An Extension and Correction to the Illuminative Adaptive Transformer

Prem Shanker Mohan
School of Computer Science
University of Windsor
Windsor, Canada
SID: 110036738

Hamza Baig
School of Computer Science
University of Windsor
Windsor, Canada
SID: 110089314

Vlad Tusinean
School of Computer Science
University of Windsor
Windsor, Canada
SID: 104823929

***Abstract*—The darkness of images can contain important information that is hard to use, so producing an effective method of enhancing the contrast of these images without harming the color is an important area of research. Here, we propose an extension of the Illuminative-Adaptive Transformer (IAT), which is a neural network specialized for enhancing low-light images. In the original paper, a fast and effective model was demonstrated. However, there are issues within the methodology that led us to question the reported performance of IAT. Specifically, we examine the code on their GitHub and provide some optimizations that provide complete clarity of the results reported. Additionally, we extend their pipeline to further improve their contrast enhancement results. The IAT model uses mean absolute error as a loss function and Adam optimization. We show that there are better options in the literature that lead to better enhancement performance. Depending on the intricacies in a given sample, our extended model produces a better enhancement than the ground-truth**

***Index Terms*—Image enhancement, low-light correction, computer vision, CLAHE, MADGRAD, IAT**

Acknowledgement

We would like to thank Dr. Kalyani Selverajah for all the guidance she provided us throughout the semester. Additionally, we give acknowledgment to the developers of PyTorch, the authors of Cui et. al, 2022, and Google Colab.

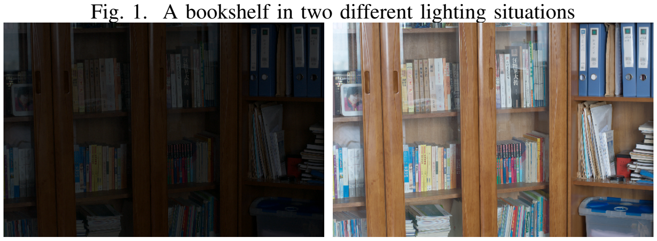
I. Introduction

Computer vision refers to a sub-domain of machine learning that operates on problems related to images. These problems can include generating, manipulating, or effectively utilizing images for some tasks. Typically, these images are standard RGB images, but computer vision can also be performed on grayscale images or other representations (such as HSV). Computer vision has been serving the world in a plethora of applications such as robotics, autonomous vehicles, smart farming, biomedical imaging, driver assistance systems, intelligent transportation, tracking of moving objects, etc. To solve problems related to images, computer vision implementations need data to work with, and this data needs to be manually procured and potentially labeled. Depending on the formulation of the problem, neural network models can learn to see what humans see, but they require a lot of input images to do so. Humans heavily rely on vision to function in the world around them, and consequently, that means we can have issues operating in the dark. Important information can be hidden behind the dark regions of an image, so there is interest in developing an effective method of brightening these dark regions without losing the important information that these regions are hiding. Modern-day cameras are equipped with sophisticated technology such as effective flashlight usage or powerful post-process algorithms to minimize the number of dark regions in an image, but more work needs to be done in this area. We also need to factor in the speed and size of any solutions - contrast enhancement algorithms need to be performant but also lightweight and fast to see real-life usage.

Computer vision is marred with issues that can complicate the data-collecting process, and these things need to be accounted for with any image enhancement algorithm. Photographs may not always be ideal, they may include issues such as lens glare, noise, blurriness, out-of-focus subjects, and darker sections that can render the image useless because some information will be difficult to understand. Generally, images plagued with these issues (especially blurry or low-light images) are immediately discarded, but what if these images contain vital importance and enhancement of them can make them useable? Computer vision has already served in a lot of image restoration tasks such as enhancing the overall quality of old pictures making them suitable for large-screen viewing by fixing the details that may be distorted as the image is zoomed in or the scale of the image is increased, fixing the color of old images, changing image color scale from grayscale to RGB, fixing blur or focus of the image in post-production through AI, etc. Also, an emergent topic in computer vision that has been gaining popularity (especially due to the fact of insufficient examples of data in benchmark datasets) is contrast enhancement in dark images.

Exposure is the setting of a camera lens that influences how much light can pass through the lens. If the image appears to be dark, it is considered to be underexposed. Conversely, if the image is too bright, that is an indication that the lens is causing overexposure. Dark images can be a consequence of poor lighting or related to the exposure setting of a camera, but it may not always be possible to tune these factors. As such, it may not be a simple task to extract important information from these images. For example, Fig. 1 shows a filled bookshelf in both low-light and high-light conditions. It is difficult to determine the titles of the books on the shelf in the low-light setting, while that is a very simple task in the high-light counterpart.

