Low-Light Homomorphic Filtering Network for Integrating Image Enhancement and Classification

Rayan Al Sobbahi and Joe Tekli*

Department of Electrical and Computer Engineering,
Lebanese American University (LAU), 36 Byblos, Lebanon

rayan.alsobbahi@lau.edu; joe.tekli@lau.edu.lb

***Abstract*—** Low-light image (LLI) enhancement techniques have recently demonstrated remarkable progress especially with the use of deep learning approaches. However, most existing techniques are developed as standalone solutions and do not take into account the impact of LLI enhancement on high-level computer vision tasks like object classification. In this paper, we propose a new LLI enhancement model titled LLHFNet (Low-light Homomorphic Filtering Network) which performs image-to-frequency filter learning and is designed for seamless integration into classification models. Through this integration, the classification model is embedded with an internal enhancement capability and is jointly trained to optimize both image enhancement and classification performance. We have conducted a large battery of experiments using SICE, Pascal VOC and ExDark datasets, to quantitatively and qualitatively evaluate our approach's enhancement quality and classification performance. When evaluated as a standalone enhancement model, our solution consistently ranks among the best existing image enhancement techniques. When embedded with a classification model, our solution achieves an average 5.5% improvement in classification accuracy, compared with the traditional pipeline of separate enhancement followed by classification. Results produce robust classification quality on both LLIs and normal-light images (NLIs), and highlight a clear improvement to the literature.

***Index Terms*—**Image enhancement, low-light conditions, deep learning, object classification, homomorphic filtering.

1. Introduction

Modern artificial intelligence-based applications like autonomous spacecrafts, drones, autopilot car systems, robots, and security surveillance systems, among others, essentially rely on visualizing and understanding outdoor environments. Such systems use cameras as their vision sensors to perform high-level computer vision tasks like classification, detection, semantic segmentation, and tracking. While these systems show good performance during normal and clear outdoor conditions, yet varying weather conditions and poor illumination might challenge their visual perception and compromise their performance [1] [2]. More specifically, low-light conditions account to a considerable time of our daily life and can significantly affect the robustness of such systems and hinder their market deployment [1]. Therefore, low-light image (LLI) enhancement has emerged: i) as a standalone image processing task that aims at illuminating LLIs and improving their visual quality, and ii) as a pre-processing step embedded with another high-level computer vision task (e.g., object classification) to improve its performance.

LLI enhancement techniques have been largely investigated in recent years. Many traditional approaches use gamma correction methods [3] [4], some rely on histogram equalization methods like CLAHE [5], DHE [6], while others follow the Retinex theory [7] like Multi-scale Retinex (MSR) [8], SRIE [9] and LIME [10]. More recently, Deep Learning (DL) techniques have demonstrated better performance and efficiency compared with traditional methods [11] [12]. Most DL solutions like DeepUPE [13], RetinexNet [14], and MBLLEN [15], follow a supervised training setting which requires training datasets of paired LLIs and their corresponding NLI counterparts. Other approaches like EnlightenGAN [16] use an unsupervised training setting, while recent solutions like ZeroDCE [17] are zero-reference models which do not require any paired or unpaired training data. However, most of these models are designed to perform LLI enhancement without considering the target high-level computer vision task. One major question is whether a LLI enhancement method – which performs well as a standalone component – can improve (or not) the performance of the high-level computer vision task as a whole. According to a recent study in [1], LLI enhancement may loosen the original image's discriminative semantic features, thus deteriorating classification and detection performance. Another empirical evaluation in [18] shows that good enhancement quality does not necessarily correlate with good object detection and classification quality, attributing this to the fact that most existing LLI enhancement models are designed as standalone solutions.

In this paper, we introduce a novel LLI enhancement solution designed for the object classification task. Our solution consists of two contributions: i) introducing a novel *LLI enhancement* model titled LLHFNet (Low-light Homomorphic Filtering Network) based on image-to-frequency filter learning, and ii) introducing a *LLI enhancer-classifier* model, which integrates the enhancement model into a state-of-the-art object classification solution. On the one hand, LLHFNet performs image-to-frequency filter learning, inspired from homomorphic filtering traditionally used for LLI enhancement. On the other hand, the LLI enhancer-classifier model

* Corresponding author. Joe Tekli is also an adjunct researcher with the SPIDER research team, LIUPPA Laboratory, University of Pay and Pays Adour (UPPA), 64600, Anglet, Aquitaine, France.

integrates LLHFNet into a typical classification model, namely ResNet50 [19], to perform a joint training that optimizes both enhancement and classification performance simultaneously. Note that our solution is not tied to ResNet50, and is designed to be coupled with typical feature extractors utilized with existing classification models including VGG16 [20], MobileNetv2 [21], and SqueezeNet [22], among others, thus making it unconstrained from any special architecture.

We perform a large battery of experiments to evaluate the performance of our approach. On the one hand, quantitative and qualitative evaluations on LLHFNet show competitive results compared with state of the art enhancement models like ZeroDCE [17], EnlightenGAN [16], and DeepUPE [13]. On the other hand, we compare our LLI enhancer-classifier model against the traditional pipeline consisting of separate enhancement followed by classification. Results show a robust classification against both LLIs and NLIs, producing an average 5.5% improvement in classification accuracy on synthetic LLIs form the Pascal VOC 2007 dataset [23] and real-world images from the ExDark dataset [24], highlighting a clear improvement to the literature.

The remainder of this paper is organized as follows: Section 2 provides a brief review of the related works, Section 3 explains preliminaries on homomorphic filtering, Section 4 describes our LLI enhancement model design, Section 5 describes our LLI enhancer-classifier model design, Section 6 describes our experimental evaluation and results, before concluding in Section 7 with future works.

2. Related Works

2.1. Traditional Methods

Traditional LLI enhancement techniques rely on mathematical or algorithmic models to perform the enhancement task. For instance, gamma correction methods, e.g., [3] [4], use a nonlinear transformation-based function in which a gamma correction parameter is adjusted to stretch or compress different gray regions of the image, aiming to enhance it. Also, histogram equalization methods, e.g., [6] [5] [25], rely on a cumulative distribution function to change the image output gray levels such that they fit into a uniform distribution. The original LLI is mapped to its enhanced counterpart with an approximately uniform gray-level distribution. Yet the latter methods generally ignore spatially varying lightness and result in under or over brightened regions. In addition, Retinex theory –i.e., the theory of the human retinal cortex [7] has been utilized to perform LLI enhancement. According to the theory, an image comprises two components: i) reflectance which is considered to be constant under varying light conditions and holds the inherent characteristics of visual objects, and ii) illumination which represents the varying lighting conditions of the image. Hence, the Retinex model is used to estimate the illumination component of the image and retain its reflectance component, preserving the image's inherent features to allow more accurate image processing. Yet most Retinex-based approaches like Single Scale Retinex (SSR) [26], MultiScale Retinex (MSR) [8], MultiScale Retinex with Color Restoration (MSRCR) [27], LIME [10] and SRIE [9] assume that enhancement does not affect image reflectance, regardless of the color distortions or lost details that result from applying the Retinex model [28]. In addition, Retinex-based enhancement quality is dependent on a set of carefully hand-crafted parameters allowing to estimate the resulting illumination map [14] . More recent techniques rely on homomorphic filtering (HF) to perform enhancement. In [29], the authors propose an HF algorithm to improve image brightness and avoid the edge blocking effect. They integrate a denoising approach based on guided image filtering to eliminate the amplified noise. The authors in [30] design a two-channel HF image enhancement method, where they first convert the input image from RGB to HSV color domain, and then perform enhancement separately on the saturation channel (S) and on the illumination channel (V) using Butterworth HF and Gaussian HF respectively. In [31], the authors combine HF with a parametric fuzzy transform. They first use an HF algorithm to acquire the exact illumination image of the V channel in the HSV domain, and then perform fuzzy image processing through a parametric transform to smooth and enhance the image's illumination.

