JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Color Clustering and Recommendation System

[1]Routhu Nikhil Joshi, [2]K.Tulasi Krishna Kumar,

[1]MCA 2nd year, [2]Associate Professor,
[1]Master of Computer Application,
[1]Sanketika Vidya Parishad Engineering College, Visakhapatnam, India

ABSTRACT
This project involves the development of a color clustering and recommendation system. The system imports an image, allows the user to sample a color from the image, and generates a set of color harmonies based on the selected color. The color harmonies include monochromatic, analogous, complementary, triadic, and tetradic color schemes. The system also provides a real-time camera preview and the ability to capture images from the camera. The system is built using the OpenCV, NumPy, PIL, and Tkinter libraries in Python. The main interface includes image display, color sampling, color preview, color scheme selection, and a button to generate color harmonies. The system provide users with an intuitive and user-friendly tool for color selection and design.
Keywords: color clustering, color harmonies, color sampling, generate color harmonies

1.1 INTRODUCTION

Color reproduction is an essential aspect of various fields, from art and design to photography and printing. However, reproducing colors accurately can be a challenging task, and it often requires extensive knowledge and skills. The aim of my project is to develop a color clustering and recommendation system that can help improve human color reproduction abilities. By allowing users to sample colors from images and generate harmonious color schemes, the system aims to provide an intuitive and user-friendly tool for color selection and design. The key point of the project is to help users enhance their color reproduction skills and create visually appealing designs.

2.1 SYSTEM ANALYSIS

The use of color has a long and complex history, with various civilizations developing their own color theories and palettes. However, reproducing colors accurately has always been a challenge. In the past, color reproduction was limited to natural dyes and pigments, which were often inconsistent and limited in range. With the advent of synthetic pigments and color printing technologies, color reproduction has become more accurate, but challenges remain.

One of the challenges is the human perception of color. Different people perceive color differently, and color perception can also be influenced by environmental factors such as lighting. This can lead to inconsistent color reproduction, especially in fields such as art, fashion, and product design where color plays an important role.

The project aims to help improve human color reproduction by developing a color clustering and recommendation system. The system allows users to import images, sample colors from the images, and generate color harmonies based on the selected color. It provides a range of color schemes, including monochromatic, analogous, complementary, triadic, and tetradic. The system also features a real-time camera preview and the ability to capture images from the camera. By providing an intuitive and user-friendly tool for color selection and design, the project aims to improve the accuracy and consistency of color reproduction, ultimately enhancing the human color reproduction ecstatic ability.

2.2 EXISTING SYSTEM

In the past, generating color schemes was a tedious and time-consuming process that involved manually selecting colors from a color chart. This method was inefficient and often resulted in poorly coordinated color schemes. With the advent of technology, it became possible to generate color schemes digitally. However, the process still required the user to manually select colors. The lack of a proper system to extract colors directly from an image was a major limitation.

2.3 PROPOSED SYSTEM

The proposed system aims to provide a user-friendly and intuitive color clustering and recommendation system. The system allows the user to import an image and select a color from it to generate a set of color harmonies. These color harmonies include monochromatic, analogous, complementary, triadic, and tetradic color schemes. The system also features a real-time camera preview and the ability to capture images from the camera. The system is built using the OpenCV, NumPy, PIL, and Tkinter libraries in Python. The main interface includes image display, color sampling, color preview, color scheme selection, and a button to generate color harmonies. This system

will help improve human color reproduction abilities and provide a valuable tool for designers to create aesthetically pleasing color combinations.

