

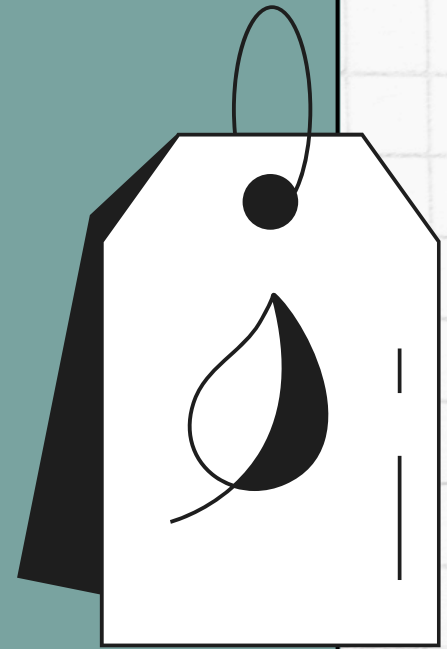
GENERATING REALISTIC IMAGES

By: Reality crafters

A S Sushmitha URS, Vaibhavi B Raj, Pooja S, Prasanna Kumar K

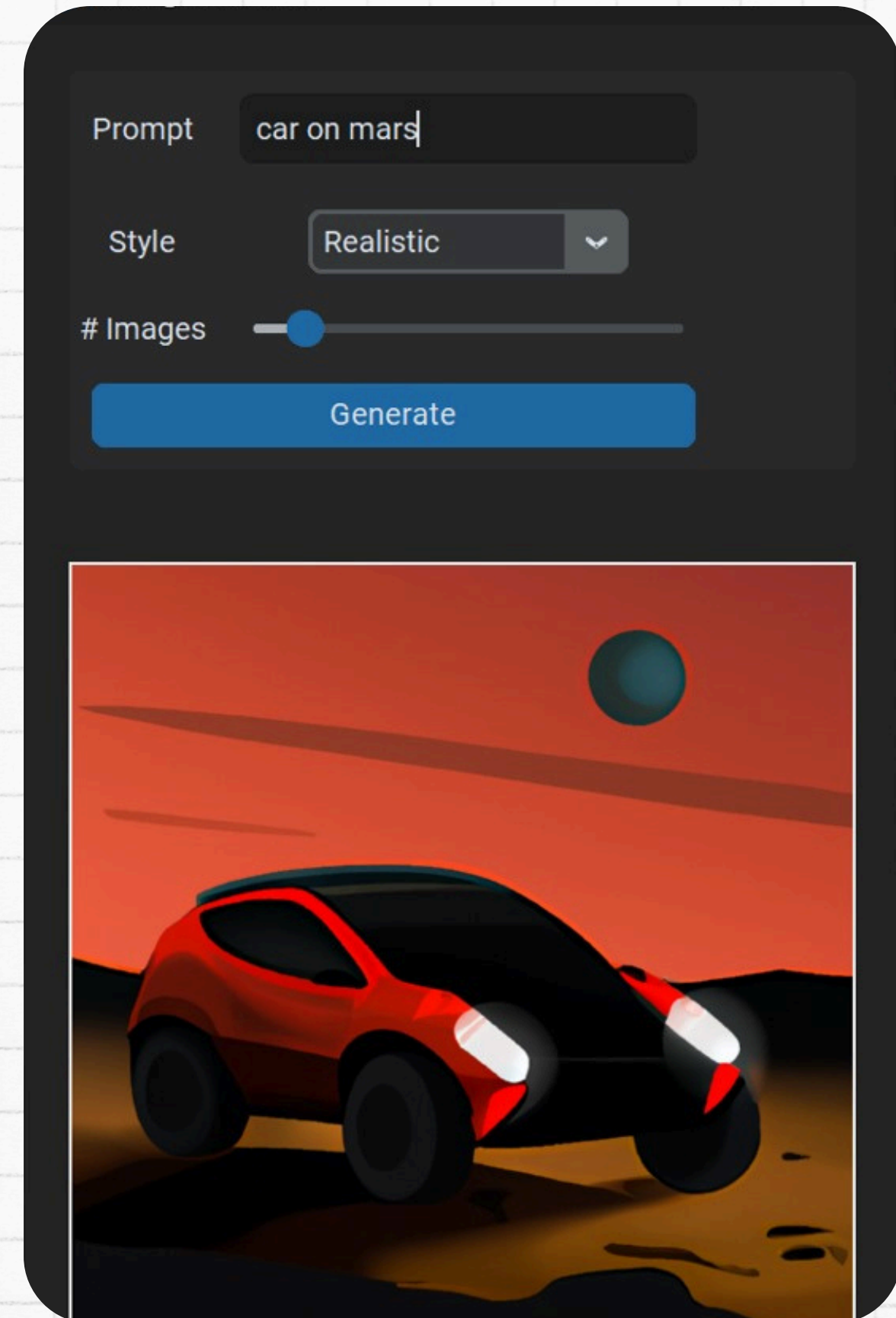
INTRODUCTION

The AI Image Generator is a sophisticated Python application that combines the power of OpenAI's GPT-3 with the ingenuity of Variational Autoencoder (VAE) technology to create captivating images tailored to user prompts and style preferences. With a user-friendly graphical interface developed using Tkinter, this application offers an intuitive and seamless experience. Whether you're an artist in search of creative inspiration or a designer looking to explore unique visual concepts, the AI Image Generator introduces a groundbreaking solution. Unlock the potential of artificial intelligence and VAE to transform your ideas into visually stunning masterpieces, all within this innovative tool.



PROBLEM STATEMENTS

The objective is to create a user-friendly graphical application that utilizes OpenAI's GPT-3 API to generate images, allowing users to specify styles and input prompts for image creation, simplifying the image generation process through a visually intuitive interface.



METHODOLOGY:

IMPORTING LIBRARIES:

The code begins by importing necessary libraries and modules such as `os`, `openai`, `customtkinter`, `tkinter`, `requests`, `io`, `Image`, `ImageTk`. These libraries are used for various purposes, including interfacing with OpenAI, creating a GUI, and handling image generation and display.

OPENAI CONFIGURATION:

The OpenAI API key is set in both Python scripts to authenticate with OpenAI's services. Make sure to replace the API key with your actual OpenAI API key for this code to work.

IMAGE GENERATION:

The user's prompt and style preference are passed to OpenAI's GPT-3 model using the `openai.Image.create()` function. This function is called with the user's inputs and parameters, such as the desired image size and number of images to generate.

METHODOLOGY:

USER PROMPTS AND STYLES:

The main functionality of this code involves taking user prompts and style preferences as inputs. The user prompt is extracted from the GUI text input box, and the chosen style is retrieved from the dropdown menu.