2.2. Deep learning Approaches

In contrast to the traditional algorithmic or mathematical enhancement approaches, Deep Learning (DL) techniques are essentially data-driven, where training datasets of LLIs and NLIs are used to drive the learning process. They have gained great attention in the past few years as the most effective solutions to perform LLI enhancement, outperforming many traditional methods based on histogram equalization, e.g., [6] [5] and Retinex theory, e.g., [26] [27] [8]. LLNet [11] is one of the first DL approaches for LLI enhancement. Its architecture is based on a stacked-sparse denoising autoencoder (SSDA) made of three denoising autoencoder layers comprising hidden units with no use of convolutional layers. LLCNN [32] is proposed as a Convolutional Neural Network (CNN) based model for LLI enhancement. It is built using specially designed inception and residual modules, convolving an input using different size convolutional layers and then combining their outputs to the next layer. LightenNet [12] is another CNN-based model which estimates the Retinex-based illumination component from the original LLI and then uses it to produce the enhanced image. The model architecture consists of 4 convolutional layers used for: i) patch extraction and representation, ii) feature enhancement, iii) non-linear mapping, and iv) reconstruction. In [14] authors introduce RetinexNet consisting of two subnetworks: i) DecomNet which learns the Retinex decomposition of the image into its reflectance and illumination components, and ii) EnhanceNet which uses a dedicated encoder-decoder structure to perform illumination adjustment and enhancement. The model is embedded with a denoising operation using the BM3D [33] denoising algorithm. Wang et al. introduce a Global Illumination Aware and Detail-preserving NETwork (GLADNET) [34] made of a global illumination estimation step that uses an encoder-decoder structure followed by a reconstruction step through a series of convolutional layers. In [15] authors propose MBLLEN, a Multi-Branch Low-Light Enhancement Network which extracts the LLI features at each of its 10 convolutional layers through a special feature extraction module, and then enhances the features at each layer using an encoder-decoder network. It then fuses the multi-branch enhanced features into a final enhanced image. Wang et al in [13] propose a Deep Underexposed Photo Enhancement (DeepUPE) model which performs an image-to-illumination map learning. It consists of an encoder network based on a pre-trained VGG16 model [20] which extracts the local and global features of the input image. Then, a bilateral grid based up-sampling is added to produce the image's full resolution illumination map, which in turn is used to enhance the image based on the Retinex model. In [28] authors describe Retinex Decomposition based Generative Adversarial Network (RDGAN) which consists of two subnetworks: i) Retinex Decomposition Net (RDNet) that decomposes the LLI into its illumination and reflectance components, and ii) Fusion Enhancement Net (FENet) that fuses the decomposed parts into an enhanced image.

In a recent approach, the authors in [16] introduce EnlightenGAN, an unsupervised GAN approach which achieves state-of-the-art enhancement performance and successfully generalizes to real-world scenes. The model uses an attention guided U-Net [35] as its generator backbone, in addition to a global relativistic discriminator [36], and a local discriminator to handle spatially varying light conditions in the image. It is driven by non-reference loss functions combining local and global discriminator adversarial losses and a self-feature preserving loss. A more recent approach in [17] proposes a Zero Reference Deep Curve Estimation (ZeroDCE) model which does not require any paired or unpaired training data (hence the name "Zero Reference"). The authors entirely reformulate the LLI enhancement task: from image-to-image learning into image-to-light curve learning. The light enhancement curves are estimated using a lightweight deep curve estimation network (DCE-Net), and are iteratively applied on the input LLI to produce the final enhanced image. The model is optimized by non-reference loss functions including spatial consistency, exposure control, color constancy, and illumination smoothness losses.

2.3. Discussion

While many of the traditional and DL-based models show success when utilized as standalone LLI enhancement solutions, yet they share a common limitation: they are not tailored for high-level computer vision tasks like object classification. The models are designed solely to perform enhancement without considering the impact of this enhancement when used as a preprocessing stage for object classification. A recent study in [18] performs an evaluation for the performance of state of art classification and detection models on LLI datasets preprocessed by recent enhancement models. Results show that DL-based LLI enhancement solutions add slight (or no) improvement to detection and classification performance. The authors conclude that a good enhancement solution does not necessarily produce improved detection and classification quality since existing LLI enhancement models are designed as standalone solutions.

In this paper, we introduce a novel LLI enhancement solution designed to be integrated and embedded with the object classification task, in order to make it more robust against low-light and normal-light conditions. Our model builds on homomorphic filtering (HF), and devises a special filter to transform the image frequency components in the Fourier transform domain. It then estimates the filter parameters using DL-based feature extractors used for object classification. We first provide preliminary notions on HF, and then describe our DL-based enhancement model.

3. Preliminaries on Homomorphic Filtering

LLI enhancement models based on homomorphic filtering (HF) adopt the Retinex model representation of an image as a combination of illumination and reflective components (cf. Section 2.1). HF aims at converting the illumination and reflectance components which combine multiplicatively, into an additive form in the logarithmic domain [37]. The additive components are separated linearly in the Fourier transform frequency domain in which high frequency components are associated with reflectance while low frequency components correspond to illumination. A high-pass filter is used to suppress low frequencies and amplify high frequencies [37]. Figure 1 depicts the flow of the HF algorithm adopted in our approach. We describe its main steps below.

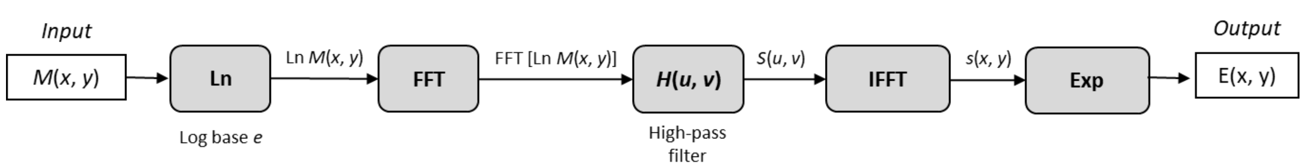


Figure 1. HF algorithm flow (adapted based on [37])

Step 0. The algorithm accepts as input an image representation following the Retinex Model:

$$M(x,y) = I(x,y) \times R(x,y) \quad \textbf{(1)}$$

where $M(x,y)$ is the original image, $I(x,y)$ is the illumination component, and $R(x,y)$ is the reflectance component.

Step 1. The logarithm of both sides of the Retinex model is taken to convert the illumination and reflective components from multiplicative form to additive form:

$$\ln M(x,y) = \ln I(x,y) + \ln R(x,y) \tag{2}$$

Step 2. The fast Fourier transform is applied to convert the image from the spatial domain to the frequency domain:

$$FFT[\ln M(x,y)] = FFT[\ln I(x,y) + \ln R(x,y)] \tag{3}$$

More concisely, Formula 3 can be written as:

$$M(u,v) = I(u,v) + R(u,v) \tag{4}$$

where $M(u,v)$, $I(u,v)$ and $R(u,v)$ are the Fourier transforms of $M(x,y)$, $I(x,y)$ and $R(x,y)$ respectively. Note that $I(u,v)$ is mainly concentrated in the low frequency range while $R(u,v)$ is concentrated in the high frequency range.

Step 3. An appropriate high-pass filter with transfer function $H(u,v)$ is applied to perform the enhancement:

$$S(u,v) = H(u,v) \times M(u,v) \ = H(u,v)I(u,v) + H(u,v)R(u,v) \tag{5}$$

Step 4. The inverse Fourier transform is applied to transform the image from the frequency domain to the spatial domain. Let $s(x,y)$ be the inverse Fourier transform of $S(u,v)$, then the inverse Fourier transform of Formula 5 becomes:

$$s(x,y) = IFFT\big(H(u,v)I(u,v)\big) + IFFT\big(H(u,v)R(u,v)\big) = h_I(x,y) + \ h_R(x,y) \tag{6}$$

Step 5. Finally, the exponential operation is applied on Formula 6 to obtain the enhanced image denoted by $E(x,y)$:

$$E(x,y) = \exp[s(x,y)] = \exp[h_I(x,y)]\exp[h_R(x,y)] \tag{7}$$

4. LLI Enhancement Model

We design a novel LLI enhancement model titled LLHFNet (Low-light Homomorphic Filtering Network) which performs image-to-frequency filter learning instead of the typical image-to-image learning paradigm adopted by most existing solutions. The overall model architecture is depicted in Figure 2. It is based on HF where a special filter of two parameters is devised to filter the image frequency components in the Fourier transform domain. The two parameters are estimated using a typical DL-based feature extractor utilized in classification models. In the following, we describe the main components of our model including: i) enhancement filter design, ii) DL network architecture, and iii) loss function.