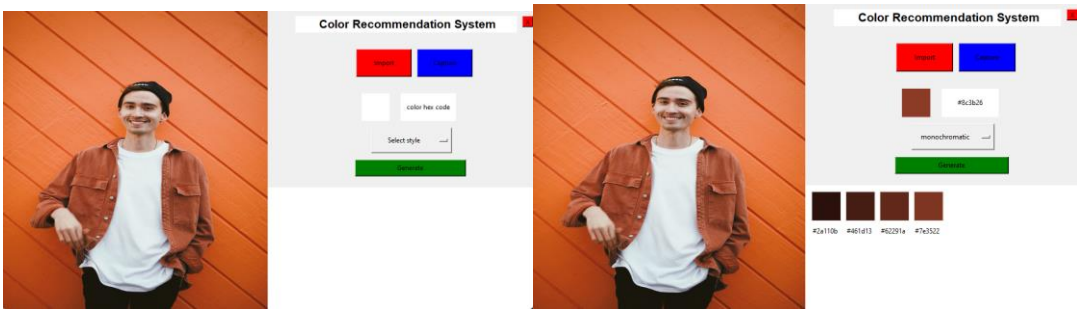


Figure2.3: Creating color combinations

2.4 FEASIBLITY STUDY

A feasibility study was conducted to evaluate the viability of the proposed color clustering and recommendation system. The study analyzed the technical, economic, operational, legal, and scheduling aspects of the project. From a technical perspective, the project was deemed feasible as the required software libraries and tools were readily available and compatible with the chosen programming language. The economic analysis showed that the project's development costs were within the budget, and the potential benefits of the system outweighed the costs. From an operational perspective, the system was found to be user-friendly and intuitive, making it suitable for ordinary people to use for color selection in their clothing and home paintings. Legal aspects were considered and addressed and ensuring the project complied with relevant laws and regulations. A detailed project schedule was created, taking into account potential delays and allowing for ample testing and debugging time. Overall, the feasibility study confirmed that the proposed system was viable and should proceed to the development phase.

2.4.1 Economically Feasibility

The economic feasibility of the proposed system is promising as the required hardware and software resources are widely available and affordable. The system can be easily installed on a computer or a mobile device with minimum specifications, reducing the need for additional hardware investments. Additionally, the software libraries used in the system are open source and freely available, lowering the development and maintenance costs. The system's potential to attract a wide range of users, including ordinary people interested in color selection for clothing and house painting, also supports its economic viability.

2.4.2 Technical feasibility

Technical feasibility this project, the technical feasibility has been determined to be high as it utilizes well-established libraries and frameworks such as OpenCV, NumPy, PIL, and Tkinter. These libraries provide the necessary tools for image processing, color analysis, and graphical user interface development, enabling the project to be developed in a technically feasible manner.

2. Furthermore, the system has been designed with scalability in mind, allowing for future enhancements and improvements to be made. This technical feasibility is important in ensuring that the project can be developed and deployed within a reasonable time frame and budget while meeting the necessary technical requirements. The use of widely adopted and well-documented libraries also ensures that the project can be maintained and updated as needed. Overall, the technical feasibility of this project provides a strong foundation for its successful development and implementation.

2.4.3 Operational Feasibility

In this proposed project, the operational feasibility will depend on how easy it is for the end-users to interact with the system and integrate it into their workflow. This will require a user-friendly interface that is intuitive and easy to navigate, as well as training and support to ensure that users can effectively utilize the system's features. By ensuring operational feasibility, the project will be able to deliver on its objectives and provide an effective tool for color selection and design.

3 SPECIFICATION

3.1 HARDWARE REQUIREMENTS (Minimum Requirement)

1.RAM:4GB+RAM
2.PROCESSOR: i3 5th Gen 2.2 Ghz

3.2 SOFTWARE REQUIREMENTS

1.Domain: Python
2.Version: Python IDLE (3.8.0)
3.Code Editors: Notepad++ or any basic text editors
4.Frameworks and Dependencies: tkinter, pillow, Open CV
5.Operating System: Windows 11

4 CODE EDITORS

4.1 Notepad++

Notepad++ is a source code editor and a replacement for Microsoft Windows' default Notepad, available for free under the GNU General Public License. It supports multiple languages and is built using C++ with Scintilla, ensuring a faster execution speed and a smaller program size through optimized routines while maintaining user-friendliness. The developer of Notepad++, Don Ho, originally used JEXT but decided to build a C++ text editor with Scintilla after being dissatisfied with JEXT's poor performance. He developed it in his spare time since his company rejected the idea. Notepad++ was released in 2003 and was initially available only for Windows. Although the author considered porting it to Mac OS X and Unix platforms using wx Widgets, this idea was ultimately rejected. In 2010, the US government obliged US-based open-source project hosts to deny access to users in certain countries, which the developer felt was a violation of the free and open-source software (FOSS) philosophy. As a response, Notepad++ moved out of US territorial jurisdiction by releasing a version on Tux Family in France. Notepad++ was criticized by Lifehacker for its user interface but was still voted as the most popular text editor by its readers in 2014. In 2015, it was voted the most used text editor worldwide by Stack Overflow respondents. Notepad++ left Source Forge completely in 2015 and moved its forums to Node BB and its bug tracker to GitHub.

5 MODULE DESCRIPTION

There are two modules in this project.

1. Importing Image Module: This module allows the user to import an image into the system. The image is displayed on the main interface, and the user can sample a color from the image. The module utilizes the OpenCV and PIL libraries in Python to import and display the image.
2. Capturing Image Module: This module provides a real-time camera preview and the ability to capture images from the camera. The user can take a picture and then sample a color from the image. The module utilizes the OpenCV library in Python to access the camera and capture images. This module provides flexibility to the user to capture any object, pattern or texture from real life and sample the color to generate a set of color harmonies.

