# WHITE BOX CARTOONIZATION USING AN EXTENDED GAN FRAMEWORK

Mini- Project


***Project by:***

| | | |
|---|---|---|
| AMEY THAKUR | TE COMPS B-50 | TU3F1819127 |
| HASAN RIZVI | TE COMPS B-51 | TU3F1819130 |
| MEGA SATISH | TE COMPS B-58 | TU3F1819139 |

What WBC model is capable of?

Short clip in cartoonized version


A. A frame in animation “Garden of words”


B. A real photo processed by WBC method

Comparison between a real cartoon image and an image processed by our method.

# ABSTRACT

- We propose to implement a new framework for estimating generative models via an adversarial process to extend existing GAN framework and develop a white-box controllable image cartoonization, which can generate high-quality cartoonized images from real-world photos.

- We suggest to separately identify three white-box representations from images: the surface representation that contains a smooth surface of cartoon images, the structure representation that refers to the sparse colour-blocks and flatten global content in the celluloid style workflow, and the texture representation that reflects high-frequency texture, contours, and details in cartoon images.

- The learning objectives are separately based on each extracted representations, making our framework controllable and adjustable. The project demonstrates the potential of the framework through qualitative and quantitative evaluation of the generated samples.

# INTRODUCTION

- Cartoons are a very popular art form that has been widely applied in diverse scenes, from publication in printed media to storytelling for children. Some cartoon artwork were created based on real world scenes. However, manually re-creating real life based scenes can be very laborious and requires refined skills.

- The evolution in the field of Machine Learning has expanded the possibilities of creating visual arts. Some famous products have been created by turning real-world photography into usable cartoon scene materials, where the process is called image cartoonization.

- White box cartoonization is a method which reconstructs high quality real life pictures into exceptional cartoon images using GAN framework.

# LITERATURE SURVEY

- Cartoonization is itself a classic art but the evolution in the field of Artificial Intelligence has opened many opportunities. Many models have been developed to generate cartoon images from pictures, but have many drawbacks.

- CartoonGAN is one of the technology to generate cartoonized images but it adds noise and reduce quality of image. On the other hand White Box Cartoonization overcomes these problems and results in more precise and sharp images.

- This project will be based on the model proposed by Xinrui Wang and Jinze Yu - Learning to Cartoonize Using White-Box Cartoon Representations. The main objective of this project is to develop modern cartoon animation workflows which allow artists to use a variety of sources to create content.

# PROBLEM STATEMENT

- AI-powered cartoonization has many practical applications these days — from personalized anime-style avatars to video and even fine art. Many black-box cartoonization frameworks however provide users with limited control or adjustability when rendering real-world photography into cartoon scenes.

- So we are proposing a less arduous web app capable of cartoonize images with lots of precision and accuracy just by given the input image into a GAN framework which will give as an output a cartoon image.

# OBJECTIVES

The objectives of this project are:

- Develop modern cartoon animation workflows which allow artists to use a variety of sources to create content without the need of drawing.
- Deploy GAN framework and create a Web application that can convert any images into high quality cartoon images by simply uploading the picture online and getting the cartoon image.

# SCOPE

- Generative Adversarial Networks (GAN) have attained remarkable results in image enhancement and augmentation. Currently, our project focuses on the cartoonization of high-quality images.

- Further, deep work and development in this white box cartoonization can lead to many various applications:

    1. Generates quick prototypes or sprites for anime, cartoons and games.
    2. Since it subdues facial features and information in general, it can be used to generate minimal art.
    3. Games can import shortcut scenes very easily without using motion-capture.
    4. Can be modelled as an assistant to graphic designers or animators.

# TIMELINE

**WEEK 1**

**Research Paper**

Reading and collecting relevant information.
Defining Requirements.

