Improving the Resolution and the Readability of Two Dimensional Medical Images through Mathematical Transformations

Author: David Kamenov

Mentor: Assoc. Prof. Stanislav Harizanov

Institute of Mathematics and Informatics,

Bulgarian Academy of Science

Abstract

Digital image processing is a modern, extremely active and applicable scientific field that combines research techniques from both mathematics and informatics. Its applications are countless and can be useful in every aspect of human life related to digital images. Some of the most common applications include medicine, engineering, security, photography, archaeology, architecture and are extremely useful in the space exploration, mostly satellites and GPS technology.

The project “Improving the Resolution and Readability of Two Dimensional Medical Images through Mathematical transformations” is focused mainly in the field of bioinformatics and image processing. It aims improving the resolution and readability of images with medical purpose. Despite this, it is multipurpose and can be applicable in many other areas like non-medical images. The main idea is devoted to the important and still unsolved issue in the medical X-ray and CT imaging how to improve the image resolution and get better image readability without exposing the patient to higher radiation because the standard approach of solving this problem is to use higher radiation. The purpose is to lower the chance of doctor error caused by not enough information and to increase the chances of successful disease treatment while applying as little radiation as possible. Thus, it could help saving human lives.

The main goal is to automatize the process, and apply it to large datasets of images without the need of a programmer, doctor or a person to the whole. The improved data can be used for further analysis from specialists, educating inexperienced students or even improving a machine learning algorithm process or dataset.

Contents

1. Introduction
2. Problem Setting
3. Detailed Description
   3.1. Converting Input Image to Greyscale
   3.2. Converting the Greyscale Image to Matrix
   3.3. Extracting Variables from the Matrix
   3.4. Upsampling Algorithm
   3.5. Downsampling Algorithm
   3.6. Improving Image Readability Algorithm (Main)
      3.6.1. The Algorithm Formula
      3.6.2. The Way the Algorithm Works
   3.7. Converting Matrix to Image
   3.8. Edge Preservation (Future Development)
4. Results
5. Future Development
6. Conclusion
7. Acknowledgements
8. References

Input Image 200 x 200 px.

Output Image 800 x 800 px.

1. Introduction

The main purpose of the project is to improve the resolution and the readability of medical images without exposing the patient to higher radiation because it could be harmful or even fatal to the human. However, converting from low to high resolution leads to making image quality worse. Because of that, the applied algorithms not only improve the image resolution but they also improve the image readability.

As the project is in its early stage of development, the images being used are only for illustrative purposes and testing the algorithms. Because of that, no medical images were used. However, some of the images have some key elements of the medical ones as being rich in details as well as having very low resolution and poor image quality. In the future the project will be using actual medical images.

The photo above shows an input 200 x 200px (left picture) highly detailed pixelated image of a rose. After applying the algorithms used in this project, the image quality is better and resolution is improved to 800 x 800px (right picture). Consequently, the picture is easier to read to get general knowledge or even if applied to medical images (in the future), be used for diagnostics by specialists.

The example above shows that the algorithm can be applied not only for medical images but for a whole lot of other non-medical ones. The project is implemented in C# and features its own graphical interface, which means individuals without programming knowledge could use it. C# was chosen because it provides a great balance between speed, system resources usage, graphical interface and possibilities for future development.

2. Problem Setting

There are some problems which have been overcome for the purpose of the project. The sequence of events is as follow, in the next pages each of them is described in details:

1. Image to Greyscale: The input image is being converted to greyscale directly from the loading.

2. Greyscale to Matrix: Converting greyscale image to matrix because as every image-processing algorithm it works directly with two-dimensional arrays (matrixes) not with images.

3. Extracting Variables from the Matrix: The algorithm of extracting the value of every pixel and the eight pixels around it plays a key role because these 9 variables are used in the algorithms.

4. Upsampling Algorithm: Algorithm for increasing the resolution of the image.

5. Downsampling Algorithm: Algorithm which decreases the resolution of the image and is being used as a part of the improving the readability algorithm.

6. Improving Image Readability Algorithm: The main algorithm, which is intended to improve the readability of the image.

7. Matrix to Output Image: The matrix is converted to image and then displayed or saved.

3. Detailed Description

Here is applied the detailed description of the methods and algorithms used, which were briefly shown in section Problem Setting.