Fig. 1. A bookshelf in two different lighting situations



This work provides an extension to the paper *You Only Need 90K Parameters to Adapt Light: A Light-Weight Transformer for Image Enhancement and Exposure Correction* (Cui et. al, 2022). As such, a majority of the efforts presented here are based on the work done by (Cui et. al, 2022), and all extensions were based on the code they provided on Github.

As we will discuss later in later sections, we wanted to roughly preserve the utilized architecture because its benefits are that its light-weight and fast. Had we decided to overhaul the modeling approach, it is likely we could attain a more performant model. However, the increase in performance would result from a decrease in efficiency, which would mean we could not reasonably call our work an extension of theirs. Additionally, there are problems with their experimentation setup, which we rectified. This will also be outlined further in a later section. (Cui et. al, 2022) does not only deal with exposure correction: some of the main contributions of their work is a neural network that can perform low-light image enhancement (which will be the focus of this extension)

II. problem Statement

A. Problem Description and Formulation

We aim to extend the Illuminative Adaptive Transformer by (Cui et. al, 2022) in two ways. 1) Modify their code to correct all the errors with validation reporting and randomization. The end goal of this is to have certainty in all results produced by our models, and 2) To improve their results by applying a more suitable loss function and changing the optimization method. Additionally, we will apply CLAHE as a preprocessing step to improve the quality of the training samples that are passed to the model.

B. Motivation

The base paper had achieved decent results but the images still needed color correction and weren't as close to the ground truth as they should've been. The images looked dull in most cases, whites had a dull yellowish tint on them, and overall the model needed some improvements. In our extension to the base paper, we aimed at achieving images that almost resembled the ground truth, and the model wasn't to overfit too. The images that would result from the model, should have the correct colors and should match the ground truth with keeping the main objective in mind, which is to fix the exposure of the images. The sharpness of the image should be fixed as well as the base paper's output resulted in noisy images. The extension that we offer will improve the model's overall performance. The learning rate will be improved which means that the model learns more quicker. The loss is changed to MS-SSIM which compares the predicted image and the ground truth and calculates the loss, this loss will perform better as pixel-wise comparison of the images will adjust the weights better and once the model is trained completely the predicted images will be close to the ground truth, as it is learning from the comparison of the images.

The benefit of having these improvements is faster convergence, and better performance in terms of color accuracy and exposure fixes than the base model, which itself was performing decent, but with the improvements, the model performs and achieves excellent results.

III. Literature Review

Cui et. al provides a branched neural network utilizing a transformer for exposure correction; one branch - referred to as the local branch - maps the input image to a latent feature space and replaces the transformer's attention block with a depth-wise convolution for light-weight design. In the other branch - known as the global branch - the transformer's attention queries are used to control and adjust the global ISP-related parameters.

Of course, this is not the first work published in the domain of image enhancement relating to exposure correction. One such example is (Afifi et. al, 2022), which proposes a sequential method of correcting camera exposure-related issues. Specifically, they use a Laplacian pyramid as a multi-resolution decomposition, which is derived from the Gaussian pyramid of an 8-bit sRGB input image. The proposed network is a coarse-to-fine deep network that serves to progressively correct exposure errors in images. Each network in the sequence further corrects the exposure issues in an image. For example, the first network in the sequence corrects the global color captured at the final level of the Laplacian pyramid. The results shown in the research demonstrate the effectiveness of their approach, boasting compelling results compared to available solutions. There are a few disadvantages to this approach. While the proposed network is fully convolutional and generally functional across different resolutions, the computational power required to process high-resolution images might not be available. Specifically, regarding higher resolution images (such as 16-megapixel images) that were not involved in the training process, the model may not generalize well due to the larger homogeneous image regions. Regions that have insufficient semantic segmentation will produce unsatisfactory results.

Regarding low-light image enhancement, (Chen et. al, 2018) proposes an end-to-end fully-convolutional neural network. However, as most works on this topic focused on low-light enhancement in non-extreme settings, a custom dataset of raw short-exposure low-light images was created. Each image in this dataset has a corresponding long-exposure high-quality reference image, meaning the neural network can take the short-exposure image as a training sample and refer to the high-quality image as ground truth. The usual image processing pipeline involves applying sequence modules such as white balance, demosaicing, denoising, sharpening, etc. However, (Chen et. al, 2018) proposes a singular end-to-end neural network that can handle the entire image processing pipeline at once. However, rather than operating directly on the standard sRGB images like most image processing algorithms, this network operates directly on the raw sensor data. In terms of neural network architectures, two general structures are focused on: a multi-scale context aggregation network, and a U-Net. Since their method overhauls the traditional image processing pipeline, the performance of the network is compared directly to the performance of this traditional pipeline, where it was shown that this network outperforms BM3D and burst denoising.