**WEEK 3**

**Comprehension of Mathematical concepts and Algorithms**

**WEEK 5**

**Start with the implementation of WBC Model**

**WEEK 7**

**Start with the deployment of WBC Model**

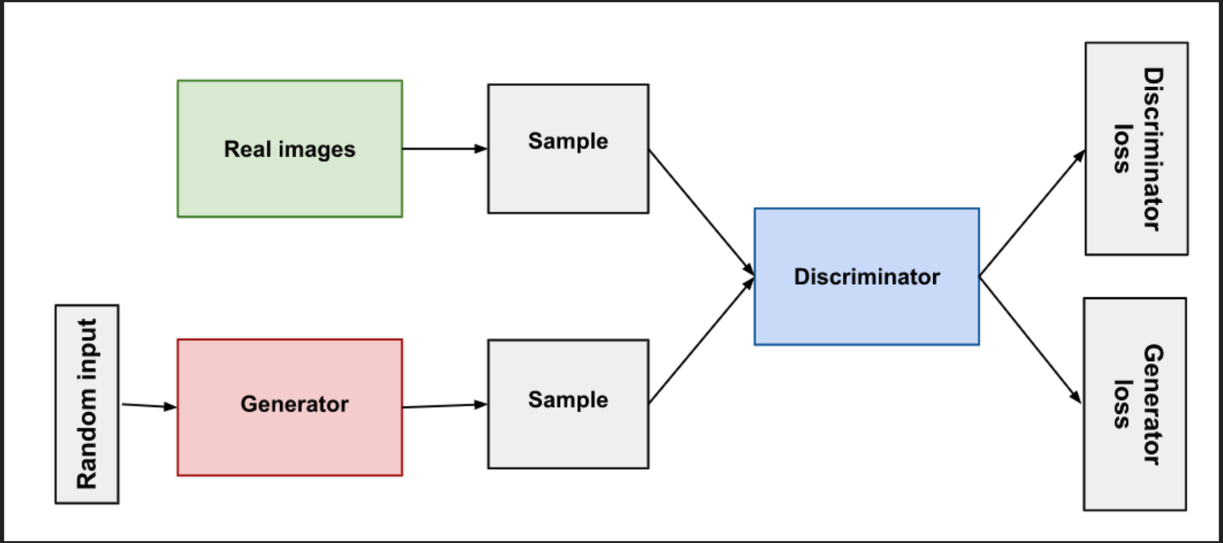
**WEEK 9**

**Prototype Developed Successfully**

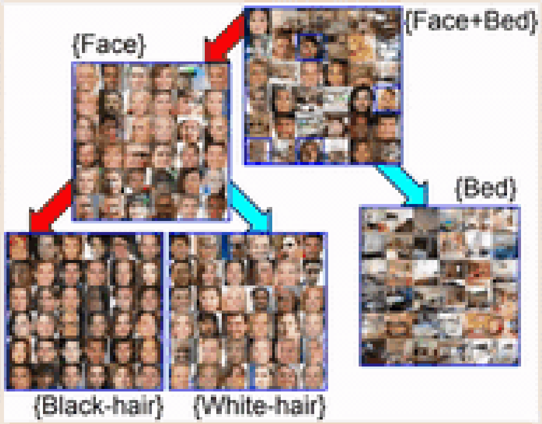
# GAN-TREE: A GAN FRAMEWORK FOR GENERATING DIVERSE DATA

- Generative Adversarial Networks (GANs) are a class of Neural Networks, which can be used for modeling the underlying data distribution from a given set of unlabeled samples.
- GAN comprises of two competing networks, a generator, and a discriminator.
- The generator learns to map a random vector to the high dimensional data such as image, while the discriminator network learns to discriminate between the real and the synthesized samples.
- Finally, this competition leads to the generation of realistic samples, which are undiscriminating to human observers. These models are of prime importance in modern AI research with enormous potential towards realizing Generalized AI.

