Vol. 34, No. 4, 2025
(this issue contains accepted papers online and is not closed yet)

Machine GRAPHICS & VISION

International Journal

Published by
The Institute of Information Technology
Warsaw University of Life Sciences – SGGW
Nowoursynowska 159, 02-776 Warsaw, Poland
in cooperation with
The Association for Image Processing, Poland – TPO

Skin Lesion Segmentation using SegNet with Spatial Attention

Maryam Arif[1], Almas Abbasi[1], Muhammad Arif[2] and Muhammad Rashid[3]
[1]*Department of Computer Science, Faculty of Computing and Information Technology, International Islamic University, Islamabad, Pakistan*
[2]*Department of Computer Science and Artificial Intelligence, College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia*
[3]*Department of Computer and Network Engineering, College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia*
[*]*Corresponding author: Muhammad Arif (mahamid@uqu.edu.sa)*

Submitted: Jan 2, 2025 **Accepted:** May 20, 2025 **Published:** Nov 5, 2025

Abstract Skin lesion segmentation identifies and outlines the boundaries of abnormal skin regions. Accurate segmentation may help in the early detection of skin cancer. Accurate Skin Lesion Segmentation is still challenging due to different skin color tones, variations in shape, and body hairs. Moreover, variability in the lesion appearance, quality of the images, and lack of clear skin boundaries make the problem even harder. This paper proposes a SegNet model with spatial attention mechanisms for skin lesion segmentation. Adding one component of spatial attention to SegNet allows the proposed model to focus more on specific parts across the image, eventually leading to a better segmentation of the lesion boundary. The proposed model was evaluated on the ISIC 2018 dataset. Our proposed model attained an average accuracy of 96.25%, and the average dice coefficient equals 0.9052. The model's performance indicates its possible application in automated skin disease diagnosis.

Keywords: skin lesion segmentation; deep learning; spatial attention; SegNet.

1. Introduction

Skin is the largest organ of the human body that is usually directly exposed to the air. In other words, it is the most vulnerable organ due to its exposure to ultraviolet rays from the Sun and other environmental toxins. It leads to various skin diseases, including skin cancer [28]. According to the International Agency for Research on Cancer (IARC), approximately 3 330 000 new cases of skin cancer were diagnosed worldwide in 2022 [14]. Moreover, almost 60 000 people died from the disease. Furthermore, the IARC has observed that there are 5.4 million new cases of skin cancer every year [31]. Therefore, the World Health Organisation ranks skin cancer as one of the most prevalent and fastest-growing cancers globally [12].

The cause of skin cancer is the proliferation or formation of skin cells unevenly or abnormally. Depending on their type and strength, this proliferation of skin cells can infiltrate or disseminate to other areas of the body. Based on different skin cells, the three important types of skin cancers are basal cell skin cancer, squamous cell skin cancer, and Melanoma. Physicians use these abnormal cells to determine the type of

skin cancer [2, 13, 21]. Basal cell skin cancer and squamous cell skin cancer are less dangerous as they hardly result in death. However, the most dangerous type of cancer is Melanoma, accounting for around 75% of deaths attributed to skin cancer. Its formation starts in melanin-producing cells that develop in melanocytes [11].

Even though Melanoma, a frequently occurring skin cancer, is lethal and the death rate of this disease is very high, it is easily curable if the detection is made in its early stages. According to [14], the in-time diagnosis of Melanoma decreases the mortality rate by 90%. Some other studies reveal that there is a 95% early diagnosis (stage I of the disease) survival rate and a 20% late discovery rate (stage IV of the disease) [19, 31]. It implies that early detection increases the chances of survival and improves therapy efficacy. For this reason, it is critical to diagnose and treat dermatoses as soon as possible.

