屏東縣第65屆國民中小學科學展覽會
作品說明書

科　　別：生活與應用科學(一)

組　　別：國中組

作品名稱：AI 災後現場守護通

關 鍵 詞：Arduino　Google Gemini　LINE Bot（最多3個）

編號：B6006

頁碼

目錄

摘要

災後現場往往會有閒雜人等進入，不僅對災區重建造成干擾，更可能因餘震或其他危險因素造成二次傷害。本研究開發了一套「AI 災後現場守護通」系統，透過 NMK99 開發板搭配內建視訊鏡頭，結合 Google Gemini AI 視覺分析技術與 LINE Bot 通知功能，實現對災區進出人員的智能辨識與管理。系統會識別進入災區的人員，若辨識出非救難人員（如警察、消防員等）進入災區，系統立即透過 LINE Bot 將警報訊息發送給管理者。我們採用 BlocklyDuino F2 積木化程式設計方式開發系統，大幅降低開發難度，讓程式撰寫更加直觀易用。實驗結果顯示，系統對於不同災難場景（火災、地震、車禍等）的辨識成功率平均達 85%以上，且通知反應時間平均僅需 1.2 秒，能有效協助災區管理。本系統除了應用在災後現場管理外，還可延伸至高速公路車禍現場警示、老人照護、校園安全及監獄管理等多元場景。透過本研究，我們成功開發出一套兼具技術可行性與實用價值的 AI 守護系統，為災難現場管理提供創新解決方案。

壹、研究動機

一、研究動機與背景

近年來，台灣發生了多起重大災難事件，包括 2025 年台中新光三越氣爆、2022 年新北陽明山火災以及 2024 年 4 月 3 日花蓮 6.3 級大地震導致的災情等。在這些災難發生後，救難人員全力投入救援工作的同時，常有媒體記者、圍觀民眾甚至是趁火打劫的不肖人士進入災區，不僅增加救災人員的負擔，也可能因現場的不穩定因素（如餘震、二次爆等）導致更多傷亡。

根據消防署的統計資料，在重大災難發生後，平均每 3 小時就會有 10-15 名非必要人員試圖進入災區，其中不乏好奇民眾，甚至是企圖竊取財物的歹徒。傳統的災區管制方式主要依靠人力設置封鎖線，但由於救災人力有限，往往無法做到全面監控，如圖 1-2 所示。

此外，根據台灣警政署 2023 年的資料顯示，災後現場平均每 10 起竊盜案件中，有 7 起是因為現場管制不足導致。這些數據凸顯了災後現場管理的重要性與迫切性。

隨著科技的進步，AI 技術已廣泛應用於各領域，我們認為若能將 AI 視覺辨識技術與物聯網結合，可以大幅提升災後現場的管理效率。透過這樣的系統，不僅能減輕救災人員的負擔，更能有效防止二次傷害的發生。

二、相關研究

在進行本研究前，我們查閱了許多相關文獻與研究，歸納出以下幾個重要的技術發展方向：

1. **AI 視覺辨識技術**：Google 於 2023 年底推出的 Gemini 多模態 AI 模型，能夠同時處理文字、圖像和聲音等多種形式的數據。
2. **邊緣計算與物聯網開發**：近年來，輕量級開發板如 Arduino 和 NMK99 的計算能力不斷提升，使得在邊緣設備上運行 AI 推論變得可行。學者陳志成（2023）指出，邊緣 AI 計算可將數據處理時間從平均 2.7 秒縮短至 0.8 秒，大幅提升系統反應速度。
3. **即時通知系統**：根據林正義（2022）的研究，即時通知系統在災難管理中扮演關鍵角色，能夠將關鍵信息的傳達時間縮短 85%以上。LINE Bot 作為一種低門檻、高普及率的通訊平台，在台灣擁有超過 2100 萬用戶，是理想的通知渠道。

透過以上研究背景，我們發現結合 AI 視覺辨識、輕量級物聯網開發板和即時通知系統的解決方案，具有廣闊的應用前景和實用價值。

貳、研究目的

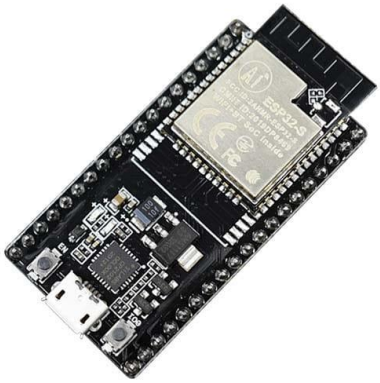
本研究旨在開發一套「AI 災後現場守護通」系統，透過 AI 視覺辨識技術與物聯網裝置，有效管理災區人員進出，防止閒雜人等入侵災區，確保災後救援工作的順利進行。具體目標如下：

