International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2015): 6.391

Photo Restoration using Open CV in Python: A Computer Vision Based Project

Subhojit Paul[1], Saikat Ghosh[2], Dhrupad Saha[3], Sourajit Paul[4], Rajesh Dusad[5], Mrinmoy Biswas[6]

[1, 2, 3, 4, 5, 6]Department of Electrical Engineering, UEM Kolkata, Kolkata, India

Abstract: ***At present image processing has become one of the very famous subjects for research area among rapidly growing technologies within engineering. This paper will explain how a damaged or corrupted image or photograph (cracked, torn or folded) can be restored using OpenCV(Open Source Computer Vision) in Python. Here we feed a damaged or corrupted image in the program along with a masked image as input and eventually we will get restored image as output. There are some methods like Denoising, Gray scaling, Thresholding, Dilation etc and few inbuilt function and algorithms are used to construct the code. Thus this is a very interesting, simple, and fascinating image processing technique and can be used for several purposes like restoring old, damaged or corrupted image for archiving purposes.***

1. Introduction

Photos are generally collection of memories that might be very special for an individual. Thus if any kind of damage occurs, we should be able to recover or restore the original photo. In a broader approach, the photographs we usually see which have been preserved and restored from important moments in history. Hindsight's a wonderful thing, and we often don't understand that, when we are capturing something that generations will be interested in years from now. You can't keep everything, but when you think carefully about which photos to delete and which to keep. This is really based on luck. Photographs are one of the best ways of recording what. happens, they can capture a place, events in a place, or people's lives. Photographs are something that people have used more than 100 years to record, so they are pretty useful when we looked back and notice how life has changed. People are more keen to look at the picture than to read the whole descriptive lines.

We should be much selective in case of capturing a single moment and to showcase it. For an example-You don't need to capture 20 photos for showing the same thing, so just keep the best ones unless they became the part of a series chronicling a time or event. In recent days the popularity of digital cameras means that there are a lot more photographs being taken and almost everyone has a camera on their mobile phone. As a result about any event is being photographed and documented by several people. In the past, it was difficult to find photographs of events but now there’s almost too much being produced and you are filled with information.

The damaged photo doesn’t mean that we should discard them. If photographs are damaged you don't necessarily have to write them off. There is a lot we can do digitally ourselves these days to bring them back to an acceptable condition. You can also send off a hard copy to be restored professionally if it’s a photograph that's really important to you.

Thus we are introducing a programmable language using Open CV and Python to restore the damaged images in proper way.

We should counter one problem with photographs when they’ve been used in displays or albums; they are often put onto paper or cardboard, which degrades at a different rate to the photograph. They absorb moisture differently and are prone to insect damage. Our main objective is to minimise these effects, clean and stabilise the photograph again

2. System Design

The photo-restoration process is executed with the help of two different software -

1) OpenCV (*Open Source Computer Vision*)
2) Python

OpenCV*(Open Source Computer Vision)*:

OpenCV is one type of library of programming functions. It is mainly implemented at real-time computer vision technology. Initially Intel introduce this kind of technology and later it was supported by **Willow Garage.** Nowadays it is maintained by **Itseez**. The library is cross-platform which means that two different kinds of platforms can participate separately but work as an integrated system. OpenCV free for use under the open-source BSD license. As OpenCV basically using two separate platforms hence it is refer as multi-platform or platform-independent software. OpenCV is represented in **C language/C++**.

Deep learning frameworks like **TensorFlow**, **Torch/PyTorch** and **Caffe**, is explained by OpenCV. Primary interface of OpenCV is C++, but still it is also comfortable C language also. There are several segments in- 1)**Python**, 2) **Java** and 3)**MATLAB/OCTAVE**. Whenever the library discover Intel’s Integrated Performance Primitives on the system, it will be use these proprietary optimized routines by means of accelerate itself. This is able to run on following desktop OS: **Windows, macOS, Linux, FreeBSD, NetBSD, OpenBSD.** Eventually OpenCV mobile OS: **Android, iOS, Maemo, BlackBerry 10.**

Figure 1: Logo of OpenCV

Python

Python is an interpretable high-level language. However, it is the first general utility programming language. It is developed by **Guido van Rossum** and was first released in 1991. Python has a design motive that encourages code clarity, and a syntax that doesn't needs long blocks of functions.

Python interpreters are available for most of the operating systems. **CPython**, is an open source software. The software has a community-based development model. CPython is handled by the non-profit **Python Software Foundation**.

Figure 2: Logo of Python

It aims for a dynamic type system and automatic memory management. It is compatible for multiple paradigms, which includes object-oriented, functional, procedural and imperative, and has a vast and comprehensive standard library.

Anaconda *(Python distribution)*:

Anaconda is an open-source distribution of Python. The **R programming** languages is using for large-scale data processing, predictive analytics and scientific computing, that aims to simplify package management and development. Package versions are managed more easily by package management system conda.

