Efficient Image Colorization using Parallel Environment

Jadhav Pooja Vishnu[1], Chandak Pooja Shamlal[2], Date Dipali Kisan[3], Bagwan Altamash H[4]

[1] *Student, Department of Computer Engineering, AVCOE, Maharashtra, India*
[2] *Student, Department of Computer Engineering, AVCOE, Maharashtra, India*
[3] *Student, Department of Computer Engineering, AVCOE, Maharashtra, India*
[4] *Student, Department of Computer Engineering, AVCOE, Maharashtra, India*

ABSTRACT

Now a days we see many old time films getting a new popularity due to their coloured version being appreciated in the market. Technologies are getting evolved so as to improve those monochrome images into coloured ones. This stream is becoming a great field of development in the image related industry. Our perspective is to develop a software which would prove to be a handy tool for professional photographers and mainly for the editors in concern with film making. So that they can convert more images into better quality coloured images. The Distributed environment perspective added to the main idea is to let these images be processed at a faster pace.Using Distributed environment we can store larger amount of data and process it accordingly. The algorithm to be used will increase the speed of colouring areas with perfect RGB combination. So, the entire intention is to make faster and effective application for colourization.

Keyword - *Colorization , Quantization, Histogram, Luminance, Chrominance*

1. Introduction

The idea behind this project is to obtain a system which can change a monochrome image into a coloured one. There is a big tide in the editing field where the old images are processed and converted into new looking coloured images. This is done to manipulate ages old images.

1.1 Motivation of Project

The photo editing tools popular all over the market makes us think of what if we come up with a tool which can be more efficient and can colourize multiple images at the same instance of time.

1.2 Problem Statement

To develop a project so as to colourize monochrome images into coloured images with better quality by,first,locally weighted regression (LWR) on the histograms of grayscale image and the source image is performed to analyse their feature distributions. Then, we use a novel method to match these features by finding and adjusting the zero-points

2. Purpose and Scope Document

This project will be resulting in a software which will be able to colourize video data and images at a faster pace with increase in the quality of the output video. This will also make it possible to handle a larger video file which may contain more frames.

2.1 Normal Requirements

These requirements are extrinsically specified by the user to the software developer. This may include:

1. This system will accept any monochrome image.
2. User shall select it at the start of software.
3. It will give users freedom of choosing the color and coloring it on the image
4. Only the specified region on the image would get colored.
5. Proper output image is written in the System folder.

2.2 Expected Requirements

This are not specified by the user during the communication stage. Nevertheless, They are expected to be present in the software. Some of these are:

1. Response should be quick and not time consuming.
2. The software should be user friendly.
3. Software should be light and not occupy large space.
4. Help Files should be provided, explaining the exact functioning and FAQs regarding the software

2.3 Expected Requirements

This are neither specified by the user nor expected. This is additional features included by the developer in the software to increase ease of customer interaction, thereby increasing customer satisfaction. This include

1. Parallel multithreading to enhance the speed of system.
2. Integration of other technologies like CUDA, HADOOP to enhance the process.

3. Design

The major task of extraction of twitter data would be handled by JTwitter API in our program. JavaFX powered UI would provide clearer and elegant look and feel to the software.

Graphical user interface is the mediator between end user and main system. Main system consists of Login Page and the colour pallet. All the interaction between user and system is done due to this interface. User can get coloured image as an output from system