5.1 ARCHITECTURE

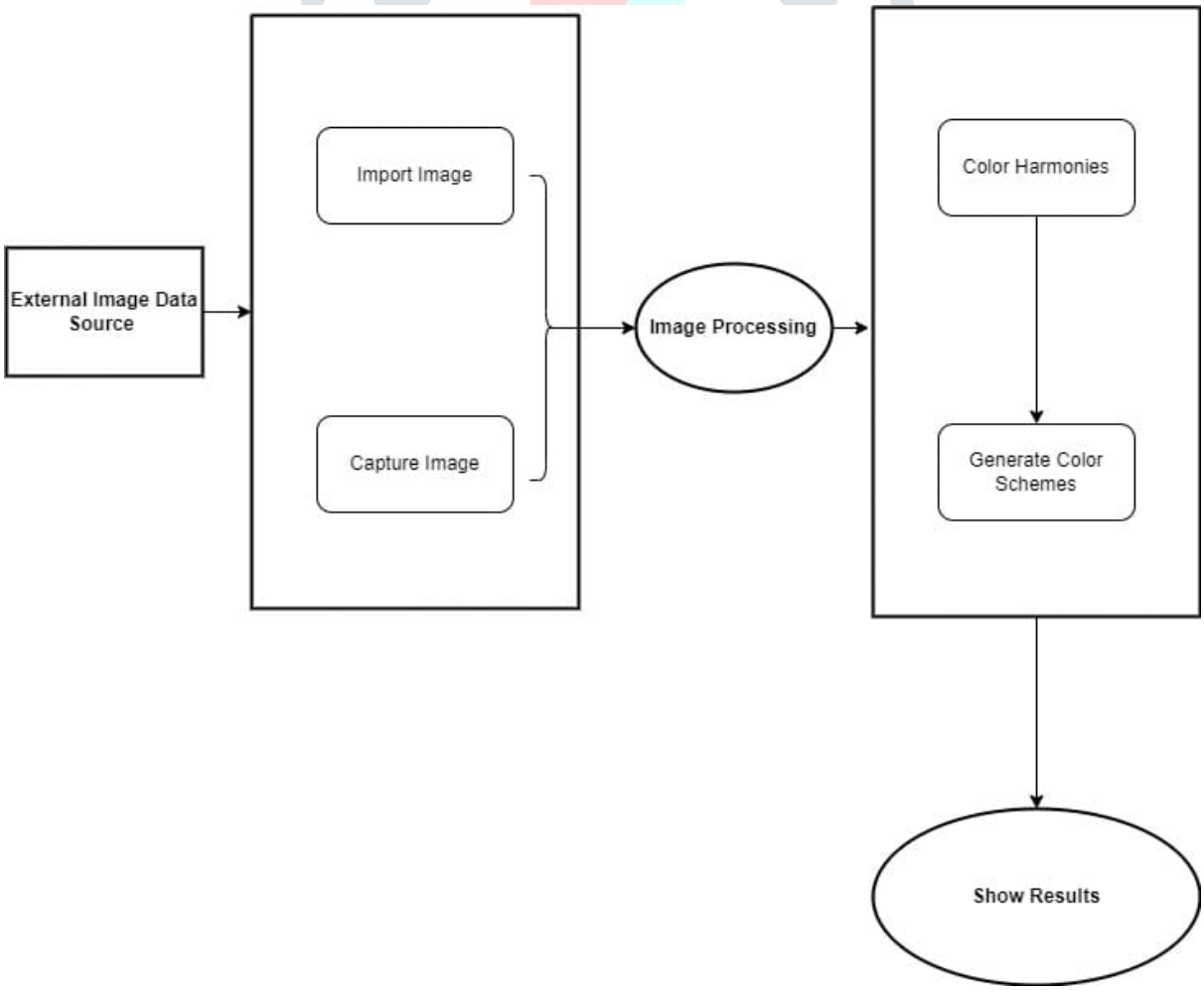


Figure 5.1 : ARCHITECTURE of color clustering and recommendation system

5.2 DATA FLOW DIAGRAMS

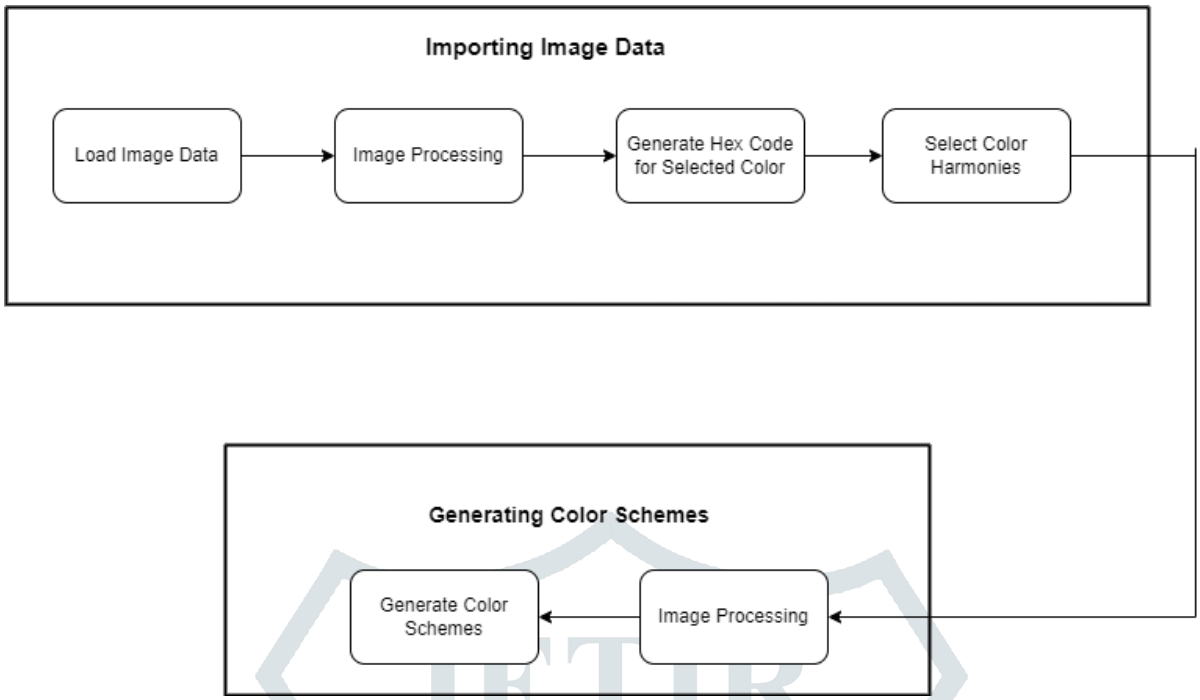


Figure5.2: Data flow diagram

5.3Things in UML

Things are the abstractions that are first-class citizens in a model; relationships tie these things together; diagrams group interesting collections of things.

There are four kinds of things in the UML:

•Structural hings

•Behavioral things[10]

•Grouping things

•An notational things

Structural things are the nouns of UML models. The structural things used in the project design are:

First, a class is a description of a set of objects that share the same attributes, operations, relationships and semantics.

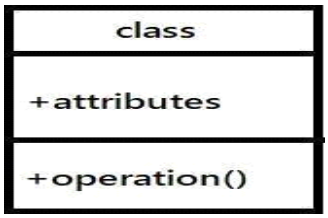


Figure5.3: Classes

Second, a use case is a description of set of sequence of actions that a system performs that yields an observable result of value to particular actor.

Figure5.3.1: Use Cases

Third, a node is a physical element [20] that exists at runtime and represents a computational resource, generally having at least some memory and often processing capabilit

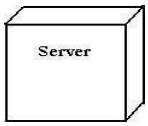


Figure5.3.2: Nodes

Behavioural things are the dynamic parts of UML models. The behavioural thing used is:

Interaction: An interaction is a behavior that comprises a set of messages exchanged among a set of objects within a particular context to accomplish a specific purpose. An interaction involves a number of other elements, including messages, action sequences (the behavior invoked by a message, and links (the connection between objects)

5.4Relationships in UML

