isep
École d'ingénieurs du numérique

Multimedia Application Project GIMSEP

MAILLEY Charles, MOREAU Lucie, VETTORATO Maxime

Summary

01

Functionnalities

02

Technical Aspect

03

Project Management

04

Demonstration

Functionnalities

Media Selection

Image transformations

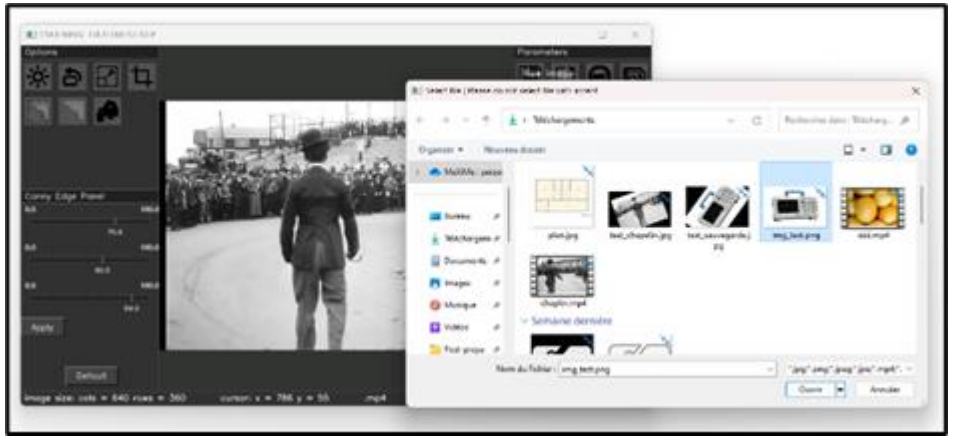
Backward and forward steps

Image saving

Media Selection & Save

01

Using native file browser to get media (Image/video)



02

Using user camera to take a picture

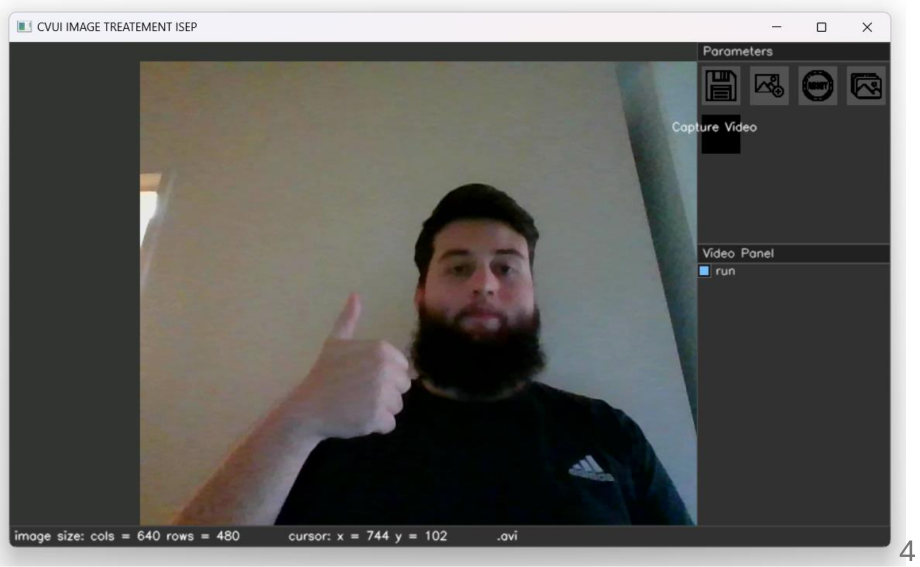
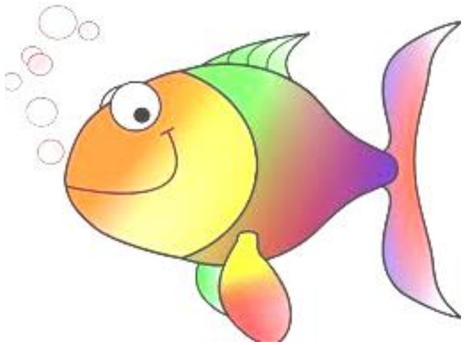
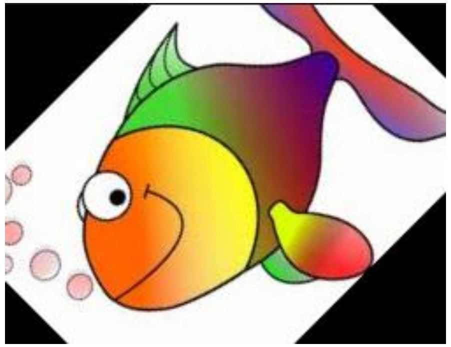