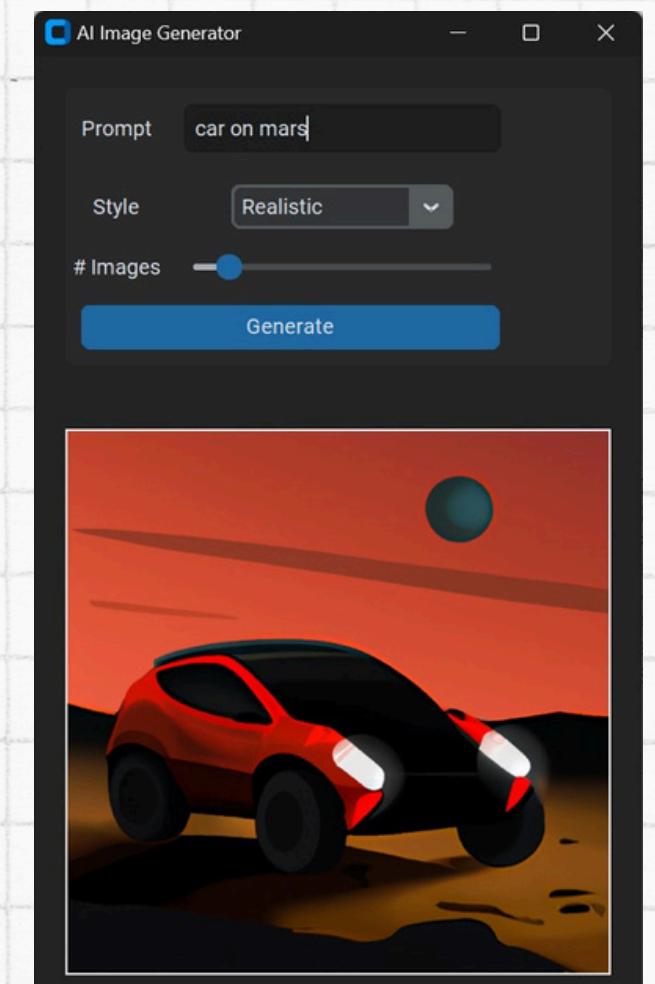
