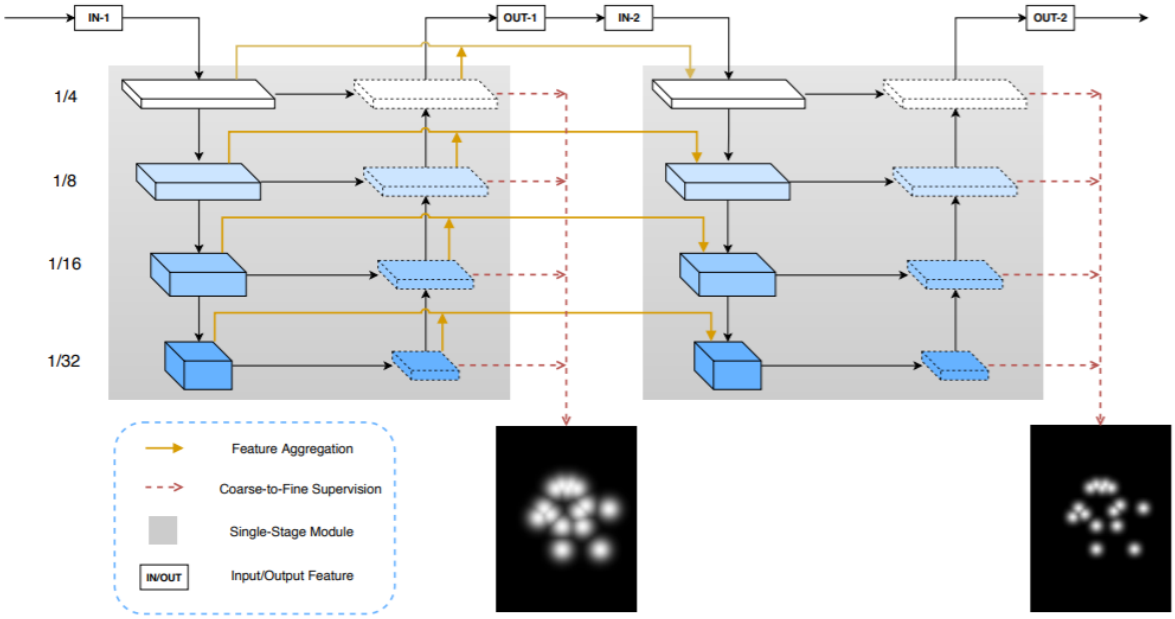


圖 10、MSPN[8]網路架構圖

Components			Hourglass	MSPN
BaseNet	CTF	CSFA		
√			71.3	73.3
√	√		72.5	74.2
√	√	√	73.0	74.5

圖 11、[8]所提出的三個改善方法在 MS COCO 資料集上的結果

3.2 Single RGB Frame 3D Hand Pose Estimation [15]

這項技術是在 2018 年由 Pantelerish 等人所提出的論文[15]，提出一個 3D 手部關鍵點估計的方法，分為三個步驟，首先藉由物體偵測(object detection)框出圖片中左手、右手的位置並將其裁切下來，再對裁切下來的圖片進行手部關鍵點估計，得到 2D 關鍵點位置，最後將左手圖片翻轉成為右手，搭配 3D 手部模型，以逆向運動學找出對應的 3D 手模型參數，以此求出 3D 關鍵點位置，圖 12 為發表在[15]的系統架構圖。論文中物體偵測和關鍵點估計直接使用當時的 state-of-the-art 模型。下面介紹逆向運動學以及 Levenberg-Marquardt 最佳化演算法。

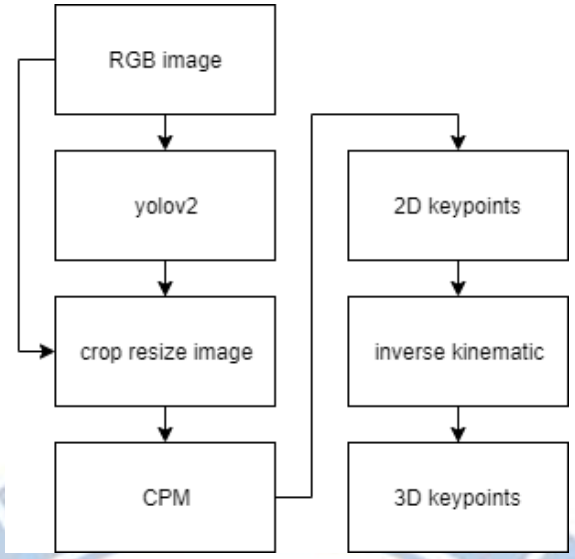


圖 12、[15]所提出的系統架構圖

3.2.1 逆向運動學(inverse kinematics)

在 3D 模型中我們有兩種空間來描述模型當前的狀態，分別為關節空間(joint space)以及直角坐標空間(Cartesian space)，前者記錄關節的旋轉角度，後者則是記錄關節的位置。當由 3D 模型的關節空間參數計算出對應的直角座標位置時稱為正向運動學，反之由以直角坐標空間位置計算出對應關間空間參數時稱為逆向運動學。而在逆向運動學中，因為通常不會有解析解，因此大多用非線性的最佳化演算法來求解。

論文中以神經網路估計 2D 關鍵點位置做為目標，因此可將最小化目標寫為 **Eq. (1)**

$$\sum_{i=0}^{21}(p_i^3(x_i-u_i))^2+(p_i^3(y_i-v_i))^2 \quad \textbf{(1)}$$

其中(x_i,y_i)為神經網路估計之 2D 關鍵點位置，(u_i,v_i)則為手部模型映射在圖片中的位置，p_i則是神經網路所估計該點是關鍵點的機率。

3.2.2 Levenberg-Marquardt(LM)演算法

LM 演算法是一個非線性最佳化演算法，結合了梯度下降法和牛頓法的優點，改善

梯度下降法的收斂問題，並解決牛頓法中海森矩陣可能不可逆的狀況。下面說明演算法迭代方式。

考慮函數$\boldsymbol{f}: \boldsymbol{R}^n \to \boldsymbol{R}^m$，若想對 **Eq. (2)**進行最佳化則可依 **Eq. (3)**進行迭代

$$S(x) = \frac{1}{2}\sum_{i=1}^{m}(f_i(x))^2 \quad \textbf{(2)}$$

$$x_{i+1} = x_i - \left(J^T J + \lambda\, diag(J^T J)\right)^{-1}\nabla f(x_i) \quad \textbf{(3)}$$