CVUI IMAGE TREATEMENT ISEP
Parameters
Capture Video
Video Panel
run
image size: cols = 640 rows = 480 cursor: x = 744 y = 102 .avi

Image Transformations

Brightness



Rotation

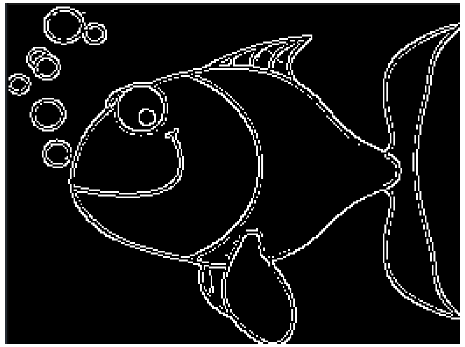


Erosion

Panorama

Cropping

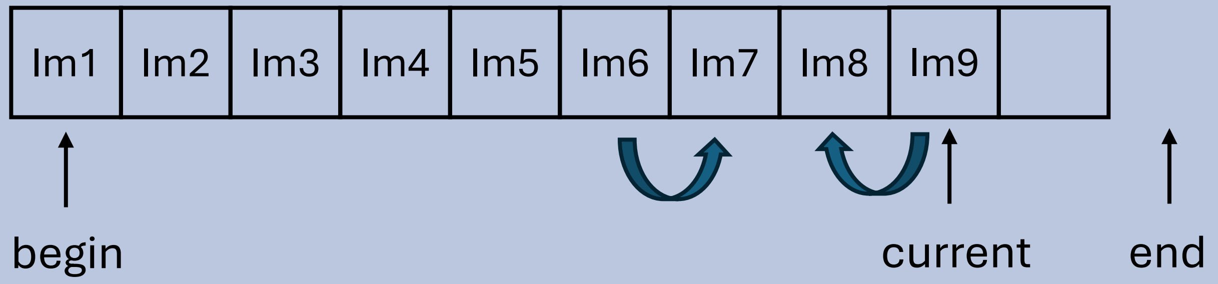
Canny edge



Dilatation

Backward and forward steps

Image Vector



Im1	Im2	Im3	Im4	Im5	Im6	Im7	Im8	Im9	

begin

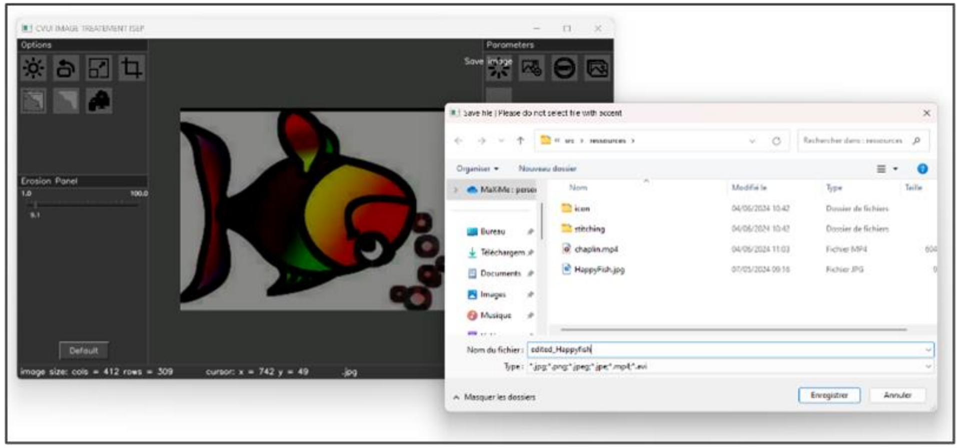
current

end

Control Z
Undo previous changes up to 10 times
Displays previous images

Control Y
Redo undone changes up to the last modification

Image Saving



Save file | Please do not select file with accent

