

2025仰望盃全國科學HomeRun實作大賽

決賽成果報告書

隊伍名稱: 三隻猴子闖Homerun

作品名稱: 無視重力竄天猴

科學概念1: 白努利定律。根據 $\frac{1}{2}\rho v^2 + \rho gh + p = constant.$, 當氣流速度 v 增加時, 壓力 p 會降低。爬牆車上的風扇運作時, 會高速抽取車底閉環內的空氣, 使底部的氣流流速增加, 根據白努利定律, 這會導致車底閉環內的氣壓降低。此時, 車外的環境氣壓相對較高, 因而產生了一個壓力差。

科學概念2: 壓力原理 $F=A \times P$, 當面積 A 相同時, 壓力 P 越大, 正向力 F 也越大。風扇運轉時, 車底閉環內的氣壓逐漸降低, 閉環內氣壓越低, 與外界壓力差就越大, 產生的正向力 F 也越大, 便能夠將車體壓在牆壁上。

科學概念3: 正向力和摩擦力的原理。在相同牆面上有相同的摩擦係數 μ_s , 根據 $f_{s(max)} = \mu_s N$, 正向力 N 越大時, 物體的最大靜摩擦力 $f_{s(max)}$ 也會變大。當車底和牆壁間形成一股吸附力時, 車體對於牆面的正向力 N 變大, 平行牆面方向的最大靜摩擦力 $f_{s(max)}$ 變大, 車體能承受、吊掛的重量也隨之增加。

科學概念4: 力矩原理 $\tau = R \times F$, 當作用力相同時, 力臂越長, 力矩變越大。爬牆車的無刷馬達、直流電機, 以及開發板和感測器, 幾乎都裝在底板上, 因此車體重心也是在底板上。當車體底板和牆面的距離越遠時, 力臂便越長, 重力力矩也越大。

決賽成果報告書內文

(最多10頁)

1. 發想動機：

在日常生活中，我們經常可以看到工人吊掛在幾十層樓高的建築外牆上進行清潔、維修或施工工作。他們身處高空，僅靠幾條繩索支撐身體，來回移動於樓層之間，這樣的畫面總讓人不禁捏了一把冷汗，為他們的安全感到擔憂。根據勞動部的統計資料顯示，112年因職業災害而導致死亡的案例中，有將近一半是因「墜落」或「滾落」所造成，這不僅凸顯高空作業的危險性，也反映出目前在安全防護上仍有相當大的改進空間。

這樣的數據與現象引發我們的思考—是否有可能透過科技的力量，減少工人暴露在高風險工作環境中的機會？如果能研發出一款能在牆面自由爬行、穩定移動的機器，就能夠在不需要人工高空作業的情況下完成部分甚至全部的任務，像是外牆清洗、建築檢修或災後搜救等，為工作安全開創新的局面。

因此，我們的目標是設計並打造一款具有高度靈活性與環境適應性的爬牆車。這款機器將結合多種感測與動力控制技術，能夠在不同材質、不同角度甚至風力干擾的牆面上穩定行駛，不受傳統遙控車只能在地面移動的限制所束縛。未來，我們希望它不僅能取代部分高風險人力操作，更能應用於更多創新場景中，如智慧建築維護、城市探索與環境監測等，成為推動自動化與智慧化工程的關鍵一環。

2. 作品創意性：

傳統的遙控車受限於重力與地形，只能在水平地面或坡度有限的表面上行駛，難以應對如垂直牆面等極端環境。為突破此限制，我們設計並打造了一款具備風扇吸附機制的創新型爬牆車。其核心原理是透過風扇產生氣壓差，使車體可牢牢吸附於牆面之上，實現穩定攀爬、自由移動，有效防止掉落，即使在垂直或傾斜牆面上也能順利作業，顯著提升了其應用彈性與環境適應力。