One of the conventional methods for the diagnosis of melanoma and other skin cancer types is the biopsy. This procedure involves taking a sample from a suspected skin lesion to perform medical tests and determine if it is cancerous. However, undergoing a biopsy can be challenging as it involves extracting a sample of the lesion. It can be uncomfortable and requires time for the procedure and the subsequent analysis. The alternative to biopsy is the visual assessment of skin lesions. Since pigmented lesions are visible on the skin's surface, a skilled visual examination can often detect Melanoma at an early stage. It often involves ABCD Scale [13] that evaluates asymmetry, border irregularity, color variegation, and lesion diameter. The ABCDE Scale [6, 24] is an extension of the ABCD scale and adds evolving to account for changes in the lesion over time. Similarly, Glasgow 7-point Checklist [8] includes major criteria such as a change in size, shape, and color, along with minor criteria like inflammation, crusting or bleeding, sensory changes, and the diameter of the lesion. These algorithms provide a structured approach to assess skin lesions and help in the early detection of Melanoma by identifying key warning signs.

Dermatologists often use a dermatoscope to enhance the visibility of skin lesions by magnifying them with light. This enhanced visibility allows dermatologists to detect early Melanoma that might be invisible to the naked eye. While dermoscopy increases detection accuracy, the complexity of skin lesions and the sheer volume of dermoscopic images make visual inspection potentially non-reproducible, time-consuming, and subjective in medical practice. That is why advancements in computer-aided diagnostic systems have become so important, offering more consistent and efficient analysis of dermoscopic images [9].

Due to the above limitations of visual inspection and dermoscopy, there is a strong motivation to develop computer-assisted diagnosis (CAD) systems to support dermatologists in their examinations. A critical aspect of CAD systems for efficiently analyzing dermoscopic images is automatically segmenting skin lesions from dermoscopic images, enabling more focused and efficient automated analysis of those areas. This automatic segmentation significantly aids early skin cancer detection and diagnosis by improving

diagnostic accuracy. However, accurate automatic segmentation of skin lesions is challenging due to three main factors: (1) Melanomas can vary greatly in shape, size, color, texture, and skin type. Distortions and natural features like hair, air bubbles, and blood vessels complicate the segmentation process. (2) Skin lesions often have fuzzy or uneven edges, and the contrast between the lesion and surrounding skin can be minimal. (3) Early algorithms were trained on relatively small datasets. Gathering large-scale skin lesion annotations from medical experts is difficult.

Existing CAD methods to address the above limitations often give unsatisfactory outcomes because they struggle with artifacts such as corners and low-contrast regions. Moreover, hairs against the background remain critical challenges, making boundary definition a difficult task [25]. On the other hand, the spatial attention mechanism uses spatial relationships of features and creates a spatial attention map. It has been observed that spatial attention modules are helpful in many image processing applications [22, 30, 36]. Therefore, this paper proposes a transfer learning-based approach combined with the spatial attention technique utilizing SegNet to improve the accuracy of skin lesion segmentation.

The paper's main contributions are summarized below:

1. **The Spatial Attention** module is introduced in the feature extraction process of the encoder. This module effectively captures spatial dependencies. This module enables the network to selectively emphasize important regions in the feature maps, improving the understanding of fine details.
2. **The bottleneck layer structure** of SegNet is modified by integrating a spatial attention module. This design increases the receptive field and allows the network to capture contextual information, resulting in more precise segmentation.

We have evaluated the performance of our proposed method on the ISIC 2018 dataset [7, 32]. This dataset contains dermoscopic images for skin lesion analysis. It was selected because it contains diverse skin lesion types collected from many patients. The proposed model showed the highest segmentation accuracy on this dataset compared to the published results on the same dataset. Hence, it proves the efficacy of the proposed model in accurate skin lesion segmentation.

2. Background and related works

This section first describes the essential background of skin lesion segmentation techniques. Subsequently, state-of-the-art skin lesion segmentation techniques are presented.

2.1. Background on skin lesion segmentation techniques

Skin lesion segmentation is pivotal in the fight against skin cancer, particularly Melanoma. Automated image analysis technique separates suspicious moles or lesions from the surrounding healthy skin in digital images, offering several benefits for dermatologists:

- **Enhanced Diagnostic Accuracy:** It provides a clear picture of the lesion's boundaries, allowing for a detailed analysis of its characteristics, such as color, texture, and borders. Moreover, it helps to distinguish between benign and malignant moles, detecting subtle variations that might be missed otherwise.
- **Earlier Detection:** By clearly highlighting suspicious areas, segmentation aids in identifying melanomas at an early stage, when treatment is most effective.
- **Improved Workflow Efficiency:** Automating the isolation of lesions saves time for dermatologists, allowing them to focus on interpreting the segmented data and making diagnoses, especially for complex cases.