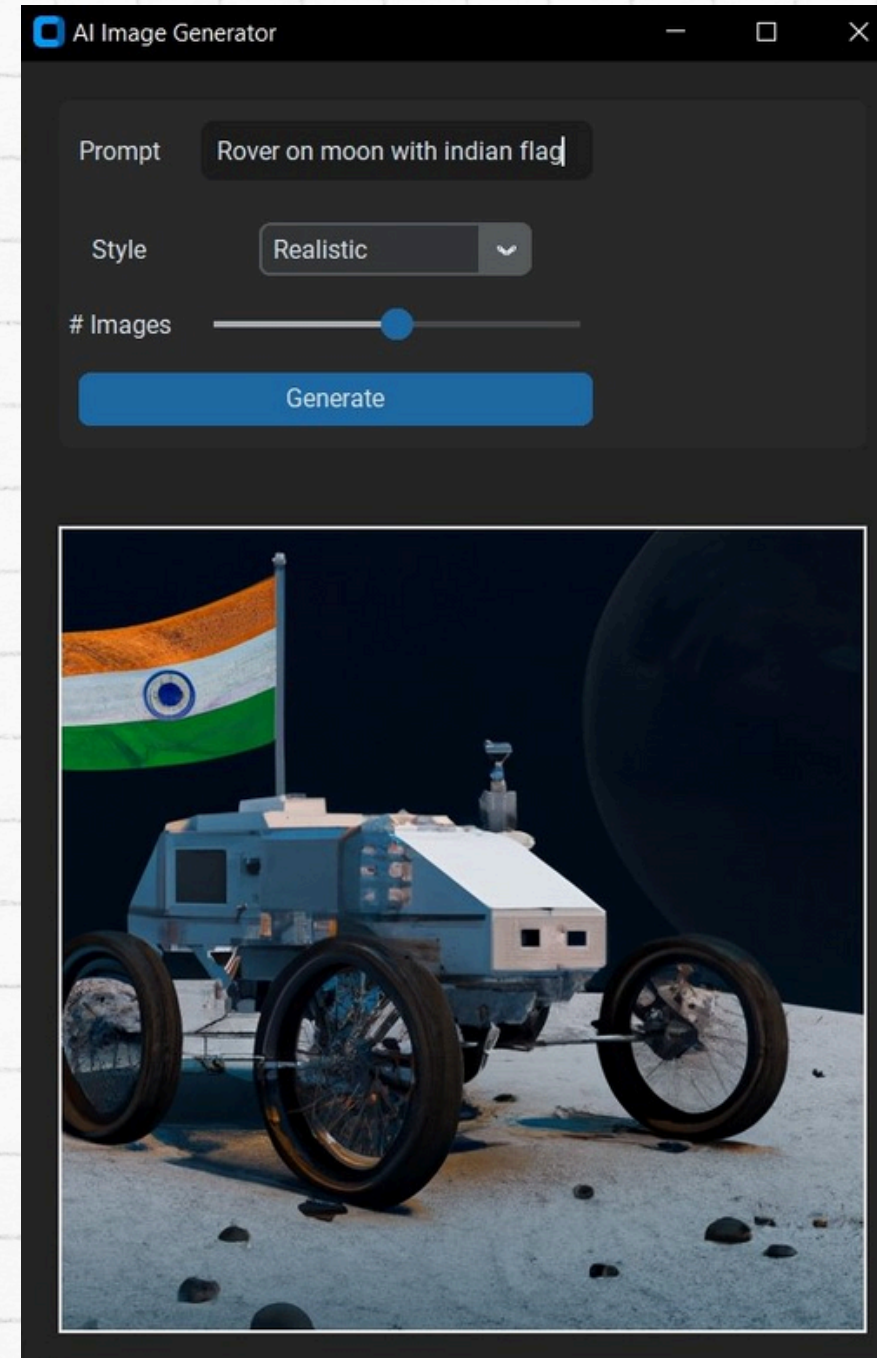
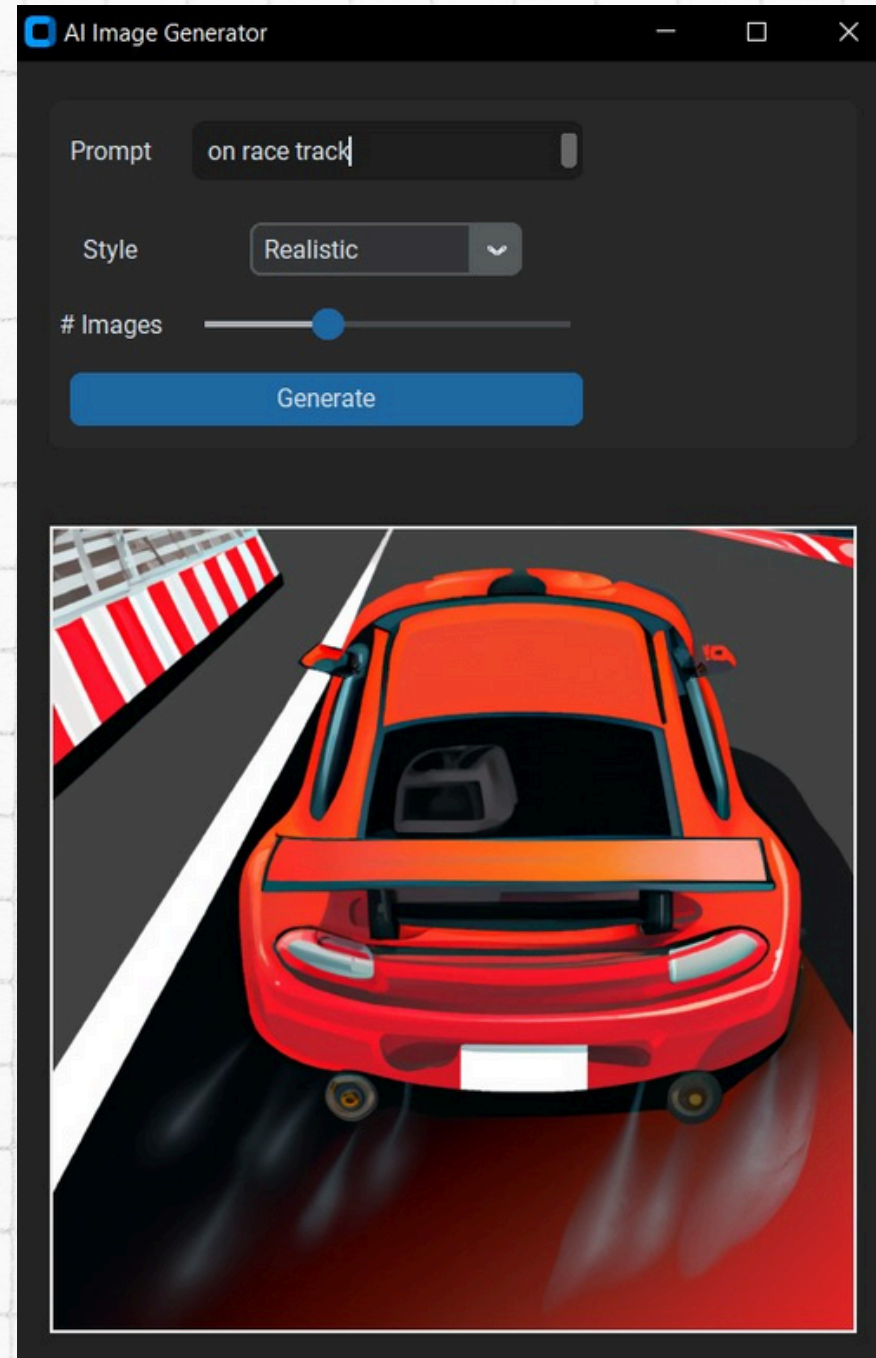