There are four kinds of relationships in the UML:

•Dependency
•Association
•Generalization
•Realization

A dependency is a semantic relationship between two things in which a change to one thing may affect the semantics of the otherthing (the dependent thing)

Figure5.4: Dependencies

An association is a structural relationship that describes a set links, a link being a connection among objects. Aggregation is a special kind of association, representing astructural relationship[14] between a whole and its parts.

Figure5.4.1: Association

A generalization is a specialization/ generalization relationship in which objects of the specialized element (the child) are substitutable for objects of the generalized element (the parent).

Figure5.4.2: Generalization

A realization is a semantic relationship between classifiers, where in one classifier specifies a contract that another classifier guarantees to carry out.

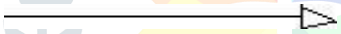


Figure5.4.3: Realization

6 SYSTEM IMPLEMENTATION

Systems implementation is the process of:

1.defining how the information system should be built (i.e., physical system design),
2.ensuring that the information system is operational and used,
3.ensuring that the information system meets quality standard (i.e., quality assurance) is the number of passes over the data. Loss is the error over the training set typically in terms of mean squared error (for regression) or log loss (for classification).

6.1 PURPOSE

During an epoch, the loss function is calculated across every data items and it is guaranteed to give the quantitative loss measure at the given epoch. But plotting curve across iterations only gives the loss on a subset of the entire dataset.