However, achieving accurate segmentation is challenging due to:

- **Variability in Lesions:** Skin lesions can vary significantly in color, shape, texture, and size, making a one-size-fits-all approach difficult.
- **Artifacts:** Features like hair, blood vessels, or wrinkles can mimic lesion features and complicate the segmentation process.
- **Image Quality:** Variations in illumination, camera focus, and resolution can hinder accurate delineation.

These challenges underscore the importance of advanced techniques and tools in improving the precision and reliability of skin lesion segmentation.

2.2. Related work on skin lesion segmentation techniques

Researchers are actively working to overcome the hurdles above. The field of skin lesion segmentation primarily relies on two approaches:

- **Traditional Image Processing Techniques:** These methods use algorithms to analyze various image properties like color intensity and texture. While they can be effective, they often struggle with the high variability in skin lesions, limiting their accuracy and reliability.
- **Deep Learning-based Techniques:** These have revolutionized the field by leveraging Convolutional Neural Networks (CNNs). CNNs are trained on extensive datasets of labeled images, allowing them to learn complex patterns and identify subtle features that distinguish lesions from healthy skin. Due to their ability to handle the variability in lesions and their superior accuracy, deep learning approaches are considered state-of-the-art.

The initial research on combining transfer learning and fine-tuning techniques with a melanoma segmentation strategy based on U-net and LinkNet deep learning networks is found in [3]. The experiments were performed on PH2, ISIC 2018, and DermIS datasets.

The method faced limitations due to the image capture device, which affected the model's learning of disease characteristics like resolution, color, sharpness, and lighting. The authors claimed that the reproducibility of results is also limited by the diversity of skin tones across different populations.

A pyramid module incorporating lateral connections and top-down paths was used to compensate for the loss of spatial feature information [1].

This method integrated RetinaNet and MaskRCNN, with the Melanoma ISIC 2018 and PH2 datasets serving as training and validation grounds. The method's limitations included segmentation accuracy being affected by high occlusions near lesions and data class imbalance due to the absence of additional lesion data.

In [16], a fully automated multi-class skin lesion segmentation and classification approach was proposed using the most discriminant deep Learning Features and Improved Moth Flame Optimization. The proposed methodology's segmentation performance was evaluated on the ISBI 2016, ISBI 2017, ISIC 2018, and PH2 datasets. However, the computational time was one of the work's limitations.

In response to skin lesion segmentation, a novel EIU-Net method was proposed to tackle the challenging task [35]. Inverted residual blocks and an efficient pyramid squeeze attention (EPSA) block were proposed as the main encoders at different stages to capture the local and global contextual information. In contrast, atrous spatial pyramid pooling (ASPP) was utilized after the last encoder, and the soft-pool method was introduced for downsampling. Also, they proposed a novel method named multi-layer fusion (MLF) module to effectively fuse the feature distributions and capture significant boundary information of skin lesions in different encoders to improve the network's performance. Furthermore, a reshaped decoder fusion module was used to obtain multi-scale information by fusing feature maps of different decoders to improve the final results of skin lesion segmentation. To validate the performance of this network, it was compared with other methods on four public datasets, including the ISIC 2016, ISIC 2017, ISIC 2018, and PH2 datasets. Moreover, the main metric Dice Score achieved by the proposed EIU-Net are 0.919, 0.855, 0.902, and 0.916 on the four datasets; our EIU-Net can improve the accuracy of skin lesion segmentation [35].

In [23], the authors introduce a novel end-to-end trainable network for skin lesion segmentation. The proposed methodology comprises an encoder-decoder, a region-aware attention approach, and a guided loss function. The trainable parameters are reduced using depth-wise separable convolution, and the attention features are refined using a guided loss, resulting in a high Jaccard index. We assessed the effectiveness of our proposed RA-Net on four frequently utilized benchmark datasets for skin lesion segmentation: ISIC 2016, ISIC 2017, ISIC 2018, and PH2.