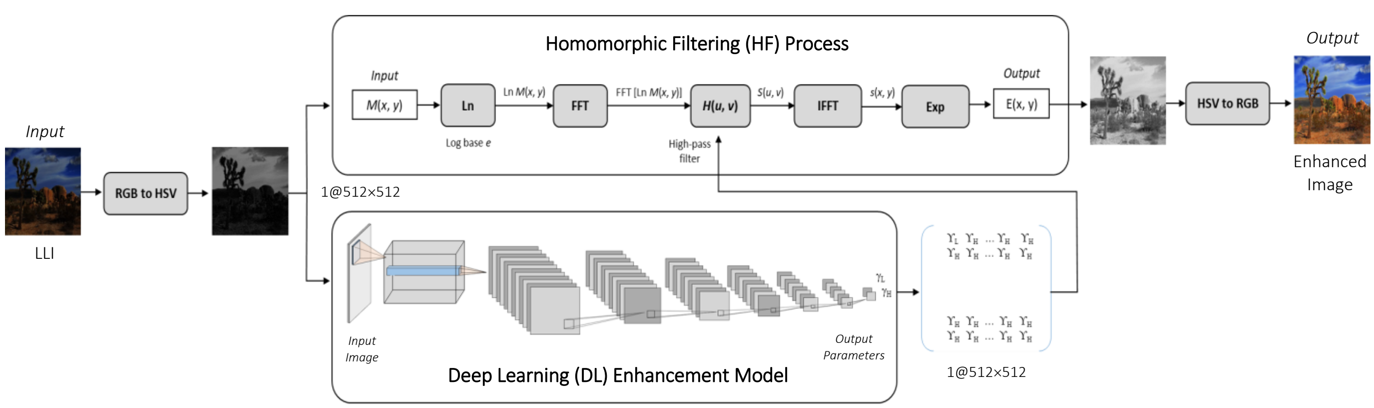


Figure 2. LLHFNet image enhancement framework

4.1. Enhancement Filter Design

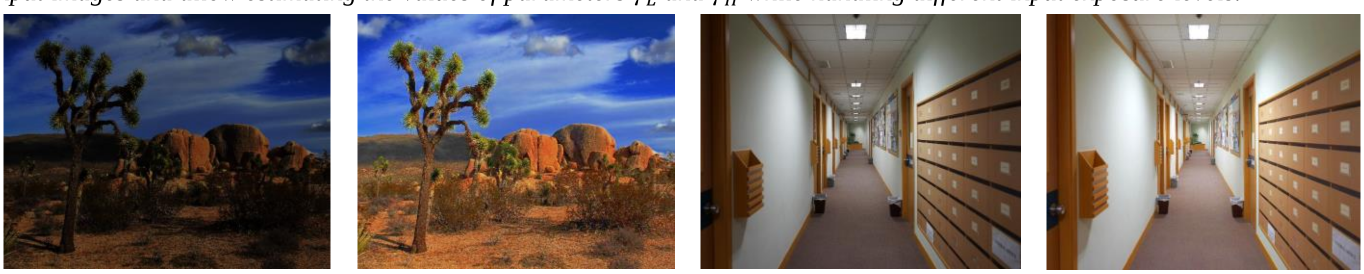
A core part of the HF algorithm is the frequency filtering transform $H(u, v)$. In our design, we aim to produce a simple and effective filter transform that can be easily learned by the enhancement network. Here, the Fourier transform of the original image, i.e., $M(u, v)$ at (0,0), represents its DC-term[1] which corresponds to its average brightness in the spatial domain [38]. In this context, we make two interesting observations: i) $M(0,0)$ with LLIs is a large negative value reflecting the low brightness of these images, whereas ii) $M(0,0)$ for NLIs is either a small negative value or a positive value reflecting the normal brightness of these images. Based on the latter observations, we assume that brightness can be enhanced by increasing $M(0,0)$. As a result, we define our enhancement filter as follows:

$$H(u, v) = \begin{cases} \gamma_L & (0,0) \\ \gamma_H & otherwise \end{cases} \quad \textbf{(8)}$$

where $\gamma_L \in [0,1]$ denotes the lightness parameter associated with low-frequency components and is placed at $H(0,0)$, and $\gamma_H \in [0,1]$ denotes the sharpness parameter associated with the remaining higher-frequency components of $M(u, v)$ corresponding to the image variations. The filter's behavior can be described as follows: i) the smaller (larger) the value of parameter γ_L, the higher (lower) the brightness level of the image, ii) the larger (smaller) the value of γ_H, the sharper(blurrier) the contents of the image.

Consequently, we run the HF algorithm by applying our enhancement filter on the Value channel of the HSV (Hue-Saturation-Value) color domain, instead of using the Red, Green, and Blue channels of the RGB domain. We make this design choice for the following reasons: i) it is more efficient to apply the Fourier transform and its inverse on one channel only instead of three, ii) the Value channel in HSV describes the lightness of the image which we aim to improve; while Hue and Saturation remain unchanged, and iii) HSV allows more simplicity with only two required parameters, compared with the RGB domain which may require two parameters for each of its channels to achieve a good enhancement quality.

Figure 3 provides two examples highlighting the behavior of our enhancement filter with different exposure levels. On the one hand, Figure 3a presents a LLI with a low exposure level, requiring parameter values <0.5 ($\gamma_L = 0.35$, $\gamma_H = 0.45$*) to produce a visually pleasing enhanced image with minimal artifacts. On the other hand, Figure 3b presents a LLI with a medium exposure level, requiring relatively higher parameter values (*$\gamma_L = 0.60$, $\gamma_H = 0.70$*) to perform a minimal enhancement while avoiding overexposure. Here, there is a need to identify and fine-tune the parametric values of the filter function in order to maximize image enhancement quality. So, we develop a DL network model which can powerfully and efficiently extract high-level features from input images and allow estimating the values of parameters* γ_L *and* γ_H *while handling different input exposure levels.*



LLI | Enhanced Image ($\gamma_L = 0.35$, $\gamma_H = 0.45$) | LLI | Enhanced Image ($\gamma_L = 0.60, \gamma_H = 0.70$)

a. LLI with low exposure level and its enhanced counterpart

b. LLI with medium exposure level and its enhanced counterpart

Figure 3. LLIs with from the SICE dataset *[39]* and their enhanced counterparts using the HF algorithm

4.2. Deep Learner Network Architecture

Our DL network architecture is depicted in Figure 4. It consists of two main parts: i) feature extractor, and ii) enhancement head.

The *feature extractor* is responsible for extracting high-level features from the input images. In contrast with image-to-image learning models where custom architectures are required for specific approaches, our solution allows the usage of any feature extractor network (e.g., VGG16 [20], ResNet50 [19], MobileNetv2 [21], SqueezeNet [22], among others) to perform the image-to-filter mapping, which comes down to estimating filter parameters γ_L and γ_H. We modify the first layer of the extractor to accept as input the Value channel of the image represented in the HSV domain.

The *enhancement head* consists of four convolutional layers followed by ReLU activation and max pooling layers, allowing to downsize the feature maps obtained from the feature extractor. The last convolutional layer is followed by an adaptive average pooling layer to resize the network output to size 1x2x1, and then a Sigmoid activation function to limit the 2 output values representing γ_L and γ_H to the range [0,1], following our enhancement filter definition described in the previous section.

[1] The DC-term is the 0 Hz term and is equivalent to the average of all the samples in the sampling window.

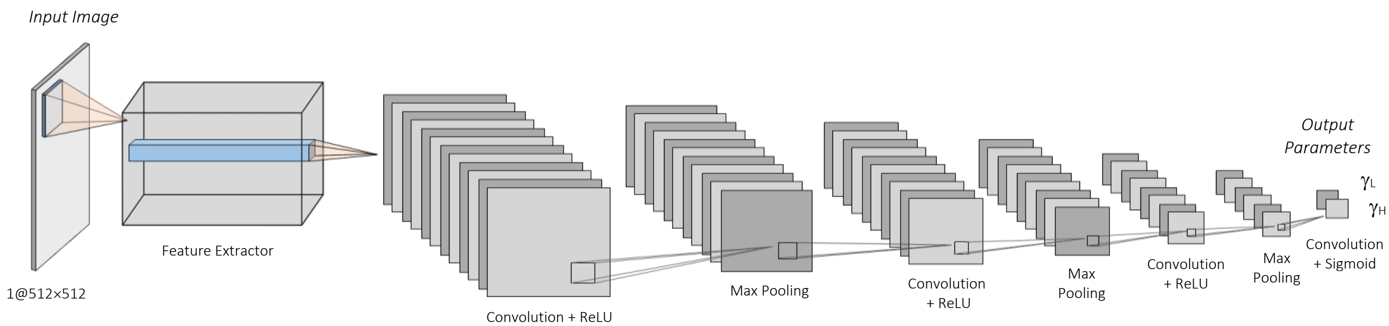


Figure 4. DL enhancement model network architecture

4.3. Enhancement Loss Function

The loss function is a major element of the LLI enhancement model and drives the entire learning process. In our approach, we adopt a supervised training setting in which reference-based loss functions are needed. We rely on Multi-scale Structural Similarity Index Measure (MS-SSIM) [40] for our loss function. MS-SSIM is an advanced version of SSIM (cf. Formula 12) which conducts assessment over multiple scales of the image. Yet a recent empirical evaluation in [18] shows that quantitative image quality assessment metrics do not always correlate with the human perception of visual quality, due to the disparity between computational enhancement (done by the machine) and enhancement quality (perceived by humans). While the latter miscorrelation is difficult to evaluate through the loss function with existing image-to-image learning models, yet it is easier to monitor with our image-to-filter enhancement model (which seeks to learn two filter parameters only, rather than learning the image as a whole). In this context, a preliminary evaluation of our enhancement model shows two contradictory observations. On the one hand, an MS-SSIM based loss function may show a tendency to generate values for the lightness parameter γ_L which are greater than those of the sharpness parameter γ_H. This tends to produce enhanced images which are smoothed with distinctive color deviations, making them perceptually unpleasing. On the other hand, this tendency is encouraged by lower MS-SSIM loss values indicating that the metric is failing to properly quantify the quality of these enhanced images. To minimize the impact of this miscorrelation between qualitative perception and quantitative measure, we add a regularization term to the loss function, encouraging the learner model to generate values for γ_H which are greater than γ_L while reducing the overall loss value. More formally:

$$\text{enhLoss}(I_{\text{Enhanced}}, I_{\text{NLI}}) = 1\text{-MS\_SSIM}(I_{\text{Enhanced}}, I_{\text{NLI}}) + \alpha \times \ell \qquad \textbf{(9)}$$

where *enhLoss* designates the enhancement loss function, $I_{Enhanced}$ is the enhanced image, I_{NLI} is the normal light image, $\ell = \gamma_L - \gamma_H$ is the regularization term, $\alpha >= 0$ is a linear weight parameter highlighting the impact of regularization on overall loss. Our empirical evaluation (cf. Section 6) shows that values of α ranging between $[0.05, 0.1]$ produce satisfactory LLI enhancement results (in our experiments, we use $\alpha = 0.08$). Deciding on the best value of α that optimizes the loss function and image enhancement quality requires a dedicated optimization process, and will be handled in a future study.

5. Enhancer-Classifier Model

In this section, we describe our *LLI enhancer-classifier* model, which integrates LLHFNet into an existing object classification model, namely ResNet50 [19], in order to perform a joint training that optimizes both enhancement and classification simultaneously. We choose ResNet50 [19] as one of the most effective classification solutions in the literature, boasting a smaller model size and requiring a lower training time compared with other denser alternatives [41]. Note that LLHFNet is not tied to ResNet50, and is designed to use typical feature extractors utilized with existing classification models including VGG16 [20] , MobileNetv2 [21], and SqueezeNet [22], among others. The enhancement capability is embedded to the classifier model, allowing it to handle LLIs and NLIs simultaneously, and produce a more robust classification result.

5.1. Enhancer-Classifier Design

Our enhancer-classier model design is shown in Figure 5. It accepts the input image and feeds it into a first classification feature extractor (e.g., ResNet50 in our case, yet other classification feature extractors can be used). The output feature maps of the first feature extractor are passed to the enhancement head (cf. Figure 4) to estimate the frequency filter parameters. The latter are then processed through the HF algorithm (cf. Figure 1) to produce the enhanced image, which will be optimized by the enhancement loss (cf. Formula 9). We add the fully connected (FC) layer of the first feature extractor at its output branch, and use it to evaluate classification loss (cf. Formula 11). Note that the first feature extractor is performing a dual task of optimizing both enhancement and classification performance, and adding the FC layer allows to improve its classification performance (cf. experiments in Section

6). Then, the enhanced image is fed into a second classification feature extractor (e.g., RestNet50 in our case), to extract features which are at the same level of those obtained from the first extractor. Both feature maps from the first and the second feature extractors are merged in an element-wise addition. The merged maps are passed to the FC layer of the second feature extractor to produce as output the final classification scores.

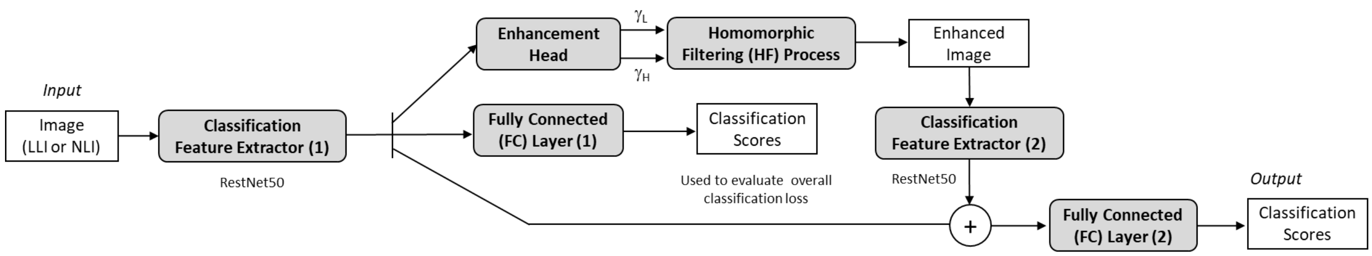


Figure 5. LLHFNet Enhancer-Classifier model

5.2. Enhancer-Classifier Loss Function

The enhancer-classifier loss function is used to perform a joint optimization for both classification and enhancement. It consists of the aggregation of two loss functions: enhancement loss (*enhLoss*) and classification loss (*clsLoss*), and is formally defined as follows:

$$\mathrm{Loss}(\mathrm{I_{Enhanced}}, \mathrm{I_{NLI}}, \mathrm{L_P}, \mathrm{L_G}) = \beta \times \mathrm{enhLoss}(\mathrm{I_{Enhanced}}, \mathrm{I_{NLI}}) + \gamma \times \mathrm{clsLoss}(\mathrm{L_P}, \mathrm{L_G}) \quad \textbf{(10)}$$

Where L_P is the predicted class label, L_G is the ground truth class label, *enhLoss* is defined following Formula 9 (cf. Section 4.3), $\beta, \gamma \geq 0$ are linear weight parameters, and *clsLoss* is defined as the cross entropy loss (commonly used in the classification task) for both FC layers, more formally:

$$\mathrm{clsLoss}(\mathrm{L_P}, \mathrm{L_G}) = \rho \times \mathrm{clsLoss1}(\mathrm{L_P}, \mathrm{L_G}) + \sigma \times \mathrm{clsLoss2}(\mathrm{L_P}, \mathrm{L_G}) \quad \textbf{(11)}$$

where $clsLoss1$ is the loss from the first FC layer, and $clsLoss2$ is the loss from the second FC layer, $\sigma, \rho \geq 0$ are linear weight parameters. Our empirical evaluations show that equal linear weight parameters $\beta = \gamma = \sigma = \rho = 1$ produce satisfactory enhancement and classification results. Note that deciding on the best parameter values requires a dedicated optimization process, which we will address in a dedicated future study.

6. Experimental Evaluation

Our empirical evaluation consists of two main experiments to evaluate the performance of LLHFNet: i) as a standalone LLI enhancement model, and ii) as an integrated enhancer-classifier model. Our prototype implementation and experimental data are available online[2].

6.1. LLI Enhancement Evaluation

In this experiment, we perform an image quality assessment (IQA) that aims at evaluating whether an image is visually pleasing and how it is visually perceived. Image quality refers to the different visual attributes of the image and focuses on the perceptual assessment of viewers. IQA methods are either i) quantitative: based on objective evaluation metrics, or ii) qualitative: based on the human perception of visual quality. Here, we conduct both quantitative and qualitative evaluations, by evaluating the visual quality achieved by 5 prominent enhancement models (2 traditional solutions: SRIE [9] and LIME [10], and 3 DL-based solutions: ZeroDCE[3] [17], EnlightenGAN[4] [16] and DeepUPE[5] [13]). We compare the models with our LLHFNet[2] implemented using PyTorch on a P100 Tesla Nvidia GPU, with a batch size of 8. We utilize an Adam optimizer with default parameters and a reduce-on-plateau based decaying learning rate with an initial value of 1e-4 for network optimization.

6.1.1. Experiment Data

We use the well-known SICE dataset [39] to conduct our training and testing experiments. It includes 360 multi-exposure sequences allowing the model to be trained on a variety of exposure conditions ranging from low-exposure to high-exposure images. We adopt two subsets for: i) training and ii) testing. The training subset consists of 2,150 image pairs from Part 1 of SICE,

[2] https://github.com/rayanalsubbahi/LLHFNet
[3] https://github.com/Li-Chongyi/Zero-DCE
[4] https://github.com/TAMU-VITA/EnlightenGAN
[5] https://github.com/wangruixing/DeepUPE

excluding extremely underexposed and overexposed images (which are difficult to handle and may tend to disrupt the training process). We resize all the training images to 512x512, and perform cross validation where 1700 pairs (i.e., 80%) are used for model learning and 450 pairs (i.e., 20%) are used for model evaluation. Although the training dataset seems relatively small, yet our enhancement model does not require huge training data since it relies on powerful pre-trained feature extractors for its backbone. In this experiment, we utilize five pre-trained extractors including VGG16 [20], ResNet50 [19], MobileNetv2 [21], SqueezeNet [22] and DenseNet [42].

