

# Face recognition

---

- [Face recognition](#)
  - [Description](#)
  - [Compatibility](#)
  - [Getting Started](#)
  - [Flow element](#)
  - [Output connection](#)
    - [Original](#)
    - [Cropped](#)
    - [Outlined](#)
    - [Log](#)
  - [Examples](#)
    - [Best Hit Only](#)
    - [All Hits](#)
  - [Error handling](#)
  - [License](#)

## Description

---

This app detects faces on the spot using the OpenCV library without the use of remote services.

These matches can then be cropped and resized as desired.

## Compatibility

---

- Switch Fall 2022
- Windows or MacOS x64

## Getting Started

---

Use one of our sample flows and drop a sample image into the flow.

## Flow element

| Property                          | Type   | Description  |
|-----------------------------------|--|--|
| <i>Mode</i>                       | Enum<br>[bestHitOnly<br>  oneHitOnly<br>  allHits] | <b>general:</b> if no hit it will result in an error; <b>bestHitOnly</b> : The most accurate hit will be used, all others are ignored; <b>oneHitOnly</b> : Only one hit is allowed, if there are multiple hits it will result in a warning; <b>allHits</b> : Every hit will be processed, no warning |
| <i>Append position as dataset</i> | Boolean  | Defines if the resulting JSON (position of each hit) should be appended as dataset   |
| <i>Dataset name</i>               | String   | Name of resulting dataset  |
| <i>ExpertSettings</i>             | Boolean  | Enable / Disable expert settings of the OpenCV module  |
| <i>Haarcascade</i>                | String   | Path to haarcascade (trained model) to be used for object / face recognition   |
| <i>Scale factor</i>               | Float  | Scale factor of the OpenCV library   |
| <i>Minimum neighbours</i>         | Integer  | Minimum distance between hits  |
| <i>Minimum width</i>              | Integer  | Minimum width of a hit in pixel  |
| <i>Minimum height</i>             | Integer  | Minimum width of a hit in pixel  |

## Output connection

There are four different output options.

Depending on the output selected, the options / properties of the connection vary as described below.

| Property      | Type                                       | Description                          |
|---------------|--|--------------------------------------|
| <i>Output</i> | Enum [Original   Cropped   Outlined   Log] | Defines the output of the connection |

## Original

The original file will be send to this connection. If the `severity` is `success` or `warning` and `appendResultAsDataset=true` the log file (JSON that contains the position of each hit) will be appended as dataset.

| Property        | Type                             | Description |
|-----------------|----------------------------------|-------------|
| <i>Severity</i> | Enum [success   warning   error] |             |

## Cropped

Only on success the cropped image will be send to this connection.

Options:

| Property                        | Type   | Description   |
|---------------------------------|--|---|
| <i>Crop image to percentage</i> | Percentage                                       | The detected face/object has a certain size, which will be extended by the defined value (percentage of size of the hit). E.g. The size of the hit is 200x200 pixel, it will be multiplied by the defined percentage resulting in 300 x 300 pixel for a value of 150 %. Default = 175 |
| <i>Output format</i>            | Enum [Input   TIFF   PNG   JPEG]                 | Defines the output format of the cropped image. Default = Input   |
| <i>Image quality</i>            | Integer  | If <code>Output format</code> is not <b>Input</b> Quality of the cropped image. Values from 1-100; Default = 100  |
| <i>Resize image</i>             | Boolean  | Enable / Disable resizing   |
| <i>Width</i>                    | Integer  | The width of the cropped image in pixel   |
| <i>Height</i>                   | Integer  | The height of the cropped image in pixel  |
| <i>Fit</i>                      | Enum [contain   cover   fill   inside   outside] | The method for resizing. For more details checkout: <a href="https://sharp.pixelplumbing.com/api-resize#resize">https://sharp.pixelplumbing.com/api-resize#resize</a>   |

## Outlined

Carries the original image that contains rectangles for each hit.

| Property        | Type                     | Description         |
|-----------------|--------------------------|---------------------|
| <i>Severity</i> | Enum [success   warning] | The severity levels |

## Log

Carries a JSON file containing the position of each hit.

| Property        | Type                     | Description         |
|-----------------|--------------------------|---------------------|
| <i>Severity</i> | Enum [success   warning] | The severity levels |

