**Image Style Transfer: A Comparative Analysis of Classifier Performance**

Project Report in partial fulfillment of the degree

# Bachelor of Technology

in

# Computer Science &Engineering

## By

**2203A51213 M.Tejasri**

**2203A51142 M.Manitha**

**2203A51014 K.Akshitha**

**Under the Guidance of**

# Dr. Mamta Pandey

# Associate Dean (Data Science)& Associate Professor Department of CS&AI.

**Submitted to**



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

## SR UNIVERSITY, ANANTHASAGAR, WARANGAL



**CERTIFICATE**

This is to certify that the Project Report entitled“Image Style Transfer Using Machine Learning: A Comparative Analysis of Classifier Performance"”is are cord of Bonafide work carried out by Tejasri Muthyala, Manitha Mallela and Akshitha Kalva bearing Roll No(s) **2203A51213, 2203A51142, 2203A51014** during the academic year 2023-2024 in partial fulfillment of the award of the degree of ***Bachelor of Technology*** in **Computer Science Engineering** by the SR UNIVERSITY, WARANGAL.

# Supervisor Head of the Department

Dr. Mamta Pandey Dr. M. Sheshikala

Associate Dean (Data Science) & Assoc. Prof

SR University SR University

**ACKNOWLEDGEMENT**

We express our thanks to course coordinator Dr. Mamta Pandey, Associate. prof. for guiding us from the beginning through the end of the course project. We express our gratitude to head of the department CS&AI, Dr. M. Sheshikala, Professor for encouragement, support and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Dean, School of Computer Science and Artificial Intelligence, Dr C. V. Guru Rao, for his continuous support and guidance to complete this project in the institute.

Finally, we express our thank to all teaching and non-teaching staff of the department for their suggestions and timely support.

# ABSTRACT

Image style transfer, a technique that reimagines images in various artistic or aesthetic styles, has gained significant attention due to its creative and technical implications. This project explores the potential of deep learning models to perform style transfer, focusing on transforming a target image's visual style to match a reference style while preserving the content structure. Utilizing convolutional neural networks (CNNs) and generative adversarial networks (GANs), this project investigates the effectiveness of different architectures and loss functions in achieving seamless style transfer.

The project's core objectives include:

Content Preservation: Ensuring that the underlying structure and identifiable elements of the original image remain intact despite the applied style transformation.

Style Fidelity: Achieving a high degree of fidelity to the reference style, capturing its distinct visual characteristics without distorting the content.

Computational Efficiency: Exploring techniques to optimize the process for faster and more resource-efficient style transfer.

To evaluate the effectiveness of the approach, a series of experiments are conducted with various styles, ranging from classical art to contemporary designs. The outcomes are assessed through quantitative metrics such as content loss, style loss, and total variation loss, as well as qualitative evaluations by human reviewers. The project also considers practical applications, like enhancing photographs, creating artistic designs, and video frame style transfer.

The fundamental challenge in image style transfer is achieving a balance between content and style. Content preservation requires that key elements like objects, shapes, and spatial relationships in the original image remain recognizable after transformation. At the same time, style fidelity ensures that the visual features characteristic of the style image, such as brush strokes, color palettes, and artistic techniques, are accurately reproduced in the transformed image.

# Table of Contents

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Content** | **Page No** |
| 1 | Introduction | 1 |
| 2 | Literature Review | 2 |
| 3 | Design | 3 |
| 4 | Methodology | 4 |
| 5 | Data Pre-processing | 5 |
| 6 | Results | 9 |
| 7 | Conclusion | 10 |
| 8 | Future scope | 11 |
| 9 | References | 12 |

1. **INTRODUCTION:**

In recent years, image style transfer has emerged as a fascinating area of research within the field of computer vision and machine learning. The central idea is to take the artistic style from one image and apply it to another while preserving the original image's content and structure. This unique blend of art and technology has led to innovative applications across various domains, from creating digital art to transforming everyday photographs into masterpieces.

The origins of image style transfer are closely tied to advancements in deep learning, specifically convolutional neural networks (CNNs). These networks excel at extracting features from images, enabling them to understand complex patterns, textures, and styles. By leveraging these capabilities, researchers have developed algorithms that can separate content from style, allowing for a wide range of creative transformations.