其中J為 Jacobian matrix，x_i為第 i 次迭代結果。

3.3 Yolo v3[34]

Yolo 為 2018 年提出的物體偵測神經網路架構，當時在[17][40]兩個資料集皆為 state-of-the-art，同時能達到實時(real-time)偵測。本篇為 yolo 系列的第三代，方法與前一代相似，主要以兩種方式改良網路架構，首先，它使用了 resnet[32]中的殘差網路架構設計更深的網路，其次，它參考了 FPN 的架構設計，加入多尺度預測以及多尺度特徵融合。

在物體偵測的方法上，yolo 採用 one-stage 的方式，直接從圖片中偵測定界框(bounding box)，其方式如下，首先將圖片等分成S × S個區塊(如圖 13 所示)，而每個區塊會有 N 個事先設計好的 anchor box，而神經網路則負責預測每個 anchor box 的長寬變化、中心位置位移以及所屬類別。網路估計結果與定界框關係可參考 **Eq. (4)**到 **Eq. (7)**，

$$b_x = \sigma(t_x) + c_x \quad \textbf{(4)}$$

$$b_y = \sigma(t_y) + c_y \quad \textbf{(5)}$$

$$b_w = p_w e^{t_w} \quad \textbf{(6)}$$

$$b_h = p_h e^{t_h} \quad \textbf{(7)}$$

其中t_x, t_y, t_w, t_h為網路估計的結果，(b_x, b_y)為定界框中心位置，(b_w, b_h)為定界框的寬與長，(p_h, p_w)為 anchor box 的寬與長，(c_x, y_y)為 anchor box 所在區域左上角的位

置，$\sigma()$則代表羅吉斯函數。

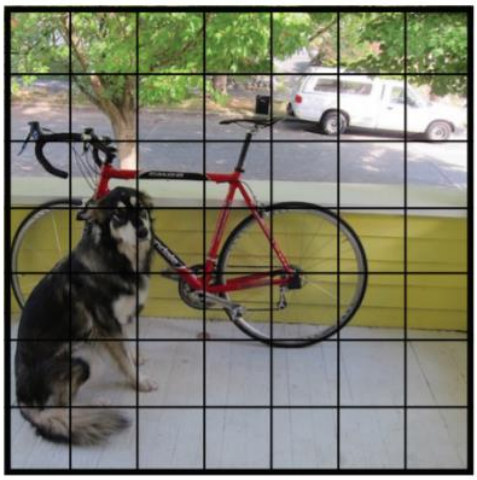


圖 13、圖片等分S × S個區塊示意圖，圖片出自[42]

第四章、問題描述及系統架構

4.1 問題描述

本論文研究目標為以單一相機做為使用平台的手部姿勢(hand pose)估計，本論文系統以一般低價網路攝影機拍攝的彩色影像作為輸入，主要輸出為相對於手掌心的 3D 關鍵點位置(也就是 3D 坐標的原點是定位於掌心，而坐標的 X, Y, Z 方向在第 4.3 節會再說明)、以及 2D 手部關鍵點位置。此處 2D 關鍵點是指手部關鍵點在影像中的位置，因為我們假設了一個 3D 的手部模型，即使只用單一攝影機以及 2D 的關鍵點也可大約推算出 3D 的手部關鍵點坐標。手部關鍵點指的是手掌的中心、手指關節、手指指尖共 21 點，請參考圖 14。而手部 3D 關鍵點通常被稱為手部姿勢。

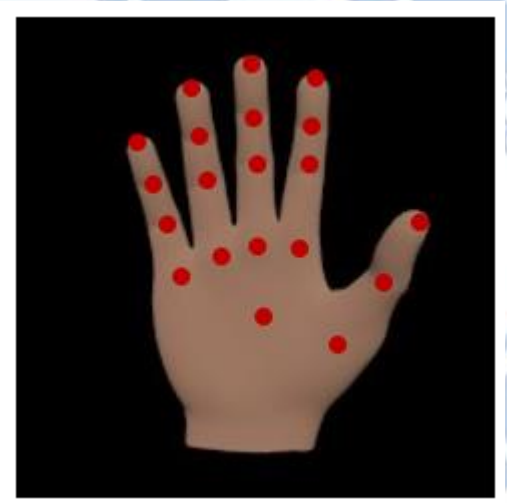


圖 14、手部關鍵點示意圖，紅點為手部關鍵點位置

本論文最終選擇以 Panteleris 等人所設計的系統作為基礎的系統架構[15]，在此架構上最需要改進的地方有兩個，第一是手部關鍵點位置的正確性，第二是逆向運動學中最佳化演算法的收斂結果正確度。本論文主要針對此二問題做改進。

4.2 系統架構

本論文所提出的系統架構如圖 15 所示，是以在 3.2 節所介紹的系統(圖 12)[15]為基礎進行修改。在網路架構的部分，我們最終選擇 MSPN[8]、yolov3[34]所採用的架構，同時我們將系統中 2D 關鍵點估計的神經網路，與 deeplabv3+[31]進行整合，設計一個多任務學習(multi-task learning)的網路架構。逆向運動學的部分，本論文以 Levenberg-Marquardt 演算法最佳化，並以 ResNet[32]預測手掌 orientation 作為迭代的初始值，以及[33]中的方法修改最佳化方式，改善迭代的收斂狀況。以下介紹本論文對於 Panteleris 等人所提出的架構[15]所修改的部分。

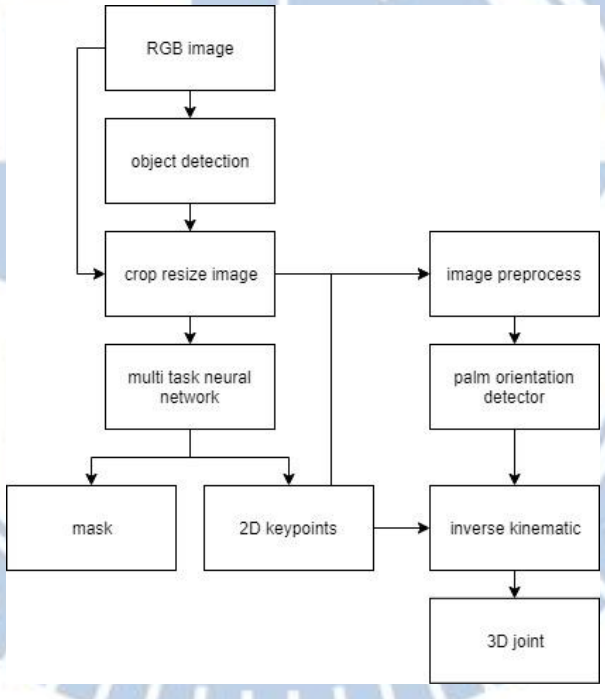


圖 15、本論文所提出的系統架構圖

4.3 迭代初始值

過去的研究皆顯示良好的初始值對於非線性最佳化有巨大的影響([21][26])，然而 Xu 和 Cheng[21]或 Ye 等人[26]所提出的技術是基於深度資訊的方法，無法直接使用在我們的研究中。根據前人研究所獲得的經驗[21][44][28]，我們可以知道，手掌是否能