There are many different ways to handle model optimization when training on data for exposure correction. One of the most important hyperparameters in this task is the choice of loss function, as it can completely dictate convergence and performance. The loss function used in (Cui et. al, 2022) was *Mean Absolute Error (MAE)* which was prioritized due to its ability to penalize raw pixel inaccuracies, leading to a network that could output corrections closer to the ground truth in RGB than other loss functions such as *Mean Squared Error (MSE)*. However, there do exist other loss functions for exposure correction that can optimize model performance without necessarily losing the effectiveness on the specific RGB values of the ground-truth output. *Loss Functions for Neural Networks for Image Processing* (Zhao et. al, 2015) is a survey on the different loss functions that are beneficial in the domain of image correction. Specifically, they review several loss functions that can be reasonably applied in this domain and evaluate their performance via demosaicing, denoising, deblocking, and super-resolution on the MIT-Adobe FiveK dataset. Their results show that the Multi-Scale Structural Similarity Index Measure (MS-SSIM) with l1 error as an additional penalty term is the most performant loss function for image correction. Interestingly enough, this paper does not review using MAE as a loss function for image correction, which is what is used by (Cui et. al, 2022).

Similarly to the loss function, another important hyperparameter is the choice of optimizer. In (Cui et. al, 2022), the Adam optimizer was used. Adam has been proven to be an optimal optimization algorithm time and time again and is a general choice in most modern-day neural networking approaches, However, that does not mean Adam has no shortcomings. (Defazio & Jelassi, 2021) discusses the many issues of Adam such as its potential to converge to a bad local minimum on important problems like image classification, problems that can be constructed where ADAM fails, Adam isn't well-suited to sparse problems, etc. Thus, a Momentized, Adaptive, Dual-averaged GRADient (MADGRAD) method is proposed specifically to address the shortcomings of Adam. MADGRAD will be further discussed in a later section

Of course, low-light image quality enhancement is not only possible via the application of a neural network. Many algorithmic approaches for boosting the contrast for low-light images, but see limited use due to generalization issues as well as undesirable performance. However, it is not a fixed requirement that one method is used over the other. Rather, we can use these methods in tandem to produce better overall results.

Contrast Limited Adaptive Histogram Equalization (CLAHE) is an algorithmic approach to image quality enhancement introduced by (Yadav et. al, 2014) based on Adaptive Histogram Equalization (AHE). Histogram equalization works to globally improve the contrast of an image by separately normalizing the three channels of an RGB image. In the adaptive version of histogram equalization, contrast boosting via normalization is applied to specific regions of the image, and the contrast is boosted based on neighboring regions. CLAHE, however, applies this enhancement function over all neighborhood pixels, and a transformation function is derived, a gray-level mapping is generated, and finally, an interpolation over the grey-level map is performed to produce the final output image.

CLAHE as discussed in (Yadav et. al, 2014) was applied frame-by-frame on a video of a foggy scene, but other works extended CLAHE to focus on low-light image enhancement. (Manju et, al, 2019) repurposes CLAHE for low-light enhancement by using an illumination reflection model. Reconstruction of an image uses CLAHE and morphologically processing with a top-hat transformation. Specifically, this method works with images in HSV rather than RGB, and CLAHE is applied to the inverse of the normalized intensity component V. Afterwards, multiscale image enhancement gamma enhancement, and principal component analysis are used to produce the final output image.

IV. Methodology

A. Material and Data

The dataset used by (Cui et. al, 2022) for low-light image enhancement is the **LO**w-**L**ight (**LOL**) dataset created by (Wei et. al, 2018). This dataset consists of 500 images of low- and high-light background scenes. Here, the low-light images are used as training samples and the high-light images are used as ground truth. This work was done with the following libraries:

- pytorch 1.10.1+cuda113
- scikit-learn 1.1.3
- numpy 1.23.4
- opencv-python 4.6.0.66
- madgrad 1.2
- matplotlib 3.6.2

B. Proposed Method

To improve the low-light image enhancement section of Cui et al's work, the proposed steps recommend addressing the current drawbacks of the model, introducing more randomness into the process, merging the training and validation sets, using a cross-validation scheme, and considering alternative loss functions and optimization algorithms. The use of the CLAHE technique is also proposed as a way to enhance the model's performance.

Mean Absolute Error as loss

Mean Absolute Error (MAE) is a widely used regression metric that measures the average absolute difference between predicted and actual values. It is a loss function that is used in regression problems and is calculated as the average of the absolute differences between the predicted values and the actual or ground-truth values.

One of the key advantages of using MAE as a loss function is that it is easy to interpret. Since the error is calculated in the same units as the original data, the error can be easily understood in the context of the problem.

Despite its advantages, MAE is not without its drawbacks. One of the main limitations of MAE is that it treats all errors equally, regardless of their magnitude. This means that large errors will have the same impact on the MAE as small errors.