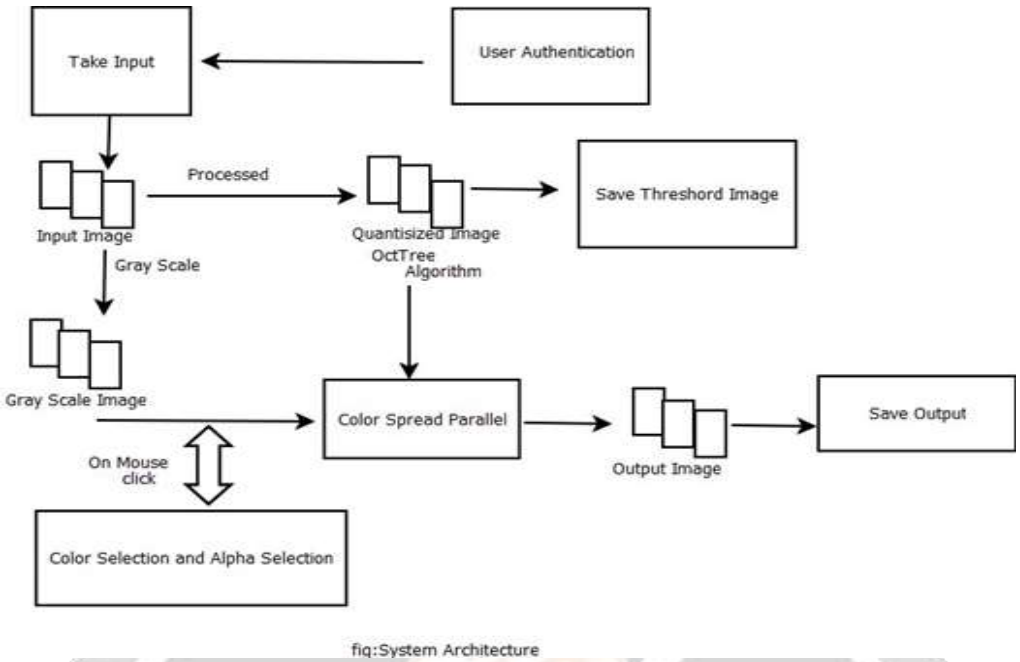


fig:System Architecture

Chart -1: Architecture Design

4. Image Colorization

4.1 Octree Algorithm

Most standard techniques treat color quantization as a problem of clustering points in three-dimensional space, where the points represent colors found in the original image and the three axes represent the three color channels. Almost any three-dimensional clustering algorithm can be applied to color quantization, and vice versa. After the clusters are located, typically the points in each cluster are averaged to obtain the representative color that all colors in that cluster are mapped to. The three color channels are usually red, green, and blue, but another popular choice is the Lab color space, in which Euclidean distance is more consistent with perceptual difference.

The most popular algorithm by far for color quantization, invented by Paul Heckbert in 1979, is the median cut algorithm. Many variations on this scheme are in use. Before this time, most color quantization was done using the population algorithm or population method, which essentially constructs a histogram of equal-sized ranges and assigns colors to the ranges containing the most points. A more modern popular method is clustering using octrees, first conceived by Gervautz and Purgathofer and improved by Xerox PARC researcher Dan Bloomberg.

4.2 Color Spread Algorithm

Flood fill, also called **seed fill,** is an algorithm that determines the area connected to a given node in a multi-dimensional array. It is used in the "bucket" fill tool of paint programs to fill connected, similarly-colored areas with

a different color, and in games such as Go and Minesweeper for determining which pieces are cleared. When applied on an image to fill a particular bounded area with color, it is also known as **boundary fill**.

4.3 Gray Scale Conversion Algorithm

When the input is taken firstly to reduce the noise, Image would be passed through a gray scale filter.The **lightness** method averages the most prominent and least prominent colors: (max(R, G, B) + min(R, G, B)) / 2. The **average** method simply averages the values: (R + G + B) / 3. The **luminosity** method is a more sophisticated version of the average method. It also averages the values, but it forms a weighted average to account for human perception. We're more sensitive to green than other colors, so green is weighted most heavily. The formula for luminosity is 0.21 R + 0.72 G + 0.07 B.