« src › ressources ›

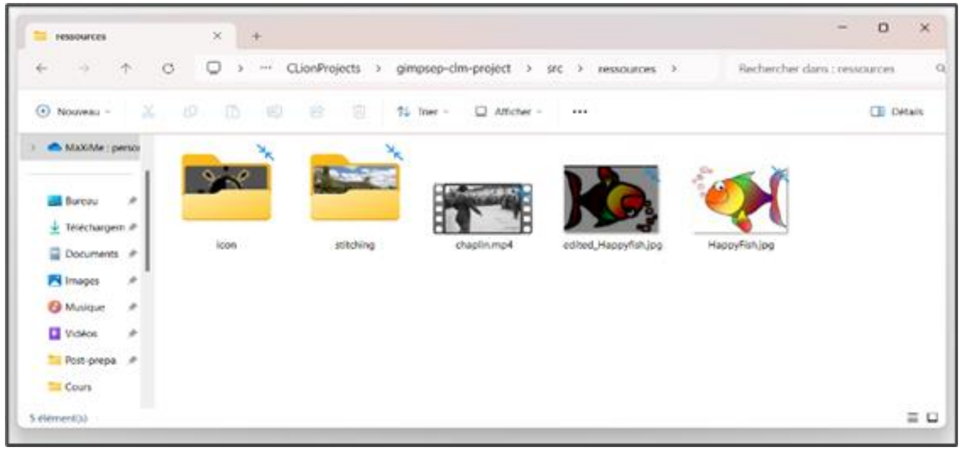
Rechercher dans : ressources

Nom	Modifié le	Type
icon	04/06/2024 10:42	Dossier de fichiers
stiching	04/06/2024 10:42	Dossier de fichiers
chaplin.mp4	04/06/2024 11:03	Fichier MP4
HappyFish.jpg	07/05/2024 09:16	Fichier JPG

Nom du fichier : edited_Happyfish

Type : *.jpg;*.png;*.jpeg;*.jpe;*.mp4;*.avi

image size: cols = 412 rows = 309 cursor: x = 742 y = 49 .jpg



CLionProjects › gimpsep-clm-project › src › ressources ›

Rechercher dans : ressources

icon stitching chaplin.mp4 edited_Happyfish.jpg HappyFish.jpg

5 élément(s)

Simple And efficient !

