

基於機器學習可供抓枝機器人抓握之凸起物體辨識

楊辰修¹ 王奕凱² 林紀穎³

¹ 國立臺灣科技大學 機械工程系研究所

E-mail: M10603402@mail.ntust.edu.tw

² 國立臺灣科技大學 機械工程系研究所

E-mail: M10903403@mail.ntust.edu.tw

³ 國立臺灣科技大學 機械工程系研究所

E-mail: chiying@mail.ntust.edu.tw

摘要

本研究針對全自主性攀爬機器人利用神經網路開發一套視覺系統，對於冷氣、水泥屋簷與招牌等三種常出現於住宅外牆且可供攀爬機器人抓握之凸出物進行辨識與定位。由於攀爬型機器人無法背負過重的電腦進行神經網路運算，故我們導入了 IOT 與雲端運算的概念，將照片經由無線傳輸由機器人傳送至伺服器進行神經網路的運算。本研究使用快速物件檢測神經網路 (Fast Region Convolutional Neural Network, Fast R-CNN) 架構實現攀爬場景辨識系統，並結合立體相機裝置估算物體與機器人間的實際距離。最後透過實際在室外住宅外牆實驗，測試在不同光源下的辨識結果；並且模擬攀爬機器人在不同視角下的測試結果，證實了本視覺系統的可行性。

關鍵字：影像辨識、電腦視覺、機器學習、攀爬機器人。

1. 前言

高樓外牆經常需仰賴大量的人力來進行結構檢測、管路維修或是窗戶清潔等工作。但高空作業對於人類來說具有相當的危險性，於是本研究希望藉由機器人來取代人類進行上述的高風險工作，而「自主性」則是決定攀爬機器人價值的關鍵因素。自主性牆面攀爬，其中最重要的部分是接收環境資訊，即視覺感測器。對於攀爬機器人上的應用而言，系統需要在一張影像中找出可抓握的物體。但可抓握的物體種類繁雜，外觀是不固定的，這使得物體辨識較為困難，無法透過單純的影像處理得出結果，需要透過複雜的演算法對整張影像進行「理解」，利用神經網路演算法對其特徵做搜尋、整理與辨識。台灣為潮濕多雨的自然環境，在台灣的住宅外牆可以發現許多適合作為機器人抓握的物體，比如冷氣室外機、水泥屋簷、招牌上方。這些物體表面平坦適合攀爬機器人抓握，且具有適當的水平間距，故本研究以該三種物體做為辨識目標。

由於神經網路運算十分複雜，若想在機器人攀爬期間同時進行影像處理，則需使用電腦進行運

算。然而機器人本體因重量限制等因素並無法搭載電腦，因此，視覺系統需要分成兩部分，一部分僅進行資料收集拍照，另一部分進行複雜運算；兩部份的溝通使用無線傳輸進行連接。在神經網路架構與系統運作流程方面，[1]對物件檢測、語意分割與三維物體辨識做了統整性的流程概述，且分析方法上的優劣與適用範圍；[2]比較各種預訓練網路，利用轉移學習概念改成物件檢測神經網路架構後的精確度；[3]利用了物件檢測概念，在整張圖裡標記目標的位置，並取得了良好的辨識結果；[4]以 Fast R-CNN 架構做為視覺系統的基礎，以辨識特定目標。

本研究參考[5]的攀爬步態，如圖 1.1。提出一套從資料蒐集到環境辨識的完整流程，利用神經網路作為住宅外牆常見的冷氣、水泥屋簷與招牌辨識，加上 IOT 與雲端運算概念，實現機器人自主攀爬的輕量級視覺系統，並且為了未來的自主路徑規劃提供完善的基礎。結合深度視覺，判斷前方物體種類與距離。最後以模擬機器人攀爬時的視角進行實驗，並以此證實系統的可行性。