As for the testing subset, it consists of 767 paired LLIs/NLIs collected from Part 2 of the SICE dataset [39] and resized to 1200x900x3 following the same approach adopted in [17] to perform our empirical evaluations.

6.1.2. Evaluation Metrics

To perform a quantitative evaluation, we run the enhancement models against three objective metrics commonly used in the literature: Structural Similarity index (SSIM) [43], Peak Signal to Noise Ratio (PSNR), and Mean Absolute Error (MAE). SSIM [43] measures the structural similarity between images based on independent comparisons of their luminance, contrast, and structure features, and is defined as:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C1)(2\sigma_{xy} + C2)}{(\mu_x^{\ 2} + \mu_y^{\ 2} + C1)(\sigma_x^{\ 2} + \sigma_y^{\ 2} + C2)} \quad \textbf{(12)}$$

where x is a ground truth image with N pixels and maximum pixel value L, y is the enhanced image, $\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$, $\sigma_x = (\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2)^{1/2}$, $\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$ $C1 = (k_1 L)^2$ and $C2 = (k_2 L)^2$ are constants for avoiding instability, with $k_1 << 1$ and $k_2 << 1$.

PSNR considers the maximum pixel value (denoted as L) and the mean squared error (MSE or L2 loss) between images. Given a ground truth image x with N pixels and the corresponding enhanced image y, PSNR is defined as:

$$\mathrm{PSNR}(x, y) = 10 \times \log_{10}\left(\frac{L^2}{\frac{1}{N}\sum_{i=1}^{N}(x(i) - y(i))^2}\right) \quad \textbf{(13)}$$

Similarly, MAE measures the absolute errors between paired observations, and allows to perform IQA by measuring the absolute difference in pixel values between LLI/NLI pairs:

$$\mathrm{MAE}(x,y) = \frac{1}{N}\sum_{i=1...N} |x(i) - y(i)| \quad \textbf{(14)}$$

6.1.3. Quantitative Evaluation Results

Table 1 shows quantitative IQA results comparing LLHFNet with its prominent counterparts, applied on the SICE testing subset. Results show that our solution produces the best PSNR and MAE average scores, as well as the second best scores following SSIM. In addition, Table 2 provides the average scores of LLHFNet using different feature extractors, including: ResNet50, MobileNetv2, VGG16, DenseNet, and SqueezeNet. MobileNetv2 and VGG16 produce some of the best average scores across all evaluation metrics. This is probably due to their dense architectures. ResNet50 and DenseNet are ranked 3rd and 4th respectively, while SqueezeNet produces the worst results across all evaluation metrics, which is probably due to its lightweight architecture. Yet all feature extractors show consistently competitive results when compared with the enhancement solutions in Table 1.

To sum up, results in this experiment show that LLHFNet can be effectively used with different feature extractors, making it independent of any specific architecture and thus easy to integration with object classification models.

Table 1. Quantitative results comparing the quality of existing LLI enhancement models and LLHFNet. The best result is shown in red and the second best result is shown in green. LLHFNet uses MobileNetv2 [21] as its feature extractor.

Model	SSIM ↑	PSNR ↑	MAE ↓
LLHFNet	0.58	16.89	94.99
ZeroDCE	0.59	16.57	98.78
EnlightenGAN	0.59	16.21	102.78
DeepUPE	0.49	13.52	142.01
LIME	0.57	16.17	108.12
SRIE	0.54	14.41	127.08

Table 2. Quantitative results comparing different feature extractors used with LLHFNet. The best result is shown in red and the second best result is shown in green.

Feature Extractor	SSIM ↑	PSNR ↑	MAE ↓
MobileNetv2	0.583	16.896	94.992
VGG16	0.582	16.897	94.064
ResNet50	0.577	16.686	96.152
DenseNet	0.576	16.716	97.253
SqueezeNet	0.575	16.593	99.129

6.1.4. Qualitative Evaluation Results

In addition to the quantitative evaluation, we also perform a qualitative evaluation to assess the human visual perception of images enhanced by our model and its five counterparts considered in this experiment. To do so, we randomly select 20 images from the SICE testing subset, and display the reference input LLI and the enhanced image side by side in a dedicated online survey[6]. Responders are asked to rate each image considering three visual IQA criteria including: i) level of exposure (over/under-exposed regions), ii) color deviations, and iii) overall beauty of the image. A total of 76 testers (senior computer engineering and master's students) were invited to contribute in the experiment, and independently rate every enhancement model on an integer scale from 1 to 10 (i.e., worst to best). We also deal with inconsistencies in image ratings by computing the average score for every image, and then eliminating ratings which have an extreme deviation from the average (e.g., ratings which are extremely low/high for images deemed visually pleasing/unpleasing by most testers). A total of at least 1200 responses were collected for each model, with every image receiving 60 rating scores. The ratings are aggregated for every enhancement model to evaluate its overall perceptual quality. Results are provided in Figure 6, and sample LLIs and NLIs are visualized in Figure 7 and Figure 8 respectively.

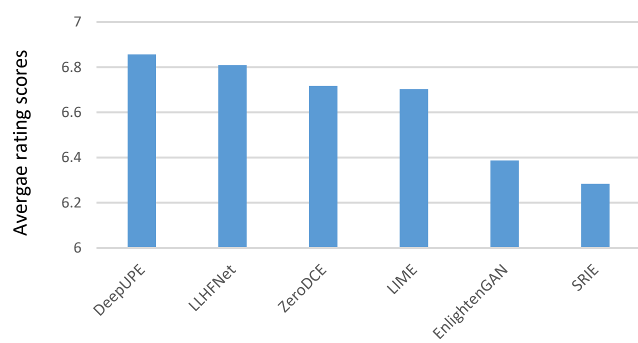


Figure 6. Average user ratings for the enhancement models ranked from best to worst

Results in Figure 6 show that LLHFNet ranks second best among the five compared models, and is thus favored by human testers. Sample LLIs in Figure 7 show that LLHFNet produces visually pleasing enhanced images with minimal artifacts. In the first image (Figure 7a), our model is able to uncover the dark regions of the fence and is able to effectively restore the green colors of the trees. In the second image (Figure 7b), our model properly restores the colors of the trees, grass, and white clouds without overexposing them (compared with EnlightenGAN where the clouds are overexposed, and ZeroDCE and SRIE where the cloud colors and overall image colors deviate into blue). In the third image (Figure 7.c), our model shows a good illumination level and produces results comparable with to ZeroDCE and SRIE.

[6] https://forms.gle/FrjzGAZXpyKqGRnw9

In addition to the LLIs in Figure 7, Figure 8 shows the enhancement results applied on a sample LLI with almost a normal exposure level. While models like LIME and EnlightenGAN tend to overexpose certain parts of the image, especially the light from the windows, LLHFNet performs a slight and minimal enhancement and preserves most of the original colors of the image. Recall that our enhancement approach is designed to properly handle normal exposure levels in input images, by processing the input image with minimal enhancement or even no enhancement with filter parameters γ_L, $\gamma_H \approx 1$.

6.1.5. Discussion

Results show that LLHFNet is ranked among the best enhancement models compared with its five counterparts. Its good performance is mainly due to its easy integration and usage of very powerful pre-trained extractors like ResNet50 as its backbone, allowing LLHFNet to easily generalize and handle both LLIs and NLIs. It can also be partly due to its original image-to-filter approach where the enhancement network focuses on optimizing two filter parameters only, compared with the image-to-image paradigm adopted by most existing models following a more complex multi-parameter optimization process.

6.2. Enhancer-Classifier Evaluation

High-level computer vision tasks like object classification usually suffer from a degraded performance when processing LLIs [1] [18]. In this experiment, we aim to verify whether our integrated LLHFNet enhancer-classifier model can improve the performance of the object classification task. To do so, we perform a comparative analysis: i) using the five LLI enhancement models considered in our first experiment in addition to our standalone LLHFNet, following a typical classification process consisting of separate image enhancement followed by object classification on the enhanced images, ii) using LLHFNet integrated within its novel enhancer-classifier model, performing a joint training to optimize both enhancement and classification performance simultaneously.

6.2.1. Experiment Data

To train our enhancer-classifier model, we utilize the well known Pascal VOC (2012 + 2007) dataset [23] considering a training subset of 9,625 images labelled under 20 single generic classes (e.g., *chair*, *cat*, *car*, etc.). We synthetically generate LLIs considering five different exposure levels using gamma correction with gamma values {4.5, 3.5, 2.5} corresponding to low-exposure levels, and gamma values {0.5, 0.8} corresponding to high-exposure levels. We then perform cross validation where 8500 LLI/NLI pairs (i.e., 88%) are used for model learning and 1125 pairs (i.e., 12%) are used for model evaluation, such that both training and evaluation subsets are equally divided among the used exposure levels along with their class labels. All images are converted to the HSV domain where the Value channel-based image is used in training.