找到是結果好壞的關鍵。因此我們以估計手掌朝向的方向，作為初始值。近期較為相關的研究把估計人體的方向視為一種分類問題[22][43]，，如圖 16 所示，將水平 360 度分成八個方向，並用神經網路進行分類，來達到估計人體方向的目的。

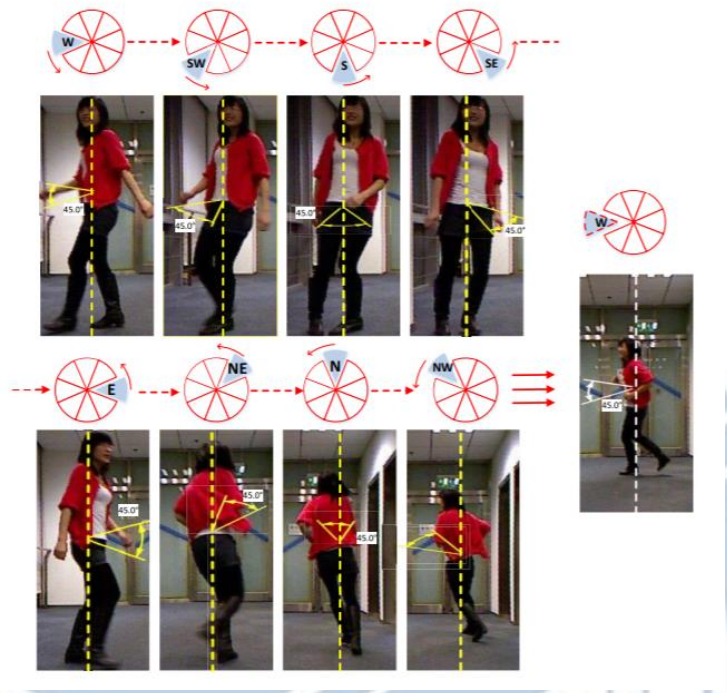


圖 16、人體朝向分類示意圖，圖片出自[43]

本論文的問題與 Choi 等人[22]或 Liu 等人[43]所提出的情況略有不同。在我們的使用情境下，手掌朝向可能是任意方向，因此本論文必需使用不同的分類方式。我們採用了兩種不同的方法，一是把所有 3D 方向依歐拉角分成 256 個不同的方向(x, z 軸各分成 8 類角度，y 軸分成 4 類角度)。另一種方法則是假設掌心到中指的 MP 關節的直線在圖片中為垂直線，以 y 軸為選轉軸的 360 度分 8 類，以 x 軸為旋轉軸的 180 度分 4 類，共 32 類。經實驗證實，以這種假設做分類相較於不做假設分成 256 類的好處有二，第一是容易製造訓練資料集，第二是圖片經過預處理後神經網路訓練結果也較好，詳細實驗請參考 5.7 節。文中坐標軸方向請參考圖 17。

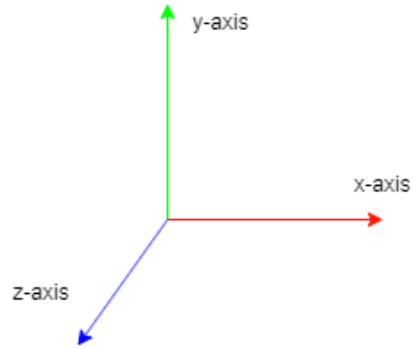


圖 17、角度分類法的坐標軸示意圖

4.4 Hierarchical Optimization

Hierarchical optimization 的方法是由 Schröder 等人所提出的最佳化方法[33]，主要目的為改善逆向運動學中最佳化演算的收斂性，可以減少不自然姿勢以及收斂至局部最小值的情況。

在實際情況下，手部不同關節的靈活度並不相同，然而在一般的最佳化演算法中是直接對所有的維度一起求解，並且所有的維度都是一樣的權重，所以較容易有不自然的手部姿勢出現。因此論文提出 coarse-to-fine 的多階段最佳化方式，並不直接使用全部的維度進行迭代，而是先在較低的維度中求解，再依序增加維度大小，讓演算法先在較重要的空間中找解，再逐漸微調。而最佳化求解的空間則是以 PCA 對手勢資料集的關節角度計算來取得，得到的 PCA 空間能表達各個關節之間的聯繫以及各個關節的靈活度。演算法 pseudo code 可參考圖 18。

Algorithm 1: Hierarchical Optimization

Input : target $T \in \mathbb{R}^m$, pca space basis $\{P_i\}_{i=1}^{n}$
Output : result $x \in \mathbb{R}^n$
Parameter: number of stage S
set initial x_0;
for $i \leftarrow 1$ **to** S **do**
  $k = round(i\frac{n}{S})$;
  $P^s = span(\{P_j\}_{j=1}^{k})$;
  $x_i = \arg\min_{x \in P^s} f(x, T)$ with initial x_{i-1} ▷ $f : \mathbb{R}^n \to \mathbb{R}$ be a loss function
end

圖 18、Hierarchical optimization 演算法 pseudo code

4.5 多任務學習

過去多項研究[10][38][39]皆表明，若同時學習多個相關性高的任務，對不同任務皆有正面的影響，Gkioxari 和 Girshick[10]提出以物體偵測的網路架構為基礎進行修改，同時進行物體偵測、切割、和關鍵點估計三種任務，圖 19 為實驗結果，可以看出預測切割可以提升關鍵點估計的效果。而 Popa 等人[38]或 Wang 等人[39]皆以關鍵點估計網路架構為基礎做修改，多任務的部分，Popa 等人採用的方法同時進行關鍵點估計、語意切割和 3D 重建，以提升 3D 重建的結果，而 Wang 等人所提出的技術則是同時進行關鍵點估計和語意切割兩種任務。在網路架構上，前面提到的三種系統都是採用 encoder-decoder 架構，對於不同的任務使用同一個 encoder 獲取特徵，再分別以不同的 decoder 估計結果。在網路訓練上，則是將不同任務的損失函數依照權重相加，成為一個新的損失函數，來進行訓練，Eq. (8)為新的損失函數方程式。

$$L(x,T)=\sum_{i=1}^{n} w_i L_i(x_i,T_i) \tag{8}$$

其中L為新損失函數，L_i為第i個任務的損失函數，x_i為第i個任務的輸出，T_i為第i個任務的 ground truth。

	AP^{bb}_{person}	AP^{mask}_{person}	AP^{kp}
Faster R-CNN	52.5	-	-
Mask R-CNN, mask-only	**53.6**	**45.8**	-
Mask R-CNN, keypoint-only	50.7	-	64.2
Mask R-CNN, keypoint & mask	52.0	45.1	**64.7**

