Deep Learning based Low Light Image Enhancement for Advanced Driver Assistance Systems

Project report for IITB EE610 Image Processing 2021

Sunaina Saxena
Department of EE
IIT Bombay
Mumbai, India
213070001@iitb.ac.in

Nihar Mahesh Gupte
Department of EE
IIT Bombay
Mumbai, India
213070002@iitb.ac.in

Harsh Diwakar
Department of EE
IIT Bombay
Mumbai, India
213070018@iitb.ac.in

Mohit Kumar Meena
Department of EE
IIT Bombay
Mumbai, India
213070021@iitb.ac.in

***Abstract*—Low light image enhancement has always been a challenging task since last few decades especially for the connected autonomous vehicles (CAVs) as it increases the risk of crash in low vision conditions. Although, quite a few amount of work is being done in past to develop an effective and efficient algorithm but still there is a scope of improvement. In this paper, Histogram Equalization(HE) algorithms (e.g. Contrast limited adaptive histogram equalization (CLAHE)), Retinex-based algorithms(e.g. Multi-Scale Retinex (MSR)) and Learning based algorithms(e.g. U-Net, Pix2Pix and Fast-IP Net) are implemented with different parameters settings and their comparison is made quantitatively and qualitatively. Results showed that Pix2Pix and U-Net model outperforms the other methods and generate better results.**

I. Introduction

In reality, when we capture an image in a low light environment, the image quality would be strongly infuenced by noise and low contrast, which makes it more diffcult to deal with the following tasks such as image segmentation, object detection etc. At the present, digital video technology has been widely used in various fields, for example, safety monitoring of important places, surveillance, traffic management, driving assistance and so on. In this paper, The focus is on driving assistance systems for autonomous vehicles and the Dataset used focus primarily on images of roads that are synthetically degraded to low light images as shown in section III. Under the condition of good daytime illumination, the image quality can meet the application requirements, but when it comes to night, the image quality of low light image worsens, which brings a big challenge in digital image processing. Sorts of enhancement methods were proposed and they can be divided into three categories: methods based on histogram equalization algorithms (HE) as discussed in section III (b), Retinex based methods as discussed in section III(b) and Deep-Learning based methods III(c). The contribution of our work can be summed up as three aspects: First of all, we did data preparation, the original image that is taken from BDD100K video is degraded by applying transforms like gamma transformations followed by Histogram matching with several night images as an reference which is thoroughly discussed in section III(a). Secondly, we elaborate the traditional techniques like CLAHE, Single scale retinex (SSR), Multi-scale retinex (MSR), Multi-scale retinex color restoration(MSRCR) and Multi-scale retinex color preservation (MSRCP) as illustrated in section III(b) . And Finally, the performance is evaluated by using Deep learning based techniques like U-Net, Pix2Pix [1] and Fast-IP Net as shown in section III(c). The results were presented in section III (c) and Mean squared error (MSE) loss, Peak signal-to-noise ratio (PSNR) loss and Structural similarity index (SSIM) loss of all the deep learning based techniques III(c) were listed out. Lastly, the conclusion is being made in section V that the performance of Pix2Pix model is better on synthetic as well as on real-world images captured by different camera settings.

II. Background and Prior Work

For this project, the methodologies taught in the course have a heavy influence. The intensity transformation techniques of image processing such as logarithmic transform, gamma transform, histogram equalization and CLAHE were necessary for classical low light image enhancements, whereas the advanced methods included understanding of Deep learning, especially Convolutional Neural Networks (CNNs) as well as the advanced version of CNNs such as Generative Adversial Network (GAN).