本文剩餘內容說明如下。第二章介紹物體辨識與測距方法；第三章說明系統完整架構與實驗流程；第四章為實驗結果分析；最後為結論以及未來研究展望。

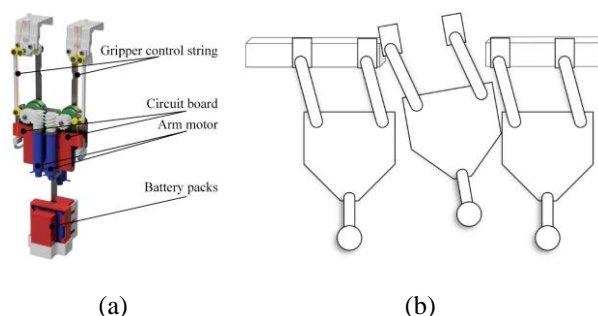


圖 1.1、(a)參考機器人硬體 (b)機器人跳躍方法[5]

2. 物體辨識與測距

2.1. 3D 立體影像

3D 立體影像是具有三個維度的資料，個別維

度資料獨立，且具有相同的解析度，在影像擷取方法上可利用深度相機達成。深度相機具備兩種鏡頭，除了一般的 RGB 鏡頭用以拍攝 RGB 照片外，另外擁有一組紅外線接收與發射鏡頭，以計算深度資訊，並儲存為三維點雲(Point Cloud) 模型，圖 2.1 為原始 RGB 照片與相應的點雲模型。

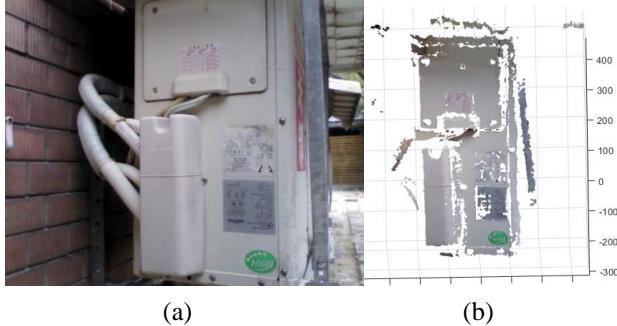


圖 2.1、冷氣 RGB 與對應點雲

2.2. 物件檢測與卷積神經網路

物件檢測(Object Detection)是一種在影像中查找與分類物件的深度學習方法，當我們需要在一張影像中找出某種特定目標物並定位時，該方法將矩形區域建議與卷積神經網路功能結合起來，即為 Regions with Convolutional Neural Networks (R-CNN)。R-CNN 演算法使用區域建議方法，例如選擇性搜尋(selective search)[6][7]從影像中提出許多方框，並檢查這些方框中是否包含任何物件。

為減少計算次數，本研究使用 Fast R-CNN 作為神經網路的架設。Fast R-CNN[8]架構中每個影像只執行一次 CNN，而在 2 千個區域中找到一種計算方法並共享之。首先直接將整張影像輸入至 CNN，並得到影像的特徵圖；利用特徵圖結合區域建議方法提取感興趣區域，輸入至 ROI 池化層以確保每個區域大小皆相同。最後使用全連接層(fully-connected layers)進行分類判斷邊界框。

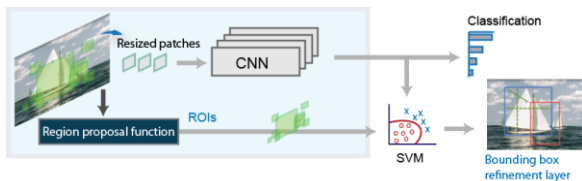


圖 2.2、R-CNN [9]

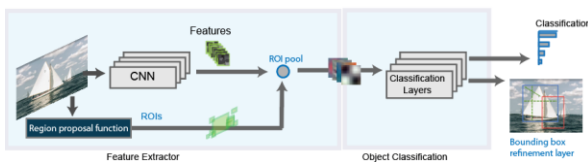


圖 2.3、Fast R-CNN [9]