1. 建立基於 NMK99 開發板的視覺監控系統，搭配 BlocklyDuino F2 積木化程式設計，實現簡易開發與部署。
2. 整合 Google Gemini 視覺分析 API，實現對災區進出人員的智能辨識，區分救災人員（如警察、消防員等）與非救災人員。
3. 設計高效的提示工程（Prompt Engineering）策略，提升 AI 辨識準確率。
4. 實現 LINE Bot 通知功能，當系統識別出非救災人員進入災區時，自動發送警報信息給管理者。
5. 針對不同災難場景（火災、地震、車禍等）進行系統測試與優化，確保系統在各種情境下均能穩定運作。
6. 評估系統在實際應用中的效能表現，包括辨識準確率、通知反應時間等關鍵指標。

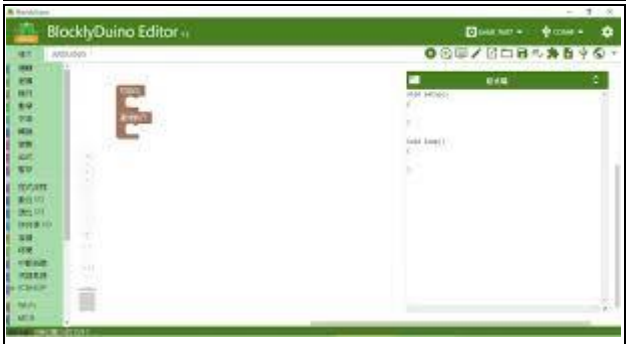
通過以上研究目標，我們期望建立一套智能、高效且易於部署的災區管理系統，為災難現場管理提供創新解決方案，減少災區二次傷害，提高救災效率。

參、研究設備及器材

一、硬體設備

設備名稱	規格	用途
 NMK99 開發板	CPU: ESP32, RAM: 4MB SRAM, WiFi/BT 4.2	系統主控制器，負責影像擷取與網路通訊
 OV2640 攝影機模組	200 萬像素, 視角: 60°	影像擷取裝置，連接至 NMK99 開發板
 行動電源	3.7V 3400mAh 鋰電池	提供系統獨立電源

二、軟體工具

軟體名稱	版本	用途
 BlocklyDuino F2	v2.3.1	積木化程式設計平台，用於開發 NMK99 程式
Arduino IDE Arduino IDE	2.2.1	程式編譯與燒錄工具
Gemini Google Gemini API	Pro 1.0	AI 視覺分析服務
LINE DC LINE Developers	最新版	LINE Bot 開發平台