# ARCHITECTURE OF GAN




Architecture of GAN


{Face+Bed}
{Face}
{Bed}
{Black-hair}
{White-hair}

## PROS OF GAN FRAMEWORK

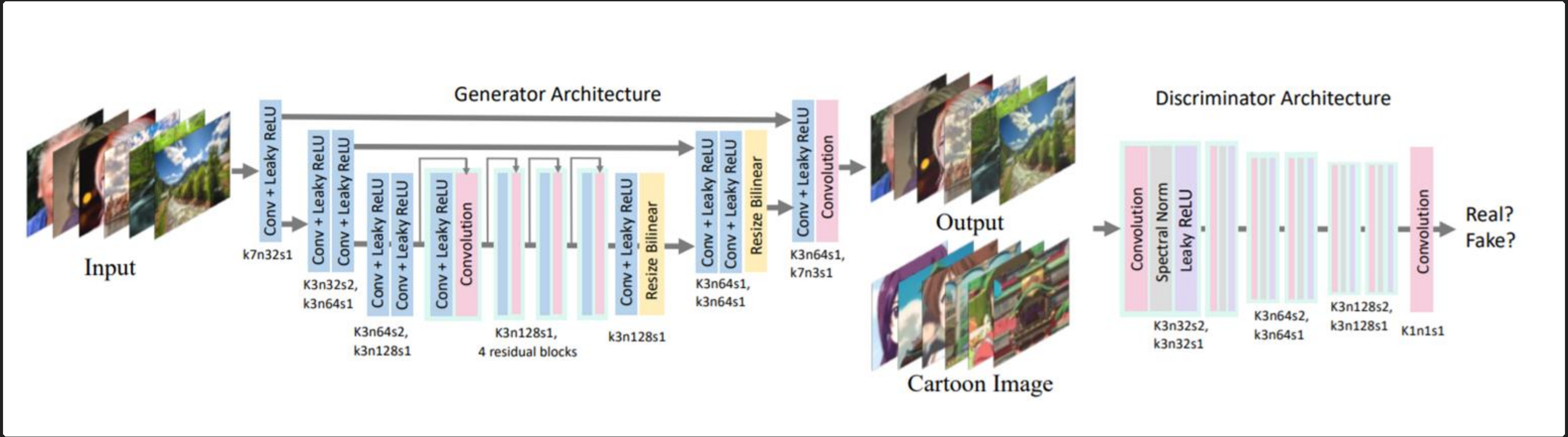
+ Very impressive result related to content generated, and realistic visual content.
+ Also can be modelled for other applications like image restoration, etc.

## CONS OF GAN FRAMEWORK

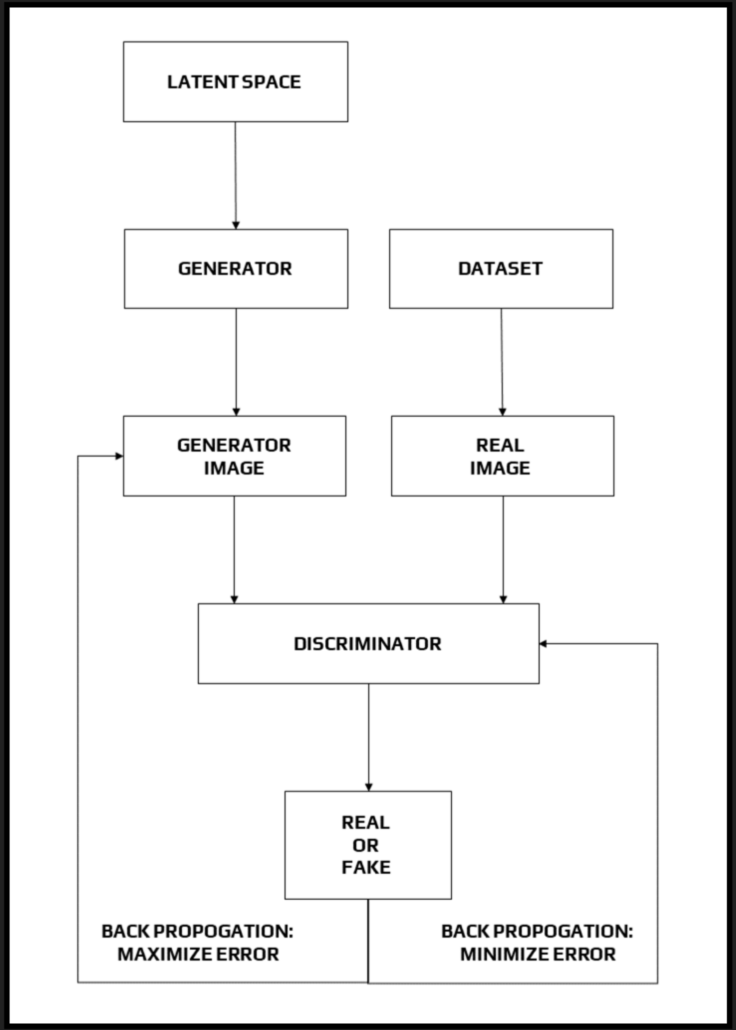
Despite the success of GAN, the potential of such a framework has certain limitations.