Integrating conventional treatment methods with deep learning frameworks to enhance skin lesion identification is proposed in [29]. The study used image data, handcrafted lesion features, and patient-centric metadata for effective skin cancer diagnosis. It combines image features transferred from Efficient Nets, color and texture information extracted from images, and pre-processed patient metadata to build a hybrid model. Each model underwent training and evaluation using the ISIC 2018 and ISIC 2019 datasets widely used for skin cancer analysis. However, a notable limitation of this

approach is the extreme imbalance in the datasets, which requires careful consideration of appropriate evaluation metrics. Despite achieving high sensitivity (90.49%) and specificity (97.76%) on the ISIC 2018 dataset, the model's performance may vary when applied to different datasets or real-world scenarios with varying data imbalance or complexity levels.

A novel deep-learning method named ChimeraNet was proposed for detecting hair and ruler marks in skin lesion images [17]. ChimeraNet employs an encoder-decoder architecture, incorporating a pre-trained EfficientNet and the decoder's squeeze-and-excitation residual (SERes) structures. However, this technique demands significant computational resources and training time due to the complexity of the encoder-decoder architecture and pre-trained models.

For accurate detection and delineation of hair in skin images, a researcher in [4] proposed a deep learning strategy based on a hybrid network of convolutional and recurrent layers for hair segmentation using weakly labeled data and deep encoded features. The spatial dependencies between disjointed patches were encoded by feeding the encoded features into recurrent neural network layers. The proposed method achieved segmentation accuracy with a Jaccard Index of 77.8 percent.

In [27], the researcher presented a machine learning-based methodology for segmenting skin lesions with novel borders and hair removal. The suggested approach removes any corner boundaries from an RGB skin picture as input. The skin hairs covering the image are then found and eliminated. The generated picture is then improved, and the GrabCut method is used to segment lesions. The research showed that the skin lesion segmentation method proposed in this paper had Jaccard indices of 0.77 and 0.80 on PH2 and ISIC 2018 datasets, respectively, and Dice indices of 0.87 and 0.82, respectively. The method failed to perform well on images with tiny affected areas. It automatically draws a rectangle around the region using the GrabCut method. However, when we deal with dermoscopic pictures with tiny lesions, initiating too big or too small rectangles for over-segmentation will occur during this method's selection process.

Segmentation accuracy degradation and occlusions in dermoscopic images constitute the significant problems identified here. High resolution and elaborate surface structures make conventional segmentation algorithms struggle with dermoscopic pictures. A mistake in segmentation accuracy may give wrong interpretations or cause detection failures, which affect the reliability of the diagnosis results. Moreover, occlusions within these images, such as those brought about by artifacts, hair follicles, and other foreign matter, block important details required during skin lesion boundary determination, leading to distortion. Therefore, it is essential to address these challenges if automated methods of segmenting medical pictures are to be effective in the field of dermatology.

Our proposed method is the Spatial SegNet model, designed based on attention mechanisms, which work well for increasing precision levels and handling occluded areas in dermoscopic images.

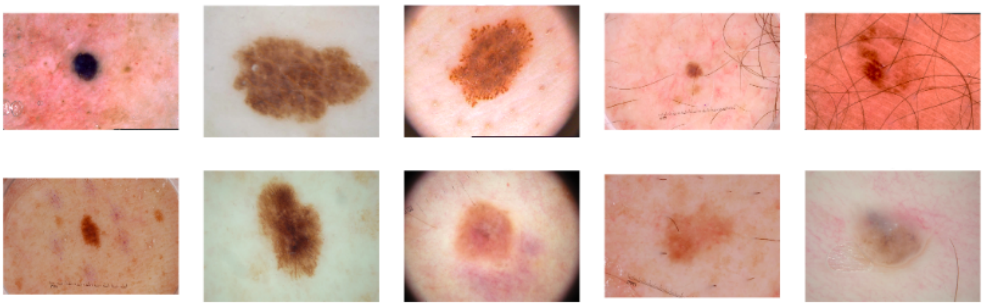

Fig. 1. Few examples of skin lesion samples in ISIC 2018.