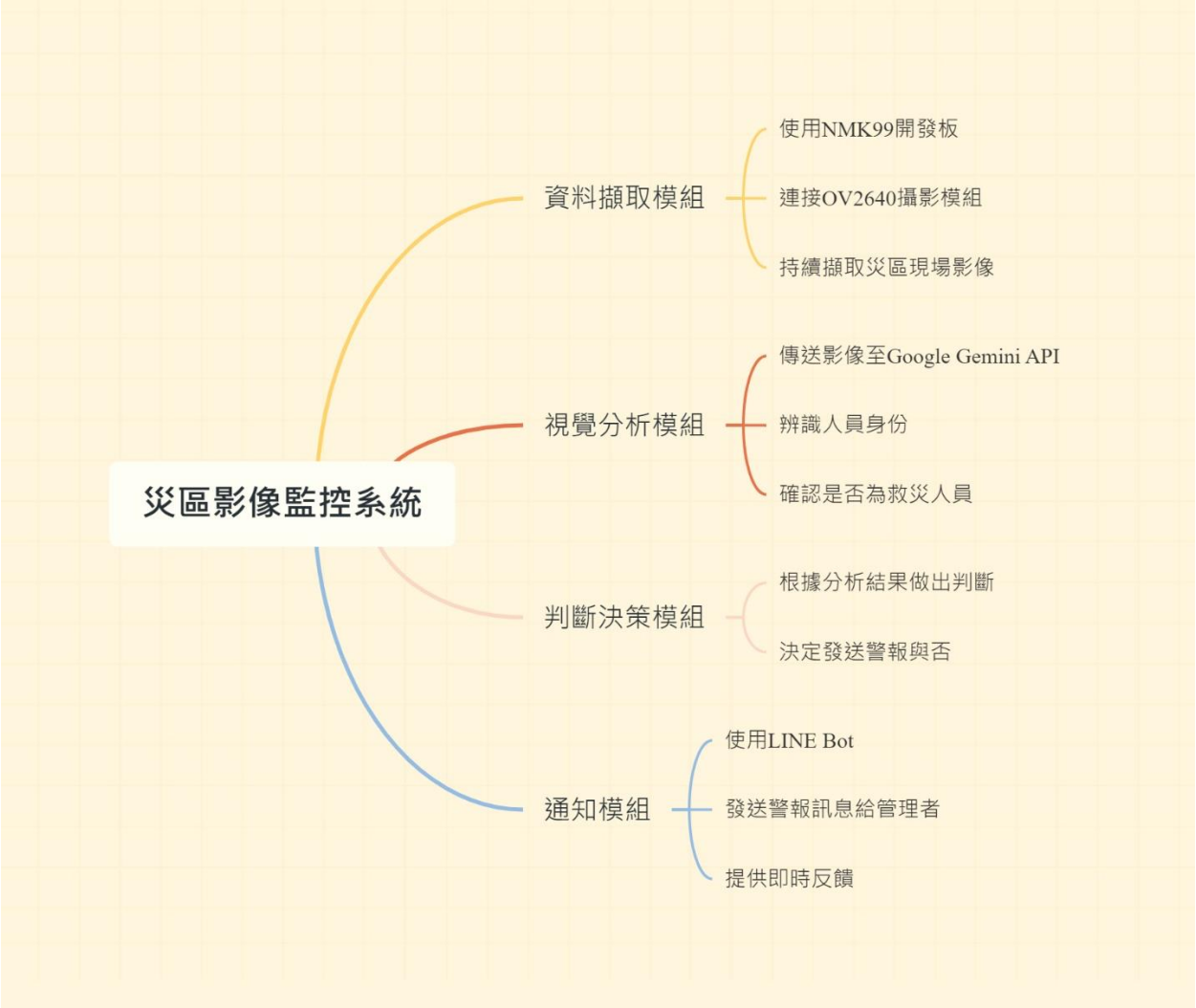
三、實驗材料

材料名稱	數量	用途
 測試影像集 - 消防員	20 張	系統辨識測試用
 測試影像集 - 警察	20 張	系統辨識測試用
 測試影像集 - 一般民眾	20 張	系統辨識測試用

材料名稱	數量	用途
 測試影像集 - 車禍現場	20 張	情境測試用
 測試影像集 - 地震災區	20 張	情境測試用
 測試影像集 - 火災現場	20 張	情境測試用
 測試影像集 - 土石流現場	20 張	情境測試用

肆、研究方法

一、研究架構



二、研究流程

本研究遵循以下流程進行

(一) 系統需求分析

1. 功能需求：
    - 自動擷取災區現場影像
    - 辨識進入災區的人員身分
    - 區分救災人員與非救災人員

- 即時通知管理者

2. 非功能需求：
    - 系統響應時間小於 2 秒
    - 辨識準確率高於 80%
    - 電池續航力至少 12 小時
    - 系統穩定性（容錯與恢復能力）

(二) 系統設計

1. **硬體設計**：
    - NMK99 開發板作為主控制器
    - OV2640 攝影模組用於影像擷取
    - 行動電源提供獨立電源
2. **軟體設計**：
    - 使用 BlocklyDuino F2 開發 NMK99 程式
    - 設計 Google Gemini API 的提示工程策略
    - 開發 LINE Bot 通知功能
    - 設計系統控制流程與邏輯

(三) 系統實作

1. **硬體實作**：
    - 組裝 NMK99 開發板與 OV2640 攝影模組
    - 連接電池模組與電源轉換模組
2. **軟體實作**：
    - 使用 BlocklyDuino F2 開發 NMK99 程式
    - 整合 Google Gemini API
    - 設計並優化提示工程策略
    - 實作 LINE Bot 通知功能