- Difficult for training: huge computation.
- "Realistic" but not Real. A fake pattern can be created, especially for small scale structure or non-nature object like text ---> problem of the loss function.
- Some physical effects in the picture like shadow will become unrealistic.

# ARCHITECTURE OF THE PROPOSED WBC MODEL




Architecture of WBC Model




Block Diagram of WBC Architecture

# METHODOLOGIES OF WHITE BOX CARTOONIZATION

We propose to separately identify three white-box representations from images:

1. The surface representation
2. The structure representation
3. The texture representation

- The surface representation that contains a smooth surface of cartoon images.
- The structure representation that refers to the sparse color-blocks and flatten global content in the celluloid style workflow.
- The texture representation that reflects high frequency texture, contours, and details in cartoon images.
- A Generative Adversarial Network (GAN) framework is used to learn the extracted representations and to cartoonize images.

# SURFACE REPRESENTATION

- The surface representation imitates cartoon painting style where artists roughly draw drafts with coarse brushes and have smooth surfaces similar to cartoon images.

- To smooth images and meanwhile keep the global semantic structure, a differentiable guided filter is adopted for edge preserving filtering.

- Edge-preserving filtering is an image processing technique that smooths away noise or textures while retaining sharp edges. Examples are the median, bilateral, guided, and anisotropic diffusion filters.

# SURFACE LOSS FORMULA

$$L_{surface}(G, D_s) = \log D_s(F_{dgf}(I_c, I_c)) + \log(1 - D_s(F_{dgf}(G(I_p), G(I_p))))$$

Where,

$G$ = Generator,

$D_s$ = Discriminator,

$I_c$ = Reference Cartoon Image,

$I_p$ = Input Photo,

$F_{dgf}$ = It takes an image I as input and itself as guide map, returns extracted surface representation Fdgf (I, I) with textures and details removed.

Note: A discriminator Ds is introduced to judge whether model outputs and reference cartoon images have similar surfaces, and guide the generator G to learn the information stored in the extracted surface representation.