As for model testing, we utilize another subset of Pascal VOC 2007 consisting of 3,000 image pairs, divided equally among the used (Υ corrected) exposure levels. In addition to these synthetic images, we utilize 3000 real LLIs from the ExDark dataset [24], to further test the performance of our enhancer-classifier model in a real-world setting. All the images for training and testing are resized to 512x512, following the same approach adopted in [17].

6.2.2. Experiment Set-up

Similarly to our previous experiment, we implement our LLHFNet enhancer-classifier using PyTorch on a P100 Tesla Nvidia GPU, with a batch size of 8. We utilize an Adam optimizer with default parameters and a reduce-on-plateau based decaying learning rate with an initial value of 1e-5 for network optimization. For the first epoch, we multiply the classification loss by 0.1 (i.e., $\beta = 1, \gamma = 0.1$ cf. Formula 10) to give it less weight to the advantage of stabilizing enhancement and warming-up the joint models. As for the embedding classifier, we use ResNet50 pre-trained on the ImageNet database [44] (any other classifier could have been used, as mentioned previously).

6.2.3. Experiment Results

Our empirical evaluation consists of four main tests: i) evaluating the traditional classification pipeline, ii) evaluating the new integrated enhancer-classifier solution, iii) comparing the enhancer-classifier model with existing enhancement solutions, and iv) evaluating the relationship between enhancement quality and classification performance.

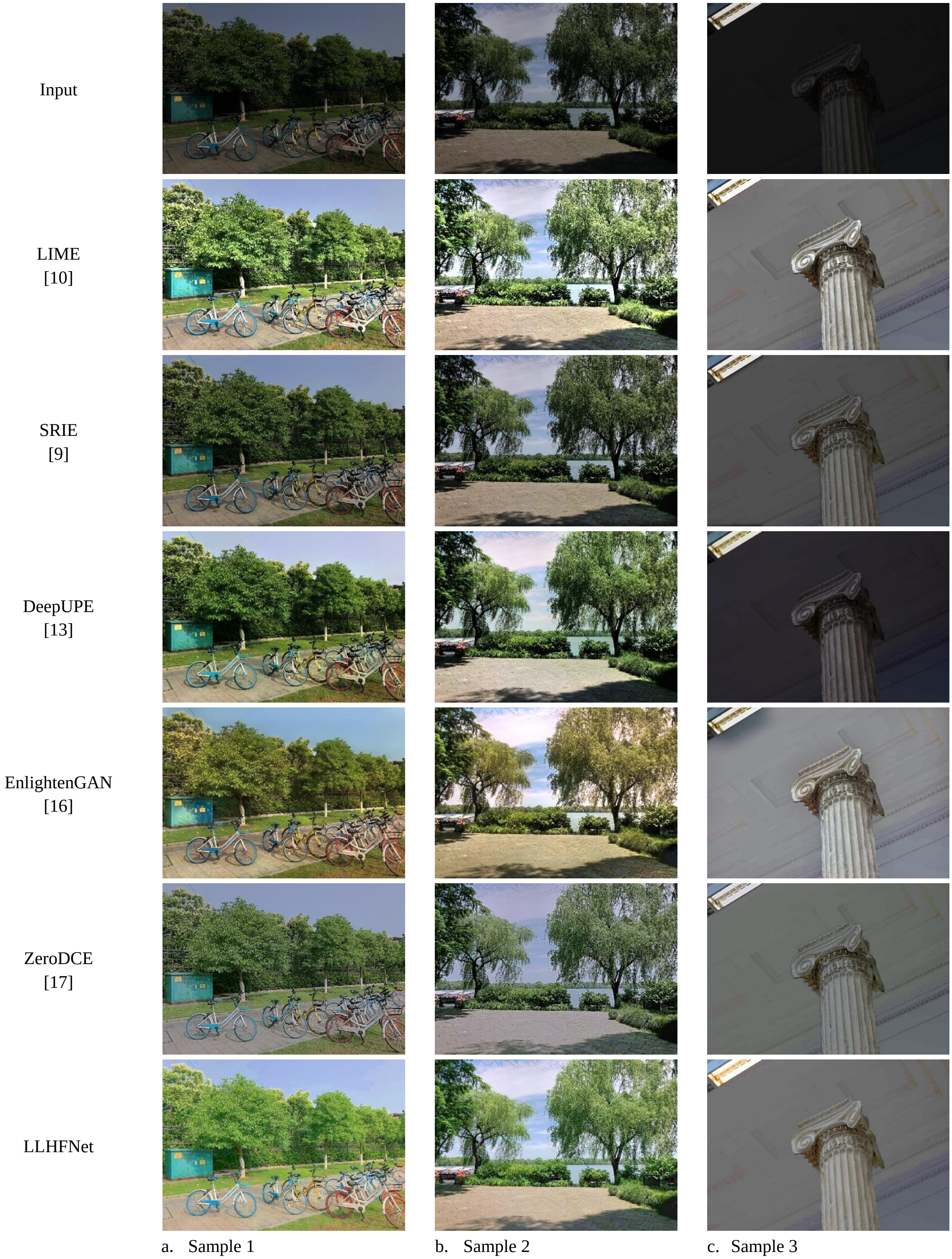


Figure 7. Visual comparison of sample LLIs from the SICE Part 2 subset [39] and their enhanced versions.

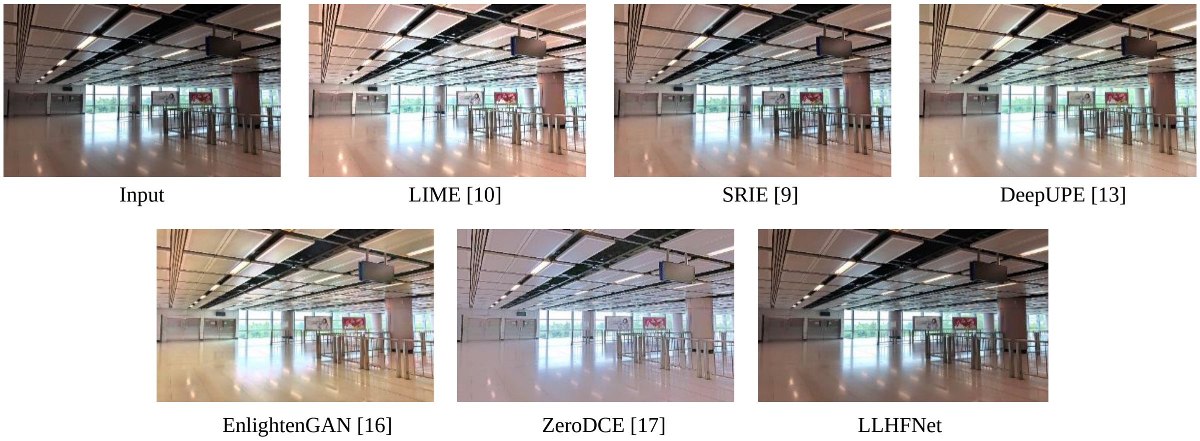


Figure 8. Visual comparison of an input image with almost normal exposure level from SICE Part 2 [39] and its enhanced versions.

Test 1: Evaluating the Traditional Classification Pipeline

We first start by evaluating the traditional classification pipeline adopted in the literature, i.e., performing LLI enhancement separately as a preprocessing step, and then performing classification on the enhanced images using a classifier pre-trained on abundantly available NLIs. We build six variants of the traditional pipeline using the five enhancement models considered in our first experiment (i.e., SRIE, LIME, ZeroDCE, EnlightenGAN, and DeepUPE) in addition to our standalone LLHFNet. We then perform the classification task using the ResNet50 classifier trained on the NLIs of our training dataset, and tested on the standalone enhanced images produced by each of the six enhancement models. Average classification accuracy levels are provided in table 3. Results show that LLHFNet produces the best RestNet50 classification accuracy results on both Pascal VOC and ExDark test subsets (i.e., 80.81% and 66.18% respectively) compared with the existing enhancement solutions. In other words, the standalone LLHFNet model, used as a LLI pre-processing step with the traditional pipeline, is able to significantly improve the target classification task.

Table 3. Classification accuracy results following the traditional pipeline (enhancement, followed by classification), applied on Pascal VOC and ExDark test subsets. We apply the five enhancement models considered in the first experiment in addition to our standalone LLHFNet, and we utilize ResNet50 classification.