We discuss alternatives used further in the research paper.

SSIM and MS-SSIM

A method for forecasting the perceived quality of digital television and film images, as well as other types of digital images and videos, is the structural similarity index measure (SSIM). SSIM is a tool for calculating how similar two photos are to one another. The SSIM index is a full reference metric, meaning that the initial uncompressed or distortion-free image serves as the baseline for the measurement or prediction of image quality.

The SSIM is a perception-based model that incorporates crucial perceptual phenomena, such as luminance and contrast masking terms, and views image degradation as a perceived change in structural information. The distinction between these methods and others is that they estimate absolute errors, unlike MSE or PSNR. The concept of structural information holds that pixels have high interdependencies, particularly when they are spatially close to one another. Important details regarding the organization of the items in the visual scene are carried by these dependencies. Contrast masking is a phenomenon where distortions become less evident where there is high activity or ”texture” in the image, whereas luminance masking is a phenomenon where visual distortions (in this context) tend to be less visible in bright places.

By including image information at various resolution scores, the Multi-scale Structural Similarity Index Measure generalizes the Structural Similarity Index Measure. Similar to SSIM, the MS-SSIM technique starts by scaling the two images to the same size and resolution, normalizing them for brightness and contrast, and then analyzing them with a sliding window. The same SSIM comparison criteria—change in brightness, change in contrast, and correlation—is used to compare the images (or structure). The photos are down-scaled, the comparisons are repeated, and the scores for each comparison are tallied.

For most implementations, between three and five iterations are used, and this procedure continues for that number of iterations. Each of the several image scales produces comparison results, thus the findings must be merged to provide an overall score for the image. Different weightings are applied to the results at each image scale since the human perception of the impact of distortions varies with the image scale. These weights were created by tests involving several image sets and human observers who were instructed to find images with the same level of perceptible distortion across all scales. One intriguing finding of this research was that while the contrast and structure comparisons are aggregated at each picture size, the luminance comparison only needs to be performed at the smallest scale (most strongly downsampled).

The outcomes must be better to justify the added workload MS-SSIM places on an analysis system. Indeed, this is the case, as evidenced by several studies that are discussed in more detail in the section below. It's noteworthy to note that the advancement of MS-SSIM over SSIM is comparable to the advancement made by SSIM over PSNR. The added expense of employing MS-SSIM is negligible in comparison to the value of the higher precision that can be obtained by applying the algorithm given the processing capacity that is now available in measuring equipment.

MADGRAD

Optimization for deep learning is a relatively new area of study within the broader field of optimization. Deep learning presents unique challenges that require new tools and approaches to overcome. One of the main challenges is the large size of the parameter vectors used in deep learning models, which can make it impractical or even impossible to store and manipulate matrices of the same size. This problem is further compounded by the increasing size of deep learning models, which can have billions of parameters. As a result, the practical limits of optimization for deep learning are often determined by the available storage space, which is typically fixed at a small multiple of the parameter vector size.

Because of the challenges associated with large parameter vectors in deep learning, diagonal scaling methods have become the industry standard. These methods allow for adaptivity on a coordinate-by-coordinate basis, which makes them more efficient in terms of memory usage compared to other methods. Adam, introduced by Kingma and Ba in 2014, is the most widely used method in this class and is considered the benchmark. Currently, there are no alternative adaptive methods that consistently outperform Adam.

Adam is a type of diagonal adaptive method that builds on previous work in the field, such as AdaGrad and RMSProp. AdaGrad was the first method to introduce a principled approach to diagonal adaptivity, and it offers natural convergence rate bounds for convex losses. RMSProp, on the other hand, was developed as an empirical method that performed well in practice, but without a strong theoretical foundation. Adam extends the scaling used in RMSProp to include a form of momentum, as well as a bias correction that helps stabilize the adaptivity and step size during the early stages of optimization.

Together, these features make Adam a powerful and effective optimization method for deep learning.

Despite its success in many applications, Adam is not without its limitations. In particular, (Defazio & Jelassi, 2021) showed that Adam and other common adaptive optimizers could converge to bad local minima on certain problems, such as image classification. This has led some to claim that adaptive methods generally do not perform well in terms of generalization. However, this is not necessarily true.