3. Materials and methods

3.1. Data acquisition

In this research work, the ISIC 2018 dataset [7,32] was used to evaluate the results of our proposed method. The ISIC 2018 dataset is a comprehensive collection of dermoscopic images curated for skin lesion analysis. It contains 2 594 images in JPG format, each accompanied by ground truth segmentation masks. This dataset was selected for its diversity and the high-quality annotations it provides, which are essential for an accurate evaluation of segmentation methods. The images in the ISIC 2018 dataset vary significantly in lesion type and appearance, offering a robust challenge for our segmentation model. The ground truth masks serve as a benchmark for assessing the performance of our method, allowing us to measure the accuracy and effectiveness of the segmentation results quantitatively. Using this well-established dataset, we ensure our rigorous evaluation is relevant to real-world clinical scenarios. Figure 1 presents some samples of complex skin lesions in the dataset of ISIC 2018.

3.2. Proposed model

This section presents the details of our proposed method. SegNet has an encoder-decoder structure followed by a pixel-wise classification layer. The encoder architecture of SegNet is identical to VGG16's convolutional layers in topology. The decoder network maps the encoder feature maps to input resolution-sized feature maps for pixel-wise classification. In the proposed model, spatial attention layers are added to the encoder network of the SegNet architecture. In skin lesion images, spatial attention will help the model to focus on essential parts or regions of interest, thereby improving accuracy in segmentation by concentrating on relevant areas while ignoring irrelevant or noisy parts. The SegNet decoder network has several decoders organized in a hierarchy, each corresponding to

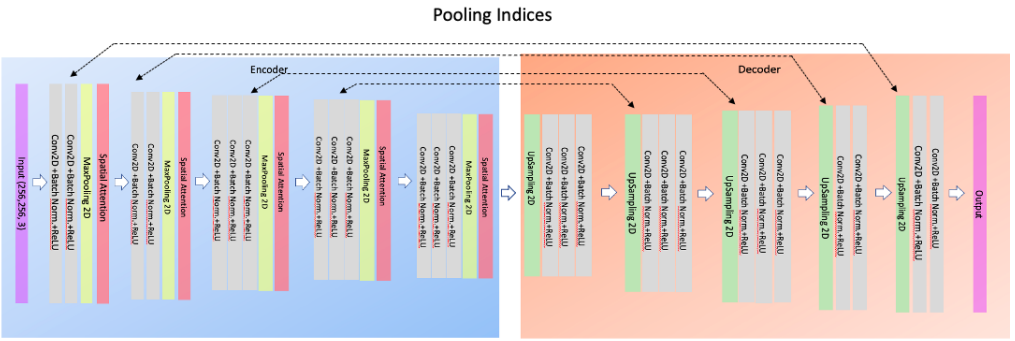

Fig. 2. SEGNET with Spatial Attention Architecture.

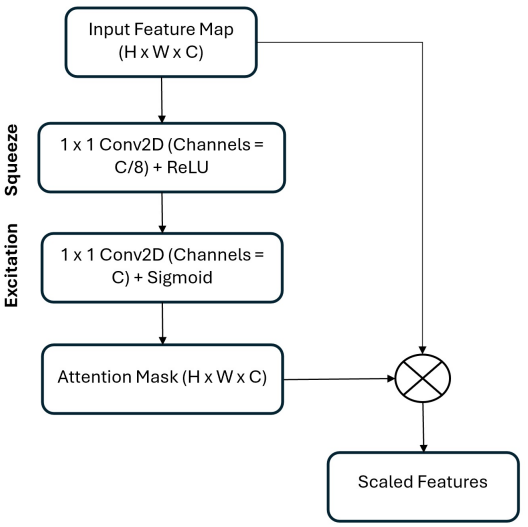

Fig. 3. Spatial Attention Module

one encoder. The correct decoders take their input feature maps and perform non-linear upsampling using max pooling indices that they receive from their respective encoders. It was derived from an architecture used for unsupervised feature learning [26]. Here are many practical advantages of reusing max-pooling indices during decoding. The architecture of SegNet with Spatial Attention is shown in Figure 2.

This model combines pre-trained VGG16 layers with spatial attention mechanisms in the encoder network. In the following subsections, every layer is explained in detail.

Input layer

An RGB image of a fixed size is an input to this layer. It is usually prepared after normalization, subtracting the mean from the image or scaling all values within a certain range. It accepts input images of size 256×256×3 (height, width, color channels).

