Huffman Encoding

Mostafa Mohamed Essam
mostafa.mohamed00@eng-st.cu.edu.eg

Habiba Mahmoud Abdelhalim
habiba.ahmed98@eng-st.cu.edu.eg

Mo'men Maged Mohamed
momen.rizk99@eng-st.cu.edu.eg

Mariam Mohamed Osama
mariam.hamed99@eng-st.cu.edu.eg

Anas Mohamed Abdelrahman
fakestar.anas19019@gmail.com

1. Introduction

In 1951, the MIT student David A. Huffman was working on his term paper which his professor gave to Huffman's class for whom didn't want to take the final exam. The Problem that was assigned to them was to find the most effecient binary code. Huffman got the idea to use a frequency sorted binary tree and proved that his method is the most efficient.[1]
Huffman coding assigns binary codes to character in such a way that the code's lenght depends on the frequency/weight of its corresponding character. The codes have variable lenghths, and they're prefix-free which means that any code isn't a prefix of any of the other codes.[2]
The Huffman tree is a binary tree in which any leaf in the tree corresponds to the character/data that is being coded.
We have managed to utilize this technique to compress images with ".pgm" format by encoding each pixel in the image according to its frequency thus storing it in less number of bits.

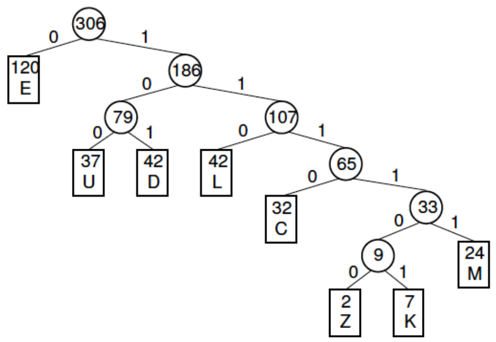


Figure 1. Example of a coded Huffman Tree

2. Motivation

We were really motivated and enthusiastic about this project cause it helped us learn a lot of new things:

1. The concept of file compression and how much useful it is.
2. How to read/write files using `C++`.
3. How to handle command line arguments and flags.
4. Storing data in binary files.
5. Learned how useful some data structures like hashmaps are.
6. How to make user friendly GUI with a descent UI.

3. Resources

We used the pgmb_io library to read and write the pgm file. Its written mainly in c, we added more c++ functionalities to make it compatible with our application and we removed the unneeded functions in it.