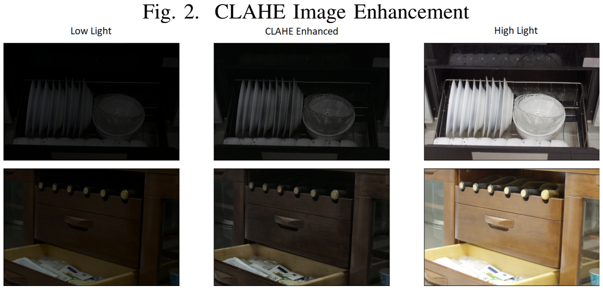
MADGRAD (Momentumized, Adaptive, Dual averaged GRADient) is a new method that combines adaptivity with strong generalization performance. MADGRAD consistently achieves state-of-the-art performance across a variety of large-scale deep-learning problems and does not require any additional tuning beyond what is needed for Adam. MADGRAD is based on the dual averaging form of AdaGrad and is modified through a series of direct and systematic changes that make it well-suited for deep learning optimization.

CLAHE

As covered in the literature review section of this paper, CLAHE stands for Contrast Limited Adaptive Histogram Equalization. It is a technique for enhancing the contrast in images, particularly for those with low contrast. This is achieved by stretching the contrast of the image so that the intensity values are spread out over the full range of available values, making the image easier to see and interpret.

Some examples of the effects of CLAHE are shown below:

Fig. 2. CLAHE Image Enhancement



C. Drawbacks of base paper

One of the main benefits of (Cui et. al, 2022) is that they provided a GitHub repository with all of their code, so we were able to examine their experimentation and understand the specific details of their modeling approach. However, upon further investigation into their methodology, we found numerous issues with how these tests were performed, which led us to question the results they outlined in the paper. There is no way to confidently prove that the issues we discovered were present when they reported their results, but there is no way to disprove their existence either. Thus, we assume that whatever functionality is present or missing in their GitHub repository influenced their reporting.

- Fixed random seed
  At the beginning of the section of code responsible for data loading and data augmentation, a random seed is fixed to 1143. This means that no matter what test is performed, the data that is being passed to the model will always be the same. In other words, data augmentation is essentially useless, and there is bias present in their results. This was likely done in order to ensure consistency between model evaluations if they were testing specific hyperparameters, but the consequence of this fixed seed is that we cannot conclude their model would generalize to unseen data.
- Fixed validation data
  The LOL dataset (Wei et al, 2018) is one of the datasets used by (Cui et. al, 2022) to examine the performance of their network on the task of low-light image enhancement, which is the specific work we are providing an extension for. This dataset consists of 500 images of random scenes, each scene having a high-light and a low-light counterpart. Some images are duplicated, but have had their contrast adjusted so the network could be undeterred by differing contrast levels
  However, the dataset is pre-partitioned into 485 samples for training, and 15 samples for validation. There is no testing dataset. The primary issue with this is that the validation dataset is fixed for every single network that is trained, meaning there is an extremely high likelihood of overfitting to their validation data.
- Lack of cross-validation
  Perhaps one of the most important parts of the experimentation that was overlooked by Cui et. al is cross-validation. Model weight initialization is randomized (unless you fix the random seed), so the results of one specific model cannot be trusted - there is the possibility that this specific network had a good initialization that is difficult to replicate, which could harm your generalizability and invalidate your results. One of the best ways to guard against this possibility is to use cross-validation, which involves generating several models and shuffling your data to ensure that the models are looking at different partitions of the training data. This also means that the validation dataset is expected to shift with every iteration of the cross-validation. The lack of cross-validation in this work means that there is a possibility that (Cui et. al, 2022) repeated the training process until they trained a model that performed well specifically on their validation data. Since their validation data is fixed rather than randomly partitioned from the overall data as outlined in the above point, this is very likely.

Taking these three points into consideration, we are confident that the results displayed by Cui et. al in their original paper cannot be trusted. Of course, we cannot prove beyond a shadow of a doubt that the original paper was published with these flaws, but enough information was given via their GitHub repository to have us concerned. For example, it is more than likely that the fixed random seed did not exist when the original paper was published (though we can't prove that), and there is no indication of a cross-validation scheme anywhere in the repository. It does not make sense that they would use a cross-validation scheme and delete that code before uploading it to GitHub.

D. Improvements over base paper

Thus, one of the main extensions we will provide to this work is an overhauling of their code to ensure these flaws are removed and the results of Cui et. al's work can be trusted, We have no doubt that their network is incredibly performant and as fast as was stated, and we will ensure there is accountability in statistical reporting such that there is no concern going forward.

We begin by removing all traces of random seed fixing throughout the various files in the repository. This way, all model training is not consistent and we can get a sense of model generalizability. Next, we merge the validation and training datasets, which will allow us to dynamically split the dataset into unique partitions of training and validation sets for every training session, further giving us a sense of model overfitting.