# STRUCTURE REPRESENTATION

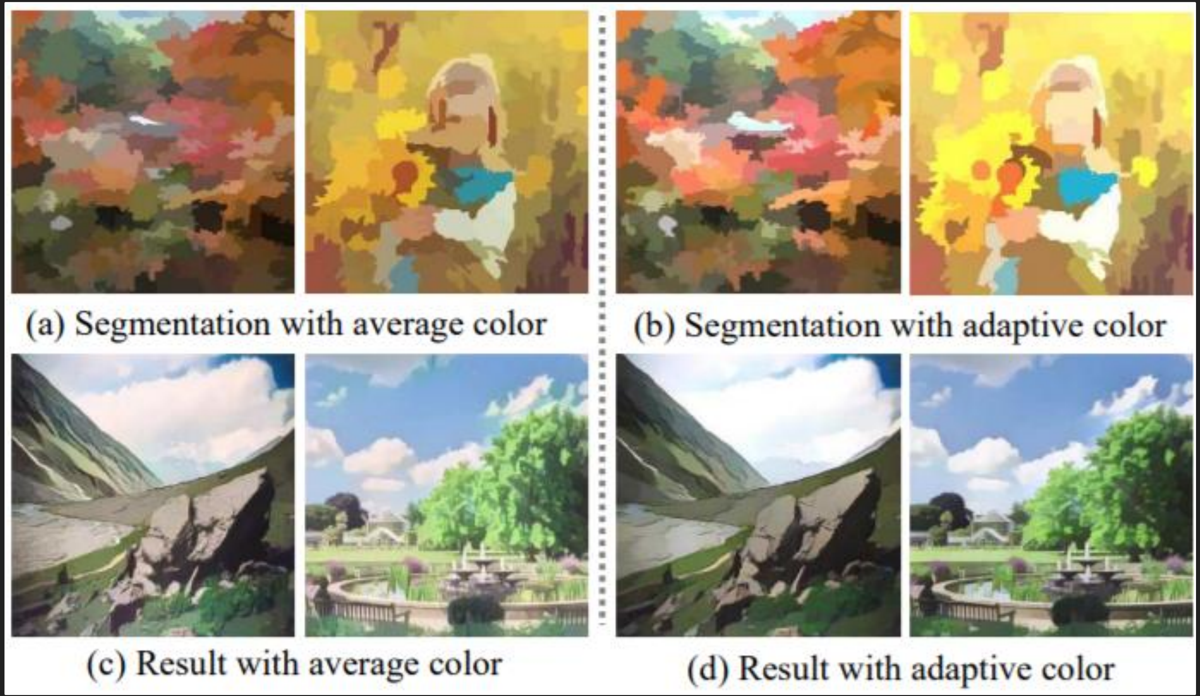
- We at first use felzenszwalb algorithm to segment images into separate regions. As superpixel algorithms only consider the similarity of pixels and ignore semantic information, we further introduce selective search to merge segmented regions and extract a sparse segmentation map.

- Standard superpixel algorithms color each segmented region with an average of the pixel value. By analyzing the processed dataset, we found this lowers global contrast, darkens images, and causes hazing effect on the final results. We thus propose an adaptive coloring algorithm

- Adaptive coloring formula, $$S_{i,j} = (\theta_1 * \bar{S} + \theta_2 * \check{S})^{\mu}$$

$$(\theta 1, \theta 2) = \begin{cases} (0, 1) & \sigma(S) < \gamma 1, \\ (0.5, 0.5) & \gamma 1 < \sigma(S) < \gamma 2, \\ (1, 0) & \gamma 2 < \sigma(S). \end{cases}$$

where we find $\gamma 1 = 20$, $\gamma 2 = 40$ and $\mu = 1.2$ generate good results.
This effectively enhances the contrast of images and reduces hazing effect.




Adaptive coloring algorithm. (a) and (b) show segmentation maps with different coloring method, while (c) and (d) shows results generated with different coloring method.
Adaptive coloring generates results that are brighter and free from hazing effects.

## STRUCTURE LOSS FORMULA

$$L_{structure} = \| VGG_n(G(I_p)) - VGG_n(F_{st}(G(I_p))) \|$$

Where,

$G$ = Generator,

$I_p$ = Input Photo,

$F_{st}$ = Structure Representation Extraction.

Note: We use high-level features extracted by a pre-trained VGG16 network to enforce spatial constraint between our results and extracted structure representation.

# TEXTURE REPRESENTATION

- The high-frequency features of cartoon images are key learning objectives, but luminance and color information make it easy to distinguish between cartoon images and real-world photos. We thus propose a random color shift algorithm. Random color shift can generate random intensity maps with luminance and color information removed.

- $F_{rcs}$ to extract single-channel texture representation from color images, which retains high-frequency textures and decreases the influence of color and luminance.

$$F_{rcs}(I_{rgb}) = (1 - \alpha)(\beta_1 * I_r + \beta_2 * I_g + \beta_3 * I_b) + \alpha * Y$$

Where,

$I_{rgb}$ represents 3-channel RGB color images, $I_r$, $I_g$ and $I_b$ represent three color channels, and $Y$ represents standard grayscale image converted from RGB color image.