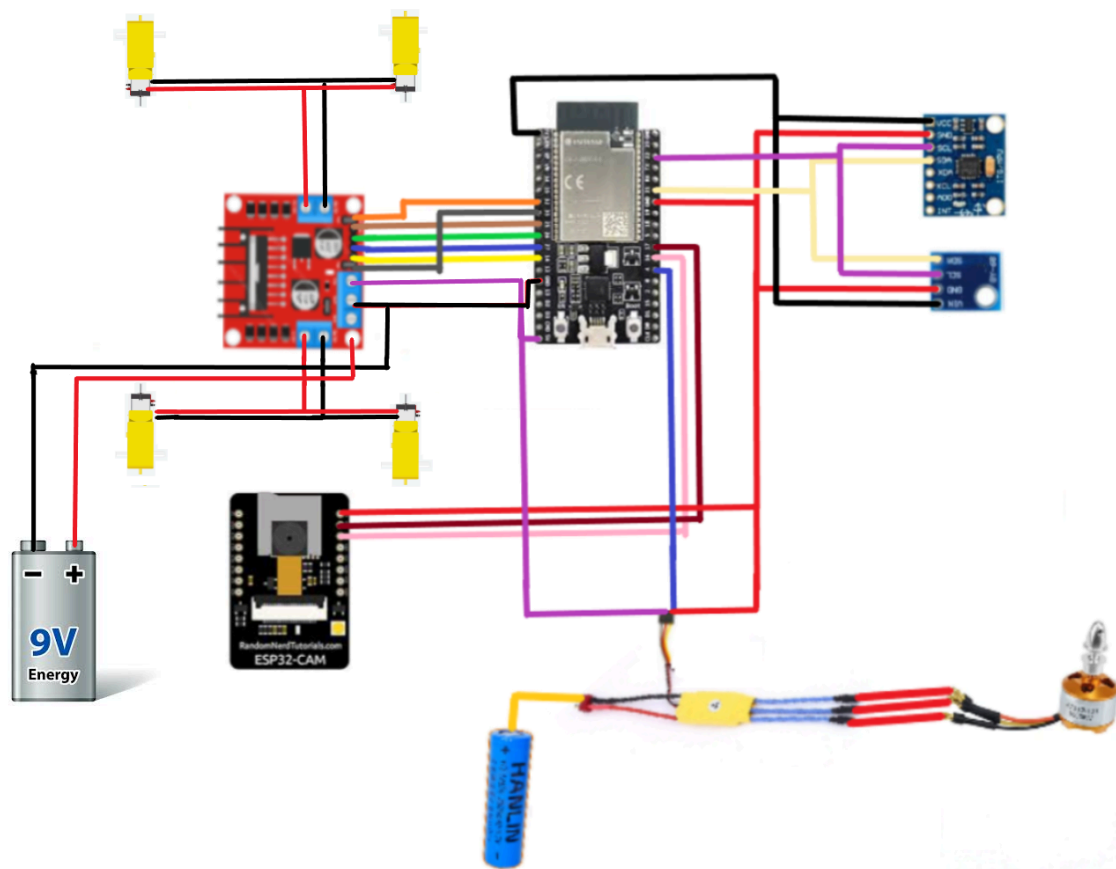
除了物理結構上的創新，我們更進一步整合ESP32-CAM模組進行即時影像擷取，並透過Wi-Fi進行影像回傳，使操作人員可遠端監看牆面狀況。更具創意的是，我們導入Python程式設計結合YOLO(You Only Look Once)深度學習模型訓練與應用，賦予爬牆車基礎的視覺辨識與判斷能力，能自動偵測牆面裂縫、表面剝落等潛在缺陷，提升作業的智慧化層次。

本作品不僅著眼於功能實現，也重視實際運作時的穩定性與能源效率。我們採用BMP180氣壓感測器即時監測風扇吸附區的氣壓變化，並搭配MPU6050加速度計與陀螺儀模組監控車體傾斜角度，根據環境變化自動調整風扇轉速，達