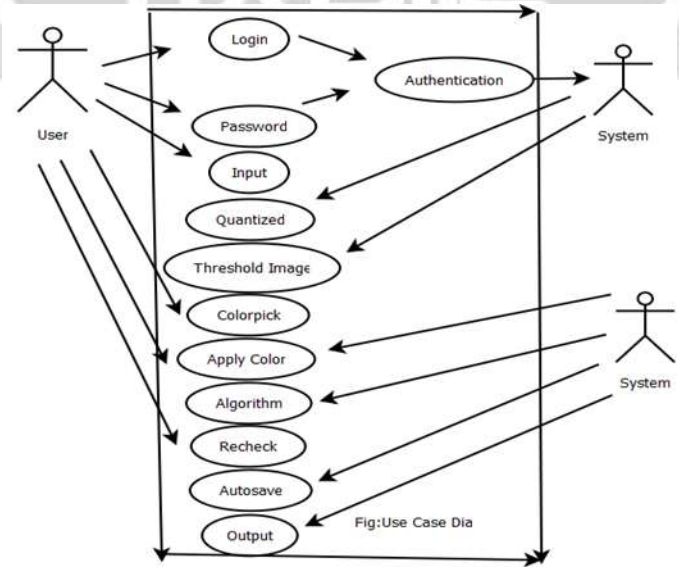

4.4 Multithreading

Multithreading is the ability of a program or an operating system process to manage its use by more than one user at a time and to even manage multiple requests by the same user without having to have multiple copies of the programming running in the computer.

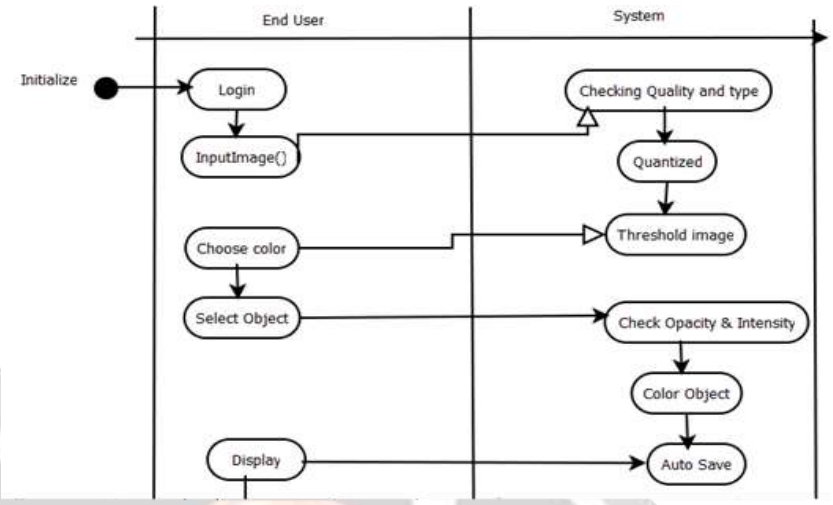
Where multiprocessing systems include multiple complete processing units, multithreading aims to increase utilization of a single core by using thread-level as well as instruction-level parallelism. As the two techniques are complementary, they are sometimes combined in systems with multiple multithreading CPUs and in CPUs with multiple multithreading cores.