The purpose of System Implementation can be summarized as follows:

making the new system available to a prepared set of users (the deployment), and positioning on-going support and maintenance of the system within the Performing Organization (the transition). At a finer level of detail, deploying the system consists of executing all steps necessary to educate the Consumers on the use of the new system, placing the newly developed system into production, confirming that all data required at the start of operations is available and accurate,

6.2 SYSTEM MAINTENANCE

Software Maintenance is the process of modifying a software product after it has been delivered to the customer. The main purpose of software maintenance is to modify and update software application after delivery to correct faults and to improve performance.

Need for Maintenance –

Software Maintenance must be performed in order to:

•Correct faults.
•Improve the design.
•Implement enhancements.

•Interface with other systems.
•Accommodate programs so that different hardware, software, system features, and telecommunications facilities can be used.
•Migrate legacy software.
•Retire software.
Categories of Software Maintenance –Maintenance can be divided into the following:

1.CORRECTIVE MAINTENANCE

Corrective maintenance of a software product may be essential either to rectify some bugs observed while the system is in use, or to enhance the performance of the system.

2.ADAPTIVE MAINTENANCE

This includes modifications and updations when the customers need the product to run on new platforms, on new operating systems, or when they need the product to interface with new hardware and software.

This type of maintenance includes modifications and updations to prevent future problems of the software. It goals to attend problems, which are not significant at this moment but may cause serious issues in future.

Software Reverse Engineering[15] is the process of recovering the design and the requirements specification of a product from an analysis of it's code. Reverse Engineering is becoming important, since several existing software products, lack proper documentation[13], are highly unstructured, or their structure has degraded through a series of maintenance efforts.

7. TESTING

demonstrates that the software functions Software testing[12] is a critical element of software quality assurance and represents the ultimate review of specifications, design and coding .The user tests the developed system and changes are made according to their needs.

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development.

During testing the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to perform. When a system is developed, it is a hoped that it performs properly. In practice however some errors always occur. The main purpose of testing an information system is to find the errors and correct them. A successful test is one that found an error. The main objectives of the system testing are

•To ensure during operation the system will perform as per specification.
•To make sure that the system meets user requirements during operation.
•To verify that the controls incorporated in the system function as intended.
•To see that when correct inputs are fed to the system and the Outputs are correct.
•To make sure that during operation, incorrect input processing and output will be deleted.

If the testing is conducted successfully, it will uncover errors in the software. As a secondary benefit, testing appear to be working according to specification and that performance requirements appear to have been made.

7.1Validation Testing

Validation Testing is the process of ensuring if the tested and developed software satisfies the client /user needs. The business requirement scenarios have to be tested in detail. All the critical functionalities of an application must be tested here. As a tester, you need to evaluate if the test execution results comply with that mentioned in the requirements document. Any deviation should be reported immediately and that deviation is thus called a bug. The business requirement scenarios have to be tested in detail. All the critical functionalities of an application must be tested here.