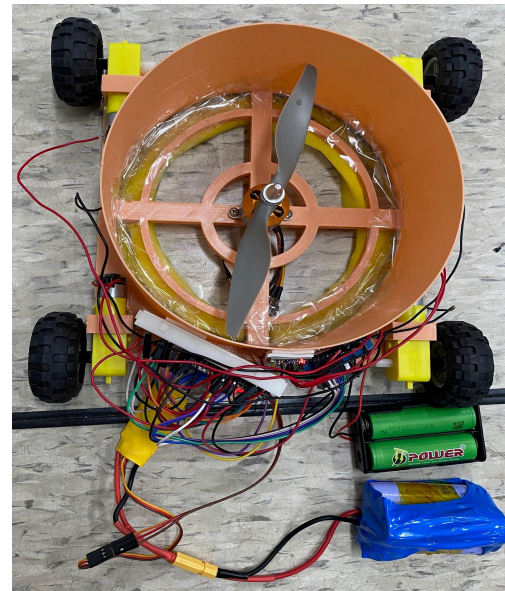
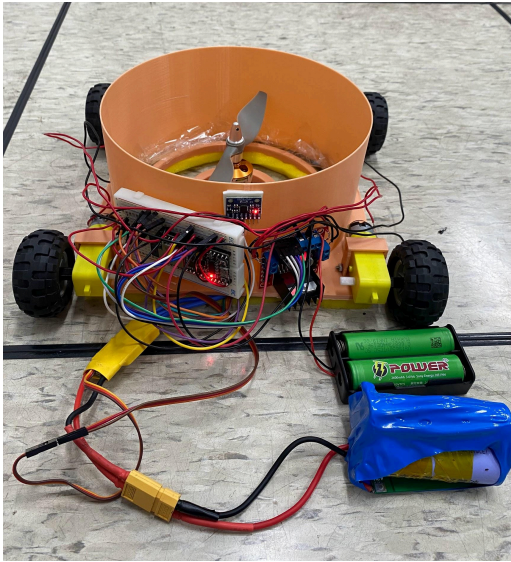
到最佳吸附效果與節能表現。在設計上實現了「主動式穩定控制」，大幅提升車體於牆面上的貼合性與續航力。

整體而言，這款爬牆車融合了機構創新、智慧辨識與動態控制技術，具備高度創意與實用潛力。未來可廣泛應用於高樓外牆檢測、危樓巡檢、水壩裂縫監控、救災搜索、城市清潔維護等多元場景，不僅提升作業效率，也可大幅降低人員進行高風險作業的需求，展現出科技創新服務社會安全的無限可能。

3. 硬體及電路架構圖：



圖一:硬體及電路架構圖 ▲

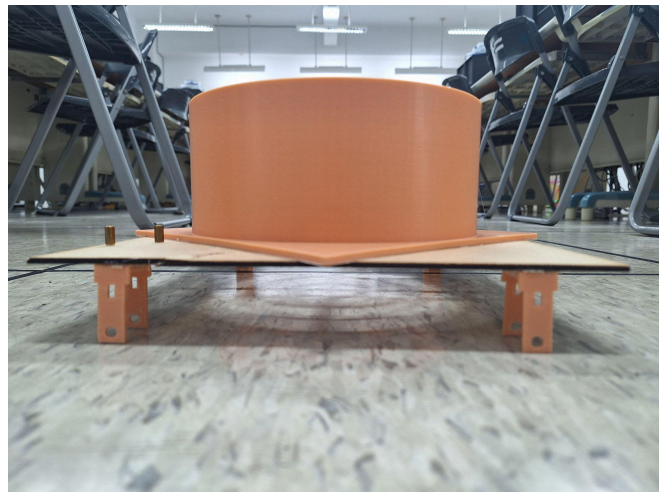


圖二、三:成品實際硬體及電路

4. 作品成果報告:

