ThirdFactor.ai — API Gateway for Face & Document Intelligence (v0.1.0)

This documentation describes **ThirdFactor.ai's** authenticated API Gateway for **Face Detection, Face Verification, and Document Analysis**.

Overview

ThirdFactor.ai provides AI-powered identity verification and document understanding services through this secure API gateway.

- **Base URL**: `https://<CUSTOM_URL>` — issued to your organization by ThirdFactor.ai.
- **Auth**: HTTP Bearer Token (`Authorization: Bearer <TOKEN>`) — issued to your organization by ThirdFactor.ai.
- **Content Types**:
  - `application/x-www-form-urlencoded` — for most image-as-base64 inputs
  - `application/json` — for JSON payloads (e.g., face comparison)
  - `multipart/form-data` — for file uploads (e.g., raw image conversion)
- **Success**: HTTP `200 OK`
- **Validation errors**: HTTP `422 Unprocessable Entity`

  All requests are authenticated and encrypted over HTTPS. API tokens must be kept confidential.

Authentication

Every endpoint except `/health` requires a valid **Bearer Token**.

`Authorization: Bearer <YOUR_API_TOKEN>`

Error Model

Sample validation error:

```json
{
  "detail": [
    {
      "loc": ["body", "field_name"],
      "msg": "Field is required",
      "type": "value_error.missing"
    }
  ]
}
```

- **422**: Input validation failed
- **401**: Authentication error
- **500**: Internal server error

Endpoints

1. **Health Check** — `GET /health`

Check system uptime and service status.

Auth: Not required

Response 200

```json
{
  "status": "healthy",
  "service": "thirdfactor-ai-gateway"
}
```

2. **Image → Base64** — `POST /image-to-base64/`

Converts an uploaded image file into a base64 string.

Auth: Required

Form Data

- `image`: *(file, required)* — image file to convert

Response 200

```json
{
  "data_url": "data:image/jpeg;base64,/9j/4AAQ...",
  "content_type": "image/jpeg",
  "filename": "uploaded_image.jpg"
}
```

3. **Detect Face** — `POST /detect-face/`

Detects faces and optional attributes (age range, gender, occlusion, etc.).

Auth: Required

Body (x-www-form-urlencoded)

- `base64_image`: *(string, required)*
- `features`: *(string, optional, default* : ALL) — comma-separated list (`ALL` | `AGE_RANGE` | `EYEGLASSES` | `GENDER` | `FACE_OCCLUDED` | `SUNGLASSES`)

Response 200 (example)

```json
{
  "result": true,
  "total_faces": 1,
  "faces": [
    {
      "confidence": 0.84,
      "age_range": { "min_age": 27, "max_age": 32, "detected_age": 29 },
      "gender": "Male",
      "occluded": false
    }
  ]
}
```

No Faces Found

```json
{ "result": false, "total_faces": 0, "faces": [] }
```

4. Compare Face — POST /compare-face/

Compares two face images to verify if they belong to the same person.

Auth: Required

Query Parameters

- threshold *(float, default: 52.0)* — match threshold
- live_check *(boolean, default: false)* — optional liveness check
- occlusion_check *(boolean, default: false)* — check for obstructions

Body (JSON Array)

```json
[
  "<BASE64_IMAGE_1>",
  "<BASE64_IMAGE_2>"
]
```

Response 200 (match)

```json
{
  "verified": true,
  "confidence": 98.5,
  "percentage_match": 99.9
}
```

Response 200 (no match)

```json
{
  "verified": false,
  "percentage_match": 0
}
```

5. Detect & Crop Document Type — POST /type-of-document-crop/

Automatically detects the type of document (e.g., citizenship-front, citizenship-back, driving-license, passport-front, passport-back, pan-id, disability-id, national-id) and returns a cropped image.

Auth: Required

Body (x-www-form-urlencoded)

- base64_image: *(string, required)*

Response 200

```json
{
  "document_type": "national-id-front",
  "score": 0.98,
  "cropped_image": "data:image/jpeg;base64,/9j/..."
}
```

6. Analyze Document (OCR) — POST /document-extract-information/

Performs OCR and structured information extraction from supported document types.

Auth: Required

Body (x-www-form-urlencoded)

- base64_image: *(string, required)* — the document image

Response 200

```json
{
  "success": true,
  "data": {
    "id_number": "XXXX-XXXX-XXXX",
    "name": "Jane Smith",
    "address": "Sample Address",
    "gender": "F",
    "dob": { "year": "1995", "month": "07", "day": "10" },
    "detected_type": "national-id",
```

```
    "confidence": 0.97
  }
}
```

Security & Compliance

- All traffic must use HTTPS (TLS 1.2+).
- Face and document data are processed **transiently** — no images are stored unless explicitly configured for audit logging.
- ThirdFactor.ai is mostly compliant with major privacy frameworks (GDPR, NDPA).

Version History

Version	Date	Changes
0.1.0	2025-11-21	Initial public release of Face & Document APIs