

ANDROID APP FOR RETOUCH OF SCRATCHES AND CURVES ON DIGITAL OLD PHOTOS

Executed by: Denys Hromovyi
Supervisor: Yuriy Bilogorodskyy

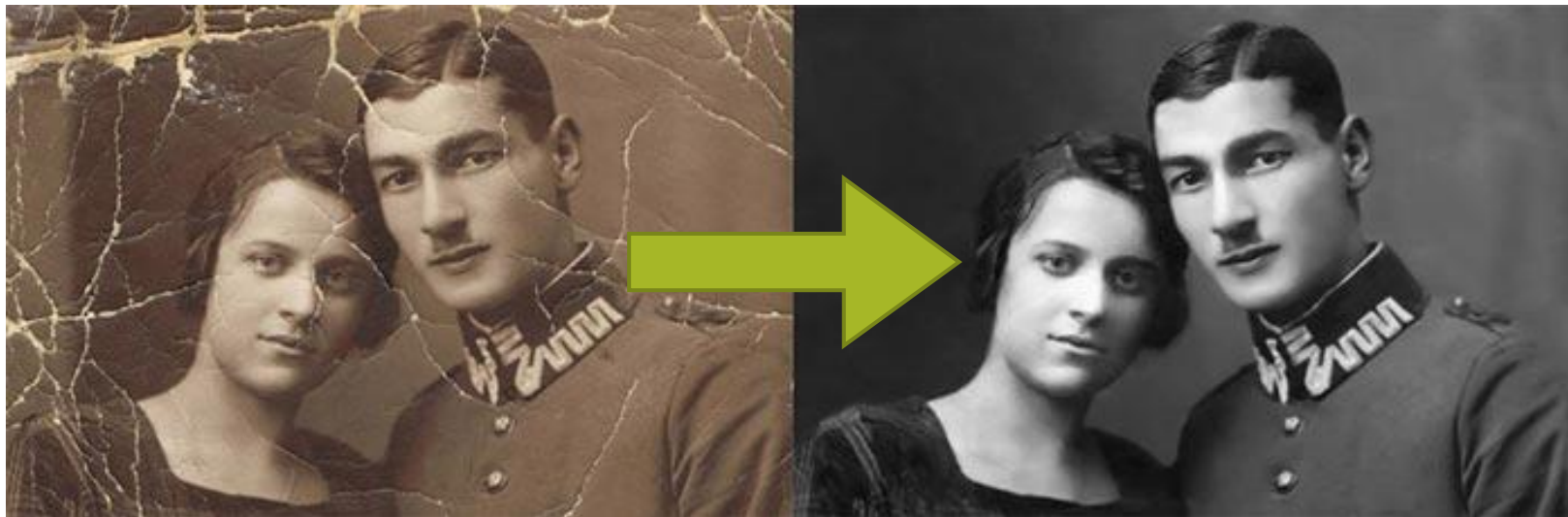


United Nations
Educational, Scientific and
Cultural Organization



Junior Academy of Sciences
of Ukraine

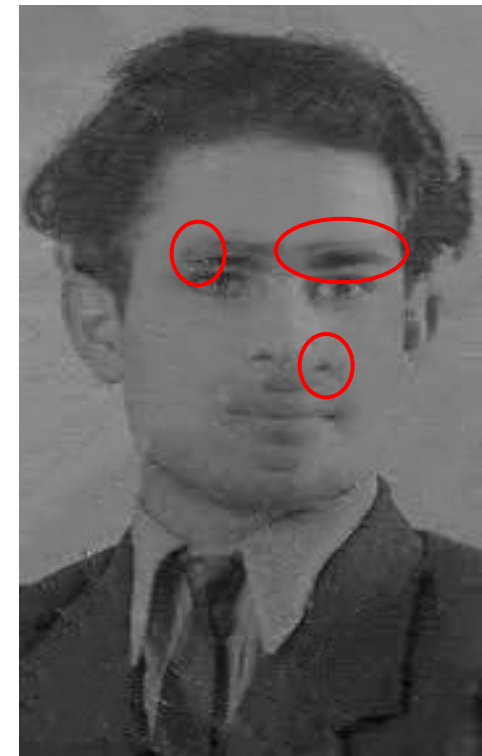
Our aim is to develop an android app that helps professional and amateur artists to restore old photos. Now the app focuses on restoring scratches and curves, which are white areas with lost information.