Pascal VOC	
Model	**Accuracy (%) ↑**
LLHFNet	**80.81**
ZeroDCE	**79.87**
SRIE	79.71
DeepUPE	79.53
EnlightenGAN	78.40
LIME	75.90
Original	78.09

ExDark	
Model	**Accuracy (%) ↑**
LLHFNet	**66.18**
ZeroDCE	**65.30**
DeepUPE	64.34
SRIE	63.27
EnlightenGAN	62.17
LIME	61.51
Original	61.17

Test 2: Evaluating the New Enhancer-Classifier Approach

Second, we perform a comparative analysis by training three classification models: 1) Classifier_NLIs: the classifier from the traditional pipeline described in Test 1, consisting of ResNet50 with no enhancement capability trained on the NLIs of the training dataset, 2) Classifier_LLIs: the classifier from the traditional pipeline, consisting of ResNet50 with no enhancement capability trained on the LLIs of the training dataset, and 3) Enhancer_Classifier: our LLHFNet based enhancer-classifier model trained on the LLIs of the training dataset. Table 4 shows the classification accuracy results obtained by the three trained models, evaluated on: i) the Pascal VOC (synthetic) LLI test subset, ii) the ExDark (real) LLI subset, and iii) the Pascal VOC (reference) NLI test subset. Based on the results, we highlight the following observations:

i. With **Pascal VOC (synthetic) LLI test subset**: Enhancer_Classifier shows an improvement of 3.86% over Classifier_LLIs, and a 8.13% improvement over Classifier_NLIs, highlighting LLHFNet's higher performance in handling synthetic LLIs.
ii. With **ExDark (real) LLI subset**: Enhancer_Classifier produces a 3.25% improvement over Classifier_LLIs, and a 10.56% improvement over Classifier_NLIs, confirming LLHFNet's good performance on real-world LLIs.

iii. With **Pascal VOC (reference) NLI test subset**: Enhancer_Classifier also achieves the best accuracy levels, producing a 2.16% improvement over Classifier_LLIs, and a 1.88% improvement over Classifier_NLIs which is trained on NLIs. This confirms the ability of LLHFNet in handling NLIs, producing improved classification levels equivalent to training the classifier on NLIs only, by adding only minor enhancements through LLHFNet's filter parameters and avoiding over exposure (cf. Section 6.1).

iv. With **Pascal VOC (synthetic) LLI and (reference) NLI test subsets**: The Classifier_LLIs model shows a better performance compared with Classifier_NLIs, with a 4.27% improvement on LLIs, and approximately similar performance with a slight regression of -0.28% on NLIs. This indicates that Classifier_LLIs seems robust against varying light conditions. In addition, the degraded performance of Classifier_NLIs when evaluated against LLIs shows that training a classifier on varying exposure levels seems better than limiting the training to normal light conditions only.

Table 4. Classification accuracy (in %) of the three trained models evaluated using Pascal VOC and ExDark test subsets.

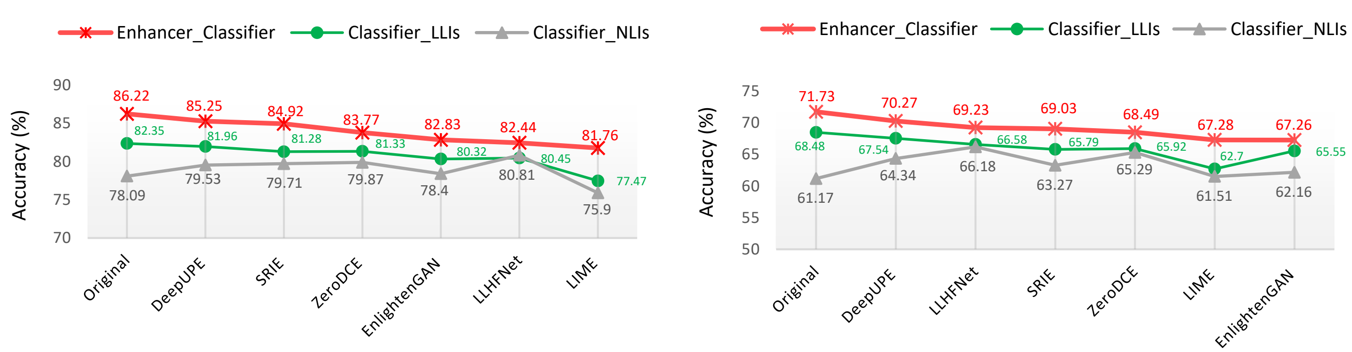
	LLIs		NLIs
	Pascal VOC (synthetic) subset	**ExDark** (real) subset	**Pascal VOC** (reference) subset
Enhancer_ Classifier	**86.22**	**71.73**	**88.42**
Classifier_ LLIs	**82.36**	**68.48**	86.26
Classifier_ NLIs	78.09	61.17	**86.54**

Test 3: Comparing the Enhancer-Classifier Approach with Existing Enhancement Models

In our third evaluation, we compare the performance of the three aforementioned classification models using the standalone enhanced images from the Pascal VOC and ExDark test subsets. We perform image enhancement using the five enhancement models considered in our study (i.e., SRIE, LIME, ZeroDCE, EnlightenGAN, and DeepUPE) in addition to our standalone LLHFNet. Average classification accuracy levels are provided in Figure 9. Based on the results, we highlight the following observations:

i. Enhancer_Classifier produces the best classification results, compared with all other classifiers following the traditional pipeline. This is probably due to the fact that enhancement is performed at two stages: i) at the initial pre-processing stage done by the standalone enhancement models, and ii) internally through LLHFNet and its integrated classification feature extractors in which minor restorations are added when needed. This indicates that Enhancer_Classifier can effectively adapt to the data domain of enhanced images which may contain artifacts and amplified noise, while improving classification performance.

ii. The Classifier_NLIs model trained only on NLIs shows the worst classification results compared with the other two classification models. This may indicate that the data domain of NLIs does not correlate with that of enhanced images which contain artifacts, noise, and varying light conditions. Therefore, processing separately enhanced images using classifiers pretrained on NLIs may not be the best strategy to benefit from the pre-processing enhancement task.



a. Results on the Pascal VOC test subset

b. Results on the ExDark test subset

Figure 9. Classification accuracy (in %) obtained by the three trained models evaluated on the original and standalone enhanced images from Pascal VOC (a) and ExDark (b) test subsets

Test 4: Evaluating the Relationship between Enhancement Quality and Classification Performance

In our fourth evaluation, we take a closer look at the enhancement performance of our Enhancer_Classifier model, in order to better understand the impact of enhancement quality on classification quality. In other words, we aim to discover whether the Enhancer_Classifier model producing the best classification quality results can also produce the best enhancement quality results. Average scores of the IQA evaluation metrics (i.e., SSIM, PSNR, and MAE) applied on the Pascal VOC test subset[7] are shown in Table 5. Based on the results, we highlight the following observations:

i. Enhancer_Classifier achieves the best enhancement performance following all three metrics, indicating the effectiveness of its internally embedded enhancement. This reflects that the joint optimization used in the integrated Enhancer_Classifier model has effectively improved the quality of both image enhancement and image classification tasks.

ii. LLHFNet, used a standalone enhancement model, achieves the second best performance levels following all three evaluation metrics. This confirms the results obtained in our first experiment (cf. Section 6.1) and shows that our designed enhancement algorithm is effective and provides a clear improvement compared with existing state of art enhancement models.

Table 5. Quantitative evaluation of image enhancement quality of Enhancer_Classifier compared with our standalone LLHFNet model and other existing enhancement models.

Model	SSIM ↑	PSNR ↑	MAE ↓
Enhancer_Classifier	**0.76**	**16.96**	**105.64**
LLHFNet	**0.731**	**15.69**	**119.92**
DeepUPE	0.730	14.30	143.89
ZeroDCE	0.67	14.96	139.05
SRIE	0.629	13.50	154.69
LIME	0.6286	13.33	159.68
EnlightenGAN	0.6284	13.63	152.32

7. Conclusion

In this paper, we introduce a new LLI enhancement solution designed for the object classification task. Our solution consists of two contributions: i) a novel *LLI enhancement* model titled LLHFNet (Low-light Homomorphic Filtering Network) based on image-to-frequency filter learning, and ii) a *LLI enhancer-classifier* model, which integrates the enhancement model into a state of the art object classification solution (e.g., RestNet50). Through this integration, the classification model is embedded with an internal enhancement capability and is jointly trained to optimize both enhancement and classification performance. Experimental results show improved enhancement quality on LLIs, and robust classification quality on both LLIs and NLIs. Compared with the traditional pipeline consisting of separate enhancement followed by classification, our integrated enhancer-classifier model highlights a clear improvement compared with existing solutions.

As ongoing work, we are currently conducting an empirical study to evaluate the performance of our solution on extremely LLIs. Initial results show that LLHFNet usually fails to handle extremely LLIs and tends to produce artifacts, similarly to most existing enhancement models. In future works, we aim to improve our enhancement algorithm to address this problem. We also aim to integrate our enhancement model in other high-level computer vision tasks like object detection [45], localization and tracking [46], and multi-label image recognition [47].

