**人臉不對稱性分析系統**

**程式說明書**

**Face Analysis API**

**Overview**

comprehensive RESTful API for Alzheimer's Disease (AD) assessment combining facial asymmetry analysis and cognitive evaluation using computer vision and machine learning techniques.

**Features**

🔍 **Automated face detection** using MediaPipe FaceMesh (468 landmarks)

📊 **Dual ML predictions** powered by XGBoost models:

- **6QDS Cognitive Assessment** based on questionnaire responses

- **Facial Asymmetry Classification** from facial landmark analysis

🖼️ **Visual analysis** with marked facial landmarks and symmetry lines

📦 **Multi-format support** for compressed image archives

📝 **Integrated questionnaire processing** for comprehensive assessment

**Input Requirements**

1. **Compressed Image Archive**

Upload a compressed archive containing facial photographs:

Supported formats: `.zip`, `.7z`, `.rar`

File size limit: 50MB

Image formats: JPG, JPEG, PNG, BMP, TIFF

Recommended: 5-20 front-facing photos for optimal accuracy

1. **Questionnaire Data**

Provide demographic and assessment questionnaire responses:

age: Age in years

gender: Gender (0: Female, 1: Male)

education\_years: Years of education

q1-q10: Questionnaire responses (10 questions)

**Output**

Returns a JSON response with comprehensive analysis results:

{

"success": **true**,

"error": **null**,

"q6ds\_classification\_result": 0.75,

"asymmetry\_classification\_result": 0.85,

"marked\_figure": "..."

}

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| success | boolean | Analysis completion status |
| error | string/null | Error message if analysis failed |
| q6ds\_classification\_result | float/null | 6QDS cognitive assessment prediction  (0.0-1.0) |
| asymmetry\_classification\_result | float/null | Facial asymmetry ML prediction  (0.0-1.0) |
| marked\_figure | string/null | Base64-encoded image  with facial landmarks |

**API Usage Examples**

**Curl**

curl -X POST "http://localhost:8000/analyze" \

-H "Content-Type: multipart/form-data" \

-F "file=@face\_photos.zip" \

-F "age=70" \

-F "gender=1" \

-F "education\_years=12" \

-F "q1=1" \

-F "q2=0" \

-F "q3=1" \

-F "q4=1" \

-F "q5=0" \

-F "q6=1" \

-F "q7=0" \

-F "q8=1" \

-F "q9=0" \

-F "q10=1"

**Python**

import requests

questionnaire\_data = {

"age": 70,

"gender": 1, # Male

"education\_years": 12,

"q1": 1, "q2": 0, "q3": 1, "q4": 1, "q5": 0,

"q6": 1, "q7": 0, "q8": 1, "q9": 0, "q10": 1

}

with open("face\_photos.zip", "rb") as f:

response = requests.post(

"http://localhost:8000/analyze",

files={"file": f},

data=questionnaire\_data

)

result = response.json()

print(result)

**API Endpoints**

**POST /analyze\_1200\_pics** -   
Upload non-selected photos and questionnaire data for comprehensive analysis

**POST /analyze\_n\_pics** -   
Upload selected photos and questionnaire data for comprehensive analysis

**GET /health** - Service health check with supported formats

**GET /docs** - Interactive Swagger API documentation

**GET /redoc** - Alternative ReDoc API documentation

**GET /** - API information and configuration details

**Docker Quick Start**

1. **change path**

cd path/to/AD-Sensor-Project

一張含有 文字, 螢幕擷取畫面 的圖片

AI 產生的內容可能不正確。

1. **build image and named face-analysis-api**

Make sure to start your Docker app before executing the command

docker build -t face-analysis-api .

一張含有 文字, 電子產品, 螢幕擷取畫面, 軟體 的圖片

AI 產生的內容可能不正確。

1. **start api**

docker-compose up -d

一張含有 文字, 電子產品, 螢幕擷取畫面, 軟體 的圖片

AI 產生的內容可能不正確。

**API Testing**

Visit <http://localhost:8000/docs> for interactive API testing with Swagger UI, where you can:

- Upload test image archives

- Input questionnaire responses

- View real-time analysis results

- Download marked facial images

一張含有 文字, 螢幕擷取畫面, 軟體, 數字 的圖片

AI 產生的內容可能不正確。

**Code list**

|  |  |  |
| --- | --- | --- |
| **path** | **type** | **description** |
| api/ | dir | Project root directory |
| api/data/ | dir | Contains image models and associated data |
| api/data/.gitignore | file | Ignore unnecessary files in data folder |
| api/data/ haarcascade\_frontalface\_default.xml | file | OpenCV face detection Haar Cascade model |
| api/data/ symmetry\_all\_pairs.csv | file | Facial feature symmetry mappings |
| api/data/ xgb\_face\_asym\_model.csv | file | Facial asymmetry XGBoost model (CSV) |
| api/data/ xgb\_6qds\_model.json | file | Pre-trained 6QDS cognitive assessment XGBoost model (JSON) |
| api/.gitignore | file | Ignore unnecessary files in project root |
| api/main.py | file | Application entry point (FastAPI server) |
| api/poetry.lock | file | Locked dependency versions by Poetry |
| api/pyproject.toml | file | Poetry project configuration  and dependencies |
| api/README.md | file | Project documentation |