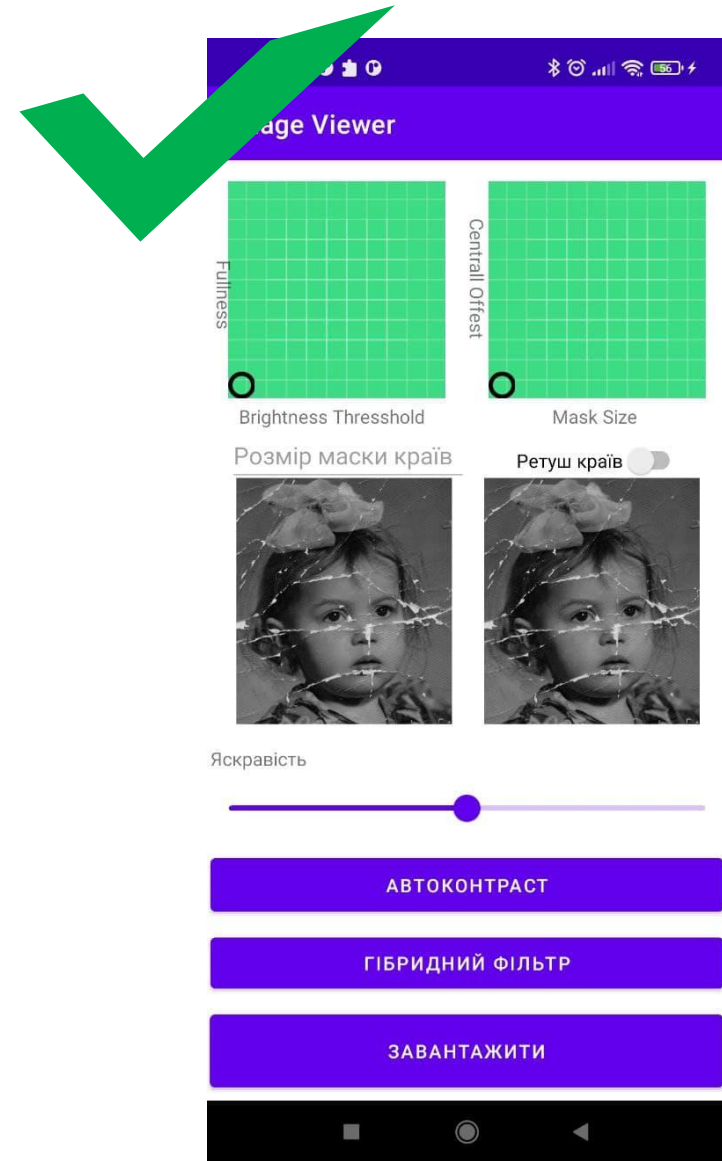
Sure Photoshop or something like that can cope with this problem properly. When those programs are used by amateurs, many artefacts appear, for example, details duplicating or losing.



Also, Photoshop analogues for android have an extra drawback: it is touch control, which doesn't allow to highlight the damaged area that causes bigger artefacts.



Our app won't have those issues. Vice versa, the key principles of our app are easy to use by amateurs as well as professionals and adaptation of user interface for android.