3.1. Converting Input Image to Grayscale

The first problem being faced is loading the image and converting it to greyscale. This is required because of the mainly medical purpose of the project where medical images are 99% greyscale. However, because of the multipurpose of the project, it is allowed to input greyscale as well as colourful images, which are being converted to greyscale. Furthermore, working with 8bit (256 shades of grey) in comparison to 16 million colour RGB images, makes the algorithm hundreds or even thousands times faster. Otherwise, working with colour images would require more computational power, would be less precise and much slower.

3.2. Converting the Grayscale Image to Matrix

After being converted to greyscale, the next step is converting the image to two-dimensional array (matrix). All image-processing algorithms are working with matrixes because they are flexible and easy to access. Because of that, all the algorithms are set to work with matrixes.

3.3. Extracting the Variables from the Matrix

	j-1	j	j+1
i-1	F	U	C
i	L	I	R
i+1	N	D	M

The used variables

Extracting the variables from the converted into matrix image is one of the most important tasks. Let us name the variables of a matrix 3x3 where:

- I = central pixel;
- F = up left pixel;
- U = up central pixel;
- C = up right pixel;
- R = right central pixel;
- M = down right pixel;
- D = central down pixel;
- N = down left pixel;
- L = central left pixel.

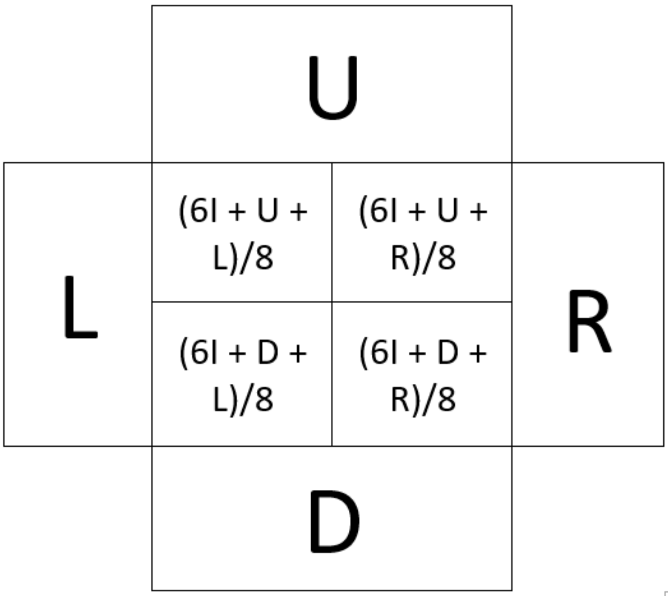
It features a matrix with two nested loops with iterators, respectively i and j and the central pixel (I) is being acquired directly by their coordinates. In the most common case when not only (I) but all of the others “exist” (are in the bounds of the matrix), it is easy to extract their value by standard matrix numeration where I = matrix[i,j] and the others are acquired by the typical way.

Unfortunately, not all the times all the variables “exist” so in the case that the out of bounds elements acquire the value of their closest neighbour. There are 8 possible situations where some variables are out of bounds: the four corners and the four walls.

Let us take the situation where I = matrix[0, 0] (the top left corner). Then variables F, U, C, L and M are out of bounds. In this case F, U and L as corner variables get the value from the closest corner consequently I, so F = I, U = I , L = I, and C and N get the value of the closest existing neighbour so C = R and N = D. The values of the rest of the variables could be set by the standard approach. An analogical procedure can be applied for the rest of the corners or walls by detecting the “non-existing” ones and giving them the value of their closest neighbour.

The algorithm is implemented to work not only for square images, but also for rectangle ones. Because of that, the values of image width and height are saved in variables which are used as number of iterations for the loops.