Finally, we implement a K-Fold cross-validation scheme, using a K value of 5. In K-Fold cross-validation, the initial dataset is split into K partitions. From there, the first partition is chosen as the validation set, and the remaining partitions are used as the training set. Then, a model is initialized with random weights and the training process begins. Once training is complete with this specific training and validation set, the process is restarted. Next, partition 2 is used as validation, and the training set is generated in a similar manner (all partitions but partition 2). K-Fold cross-validation is complete once all partitions have been used as the validation set exactly once. There is an extension to this process known as nested cross-validation, which does the normal K-Fold cross-validation split but has an additional partition reserved for testing. In this case, the testing partition is fixed for an iteration of K, and the cross-validation partitioning for the training and validation sets is performed on the remaining partitions. As an example, considering a value of 5 for K, this means that we would evaluate 5 disjoint testing partitions and 25 validation partitions.

These improvements relate to fixing the flaws present in Cui et. al's work. However, we also plan on adding more. For example, we will be testing different loss functions to improve their results if possible, we'll use a different optimizer, and we'll be including CLAHE image enhancement as part of the preprocessing to improve the overall performance of any model we train.

V. Experiments

The main contribution of this work (aside from the more robust validation metric tracking) is the development of a more efficient method for processing low-light images, which results in improved performance of the model. This is achieved by upgrading the optimizers and loss functions used during training.

The use of more advanced hyperparameters allows the model to better capture the patterns and features present in low-light images, leading to improved prediction accuracy. In this way, our approach builds on the model presented in the base paper but improves its performance by using more sophisticated techniques for processing the data.

By preserving the original model, dataset, and image augmentation from the base paper, this work allows for a direct comparison of the performance of the model before and after the improvements were made. This enables the significance of the contribution to be demonstrated, as the improvements can be quantitatively measured and compared. This also allows for a fair comparison with the results presented in the base paper, ensuring that any differences in our performance are due solely to the improvements made and not by any other factors.

A. Trials

In this study, the effects of various optimizers, loss functions, and image augmentations are explored by conducting several trials. The base paper uses mean absolute error with the Adam optimizer. No changes are made to the model or the preprocessing of images within the model. In total, three loss functions (MAE, MSSIM, and MSSIM with L1 loss) and two optimization methods (Adam and MADGRAD) are compared. We expect CLAHE preprocessing to positively impact model performance, so CLAHE preprocessing will be used on every trial except for the re-evaluation of the base model from (Cui et. al, 2022).

There are 7 models we are comparing:

- MAE with Adam
- MAE with Adam, CLAHE augmentation
- MAE with MADGRAD, CLAHE augmentation
- MS-SSIM with Adam, CLAHE augmentation
- MS-SSIM with MADGRAD, CLAHE augmentation
- MS-SSIM + L1 with Adam, CLAHE augmentation
- MS-SSIM + L1 with MADGRAD, CLAHE augmentation

B. Evaluation metrics

To evaluate our model, we use SSIM and PSNR.

SSIM (Structural Similarity Index) is a metric that measures the similarity between two images as described previously.

PSNR (Peak Signal-to-Noise Ratio) is a metric that measures the quality of a reconstructed image compared to the original image. It is commonly used in the field of image and video processing.

PSNR is calculated by first converting the images to grayscale and then taking the square of the difference between the original image and the reconstructed image, for each corresponding pixel. The mean squared error (MSE) is then calculated by taking the average of these squared differences across all pixels in the images. The PSNR is then calculated by taking the ratio of the maximum possible pixel value to the MSE and expressing this ratio in decibels (dB).

The higher the PSNR, the better the quality of the reconstructed image. A PSNR of 30 dB or higher is generally considered to be acceptable for most applications, although higher values may be required for more demanding applications.

One limitation of PSNR is that it only considers the difference in pixel values between the original and reconstructed images, and does not take into account other factors that can affect the perceived quality of an image, such as color, contrast, and spatial resolution. This means that two images with the same PSNR value may still look different to the human eye.

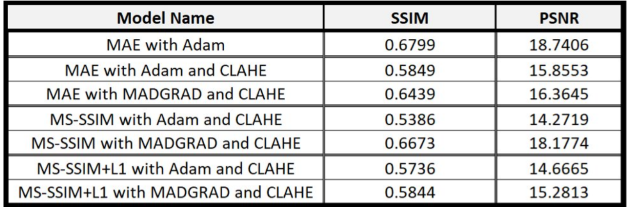
In summary, PSNR is a metric that measures the quality of a reconstructed image compared to the original image. It is calculated by taking the ratio of the maximum possible pixel value to the mean squared error between the images and expressing this ratio in decibels. A higher PSNR indicates a better-quality reconstructed image.

It must be stated that SSIM as an evaluation metric is somewhat flawed. The task here is low-light image enhancement rather than image recreation, and as such our models have the possibility of producing an output that is better than the expected ground truth for a sample. Since the model's output, in this case, is different from the ground truth, the SSIM score will inherently be lower than other, less performant networks. However, we wanted to discuss our work as an extension of Cui et. al's IAT model, so we kept the same evaluation metrics. Additionally, this means that using SSIM/MS-SSIM as a loss function is not the most optimal, and more investigation needs to be done into what loss functions are optimal for this problem. This investigation will be future work.

