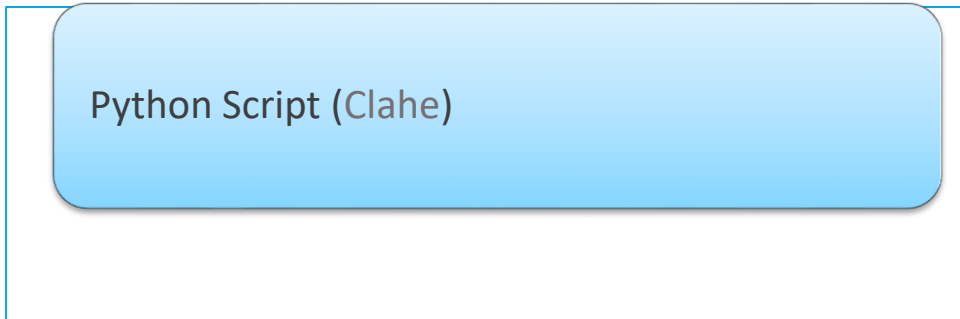# Application Note #32

How to perform :
«Contrast limited adaptive histogram equalization (CLAHE)»

The application-note purpose is to guide the user in applying the contrast limited adaptive histogram equalization (CLAHE) algorithm. The CLAHE algorithm partitions the images into contextual regions and applies the histogram equalization to each one. This evens out the distribution of used grey values and thus makes hidden features of the image more visible. The full grey spectrum is used to express the image. Contrast Limited Adaptive Histogram Equalization, CLAHE, is an improved version of AHE, or Adaptive Histogram Equalization. Both overcome the limitations of standard histogram equalization.

Application Flowchart

Python Script (Clahe)

Index

1. Open the working dataset
2. Load the Python Script
3. Set the Script features
4. Run the Python Script

NOTE :
Clahe script requires the Anaconda3 package installed on you workstation. The OpenCV package can also be required.
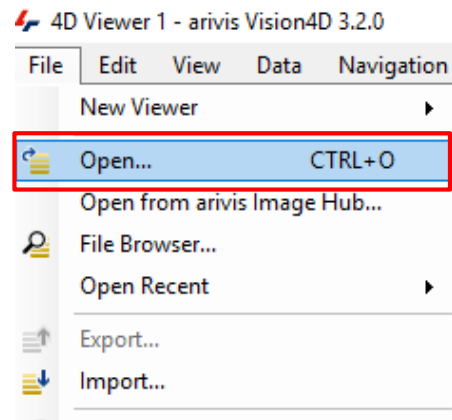
Application Flowchart:

1. Open the working dataset on Vision4D
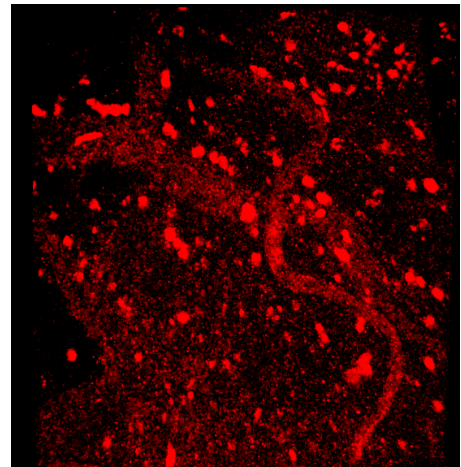
Step 1.1
Select the *Open..* item from the file menu.

Step 1.2
Select the dataset from the file browser.