Prior works include low light image enhancement using CLAHE (Contrast Limited Adaptive Histogram Equalization), Single scale Retinex(SSR), Multi scale retinex(MSR), Multi scale retinex with colour restoration(MSRCR), Multi scale retinex with colour preservation(MSRCP) [2], CAN (Context aggregation network) [3], U-NET [4], and pix2pix Generative adversial network (GAN) [1]

III. Data and Methodology

A. Data Preparation

In [5] the authors have implemented data set preparation in three steps. Initially, the gamma correction is applied to the image, followed by contrast adjustment of the image using a scalar multiplier α, and finally histogram matching using statistical mean of real night images, multiplied with scalar multiplier β.

$$\gamma \sim uniform(1, 1.4)$$
$$\alpha \sim uniform(0.8, 1)$$
$$\beta \sim uniform(0.8, 1)$$

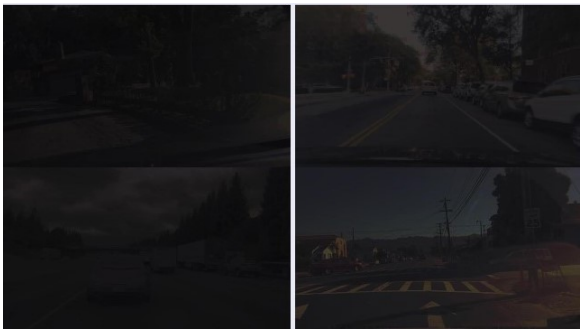

Figure 1: Image degradation using method proposed in [5]

In the project, initially we used the above mentioned method, which failed to provide excellent results, as the images only looked like low brightness images. For the project, we fixed γ, α and β using the following distribution.

$$\gamma \sim uniform(1, 1.4)$$
$$\alpha \sim uniform(0.8, 1)$$
$$\beta \sim uniform(0.8, 1)$$

Apart from this, for histogram matching, we used 11 real night images, and rather than using their statistical mean, one reference was randomly selected from the reference image and then applied to the original image for degradation. The results obtained showed better results than the original method mentioned in [5] as they looked more realistic and more night like images.

B. Conventional Approach

Two major categories of conventional methods are implemented in this project for low light image enhancement. First being Histogram Equalization methods in which we have implemented CLAHE and second one is Retinex based methods. Both of the methods are explained in sections III-B1 and III-B2

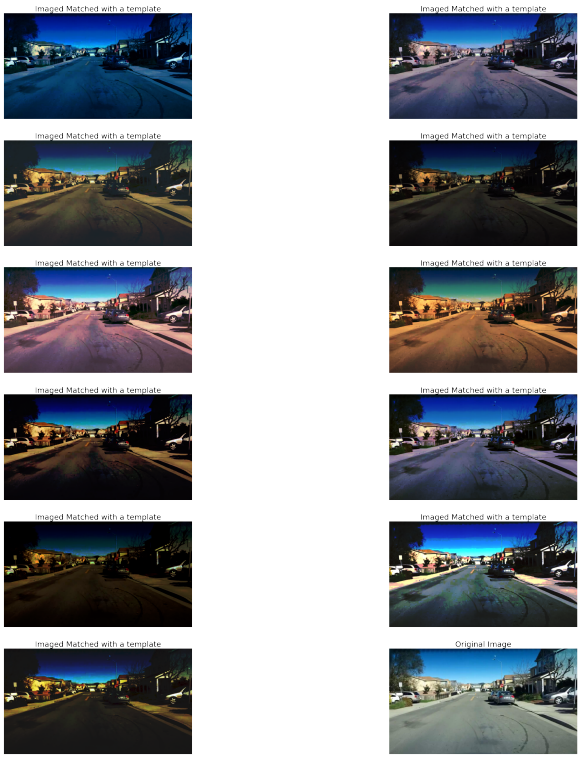

Figure 2: Same Image after histogram matching with different real night images

Figure 3: Image degradation using our method

1) Contrast Limited Adaptive Histogram Equalization(CLAHE): CLAHE performs histogram equalization in small patches or tiles of same dimension and then set desired crop limit for each patch. CLAHE is basically used to optimize the input image by maximizing entropy and limiting its contrast.

$$g = [g_{max} - g_{min}] * P_f + g_{min}$$

where, g is the computed picture element magnitude, g_{max} is the maximum picture element value, g_{min} is the minimum picture element magnitude, p_f is the cumulative distribution function.