圖 19、[10]的多任務學習實驗結果

如上所述，語意切割與關鍵點估計的多任務學習可以提升關鍵點估計的結果，因此我們嘗試將 Wang 等人提出的 MSPN 架構[8]與語意切割系統 deeplabv3+的網路架構[37]進行合併。讓兩者共用同一個 encoder，並將語意切割系統的結果與 encoder 結果輸入到 MSPN 的 decoder 中。

圖 20 為示意圖，虛點紅色框為 MSPN 的網路架構，虛線藍色框則為 deeplabv3+的

網路架構。

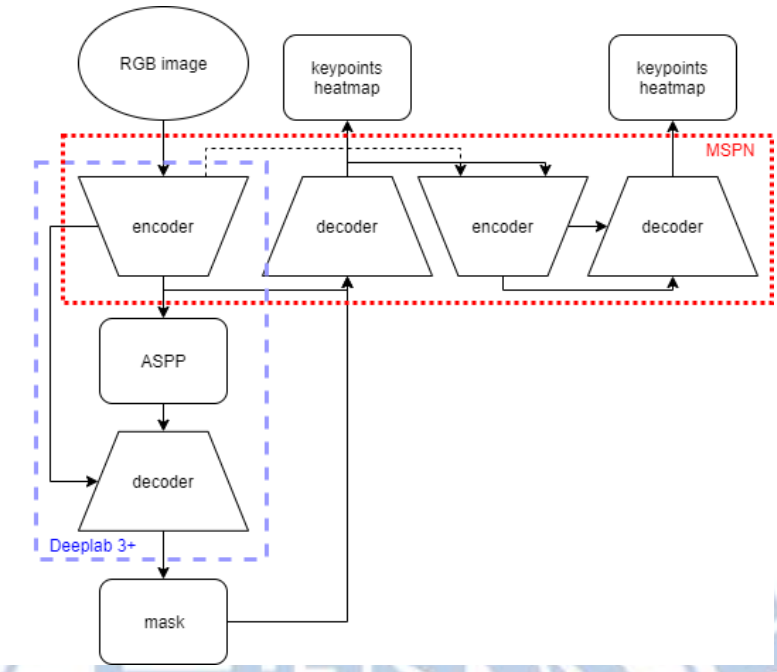


圖 20、MSPN 與 Deeplab V3+合併示意圖

第五章、實驗過程與結果

5.1 實驗環境

硬體部分：

- 中央處理器：Intel Core i7-8700k @ 3.20GHz
- 記憶體大小：Transcend 8GB DDR3 1600 兩條，共 16GB
- 顯示卡：NVIDIA GeForce GTX 1060 with 6GB Memory
- 硬碟大小：WD Blue 1TB

軟體部分：

- 作業系統：ubuntu 16.04 LTS 64-bit
- OpenCV 版本：3.4.3
- CUDA 版本：9.2
- cuDNN 版本：7.1.2
- Python 版本：3.5.2
- PyTorch 版本：0.4.1

5.2 效果評估方式

2D、3D 關鍵點的部分，我們用以下三種方法作為衡量效果的標準，mean EPE(endpoint error):估計結果與正解之間的歐式距離的平均，PCK(percentage of correct keypoints):估計結果 EPE 低於某個閥值的資料在資料集中的占比，AUC(area under the curve):用不同閥值的 PCK 所畫成的曲線下的面積。此外由於我們是以單張圖片做估計，因此我們不考慮尺度(scale)以及平移(translation)的差異，也就是說我們會將估計結果按比例縮放、平移。而語意分割則是使用 mIOU(mean intersection over union)當作評估的標準。手掌朝向分類的部分，我們以正確率以及角度誤差來評估結果優劣，正確率代表分類正確的百分比，而角度誤差是預測手掌朝向方向的結果與真正的結果之間的最短的旋轉角度。

5.3 資料集介紹

5.3.1 Rendered Handpose Dataset [14]

此資料集是由 3D 模型渲染的人造資料集，共包含 41258 張訓練資料以及 2728 張測試資料，並且提供手和人的分割、手部關鍵點以及場景深度三種不同的標籤。之後以 RHD 稱呼此資料集。圖 21 為 RHD 資料的示意圖。

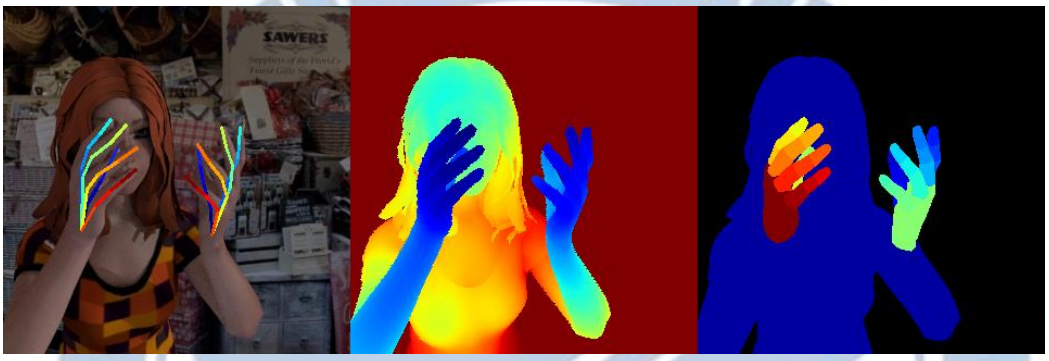


圖 21、RHD[14]資料示意圖，由左到右分別為關鍵點、深度、切割

5.3.2 Stereo Tracking Benchmark Dataset [25]

該資料集為真實影像資料，在六個不同背景的六段影片，每段影片 3000 張圖片，共 18000 張圖片，我們以其中五段影片作為訓練資料，剩下一段做為測試資料。之後以 STB 稱呼此資料集。圖 22 為 STB 資料示意圖。