2.3. 轉移學習

轉移學習是把已經訓練好的模型架構、參數，轉移至另外的一個新模型上，使得不需要從零開始重新訓練一個新模型，我們可以利用一個訓練好的

CNN 模型，該模型已經在別處透過數百萬張的圖片訓練而成，接著把這個模型套用至其他影像辨識的模型上，因為將原模型對圖片的理解能力保留下來，大部分的權重不需要再重新更新，不但節省大量時間，也保持了準確率。在 MATLAB 工具箱有提供約 11 種模型可供下載，如表 2.1，網路架構越深代表訓練與執行速度會越慢；並參考圖 2.來選擇最適當的架構。縱軸為正確率，橫軸為進行一次正向傳輸所需的操作數量，圓圈大小則正比於網路中的參數數量。我們希望選擇一個正確率高、執行速度又快的網路。在縱軸上，越上方的網路正確率越高；而在橫軸越右方的網路，其操作數量越多，代表執行速度越慢。比較各架構模型的精確度後，本研究選擇 GoogLeNet 作為主要神經網路架構。

表 2.1、神經網路架構與深度

架構	深度	架構	深度
AlexNet	8	ResNet50	50
VGG-16	16	ResNet101	101
VGG-19	19	GoogLeNet	22
SqueezeNet	18	MobileNet-v2	54
ResNet18	18	Inception-v3	48

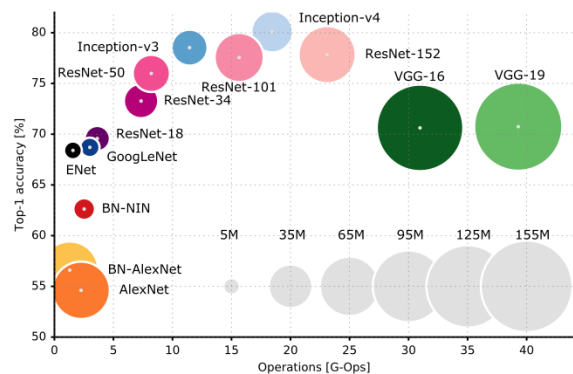


圖 2.4、各神經網路正確率與計算量 [10]

2.4. 神經網路精確度

在物體檢測中，模型的性能指標主要透過平均精度，mean Average Precision(mAP)來確定。平均精度則透過準確率(Precision)與召回率(Recall)之間的關係來計算，其值介於 0 到 1 之間。如圖 2.，可將資料分為四個部分：true positives(TP, 正確分類的正例)、false negatives(FN, 本來是正例，錯分為負例)、true negatives(TN, 正確分類的負例)、false positives(FP, 本來是負例，錯分為正例)，透過這四種資料型態可以計算模型的準確率與召回率。

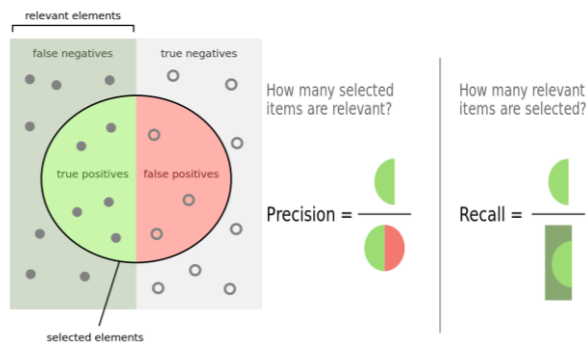
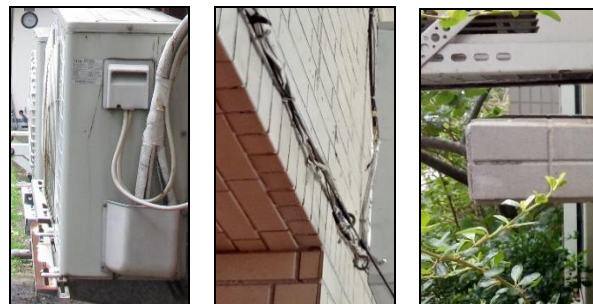


圖 2.5、資料分類 [11]



(a) (b) (c)