2) Retinex Based Algorithms: The Retinex is an image enhancement algorithm[2] which is used to improve the

contrast, brightness and perceived sharpness of images by performing a non-linear spatial/spectral transform to synthesize dynamic range compression and provide colour constancy, thus it removes the effects caused by different illuminations on a scene. Retinex variants, and their performance analysis

- **Single scale retinex (SSR)**
  The basics of SSR include a logarithmic photoreceptor function that approximates the vision system based on a center/surround function.

$$R_i(x,y) = \log I_i(x,y) - \log[I_i(x,y) \circledast F(x,y)]$$

$$F(x,y) = Ce^{\frac{-(x^2+y^2)}{2\sigma_n^2}}$$

  where, R_i is the retinex image, I_i is the applied source image, F is the surround function for each color channel of the image which is a gaussian low pass filter.

- **Multi scale retinex (MSR)**
  To overcome the drawbacks of SSR, MSR was introduced to preserve both the dynamic range compression and color rendition, Multi-scale retinex is a combination of weighted different scales of SSR.

$$R_{MSRi} = \sum_{n=1}^{N} W_n[\log I_i - \log(F_n * I_i)]$$

$$I(x,y) = \frac{(I_r + I_g + I_b)}{3.0}$$

  where N is the number of the scales used and W_n is the weight of the nth scale.

- **Multi scale retinex with colour restoration (MSRCR)**
  As MSR fails in the restoration of image contrast, MSRCR algorithm is introduced. Here color restoration stage is added to the MSR algorithm by multiplying MSR output with a color restoration function.

$$R_{MSRCR_i}(x,y) = C_i.R_{MSR_i}(x,y)$$

$$C_i = f(I_i^{'})$$

$$C_i = .\log(.I_i^{'})$$

$$I_i^{'} = \frac{I_i(x,y)}{\sum_{j=1}^{S} I_j(x,y)}$$

  where,
  $I_i^{'} \rightarrow$ colour balance/restoration
  $S \rightarrow$spectral channels(S=3 for RGB)
  $\beta \rightarrow$gain
  $\alpha \rightarrow$controls non linearity

$$min = min_i(min_(x,y)R_{MSRCR_i}(x,y))$$

$$max = max_i(max_{(x,y)}R_{MSRCR_i}(x,y))$$

$$R_{MSRCR_i}(x,y) = 255.\frac{R_{MSRCR_i}(x,y) - min}{max - min}$$

$$R_{MSRCR_i}(x,y) = G[R_{MSRCR_i}(x,y) - b]$$

  where,
  G and b$\rightarrow$gain and offset parameters

- **Multi scale retinex with colour preservation(MSRCP)**
  MSRCP algorithm uses each image channel"s intensity information and the MSR algorithm to enhance the image. For images having accurate color distribution and exposure of white light, the outcome on implementing MSRCP conserves the color balance.

$$I(x,y) = \frac{\sum_{j=1}^{S} I_j(x,y)}{S}$$

  $S \rightarrow$number of channels

$$R_{MSRCP_i}^{'} = min(\frac{255}{max(I_{R_i} + I_{G_i} + I_{B_i}, \frac{int1_i}{int(i)})})$$

$$R_{MSRCP_i} = G.[R_{MSRCP_i}^{'}] + b$$

  $I_i \rightarrow$input image
  $int_1 \rightarrow$ colour restoration
  G and b$\rightarrow$gain and offset parameters

C. Learning Based Algorithms

1) Context Aggregation Networks: In [3], context aggregation networks (CAN) are used, where the intermediate representation as well as the output have the same size as that of input. The network consists of 2D convolutional layers one after the other, and the convolutions are dilated. One such architecture, CAN24, (shown in 4) consists of 10 convolutional layers with kernel size of 3x3, with 24 filters in each layer. Each convolutional layer performs dilated convolution. The dilation is increased exponentially with depth: $r_s = 2^{s-1}$ for $1sd2$. For L^{d-1}, dilation is not used. For the output layer L^d, a linear transformation (1×1 convolution with no nonlinearity) is used that projects the final layer into the RGB color space. Each layer has the leaky rectified linear unit (LReLU): (x) = max(x, x), where = 0.2. For our project, we rescaled the images to the size of (224x224x3) and fed the images into the CAN24 Network.