4. Challenges and Problems

1. Learning how to use fstream functionalities.
2. Learning how to build a GUI.
3. Converting data to bitsets.

5. System Block Diagram

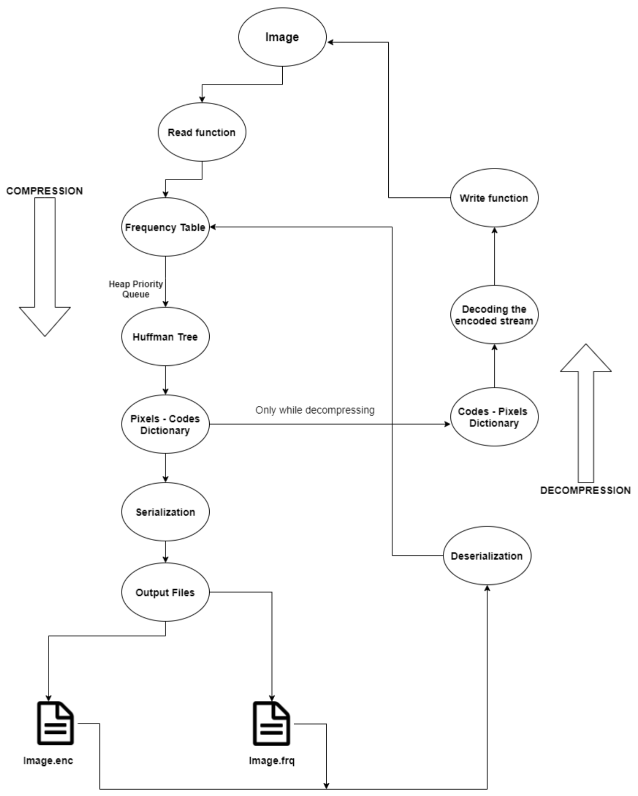


Figure 2. System Block Diagram

6. User manual for the system

6.1. Command Line Program

6.1.1 Windows

1. Open the CMD by pressing " ⊞ + R" and typing cmd in the "Run" window then pressing enter.
2. Choose the directory of the app by navigating through "cd" command. i.e C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app
3. Build an executable file by typing the command "g++ *.cpp *.hpp -std=c++17 -o Huffman.exe -O3" on the command prompt.
4. For compression type "Huffman.exe image_name.pgm" if the image in the same folder with the app or "Huffman.exe 'image_path' " to select any image on the hard disk. Two new files will be created in the folder in which the image is located 2 more files "image_name.enc" which is the encoded image and "image_name.frq" which is the frequency table of the pixels.
5. for decompression type in the cmd "Huffman.exe -t image_name.frq image_name.enc" or the path of each file like in compression.
   The decompression process will return the original image with its original size.

6.1.2 On Linux

Check the readme.txt file attached in the GitHub repository.

6.2. GUI

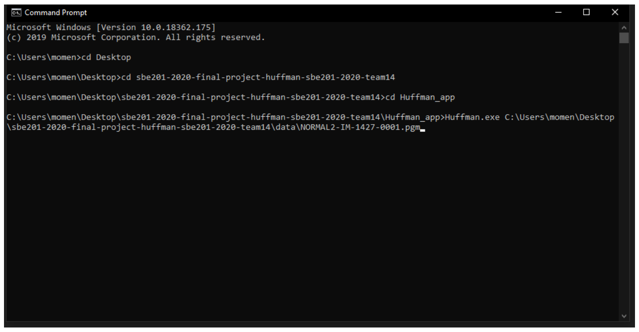
1. Open Qt creator then choose the option to open a new project.
2. Choose the CMakeLists.txt file and configure it.
3. Run the code and the GUI window will open.
4. For compression:
   (a) Click on the compress button.
   (b) Choose a file with a .pgm format.
   (c) Click open.
5. For decompression:
   (a) Click on the decompress button.
   (b) Choose 2 files one of them with a .enc format and the other with .frq format.
   (c) After selecting the two files click open.

7. Results

7.1. Command Line

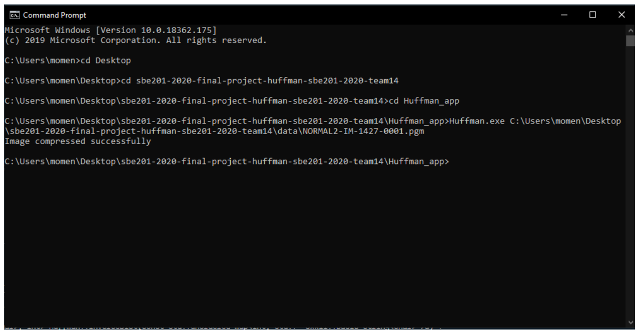
7.1.1 On Windows

1. Compression:



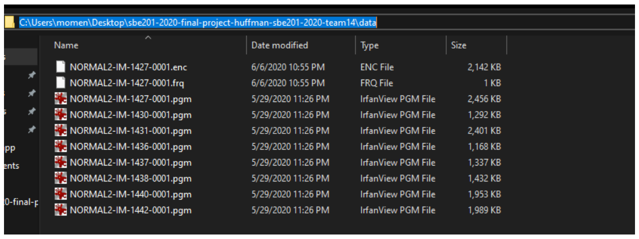
```
Command Prompt
Microsoft Windows [Version 10.0.18362.175]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\momen>cd Desktop

C:\Users\momen\Desktop>cd sbe201-2020-final-project-huffman-sbe201-2020-team14

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14>cd Huffman_app

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe C:\Users\momen\Desktop
\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.pgm
```