一.作品改良

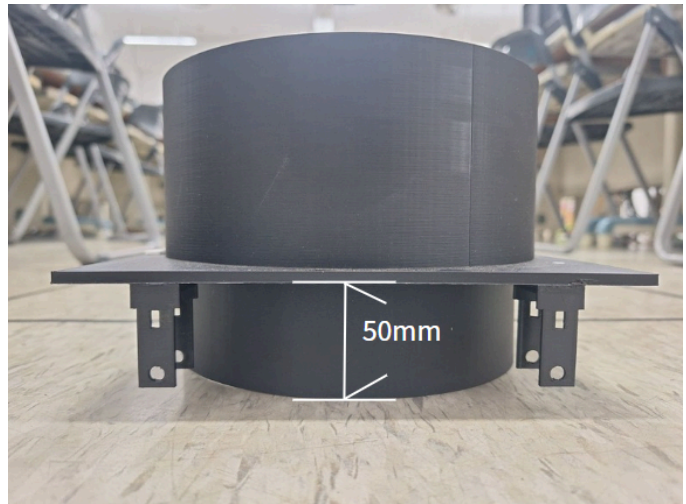
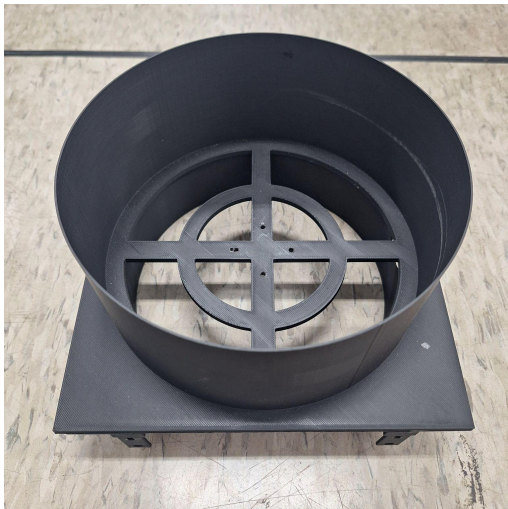
A.第一版爬牆車



圖四、五: 第一版爬牆車示意圖

我們在第一版爬牆車中，使用厚度為3mm木板當作車體底座，3D列印的環狀車殼厚度為6mm，但當風扇運轉時，車子朝牆壁方向的吸附力明顯不足，無法吸附在牆壁上。我們認為造成失敗的原因有二，一是車體重量過重，木板與過厚的環狀車殼皆增加了許多非必要的車體重量；二是車體底部需要類似密閉空間的閉環，才能形成一個壓力較小的空間，跟外界產生足夠的壓力差，將車子壓在牆壁上。

B.第二版爬牆車



圖六、七:第二版爬牆車示意圖

基於第一版的基礎，我們放棄了過重的木板，整個車體都改用3D列印材質，底板部分厚度為4mm，環狀車殼的厚度減少為1mm，讓車子重量大幅降低；並且在車底加裝閉環，當風扇將車體下方的空氣吸出時，使車底形成一個壓力較小的空間。經過以上改進，朝牆壁方向的吸附力雖然有顯著增加，但當我們將車子放置於牆壁上時，車子仍然會下滑，無法緊緊吸附於牆壁。並且我們發現一個新問題，當我們把風扇轉速開到大於7成時，馬達發生嚴重的震顫，風扇的葉片會撞擊到環狀車殼內部。

此次的失敗可歸咎於兩個原因，其一是車子仍會下滑，我們認為是底板與牆壁的距離過大(50mm)，導致力臂過長，使重力力矩變得太大，而風扇產生的吸附力不足以對抗重力力矩，其二是馬達因為震顫，無法開到最大值，我們認為此問題的原因是底板厚度太薄，導致強度不足，無法承受風扇產生的反作用力。

C.第三版爬牆車



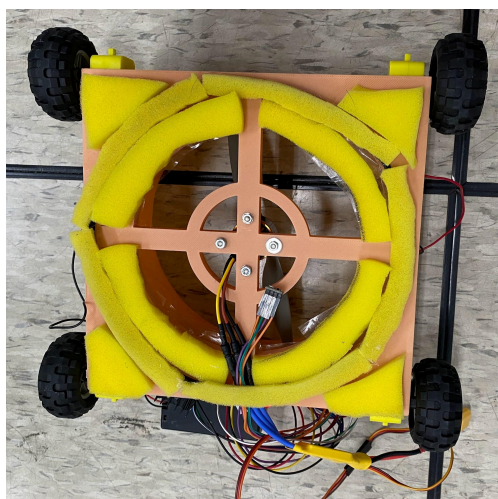
圖八、九:第三版爬牆車示意圖

經過前面的失敗，我們又進行了改良，將車體底板加厚為8mm，使其可以承受風扇全速運轉的反作用力；並縮小了底板與地面的距離(12mm)，使力臂減短，

重力力矩也因此變得更小。我們原先對以上改良充滿信心，一致覺得要成功了，但現實是殘酷的，底板加厚到8mm後，風扇全速運轉確實不會震顫，但當車體再次吸附於牆壁上時，還是會下滑，雖然跟第二版相比已較緩慢，但仍然無法完美吸住。

