**How do I identify someone’s biometric data in BioPass ID ABIS?**

 To perform the biometric identification for an individual, you need to use the **Identify** method. In this operation, a search is conducted for a match in the biometric database, if one exists. The biometric values of the individual’s face or fingerprints are used as input for identification.

To use this operation, the client must make the call through the API. The complete documentation is available on the website<https://docs.biopassid.com/>.

**Input fields:**

* **Face:** Information about facial biometrics, including:
	+ **ImageFileName:** Type of image to be provided (“WSQ”, “PNG”, etc.).
	+ **ImageBase64:** Facial biometric image encoded in a base64 string.
	+ **HorzResolution:** Image size in PPI horizontally.
	+ **VertResolution:** Image size in PPI vertically.
* **Fingers:** List of objects containing each digital biometric to be used, as well as additional information about them:
	+ **FingerName:** Name of the finger, or if there is no name, use the same value as “Position”.
	+ **Position:** Position of the finger from which the biometric was extracted, which can be: Unknown, RightThumb, RightIndex, RightMiddle, RightRing, RightLittle, LeftThumb, LeftIndex, LeftMiddle, LeftRing, LeftLittle.
	+ **CollectedType:** Type of fingerprint capture, which can be: “Plain” for flat capture, “Rolled” for rolled capture, and “Latent” for latent capture.
	+ **ImageFileName:** Type of image to be provided (“WSQ”, “PNG”, etc.).
	+ **ImageBase64:** Digital biometric image encoded in a base64 string.
	+ **HorzResolution:** Image size in PPI horizontally.
	+ **VertResolution:** Image size in PPI vertically.
	+ **AnomalyDescription:** Additional information about fingerprint anomalies, if any, specifying the type and period (permanent, temporary) of the anomaly.

Example of how the request body would look like:

{

 "Face": {

 "ImageFileName": "string",

 "ImageBase64": "string",

 "HorzResolution": 0,

 "VertResolution": 0

 },

 "Fingers": {

 "Finger": [

 {

 "FingerName": "string",

 "Position": "string",

 "CollectedType": "string",

 "ImageFileName": "string",

 "ImageBase64": "string",

 "HorzResolution": 0,

 "VertResolution": 0,

 "AnomalyDescription": {

 "AnomalyType": "string",

 "AnomalyTimeFrame": "string"

 }

 }

 ]

 },

 "MatchingWithDetails": true

}

The request header, in turn, is Where the content type should be inserted. Below is an example of how the header should be represented:

{

 'Content-Type': 'application/json'

}

Finally, in the request, the parameters should specify that it is a POST method, the URL ‘https://abis.api.biopassid.com/identify/post’, along with the previously described header and body. Below, we present an example of Python code that performs this request. Examples in other programming languages can be found in our documentation.

import http.client

import json

conn = http.client.HTTPSConnection("abis.api.biopassid.com")

payload = json.dumps({

 "Face": {

 "ImageFileName": "string",

 "ImageBase64": "string",

 "HorzResolution": 0,

 "VertResolution": 0

 },

 "Fingers": {

 "Finger": [

 {

 "FingerName": "string",

 "Position": "string",

 "CollectedType": "string",

 "ImageFileName": "string",

 "ImageBase64": "string",

 "HorzResolution": 0,

 "VertResolution": 0,

 "AnomalyDescription": {

 "AnomalyType": "string",

 "AnomalyTimeFrame": "string"

 }

 }

 ]

 },

 "MatchingWithDetails": True

})

headers = {

 'Content-Type': 'application/json'

}

conn.request("POST", "/identify/post", payload, headers)

res = conn.getresponse()

data = res.read()

print(data.decode("utf-8"))

The response received in case of a successful operation will include various response fields, as described below.

**Responde fields:**

* **Success:** Boolean value indicating the success of the operation.
* **Message:** Additional message regarding the success of the operation.
* **MatchingSubjects:** Object containing information about the individual with whom the biometric match occurred.
	+ **CandidateId:** Internal identifier of the candidate passed in the operation call.
	+ **Score:** Numerical value representing the degree of confidence of the biometric match.
	+ **FacesMatched:** Object containing the confidence score of the face match and additional information.
	+ **FingersMatched:** Object containing the confidence score of the fingerprint match and additional information.
* **Faces:** Object containing information about the submitted facial biometrics and the confidence score of the match.
* **Fingers:** Object containing information about the submitted fingerprint biometrics and the confidence score of the match.

The response message for the provided example can be checked below:

{

 "Success": true,

 "Message": "string",

 "MatchingSubjects": [

 {

 "CandidateId": "example-candidate-id",

 "Score": 0,

 "Threshold": 0,

 "FacesMatched": [

 {

 "MatchedIndex": 0,

 "Score": 0,

 "SondaTemplateFaceId": "example-candidate-id",

 "ReferenceTemplateFaceId": "example-candidate-id"

 }

 ],

 "FingerMatched": [

 {

 "FingerMatchedId": 0,

 "Score": 0,

 "FingerSourceMatchedId": 0,

 "FingerSourcePosition": 0,

 "SondaTemplateFingerId": "example-candidate-id",

 "ReferenceTemplateFingerId": "example-candidate-id"

 }

 ]

 }

 ],

 "Status": "string",

 "Faces": {

 "Faces": [

 {

 "Id": "string",

 "CreateDateUtc": "2024-07-05T18:12:23.192Z",

 "PathImageBlob": "string",

 "ImageFileName": "string",

 "ImageBase64": "string",

 "ImagePngBase64": "string",

 "ImageThumbnail120PngBase64": "string"

 }

 ],

 "FacesMatch": [

 {

 "InicialImageGuid": "example-candidate-id",

 "ReceivedImageFileName": "string",

 "Score": 0,

 "Threshold": 0

 }

 ]

 },

 "Fingers": {

 "Fingers": [

 {

 "Id": "string",

 "CreateDateUtc": "2024-07-05T18:12:23.192Z",

 "PathImageBlob": "string",

 "ImageFileName": "string",

 "ImageBase64": "string",

 "ImagePngBase64": "string",

 "ImageThumbnail120PngBase64": "string",

 "Nfiq1": 0,

 "Nfiq2": 0,

 "Nfiq21": 0,

 "Position": 0,

 "PositionName": "string",

 "CollectedType": "string",

 "AnomalyTimeFrame": "string",

 "AnomalyType": 0,

 "AnomalyTypeName": "string"

 }

 ],

 "FingersMatch": [

 {

 "InicialImageGuid": "example-candidate-id",

 "ReceivedImageFileName": "string",

 "Score": 0,

 "Threshold": 0,

 "StatusName": "string",

 "ErrorMessage": "string"

 }

 ]

 }

}

If an error is returned, please verify that the request body and content are correctly filled out. If you receive a message with the code 200 (OK), it indicates that your request was successfully submitted and processed. The response will include a message with the ID and information of the user for whom the biometric match occurred.