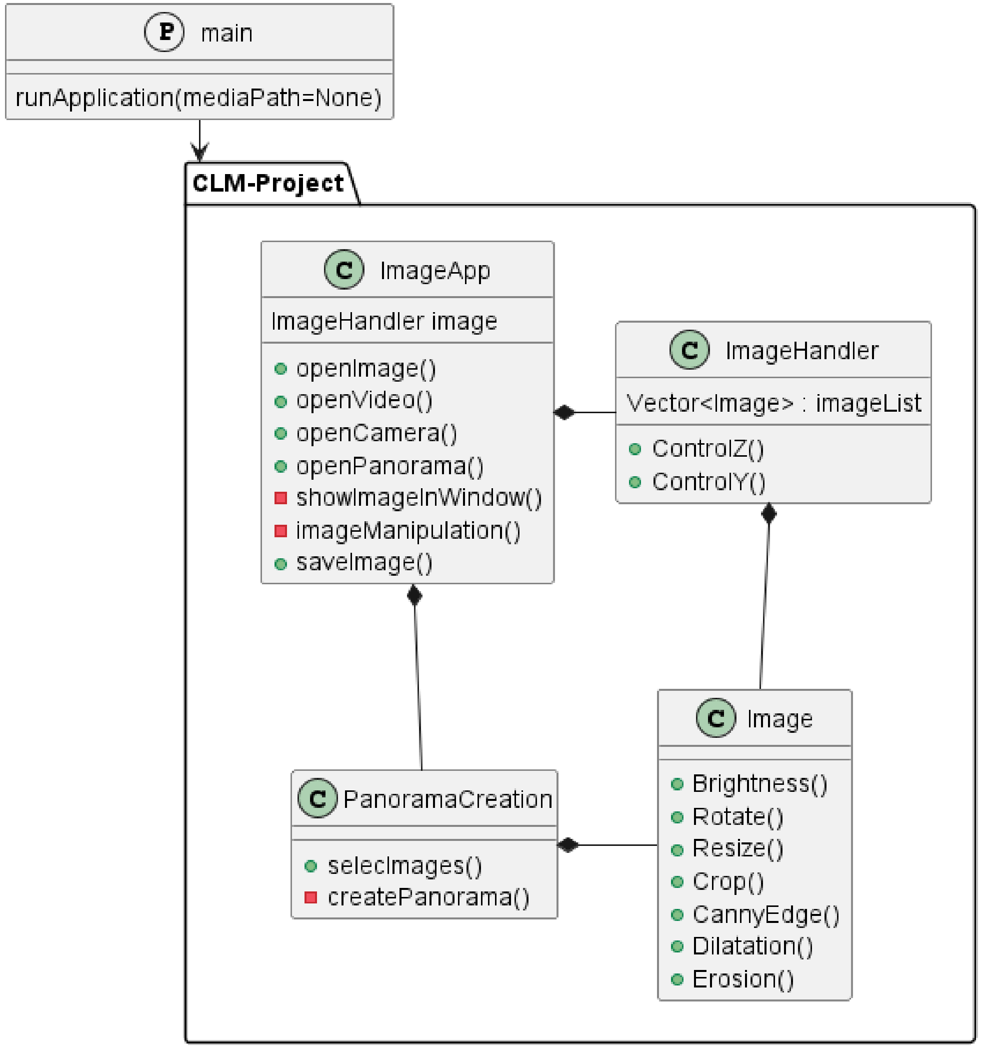
- 1. Open file dialog system
- 2. Let user select, the folder and the name for edited image
- 3. Get the path and use cv::imwrite() to save the creation

Technical aspect

OpenCV

tinyfile dialogs

cvui 2.7.0



P main

runApplication(mediaPath=None)

CLM-Project

C ImageApp

ImageHandler image

- openImage()
- openVideo()
- openCamera()
- openPanorama()
- showImageInWindow()
- imageManipulation()
- saveImage()

C ImageHandler

Vector<Image> : imageList

- ControlZ()
- ControlY()

C PanoramaCreation

- selecImages()
- createPanorama()

C Image

- Brightness()
- Rotate()
- Resize()
- Crop()
- CannyEdge()
- Dilatation()
- Erosion()

Project management

	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6
Charles	Git creation, computer set up, etc.	Dilatation, Erosion	Panorama	User interface base ImageApp	User interface merged with use of all functions (class: Image(), ImagePanorama(), ImageHandler()	User interface and merging of last functions (Open, save, image) + report
Maxime		Lighten, darken		Search on filedialog library and implementation (window + linux)	Select media, save image	Select media, save image + report
Lucie		Rotation, resizing, cropping	Canny edge detection	Merging of all image functionalities in a class Image	Image vector, undo and redo	Report

Time for the demo!