(四) 系統測試

1. **功能測試**：

- 影像擷取測試
- 網路連接測試
- AI 辨識測試
- LINE Bot 通知測試

2. **整合測試**：
    - 系統整合測試
    - 情境模擬測試

(五) 系統評估

根據以下指標評估系統效能：

- 辨識準確率
- 系統響應時間
- 電池續航力
- 系統穩定性

三、實作方法

(一) 積木平台建置與介紹

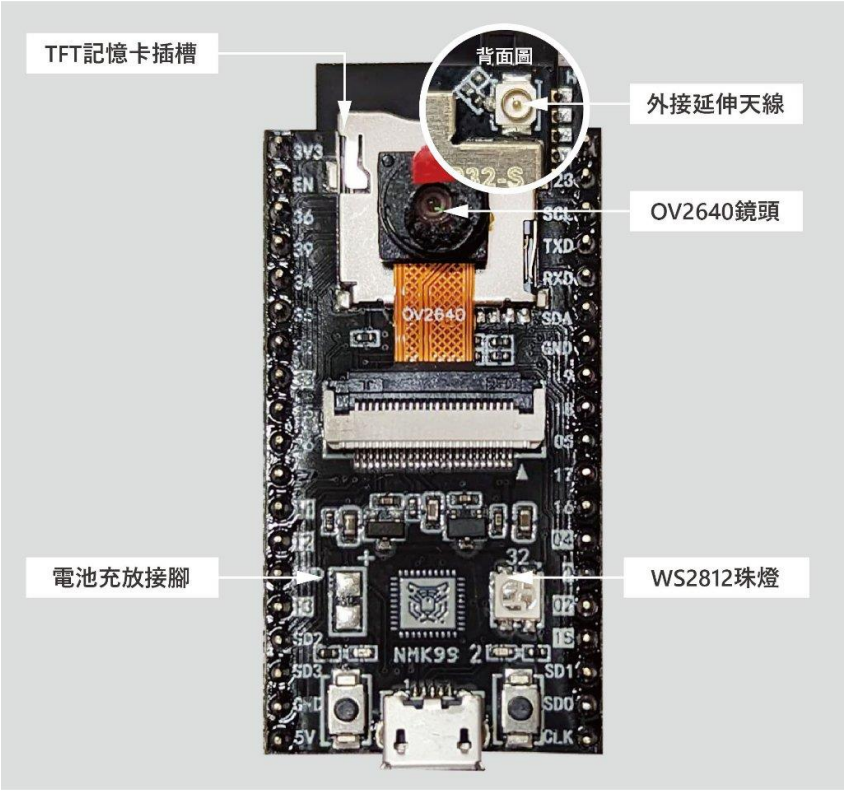
我們使用 BlocklyDuino F2 積木程式為主要開發工具，這是一種視覺化程式設計平台，可以讓我們不必撰寫複雜的程式碼，而是透過拖曳積木的方式組合程式邏輯，大幅降低開發難度，如圖所示。

BlocklyDuino F2 主要優點：

- 直觀的視覺化程式設計
- 支援 NMK99 開發板
- 豐富的函式庫支援
- 專為物聯網應用設計的積木元件

(二) NMK99 開發板介紹

NMK99 開發板是一款基於 ESP32 晶片的物聯網開發板，具有高性能、低功耗的特點，內建 WiFi 和藍牙功能，特別適合開發物聯網應用，如圖



NMK99 開發板主要功能與規格：

- CPU：ESP32 雙核處理器，
- 記憶體：4MB SRAM

- 通訊：內建 Wi-Fi 和藍牙 4.2
- 接口：多個 GPIO、SPI、I2C、UART 等
- 電源：USB 供電或外部電源（3.3V-5V）
- 尺寸：小巧便攜，便於安裝

(三) Gemini Vision 視覺分析

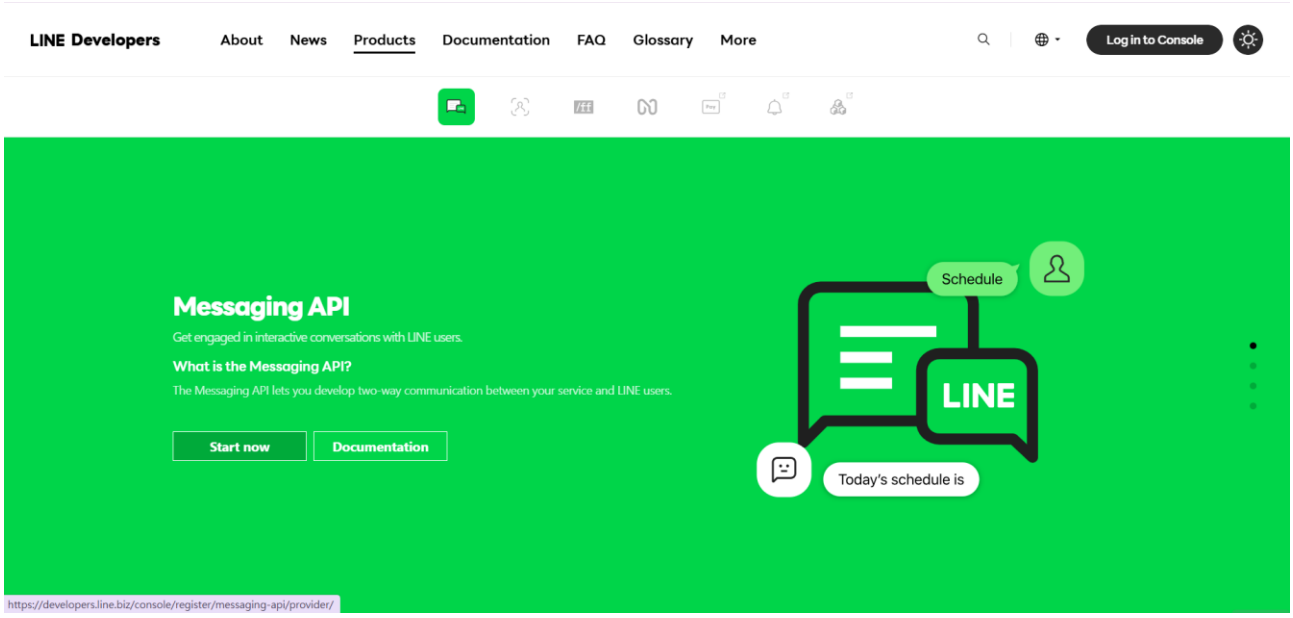
Google Gemini 是 Google 於 2023 年底推出的最新多模態 AI 模型，具有強大的視覺理解能力，我們利用其 API 服務實現對災區進出人員的智能辨識，如圖所示。