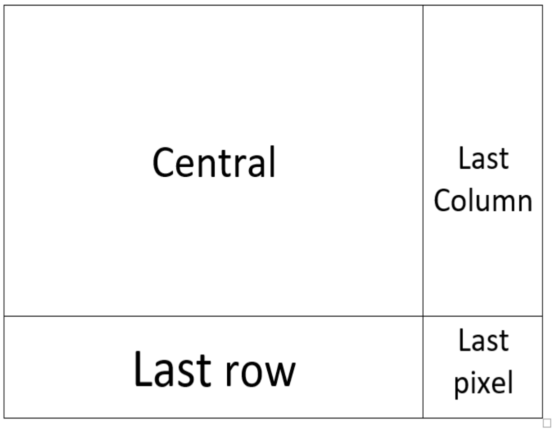
3.4. Upsampling Algorithm



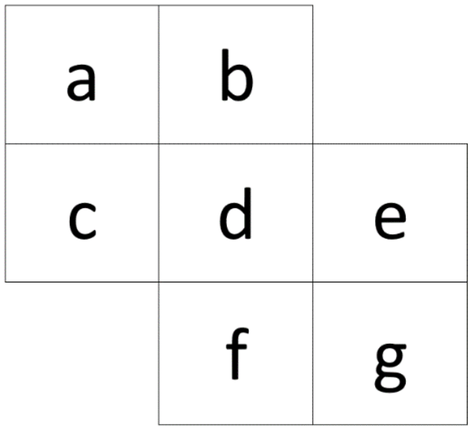
The algorithm for upsampling improves the resolution of each side of the image twice for every iteration. Therefore, if the input image is 16x16 after a single iteration it becomes 32x32, increasing the number of pixels by 4 times using the formula 2^2n where n is the coefficient of increasing the resolution. Upsampling uses the standard "Chaikin" approach, which converts every pixel to four using weighted average. We are using the standard notation explained in the algorithm for getting variables.

One iteration of the loop means the image resolution is doubled by each side which means improving the resolution four times. The upper graphic shows an example of five pixels where the central (I) is being divided to four pixels. After some tests it was concluded that the best coefficient of multiplying the pixel we are working on (I) is 6 plus once the two neighbour pixels all divided by 8.

3.5. Downsampling Algorithm



Downsampling Segments



Downsampling Notations

By applying mathematical transformations to the upsampling algorithm explained above and dividing twice the height and width of the new matrix, we get the downsampling algorithm. It is divided into four different segments of the matrix. It features the (I) working for every matrix[i, j], except last row = matrix[max row, j], last column = matrix[i, max col] and the last pixel = matrix[max row, max col].

Without applying the three edge segments (last row, last column, last pixel), which use different coordinates for “x” and “y”, the matrix will go out of bounds. Because of that, the area for downsampling is divided to the four mentioned areas.

Given the upsampling algorithm, we are computing its inverse mapping.

$$
\begin{aligned}
d &= \frac{6I + D + R}{8} && \Longrightarrow 6d - f - d = \frac{34I - 2M}{8} = \frac{17I - M}{4} \\
f &= \frac{6I + I + M}{8} && \Longrightarrow g - d = \frac{6M - 6I}{8} = \frac{3M - 3I}{4} \\
e &= \frac{6M + D + R}{8} && \Longrightarrow 3(6d - f - e) + g - d = \frac{51I - 3I}{4} = 12I \\
g &= \frac{6R + I + M}{8} && \Longrightarrow I = \frac{17d - 3e - 3f + g}{12}
\end{aligned}
$$

The upper formula shows the formula of the downsampling algorithm and how it was transformed from the upsampling algorithm. The same notation was used as in the algorithm for extracting variables. However, there are some new ones which are shown on the graphics “Downsampling notations” whose value could be acquired analogically.

3.6. Improving Image Readability Algorithm (Main)

The algorithm for improving image readability is the most important one. It is based on mathematical transformations and represents greatest interest for us. It aims to improve image's readability by the fastest and most accurate way possible.

3.6.1. The Algorithm Formula

$$matrix[i,j] = \lambda[\widetilde{h} * (\bar{g} - T(h * u^n))]_{i,j} +$$

$$\frac{\epsilon + (U-D)^2(R-2I+L) - (\epsilon + (R-L)^2(U-2I+D)) - \frac{1}{2}(R-L)(U-D)(C-F-M+N)}{4h_x^2 h_y^2 \epsilon + h_x^2(U-D)^2 + h_y^2(R-L)^2}$$

3.6.2. The Way the Algorithm Works

1) The input image is being upsampled
2) From it is applied a convolution (a matrix +, - , *, / another matrix) between the upsampled minus downsampled then upsampled image.
3) Deburring is applied
4) The output is saved as the new (I) and applied the algorithm is applied for all the other pixels.
5) This is being applied numerous times.