Phases of photo processing

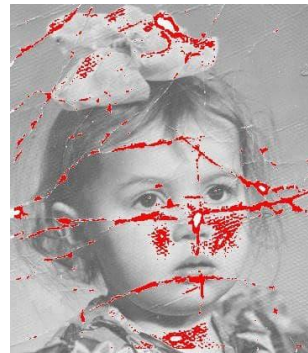
1. Over brightening of image for levelling brightness of damaged areas
2. Detecting of damaged areas based on their brightness
3. Restore corruption with hybrid filter
4. Detecting the edges of the corruptions
5. Blurring edges of the ccorruption with Gaussian filter



Original



1



2



3



4

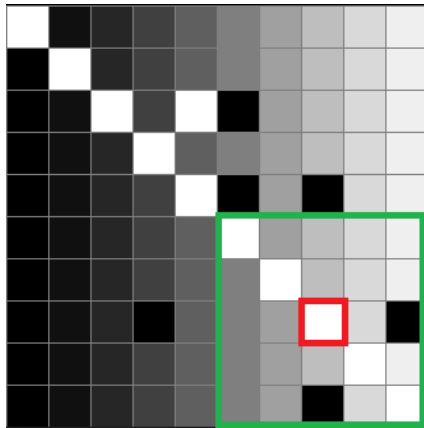


5

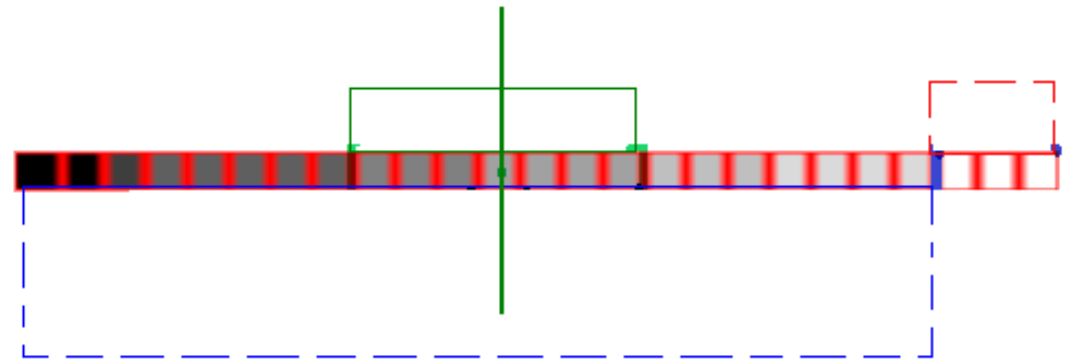


Corruption filling algorithm. Idea

How does it work: filter sorts an array of pixels, removes corrupted, and then finds arithmetic average some central pixels, in this way it removes impulse injections creates a gradient.

