Simple Face Recognition API for FaReS server 1.1.0

This is a simple API
More information: on GitHub
Contact Info: anonymous@anonymous.ro
Version: 1.1.0

Access

Methods

[Jump to Models]

Table of Contents

Everyone

POST /api/v1/compare   Up

Performs a face recognition/comparison operation (**compare**)

Performs a comparison between two images, each with only one face

Consumes
This API call consumes the following media types via the Content-Type request header:

- application/json

Request body

body request_recognize (required)
Body Parameter — Images to compare

Request headers

Return type
response_recognize

Example data
Content-Type: application/json

```
{
  "data" : {
    "elapsed" : 0.687330178974662,
    "distance" : 0.499988263466158,
    "match" : true,
    "id" : "AAAABBBBCCCCDDDD",
    "request_id" : "DEFSERVER-1585301911.8984709-901610",
```

```
      "validate_rectangle" : {
        "top" : 43,
        "left" : 18,
        "width" : 223,
        "height" : 243
      }
    },
    "info" : {
      "code" : 0,
      "message" : "Extra information"
    }
  }
}
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses
200
all requests response_recognize
500
server errors

POST /api/v1/detect   Up

Performs a face detection operation (**detect**)

Performs a detection operation on an image sent as base64 encoded string

Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

Request body

body **request_detect** (required)
Body Parameter — Images to compare

Request headers

Return type
response_detect

Example data
Content-Type: application/json

```
{
  "data" : {
    "elapsed" : 0.687330178974662,
    "detect_facerectangle" : [ {
      "top" : 43,
      "left" : 18,
      "width" : 223,
      "height" : 243
    }, {
      "top" : 43,
```

```
        "left" : 18,
        "width" : 223,
        "height" : 243
      } ],
      "detect_numfaces" : 1,
      "id" : "AAAABBBBCCCCDDDD",
      "request_id" : "DEFSERVER-1585301911.8984709-901610"
    },
    "info" : {
      "code" : 0,
      "message" : "Extra information"
    }
  }
}
```

Produces
This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- `application/json`

Responses
200
all requests response_detect
500
server errors

POST /api/v1/recognize   Up

DEPRECATED - Performs a face recognition/comparison operation (**recognize**)

Performs a comparison

Consumes
This API call consumes the following media types via the Content-Type request header:

- `application/json`

Request body

> body **request_recognize** (required)
> *Body Parameter* — Images to compare

Request headers

Return type
response_recognize

Example data
Content-Type: application/json

```
{
  "data" : {
    "elapsed" : 0.687330178974662,
    "distance" : 0.499988263466158,
    "match" : true,
    "id" : "AAAABBBBCCCCDDDD",
    "request_id" : "DEFSERVER-1585301911.8984709-901610",
    "validate_rectangle" : {
      "top" : 43,
```

```
        "left" : 18,
        "width" : 223,
        "height" : 243
      }
    },
    "info" : {
      "code" : 0,
      "message" : "Extra information"
    }
  }
}
```

Produces
This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- `application/json`

Responses
200
all requests response_recognize
500
server errors

Models

[Jump to Methods]

Table of Contents

face_rectangle   **Up**

the coordinates to draw a rectangle around the face

top
Integer top left corner of the face rectangle distance from image top margin format: numeric
example: 43

left
Integer top left corner of the face rectangle distance from image left margin format: numeric
example: 18

height
Integer the face rectangle height format: numeric
example: 243

width
Integer the face rectangle width format: numeric
example: 223

input_image   **Up**

format
String image type
example: jpg

size
String image (binary) size format: numeric
example: 477293

image
String image, base64 encoded format: base64
example: /9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAgGBgcGBQgHBwcJCQgKDBQNDAsLD[...]ij/2Q==

`request_detect`

id
String image identifier
example: AAAABBBBCCCCDDDD

face_detect
input_image

`request_recognize`

id
String person/face identifier
example: AAAABBBBCCCCDDDD

face_base
input_image

face_validate
input_image

`response_detect`

info
response_info

data
response_detect_data

`response_detect_data`

id
String person/face identifier
example: AAAABBBBCCCCDDDD

request_id
String ID of the request to compare faces
example: DEFSERVER-1585301911.8984709-901610

elapsed (optional)
BigDecimal time elapsed to run the operation, in seconds format: numeric, float
example: 0.687330178974662

detect_numfaces (optional)
BigDecimal number of faces detected format: numeric, int
example: 1

detect_facerectangle (optional)

array[face_rectangle]

response_info

code
Integer <p>response code; 0 for success, anything else is an error</p> 0 - Success 11 - Maximum content length exceeded 21 - Wrong API key 51 - Not a valid json for request 101 - Max image size exceeded - base image 102 - No face detected - base image 103 - More than one face detected - base image 111 - Max image size exceeded - validation image 112 - No face found - validation image 113 - More than one face found - validation image 151 - Max image size exceeded - detection image 501 - Internal server error 502 - An exception was thrown format: numeric

    Enum:
      0
      11
      21
      51
      101
      102
      103
      111
      112
      113
      151
      501
      502
example: 0

message
String response message, associated with response code
example: Extra information

response_recognize

info
response_info

data
response_recognize_data

response_recognize_data

id
String person/face identifier
example: AAAABBBBCCCCDDDD

request_id
String ID of the request to compare faces
example: DEFSERVER-1585301911.8984709-901610

elapsed (optional)
BigDecimal time elapsed to run the operation, in seconds format: numeric, float
example: 0.687330178974662

match (optional)
Boolean if there is a match or not between faces format: boolean
example: true

distance (optional)
BigDecimal euclidian distance between the faces; threshold for recognition is 0.6 (lower is better)

format: numeric, float
example: 0.499988263466158

validate_rectangle (optional)
face_rectangle