圖 3.2、訓練資料集示例

(a)冷氣 (b)招牌 (c)水泥屋簷

3. 系統與實驗流程

3.1. 系統架構

本研究程式運作流程如圖 3。為符合機器人應用需求，在相機拍攝之前需設定每個運作循環的延遲時間，以配合機器人攀爬時每一步的時間，避免在機器人過大晃動時拍攝照片，確保照片的清晰與穩定。相機同時拍攝 RGB 照片與深度資訊各一組。其中 RGB 照片將會被使用在神經網路物體辨識，若沒有目標物則系統進入下一個循環；若成功辨識物體，系統將利用深度資訊建立點雲，並以辨識結果結合點雲回傳相機與物體的相對距離。

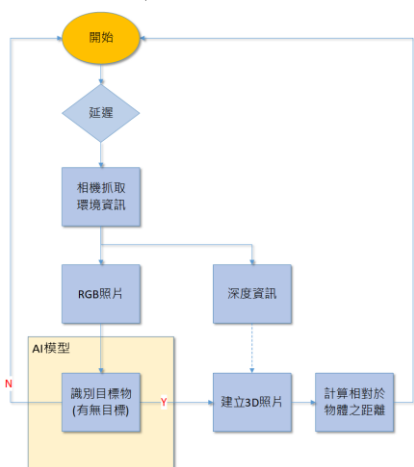


圖 3.1、系統運作流程

3.2. 資料收集

本研究所使用的物體辨識系統為神經網路，需要大量的訓練資料，包含冷氣、招牌及水泥屋簷等三種目標物體，且訓練資料需與實際應用場景相似；本研究的目的為應用於攀爬機器人上，故訓練資料必須以機器人的視角建立，即目標物體的側面。依此要求下，網路上並沒有可用的資料集供下載，所以我們自行於白天實地拍攝，建立了訓練資料集，照片為彩色，大小皆大於 600x400。圖 3。為訓練資料集各目標物拍攝方法示例。其中原圖冷氣有 819 張，招牌有 839 張，水泥屋簷有 514 張。

3.3. 系統整合

在不影響機器人的攀爬為前提，位於其上的視覺系統必須輕量化。對於攀爬型機器人來說，搭載電腦勢必導致機構龐大與設計困難。本節將介紹為配合攀爬型機器人所設計的一套系統流程，其分為機器人端與伺服器端兩部分如**錯誤！找不到參照來源**。所示。將較為輕量的作業分於機器人端，例如拍照與存檔。而需要較高硬體支援的工作交給伺服器端，像是運行神經網路、影像處理、3D 位置計算等。兩者使用無線傳輸相互溝通，將神經網路、深度感測與硬體配置結合，實現自動完整系統流程。

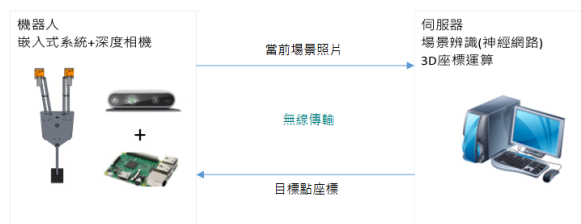


圖 3.3、機器人與伺服器概念圖

3.3.1. 機器人端

搭載於攀爬型機器人上的系統部分，在輕量化的基礎上，須具備基本的影像接收與傳遞功能，為接收環境資訊，使用一台深度相機進行拍照與深度資訊收集；再透過無線傳輸將資料傳送到伺服器端。

3.3.2. 伺服器端

系統中較為繁重的處理作業集中在伺服器端執行，由於伺服器端不需要隨機器人移動，硬體使用的桌上型電腦，以能夠順暢執行神經網路為標準。伺服器接收到資料後，將 RGB 照片作為神經網路的輸入，進行目標物檢測；當照片中存在目標物時，神經網路輸出目標物的邊框位置與目標物名稱，此時才會導入深度資訊，利用邊框位置在深度資訊中尋找相對應位置的深度，並同時將所有資訊輸出。若當前照片中無目標物存在，則進行下一張照片的檢測，減少處理深度資訊的時間。