This project explores the potential of image style transfer, focusing on achieving a harmonious balance between content and style. The process involves two key steps: extracting content and style features from the source images and then synthesizing a new image that retains the content structure while adopting the desired style. Achieving this balance is a complex task that requires careful tuning of parameters, optimization techniques, and loss functions.

One of the most well-known methods for image style transfer was introduced by Gatys et al., which uses a pre-trained CNN to extract content and style features from images. This method sparked interest in both academic and artistic communities, leading to the development of numerous variations and improvements. Other approaches, such as Generative Adversarial Networks (GANs), have expanded the scope of style transfer, enabling more flexibility and reducing the need for paired examples.

The applications of image style transfer are diverse, with potential in art, design, photography, film, and virtual reality. Artists and designers can create unique works by blending different styles, while photographers can add artistic effects to their images. In the entertainment industry, style transfer can be used to stylize video frames or create unique visual experiences in virtual environments.

# LITERATURE REVIEW

# 1. Gatys et al. (2015) proposed a seminal neural style transfer technique that uses convolutional neural networks (CNNs) to separate and recombine content and style of images. Their method demonstrated the ability to generate stylized images by optimizing a loss function that captures both content and style representations.

# 2. Johnson et al. (2016) introduced a fast and scalable approach to neural style transfer using feedforward neural networks. By training a separate network to directly generate stylized images, their method achieved real-time performance without iterative optimization, making it suitable for practical applications.

# 3. Ulyanov et al. (2016) proposed instance normalization, a technique that improves the quality and stability of neural style transfer by reducing the effects of batch normalization during training. Instance normalization enables more consistent stylization results across different input images and styles.

# 4. Huang et al. (2017) presented AdaIN (Adaptive Instance Normalization), an adaptive normalization technique that dynamically adjusts the mean and standard deviation of feature maps to match the style of a given reference image. AdaIN enhances the flexibility and control of neural style transfer by allowing users to specify arbitrary style images.

# 5. Li et al. (2018) explored the application of neural style transfer in fashion design, proposing a method for personalized clothing design based on user-provided sketches and style images. Their approach enabled users to generate stylized fashion designs tailored to their preferences and tastes.

# 6. Karras et al. (2019) introduced StyleGAN, a generative adversarial network (GAN) architecture capable of generating high-resolution and photorealistic images with diverse styles. While not specifically designed for style transfer, StyleGAN's ability to control the style and appearance of generated images has implications for image stylization and manipulation.

# 7. Liao et al. (2017) developed an interactive style transfer system that allows users to adjust stylization parameters in real-time and preview the results immediately. Their approach enables intuitive and interactive exploration of style transfer effects, empowering users to create customized stylized images.

# 3.DESIGN:

**Requirement Specifications**

## Hardware Requirements

## System

## RAM

## Hard Disk

## Input

## Output

## Software Requirements

* + - **OS**
    - **Platform**
    - **Program Language**

# 4. METHODOLOGY:

Data Processing and Visualization

The initial phase of the project involves data processing and visualization. This step is crucial for understanding the characteristics of the dataset and preparing it for further analysis. Data processing may include tasks such as cleaning, normalization, and feature extraction, while data visualization techniques like histograms, scatter plots, and heatmaps help in gaining insights into the data's distribution and relationships.

Model Application

Supervised Learning

Our project employs supervised learning, as our dataset comprises labeled data, including both target variables and feature variables. Supervised learning involves training a model on a labeled dataset, where the algorithm learns from the input-output pairs provided during the training phase.

Train-Test Split

The dataset is partitioned into two subsets: the training set and the testing set. The training set is used to train the model, allowing it to learn patterns and relationships within the data. The testing set, which is kept separate from the training set, is used to evaluate the model's performance and generalization ability.

Model Training

The training phase involves feeding the training set into the selected machine learning model. During training, the model iteratively adjusts its parameters to minimize a chosen loss function, aiming to improve its predictive capability. The model learns to map the input features to the corresponding target values, capturing the underlying patterns in the data.