8. IMPORT MODULE

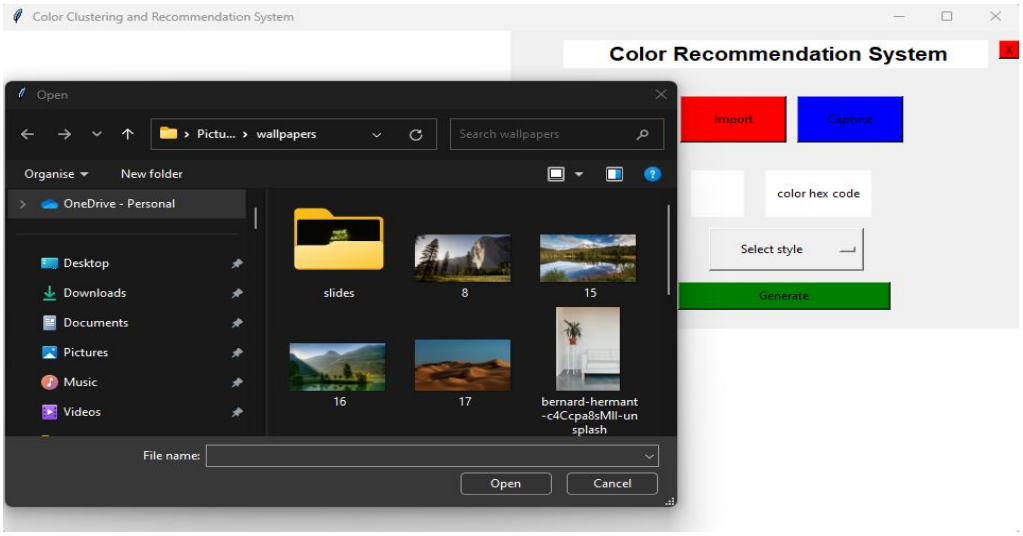


Figure8: Import module

8.1. CAPTURE MODULE

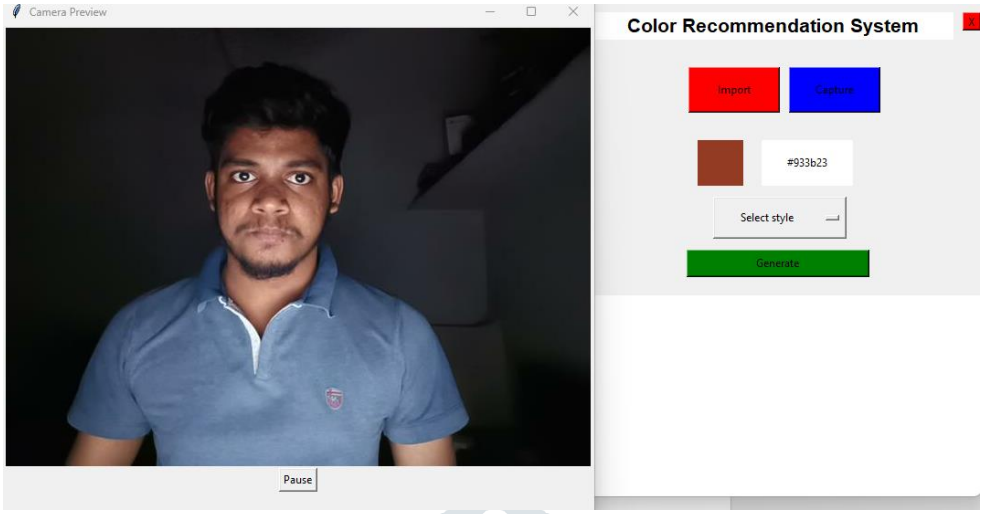
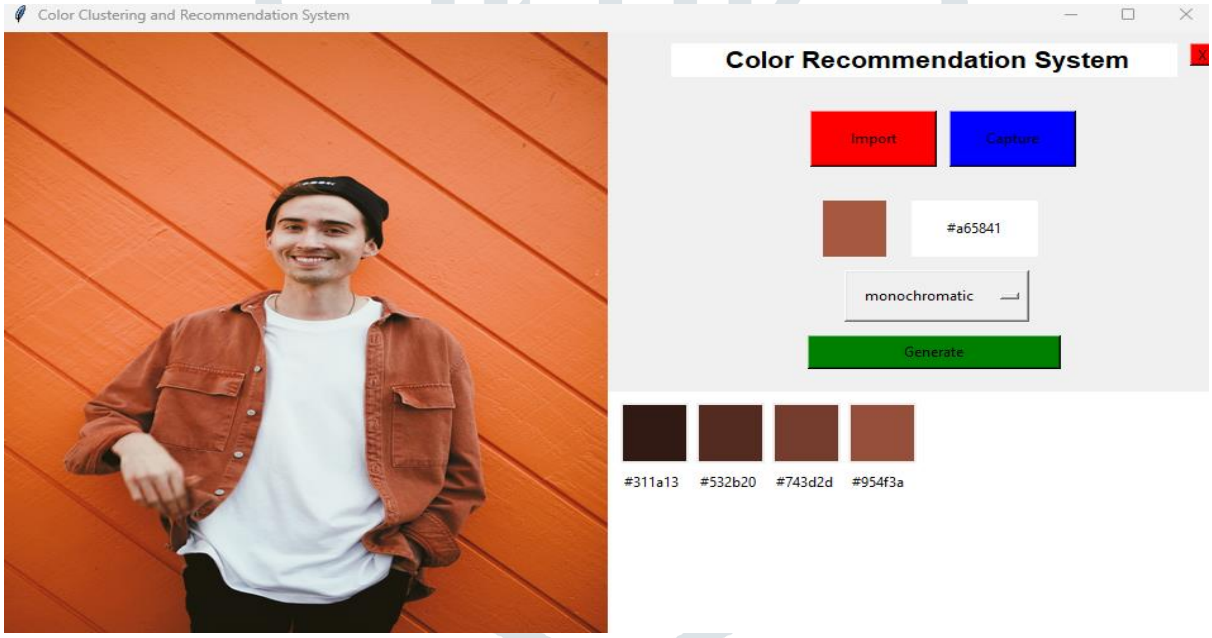