5. UML Diagrams

5.1 UseCase Diagram



5.2 Activity Diagram



5.3 Sequence Diagram

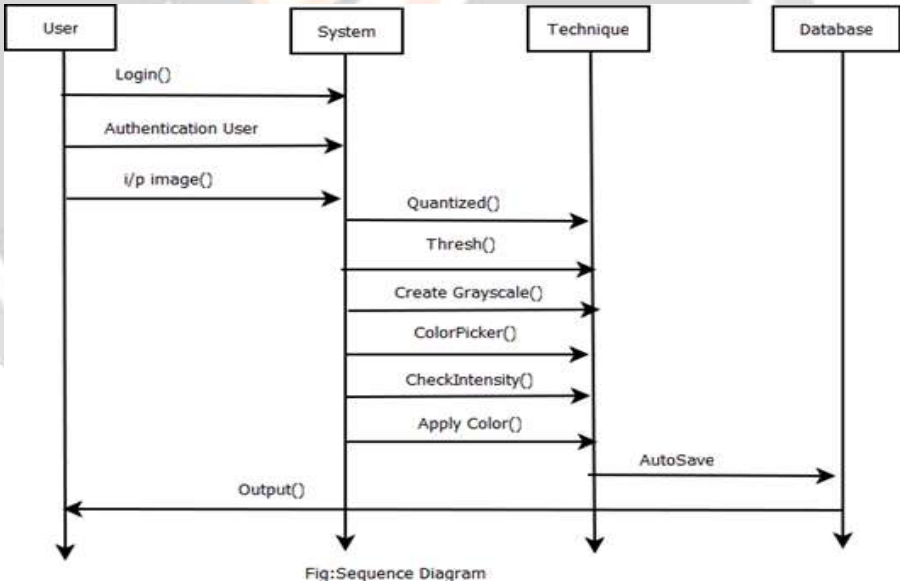


Fig:Sequence Diagram

5. Test Cases

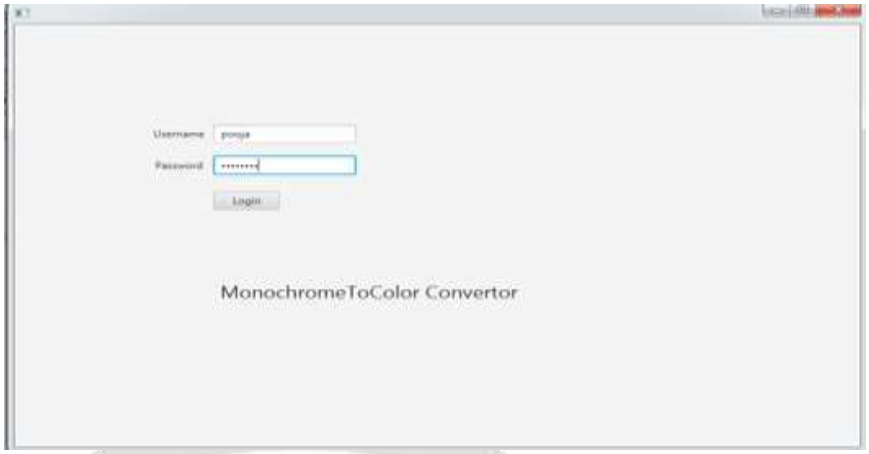


Chart -2: Login Page

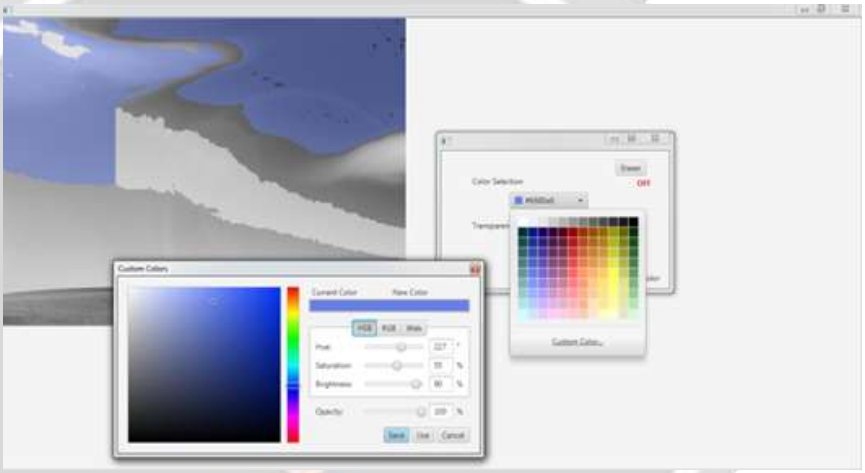


Chart -3: Colorization

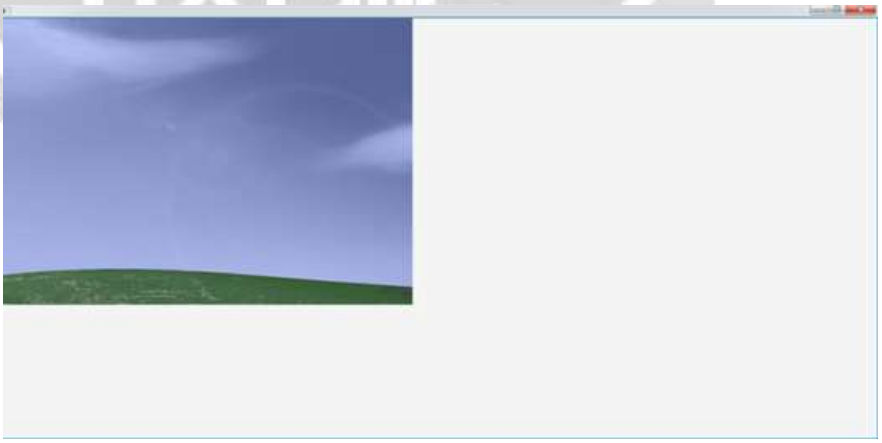


Chart -4: Output

7. CONCLUSION

As per the goal of this project an attempt is made to show how transformation of monochrome image is made to colour one. Also this colourization process is begin carried out on parallel environment using multi-threading.
Present system just performs the colourization task. Quality of the image is not taken into consideration, however our project not only colours the monochrome image but also retains the quality of the image. By developing such system, no compromise is being made with the quality of the image and image is also coloured using multi-threading

8. ACKNOWLEDGEMENT

`No plan can be implemented without proper guidance, motivation and support.' We would like to thank our internal guide Dr. B. L. Gunjal for giving us the help and guidance we needed. We are really grateful to them for their kind support. Their valuable suggestions were very helpful. I am also grateful to Prof. R. L. Paikrao, Head of Computer Engineering Department, AVCOE for his cooperation and support. In the end our special thanks to Prof. M. B. Vaidya, Project Co-ordinator, for providing various guidelines and keeping a watch on our punctuality. It helped us in completing our tasks in time. Also a Thanks to the departmental staff and technical assistants for making the resources available to us whenever we asked for them.