Model Evaluation

Once the model has been trained, it is evaluated using the testing set. Evaluation metrics such as accuracy, precision, recall, F1-score, and confusion matrix are utilized to assess the model's performance on unseen data. This evaluation step provides insights into how well the model generalizes to new, unseen instances and helps identify any potential issues such as overfitting or underfitting.

Iterative Process

The methodology outlined above is often an iterative process, where adjustments and refinements are made based on the performance of the initial model. This may involve fine-tuning hyperparameters, selecting different algorithms, or exploring additional features to improve model performance.

**4.1 Logistic Regression Algorithm:**

Logistic regression, although primarily used for binary classification tasks in traditional machine learning, can be adapted for certain aspects of image style transfer. In the context of image style transfer, logistic regression serves as a foundational understanding of parameter estimation and optimization. While logistic regression inherently deals with continuous values, its application in image style transfer involves the conversion of image representations into categorical values using an activation function such as the sigmoid function. Through this process, logistic regression aids in the initial stages of feature extraction and representation, laying the groundwork for subsequent transformations in the style transfer process.

**4.2XGBoost Classifier:**

XGBoost, renowned for its efficacy in handling structured/tabular data and its exceptional performance in various machine learning competitions, presents an intriguing avenue for image style transfer. Despite its primary application in classification and regression tasks, XGBoost's versatility and robustness make it a compelling candidate for augmenting traditional style transfer methodologies. By leveraging its ability to capture complex relationships within data and optimize predictive performance, XGBoost can contribute to enhancing the fidelity and coherence of stylized images. Through careful feature engineering and model calibration, XGBoost can potentially enrich the style transfer process by infusing nuanced patterns and textures into the synthesized images, thereby elevating the overall visual quality and perceptual realism.

XGBoost, short for eXtreme Gradient Boosting, is a powerful machine learning algorithm that belongs to the class of gradient boosting algorithms. It is designed to efficiently handle structured/tabular data and is widely used for both regression and classification tasks. XGBoost has gained popularity for its scalability, speed, and superior performance in various machine learning competitions and real-world applications.

* import numpy as np
* import pandas as pd
* from sklearn.model\_selection import train\_test\_split
* from sklearn.metrics import accuracy\_score
* import xgboost as xgb

# 4. 3 Support Vector Machine algorithm:

Support Vector Machine (SVM) algorithm, renowned for its proficiency in constructing hyperplanes to separate distinct classes of data points, offers a unique perspective on image style transfer. While traditionally employed for classification and regression tasks, SVM's capacity to delineate meaningful boundaries between disparate classes can be harnessed to delineate stylistic elements within images. By treating image stylization as a classification problem, SVM endeavors to identify and extract salient features that characterize distinct stylistic traits. Through meticulous adjustment of kernel functions and optimization parameters, SVM seeks to delineate the boundaries between content and style representations, facilitating the synthesis of visually compelling and harmonious stylized images.

# CODE FOR THE PROJECT:

# import os

# import tensorflow as tf

# # Load compressed models from tensorflow\_hub

# os.environ['TFHUB\_MODEL\_LOAD\_FORMAT'] = 'COMPRESSED'

# import IPython.display as display

# import matplotlib.pyplot as plt

# import matplotlib as mpl

# mpl.rcParams['figure.figsize'] = (12, 12)

# mpl.rcParams['axes.grid'] = False

# import numpy as np

# import PIL.Image

# import time

# import functools

# def tensor\_to\_image(tensor):

# tensor = tensor\*255

# tensor = np.array(tensor, dtype=np.uint8)

# if np.ndim(tensor)>3:

# assert tensor.shape[0] == 1

# tensor = tensor[0]

# return PIL.Image.fromarray(tensor)

# content\_path = tf.keras.utils.get\_file('YellowLabradorLooking\_new.jpg', 'https://storage.googleapis.com/download.tensorflow.org/example\_images/YellowLabradorLooking\_new.jpg')

# style\_path = tf.keras.utils.get\_file('kandinsky5.jpg','https://storage.googleapis.com/download.tensorflow.org/example\_images/Vassily\_Kandinsky%2C\_1913\_-\_Composition\_7.jpg')