TIPS :
The dataset is visualized according to the current rendering setting parameters. Please refer to the (arivis Vision4D Help) for more details


DETAILS:
The dataset is a multi dimensional, discrete, representation of your real sample volume. It can be structured as a Z series of planes (Optical sectioning) of multiple channels (dyes) in a temporal sequence of time points (located in several spatial positions).
Usually the dataset shows a single experimental situation (a complete experiment can be composed by several datasets). The datasets are available as graphic files saved in plenty of file formats (standard formats as well as proprietary formats)
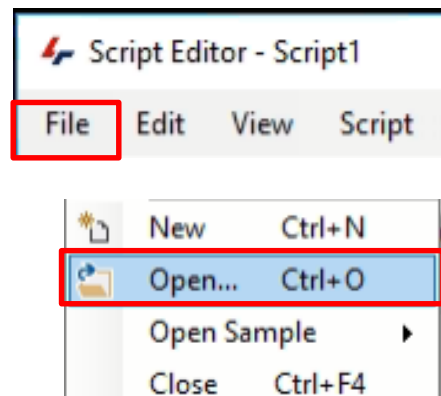
2. Load the Python Script

Step 2.1
Open Python Script Editor.
From the «*Extra*» menu, select the
«*Script Editor*» item
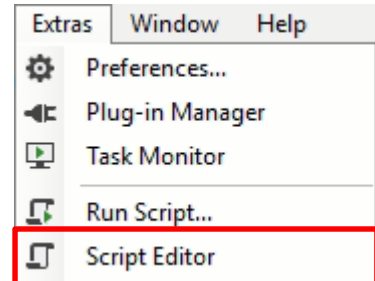
Step 2.2
Load the "*Clahe_Rev5 (3_4).py*"
Python Script.
Browse the folder on which the file
has been saved

Python script code usage rights.

```
#
# NAME: Clahe - Automatic Histogram enhancement
# FILE: Clahe_Rev5(3_4).py
# REVISION : 1.1.0  - 2021/05/03
# AUTHOR : Maurizio Abbate
# Copyright(c) 2021 arivis AG, Germany. All Rights Reserved.
#
# Permission is granted to use, modify and distribute this code,
# as long as this copyright notice remains part of the code.
#
# PURPOSE : Apply the Clahe algorithm to the ImageSet
#           Clahe = Contrast Limited Adaptive Histogram Equalization
#
# Tested for V4d Release : 3.4  -- tested on : 2021-05-03
# Uses Skimage and OpenCv2 library
#
# --------------------------- External Package Import ---------------
```

The user has the permission to use, modify and distribute this code, as
long as this copyright notice remains part of the code itself.
Copyright(c) 2021 arivis AG, Germany. All Rights Reserved.

NOTE :
Only the parameters located in the "*USER SETTING*" area can be modified. Don't
change any other number, definition or text in the code outside this dedicated area.

3. Set the Script features

Some Clahe algorithm options can be set to optimize the results.

INPUT_CHANNEL : Set the source channel to be equalized
NOTE: **_Only one channel can be processed_**

OUTPUT_CH_CHANNEL : Set the output channel name

```
# -----------------------------------------------------------------
INPUT_CHANNEL = 0    # <---- Count start from 0 (Ch#1 == 0)
# name of the new channel (clahe storage)
OUTPUT_CH_NAME = "Clahe_"
#
# options
# -----------------------------------------------------------------
CLAHE_CV2 = True         #    True == Clahe using OpenCV2
HISTO_STRETCH = True     #    True == stretch the histogram of Skimage clahe
# -----------------------------------------------------------------
#    BLOCK_NUMBER_X -> # of X tiles
#    BLOCK_NUMBER_Y -> # of Y tiles
# -----------------------------------------------------------------
BLOCK_NUMBER_X = 16
BLOCK_NUMBER_Y = 16
CLIP_LIMIT_CV2 = 30        # <-- Range between 0 and 100
CLIP_LIMIT_SKIMAGE = 0.02        # <-- Range between 0 and 1
# -----------------------------------------------------------------
```

The size of the used to compute the Clahe can be freely set.
CLAHE_CV2 : **_True_** set the OpenCV2 algorithm – **_False_** the Skimage one
HISTO_STRETCH : With the **_CLAHE_CV2 == False_** sets the result stretching
BLOCK_NUMBER_X = Sets the number of horizontal patches (tiles) used
to compute the local equlization
BLOCK_NUMBER_Y = Sets the number of vertical patches (tiles) used
to compute the local equlization
CLIP_LIMIT_CV2: Clipping limit, normalized between 0 and 100 (higher values give more contrast). It is applied to the OpenCv2 algorithm