D. VGG16 and Residual Learning Network

The method described by [6] includes neural network based on VGG16 [7]. The VGG-16, composed of 16 layers of which only convolutional only layers were considered (13), the dense layers were not considered. Among these groups is performed an operation called maxpool except for the dense network. The second part is composed of a so-called residual learning network (RLN). A similar version we employed in the project is based upon VGG 16 and then concatenating and upsampling the network. The architecture for this network is mentioned in figure 5.

For the original network implemented in [6], the training time lasted around six days running about 6.6 million iterations epochs to predict good results. Due to limited computation

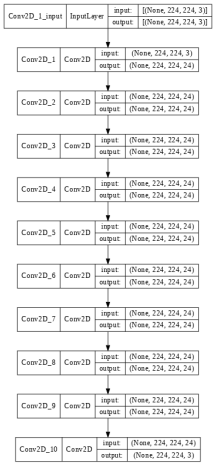

Figure 4: CAN24 architecture

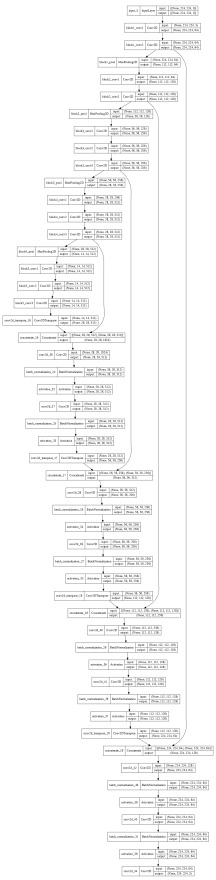

Figure 5: VGG16 and Residual learning Network

power, we could only train around 15 epochs, hence the results were not as good as compared to the original literature.

1) U-net: U-net was originally invented and first used for biomedical image segmentation but can also be used for regression task. Its architecture can be broadly thought of as an encoder network followed by a decoder network. Unlike classification where the end result of the the deep network is the only important thing, semantic segmentation and regression not only requires discrimination at pixel level but also a mechanism to project the discriminative features learnt at different stages of the encoder onto the pixel space.

- The encoder is the first half in the architecture diagram, It is usually pre-trained but we had trained this model on our dataset and then applied convolution blocks followed by a maxpool downsampling to encode the input image into feature representations at multiple different levels.
- The decoder is the second half of the architecture. The goal is to semantically project the discriminative features (lower resolution) learnt by the encoder onto the pixel space (higher resolution) to get a dense regression. The decoder consists of upsampling and concatenation followed by regular convolution operations.
- Instead of Tanh, logistic, arctan or Sigmoid as activation function it uses ReLU function which reduce likelihood of vanishing gradient problem.
- It trains faster than other deeper architectures.

UNet architecture as shown in figure 6 comprises of two 3 × 3 convolutions, followed by Rectified Linear Unit (ReLu) and 2×2 maximum pooling operations with the stride of 2 for down sampling path. In Up sampling path, 2 × 2 transposed convolution operation taken place for reducing the feature channels. Convolution path Skip connections also introduced in the UNet architecture [8]. This connection is used to skip the features from the contracting path to the expanding path in order to recover the spatial feature lost during down sampling operations.So,the regression is very fast and accurate when compared with other regression methods.Specifically, we would like to upsample it to meet the same size with the corresponding concatenation blocks from the left.You may see the gray and green arrows, where we concatenate two feature maps together.

The main contribution of U-Net in this sense is that while upsampling in the network we are also concatenating the higher resolution feature maps from the encoder network with the upsampled features in order to better learn representations with following convolutions [9]. Upsampling is a sparse operation we need a good prior from earlier stages to better represent the localization. The parameters used for the U-Net model Figure* 6 is represented in Table I.