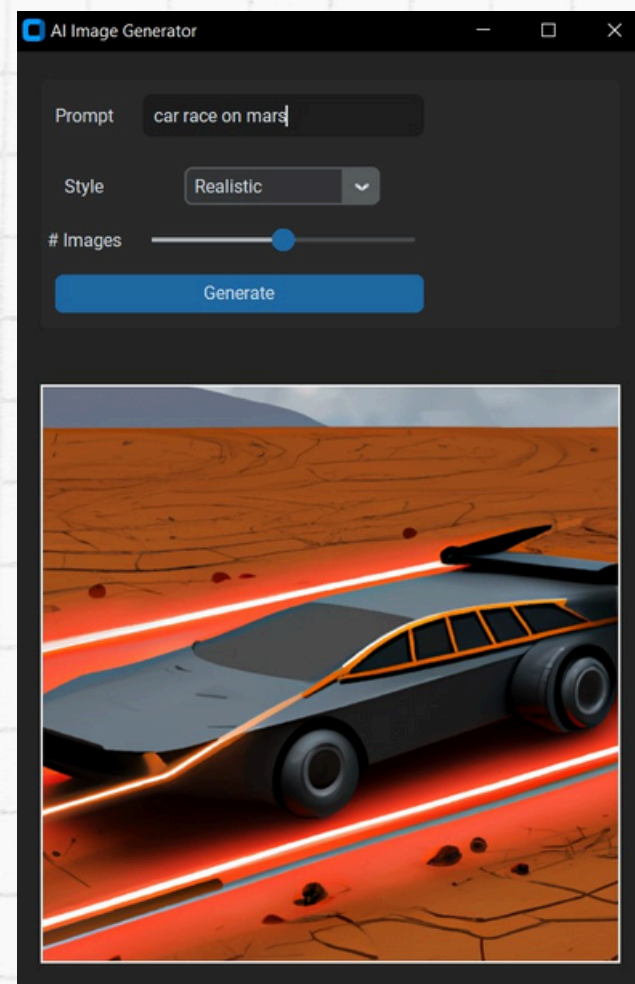
IMAGE DISPLAY:

The generated image URLs are fetched from the OpenAI response, and the images are downloaded using the requests library. The downloaded images are then converted to Tkinter PhotoImage objects to be displayed in the GUI.

UPDATING IMAGES:

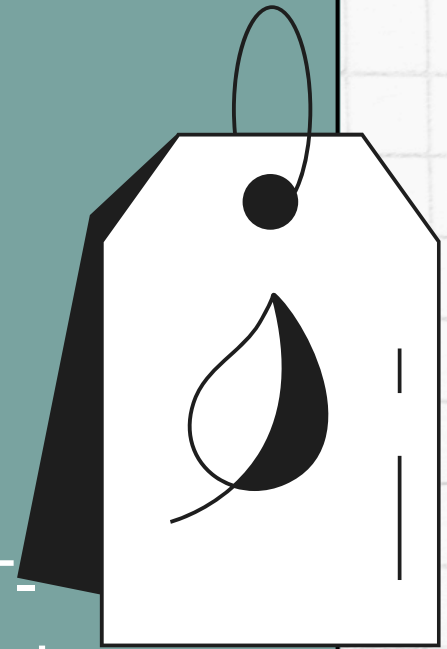
The GUI's image label is updated periodically to display different images in the generated set. It uses a timer-based approach to cycle through the images, updating every few seconds.

RESULTS



CONCLUSION

The creation of a user-friendly GUI application powered by OpenAI's GPT-3 API to produce images with customizable styles based on user-provided prompts. This application simplifies image generation, offering a seamless interface for users to express their preferences and creativity. It enables efficient and accessible image creation, catering to a wide range of artistic and design needs.



THANK
YOU!