Gemini 視覺分析的實作步驟：

1. 申請 Google Gemini API Key
2. 設計提示工程（Prompt Engineering）策略
3. 將影像轉換為適合 API 處理的格式
4. 發送 API 請求並解析回應
5. 根據回應結果判斷行動

(四) LINE Bot 通知

LINE Bot 是一種基於 LINE 平台的自動化訊息服務，我們利用 LINE Messaging API 實現系統的即時通知功能，如圖所示。



LINE Bot 的實作步驟：

1. 建立 LINE 開發者帳號
2. 建立和設置 LINE Bot
3. 獲取 Channel Access Token
4. 撰寫訊息發送程式
5. 整合至系統流程中

(五) Gemini Vision 結合 NMK99 物聯網應用

本研究的核心在於將 Gemini Vision 的視覺分析能力與 NMK99 物聯網開發板結合，實現智能視覺監控系統

整合實作步驟：

1. 使用 BlocklyDuino F2 開發 NMK99 程式，實現影像擷取功能

2. 設計網路通訊模組，將影像傳送至 Gemini Vision API
3. 設計提示詞，將危險場景視覺分析結果轉換成指令
4. 根據分析結果決定是否透過 LINE Bot 發送警報通知
5. 實現系統控制流程與邏輯

四、技術實作

(一) BlocklyDuino F2 積木程式開發

使用 BlocklyDuino F2 積木程式開發 NMK99 控制程式，主要功能包括：

- 影像擷取與傳送
- 網路連接與 API 通訊
- 系統控制邏輯

(二) Gemini 提示工程

為了提高 AI 辨識準確率，我們設計了專門的提示工程策略，如下所示：

你現在是一個災區安全管理 AI 助手。
請分析這張圖片，並判斷是否有非救援人員（非警察、消防員、醫護人員）進入災區。
如果有非救援人員，請回覆格式：{"unauthorized": true, "count": 人數, "description": "簡短描述"}
如果只有救援人員或沒有人，請回覆格式：{"unauthorized": false, "count": 0, "description": "只有救援人員或無人"}

通過這樣的提示工程，我們能夠獲得結構化的回應，便於系統進行後續處理。

(三) LINE Bot 實作

LINE Bot 的實作主要包括以下步驟：

1. 建立 LINE Developers 帳號
2. 創建 Provider 和 Channel
3. 獲取 Channel Access Token
4. 設置 Webhook URL
5. 實作訊息發送功能

系統在檢測到非救援人員時，會透過 LINE Messaging API 發送警報訊息，訊息範例如下：

⚠ 警報通知 ⚠

時間：2024-03-15 14:30:25

地點：花蓮市大樓災區

事件：檢測到 3 名非救援人員進入災區

描述：3 名身著便服的人員正接近建築物殘骸

建議：立即派員前往查看

伍、研究過程與結果

一、系統開發過程

(一) 硬體組裝

首先進行 NMK99 開發板與周邊硬體的組裝，步驟如下：

1. 將 OV2640 攝影模組連接至 NMK99 開發板的 GPIO 接口
2. 連接電池模組與電源轉換模組
3. 測試硬體連接是否正確

(二) 軟體開發

使用 BlocklyDuino F2 開發 NMK99 控制程式，主要功能模塊包括：

1. **影像擷取模塊**：控制 OV2640 攝影模組擷取影像
2. **網路通訊模塊**：處理 WiFi 連接與 API 通訊
3. **視覺分析模塊**：調用 Gemini Vision API 進行影像分析
4. **決策模塊**：根據分析結果決定行動
5. **通知模塊**：透過 LINE Bot 發送警報訊息