Figure 3. Passing command line arguments



```
Microsoft Windows [Version 10.0.18362.175]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\momen>cd Desktop

C:\Users\momen\Desktop>cd sbe201-2020-final-project-huffman-sbe201-2020-team14

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14>cd Huffman_app

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe C:\Users\momen\Desktop
\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.pgm
Image compressed successfully

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>
```

Figure 4. After running the program

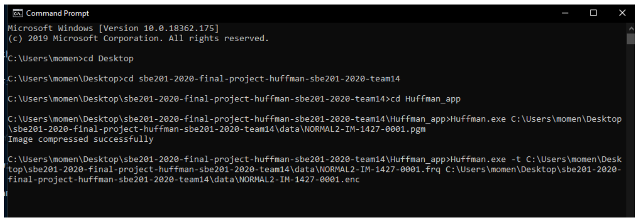


C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\data

Name	Date modified	Type	Size
NORMAL2-IM-1427-0001.enc	6/6/2020 10:55 PM	ENC File	2,142 KB
NORMAL2-IM-1427-0001.frq	6/6/2020 10:55 PM	FRQ File	1 KB
NORMAL2-IM-1427-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	2,456 KB
NORMAL2-IM-1430-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,292 KB
NORMAL2-IM-1431-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	2,401 KB
NORMAL2-IM-1436-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,168 KB
NORMAL2-IM-1437-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,337 KB
NORMAL2-IM-1438-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,432 KB
NORMAL2-IM-1440-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,953 KB
NORMAL2-IM-1442-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,989 KB

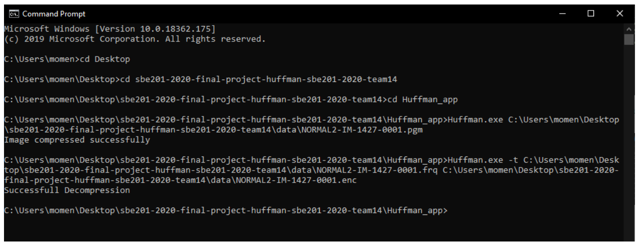
Figure 5. The result files

2. Decompression:



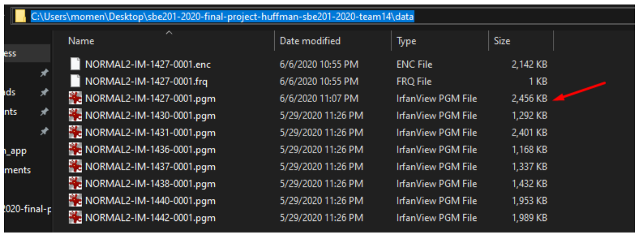
```
Microsoft Windows [Version 10.0.18362.175]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\momen>cd Desktop

C:\Users\momen\Desktop>cd sbe201-2020-final-project-huffman-sbe201-2020-team14

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14>cd Huffman_app

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe C:\Users\momen\Desktop
\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.pgm
Image compressed successfully

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe -t C:\Users\momen\Desk
top\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.frq C:\Users\momen\Desktop\sbe201-2020-
final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.enc
```

Figure 6. Passing the arguments



```
Microsoft Windows [Version 10.0.18362.175]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\momen>cd Desktop

C:\Users\momen\Desktop>cd sbe201-2020-final-project-huffman-sbe201-2020-team14

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14>cd Huffman_app

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe C:\Users\momen\Desktop
\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.pgm
Image compressed successfully

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>Huffman.exe -t C:\Users\momen\Desk
top\sbe201-2020-final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.frq C:\Users\momen\Desktop\sbe201-2020-
final-project-huffman-sbe201-2020-team14\data\NORMAL2-IM-1427-0001.enc
Successfull Decompression

C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\Huffman_app>
```