Table I: Parameter description of the model

Description of parameter for U-Net	
Functions used	**Description**
Activation function (Input)	LeakyReLU
Activation function (Output)	ReLU
Optimizer	Adam
Loss function	MSE

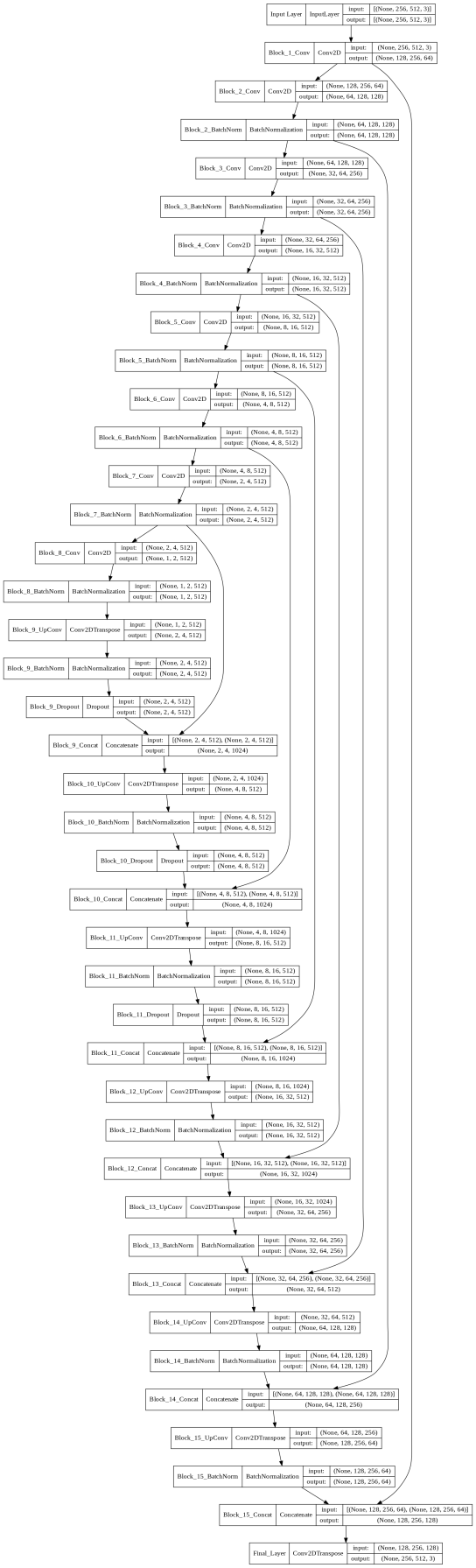

Figure 6: U-Net architecture

2) pix2pix: pix2pix [1] is a conditional generative adversarial network (cGAN) [10] that learns a mapping from an input image to output image. pix2pix can be used for a variety of tasks where the input and output both are images such as synthesizing photos from label maps, generating colour photos from grey-scale images, neural style transfer and for image enhancements. So, we are using the pix2pix to enhance the low light images. As all the Generative adversarial networks (GANs) [11], architecture of pix2pix contains a generator and a discriminator. Generator is based on a U-Net like architecture similar to the one in previous section and discriminator by a convolutional PatchGAN classifier same as proposed in the pix2pix paper. Architecture used in discriminator and generator are shown in Figure 10 and Figure 6 respectively. Generator is the same U-net architecture as described in above section III-D1. Sequential layers in Discriminator architecture is again a convolution without bias followed by a leaky ReLU activation. In the U-net model we are using a loss function to minimize the mean squared error between the predicted image and target image. However in pix2pix we train a discriminator that will learn the differences between target and generated image, where generator will try to minimize this difference and discriminator will try to learn the features to maximize the difference between generated and target image.

Generator and Discriminators are trained in an adversarial manner. The loss functions used in pix2pix is defined below

$$G^* = arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G) \tag{1}$$

where, $\mathcal{L}_{cGAN}$ is conditional GAN loss that is expressed as

$$\begin{aligned} \mathcal{L}_{cGAN}(G, D) =& \mathbb{E}_{x,Y}[logD(x, y)] + \\ & \mathbb{E}_{x,Y}[log(1 - D(x, G(x, y))] \end{aligned} \tag{2}$$

Note that D is the feature map that Discriminator is learning and G is the generator map for translating input low light images into original images, x and y are the degraded and original image respectively for this specific case. L_{L1} is the L1 penalty applied to generator as specified in pix2pix paper.