圖 22、STB[25]資料示意圖

5.4 實驗相關參數說明

本節列出神經網路訓練時所用參數，表 2 為詳細的參數設定，包含學習率、批次訓練數量、Epoch 等，其中學習率在每訓練 15 個 Epoch 後會更新一次，更新方式為當前學習率乘上 Gamma。

同時為提升網路泛化能力，我們使用資料擴增(Data Augmentation)技術，隨機將圖片旋轉-30~30 度並隨機調整圖片的大小，介於原本大小的 0.5 倍至 1 倍，圖片縮小後在周圍做 zero padding，使其與原圖大小相同。為了避免過擬合(overfitting)，在訓練時我們將 2D 關鍵點位置隨機加上雜訊，此雜訊為平均值為 0、標準差為 1.5 的高斯分布。

此外由於 STB 資料集為六段影像資料，每段影像中背景以及手部位置都無太大變化，所以資料集中只有六種背景，因此在實驗以 STB 資料集作為訓練資料時，為避免過擬合等問題，我們會先在 RHD 資料集上訓練 10 個 Epoch 作為預訓練。

本論文於 5.5 與 5.6 小節中關鍵點估計相關實驗皆以此設定做訓練，之後不特別進行說明。

表 2、實驗相關參數

Epoch	20
Batch size	8
Learning rate	1e-4
Gamma	0.1
Resize	256
Data argumentation	RandomRotate:-30~30 RandomResize : 0.5~1.0 Keypoints noise : N(0, 1.5)

5.5 模型篩選

模型篩選的部分，我們比較了三種 CPM[5]、HG[7]、MSPN[8]不同的網路架構，以 RHD 和 STB 此二公開資料集做測試，從中選出適合的網路架構，作為後續研究的基礎。從表 3、圖 23 的實驗結果可以看出 MSPN 的效果較好，故後續實驗主要以 MSPN 進行。

表 3、CPM、HG、MSPN 在 RHD、STB 資料集的結果

Mean EPE	RHD	STB
CPM	6.057	7.316
HG	5.207	6.773
MSPN	4.574	6.491

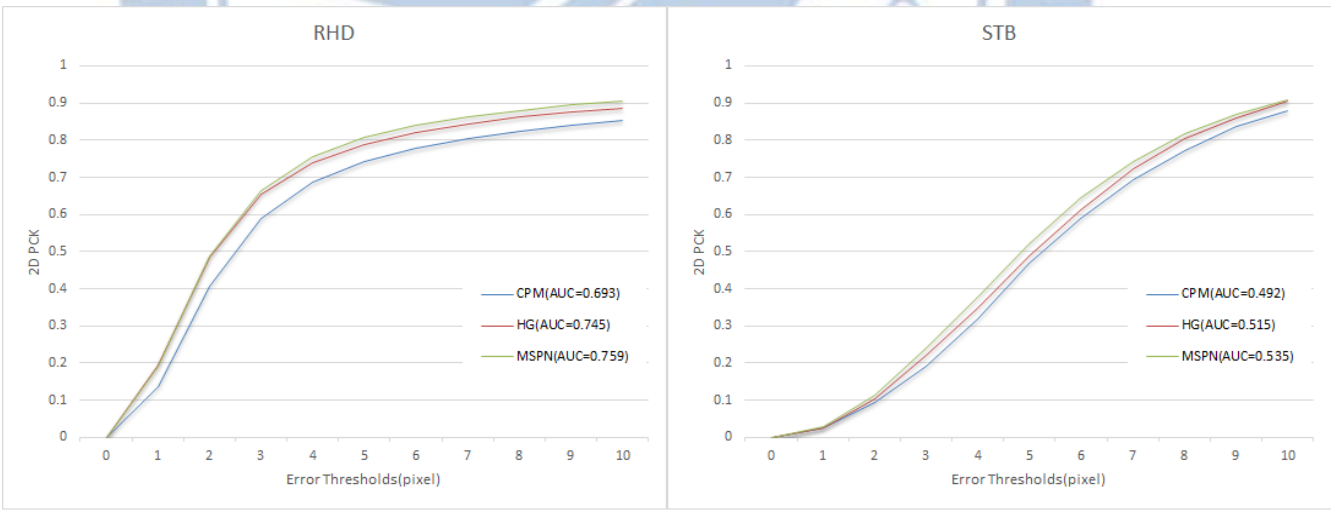


圖 23、CPM、HG、MSPN 在 RHD、STB 結果圖表

5.6 多任務學習

本節整理多任務學習結果，並與其他論文做比較。詳細多任務學習的細節請參考

4.6 節的介紹。

我們將本論文的結果與[38][39]比較，在此兩篇論文中皆是以 CPM 的網路架構為基礎做修改，因此我們在 CPM、HG、MSPN 三種網路架構上都進行實驗，表 4 為比較有無多任務學習的實驗結果，有用多任務學習時在關鍵點以及語意切割上表現皆優於不使用，表 5 則是本論文結果與[38][39]比較，可以看出不論在關鍵點或是語意切割上，本論文使用方法表現皆優於[38][39]。

表 4、有無多任務學習實驗結果比較

Mean EPE/ mIOU	Without multi-task learning		With multi-task learning	
	Keypoints	Mask	Keypoints	Mask
CPM	6.057	85.77%	5.857	88.62%
HG	5.207	86.01%	5.101	88.65%
MSPN	4.574	86.95%	4.410	88.87%

表 5、本論文與[38][39]在 RHD 上關鍵點與語意切割結果

	[38]	[39]	Proposed
Keypoints (Mean EPE)	7.468	5.908	5.857
Mask (mIOU)	87.25%	86.85%	88.62%

5.7 3D 關節點估計

本節整理本論文對 2D 關鍵點做逆向運動學的相關實驗結果，相關技術請參考 4.2 節。實驗以 RHD、STB 做為測試資料集，我們依序進行了兩組實驗，首先為測試 LM 演算法在此問題上的效果，所以我們排除會影響收斂的其他因素，因此用正確的 2D 關鍵點位置做為目標，並將手部模型的關節間長度比例調整至正確的比例，觀察迭代的收斂狀況及結果。再來我們測試演算法是否 robust，我們改為使用網路估計的 2D 關鍵