Encoder blocks

Pre-trained VGG16 layers are used for feature extraction in the SegNet encoder network. The following layers from the pre-trained VGG16 model are used in the proposed architecture. Each convolutional block typically includes 3×3 filters with a stride of 1 and padding of 1 to extract features such as edges, textures, and color patterns. These layers apply learnable filters to the input feature maps. The filters essentially slide across the input, extracting features such as edges, textures, and color patterns. The number of filters used determines the complexity and richness of the extracted features. The activation function ReLU is applied after the convolutional layers (Conv) to introduce nonlinearity and allow the model to capture more complex relationships in the data. The batch normalization layers normalize the activations of the previous convolution layer. It facilitates faster convergence during training and enhances the stability of the learning process. The batch norm essentially standardizes the activations across different mini-batches, mitigating the issue of internal covariate shift. The pooling layers downsample the feature maps spatially using max pooling. It also makes the model robust to translations that occur very close together, while simultaneously making it less sensitive to noise because features become more generalized.

A Spatial Attention Block is a custom module added to the model to improve the segmentation of skin lesions (Figure 3). It acts on feature maps encoded in the previous convolutional block borrowed from the pre-trained VGG16 model. Specifically, it is inserted after every pooling layer within the VGG16 encoder. Its main objective is to enhance the encoded features and focus on them. Details of each spatial attention block are explained as follows:

- **Squeeze Operation:** First, feature maps are made smaller through a 1x1 convolution. This step aims to reduce the dimensionality of the space and the computational cost incurred by modulating units to learn their interaction.
- **Excitation Operation:** The method spins a spatial attention map to identify vital skin detection areas. It is achieved by applying another 1x1 convolution operation and performing a sigmoid activation function. The resultant map assigns different values between 0 and 1 for each part of an image, where a value of zero means least significant and a value of one corresponds to the most significant pixels, thereby highlighting those regions necessary for segmentation through information obtained from prior layers.
- **Element-wise Multiplication:** The last part includes element-wise multiplication of the initial feature maps with the produced spatial attention map. By doing this,

characteristics identified as important by our attention mechanism are emphasized, thus enabling the model to concentrate on particular parts during its segmenting duties. Its intensified focus strengthens the model's ability to distinguish between unhealthy cells and their surrounding healthy tissues.

Further elaborating on the workings of the Spatial Attention Block, the first Conv2D layer squeezes the feature maps by decreasing the channels or properties in this module. For example, if the entry feature maps have 512 channels, the squeezing layer might bring this down to a smaller amount, like 64 channels. Then, using the sigmoid activation function, the second Conv2D generates attention weights representing a probability distribution that shows the importance of different spatial locations within the given feature maps. After that, these produced attention weights are multiplied with original features so that some features can be amplified or suppressed selectively based on their importance towards achieving the segmentation goal. Thus, resulting scaled feature maps will center around crucial areas, helping the model capture fine details and semantics necessary for accurate segmentation.

Decoder layers (segmentation mask reconstruction)

The decoder part takes the encoded feature maps obtained from the encoder along with the spatial attention. It has a symmetric structure relative to the encoder and progressively increases the resolution of feature maps through transpose convolution operations. Convolution layers are attached after these upsampling operations to refine features and learn spatial relationships between pixels. Unlike its counterpart, which extracts them, this one aims to recover spatial information while predicting probabilities for individual pixels to be part of skin lesions. For example, (background versus lesion) background versus lesion class probability maps may be obtained by applying the softmax activation function to the final output layer on a class basis. These layers receive processed features, including effects caused by spatial attention blocks within the encoder, and then gradually reconstruct an image that focuses on the segmentation task. They function oppositely from encoders, i.e., starting with a high-level understanding of the picture and adding more detailed spatial information stepwise downwards towards the lowest level segmentation features being dealt with at every decoder block stage. Each block typically consists of :

- **Upsampling 2D:** It Increases feature maps' spatial resolution by this layer. Unlike the traditional Conv layer, this layer learns upsampled filters that expand the feature maps while introducing new spatial information. It allows the model to recover spatial details lost during pooling in the encoder.
- **Convolutional layers:** After Upsampling layers, Conv layers are applied to refine the features and learn the relationships between pixels similar to the encoder. They help to combine upsampled information with high-level features from the encoder.