To achieve the adversarial nature in training a binary cross entropy loss is used for Discriminator that should discriminate target or original images as 1 and the generated image as 0 then gradients are applied to the discriminator to update the features. Similarly, for generator discriminator should label the generator output as 1 i.e. target image and along with that a L1 loss and then the gradients are applied to generator. Adversarial Training procedure is also shown in Figure 7 for Generator and Discriminator.

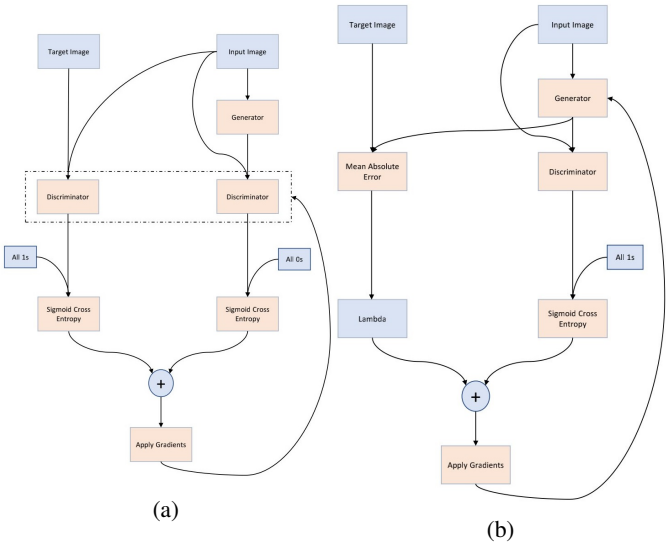

Figure 7: Flow chart for training of (a) Discriminator and (b) Generator

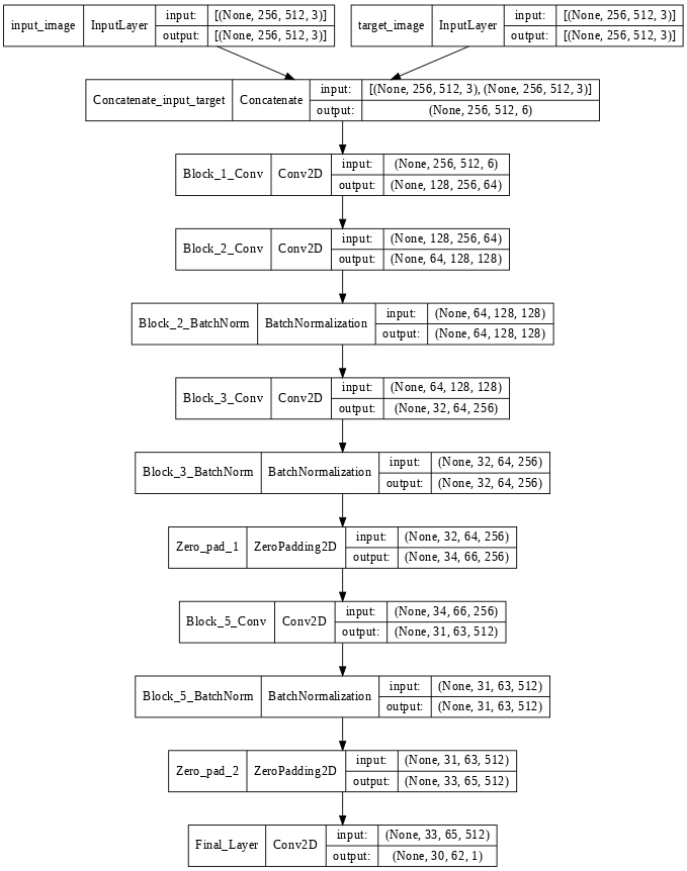

Figure 8: Discriminator Model

There is a hyperparameter lambda involved with the weight of L1 Loss. We found the good results with lambda = 100 that is shown in next section.

IV. Experiment and Results

Experiments were performed on two images, one which was obtained synthetically, and other was real night image. As evident from the results, since epoch time for VGG16 and residual network, as well as CAN network was high, they did not get sufficient time to train, hence their output was poor. However, the models that stood out were pix2pix and UNET. Among classical techniques, MSRCP was superior compared to other methods.