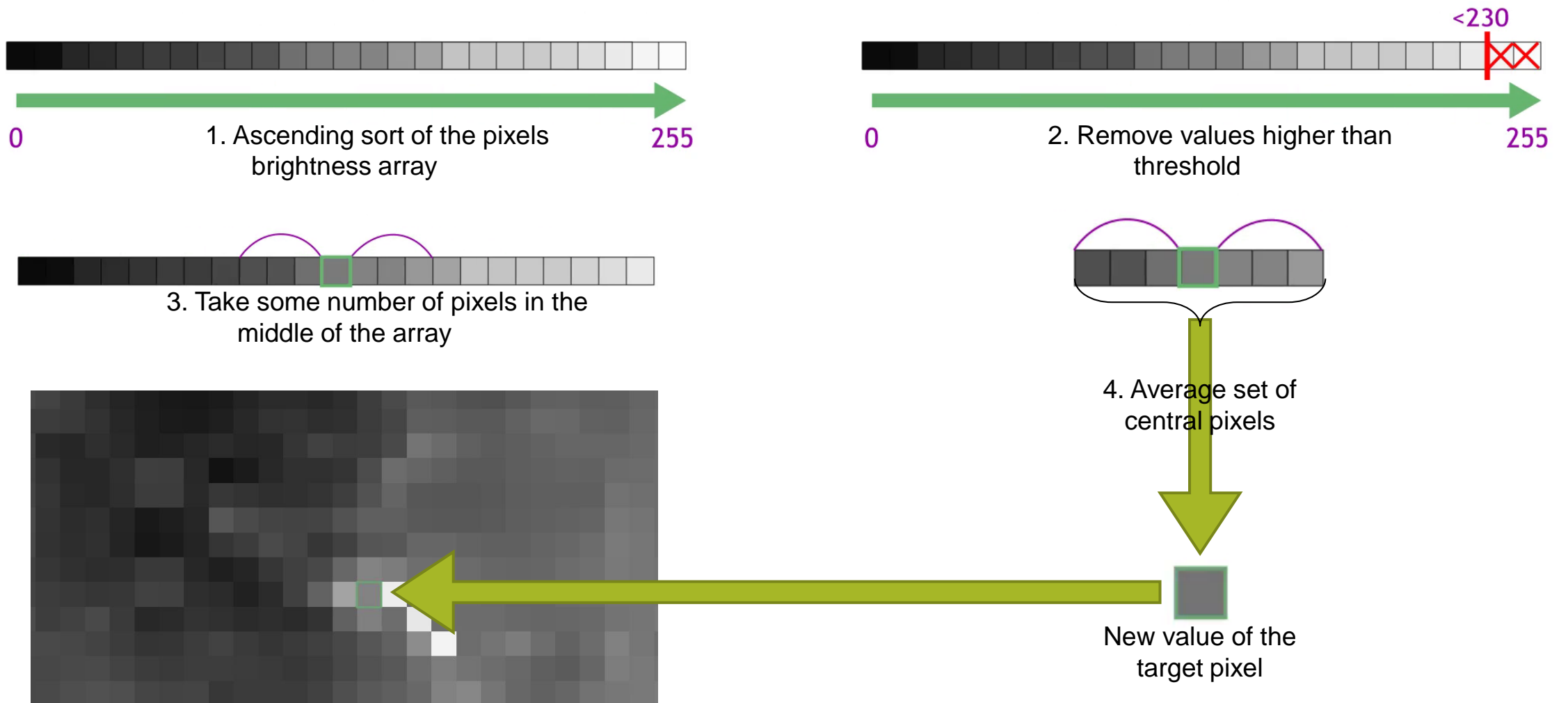


Filtration mask for white pixel (marked red), mask, which is used for restoration (marked green);

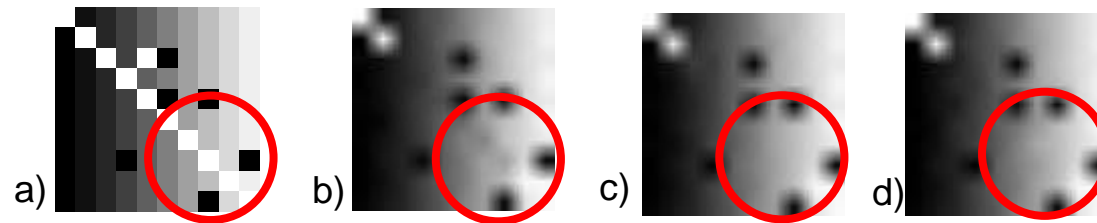


Sorted in ascending order of brightness by an array of pixels, to find new value only central pixels are used (green area) from all output pixels (blue area); corrupted pixels.

