

## INTRODUCTION

Within the EU research project “FAST and efficient international disaster victim Identification” (FASTID), the Fraunhofer-Institute IOSB started to develop a software module for content based image retrieval. Given an input image, the aim of the software is to find those images within an image database which show the same object or scene as the input image.

## SYSTEM REQUIREMENTS

**Operating system:** 64bit Windows (Windows XP, Windows Vista, Windows 7, 8, 8.1, 10)

**Main memory:** at least 2.5 GB per 100.000 images.

**Disk storage:** at least 1 GB per 1.000 images.

## LIMITATIONS OF THIS DEMO VERSION

The only limitation of this demonstration version is that when adding images to the database, only 90% of the selected images will be considered. For using this software in commercial applications, a license for the full version must be obtained.

## INSTALLING / UNINSTALLING

The software has a setup program to copy the relevant files during the installing process. Please ensure the software being installed in a directory with write permissions and enough free space as all the image data is saved into its subdirectories. This is why the default installation directory is set to `c:\IOSB Image Retrieval`.

**Copy protection:** Upon the first run of the program, a pre-key file is generated in the application directory. Please send this pre-key to IOSB ([daniel.manger@iosb.fraunhofer.de](mailto:daniel.manger@iosb.fraunhofer.de)) to obtain a corresponding key file. This key file must be put into the application directory.

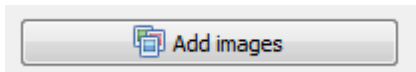
**Uninstalling:** To uninstall the software please use the default uninstall functionality in system control – software. The Registry is not accessed by the program.

## FUNCTIONALITY

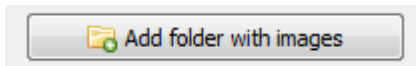
The functionality is split into three steps. At first, the images to be searched in later for similar images have to be selected. Subsequently, the image database is built using the selected images. In the third step, the database can be searched with arbitrary images.

### 1. SELECT IMAGES TO SEARCH IN

There are two ways of selecting images in order to create the image database:



This button can be used to select single or multiple images. A dialog appears in which multiple images can be selected pressing the control / shift key.

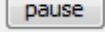


To select all images of a directory, this button can be used. After specifying the directory, the program asks if all images in the respective subdirectories should be selected as well (recursively).

Both ways of adding images can be applied multiple times to select all images.

In this step, all selected images are being opened, resized to a maximum size of 640x640 pixel and finally saved in the "DBImages" folder using consecutively numbered filenames. Thus, the original images are not modified by the software and after this step they do not necessarily need to be available to the software.

Images which have already been added are not considered any more if they are added again (detection based on file name and path). This means, a large collection of images can be added repeatedly leading while only the new images being added to the database.

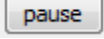
The adding of images can be paused with the button . Afterwards, if the same directory or images are added again, only the remaining images are processed.

### 2. CREATE THE IMAGE DATABASE

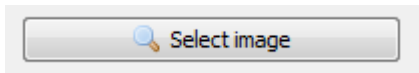


If at least 25 images have been selected in step 1, the software is ready to build the image database. Therefore, features are calculated for all images which are then registered in the database. Depending on the number of images and the computer hardware, this step can take several minutes. The progress of the operation is displayed by the progress bar in the bottom left-hand corner.

If the program crashes during this operation, please check the system requirements in terms of memory usage.

Using the  button, the creation of the database can be interrupted in the first step ("calculating features..."). The creation can be continued afterwards.

### 3. SEARCH THE DATABASE FOR SIMILAR IMAGES

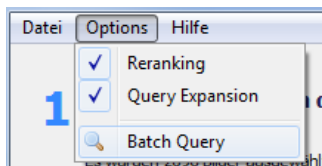


This button is activated once the database has been built successfully. It opens a dialog for selecting the query image. Alternatively, the query image can be specified by dragging an image file and dropping it on the background of the software. After selecting a query image, its features are calculated and the database is searched for similar images. The query image is displayed in the bottom left and the respective most similar result images are shown in the right. For each result image, a similarity score is given which is the higher the more local matches were found. In the tab “Result scores”, the similarity scores are listed together with the original filenames of the corresponding images.

By **clicking** one of the result images, a new window is opened showing the two images. If the **Control key** is pressed during clicking, the original file name is copied into the clipboard. If the **Alt key** is pressed during clicking, the original image is opened (using the standard application defined for that file type).

**Please note:** If an image of the database is selected as query image, it will not be part of the result list since it is the identical image. However, the images are only tested with respect to their filename and path, i.e. if a database image is copied anywhere else and used as query, it will appear in the result list. This function works with both the original filenames and the copies in the subfolder “\DBImages”.

### 4. BATCH QUERY MODE



Using the function „Batch Query“ in the menu „Options“, it is possible to select multiple query images at a time. The selected images will be used to query the database one after another. The results are displayed in all three tabs and are saved in the subfolder `\api` in three ways:

- For each query image, a file `*result.jpg` is generated showing the most similar images. The filename contains the similarity enabling to automatically sort the result files in the Windows Explorer according to their filename.
- For each query image, a file `*result.txt` is generated containing the list of the most similar images.
- One overall result file `Batchquery_Results_<timestamp>.txt` is generated containing a list of all query images and the similarities of their respective most similar images. In this sorted list, it is very easy to recognize those query images which yielded significant results.