Figure 9: Output of Different Models on synthetically degraded images

Figure 10: Output of Different Models on a real image taken of road at night

Contributions

This is a team project and each member has putted nearly equal efforts. In detailed manner, Nihar Mahesh Gupte has

Table II: Comparisons of results obtained

Parameters	SSIM	MSE	PSNR
CLAHE	0.00	2775.88	-34.43
SSR	0.00	23221.15	-43.66
MSR	0.00	23276.26	-43.67
MSRCR	0.00	14772.99	-41.69
MSRCP	0.00	19322.60	-42.86
VGG	0.06	64.04	-18.06
CAN	0.01	408.77	-26.11
U-net	0.00	10881.05	-40.37
Pix2Pix	0.00	12406.30	-40.94

worked on data preparation, CAN and VGG + residual network, Sunaina Saxena has worked on classical based approaches and reviewed several papers mentioned in Section III, Mohit Kumar Meena has worked on data preparation and implementation of UNET model, finally, Harsh Diwakar has worked on the most satisfying deep learning model namely pix2pix as well as on the creation on data pipeline that helped in the execution of all the models.

V. Learning, Conclusion and Future work

The project covers both the aspects of image processing, classical methods including histogram equalization based CLAHE algorithm, as well as Single scale Retinex(SSR), Multi scale retinex(MSR), Multi scale retinex with colour restoration(MSRCR), Multi scale retinex with colour preservation(MSRCP). Also the advanced deep learning methods including convolutional neural networks like plain CNNs, GAN, VGG16, UNET, etc. were explored in this project. Another important thing was handling a large dataset such as BDD100K which consisted of 10,000 images. This dataset was handled efficiently using the tensorflow pipelining methods learnt through this project. Keras was explored by all of us, also the various metrics for evaluation of a image processing algorithms were learnt. Dataset preparation before using it in the model was also explored, as initially the method followed did not give good results.

Conclusion : Among all the deep learning techniques trained in the project, pix2pix GAN was able to produce the image that closely resembled the original day image, thus making it easier to identify vehicles on the roads. However, the drawback for this method was time taken for implementation. Given a certain time limit, it was observed that classical techniques such as CLAHE performed better than the advanced Deep learning techniques, but given more time for training and required computational power, the deep learning methods can give excellent results.

Future work includes combining the classical approaches along with deep learning. One more aspect the team desires to look into, is trying out few combinations of the methods employed in the project to get the best results. Also, future work demands implementation of object detection for vehicles in order to observe, for which algorithm, the object detection(vehicle in this case) gives the most accurate result.

Acknowledgement

We would like to thank Prof. Amit Sethi, Dept. of Electrical Engineering, IIT Bombay for his keen efforts during teaching that helped throughout the project and also helped to avoid and correct several mistakes while implementing the project.

References

[1] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[2] U. Hari, A. R. Bevi, and B. Ramachandran, "Performance analysis of retinex based algorithms for enhancement of low light images," in *Journal of Physics: Conference Series*, vol. 1964, no. 6. IOP Publishing, 2021, p. 062046.

[3] Q. Chen, J. Xu, and V. Koltun, "Fast image processing with fully-convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2497–2506.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[5] G. Li, Y. Yang, X. Qu, D. Cao, and K. Li, "A deep learning based image enhancement approach for autonomous driving at night," *Knowledge-Based Systems*, vol. 213, p. 106617, 2021.

[6] N. Capece, U. Erra, and R. Scolamiero, "Converting night-time images to day-time images through a deep learning approach," in *2017 21st International Conference Information Visualisation (IV)*. IEEE, 2017, pp. 324–331.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition, presented at the 3rd international conference on learning representations (iclr 2015)," 2020.

[8] S. Sivagami, P. Chitra, G. S. R. Kailash, and S. Muralidharan, "Unet architecture based dental panoramic image segmentation," in *2020 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, 2020, pp. 187–191.

[9] ARCGIC Developers, "Theory of u-net," https://developers.arcgis.com/python/guide/how-unet-works/, accessed 29-10-2021.

[10] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.