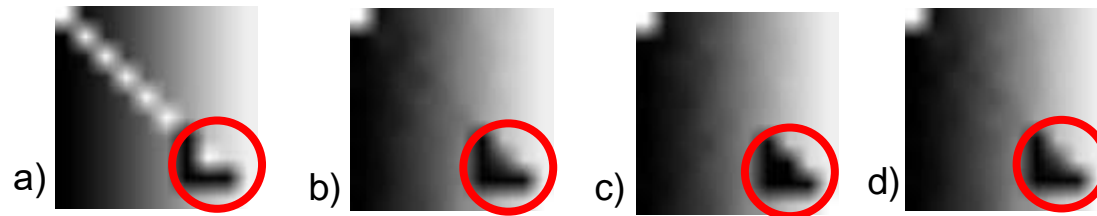
Corruption filling algorithm. Example



Corruption filling algorithm. Properties

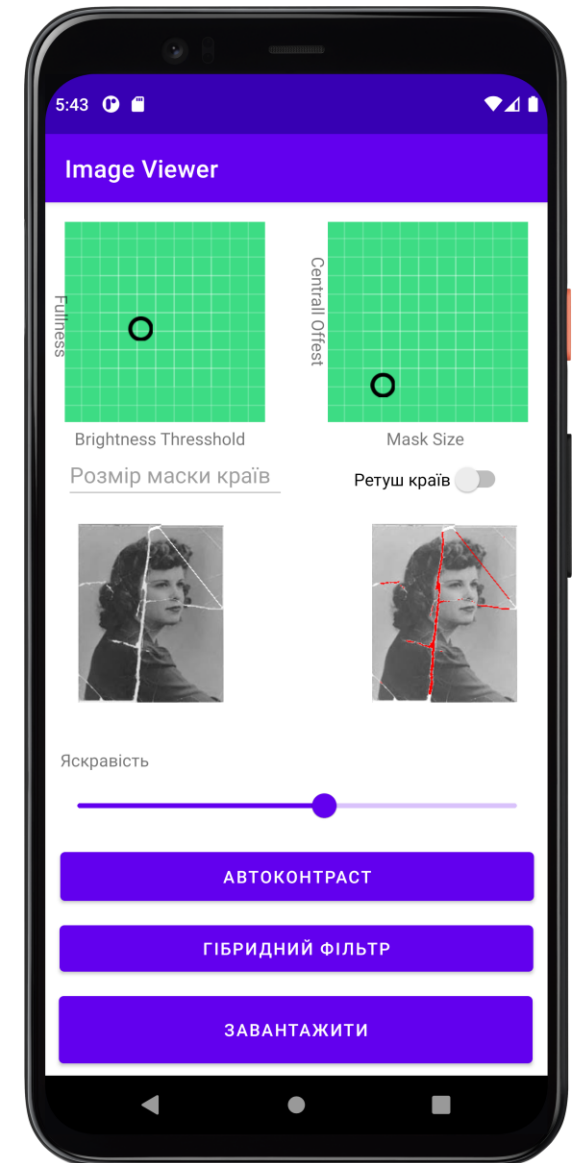
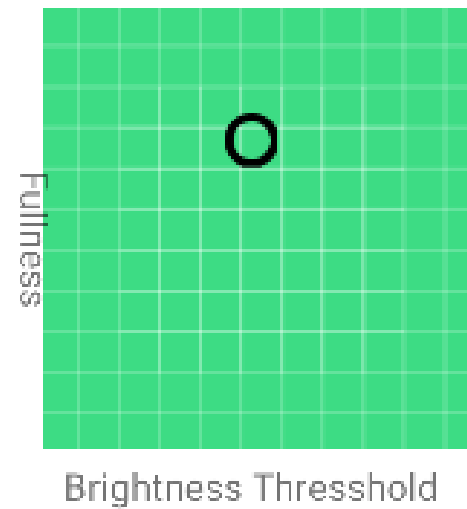


Case when median filter working better, at the same time our algorithm is working as well as median; a - original, b – Gaussian filter, c - Median filter, d – our hybrid filter.



Case when median filter working better, but at the same time our algorithm is working as well as Gaussian; a - original, b – Gaussian filter, c - Median filter, d – our hybrid filter.

Also, we have developed the author's method of regulation of settings by using special joysticks which allows us to find optimal parameters fast and watch the result in real-time.





Original



Picture processed in our app



Pictures processed in Photoshop Fix without preprocessing in our app. There are some artefacts such as duplicating or losing of details

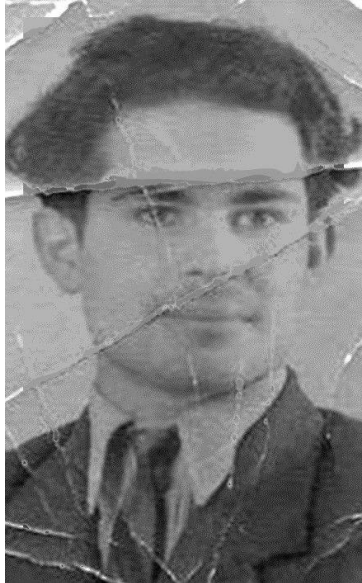


This one was also processed by Photoshop Fix, but with previous processing with our app