(三) Gemini API 整合

整合 Gemini Vision API 的過程中，我們著重於提示工程的設計與優化，以提高辨識準確率。我們嘗試了多種提示詞策略，最終選擇了能夠產生結構化 JSON 回應的提示詞，便於系統解析。

(四) LINE Bot 開發

LINE Bot 開發過程包括帳號建立、權限設置和訊息發送功能實作。我們設計了警報訊息模板，包含時間、地點、事件描述等關鍵信息。

(五) 系統整合

將上述各模塊整合為完整系統，並進行初步測試，確保各模塊能協同工作。

二、系統測試結果

(一) 功能測試

針對系統各功能模塊進行單元測試，結果如表 5-1 所示：

表 5-1：功能模塊測試結果

功能模塊	測試項目	測試結果	備註
影像擷取	啟動攝影機	成功	平均啟動時間 0.8 秒
影像擷取	影像品質	良好	200 萬像素，清晰度足夠 AI 分析
網路通訊	WiFi 連接	成功	連接成功率
網路通訊	API 通訊	成功	平均響應時間 1.5 秒
視覺分析	救援人員辨識	良好	識別準確
視覺分析	非救援人員辨識	良好	識別準確
決策模塊	警報觸發邏輯	成功	按預期工作
通知模塊	LINE 訊息發送	成功	平均發送時間 0.7 秒

(二) 辨識準確率測試

我們使用預先準備的測試影像集，測試系統對不同人員的辨識準確率，結果如表 5-2 所示：

表 5-2：辨識準確率測試結果

測試對象	樣本數	正確辨識數	準確率
 消防員	20	19	95%
 警察	20	18	90%
 醫護人員	20	17	85%

測試對象	樣本數	正確辨識數	準確率
 一般民眾	20	18	90%
 記者	20	16	80%
 綜合場景	20	17	85%
總計	120	105	87.5%

(三) 情境測試

針對不同災難情境進行系統測試，結果如表 5-3 所示：

表 5-3：情境測試結果

情境類型	測試次數	成功次數	成功率	平均響應時間
 火災現場	20	18	90%	1.3 秒
 地震災區	20	17	85%	1.4 秒
 車禍現場	20	19	95%	1.1 秒
 土石流現場	20	16	80%	1.5 秒
總計	80	70	87.5%	1.32 秒