3.4. 實驗方法

為證實本系統的實際應用於攀爬型機器人上的價值，我們設計了模擬實際應用的實驗方法，希望在不同的場景下，系統運作能夠穩定，且能達到良好的結果。本研究僅著重於探討攀爬機器人的視覺系統，並未設計一機器人以進行實際攀爬測試；故實驗仍以筆記型電腦做為機器人端的執行系統，並手持深度相機以模擬機器人攀爬與移動。伺服器端的桌上型電腦將位於室內固定位置，以有線網路連接並具備固定 IP 位址，以提供穩定伺服器服務。客戶端皆為可攜式裝置，實驗中將以客戶端代表攀爬型機器人，到各場景移動測試；筆記型電腦使用 WiFi 透過網際網路連上伺服器端，筆記型電腦連接深度相機進行拍照。

3.4.1. 不同光源測試

因本系統神經網路訓練資料僅有白天，本實驗方法希望探討在白天、陰天(黃昏)、晚上等光源下進行物體辨識，探討系統限制及其優化方法。

3.4.2. 干擾與雜訊

對於在牆面攀爬的機器人來說，會造成照片的不清晰，屏除機器人晃動時正好進行拍照以外，另一種可能就是下雨等雜訊干擾。本實驗將測試在此種情況下，系統是否仍能正常運作。

3.4.3. 偏轉角度測試

由於沒有攀爬機器人可供測試，故本實驗主要是為了模仿機器人的視角，考慮到機器人攀附在物體上，並朝下一個目標物拍照，此時相機需保持一偏轉角度 θ ，讓目標物能夠完整存在於視野範圍內。

4. 系統與實驗流程

4.1. 物體檢測精確度

每種目標物測試資料各為 200 筆，以做為計算精確度的依據，結果如表 4.1 所示。本系統的 mAP 為 54.6%，其中造成 mAP 無法達到 100% 的原因，影響最深的就是訓練資料了，本研究無法蒐集大量的資料，且也無現成資料庫可供訓練用，僅自行拍攝蒐集每種物體各約 1 千張資料，因而導致辨識率無法提升。而本研究自行拍攝照片，難免照片會出現質量參差不齊，對於神經網路來說是無法找出特徵，但在整理資料時人工難以察覺的資料。外觀差距甚大的同種物體，也是導致辨識結果較差的原因。

表 4.1、各目標物的精確度

目標物體	AP	mAP
冷氣	60.6%	54.6%
招牌	51.9%	
水泥屋簷	51.2%	

4.2. 距離檢測結果

RGB 照片輸入神經網路進行物體檢測，得到物體名稱及邊界框位置如圖 4.1(a)；接下來利用邊界框中心位置的 x,y 座標當作依據，在點雲中對應到其相對位置如圖 4.1(b)，則可以獲得該點的深度資訊。該點位置一定位於目標物體上，且接近中心的位置，故可將該點深度距離視為物體整體距離相機的距離。

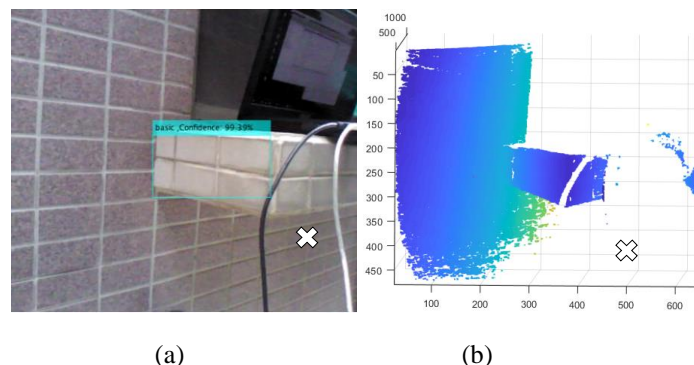


圖 4.1、(a)辨識後邊界框的中心(標記處)(b)中心對應到點雲的位置示意圖

經過多次實測發現，在實驗中深度相機所量測深度資訊與實際距離大平均約多 5 公分，統計數據如表 4.2。故在運用此深度相機進行測量時，須對所有結果進行補正 5.25 公分，以得到正確的距離。