Once the batch query is done, the subfolder is opened automatically.

**Please note:** If an image of the database is selected as query image, it will not be part of the result list since it is the identical image. However, the images are only tested with respect to their filename and path, i.e. if a database image is copied anywhere else and used as query, it will appear in the result list. This function works with both the original filenames and the copies in the subfolder “\DBImages”.

This makes it possible to find similar images within a collection of images: simply create a database with all images and start a batch query operation using the same images. The program then ensures that each query image will not find itself as the most similar image.

## 5. OPTIONS

The menu item „Options“ contains the following two parameters which are enabled by default:

- **Re-Ranking:** After the initial, global search step (considering all database images with a fast algorithm), the first 50 images will be examined in more detail. Usually, this ensures that if there are any matches, they will appear on the very first ranks in the result list.
- **Query Expansion:** if enabled, this functionality even goes one step further and starts a second search job using the features of the most similar images of the initial search results. Often, this process helps to find multiple views of an object/scene which do not share that much visual elements with the original query image but are still related to it.

## 6. COMMAND LINE INTERFACE (API)

The program provides a simple command line interface. Please note: before executing commands, the command line needs to be in the application directory.

```
C:\ImgRetr>ImgRetrieval.exe addDir "C:\private\holidays\Las Vegas"
```

Adding all images of the specified directory (and subdirectories) to the database. Please use quote signs for paths with space characters inside.

```
C:\ImgRetr>ImgRetrieval.exe addFile "C:\imageDB\img_a4cd51f10"
```

Adds a single image file to the database. In the example above, this is a JPG file without file extension. In this case, its type is determined by the header of the file.

```
C:\ImgRetr>ImgRetrieval.exe createIndex
```

After adding images, this command starts to build the database index. This can take several hours for large image collections.

```
C:\ImgRetr>ImgRetrieval.exe query "C:\Users\man\Desktop\TestImg.jpg"
```

Starts a query operation using the specified image. After searching the database for similar images, a result overview image (filename `*result.jpg`) displaying the most similar images is put into the subdirectory `api`. Furthermore, a result file `*result.txt` is created containing the list of the most similar images (similarity scores and filenames).

```
C:\ImgRetr>ImgRetrieval.exe watchquery
```

Starts the watch mode, i.e. the program continuously monitors the subdirectory `\watchquery` for new image files. If images are found, each image is used for one query operation. After searching the database for similar images, a result overview image (filename `*result.jpg`) displaying the most similar images is put into the subdirectory `api`. Furthermore, a result file `*result.txt` is created containing the list of the most similar images (similarity scores and filenames). After one query operation, the respective image in the subdirectory `\watchquery` is deleted. Please note: If you add images while the program is running, please copy the images with a temporary file extension first and rename the files later (i.e. copy an image to `\watchquery\image2.jpg.tmp` and subsequently rename it to `\watchquery\image2.jpg`). This prevents that the software tries to read the image already during the copying process.

```
C:\ImgRetr>ImgRetrieval.exe resetIndex
```

This command deletes the database and all the images (but not the original images since when adding files to the database, the program always takes a copy of the images and only those copies are deleted with this command).

## 7. MORE TIPS:

1. **Resolution of the images:** both the images added to the database and the images used as query images are resized to a maximum length of 700 pixels if necessary. This is done because high resolution images contain a lot of fine details which are not relevant for the recognition of objects visible in a large part of the image.
2. **Different sizes of the objects:** the recognition is robust to a certain degree of differences in size. However, if two similar objects/scenes differ too much in size, the algorithm begins to compare details which are too different. Example: two images showing a tattoo - the first image showing the overall tattoo (i.e. focusing on the tattoo contours) and the second image being a close-up focusing on skin pores, hairs etc. which are not visible in the first image.
3. **Background clutter:** the algorithm always processes all details of the images being susceptible to distraction by background objects visible in multiple images. In the scope of forensics, these objects can be rulers, meter sticks, measuring lines etc. being put into the scene for later assessing the size of the objects. Please make sure that those common background objects are not visible in both the database and the query image. For personal images, this holds as well, especially for images containing time stamps inside the visual part of the images. Please use other tools to remove those distracting image regions before starting query operations.
4. Recognize **rotated objects:** Typically, a rotation of an object about the axis from the object to the camera (or a rotation of the camera itself) is not a problem for the system. However, if objects rotate differently leading to the objects being visible from too different perspectives, the recognition rate can be impaired.
5. **Grayscale images vs. color images:** This software mainly builds on edge information rather than on color information. That's why objects in color images typically can be found with grayscale query images and vice versa. Please note that this does not hold for contrast inversions: If the partition of dark and light elements is alternated, the images will become dissimilar to the algorithm. One example is searching for trademark logos – please make sure to both use a query image with dark logo on bright background and one with bright logo on dark background to find both versions.