對於此次失敗的原因，我們認為是車體輪子與牆壁間的摩擦係數過小，因此我們決定增加摩擦係數更大的海綿於車底與牆壁間，幫助車子產生平行牆面的摩擦力，使其能夠對抗重力力矩，不再下滑。

D.第四版爬牆車



圖十:第四版爬牆車底部示意圖

綜合前三版爬牆車的失敗，我們這次在底板上加裝了海綿，最初，我們只在閉環的圓上貼了一圈海綿，此時車子幾乎已經可以完美吸附於牆壁上，但隨著吸附時間拉長，車體仍然以極為緩慢的速度下滑。因此我們便決定在底板其他處也貼上海綿，如此一來，爬牆車便能完美吸附在牆上了，且毫無下滑跡象。

二.ESP32CAM搭配YOLOv8模型

本成品以ESP32CAM作為影像擷取裝置，搭配電腦端端執行的YOLOv8模型，實現即時影像辨識功能。其運作流程可分為三個主要部分，說明如下：

A.影像擷取

ESP32CAM透過WiFi連線後，內部以ArduinoIDE開發並架設簡易的WebServer，提供影像串流服務。影像串流直接掛載於根網址，並以類似MJPEG的方式持續推送影像資料。這種設計無需使用額外影音串流協定，即能被電腦端的Python程式即時接收與解析。

```
// Wi-Fi connection
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

Serial.print("Camera Stream Ready! Go to: http://");
Serial.print(WiFi.localIP());

// Start streaming web server
startCameraServer();
```

圖十一:此程式會使ESP32CAM連上指定的WiFi網路，如果成功連線後才會繼續啟動串流伺服器。

```
void startCameraServer(){
  httpd_config_t config = HTTPD_DEFAULT_CONFIG();
  config.server_port = 80;

  httpd_uri_t index_uri = {
    .uri       = "/",
    .method    = HTTP_GET,
    .handler   = stream_handler,
    .user_ctx  = NULL
  };

  //Serial.printf("Starting web server on port: '%d'\n", config.server_port);
  if (httpd_start(&stream_httpd, &config) == ESP_OK) {
    httpd_register_uri_handler(stream_httpd, &index_uri);
  }
}
```

圖十二:這段程式啟動會HTTP WebServer，並將"/"路徑設定為影像串流入口，外部設備可透過該網址即可即時取得MJPEG畫面。

```
輸出  序列埠監控窗  X
訊息 (按 Enter 鍵將訊息發送到 COM4 上的 ESP32 Wrover Module)
-----
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:4888
load:0x40078000,len:16516
load:0x40080400,len:4
load:0x40080404,len:3476
entry 0x400805b4
-----
WiFi connected
Camera Stream Ready! Go to: http://172.20.10.6
```

圖十三:成功連接到指定的WiFi時，可藉由序列埠監控窗得到串流伺服器網址

B.影像辨識模型訓練

由於此成品目標是牆面裂痕的辨識，因此採用了YOLOv8模型作為訓練基礎，資料集(Wall Cracks)來源為

<https://universe.roboflow.com/object-detection-rrer5/wall-cracks/dataset/1>。

訓練過程包含資料預處理、數據增強、模型微調與驗證集評估，最終我們得到了準確率為82.2%的模型。而電腦端則使用Python撰寫影像辨識程式，負責持續向ESP32CAM所提供的網址發送請求，並接收JPEG格式的影像資料流。程式內透過"requests"套件以串流方式讀取影像資料，並解析每張影像的JPEG標記以取得完整圖片。

```
results = model.predict(frame, classes=[0], verbose=False)
annotated_frame = results[0].plot()
```