表 4.2、深度測量較準統計(單位：公分)

目標物	測量次數	實際距離平均	測量距離平均	誤差平均
冷氣	12	68.8	74.2	5.3
水泥屋簷	6	39.7	44.9	5.26
招牌	9	72.6	77.8	5.2

4.3. 實驗結果

4.3.1. 不同光源測試

在夜晚或場景過暗時，容易造成無法判定目標物體；這是因為訓練資料庫內並沒有較暗場景的訓練資料。為解決此問題，在照片輸入神經網路之前，對其進行「直方圖等化」，是影像處理領域中利用影像直方圖對對比度進行自動調整的方法。圖 4.2 為系統在較低光源下測試的結果。可以發現在低光源環境，沒有加入影像前處理的步驟，導致神經網路無法辨識目標(圖 4.2(a))；而在有經過了影像前處理步驟的系統下，能夠使輸入的照片仍然有足夠的亮度，確保系統能夠穩定辨識(圖 4.2(b))。

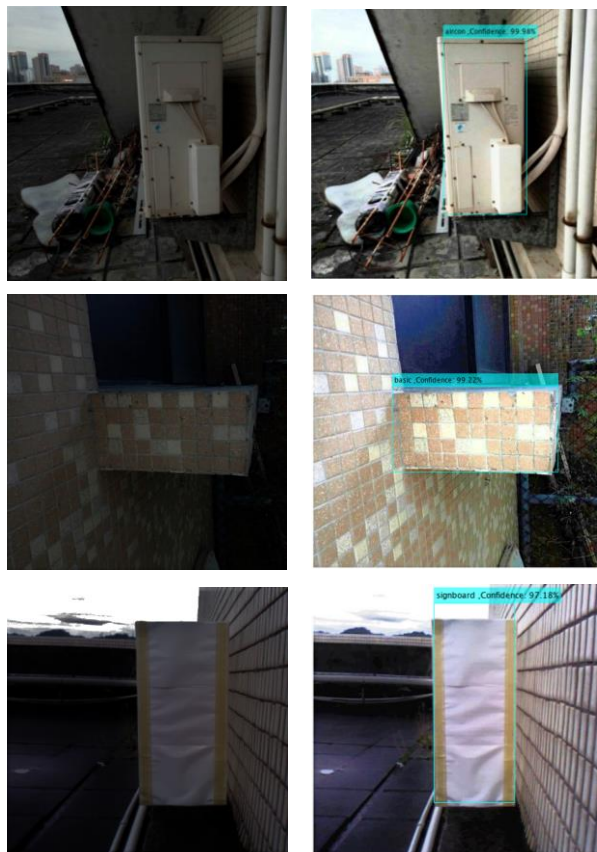


圖 4.2、三種目標物體測試 (a)光源不足 (b)亮度調整後辨識結果

4.3.2. 干擾與雜訊

本實驗在雨中拍照並將照片輸入神經網路，部分結果如圖 4。雨是造成照片雜訊的因素，使得照片中的主體不清晰，這可能導致神經網路在抽取特徵時較為困難。在結果中可以看到，即使在人眼中看起來仍然明顯可辨識，但對神經網路來說，雜訊會大大影響辨識能力。使用均值濾波等方法將雜訊去除(圖 4.3)，雖然雨滴消失，但是會導致整張照片變模糊，仍然無法穩定辨識。



(a)



(b)



(c)

圖 4.3、帶有雜訊的物件檢測
 (a) 冷氣 (b) 水泥屋簷 (c) 招牌



圖 4.3、去除雜訊








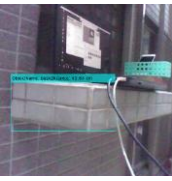
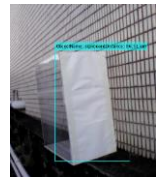
解決下雨辨別度不佳的方法為增加下雨時的訓練資料，也可以加入略為模糊的訓練資料以因應雜訊較多的特殊情況，例如鏡頭髒汙等。但本研究訓練資料並沒有包含下雨天，對於雨天的物件檢測會導致辨識能力下降，故本系統目前僅適用於能夠拍出較為清晰照片的環境。