Figure 3: Logo of Anaconda

3. Methodology

Initially the path of the damaged file is mentioned and is loaded in our program. To enhance the final output image, the damaged photo is processed by **Denoising** it by means of Fast Means Denoising method. This reduces the coarseness of the image and makes the image smooth.

Next, a greyscale masked image of the same damaged image is created by marking the regions which are to be repaired and is also loaded. The marking is done by creating strokes over the damaged area with a brush of white colour edited in MS Paint for Windows or any similar software in Mac or Linux Ubuntu.

Now, this masked file is **thresholded** and converted into a binary image. **Thresholding** is an effective way which enables us for segmentation of the image. It partitions an image by isolating into foregrounds and backgrounds, i.e., it helps in detecting the marked areas that needs to be repaired

After the above step, Dilation is incorporated. As Thresholding reduced the thickness of the marked areas which is not desired, dilation is necessary. Dilation enables us to add pixels over a region by defining a kernel value leading in making the marked lines thicker. Then this dilated masked file is saved into our machine.

And here comes the core command which basically repairs the damaged areas: Image **Inpainting** Function. This technique works upon the Fast Marching Method. The algorithm starts from the boundary of the marked area and goes on filling the areas with an average value of the pixels from a small circular neighbourhood of that area. We have used TELEA method here which is better as it integrates more seamlessly into the image. Finally the image is ready, rendered as well as restored.

4. Flowchart

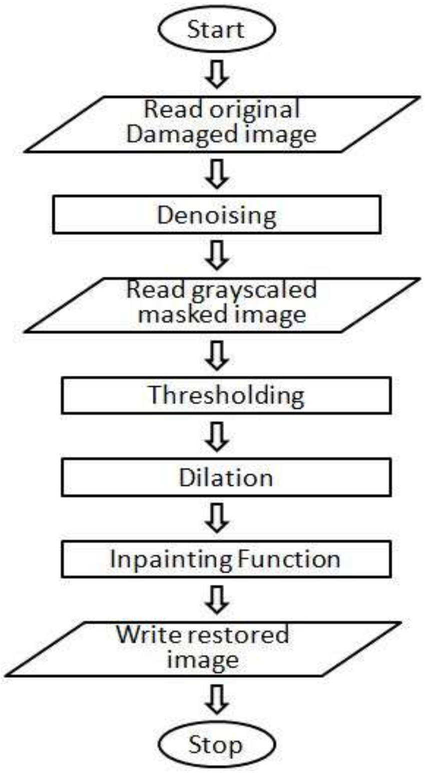

Figure 4: Flowchart of the program

5. Result

In this section we will get to know how the damaged or corrupted image is being developed to restore image in step by step format as defined in the flowchart section. The result is as following:

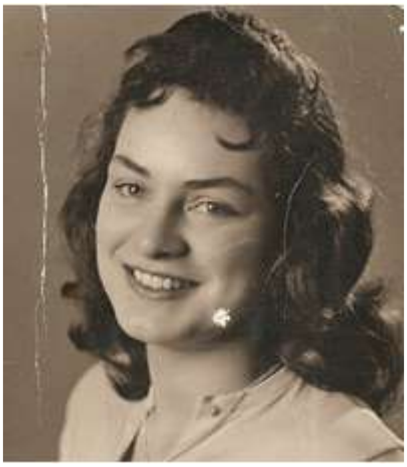

Figure 5: Original Damaged image

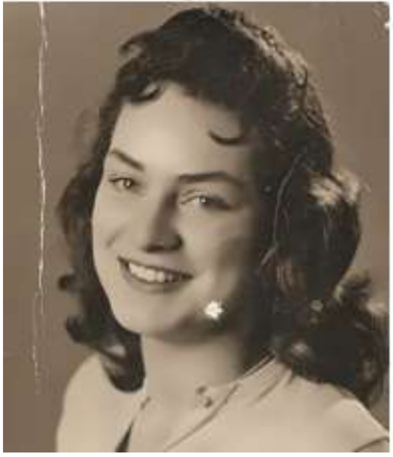

Figure 6: Denoised image

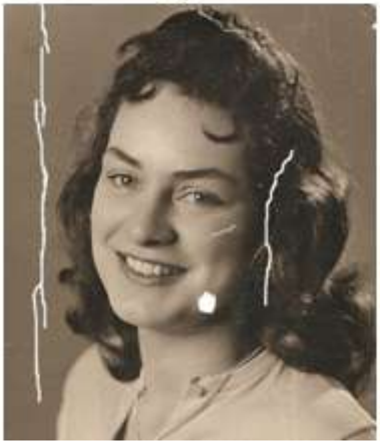

Figure 7: Mask image

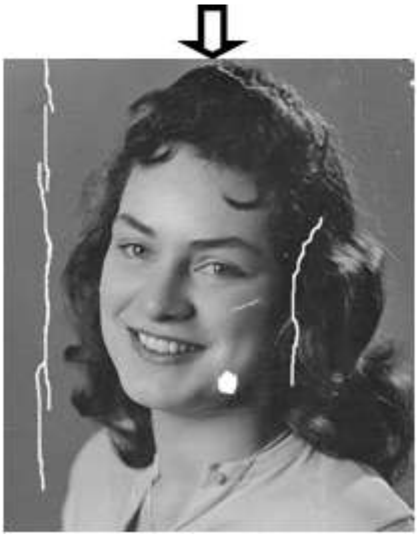

Figure 8: Grayscaled Mask image

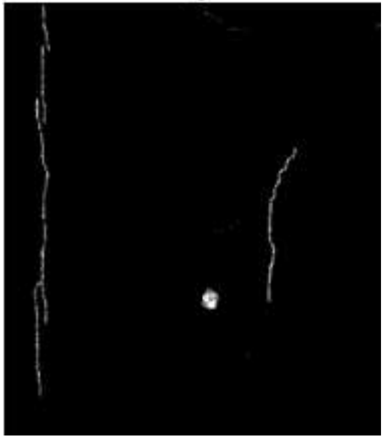

Figure 9: Threshold Binarization

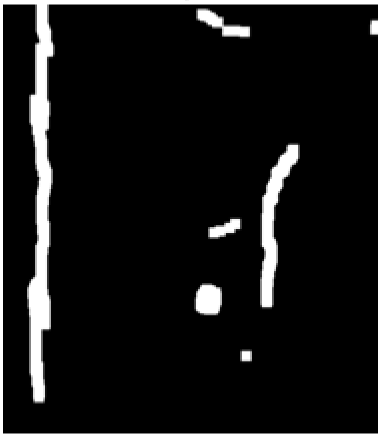

Figure 10: Threshold Binarized image after Dilation

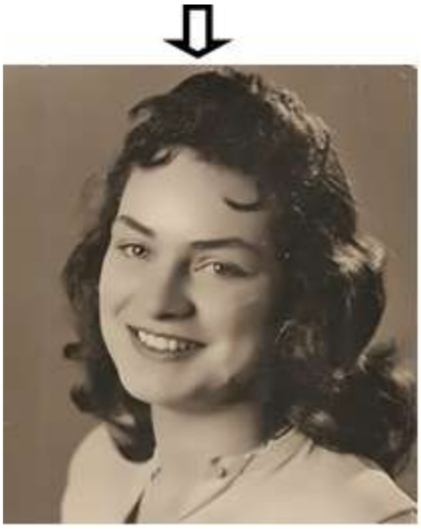

Figure 11: Restored image of damaged image

6. Conclusion

In this paper we discussed about Image restoration by the help of Inpainting function using the language of Python with the help of OpenCV software. This program gives us facility to restore image from damaged condition for archiving purpose of any old, torn photograph etc which can be considered as application of this program. We have successfully done the restoration of some photographs in our project, so we conclude that we can easily restore all types of damaged photograph using this effective and efficient program code.

References

[1] Nidhi, "Image Processing and Object Detection", International Journal of Applied Research 2015;1(9): 396-399

[2] Kruti Goyal, Kartikey Agarwal, Rishi Kumar , "Face Detection and Tracking", International Conference on Electronics, Communication and Aerospace Technology ICECA 2017

[3] Y ue Yaru, Zhu Jialin, "Algorithm of Fingerprint Extraction and Implementation Based on OpenCV", 2017 2nd International Conference on Image, Vision and Computing,978-1-5090-6238-6/1 7©20 17 IEEE

[4] Chaoxing Huang ,Dan Chen ,Xusheng Tang,"Implementation of Workpiece Recognition and Location Based on Opencv", 2015 8th International Symposium on Computational Intelligence and Design.

[5] M. Ashok Kumar, R.R. Tewari, "Human gestures and recognizing faces in natural expressions by considering computer vision techniques on mobile devices", Intelligent Computing and Control Systems (ICICCS), 2017 International Conference.

[6] Cristhian Rosales, Luis Jácome, Jorge Carrión, Carlos Jaramillo, Mario Palma, "Computer Vision for detection of body expressions of children with cerebral palsy", 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM).

[7] G.D. Illeperuma, D.U.J. Sonnadara, "Computer Vision Based Object Tracking as a Teaching Aid for High School Physics Experiments", Proc. EECSI 2017, Yogyakarta, Indonesia, 19-21 September 2017, 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI).

[8] Hao Jiang, Thomas Gonnot, Won-Jae Yi and Jafar Saniie, "Computer Vision and Text Recognition for Assisting Visually Impaired People using Android Smartphone ",2017 IEEE International Conference on Electro Information Technology (EIT).

[9] Aniket V. Patil, Mrinai M. Dhanvijay, "Engraved Character Recognition Using Computer Vision To Recognize Engine And Chassis Numbers", 2015 International Conference on Information Processing (ICIP).

[10] T. Arrighi , J. E. Rojas , J.C. Soto , C. A. Madrigal , J. A. Londoño, "Recognition and Classification of Numerical Labels Using Digital Image Processing Techniques ",2012 XVII Symposium of Image, Signal Processing, and Artificial Vision (STSIVA).