點位置做為目標，以 MANO 作為手部模型，觀察演算法結果。在本節實驗中我們專注於 3D 關鍵點估計上，因此我們假設圖片中的手已被物體偵測演算法正確框出並分類為左、右手。在目標函數上，我們以 Panteleris 等人所提出目標函數[15]為基礎進行修改，將神經網路所估計的機率p_i由六次方改為一次方，**Eq. (9)**為我們的目標方程式。下面分別說明兩個實驗的結果。

$$\sum_{i=1}^{21} p_i((x_i - u_i)^2 + (y_i - v_i)^2) \quad \textbf{(9)}$$

5.7.1 以正確 2D 關鍵點為輸入

當我們以正確的 2D 關鍵點位置作為逆向運動學的目標時，我們發現迭代的收斂狀況並不理想，容易陷入局部最小值，此類問題常見的解決方向有二，分別為迭代初始值以及最佳化方式。下面我們分別從這兩個方向做改善。

首先，如同 4.3 節所述，我們將估計手掌朝向視為分類問題，我們嘗試兩種分類方式，第一種以歐拉角表示手部方向(orientation)，共有三個維度，其中兩個維度的角度範圍為 0~2π，其中一個維度為 0~π，每$\frac{\pi}{4}$分成一類，共分成 256 類。第二種，我們先將圖片進行預處理，旋轉圖片使手掌到中指垂直，同樣以歐拉角表示方向，共有兩個維度，其中一個維度角度範圍為 0~2π，另一個維度為 0~π，共分 32 類。由表 6 的實驗結果可看出第二種方法較好，因此我們採用第二種分類方式。

表 6、兩種分類方式結果比較

正確率/旋轉誤差角度	RHD		STB	
256 類	30.1%	33.19 º	32.5%	37.69 º
32 類	66.5%	18.92 º	64.5%	26.36 º

接下來我們先我們對 Romero 等人所提出的資料集[30]做 PCA，得到 PC-space，在使用 Hierarchical Optimization 演算法[33]進行實驗，我們測試分成不同階段數量的結

果，實驗結果可見圖 24，可以看出在分成四個或以上的階段數量時，效果大致相同，因此我們最終分成四個階段來執行此演算法。圖 25 為本節實驗解果

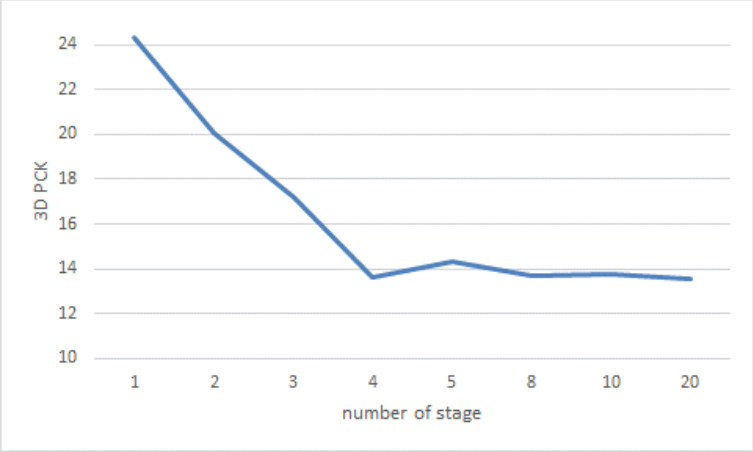


圖 24、Hierarchical Optimization 不同階段數量在 RHD 上的實驗結果

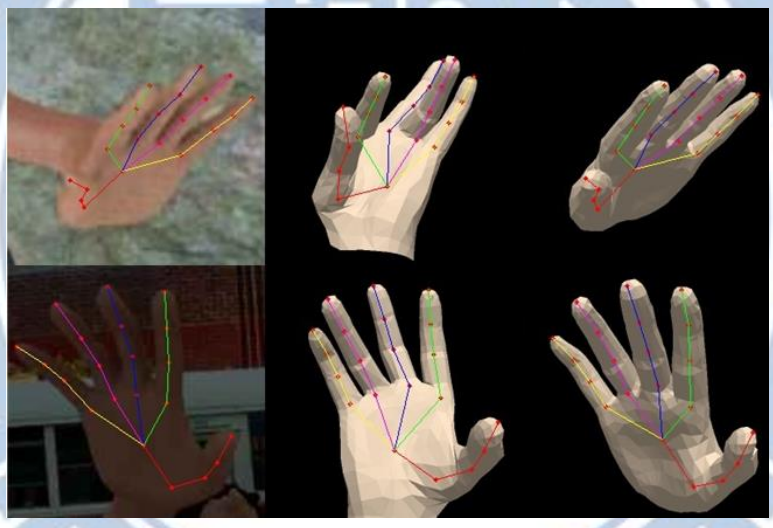


圖 25、結果比較圖，每組包含上下兩張圖片，由左到由分別為 ground truth、前人提出的逆向運動學、以及用本論文提出的設定初始值方法所算出來的結果。

5.7.2 以估計 2D 關鍵點為輸入

當我們以神經網路所估計的 2D 關鍵點位置作為輸入時，我們注意到如果有少數 2D 關鍵點估計錯誤，可能會大大影響逆向運動學之結果，因此我們將 **Eq. (10)**稍作修改，多加入介於 0 到 1 之間變數r，首先將 Eq.(9)中目標函數每個關鍵點的平方差都乘上一個變數r_i，作為每個關鍵點的權重，同時為避免最佳化演算法讓r_i為 0 來降低目標

函數的值，因此我們在目標函數後加上$\alpha(1-r_i)^2$來避免，此方法可以使最佳化演算法自己決定每個關鍵點所佔的權重，讓目標函數增加一些彈性，並非絕對相信神經網路所估計出的結果，可將式子改寫成

$$\sum_{i=1}^{21} r_i p_i((x_i-u_i)^2+(y_i-v_i)^2))+\alpha(1-r_i)^2 \qquad (10)$$

其中α為常數。

最後為了確認本節演算法在正常狀況下的表現，我們以神經網路估計的 2D 關鍵點做為目標，MANO 作為手部模型，對本節所提到的演算法做實驗，測試個別以及同時使用的效果，表 7 為實驗結果。

表 7、本節實驗結果

With Eq.(10)			✓	✓	✓	✓
Palm orientation				✓		✓
Hierarchical optimization					✓	✓
Mean EPE (mm)	RHD	29.99	28.44	25.78	24.48	21.87
	STB	31.29	31.03	30.88	29.57	29.47