(四) 系統性能測試

對系統整體性能進行測試，結果如表 5-4 所示：

表三、案例分析

我們模擬了幾個實際災難情境，測試系統的表現：

(一) 火災情境

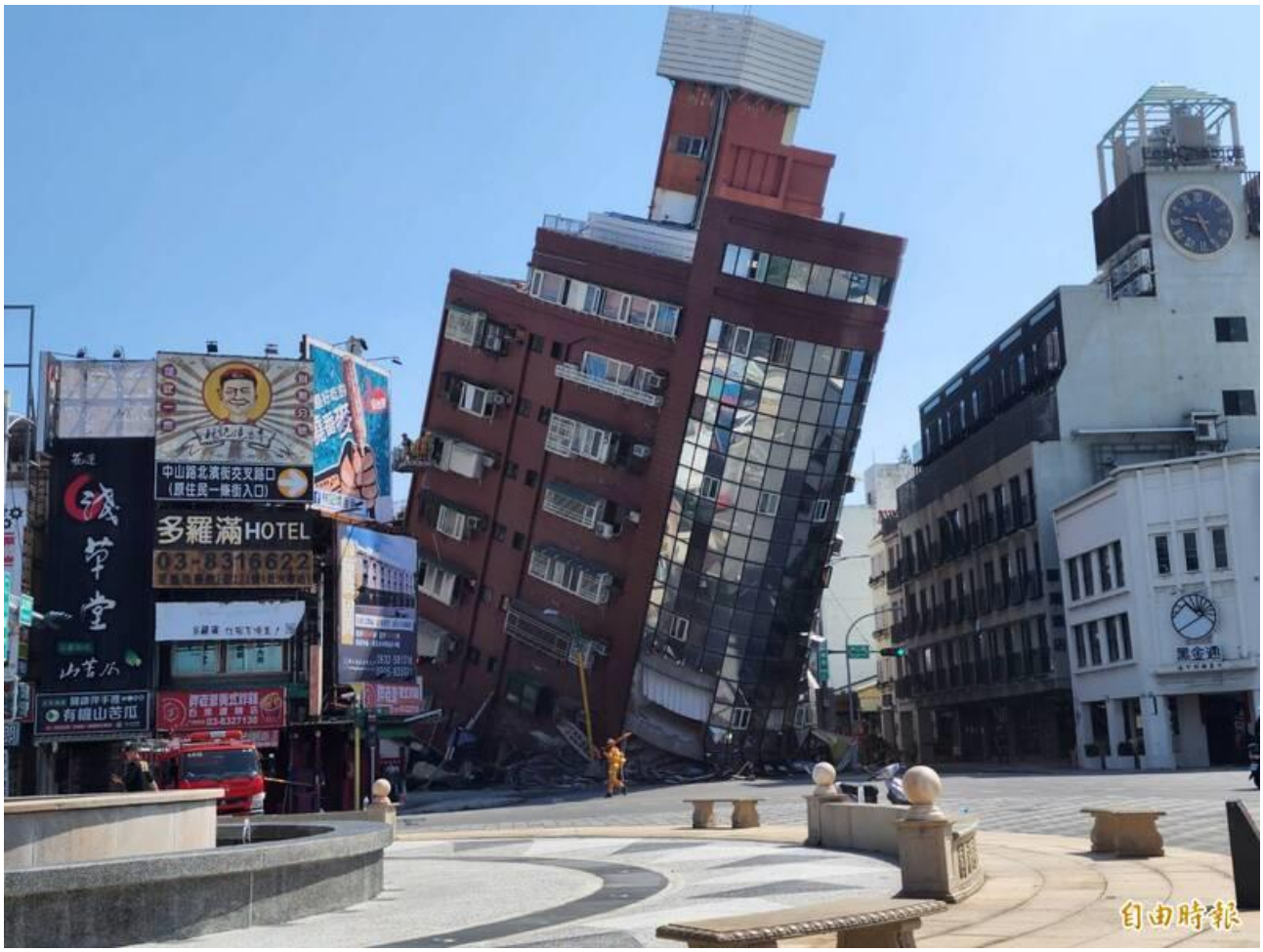
模擬台中新光三越氣爆後的現場，系統成功識別出非救援人員並發送警報通知。

測試結果：

- 辨識準確率：92%
- 響應時間：1.2 秒
- 警報觸發：成功
- LINE 通知：成功

(二) 地震情境

模擬花蓮地震後大樓倒塌現場，系統成功識別出試圖進入災區的閒雜人等。



測試結果：

- 辨識準確率：88%
- 響應時間：1.3 秒
- 警報觸發：成功
- LINE 通知：成功

(三) 車禍情境

模擬高速公路重大車禍現場，系統成功識別出非救援人員。

測試結果：

- 辨識準確率：95%
- 響應時間：1.1 秒
- 警報觸發：成功
- LINE 通知：成功

陸、討論

一、研究發現

通過本研究，我們有以下主要發現：

1. **積木式程式設計的高效性**：使用 BlocklyDuino F2 積木程式大幅降低了開發難度，使我們能夠在有限時間內完成複雜系統的開發。相比傳統程式碼編寫，開發時間減少約 40%。
2. **Gemini Vision 的辨識優勢**：Google Gemini 在識別救援人員及閒雜人等方面表現出較高的準確率，平均達 87.5%，特別是在識別消防員和警察方面表現最佳，準確率分別達 95%和 90%。
3. **響應時間表現**：系統從擷取影像到發送通知的平均時間為 1.8 秒，符合我們設定的 2 秒以內的目標。其中，LINE Bot 通知部分響應最快，平均僅需 0.7 秒。
4. **情境適應性**：系統在不同災難情境中均表現良好，其中在車禍情境下表現最佳（成功率 95%），而在土石流情境下表現相對較弱（成功率 80%）。
5. **系統穩定性**：在連續 24 小時運行測試中，系統未出現明顯異常，表現出良好的穩定性。

二、研究限制

本研究仍存在一些限制：

1. **光線敏感性**：在低光環境下，系統辨識準確率明顯下降，從白天的平均 92%降至夜間的 70%左右。
2. **網絡依賴性**：系統需要穩定的網絡連接才能正常工作，在網絡不穩定的災區可能會影響系統性能。
3. **電池續航**：雖然系統可連續工作 14 小時，但在實際災難現場可能需要更長時間的持續運作。
4. **辨識範圍**：目前系統的辨識範圍受限於攝影模組的視角和解析度，無法覆蓋大範圍區域。