Figure 7. After running the program



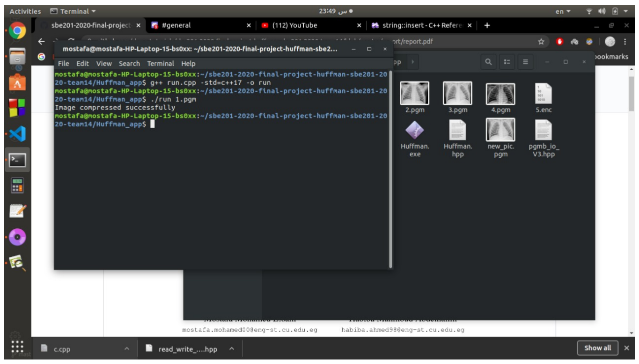
C:\Users\momen\Desktop\sbe201-2020-final-project-huffman-sbe201-2020-team14\data

Name	Date modified	Type	Size
NORMAL2-IM-1427-0001.enc	6/6/2020 10:55 PM	ENC File	2,142 KB
NORMAL2-IM-1427-0001.frq	6/6/2020 10:55 PM	FRQ File	1 KB
NORMAL2-IM-1427-0001.pgm	6/6/2020 11:07 PM	IrfanView PGM File	2,456 KB
NORMAL2-IM-1430-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,292 KB
NORMAL2-IM-1431-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	2,401 KB
NORMAL2-IM-1436-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,168 KB
NORMAL2-IM-1437-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,337 KB
NORMAL2-IM-1438-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,432 KB
NORMAL2-IM-1440-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,953 KB
NORMAL2-IM-1442-0001.pgm	5/29/2020 11:26 PM	IrfanView PGM File	1,989 KB

Figure 8. Original image

7.1.2 On Linux

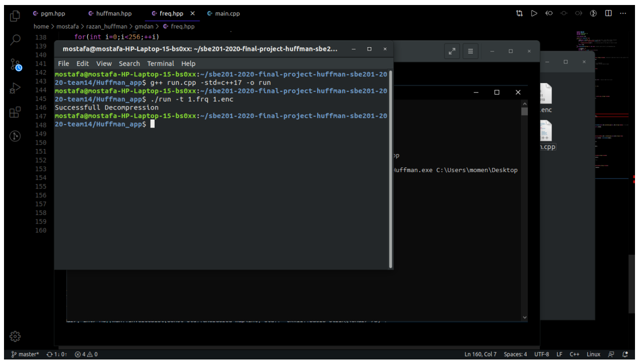
1. Compression



```
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$ g++ run.cpp -std=c++17 -o run
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$ ./run 1.pgm
Image compressed successfully
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$
```

Figure 9. Passing command line arguments and running the program

2. decompress



```
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$ g++ run.cpp -std=c++17 -o run
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$ ./run -t 1.frq 1.enc
Successfull Decompression
mostafa@mostafa-HP-Laptop-15-bs0xx:~/sbe201-2020-final-project-huffman-sbe201-20
20-team14/Huffman_app$
```