3.7. Matrix to Image

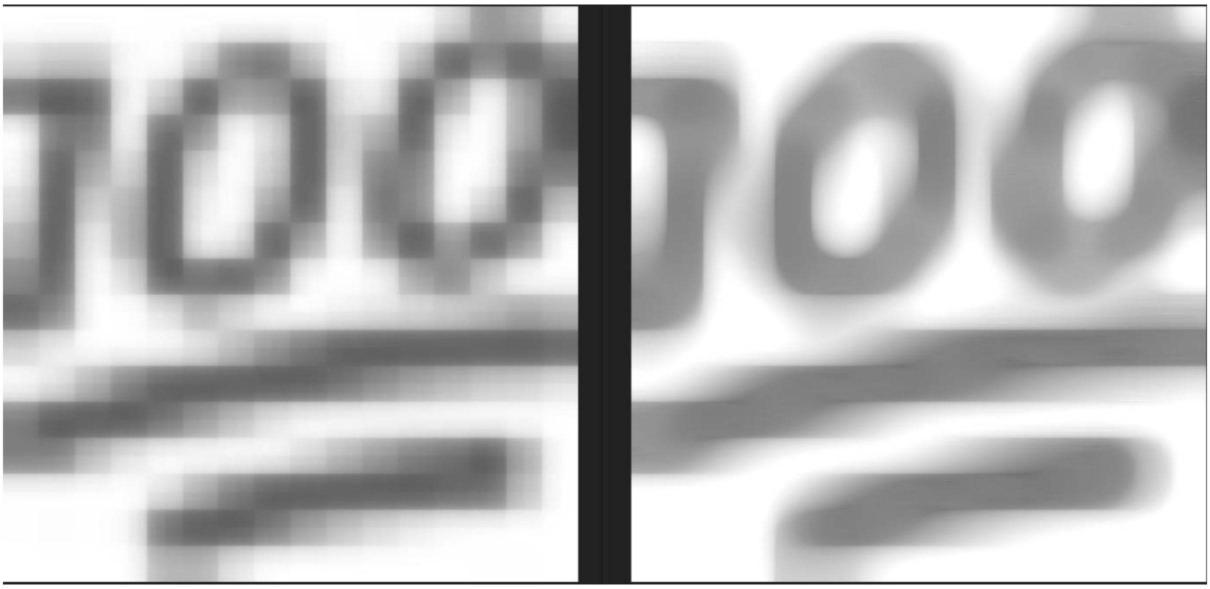
The output from the algorithms is being saved in a form of a matrix. Afterwards, the matrix is being converted to image. This image could now be displayed or saved depending on the needs of the user.

3.8. Edge Preservation(Future Development)

Now, the project is using standard “Chaikin” approach, which means improving the resolution by weighted average. The downsampling is formed form mathematical transformations of the up sampling so it is again a form of weighted average backwards. Because of that in section “Results”, could be seen that the edges of the output images are a little bit blurred. Because of that, in the future is going to be implemented an extremely significant algorithm for edge preserving Up (Down) sampling. When it is being implemented, the image is not going to be blurred at the edges and they will be kept high in contrast and highly detailed which is extremely necessary for medical purposes.