圖十四:此段程式會呼叫訓練好的YOLOv8模型，並指定僅偵測類別編號0(可自定)。

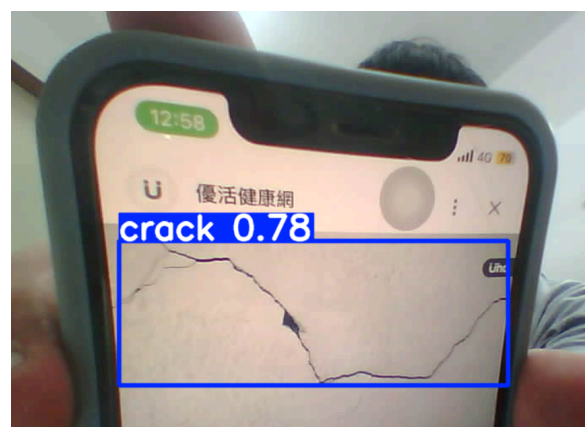
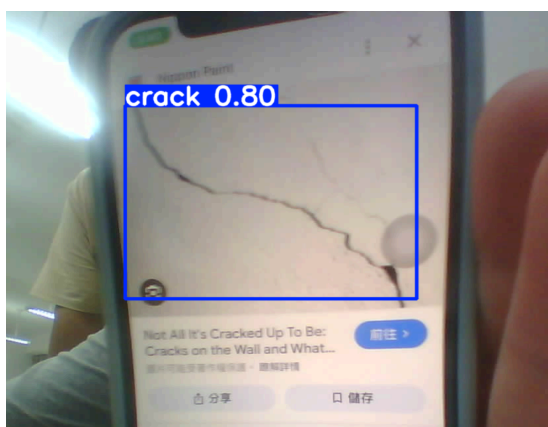
results[0].plot() 則將結果(例如邊框、標籤)畫在原始影像上。

```
cv2.imshow("YOLOv8 + ESP32-CAM", annotated_frame)
if cv2.waitKey(1) & 0xFF == ord('q'):
    break
```

圖十五:此段程式是使用OpenCV顯示每一幀偵測後的畫面，並持續進行直到按下q鍵結束。讓使用者能看到模型在影像中的推論效果。

C.訓練成果

辨識完成後，系統會將YOLOv8偵測結果進行標註(如:框選出模型視為裂縫的部分)，並以即時視窗方式回饋給使用者。若按下q鍵則可中斷程式並關閉視窗。此模組有助於開發階段的結果確認與模型效能調整。



圖十六、十七:模型辨識結果

三.實驗數據

馬達轉速與車子底部壓力的關係圖



縱軸:壓力(百帕)

橫軸:馬達轉速

備註:橫軸的40、50.....150這些數字, 是servo數值, 因為我們在程式中使用了servo函式庫, 40是它的初始值, 也就是轉速為0。

在最先開始的假設中, 我們就認為當馬達的轉速越大, 車子底部空間的壓力會因為空氣被吸出而變小。

而根據我們做出的實驗結果, 壓力也確實隨著馬達轉速的增加而遞減, 可以看到我們的圖表, 當馬達轉速為0時, 壓力為1006.5百帕; 當馬達轉速為100%時, 壓力約為1004.2百帕。

	壓力(百帕)
馬達0% (servo數值=40)	1006.5
馬達9% (servo數值=50)	1006.28
馬達18%(servo數值=60)	1006.13
馬達27%(servo數值=70)	1005.95
馬達36%(servo數值=80)	1005.71
馬達45%(servo數值=90)	1005.7
馬達54%(servo數值=100)	1005.56
馬達63%(servo數值=110)	1005.3
馬達72%(servo數值=120)	1005.04
馬達81%(servo數值=130)	1004.76
馬達90%(servo數值=140)	1004.7
馬達100% (servo數值=150)	1004.2

這成功證實了我們的假設, 馬達轉速最大與最小時, 壓力相差了2.3百帕, 而車底閉環的直徑為18cm, 根據公式, $F=P \times A$ 、 $A=\pi r^2$

其中

F 是總作用力(牛頓, N)