# def load\_img(path\_to\_img):

# max\_dim = 512

# img = tf.io.read\_file(path\_to\_img)

# img = tf.image.decode\_image(img, channels=3)

# img = tf.image.convert\_image\_dtype(img, tf.float32)

# shape = tf.cast(tf.shape(img)[:-1], tf.float32)

# long\_dim = max(shape)

# scale = max\_dim / long\_dim

# new\_shape = tf.cast(shape \* scale, tf.int32)

# img = tf.image.resize(img, new\_shape)

# img = img[tf.newaxis, :]

# return img

# def imshow(image, title=None):

# if len(image.shape) > 3:

# image = tf.squeeze(image, axis=0)

# plt.imshow(image)

# if title:

# plt.title(title)

# content\_image = load\_img(content\_path)

# style\_image = load\_img(style\_path)

# plt.subplot(1, 2, 1)

# imshow(content\_image, 'Content Image')

# plt.subplot(1, 2, 2)

# imshow(style\_image, 'Style Image')

# import tensorflow\_hub as hub

# hub\_model = hub.load('https://tfhub.dev/google/magenta/arbitrary-image-stylization-v1-256/2')

# stylized\_image = hub\_model(tf.constant(content\_image), tf.constant(style\_image))[0]

# tensor\_to\_image(stylized\_image)

# RESULT/OUTPUT:

# WhatsApp Image 2024-05-05 at 15.57.51_422d71a5

# WhatsApp Image 2024-05-05 at 15.58.16_ac78c48d

# 7. CONCLUSION:

In conclusion, image style transfer using machine learning represents a captivating intersection of art and technology, offering limitless possibilities for creative expression and visual exploration. Through the utilization of deep learning algorithms, advanced neural architectures, and innovative techniques, we have witnessed remarkable progress in generating stylized images that seamlessly blend content and style.

**8. FUTURE SCOPE:**

# The future of image style transfer holds promise for further advancements, with opportunities to enhance accuracy, diversity, and usability. By exploring ensemble techniques, fine-tuning hyperparameters, and integrating real-time data feeds, we can improve the robustness and relevance of stylized results. Moreover, the development of user-friendly interfaces, scalability optimization, and integration with IoT devices will empower users to unleash their creativity and personalize their stylization experiences.

# Ultimately, image style transfer using machine learning opens doors to new realms of artistic expression, allowing individuals to transform ordinary images into captivating works of art. As technology continues to evolve, we can look forward to even more exciting developments in this field, revolutionizing the way we perceive and interact with visual media.

# 9. REFERENCES:

[1] P. Goswami and Srividya, “A novel Neural Network design for long range prediction of rainfall pattern,” Current Sci. (Bangalore), vol. 70, no. 6, pp. 447-457, 1996.

[2] C. Venkatesanet, S. D. Raskar , S. S. Tambe , B. D. Kulkarni , and R. N. Keshavamurty , “Prediction of all India summer monsoon rainfall using Error-Back-Propagation Neural Networks,” Meteorology and Atmospheric Physics, pp. 225-240, 1997.

[3] A. K. Sahai, M. K. Soman, and V. Satyan, “All India summer monsoon rainfall prediction using an Artificial Neural Network,” Climate dynamics, vol. 16, no. 4, pp. 291-302, 2000.

[4] N. S. Philip and K. B. Joseph, “On the predictability of rainfall in Kerala-An application of ABF neural network,” Computational ScienceICCS, Springer Berlin Heidelberg, pp. 1-12, 2001.

[5] N. S. Philip and K. B. Joseph, “A Neural Network tool for analyzing trends in rainfall,” Comput. &Geosci., vol. 29, no. 2, pp. 215-223, 2003.

[6] N. Chantasut, C. Charoenjit, and C. Tanprasert, “Predictive mining of rainfall predictions using artificial neural networks for Chao Phraya River,” 4th Int Conf. of the Asian Federation of Inform. Technology in Agriculture and the 2nd World Congr. on Comput. in Agriculture and Natural Resources, Bangkok, Thailand, pp. 117-122, 2004.

[7] V. K. Somvanshi, O. P. Pandey, P. K. Agrawal, N.V. Kalanker1, M.Ravi Prakash, and Ramesh Chand, “Modeling and prediction of rainfall using Artificial Neural Network and ARIMA techniques,” J. Ind. Geophys. Union, vol. 10, no. 2, pp. 141-151, 2006.