C. Results and Analysis

The model's outputs are of higher quality than the base model (Fig. 3), despite not having the highest SSMI and PSNR values. This is because the metrics used do not accurately reflect the quality of the enhancement, and the model sometimes enhances images better than the ground truth, which affects the SSMI and PSNR values. Additionally, some models produce artifacts that negatively impact their performance.

Fig. 3. Comparison of the base model with our models



Model Name	SSIM	PSNR
MAE with Adam	0.6799	18.7406
MAE with Adam and CLAHE	0.5849	15.8553
MAE with MADGRAD and CLAHE	0.6439	16.3645
MS-SSIM with Adam and CLAHE	0.5386	14.2719
MS-SSIM with MADGRAD and CLAHE	0.6673	18.1774
MS-SSIM+L1 with Adam and CLAHE	0.5736	14.6665
MS-SSIM+L1 with MADGRAD and CLAHE	0.5844	15.2813

Fig. 4 describes the performance of a base model using K-fold cross-validation. The graph shows that the model is inconsistent on validation, which indicates that the original model is heavily overfit. One reason for this issue is that the base paper did not have an effective test set, and instead only tuned the model to perform well on selected images. As a consequence of this, the hyperparameters that were chosen by (Cui et. al, 2022) (which we used) likely did not regularize the model enough.

Fig. 5 describes the SSIM and PSNR values for the base paper's model.

Fig. 4. MAE Loss with Adam Optimizer

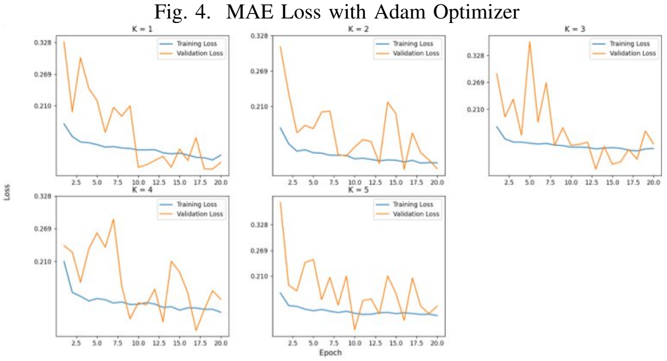


Fig. 5. SSIM and PSNR for MAE with Adam Optimizer

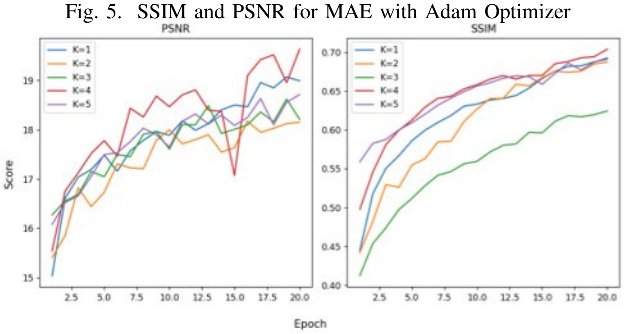
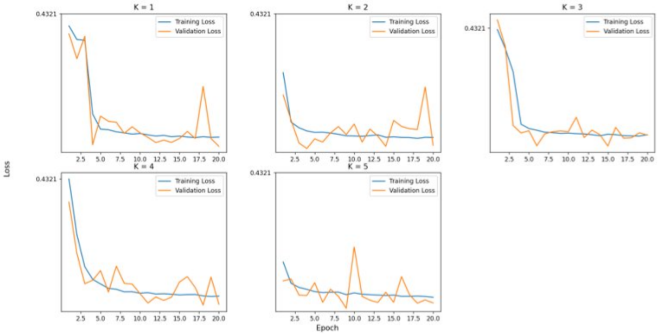


Fig. 6. Loss for MS-SSIM+L1 with MADGRAD and CLAHE



The model that used MS-SSIM + L1 as the loss function with MADGRAD optimization achieved the best results in terms of visual quality. Despite not having the highest SSIM and PSNR scores, it had a consistently smooth validation loss. However, some images still showed artifacts that hindered the model's performance.

