

# Argo 通用 AI 整合文件 version 1.2

## 一、實作架構:

### 二、溝通API

1. 設定演算法參數 (Argo → 辨識.exe)
2. 程式通道Alive(Argo → 辨識.exe)

3. 事件參數(辨識.exe → Argo)

### 三、設定詳細內容(json)

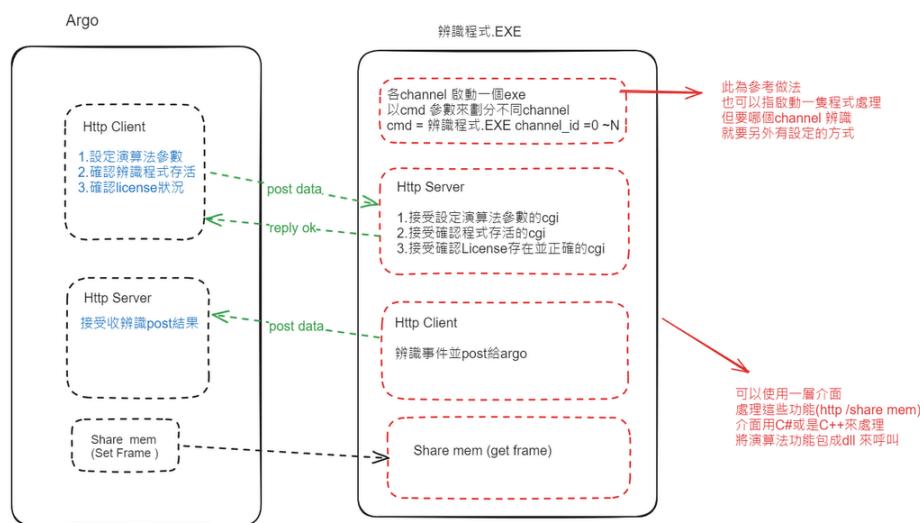
1. 設定演算法參數 (支援最多10組ROI 和每個ROI 有自己的設定值)

2. 辨識事件結果

### 四、Shared mem 設定

## 一、實作架構:

參考下圖說明



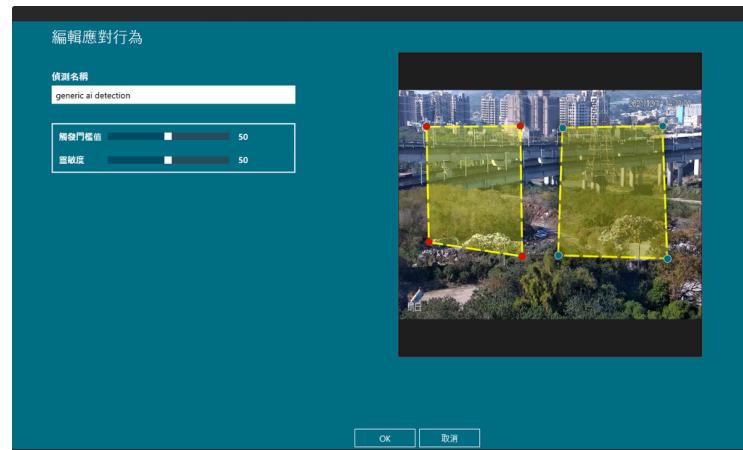
Argo 可以上傳指定的辨識程式.exe ,經由Zip 檔上傳,並產生一組AI裝置(通用AI模組偵測)



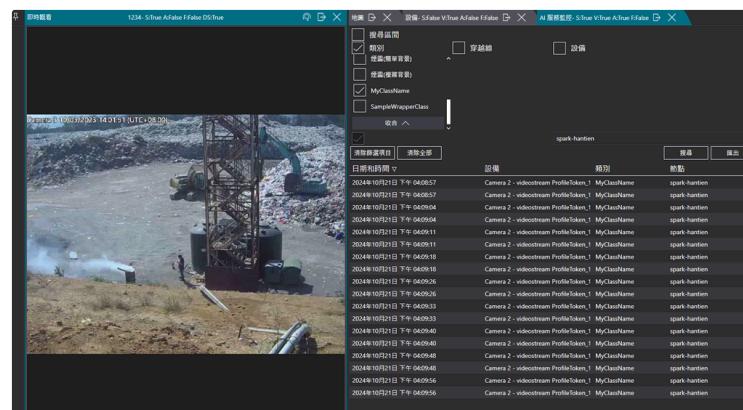
Argo 會綁定某隻攝影機的stream 來執行 "辨識程式.exe"