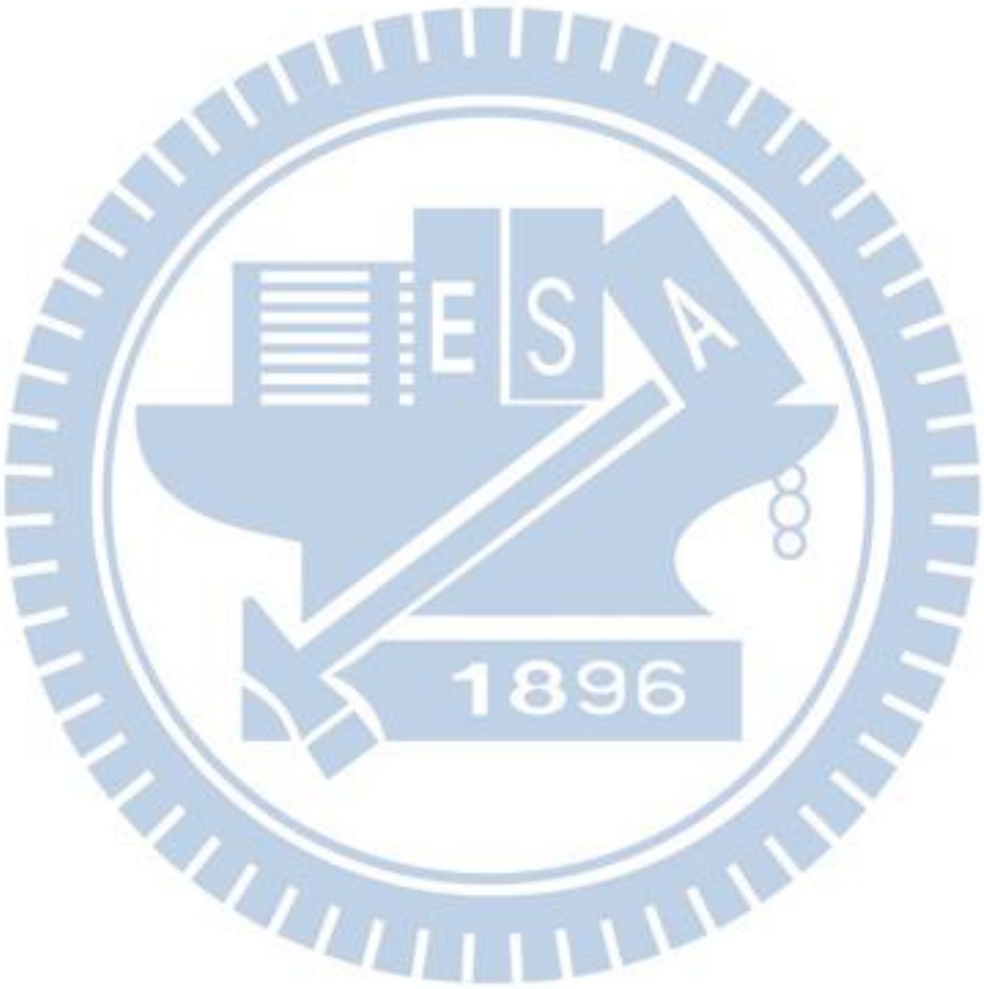
5.8 Refine 手部關鍵點

在神經網路估計 2D 手部關鍵點實驗中，我們觀察到若關鍵點被遮擋時，準確度會大幅降低，因此我們希望能藉由 3D 模型的結果去 refine 2D 的關鍵點。同時也能驗證我們的演算法是否能藉由一部分較為正確的 2D 關鍵點來推測 3D 手部姿勢，而不受錯誤的結果影響。由於被遮擋的部分會使神經網路估計的信心較低，因此我們將信心低於 0.5 的關鍵點替換成 3D 手部模型應設在圖片上的結果。表 8 為實驗結果。

表 8、藉由 3D 模型 refine 關鍵點的結果

Mean EPE	RHD	STB

	Without refine		With refine	
CPM	5.85	5.31		
HG	5.20	4.74		
MSPN	4.43	4.20	5.88	5.57



第六章、結論與未來展望

本論文以單攝影機第一人視角的頭戴式系統為使用平台，結合卷積神經網路以及手部模型，建立一套可靠的手部姿勢估計系統。

在神經網路架構方面，對於神經網路架構選擇以及架構整合，本論文進行多項實驗與分析，評估多種神經網路架構在不同資料集中的表現，藉此找到最適合的網路架構。

在手勢估計中的逆向運動學方面，我們進行一連串的實驗，發現迭代演算法的收斂容易收斂於局部最小值，以及對於少數關鍵點錯誤過於敏感的問題。對於前者，我們從初始值以及迭代演算法上下手，而後者我們透過修改目標函數，來減少此種狀況，藉由上述方法，這兩個問題皆得到不小的改善。

然而對於不同使用者，我們沒有一個好的方式去調整手部模型的骨架比例，而根據我們的實驗，若使用正確的骨架比例，可降低約 10mm 的誤差，約為本論文所提出方法誤差的一半，因此若能根據不同使用者來調整骨架比例，將可大幅提升手部姿勢估計的精確度。

要能隨不同使用者調整骨架，一個可行的做法是一開始先要求使用者做一個標準姿勢，利用該姿勢來計算出標準骨架的調整參數。另一種解決方案是引進在 4.5 節提到的多任務學習方法，讓骨架參數也是訓練過程中學習的目標之一，這些是未來可以進一步改善的方向。

參考文獻

[1] Y. Yang and D. Ramanan, “Articulated Pose Estimation with Flexible Mixtures-of-Parts,” *Proc. of 2011 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1385–1392.

[2] B. Sapp, C. Jordan and B. Taskar, “Adaptive Pose Priors for Pictorial Structures,” *Proc. of 2010 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 422-429.

[3] V. Belagiannis, S. Amin, M. Andriluka, B. Schiele, N. Navab and S. Ilic, “3D Pictorial Structures for Multiple Human Pose Estimation,” *Proc. of 2014 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1669-1676.

[4] L. Pishchulin, M. Andriluka, P. Gehler and B. Schiele, “Poselet Conditioned Pictorial Structures,” *Proc. of 2013 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 588-595.

[5] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional Pose Machines,” *Proc. of 2016 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4724-4732.

[6] A. Toshev and C. Szegedy, “DeepPose: Human Pose Estimation via Deep Neural Networks,” *Proc. of 2014 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1653-1660.

[7] A. Newell, K. Yang and J. Deng, “Stacked Hourglass Networks for Human Pose Estimation,” *Proc. of 2016 European Conf. on Computer Vision (ECCV)*, 2016, pp. 483-499.