References

[1] W. Yang et al., "Advancing Image Understanding in Poor Visibility Environments: A Collective Benchmark Study," *IEEE Trans. Image Process.*, 29:5737-5752, 2020.
[2] Y. Peng Loh, "Low-light Image Enhancement using Gaussian Process for Features Retrieval", *Signal Processing: Image Communication*, 74:175-190, 2019
[3] S.-C. Huang et al., "Efficient Contrast Enhancement Using Adaptive Gamma Correction with Weighting Distribution," *IEEE Trans. Image Process.*, 22(30):1032–1041, 2013.
[4] N. Zhi, et al."An Enhancement Algorithm for Coal Mine Low Illumination Images based on Bi-Gamma Function," *J. Liaoning Tech. Univ.*, 37(1):191-197, 2018.
[5] E. D. Pisano et al., "Contrast Limited Adaptive Histogram Equalization Image Processing to Improve the Detection of Simulated Spiculations in Dense Mammograms," *J. Digit. Imaging*, 11(4):193-200, 1998.
[6] M. Abdullah-Al-Wadud, et al., "A Dynamic Histogram Equalization for Image Contrast Enhancement", *IEEE Trans. Consum. Electron.*, 53(2):593-600, 2007.
[7] E. H. Land and J. J. McCann, "Lightness and Retinex theory", *J. Opt. Soc. Amer.*, 61 (1):1-11, 1971.
[8] Z. Rahman, et al., "Multi-scale Retinex for Color Image Enhancement", *Proc. 3rd IEEE Int. Conf. Image Process.*, pp. 1003-1006, 1996.
[9] X. Fu, et al., "A Weighted Variational Model for Simultaneous Reflectance and Illumination Estimation", *IEEE Conf. Comput. Vis. Pattern Recogn.*,2782-2790, 2016
[10] L. Li, et al., "A Low-light Image Enhancement Method for both Denoising and Contrast Enlarging," *Int. Conf. Image Process. (ICIP)*, 3730–3734, 2015.
[11] K. G. Lore, et al., "LLNet: A Deep Autoencoder Approach to Natural Low-light Image Enhancement", *Pattern Recognit.*, vol. 61, pp. 650-662, Jan. 2017.
[12] C. Li, et al., "LightenNet: A Convolutional Neural Network for Weakly Illuminated Image Enhancement", *Pattern Recognit. Lett.*, 104:15-22, 2018.

[7] Image enhancement evaluation requires LLI/NLI pairs to compute the objective evaluation metrics (similarly to the previous test). As a result, the ExDark dataset which is only made of LLIs is not utilized in this test.

[13] R. Wang, et al., “Underexposed Photo Enhancement Using Deep Illumination Estimation,” *IEEE Conf. Comput. Vis. Pattern Recognit.*, 6842–6850, 2019.

[14] C. Wei, et al., "Deep Retinex Decomposition for Low-light Enhancement", *Proc. Brit. Mach. Vis. Conf.*, pp. 1-12, 2018.

[15] F. Lv, et al., "MBLLEN: Low-light Image/Video Enhancement Using CNNs", *Proc. Brit. Mach. Vis. Conf.*, pp. 1-13, 2018.

[16] Y. Jiang et al., "EnlightenGAN: Deep Light Enhancement without Paired Supervision," *IEEE Trans. Image Process.*, vol. 30, pp. 2340-2349, Jan. 2021.

[17] C. Guo et al., "Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020.

[18] R. Al Sobbahi and J. Tekli, "Impact of Deep Learning Low-light Image Enhancement on Object Detection and Classification", *Submitted to Signal Processing: Image Communication*, 2021.

[19] K. He, et al., "Deep Residual Learning for Image Recognition", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 770-778, Jun. 2016.

[20] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in *arXiv:1409.1556*, 2015, [Online] Available: http://arxiv.org/abs/1409.1556.

[21] M. Sandler, et al., "MobileNetV2: Inverted Residuals and Linear Bottlenecks", *IEEE Conf. Comput. Vis. Pattern Recognit.*, 4510-4520, 2018.

[22] F. N. Iandola, et al., "Squeezenet: Alexnet-level Accuracy with 50x Fewer Parameters and <0.5 mb Model Size" in *arXiv:1602.07360*, 2016, [online] Available: http://arxiv.org/abs/1602.07360.

[23] M. Everingham, et al., "The Pascal Visual Object Classes (VOC) Challenge", *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303-338, Jun. 2010.

[24] Y. P. Loh and C. S. Chan, "Getting to Know Low-light Images with the Exclusively Dark Dataset", *Comput. Vis. Image Understand.*, 178:30-42, 2019.

[25] Y. Wang, et al., "Image Enhancement based on Equal Area Dualistic Sub-image Histogram Equalization Method", *IEEE Trans. Consum. Electron.*, 45(1):68-75, 1999.

[26] D. J. Jobson, et al., "Properties and Performance of a Center/surround Retinex", *IEEE Trans. Image Process.*, 6(3):451-462, 1997.

[27] D. J. Jobson, et al., "A Multiscale Retinex for Bridging the Gap Between Color Images and the Human Observation of Scenes", *IEEE Trans. Image Process.*, 6(7): 965-976, 2002.

[28] J. Wang et al., "RDGAN: Retinex Decomposition Based Adversarial Learning for Low-Light Enhancement," *IEEE Conf. Multimedia Expo (ICME*), 1186-1191, 2019.

[29] C. Zhang, et al., "Color Image Enhancement based on Local Spatial Homomorphic Filtering and Gradient Domain Variance Guided Image Filtering," *J. Electron. Imaging*, vol. 27, no. 6, 2018.

[30] L. Han et al., "Using HSV Space Real-color Image Enhanced by Homomorphic Filter in Two Channels," *Comput. Eng. Appl.*, 45(27):18-20, 2009.

[31] SyedZaheeruddin and K.Suganthib, "Image Contrast Enhancement by Homomorphic Filtering based Parametric Fuzzy Transform," *Proc. Comput. Sci.*, vol. 165, pp. 166-172, 2019.

[32] L. Tao et al., "LLCNN: A Convolutional Neural Network for Low-light Image Enhancement", *Proc. IEEE Vis. Commun. Image Process.*, pp. 1-4, Dec. 2017.

[33] K. Dabov et al., “Image Denoising by Sparse 3-D Transform-domain Collaborative Filtering,” *IEEE Trans. Image Process.*, 16(8):2080–2095, 2007.

[34] W. Wang, et al., "GLADNet: Low-light Enhancement Network with Global Awareness", *IEEE Int. Conf. Autom. Face Gesture Recognit.*, 751-755, 2018.

[35] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” in *arXiv:1505.04597*, 2015, [Online] Available: http://arxiv.org/abs/1505.04597

[36] A. Jolicoeur-Martineau, “The Relativistic Discriminator: A Key Element Missing from Standard GAN,” in *arXiv:1807.00734*, 2018, [Online] Available: http://arxiv.org/abs/1807.00734

[37] W. Wang, et al., "An Experiment-based Review of Low-Light Image Enhancement Methods," *IEEE Access*, vol. 8, 87884-87917, 2020.

[38] R. C. Gonzalez and R. E. Woods, Digital Image Processing (4th edition), New York, NY: Pearson, 2018.

[39] J. Cai et al., "Learning a Deep Single Image Contrast Enhancer from Multi-exposure Images", *IEEE Trans. Image Process.*, 27(4)2049-2062, 2018.

[40] Z. Wang et al., “Multiscale Structural Similarity for Image Quality Assessment,” *Proc. Conf. Signals, Syst. Comput.*, 2:1398-1402, 2003.

[41] Ç. F. Özgenel and A. Sorguc, "Performance Comparison of Pretrained Convolutional Neural Networks on Crack Detection in Buildings," *Proc. Int. Symp. Autom. Rob. Const. (ISAR)*, pp. 693-700, 2018.

[42] G. Huang, et al., "Densely Connected Convolutional Networks", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 4700-4708, 2017.

[43] Z. Wang, et al., "Image Quality Assessment: From Error Visibility to Structural Similarity", *IEEE Trans. Image Process.*, 13(4,):600-612, 2004.

[44] J. Deng, et al., "ImageNet: A Large-scale Hierarchical Image Database", *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 248-255, 2009.

[45] C. Salem, et al., “An Image Processing and Genetic Algorithm-Based Approach for the Detection of Melanoma in Patients,” *Methods Inf. Med*., 57(1): 74-80, 2018.

[46] D. Ebrahimi, et al., "Autonomous UAV Trajectory for Localizing Ground Objects: A Reinforcement Learning Approach," *IEEE Trans. Mobile Comput.*, 20(4): 1312-1324, 2021.

[47] L. Laib et al., "A Probabilistic Topic Model for Event-based Image Classification and Multi-label Annotation", *Signal Processing: Image Communication*, 76:283-294, 2019