透過UI的設定後，將設定結果傳給"辨識程式.exe"進行辨識



最後辨識的結果會顯示在Client 的AI服務監控裡面



## 二、溝通API

### 1. 設定演算法參數 (Argo → 辨識.exe)

方法 POST  
網址 [http :// 127.0.0.1:智慧分析程式通訊埠/SetParameters](http://127.0.0.1:智慧分析程式通訊埠/SetParameters)  
MIME application/json  
內容 http code = 只有200  
content = 詳說明

### 2. 程式通道Alive(Argo → 辨識.exe)

方法 GET  
網址 [http :// 127.0.0.1:智慧分析程式通訊埠/Alive](http://127.0.0.1:智慧分析程式通訊埠/Alive)  
MIME plain/text  
內容 http code = 只有200  
content = 空字串

### 3. 事件參數(辨識.exe → Argo)

方法 POST  
網址 [http :// 127.0.0.1:Http\\_port/PostAnalyticsResult](http://127.0.0.1:Http_port/PostAnalyticsResult)  
MIME application/json  
內容 http code = 只有200  
content = 詳說明

## 三、設定詳細內容(json)

### 1. 設定演算法參數 (支援最多10組ROI 和每個ROI有自己的設定值)

```
{  
    "version": "1.2",  
    //溝通json版號  
    "analytics_event_api_url": "http://127.0.0.1:port/PostAnalyticsResult", //設定spark server接收事件的URL  
    "image_width": 1920,           //影像大小  
    "image_height": 1080,          //影像大小  
    "rois": [  
        { "sensitivity": 50,  
            "threshold": 50,  
            "rects": [  
                { "x": 100, "y": 100},  
                { "x": 200, "y": 200},  
                { "x": 300, "y": 300},  
                { "x": 400, "y": 400}  
            ]},  
        { "sensitivity": 50,  
            "threshold": 50,  
            "rects": [  
                { "x": 500, "y": 500},  
                { "x": 600, "y": 600},  
                { "x": 700, "y": 700},  
                { "x": 800, "y": 800}  
            ]}  
    ]  
}
```

★ 支援設定的ROI最多10組, 每組ROI內最多10個點

### 2. 辨識事件結果

```
{  
    "version": "1.2",  
    //溝通json版號  
    "port_num": 51000,           // port num  
    "keyframe": "/9j/4AAQSkZJR..., //JPG image (base64)  
    "timestamp": 15003215760000, //timestamp field in share mem (Argo Fieltime timestamp)  
    "rois_rects": [               //ROI發報框位置 (可內含多個框)
```

```
[ 0, 0, 0, 0 ],  
[ 440, 104, 608, 214 ],  
[ 0, 0, 0, 0 ]  
],  
}
```

● 深度整合方請自行定義演算法參數的內容並提供文件

## 四、Shared mem 設定

使用window shared mem ,目前resolution最大支援 **1920x1080** 的資料

設定如下

```
const int MMF_DATA_HEADER = 0x1234;  
const int MMF_DATA_FOOTER = 0x4321;  
struct MMF_Data_Generic  
{  
    __int64 header = MMF_DATA_HEADER;  
    //video status : 0=no use , 1=new frame, 2=detection got frame  
    int image_status = 0;  
    //resolution  
    int image_width = 0;  
    int image_height = 0;  
    //video size  
    int image_size = 0;  
    //timestamp in ARGO FileTime style  
    uint64_t timestamp = 0;  
    //video data  
    unsigned char* image_data[1920 * 1080 * 3];  
    __int64 footer = MMF_DATA_FOOTER;  
};
```

- Shared mem Name Rule : ChannelFrame\_%d  
%d 是智慧分析程式通訊埠