Figure8.1: Capture module

8.2. COLOR GENERATION MODULE



9. REAL TIME TEASTING

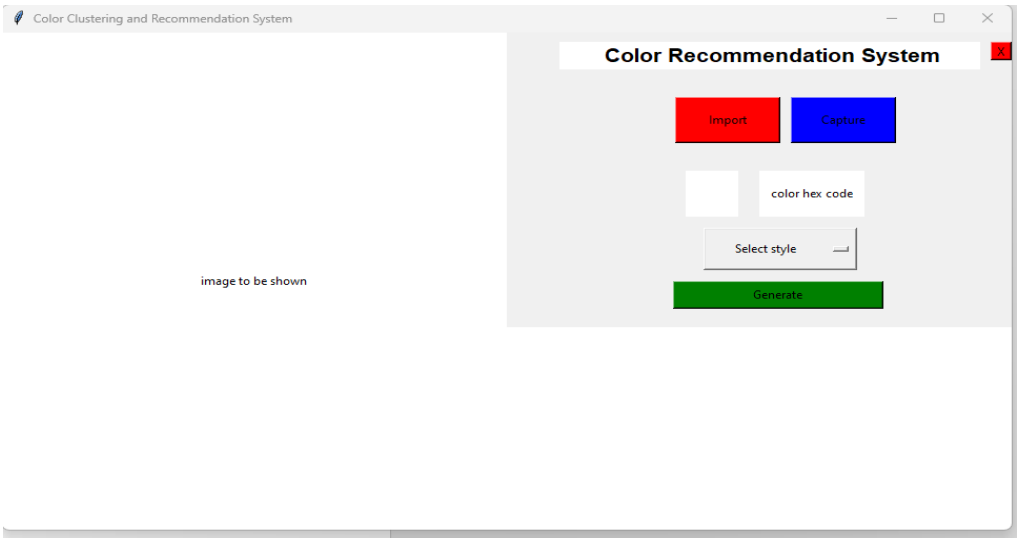


Figure9: Opening Output

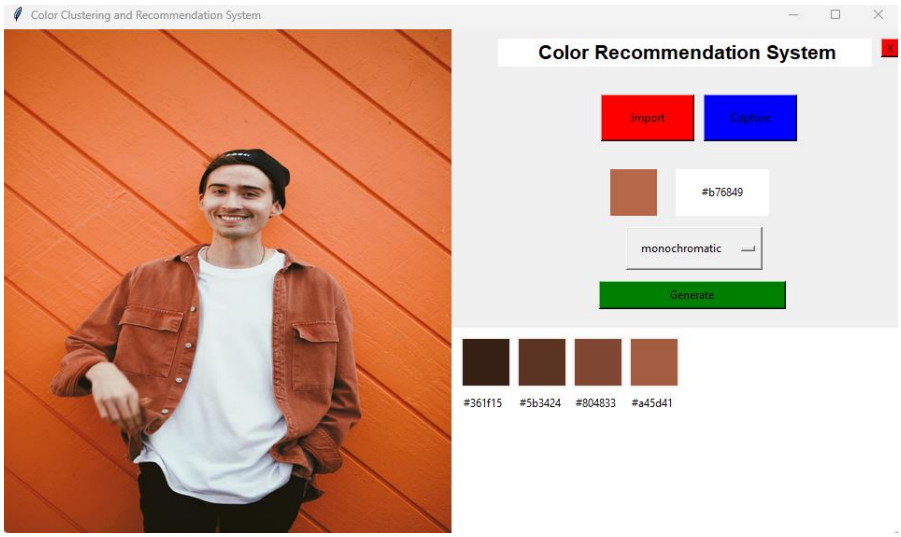


Figure9.1: End Output

10. CONCLUSION

this project aimed to develop a color clustering and recommendation system to assist in improving human color reproduction and design abilities. The system allows users to import images and sample colors, generating a range of color harmonies based on the selected color. This system provides a real-time camera preview and the ability to capture images from the camera.The proposed system is technically and economically feasible, providing an intuitive and user-friendly tool for color selection and design. With the ability to generate color harmonies, ordinary people can also make use of this application to select clothing and house paintings. The operational feasibility of the project is high, as the system is easy to use and provides valuable support for professionals and non-professionals alike.

In summary, this project contributes to the field of color design by providing a powerful and accessible tool for color selection and harmonization. It is expected to have a significant impact on the design industry and may be extended in future to cover more aspects of color design.

10.1. SCOPE FOR FUTURE DEVELOPMENT

1. Integration with e-commerce platforms: The color recommendation system could be integrated with e-commerce platforms to help customers select the right color for their products. For example, an online furniture store could use this system to suggest color schemes for their furniture.
2. Machine learning algorithms: The color recommendation system could be further improved by using machine learning algorithms to learn user preferences and suggest more personalized color schemes.
3. Mobile application: The system could be developed into a mobile application that allows users to take pictures of their surroundings and get color recommendations for their walls, furniture, and accessories.
4. Augmented Reality (AR): An AR component could be added to the system to allow users to see a real-time preview of how different color schemes would look in their room before making a final decision.
5. Expansion of color schemes: Currently, the system supports monochromatic, analogous, complementary, triadic, and tetradic color schemes. However, there are other color schemes like split-complementary and square that could be added to the system to provide users with more options.
6. Integration with image editing software: The color recommendation system could be integrated with popular image editing software like Adobe Photoshop or GIMP, allowing users to easily apply color schemes to their designs.
7. User feedback: The system could collect user feedback on the color schemes recommended and use it to improve its accuracy and usefulness.