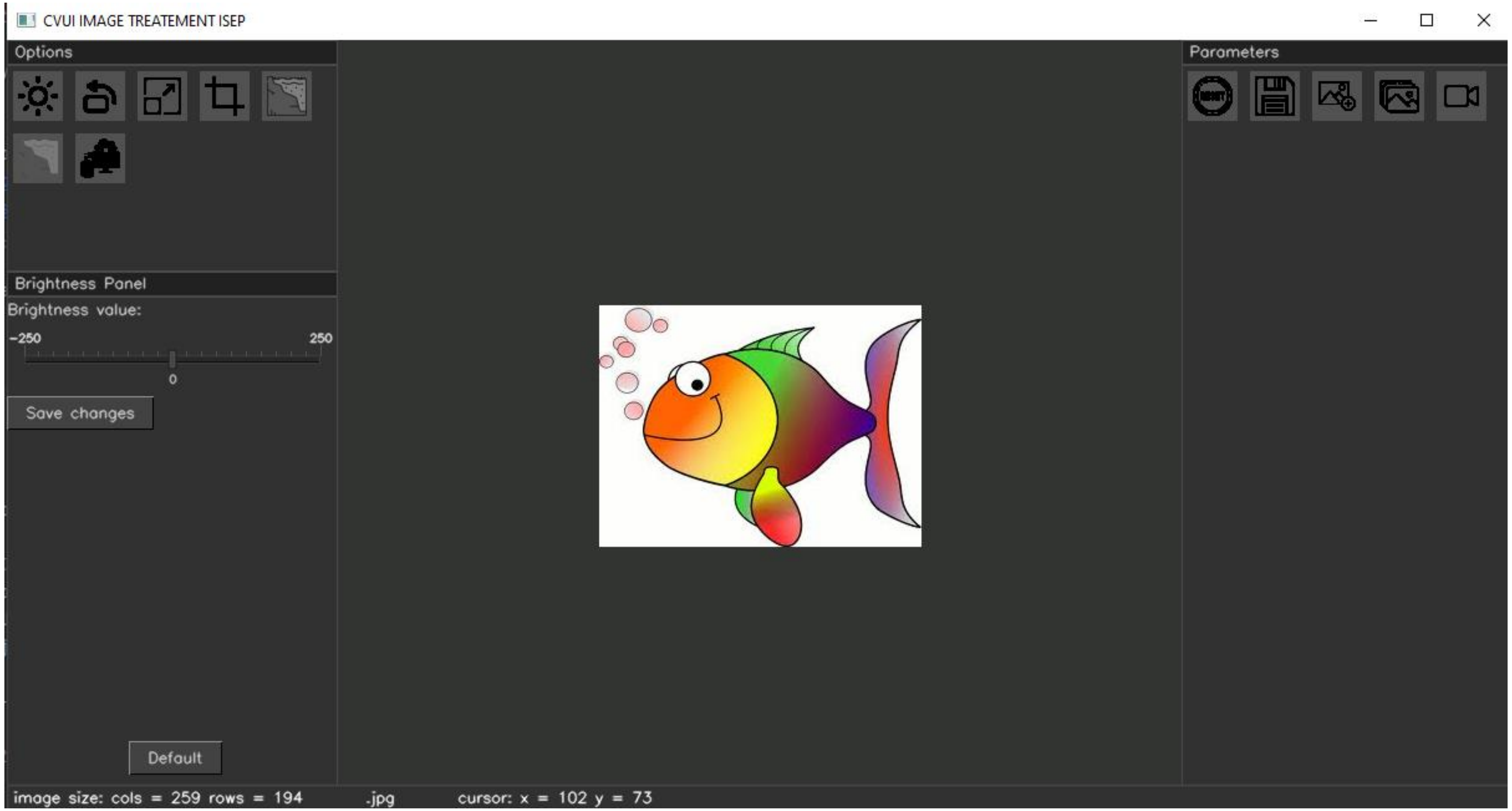
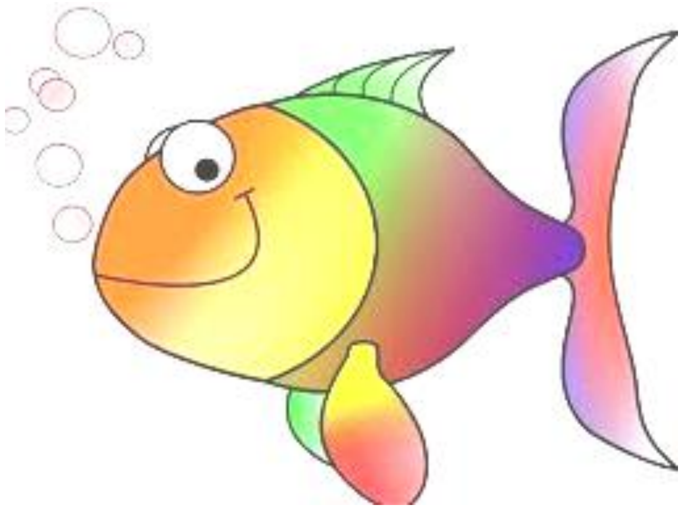