*

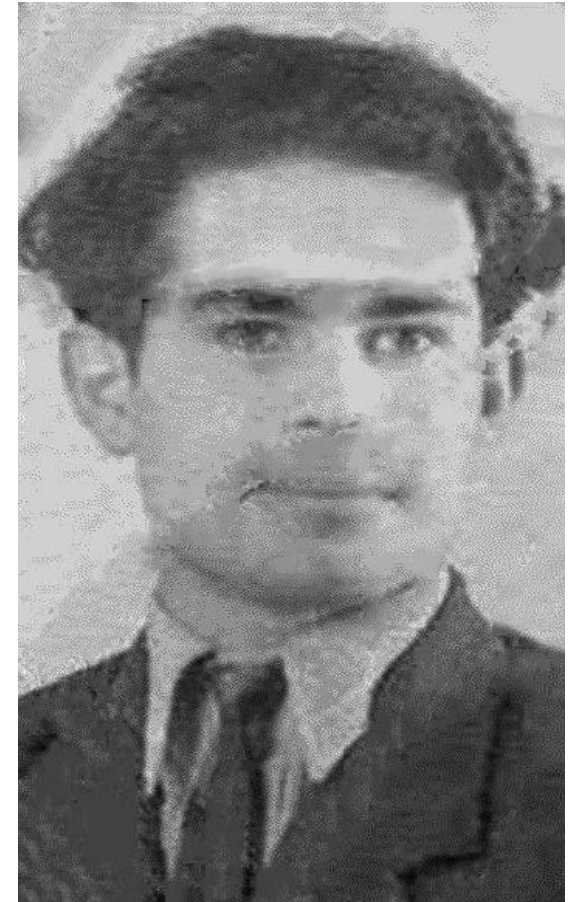
Original



Picture processed in our app



Pictures processed in Photoshop Fix without preprocessing in our app. There are some artefacts such as duplicating or losing of details



This one was also processed by Photoshop Fix, but with previous processing with our app

Other pictures processed in our app



Hotpot neural network

Using neural network you should be careful with photo resolution. Because as we can see below 198x198 photo got severe artifacts comparing to 512x512 one. In addition to that due to their nature such artifacts are very hard to fix with other tools like Photoshop.



198x198



512x512



Our app

For our approach it does not matter the resolution of an input image. As we can see below the results for 198x198 and 512x512 photos are almost the same. And it is easy to continue manual retouching in external software.



Original



198x198



512x512

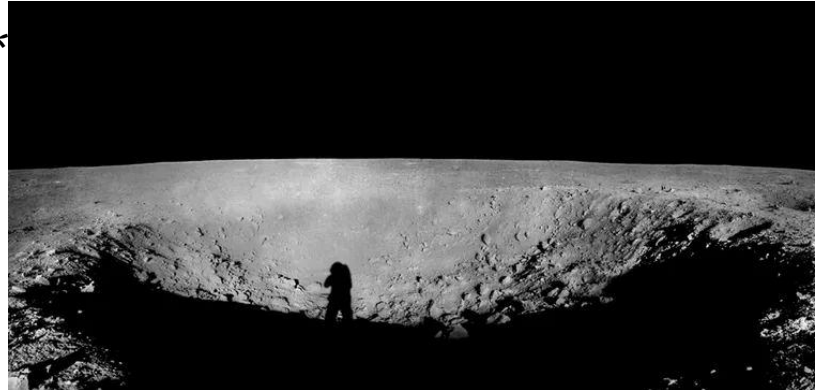


198x198

There is the picture from previous slide
additionally processed in Fotor web-based
photo editor

Conclusion

We hope that our project will make it easier and contribute in sharing old interesting pictures on the Internet.





**



*<https://bit.ly/3Ug1A2M>;
**<https://bit.ly/3xhXmMI>;
***<https://bit.ly/3QGpsrB>;

Thank you for attention