[8] S. Chattopadhyay, “Anticipation of summer monsoon rainfall over India by Artificial Neural Network with Conjugate Gradient Descent Learning,” arXiv preprint nlin/0611010, pp. 2-14, 2006.

[9] S. Chattopadhyay and M. Chattopadhyay, “A Soft Computing technique in rainfall forecasting,” Int. Conf. on IT, HIT, pp. 19-21, 2007.

[10] S. Chattopadhyay and G. Chattopadhyay, “Comparative study among different neural net learning algorithms applied to rainfall time series, "Meteorological applicat., vol. 15, no. 2, pp. 273-280, 2008.

[11] P.Guhathakurta, “Long lead monsoon rainfall prediction using deterministic Artificial Neural Network model,” Meteorology and Atmospheric Physics 101, pp. 93-108, 2008.

[12] C. L. Wu, K. W. Chau, and C. Fan, “Prediction of rainfall time series using Modular Artificial Neural Networks coupled with datapreprocessing techniques," J. of hydrology, vol. 389, no. 1, pp. 146- 167, 2010. [13] K. K. Htike and O. O. Khalifa, “Rainfall forecasting models using Focused Time-Delay Neural Networks,” Comput. and Commun. Eng. (ICCCE), Int. Conf. on IEEE, 2010

[14] S. Kannan,Subimal Ghosh, “Prediction of daily rainfall state in a river basin using statistical downscaling from GCM output”, Springer-Verlag, July- 2010.

[15] M.Kannan, S.Prabhakaran, P.Ramachandran, “Rainfall Forecasting Using Data Mining Technique”, International Journal of Engineering and Technology Vol.2 (6), 397-401, 2010.

[16] G. Geetha and R. S. Selvaraj, “Prediction of monthly rainfall in Chennai using Back Propagation Neural Network model,” Int. J. of Eng. Sci. and Technology, vol. 3, no. 1, pp. 211 213, 2011.

[17] M. A. Sharma and J. B. Singh, “Comparative Study of rainfall forecasting models,” New York Sci. J., pp. 115-120, 2011.

[18] J. Abbot and J. Marohasy, “Application of Artificial Neural Networks to rainfall forecasting in Queensland, Australia,” Advances in Atmospheric Sci.,vol. 29, no. 4, pp. 717-730, 2012.

[19] A. Kumar, A. Kumar, R. Ranjan, and S. Kumar, “A rainfall prediction model using artificial neural network,” Control and Syst. Graduate Research Colloq. (ICSGRC), pp. 82-87, 2012.

[20] R. R. Deshpande, “On the rainfall time series prediction using Multilayer Perceptron Artificial Neural Network,” Int. J. of Emerging Technology and Advanced Eng., vol. 2, no. 1, pp. 148-153, 2012.

[21] Soo-Yeon Ji, Sharad Sharma, Byunggu Yu, Dong Hyun Jeong, “Designing a Rule-Based Hourly Rainfall Prediction Model”, IEEE IRI 2012, August – 2012.

[22] G. Shrivastava, S. Karmakar, and M. K. Kowar, “BPN model for long range forecast of monsoon rainfall over a very small geographical region and its verification for 2012,” Geofizika, vol. 30, no. 2, pp. 155-172, 2013. [23] C. L Wu and K. W. Chau, “Prediction of rainfall time series using modular soft computing methods,” Eng. Applicat. of Artificial Intell., vol. 26, no. 3, pp. 997-1007, 2013.

[24] S. K. Nanda, D. P. Tripathy, S. K. Nayak, and S. Mohapatra, “Prediction of rainfall in India using Artificial Neural Network (ANN) models,” Int. J. of Intell. Syst. and Applicat., vol. 5, no. 12, pp. 1-22, 2013.

[25] A. R. Naik and S. K. Pathan, “Indian monsoon rainfall Classification and Prediction using Robust Back Propagation Artificial Neural Network,” Int. J. of Emerging Technology and Advanced Eng., vol. 3, no. 11, pp. 99-101, 2013.