



AI人工機械手臂

學生：吳杰穎、楊博善
指導老師：鄭善仁 老師

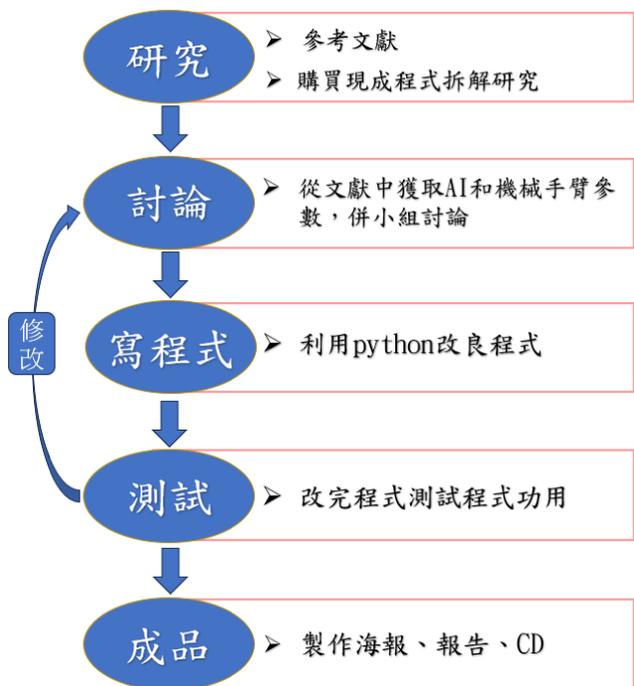
摘要

本研究旨在探討以人工智能 (AI) 技術為基礎的機械手臂系統。該系統將採用AI算法，使機械手臂具有更高的精確度和靈活性。此外，加深對AI人工機械手臂系統的理解，並探討其在工業、服務和其他領域的應用潛力。

研究目的

目的在探索以人工智能技術為基礎的機械手臂系統，以提升人類的製造能力和作業效率，並深入瞭解AI技術在機械手臂中的應用，並探討對人類生活和工作方式的影響，從而指導相關技術的發展和應用。

研究流程圖



影像控制程式

這程式是為了讓影像辨識和機械手臂結合而改寫的程式，主要目的是為了更深入了解如何改善機械手臂或是利用影像辨識來貼近生活應用。

```

ie = IECore()
model = ie.read_network(model="keras_model_serialized.xml")
compiled_model = ie.load_network(network=model, device="cpu")
input_key = next(iter(model.input_info))
output_key = next(iter(model.outputs))

# Load the class names
class_names = open("labels.txt", "r").readlines()

# Define the confidence threshold
confidence_threshold = 0.85

# Define a function for inference
def inference(image):
    # Preprocess the image for the model
    image = cv2.resize(image, (224, 224), interpolation=cv2.INTER_LINEAR)
    image = np.asarray(image, dtype=np.float32).reshape(1, 3, 224, 224)
    image = (image / 127.5) - 1

    # Make a prediction using the model
    prediction = compiled_model.infer(inputs=input_key, data=image)
    index = np.argmax(prediction[output_key])
    class_name = class_names[index]
    confidence_score = prediction[output_key][0]

    # Check if the confidence score is above the threshold
    if confidence_score > confidence_threshold:
        if index == 0:
            # 紅色的範圍
            # define range1 of Red color in HSV
            lower_red = np.array([150, 20, 30])
            upper_red = np.array([170, 235, 235])
            # Threshold the HSV image to get only Red
            mask0 = cv2.inRange(hsv, lower_red, upper_red)

            # define range2 of Red color in HSV
            lower_red = np.array([0, 20, 30])
            upper_red = np.array([20, 235, 235])
            # Threshold the HSV image to get only Red
            mask1 = cv2.inRange(hsv, lower_red, upper_red)
            mask2 = cv2.bitwise_or(mask1, mask0)

            # 藍色的範圍
            # define range of Blue color in HSV
            lower_blue = np.array([110, 20, 30])
            upper_blue = np.array([140, 235, 150])
            # Threshold the HSV image to get only blue
            mask21 = cv2.inRange(hsv, lower_blue, upper_blue)
            mask22 = cv2.bitwise_or(mask21, mask2)

            # 黃色的範圍
            # define range of Green color in HSV
            lower_yellow = np.array([25, 20, 30])
            upper_yellow = np.array([50, 235, 140])
            # Threshold the HSV image to get only yellow
            mask23 = cv2.inRange(hsv, lower_yellow, upper_yellow)
            mask24 = cv2.bitwise_or(mask23, mask2)

            # 手臂程式
            # arm.reset(wait=True)
            arm.set_vacuum_gripper(True, wait=False)
            arm.set_position(-49.5, 257.5, 252.0, -177.0)
            arm.set_position(-95, 318.2, 191.5, 179.3, -0.2)
            arm.set_position(121.9, 221.4, 245.5, 171.2, 3)
            arm.set_vacuum_gripper(True, wait=False)
            arm.set_vacuum_gripper(False, wait=False)
            arm.set_position(-124.1, 213.7, 100, 175.7, 1)
            arm.set_vacuum_gripper(True, wait=False)
            arm.set_position(-239.6, 212.4, 235.4, -174.7)
            arm.set_position(-232, 204, 183.8, 174.1, 2.5, 17)
            arm.set_vacuum_gripper(False, wait=False)
            arm.set_position(-239.6, 212.4, 235.4, -174.7)
            arm.set_position(-45.1, 328.5, 238, -167.7, 14)
            arm.reset(wait=True)
            elif index == 1:
  
```

圖1. 邊緣計算opencvino

圖2. bananapi

校外比賽驗證

這比賽主要為了驗證機械手臂在運動時，手臂所產生的運動原理和我們所理解的程式成對等，並能實際應用於生活中。

參賽作品：圖5

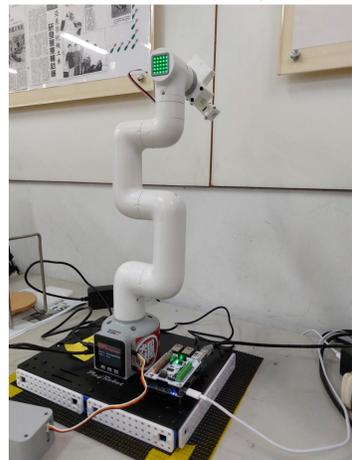


圖3 邊緣計算盒 圖4 機械大手臂

圖5 機械小手臂

最後比賽結束時，我們兩隊分別取得了金牌(圖6)和佳作(圖8)獎項，也讓我們更了解了機械手臂基礎運動原理，也了解到程式方面要如何去操控機械手臂。



圖6 獎狀

圖7 頒獎大合照

圖8 獎狀(2)

硬體介紹

邊緣計算盒是低功耗和高性能的硬體，而且部署成本和覆雜度也相對簡單快速，也配合openvino而開發出來的硬體，且不會有延遲問題。

AI影像辨識結合介紹

利用訓練完的模型結合openvino程式圖1來讓機器學習，就可以利用AI影像辨識結合機械大手臂圖4探索更多不同應用，也可以完成類似工業自動化。

專題研究成果與未來展望

此專題研究通過探索以機械學習為基礎的機械手臂系統，取得了一系列成果。未來，我們期待這項技術能在工業自動化、服務機器人等領域得到廣泛應用，並為人類生活帶來更多便利。



影像控制影片



手臂比賽影片