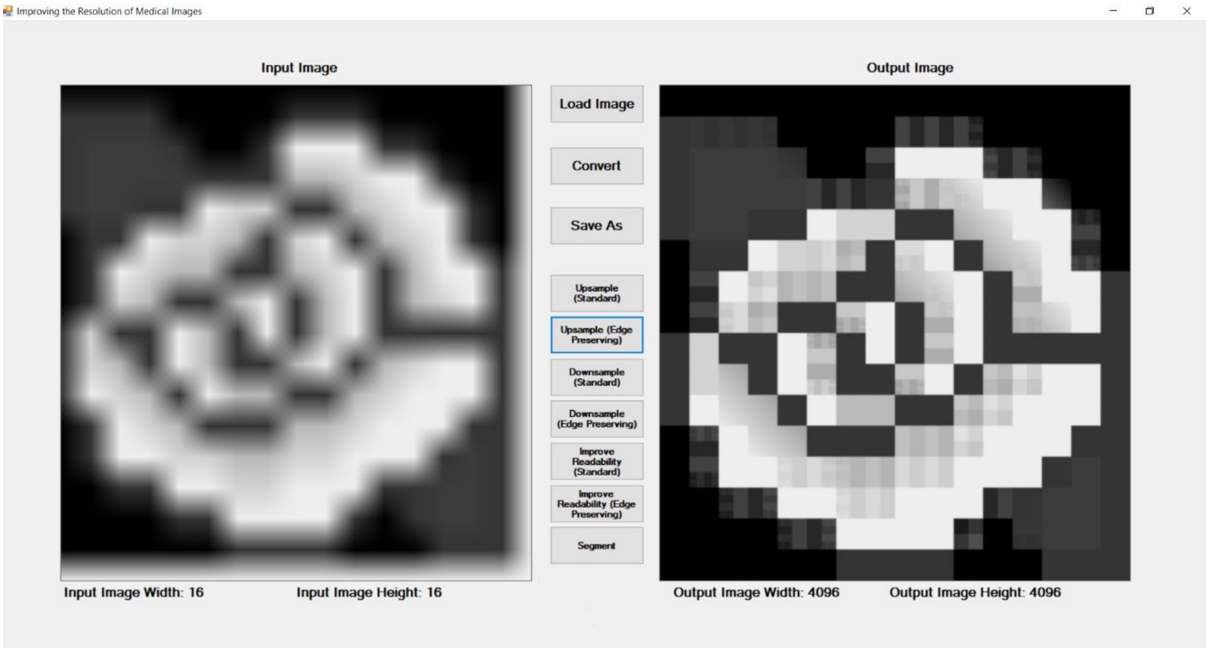
4. Results

Below are shown the input images (left) and output images (right):

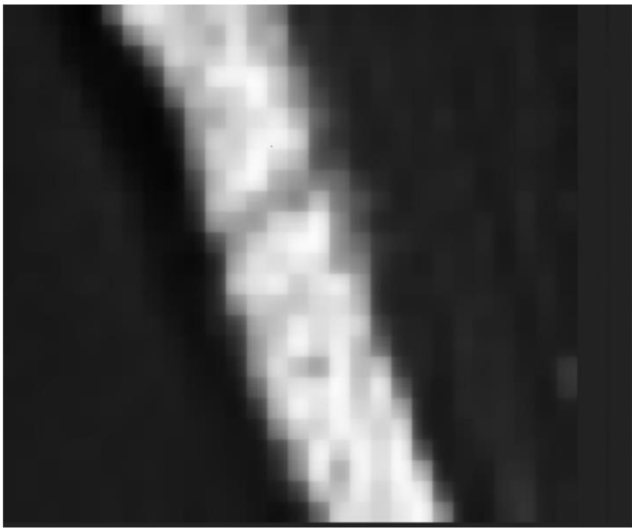


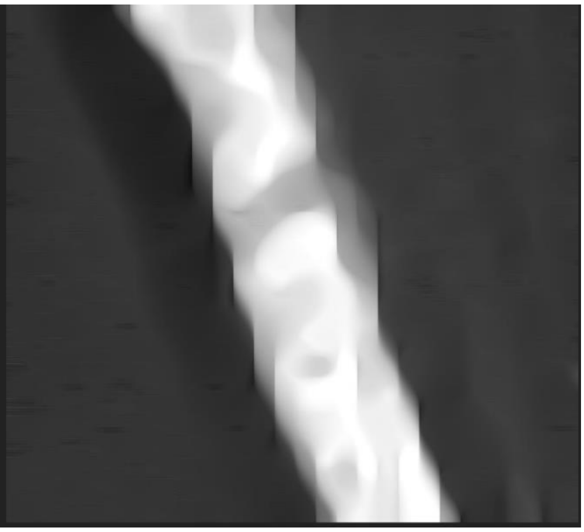
Input 16 x 16px (displayed as 1024 x 1024 only for visual purposes) Output 1024 x 1024

In the upper left picture could be seen the input image (originally 16 x 16px which the algorithm uses as an input) which is being converted to 1024 x 1024 just for comparison with the result and visual purposes. It shows what will happen if only the resolution is being improved without improving the quality. In the right could be seen the image after applying the algorithms from the project on this photo. The readability is much better and it is much easier to determine what is being displayed. Even, it would be easier for a machine learning algorithm to determine the object, not only for a person.



Desktop Application of the Project


Input 25 x 28px.


Output 1600 x 1792px.

Input 50 x 50px

Output 800 x 800px

Input 16 x 24px.

Output 1024 x 1536 px.

Output 1024 x 1536 px.

200 x 200 px. Input Image

800 x 800 px. Output Image

This project proves that for sure, the image resolution could be improved while preserving the same quality. However, „Improving the Resolution and Readability of Two Dimensional Medical Images through Mathematical Transformations” aims not only to improve the resolution of an image with medical or non-medical purpose but also to improve its readability, so that it is easier and more precise what is being displayed there. Thus, the doctors or specialists could be able to create a precise diagnostics and treat more people. The obtained results show that in some scenarios, it is working well but in some, it could be better. After implementing the edge preservation algorithm the results should be much better because the edges would not be blurred and they will be high contrast. Furthermore, the proportions of the image object would stay the same without changing them as seen on the image with the square root of five.