11. REFERENCES

1. "Colorful Image Colorization" by Richard Zhang, Phillip Isola, and Alexei A. Efros: https://richzhang.github.io/colorization/
2. "An Evaluation of Color Palette Selection Methods for Data Visualizations" by Jeffrey Heer and Maureen Stone: https://idl.cs.washington.edu/files/2012-Evaluation-Color-Palette-Selection.pdf
3. "Color Analysis and Design Tools for Digital Media" by Yingqing Xu, Xiaohua Sun, and Robert W. Proctor: https://www.tandfonline.com/doi/abs/10.1080/10447318.2012.690770
4. "Color Harmony in Web Design" by Vitaly Friedman: https://www.smashingmagazine.com/2010/02/color-harmony-in-web-design/
5. "A Survey of Color for Computer Graphics" by Scott E. Anderson: http://www.cs.helsinki.fi/group/goa/mallinnus/sgi-stuff/sgi-docs/graphics/SGI_Developer/Color/survey.html

6. Gonzalez, R.C., Woods, R.E., & Eddins, S.L. (2004). Digital Image Processing Using MATLAB. Prentice Hall.
7. Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.
8. NumPy Community. (2021). NumPy User Guide. Retrieved from https://numpy.org/doc/stable/user/index.html
9. Clark, A. (2017). Python GUI Programming with Tkinter. Packt Publishing Ltd.
10. Poynton, C. (2012). Frequently Given Answers about Color. Retrieved from http://www.poynton.com/PDFs/ColorFAQ.pdf
11. Debevec, P., Malik, J. (1997). Recovering High Dynamic Range Radiance Maps from Photographs. Proceedings of SIGGRAPH.
12. "Color in Computer Vision: Fundamentals and Applications" by Theo Gevers, Arjan Gijsenij, Joost van de Weijer, and Jan-Mark Geusebroek: https://www.springer.com/gp/book/9789048126405
13. "The Interaction of Color" by Josef Albers: https://www.albersfoundation.org/teaching/books/the-interaction-of-color/
14. "The Elements of Color: A Treatise on the Color System of Johannes Itten Based on His Book the Art of Color" by Johannes Itten: https://www.wiley.com/en-us/The+Elements+of+Color%3A+A+Treatise+on+the+Color+System+of+Johannes+Itten+Based+on+His+Book+the+Art+of+Color%2C+1st+Edition-p-9780471289296
15. "Digital Color Management: Encoding Solutions" by Eddie Tapp and Rick Lucas: https://www.wiley.com/en-us/Digital+Color+Management%3A+Encoding+Solutions%2C+1st+Edition-p-9780471429478
16. "The Science of Color" by Steven K. Shevell: https://mitpress.mit.edu/books/science-color
17. "Color Correction Handbook: Professional Techniques for Video and Cinema" by Alexis Van Hurkman: https://www.peachpit.com/store/color-correction-handbook-professional-techniques-for-9780321713117
18. "Handbook of Color Psychology" edited by Andrew J. Elliot and Mark D. Fairchild: https://www.cambridge.org/core/books/handbook-of-color-psychology/6510187D4132C345DD3B9D9EDB68C177
19. "Color Vision and Colorimetry: Theory and Applications" by Noboru Ohta and Alan Robertson: https://www.wiley.com/en-us/Color+Vision+and+Colorimetry%3A+Theory+and+Applications-p-9780471395408
20. "Color Psychology and Color Therapy: A Factual Study of the Influence of Color on Human Life" by Faber Birren: https://www.amazon.com/Color-Psychology-Therapy-Influence-Human/dp/0878770098

12. BIBILIOGRAPHY

Routhu Nikhil Joshi is studying her 2nd year, Master of Computer Applications in Sanketika Vidya Parishad Engineering College, affiliated to Andhra University, accredited by NAAC. With a keen interest in python and machine learning, Nikhil Joshi chose to work on image processing using python with the aim of addressing the flaws in conventional color selection methods and providing a user-friendly tool for color selection and design. The project includes a completely developed system along with code, which has been submitted for evaluation to Andhra University as part of the completion requirements for his MCA degree.

Kandhati Tulasi Krishna Kumar: Project Guide, Placement Officer with 13+ Years of experience in training & placing the frosh into IT, ITES & Core profiles. Sir trained more than 9,500+ Diploma, UG, PG candidates to be corporate ready. He organised & upskilled more than 560+ faculty through FDPs. Sir authored 5 books to enhance the skillset of the candidates. Being a professional Designer in Pro-E, CNC certified by CITD, an editorial member of IJIT,IJAAC & life member in MISTE, and 16 other international professional societies. He published & guided 45+ articles in various international journals on SE, AI, DB, HRM & other.