**Note:** We set $\alpha = 0.8$, $\beta_1$, $\beta_2$ and $\beta_3 \sim U(-1, 1)$.

## TEXTURAL REPRESENTATION FORMULA

$$L_{texture}(G, D_t) = \log D_t(F_{rcs}(I_c)) + \log(1 - D_t(F_{rcs}(G(I_p))))$$

Where,

$G$ = Generator,

$D_t$ = Discriminator,

$I_c$ = Reference Cartoon Image,

$I_p$ = Input Photo,

$F_{rcs}$ = Extract single-channel texture representation from color images, which retains high-frequency textures and decreases the influence of color and luminance.

# TOOLS AND TECHNOLOGIES USED

## TOOLS USED

- Anaconda
- Jupyter Notebook
- TensorFlow.js
- Bootstrap
- Flexbox
- GitHub Pages
- Colab

## TECHNOLOGIES USED

- HTML
- CSS
- JavaScript
- Python

# IMPLEMENTATION

**GitHub Repository -** https://github.com/rizvihasan/whitebox_cartoonisation-webapp

- We implemented our GAN method with TensorFlow.
- Patch discriminator is adopted to simplify calculation and enhance discriminative capacity.
- We use Adam algorithm to optimize both Generator and Discriminator networks.
- Learning rate and batch size are set to $2 * 10{-}4$ and 16 during training. We at first pre-train the generator with the content loss for 50000 iterations, and then jointly optimize the GAN based framework. Training is stopped after 100000 iterations or on convergence.

# DATASET

- Human face and landscape data are collected for generalization on diverse scenes.
- For real-world photos, we collect 10000 images from the FFHQ dataset for the human face and 5000 images from the dataset in for landscape.
- For cartoon images, we collect 10000 images from animations for the human face and 10000 images for landscape.
- For the validation set, we collect 3011 animation images and 1978 real-world photos.
- Images shown in the main paper are collected from the DIV2K dataset, and images in user study are collected from the Internet and Microsoft COCO dataset.
- During training, all images are resized to 256*256 resolution, and face images are feed only once in every five iterations.

# TIME PERFORMANCE AND MODEL SIZE

- Our model is the fastest among four methods on all devices and all resolutions, and has the smallest model size.
- Especially, our model can process a 720*1280 image on GPU within only 17.23ms, which enables it for real-time High-Resolution video processing tasks.
- Generality to diverse use cases: We apply our model on diverse real-world scenes, including natural landscape, city views, people, animals, and plants.

# CartoonGAN

- CartoonGAN, a generative adversarial network (GAN) framework for cartoon stylization.This method takes a set of photos and a set of cartoon images for training. To produce high quality results while making the training data easy to obtain, pairing or correspondence between two sets of images is not required.

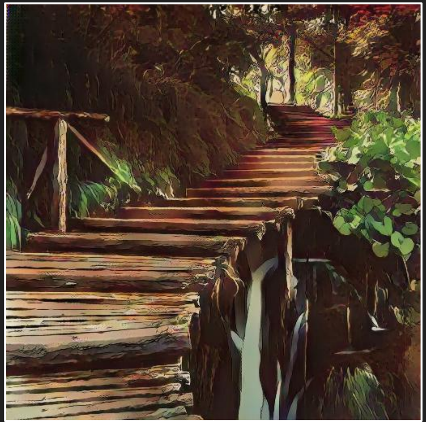
- CartoonGAN is basically a dedicated GAN-based architecture together with two simple yet effective loss functions. When cartoon images from individual artists are used for training, our method is able to reproduce their styles.