Figure 10. Passing command line arguments and running decompression

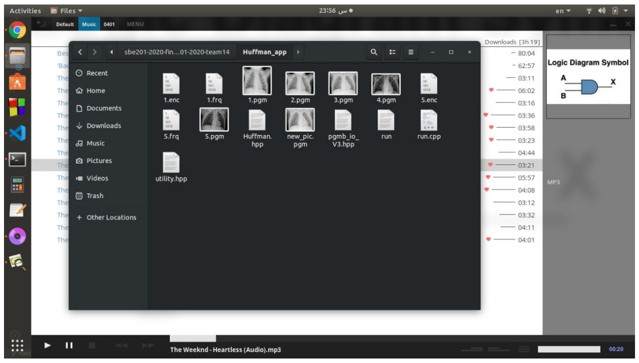
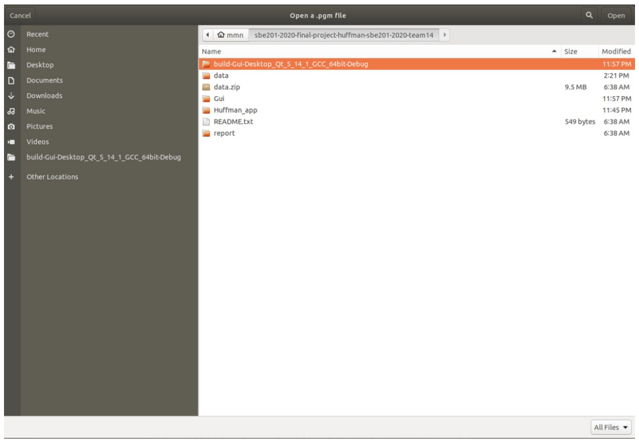


Figure 11. Resulting files

7.2. GUI

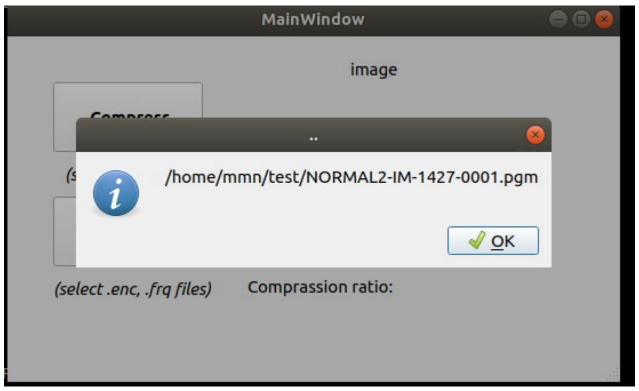
1. Compression



mmn / sbe201-2020-final-project-huffman-sbe201-2020-team14

Name	Size	Modified
build-Gui-Desktop_Qt_5_14_1_GCC_64bit-Debug		11:57 PM
data		2:21 PM
data.zip	9.5 MB	6:38 AM
Gui		11:57 PM
Huffman_app		11:45 PM
README.txt	549 bytes	6:38 AM
report		6:38 AM

Figure 12. Browsing files to choose a .pgm file



/home/mmn/test/NORMAL2-IM-1427-0001.pgm

Figure 13. You chose a valid file

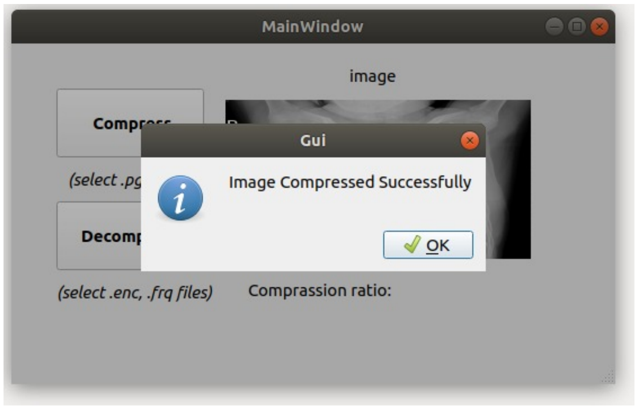
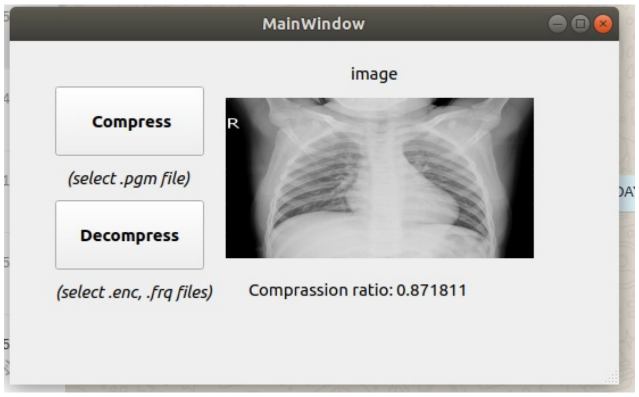