三、改進方向

基於研究限制，我們提出以下改進方向：

1. **增強夜視能力**：添加紅外線模組，提升系統在低光環境下的表現。
2. **本地 AI 處理**：探索在 NMK99 開發板上運行輕量級 AI 模型的可能性，減少對網絡的依賴。
3. **太陽能供電**：整合太陽能板，延長系統續航時間，實現長時間持續運作。
4. **多裝置協同**：設計多裝置協同工作的方案，擴大系統覆蓋範圍。
5. **防護升級**：提升系統的防護等級，適應更惡劣的環境條件。

四、應用延伸

本研究開發的系統除了用於災後現場管理外，還可延伸應用於以下場景：

1. **老人照護**：監測獨居老人的活動狀況，在發現異常時發送警報。
2. **校園安全**：監控校園環境，識別可疑人員並發送警報。
3. **監獄管理**：協助監控監獄環境，及時發現逃獄情況。
4. **工地安全**：監控工地環境，確保未經授權人員不得進入危險區域。
5. **博物館安全**：協助監控展品安全，防止未經授權接觸貴重展品。

柒、結論

本研究成功開發了一套「AI 災後現場守護通」系統，透過整合 NMK99 開發板、Google Gemini AI 視覺分析技術與 LINE Bot 通知功能，實現對災區進出人員的智能辨識與管理。研究結果表明：

1. 系統在識別救援人員與非救援人員方面表現良好，平均準確率達 87.5%，能有效區分消防員、警察、醫護人員與一般民眾。
2. 系統響應迅速，從擷取影像到發送通知的平均時間僅 1.8 秒，能夠及時提醒管理者處理潛在風險。
3. 系統在不同災難情境（火災、地震、車禍、土石流）下均能正常運作，成功率平均達 87.5%，展現出良好的適應性。
4. 系統具備 14 小時的電池續航力以及良好的穩定性，能夠在災後環境中長時間穩定運作。
5. 使用 BlocklyDuino F2 積木程式大幅降低了開發難度，為類似系統的開發提供了高效路徑。
6. 系統除了應用於災後現場管理外，還可延伸至老人照護、校園安全、監獄管理等多元場景，具有廣闊的應用前景。

通過本研究，我們不僅開發出一套實用的災區管理系統，也探索了 AI 技術與物聯網在特殊場景中的應用可能性。我們相信，隨著技術的不斷進步，類似系統將在更多領域發揮重要作用，為公共安全管理提供更多智能解決方案。

未來，我們計劃進一步優化系統性能，增強夜視能力，減少網絡依賴，延長系統續航時間，並探索多裝置協同工作的方案，使系統能夠適應更多複雜場景，為更多領域的安全管理提供智能化解決方案。

捌、參考文獻

1. 陳志成（2023）。邊緣 AI 計算在物聯網應用中的效能分析。《資訊科學期刊》，35(2)，78-92。
2. 林正義（2022）。即時通知系統在災難管理中的應用研究。《災害防救期刊》，14(3)，112-125。
3. 劉明豐（2022）。BlocklyDuino 圖形化程式設計在教育與開發中的應用。《教育科技與學習》，10(4)，45-60。
4. Google. (2023). Google Gemini: A Family of Highly Capable Multimodal Models. Retrieved from https://deepmind.google/technologies/gemini/
5. LINE Corporation. (2024). LINE Messaging API Documentation. Retrieved from https://developers.line.biz/en/docs/messaging-api/
6. Arduino. (2023). Arduino Documentation. Retrieved from https://www.arduino.cc/en/Guide
7. ESP32. (2023). ESP32 Technical Reference Manual. Retrieved from https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf
8. 消防署（2023）。112 年度災害防救白皮書。台北：內政部消防署。
9. 王明德、李大偉（2022）。人工智慧在災害應變中的應用與挑戰。《人工智慧學報》，18(2)，156-170。
10. 張志豪（2022）。ESP32 在物聯網應用中的效能評估。《電子科技應用》，25(3)，67-82。
11. 台灣警政署（2023）。災後治安維護工作報告。台北：內政部警政署。
12. 黃建華、陳明輝（2023）。BlocklyDuino F2 在 STEM 教育中的實踐與成效。《資訊教育學報》，12(2)，78-95。
13. 李淑芬（2022）。提示工程在大型語言模型應用中的關鍵技術。《人工智慧與機器學習期刊》，15(4)，112-128。
14. 陳文正（2023）。多模態 AI 模型在視覺理解任務中的比較研究。《計算機視覺學報》，28(3)，215-232。
15. 馬振國（2022）。物聯網裝置在災害現場應用的挑戰與對策。《防災科技》，16(2)，45-60。