5. Future Development

- Edge Preservation Algorithm: To begin with the most important feature that should be added is as mentioned earlier the edge preservation Up (Down) Sampling. After applying it, the edges would not be blurred and the image is going to be more visually appealing and more precise for a diagnostics and statistics.

- Anti-Aliasing Algorithm: The next feature that could show an improvement is implementing an algorithm, which applies anti-aliasing to the input image and after every iteration of the algorithm. Thus, it is believed that the edges would be even smother without blur.

- Using Actual Medical Images: After implementing these algorithms, the project should be ready to work with actual medical images. As seen, now it is working with some images just for visual purposes and for testing the algorithms. In the future actual medical images would be applied.

- Implementing a Variable Adjusting Algorithm: Implementing an algorithm which adjusts the variables depending on the input image is something of key role because it is believed that the algorithm will perform much better and faster having it.

6. Conclusion

Despite the short development time, the results of this project are promising. As "Improving the Resolution and Readability of Two Dimensional Medical Images through Mathematical Transformations" is applicable in all aspects of digital image processing, not only in the field of medicine, it can be concluded that working on it as future development is really worth it.

There are innovations applied in this project and that differentiate it from the others with this purpose. To begin with, standard readability (or quality) improvement algorithms rely on machine learning algorithms and huge datasets for this purpose. They are performing well in everyday images like ones captured with a smartphone or a camera because of the enormous dataset they have. However, when a medical image is converted using such software like "TopazAI" for example, the results are unrecognizable and useless. This is caused by the fact that the machine learning have not been trained with medical images dataset. However, it is very hard or even impossible to create such algorithm universal because the medical datasets are extremely expensive and difficult to find.

Because of that, "Improving the Resolution and Readability of Two Dimensional Medical Images through Mathematical Transformations" improves the resolution and most importantly the readability through Mathematical transformations which make it absolutely universal and multipurpose. It could work on all kinds of medical images and also different photos from the everyday life, photography, satellites and GPS technology and others.

Furthermore, during the development of this project it was concluded that it is not very efficient to apply the same blur level on photos with different resolution. Because of that, it was discovered experimentally that a dynamic blur and deburring coefficient, which changes with every iteration, performs much better in compression to the static one. Because of that the project, "Improving the Resolution and Readability of Two Dimensional Medical Images through Mathematical Transformations" uses dynamic coefficient of blur and deburring.

The project is intended to be implemented in real X-rays, CT scanners, echo graphs and it will be provided for commercial use as a desktop application in the future or in the form of a website to be accessible for everyone, not only for doctors and for specialists. Finally yet importantly, the resolution and readability improvement algorithm could be used for enhancing the datasets for machine learning algorithms.

7. Acknowledgements

- I would like to thank my mentor Assoc. Prof. Stanislav Harizanov for proposing me this project theme, for the guidance and invaluable advice and for inspiring me to take up bioinformatics and image processing. Without his priceless help, this project would have never been realised.

- I would like to thank Assoc. Prof. Jenny Sendova for introducing me to my mentor and inspiring me to create a project in this field.

- I would like to thank Victor Kolev for the invaluable guidance and helpful advice both with the algorithm and the paper formulations because without him I would not have achieved these results.

- I would like to thank to my Informatics teacher Ilian Tsenkov for inspiring me to take up computer science.

- I would also like to thank I my Mathematics teacher Dora Stoyanova who inspired my love for mathematics.

- I would like to thank and HSSIMI for giving me the opportunity to participate in SRS19 where I started this project.

- Last but not least I would like to thank Stoyan Velev, Petar Ivanov and SAP for the opportunity to take part in SAP GeekyCamp which helped me develop my project

8 **References**

- Yue, Linwei Shen, Huanfeng Li, Jie Yuan, Qiangqiang Zhang, Hongyan Zhang, Liangpei. (2016). Image super-resolution: The techniques, applications, and future. Signal Processing. 128. 10.1016/j.sigpro.2016.05.002.
- http://webhelp.esri.com/arcgisdesktop/9.2/body.cfm?tocVisable= 1&ID=-1&TopicName=Using%20the%20convolution%20filter%20process