Image Compressed Successfully

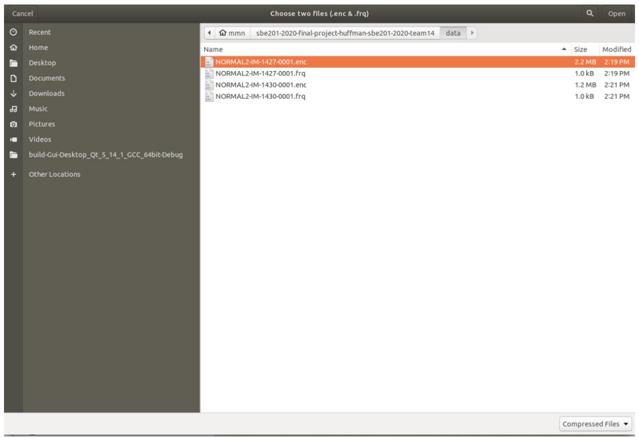
Figure 14. successful compression



Comprassion ratio: 0.871811

Figure 15. Final Window

2. Decompression



mmn / sbe201-2020-final-project-huffman-sbe201-2020-team14 / data

Name	Size	Modified
NORMAL2-IM-1427-0001.enc	2.2 MB	2:19 PM
NORMAL2-IM-1427-0001.frq	1.0 kB	2:19 PM
NORMAL2-IM-1430-0001.enc	1.2 MB	2:21 PM
NORMAL2-IM-1430-0001.frq	1.0 kB	2:21 PM

Figure 16. Decompression Browse

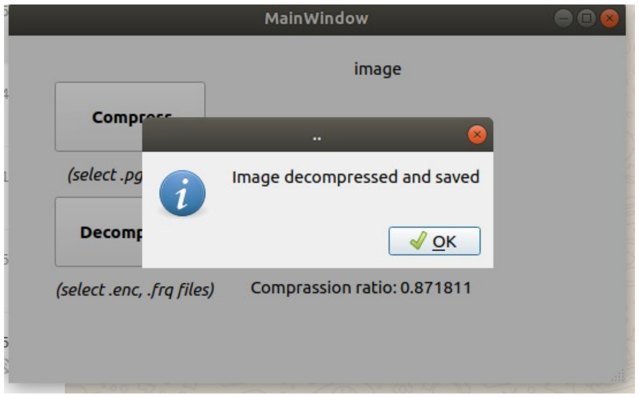


Image decompressed and saved

Compression ratio: 0.871811

Figure 17. Image Decompressed message box

8. Contributions

1. Mostafa Essam:
   (a) Read/write imagefiles in ".pgm" format.
   (b) Serializing the encoded image and frequency table.
   (c) Deserializing the frequency table.
   (d) Documentation
2. Habiba Mahmoud:
   (a) Construct a frequency table from the image.
   (b) From the frequency table,construct Huffman tree with the help of heap priority queue (Huffman Encoding).
   (c) Generate the codes for each pixel.
   (d) Documentation.
3. Mo'men Maged:
   (a) Deserializing the encoded image and frequency table.
   (b) Decoding the encoded image data.
   (c) Arguments flag.
   (d) Report.
4. Mariam Mohamed Osama: GUI.
5. Anas Mohamed Abdelrahman: GUI.

References

[1] G. Stix, "Encoding the "Neatness" of Ones and Zeroes," *Scientific American*, vol. 265, no. 3, pp. 54–58, Sep. 1991.

[2] "Huffman coding," 2014. [Online]. Available: http://homes.sice.indiana.edu/yye/lab/teaching/spring2014-C343/huffman.php