When comparing the base model to our best-performing model (Fig. 8), we can see that our model enhances the images more effectively than the base model. Additionally, our model

Fig. 7. SSIM and PSNR for MS-SSIM+L1 with MADGRAD and CLAHE

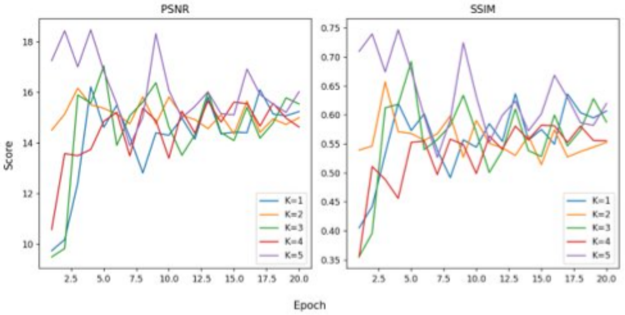
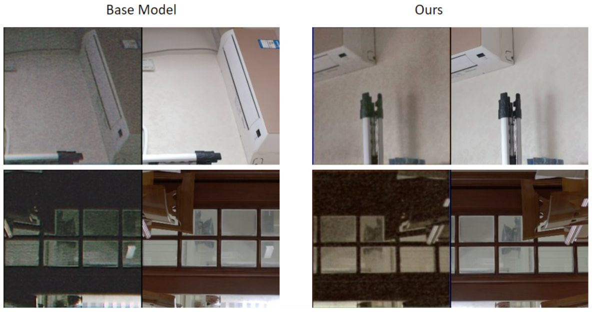


Fig. 8. Visual Comparison of the base model with our model



produces less visual noise in the images than in the base model.

VI. Conclusion

A. Summary

In this study, we extended the works of (Cui et. al, 2022) and improved their model. First, we identified the problems or shortcomings of the methodology proposed by the base paper, and then we researched for a solution. In our extensive testing of the base paper's model, we discovered that the optimizer can be changed to MADGRAD for better convergence and the loss function can be improved to MS-SSIM which would improve the performance of the model. In our tests, the extended model we proposed performs better than the base paper's model, not only in terms of analytics but also in terms of the visual quality of the enhanced image. The significant changes we made to the base model were:

1) Change the optimizer from Adam to MADGRAD which made the model converge faster and generalize better.
2) Updated the loss from Mean Absolute Error to MS-SSIM+L1 which compares the images on pixel level and gives better loss for the problem we were facing.

B. Future Research

Despite the results obtained from the extended model, which were great, some things still need improvements. The problem of having a transformer enhance low-light images is not easy, at times the model struggled with images with a broader spectrum of colors and images where edges were not refined. Though the model still managed to perform much better at later epochs, the output still could need some improvements.

A better dataset could be used for testing the resulting model. A proper divide between the training and validation dataset would give trustworthy analytics regarding the performance of the model.

Additionally, we can overhaul their data augmentation approaches to further improve the generalizability of the model. (Cui et. al, 2022) employs a combination of contrast manipulation and image flipping, rotating, cropping, and shifting. However, other image augmentation techniques exist in the literature that can be used.

C. Open Problems

Although the extended model has achieved great results, it is still to be put to test against other state-of-the-art techniques for a better understanding of the performance of the model. The extended model outperforms the base model, but we would have to run multiple state-of-the-art techniques on the same dataset to see how the model holds against these techniques.

References

[1] Cui, Ziteng, Li, Kunchang, Gu, Lin, Su, Shenghan, Gao, Peng, Jiang, Zhengkai, Qiao, Yu, & Harada, Tatsuya. (2022c). You Only Need 90K Parameters to Adapt Light: A Light Weight Transformer for Image Enhancement and Exposure Correction. Cornell University - ArXiv. https://doi.org/10.48550/arxiv.2205.14871

[2] Afifi, M., Derpanis, K. G., Ommer, B., & Brown, M. S. (2021). Learning Multi-Scale Photo Exposure Correction. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr46437.2021.00904

[3] Chen, C., Chen, Q., Xu, J., & Koltun, V. (2018, May 4). Learning to see in the dark. https://arxiv.org/abs/1805.01934 https://doi.org/10.1109/icacci.2014.6968381

[4] Zhao, H., Gallo, O., Frosio, I., & Kautz, J. (2017). Loss Functions for Image Restoration With Neural Networks. IEEE Transactions on Computational Imaging, 3(1), 47–57. https://doi.org/10.1109/tci.2016.2644865

[5] Yadav, G., Maheshwari, S., & Agarwal, A. (2014). Contrast limited adaptive histogram equalization-based enhancement for real-time video system. 2014 International Conference on Advances in Computing, Communications, and Informatics (ICACCI).

[6] Manju, R., Koshy, G., & Simon, P. (2019). Improved Method for Enhancing Dark Images based on CLAHE and Morphological Reconstruction. Procedia Computer Science, 165, 391–398. https://doi.org/10.1016/j.procs.2020.01.033

[7] Defazio, A., & Jelassi, S. (2021). Adaptivity without Compromise: A Momentumized, Adaptive, Dual Averaged Gradient Method for Stochastic Optimization. ArXiv: Learning.