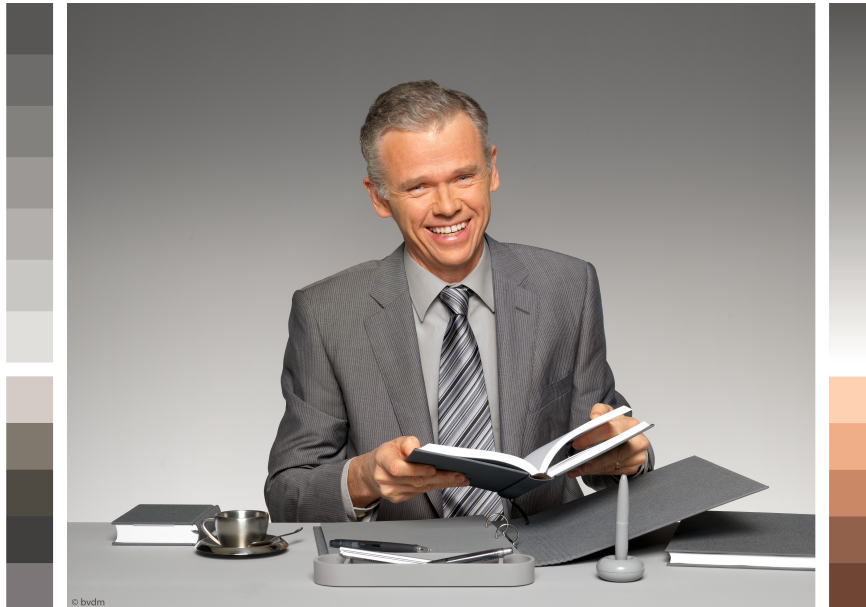
Example:

```
[
  {
    "confidence-level": 8.98906311247265,
    "x": 1510,
    "y": 447,
    "w": 1160,
    "h": 1160,
    "bestHit": true
  },
  {
    "confidence-level": 0.4774161122040823,
    "x": 2953,
    "y": 3324,
    "w": 218,
    "h": 218,
    "bestHit": false
  }
]
```

# Examples

## Best Hit Only

Use the sample flow 'sample bestHitOnly' to reproduce this example. Unfortunately we cannot provide sample images, so you have to use your own images.



**Input image:**

**256x256 PNG:**

*Connection configuration:*

| Property                        | Value   |
|---------------------------------|---------|
| <b>Name</b>                     | Cropped |
| <b>Description</b>              |         |
| <b>Corner angle</b>             | 0       |
| <b>Color</b>                    | Gray    |
| <b>Hold jobs</b>                | No      |
| <b>Output</b>                   | Cropped |
| <i>Offset top percentage</i>    | Default |
| <i>Crop image to percentage</i> | Default |
| <b>Output format</b>            | PNG     |
| <i>Image quality</i>            | Default |
| <b>Resize image</b>             | Yes     |
| <b>Width</b>                    | 256     |
| <b>Height</b>                   | 256     |
| <b>Density</b>                  | Default |
| <b>Fit</b>                      | Default |

Resulting image:



**72x72 JPEG:**

Connection configuration:

| Property                        | Value   |
|---------------------------------|---------|
| <b>Name</b>                     | Cropped |
| <b>Description</b>              |         |
| <b>Corner angle</b>             | -90     |
| <b>Color</b>                    | Gray    |
| <b>Hold jobs</b>                | No      |
| <b>Output</b>                   | Cropped |
| <b>Offset top percentage</b>    | Default |
| <b>Crop image to percentage</b> | Default |
| <b>Output format</b>            | JPEG    |
| <b>Image quality</b>            | Default |
| <b>Resize image</b>             | Yes     |
| <b>Width</b>                    | 72      |
| <b>Height</b>                   | 72      |
| <b>Density</b>                  | Default |
| <b>Fit</b>                      | Default |

Resulting image:



## All Hits

Use the sample flow 'sample allHits' to reproduce this example. Unfortunately we cannot provide sample images, so you have to use your own images.

If there are multiple hits, cropped files will be named like `<job name>_<index>`, e.g.:  
 my\_sample\_job\_1.jpeg, my\_sample\_job\_2.jpeg, ...

### Input image:

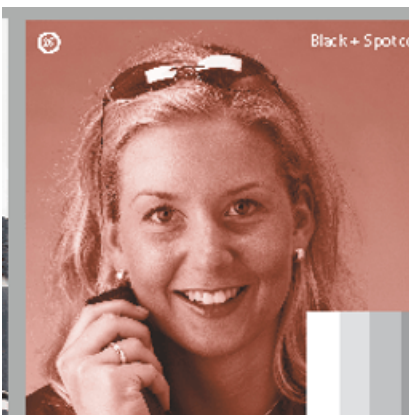


### 256x256 PNG:

#### Connection configuration:

| Property                        | Value   |
|---------------------------------|---------|
| <b>Name</b>                     | Cropped |
| <b>Description</b>              |         |
| <b>Corner angle</b>             | 0       |
| <b>Color</b>                    | Gray    |
| <b>Hold jobs</b>                | No      |
| <b>Output</b>                   | Cropped |
| <b>Offset top percentage</b>    | Default |
| <b>Crop image to percentage</b> | Default |
| <b>Output format</b>            | PNG     |
| <b>Image quality</b>            | Default |
| <b>Resize image</b>             | Yes     |
| <b>Width</b>                    | 256     |
| <b>Height</b>                   | 256     |
| <b>Density</b>                  | Default |
| <b>Fit</b>                      | Default |

Resulting images:





## Error handling

---

This app uses two types of errors:

- job fail error: if any script error occurs the job will fail and send to the problem jobs folder
- severity=error: if no face is recognized, the job will only be send to those connections with severity=error

## License

---

This app uses the Apache 2.0 License. It can be viewed at the following link:

<https://www.apache.org/licenses/LICENSE-2.0.txt>