# GANILLA

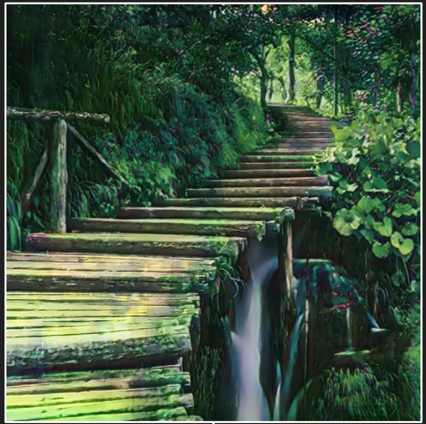
- GANILLA stands for Generative Adversarial Networks for Image to Illustration Translation.
- Illustrations in children's books has opened up a new domain in unpaired image-to-image translation. Currently, state-of-the-art image-to-image translation models successfully transfer either the style or the content, they fail to transfer both at the same time.
- GANILLA is a new network to address this issue and show that the resulting network strikes a better balance between style and content.
- So far, the success of image translation models has been based on subjective, qualitative visual comparison on a limited number of images. GANILLA is a framework for the quantitative evaluation of image-to-illustration models, where both content and style are taken into account using separate classifiers.

# COMPARISON BETWEEN CARTOONGAN, GANILLA AND WBC

Input Image



CartoonGAN



GANILLA

WBC

# COMPARISON BETWEEN CARTOONGAN, GANILLA AND WBC

Input Image

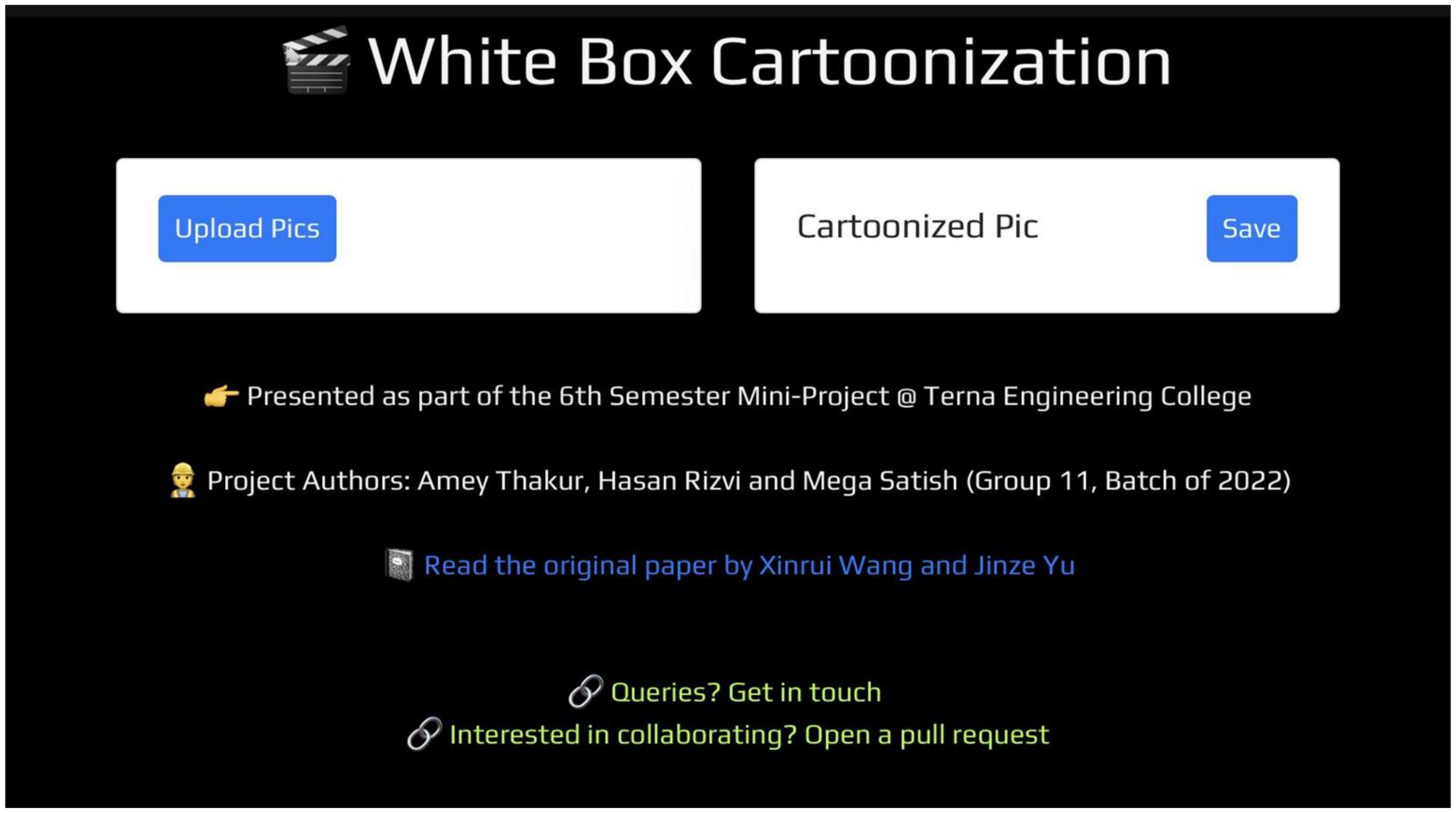
CartoonGAN

GANILLA

WBC

# RESULTS

**Web App -** https://rizvihasan.github.io/whitebox_cartoonisation-webapp



# 🎬 White Box Cartoonization

Upload Pics


Cartoonized Pic

Save


👉 Presented as part of the 6th Semester Mini-Project @ Terna Engineering College

👷 Project Authors: Amey Thakur, Hasan Rizvi and Mega Satish (Group 11, Batch of 2022)


Cartoonized
Image
→

←
Test
Image

# 🎬 White Box Cartoonization

Upload Pics

Cartoonized Pic

Save

👉 Presented as part of the 6th Semester Mini-Project @ Terna Engineering College

👷 Project Authors: Amey Thakur, Hasan Rizvi and Mega Satish (Group 11, Batch of 2022)

Cartoonized
Image
→

←
Test
Image

# 🎬 White Box Cartoonization

Upload Pics

Cartoonized Pic

Save

👉 Presented as part of the 6th Semester Mini-Project @ Terna Engineering College

👷 Project Authors: Amey Thakur, Hasan Rizvi and Mega Satish (Group 11, Batch of 2022)