image size: cols = 259 rows = 194 .jpg cursor: x = 102 y = 73

Thank you for your attention

Image Transformations

Brightness



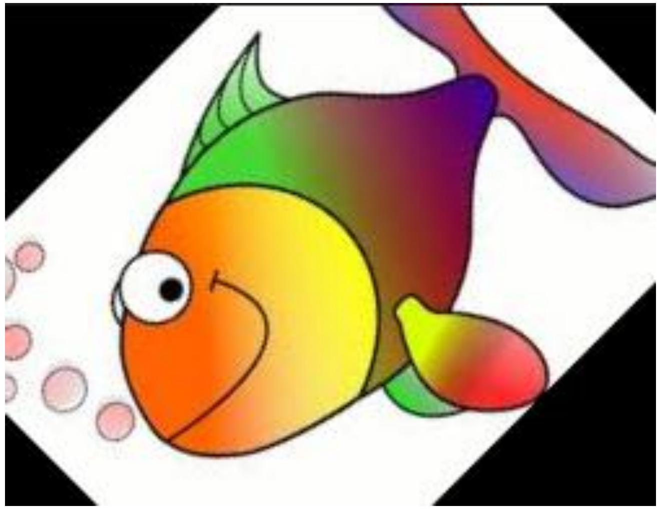
Use cv::convertTo() function to scale pixels brightness with a given *brightnessFactor*

brightnessFactor is a value between –250 and 250 where -250 is fully dark, 0 is the starting image and 250 is fully bright

brightnessFactor is defined by the user through a slider on GUI

Image Transformations

Rotation



Get centerPostion of rotation with Point2f cv::center()

Get rotation matrix with cv::getRotationMatrix2D() and rotationAngle value

Then obtain rotation with cv::warpAffine()

rotationAngle and centerPostion can be selected using three sliders: X-location Y-location and rotation angle

Image Transformations

Erosion

Erosion is obtain by using cv::erode() function which needs a structural Element

The structural Element represents the pattern that will operate erosion in the image

structuralElement is obtained with cv::getStructuringElement() with a kernel size defined by user

Image Transformations

Dilatation

Dilatation is obtain by using cv::dilate() function which needs a structural Element

The structural Element represents the pattern that will operate the dilatation on the image

structuralElement is obtained
with cv::getStructuringElement() with a kernel size defined by user

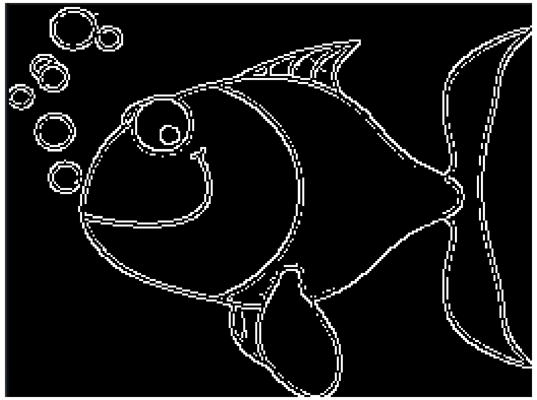
Image Transformations

Cropping

- In order to get the croping of an image, we use two couples (startRow, endRow) and (startCol, endCol)
- The two couples define a new area : The cropped image
- (startRow, endRow) and (startCol, endCol) can be set using four sliders

Image Transformations

Canny edge



- First, choose a low threshold to detect all edge points and a high threshold to detect only edge center points

- Gaussian blur is applied before the Canny edge detection algorithm to reduce noise, making edge points easier to detect

- cv::Canny is used to detect all the wanted edges