**Synopsis Report**

**on**

**IMAGE TO PENCIL SKETCH**

**& IMAGE RECOGNITION**

**by**

**Sakshi Goyal - 2200290140132**

**Aastha Gupta - 2200290140001**

**Session: 2023-2024 (IV Semester)**

**Under the Supervision of**

**MR. PRASHANT AGARWAL (Associate Professor)**

**&**

**MS. NEELAM RAWAT (Associate Professor)**

**KIET Group of Institutions, Delhi –NCR, Ghaziabad**



**DEPARTMENT OF COMPUTER APPLICATIONS**

**KIET GROUP OF INSTITUTIONS, DELHI-NCR,**

**GHAZIABAD - 201206**

**(April - 2024)**

**ABSTRACT**

This project is about how new technologies can be used to develop a python application that enables the user to make a sketch of any Images. We will try to convert a normal photo into a sketch using computer vision, machine learning in a python programming Language.

This Project make it easier for people to understand how painters create pencil drawings, one of the most basic pictorial languages for representing the abstract perception of natural scenes.

This is accomplished through the use of graphical interface Python tools such as Tkinter, matplotlib, and others. The user can easily choose an image from their files. Then it can select the convert button. The picture has been transformed into a lovely sketch.

The project's primary goal is to better human-computer interaction by creating a method to assist people by converting their various images into a beautiful sketch. The user is provided with a decent user interface to upload a sketch input image and obtain related output images.

**TABLE OF CONTENTS**

1. Introduction
2. Literature Review
3. [Project Objective](#_TOC_250001)
4. Project Flow
5. Project Outcome
6. [Proposed Time Duration](#_TOC_250000)
7. References

**INTRODUCTION**

In today’s era, we are surrounded by different types of photo manipulating [filters](https://analyticsindiamag.com/adobes-new-ai-powered-tools-make-job-photo-video-editor-redundant/) in our mobile phones, apps…etc. But do you know how they do these images manipulations…..? In the backend, they are using [computer vision](https://analyticsindiamag.com/how-do-data-scientists-create-high-quality-training-datasets-for-computer-vision/) techniques. Computer vision has a wide variety of applications not only to reduce the human effort but also used for entertainment apps. Many photo editing apps like [Face App](https://analyticsindiamag.com/the-ai-behind-faceapp/), Instagram filters etc are using computer vision techniques.

In this Project, we will try to convert a normal photo into a pencil sketch using computer vision in a [python](https://analyticsindiamag.com/comparing-python-libraries-pylearn2-vs-scikit-learn/) programming language. In this Project, we will show how to convert an image into its corresponding pencil sketch in a few steps.

This system helps people better understand how painters produce pencil drawings, which is one of the most fundamental pictorial languages to abstract human perception of natural scenes.

TECHNOLOGY USED

**Python**

Python is a high-level, interpreted programming language. It is a robust, highly useful language focused on rapid application development (RAD). Python helps in the easy writing and execution of codes. Python can implement the same logic with as much as 1/5th code as compared to other OOPs languages. Python provides a huge list of benefits to all. The usage of Python is such that it cannot be limited to only one activity. Its growing popularity has allowed it to enter some of the most popular and complex processes like Artificial Intelligence (AI), Machine Learning (ML), natural language processing, Data science, etc. Python has a lot of libraries for every need of this project such as Pytube for downloading videos, selenium for web automation, etc. Python is reasonably efficient. Efficiency is usually not a problem for small examples. If your Python code is not efficient enough, a general procedure to improve it is to find out what is taking most of the time and implement just that part more efficiently in some lower-level languages.

**OpenCV**

OpenCV (Open Source Computer Vision Library) is an open source computer vision and

machine learning software library. OpenCV was built to provide a common infrastructure

for computer vision applications and to accelerate the use of machine perception in the

commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for

businesses to utilize and modify the code. The library has more than 2500 optimized

algorithms, which includes a comprehensive set of both classic and state-of-the-art

computer vision and machine learning algorithms. These algorithms can be used to detect

and recognize faces, identify objects, classify human actions in videos, track camera

movements, track moving objects, extract 3D models of objects, produce 3D point clouds

from stereo cameras, stitch images together to produce a high resolution image of an entire

scene, find similar images from an image database, remove red eyes from images taken

using flash, follow eye movements, recognize scenery and establish markers to overlay it

with augmented reality, etc.

**LITERATURE REVIEW**

A literature review for an image to pencil sketch project would involve examining existing research, methods, and techniques related to image processing, computer vision, and artistic style transfer. Here's a structured approach you can use to conduct your literature review:

Image Processing Techniques

Explore various image processing techniques commonly used in converting images to sketches This may include edge detection algorithms, such as Sobel, Canny, or Laplacian, which are fundamental in capturing the outline and contours of objects in an image.

Artistic Style Transfer

Investigate literature on artistic style transfer techniques, which aim to mimic the style of an input image or artwork onto another image. This could involve deep learning approaches like neural style transfer or traditional methods like texture synthesis.

**Deep Learning Approaches**

Review recent advancements in deep learning models for image processing and style transfer. This may involve convolutional neural networks (CNNs), generative adversarial networks (GANs), or autoencoders that have been used for similar tasks.