[8] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei and J. Sun, “Rethinking on Multi-Stage Networks for Human Pose Estimation,” 2019, arXiv:1901.00148.

[9] Z. Cao, T. Simon, S.-E. Wei and Y. Sheikh, “Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields,” *Proc. of 2017 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7291-7299.

[10] K. He, G. Gkioxari, P. Dollar and R. Girshick, “Mask R-CNN,” *Proc. of 2017 IEEE Int. Conf. on Computer Vision (ICCV)*, 2017, pp. 2961-2969.

[11] L Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler and B. Schiele, “Deepcut: Joint Subset Partition and Labeling for Multi Person Pose Estimation,” *Proc. of 2016 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4929-4937.

[12] T. Simon, H. Joo, I. Matthews and Y. Sheikh, “Hand Keypoint Detection in Single Images Using Multiview Bootstrapping,” *Proc. of 2017 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4645-4653.

[13] A. Spurr, J. Song, S. Park and O. Hilliges, “Cross-Modal Deep Variational Hand Pose Estimation,” *Proc. of 2018 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 89-98.

[14] C. Zimmermann and T. Brox, “Learning to Estimate 3D Hand Pose from Single RGB Images,” *Proc. of 2017 IEEE Int. Conf. on Computer Vision (ICCV)*, 2017, pp. 4913-4921.

[15] P. Panteleris, I. Oikonomidis and A. Argyros, “Using a Single RGB Frame for Real Time 3D Hand Pose Estimation in the Wild,” *Proc. of 2018 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2018, pp. 436-445.

[16] Y. Cai, L. Ge, J. Cai and J. Yuan, “Weakly-Supervised 3D Hand Pose Estimation from Monocular RGB Images,” *Proc. of 2018 European Conf. on Computer Vision (ECCV)*, 2018, pp. 666-682.

[17] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar and C. L. Zitnick, “Microsoft coco: Common Objects in Context,”. *Proc. of 2014 European Conf. on Computer*

Vision (ECCV), 2014, pp. 740-755.

[18] O. Glauser, S. Wu, D. Panozzo, O. Hilliges and O. Sorkine-Hornung, “Interactive Hand Pose Estimation Using a Stretch-Sensing Soft Glove,” *ACM transactions on graphics (TOG)*, 2019, pp. 41:1-41:15.

[19] R. Y. Wang and J. Popović, “Real-Time Hand-Tracking with A Color Glove,” *ACM transactions on graphics (TOG)*, 2009, pp. 63:1-63:8.

[20] I. Oikonomidis, N. Kyriazis and A. A. Argyros. “Efficient Model-Based 3D Tracking of Hand Articulations Using Kinect,” *Proc. of British Machine Vision Conference (BMVC)*, 2011, pp. 101.1-101.11.

[21] C. Xu and L. Cheng, “Efficient Hand Pose Estimation from a Single Depth Image,” *Proc. of 2013 IEEE Int. Conf. on Computer Vision (ICCV)*, 2013 pp. 3456-3462.

[22] J. Choi, B.-J. Lee, B.-T., Zhang, “Human Body Orientation Estimation Using Convolutional Neural Network,” 2016, arXiv:1609.01984.

[23] C. Wan, T. Probst, L. V. Gool and A. Yao, “Dense 3D Regression for Hand Pose Estimation,” *Proc. of 2018 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5147-5156.

[24] L. Ge, H. Liang, J. Yuan and D. Thalmann, “3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images,” *Proc. of 2017 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5679-5688.

[25] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu and Q. Yang, “3d Hand Pose Tracking and Estimation Using Stereo Matching,” 2016, arXiv:1610.07214.

[26] M. Ye, Xianwang Wang, R. Yang, Liu Ren and M. Pollefeys, “Accurate 3D Pose Estimation from A Single Depth Image,” *Proc. of 2011 IEEE Int. Conf. on Computer Vision (ICCV)*, 2011, pp.731-738.

[27] 蔡運帷，「頭戴式 VR 的類神經網路手勢辨識系統設計」，國立交通大學，碩士論文，

民國 107 年。

[28] 許頌伶，「利用三維模型訓練類神經網路的手勢辨識技術」，國立交通大學，碩士論文，民國 105 年。

[29] Marin Saric. LibHand: A Library for Hand Articulation, 2011.version 0.9. http://www.libhand.org/

[30] J. Romero, D. Tzionas and M. J. Black, "Embodied Hands: Modeling and Capturing Hands And Bodies Together," *ACM Transactions on Graphics (TOG)*, 2017, pp. 245:1-245:17.

[31] 呂芳鎮「頭戴式 VR 的類神經網路手部語意切割系統設計」，國立交通大學，碩士論文，民國 108 年。

[32] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *Proc. of 2016 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778.

[33] M. Schröder, J. Maycock, H. Ritter and M. Botsch, "Real-Time Hand Tracking Using Synergistic Inverse Kinematics," *Proc. of 2014 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014, pp. 5447-5454.

[34] J. Redmon and A. Farhadi, "Yolov3: An Incremental Improvement," 2018, arXiv:1804.02767.

[35] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," *Proc. of 2019 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 11807-11816.

[36] H. Kato, Y. Ushiku and T. Harada, "Neural 3D Mesh Renderer," *Proc. of 2018 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3907-3916.

[37] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," *Proc. of European Conf. on Computer Vision (ECCV)*, 2018, pp. 801-818.

[38] A. Popa, M. Zanfir and C. Sminchisescu, "Deep Multitask Architecture for Integrated 2D and

3D Human Sensing," *Proc. of 2017 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4714-4723.

[39] Y. Wang, C. Peng and Y. Liu, "Mask-pose Cascaded CNN for 2D Hand Pose Estimation from Single Color Image," *IEEE Transactions on Circuits and Systems for Video Technology*, Nov. 2018.

[40] M. Everingham, L. Vam Gool, C. K. I. Williams, J. Winn and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Int. Journal of Computer Vision*, Jun. 2010, pp. 303-338.

[41] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. "Feature Pyramid Networks for Object Detection," *Proc. of 2017 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2117-2125.

[42] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *Proc. of 2016 IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779-788.

[43] P. Liu, W. Liu and H. Ma, "Weighted Sequence Loss Based Spatial-Temporal Deep Learning Framework for Human Body Orientation Estimation," *Proc. of 2017 IEEE Int. Conf. on Multimedia and Expo (ICME)*, 2017, pp. 97-102.

[44] 黃雅琦，「利用攝影機二維影像做三維手勢追蹤」，國立交通大學，碩士論文，民國 104 年。