4.3.3. 偏轉角度測試

本節實驗為為模仿機器人的視角，偏轉一角度對目標物體拍照；

表 4.3 為系統對三種目標物的辨識，以及相機偏轉一固定角度後的結果。本系統在三種物體的辨識上有良好的效果，除了正對著物體拍攝能夠辨識外，對於偏轉 15°、大至 30°等，都可以成功辨識物體與距離。在應用上即使機器人攀附在牆面物體上，視覺系統的角度較為特殊，但透過此實驗證明了本系統的開發方向，是專門為了攀爬機器人所設計的，能夠實裝於機器人上運作。

表 4.3、系統辨識結果

偏轉	冷氣辨識結果	屋簷辨識結果	招牌辨識結果
0°			
15°			
30°			

5. 系統與實驗流程

本研究提出一套在目前文獻中仍未有該應用的攀爬機器人視覺系統，針對台灣大樓外牆常見的冷氣、水泥屋簷或招牌等牆面凸起物為研究目標，透過相機對彩色與深度環境資訊的擷取，並利用 IOT 的概念，透過無線傳輸將機器人與伺服器連線，資料即時傳至伺服器進行雲端運算以節省機器人搭載的硬體。自己架設的伺服器擁有較多的硬體資源，故導入機器學習物件檢測，透過自行拍攝與製作住宅外牆之冷氣、水泥屋簷及招牌的訓練資料，以轉移學習增加神經網路的能力，並以物件檢測方法使得系統能夠圈選辨識這些凸起物，同時利用深度資料以計算相機與物體間的距離；最後透過實驗方式模擬機器人攀爬時的視角，能夠成功辨識目標物體與判斷距離，以證實本系統有實用價值。

6. 參考文獻

- [1] Z. S. Pershina, "Application of algorithms for object recognition based on deep convolutional neural networks for visual navigation of a mobile robot," *25th International Conference on Integrated Navigation Systems (ICINS)*, Saint Petersburg. May, 2018.
- [2] Chungkeun Lee, H. Jin Kim, Kyeong Won Oh, "Comparison of faster R-CNN models for object detection," *16th International Conference on Control, Automation and Systems (ICCAS)*, Gyeongju, South Korea. 16-19 Oct. 2016.
- [3] Ankush Gupta, Andrea Vedaldi, Andrew Zisserman, "Synthetic Data for Text Localisation in Natural Images," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA. 27-30 June 2016.

- [4] Lei Quan, Dong Pei, Binbin Wang, Wenbin Ruan, "Research on Human Target Recognition Algorithm of Home Service Robot Based on Fast-RCNN," *10th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, Changsha, China. 9-10 Oct. 2017.
- [5] 楊宗翰, "橫向飛躍抓枝機器人之設計改良與運動控制研究," 國立台灣科技大學機械工程系研究所. 2018.
- [6] Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, Arnold W. M. Smeulders, "Selective Search for Object Recognition," *International Journal of Computer Vision*, vol. 104, pp. 154–171, 2013.
- [7] Koen E. A. van de Sande, Jasper R. R. Uijlings, Theo Gevers, Arnold W. M. Smeulders, "Segmentation as selective search for object recognition," *International Conference on Computer Vision*, Barcelona, Spain. 12 January, 2012.
- [8] Ross Girshick, "Fast r-cnn," *IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile. 7-13 Dec. 2015.
- [9] "Faster R-CNN," https://ww2.mathworks.cn/help/vision/ug/faster-r-cnn-basics.html#mw_68ca6afd-f4b0-4c54-8df2-38d40e9e45ad.
- [10] Alfredo Canziani, Adam Paszke, Eugenio Culurciello, "An Analysis of Deep Neural Network Models for An Analysis of Deep Neural Network Models for Practical Applications," *ICLR conference submission*.
- [11] By Walber, CC BY-SA 4.0, "Precision and recall," <https://commons.wikimedia.org/w/index.php?curid=36926283>.