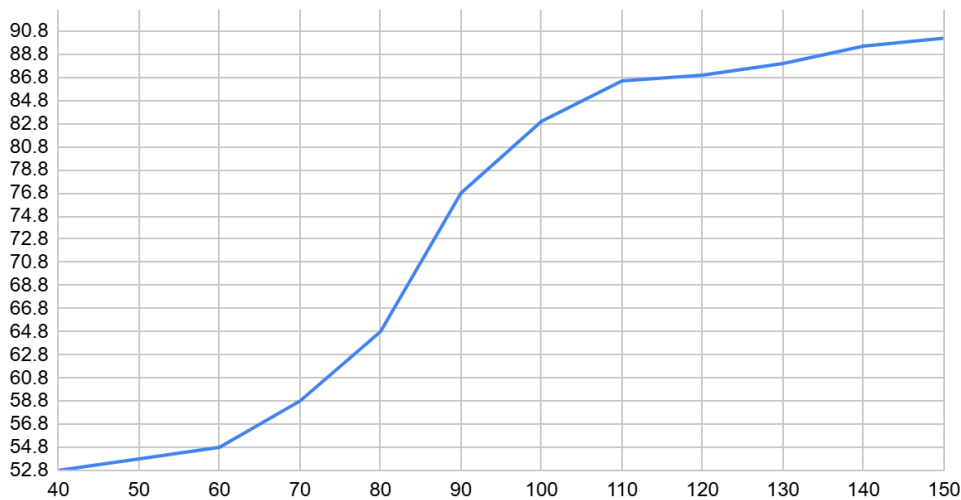
P 是壓力(帕, Pa)

A是面積(平方公尺, m^2)

$$A=\pi \times (0.18)^2=\pi \times (0.09)^2=\pi \times 0.0081 \approx 0.02545 m^2$$

$$F=230 \times 0.02545 \approx 5.854\text{N}$$

馬達轉速與可承受之最大傾角關係圖



縱軸:可承受的最大傾角

橫軸:馬達轉速

從我們的實驗結果可以看出，馬達轉速與可承受之最大傾角的關係是不成正比的。尤其是當傾角到了85度左右時，再繼續加大油門對可承受之傾角的影響已經很小了。反而是實驗的中段，馬達油門開到3成~5成時，可承受之傾角的變化很大很明顯。

四. 結論

本次作品其核心突破在於歷經四代設計後，對吸附行走機構的改進優化。初期遭遇的吸附力瓶頸，促使我們從材料科學(捨棄傳統木材，擁抱輕量化3D列印)到結構力學(優化壁環與底盤壓力分佈)進行了系統性改良。然而，真正的穩定吸附效能，是在毅然捨棄傳統壁環設計、轉而創新地採用海綿填充基座後才得以實現—此舉不僅顯著提升了摩擦係數，更關鍵地是達成了優異的氣密性，確保了風扇負壓得以穩定且集中地作用，從而牢固吸附於垂直牆面，克服了先前版本滑落與震動的頑疾。

本作品的顯著特色與主要創新貢獻，在於成功將此獨特的吸附式移動平台與一套多模態智慧感測系統進行了深度融合。藉由MPU6050提供的姿態感知、BMP180的大氣壓力監測，輔以ESP32CAM的即時影像串流，並結合尖端的YOLO深度學習模型進行即時裂縫辨識，本系統不僅能穩定巡弋於牆面，更具備了自主偵測結構瑕疵的潛力。儘管在模組整合過程中，我們積極應對並解決了諸如MPU6050在高動態下的數據跳動(透過硬體佈局優化與濾波演算法)、以及ESP32CAM初始化失敗與串流不穩(藉由精煉初始化邏輯與引入重試機制)等挑戰，最終實現了系統整體穩定性與反應速度的顯著提升。

最終我們不僅克服了爬牆機器人基礎移動的技術難點，更前瞻性地整合了AI視覺檢測能力，以低成本、高整合度的方案，打造出兼具實用性與技術前沿性的智慧型爬牆載具。其在建築安全檢測、災後評估等領域展現出的巨大應用潛力，證明了本研究在創新整合與解決實際工安問題上的價值。

5. 參考文獻：

1.勞動部職業安全衛生署(2024)。職安署力促公私合作，打擊營造墜落職災

<https://www.osha.gov.tw/48110/48417/48419/163403/post>

2.許乃文(2003)。RC伺服馬達應用於六足爬牆機器人之設計
[碩士論文，國立成功大學]。臺灣博碩士論文知識加值系統。

3.李鳳珠 林瑞興。伯努利原理在運動技術運用之探討
[屏東教大體育第十三期，國立屏東大學]。國立屏東大學網站