Existing Tools and Software

Survey existing software tools and applications that offer image to pencil sketch conversion functionality. Analyze their strengths, weaknesses, and user feedback to identify areas for improvement or innovation.

Challenges and Limitations

Identify common challenges and limitations associated with image to pencil sketch conversion, such as preserving details, handling complex scenes, or maintaining naturalness in the final output.

**PROJECT OBJECTIVE**

To develop a computer vision-based system capable of converting digital images into pencil sketches while preserving essential details and artistic quality.

Key Goals are :

Accurate Conversion

Develop algorithms and techniques to accurately convert input images into pencil sketches, capturing essential details and contours while maintaining the essence of the original image.

User Interface

Create an intuitive user interface that allows users to easily upload images, adjust conversion parameters (such as line thickness or shading intensity), and view the resulting sketches in real-time .

Customization Options

Provide users with customization options to adjust the style and characteristics of the generated sketches, enabling them to tailor the output according to their preferences.

Performance Optimization

Optimize the computational efficiency of the conversion process to enable real-time or near-real-time performance, allowing for quick and seamless sketch generation even for high-resolution images.

Quality Assessment

Implement metrics and evaluation techniques to assess the quality of the generated sketches objectively, ensuring that the converted sketches are visually pleasing and faithful to the original images.

Documentation and Support

Provide comprehensive documentation and user support resources to assist users in understanding the functionality of the system, troubleshooting issues, and maximizing their experience with the tool.

**PROJECT FLOW**

Designing a project flow for an image to pencil sketch project involves breaking down the process into manageable steps, from inputting an image to generating a pencil sketch output. Here's a proposed project flow:

Input Image Acquisition

Allow the user to input an image either by uploading a file or providing a URL.

Preprocessing

* Resize the input image to a standard size suitable for processing
* Optionally, perform noise reduction or other preprocessing techniques to enhance image quality.

Edge Detection

* Apply an edge detection algorithm to extract the outlines and contours of objects in the image.
* Common edge detection algorithms include Canny, Sobel, or Laplacian filters.

Optional Enhancement

* Offer options for adjusting parameters to enhance the quality of the edge-detected image.
* This step may include adjusting contrast, brightness, or sharpness to improve the final sketch output.

Style Transfer

* Optionally, implement a style transfer algorithm to apply a desired artistic style to the image.

Pencil Sketch Generation

* Convert the edge-detected image (with or without style transfer) into a pencil sketch.
* Apply techniques such as hatching, cross-hatching, or shading to simulate the appearance of a hand-drawn sketch
* This step may involve mapping the edge information to strokes of varying thickness and intensity to create the sketch effect.

Output Visualization

* Display the generated pencil sketch to the user for review and feedback.
* Provide options for saving the sketch to a file or sharing it on social media platforms.

User Interaction

* Allow the user to interact with the generated sketch, such as zooming in/out or applying additional effects .
* Incorporate user feedback mechanisms to refine the sketching process based on user preferences.

Optimization and Performance

* Optimize the algorithms and processing pipeline for efficiency and real-time performance.
* Consider implementing parallel processing or GPU acceleration to speed up computation, especially for high-resolution images.

Documentation and Deployment

* Document the project including the algorithms used, parameters, and implementation details.
* Package the project into a user-friendly application or web service for deployment.

**PROJECT OUTCOME**

The project outcome for an image to pencil sketch project would typically be an application or tool that takes input images and generates corresponding pencil sketch outputs. Here's a detailed description of the expected outcome:

A user-friendly interface allowing users to upload images from their device or provide a URL

Clear instructions and options for customizing the sketch generation process (if applicable), such as adjusting parameters or selecting artistic styles .A robust image processing pipeline that includes preprocessing, edge detection, optional style transfer, and pencil sketch generation.

High-quality pencil sketch outputs that accurately capture the contours, textures, and shading of the input images. The generated sketches should closely resemble hand-drawn pencil sketches, with natural-looking strokes and shading effects. Options for users to customize the appearance of the generated sketches, such as adjusting stroke thickness, intensity, or adding additional artistic effects.

Efficient processing and real-time performance, allowing users to generate sketches quickly, even for high-resolution images . The application should be scalable to handle a large number of concurrent users and accommodate future enhancements or feature additions.

Positive user feedback and satisfaction with the generated sketches, indicating that the application meets their expectations and delivers high-quality results. Overall, the project outcome should be a user-friendly and effective tool for generating pencil sketches from input images, providing users with the ability to create stunning artwork with ease.

**PROPOSED TIME DURATION**

**REFERENCES**

**Book References**

1. [Lu et al. 2012] C. Lu, L. Xu and J. Jia, Combining Sketch and Tone for Pencil Drawing Production. International Symposium on Non-Photorealistic Animation and Rendering (NPAR), 2012. [McCloud 1993] Scott McCloud, Understanding Comics. Harper Collins Publishers, New York, 1993.

**Internal References**

1. For Python, <https://www.w3schools.com/python/>
2. For Tkinter , <https://www.geeksforgeeks.org/python-gui-tkinter/>
3. For Matplotlib, https://matplotlib.org/cheatsheets/