Mr. Altamash Bagwan
Ms. Pooja Chandak
Ms. Pooja Jadhav
Ms. Dipali Date

9. FUTURE SCOPE

In this project, concept of image colourization using parallel environment is introduced first time, where one can colour image efficiently. But this has large future scope, as far as video colourization is concerned. There is lots of scope for video colourization in parallel environment using CUDA. Moreover the GUI is also made more user friendly. As we have implemented our project on parallel environment, one can take help from this to design system for video colourization. Video colourization can be achieved on parallel environment using CUDA very efficiently by reducing time complexity to a considerable amount. Thus, though we have tried to implement this new concept lots of changes are required to be done in future.

10. REFERENCES

[1]. Bin Sheng,Hanqiu Sun,Marcus Magnor and Ping Li,"Video Colorization Using Parallel Optimization in Feature Space",IEEE Transaction On Circuits And Systems For Video Technology,Vol. 24,Issue 3,2014

[2]. Shiguang Liu, Xiang Zhang,"Automatic grayscale image colorization using histogram regression",Science Direct,Pattern Recognition Letters,Vol. 33,pp. 1673-1681,2012

[3]. Zhigeng Pan, Zhaohua Dong and Mingmin Zhang "A New Algorithm for Adding Color to Video or Animation Clips" [4].

[4]. Hyung-Il Koo and Nam-Ik Cho "Colorization based on soft segmentation", JEI Letters

[5].Daniel Sykoraa, Jan Burianekb, Jir Zaraa,"Colorization of black-and-white cartoons",Sciencedirect Image and Vision Computing,Vol. 23,pp. 767-782,2005