CLIP_LIMIT_SKIMAGE: Clipping limit, normalized between 0 and 1 (higher values give more contrast). It is applied to the Skimage algorithm

```
# -----------------------------------------------------------
ACTIVE_PLANE = False   #    True == active plane
ACTIVE_FRAME = True    #    True == active time point
# -----------------------------------------------------------
```

ACTIVE_FRAME : If **_True_**, the Clahe is applied only to the active time point
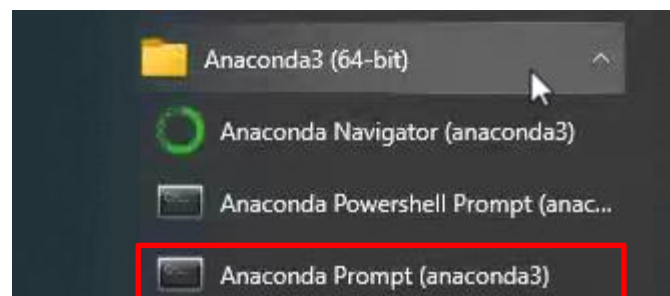
ACTIVE_PLANE : If **_True_**, the Clahe is applied only to the active Z plane

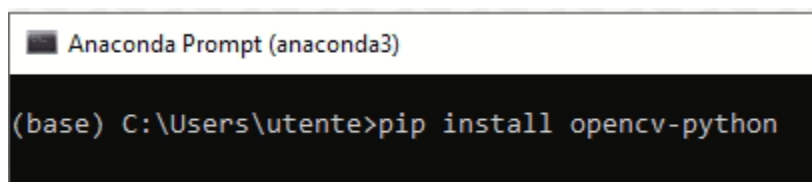# 4. Install the Anaconda3 package

Please refer to the **Application Note #20** for more details

Once the Anaconda3 package has been correctly installed, the OpenCV module can be installed too.
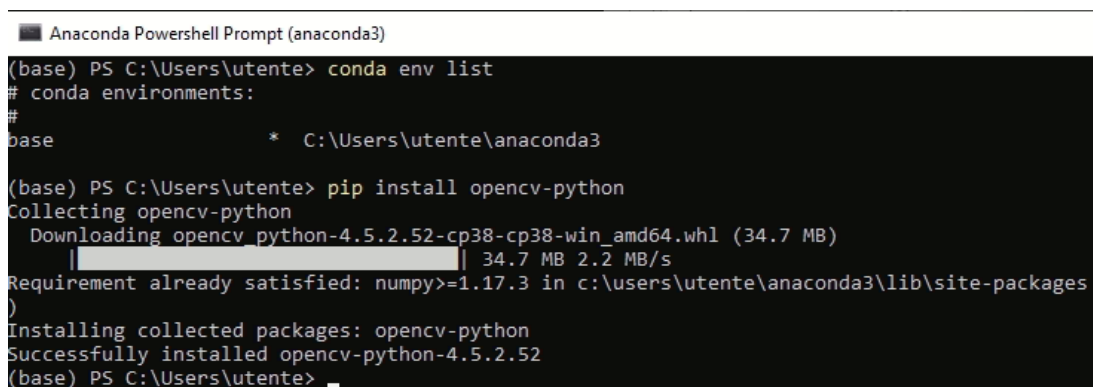
From the Window menu, select the **Anaconda Prompt**

Digit «pip install opencv-python» and press return.

A message confirms the correct OpenCv installation

Contact the arivis local area sales manager to get more information about how to get the python script mentioned here.

Contact the arivis application support to receive additional technical details about the topic described in the application note, or how to adapt the application workflow to your requirements.

"The quantitative analysis of the images represents the art of transforming a visual sensation into its schematic and discrete form allowing its univocal description, classification and mathematical and logical interpretation of its spatial and temporal components"

arivis AG, Am Kabutzenhof 21,
18057 Rostock, Germany

Email : support@arivis.com