📓 Read the original paper by Xinrui Wang and Jinze Yu

🔗 Queries? Get in touch

🔗 Interested in collaborating? Open a pull request

Test Image

Cartoonized Image

# 🎬 White Box Cartoonization

Upload Pics

Cartoonized Pic Save

👉 Presented as part of the 6th Semester Mini-Project @ Terna Engineering College

👷 Project Authors: Amey Thakur, Hasan Rizvi and Mega Satish (Group 11, Batch of 2022)

📓 Read the original paper by Xinrui Wang and Jinze Yu

🔗 Queries? Get in touch
🔗 Interested in collaborating? Open a pull request

Test Image

Cartoonized Image

# CONCLUSION

- We have proposed a deployed white-box controllable image cartoonization framework based on GAN, which can generate high-quality cartoonized images from real-world photos.

- First, images are decomposed into three cartoon representations: the surface representation, the structure representation, and the texture representation. Corresponding image processing modules are used to extract three representations for network training. Finally, extensive quantitative and qualitative experiments have been conducted to validate the performance of our method.

- Our Web App is able to produce a cartoonized image from an input image. The output result is very similar to the real image yet depicts characteristics of cartoons. Also the time for conversion is less compared to the previous system and this web application works on any browser without having any specific requirements.

# REFERENCES


- https://arxiv.org/abs/2002.05638
- https://arxiv.org/abs/1406.2661
- https://systemerrorwang.github.io/White-box-Cartoonization/
- https://www.researchgate.net/publication/329748755_CartoonGAN_Generative_Adversarial_Networks_for_Photo_Cartoonization
- https://www.researchgate.net/publication/343457093_Learning_to_Cartoonize_Using_White-Box_Cartoon_Representations
- https://www.tensorflow.org/js/models
- https://developer.mozilla.org/en-US/docs/Web/API/CanvasRenderingContext2D

# THANK YOU