- **Batch Normalization layers (Batch Norm):** These are used for normalizing activations after upsampling, similar to the encoder, for better training stability.

Output layer

The final decoder output is fed to a softmax classifier layer to produce the class probabilities. The softmax function produces a probability map for every pixel in the image. This map shows how likely each pixel is to belong to a specific class (e.g., background or skin lesion). The class with the highest probability for each pixel becomes the predicted segmentation label.

Spatial attention mechanisms are incorporated into models designed for skin lesion segmentation, improving the overall performance and reliability of the proposed model. This technique combines pre-trained features, spatial attention mechanisms, and SegNet's decoder architecture to achieve accurate skin lesion segmentation. Pre-trained VGG16 weights extract essential image features more effectively, reducing training time and enhancing its generalization ability over new data. Considering different skin lesion sizes and appearances, introducing a spatial attention block narrows down the essential parts of an image, thus leading to precise skin lesion segmentation.

3.3. Evaluation metrics

To assess the performance of our proposed skin lesion segmentation method, we employed a variety of evaluation metrics that provide a comprehensive analysis of segmentation accuracy and quality. The metrics used in this study include the Dice Coefficient and Binary Accuracy.

TP and FP refer to lesion pixels extracted as lesion pixels and non-lesion pixels extracted as lesion pixels, respectively. At the same time, FN and TN represent lesion pixels extracted as non-lesion pixels and non-lesion pixels extracted as non-lesion pixels, respectively.

Dice Coefficient

The Dice Coefficient is an essential metric for evaluating segmentation quality. It is calculated as the ratio of twice the area of overlap between the predicted and ground truth masks to the sum of the areas of both masks. The Dice Coefficient ranges from 0 to 1, with a value closer to 1 indicating better segmentation accuracy. This metric is beneficial for handling class imbalance, as it emphasizes the correct prediction of positive samples. The dice similarity coefficient is a spatial overlap index and a reproducibility validation metric, and it computes the similarity index between the given images.

$$\text{Dice Coefficient} = \frac{2\text{TP}}{(\text{FP} + \text{TP}) + (\text{TP} + \text{FN})}\,. \tag{1}$$

Accuracy

Accuracy refers to the proportion of correctly predicted pixels (lesion and non-lesion) out of the total number of pixels. It is calculated as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} . \tag{2}$$

Precision

Precision refers to the proportion of true positive predictions among all the pixels predicted as lesions. It indicates the model's accuracy in identifying the lesion pixels out of all the pixels it labeled as lesions. Precision is calculated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} . \tag{3}$$

Sensitivity

Sensitivity, also called Recall, measures the proportion of actual positives (lesions) the model correctly identifies. It indicates the model's ability to detect the lesion pixels

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} . \tag{4}$$

Specificity

Specificity measures the proportion of actual negatives (non-lesions) the model correctly identifies. It indicates the model's ability to avoid false positives.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} . \tag{5}$$

F1 Score

F_1 Score is the harmonic mean of Precision and Recall (Sensitivity). It is a balanced measure that considers both false positives and false negatives.

$$F_1 = \frac{2(\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} . \tag{6}$$

IOU

IOU is used to measure the overlap between two images.

$$\text{IOU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} . \tag{7}$$

4. Experimental results

An open-source machine learning framework, TensorFlow, implements the methodology. It is a user-friendly interface for working with deep neural networks, designed for ease of use rather than machine-level interactions. It is a library mainly used for developing real-time computer vision applications.

- **CPU Resources**:
  - Environment 1: 256MB memory limit on device `/device:CPU:0`.
  - Environment 2: XLA CPU with 16GB memory limit on device `/device:XLA_CPU:0`.
- **GPU Resources**:
  - Tesla T4 GPUs: Two GPUs with 14.8GB memory each, identified as `/device:GPU:0` and `/device:GPU:1`, with PCI Bus IDs `0000:00:04.0` and `0000:00:05.0`, respectively. Both GPUs have Compute Capability 7.5.
  - XLA GPUs: Two GPUs with 16GB memory each, denoted as `/device:XLA_GPU:0` and `/device:XLA_GPU:1`.

This combination of hardware configurations provided the computational capacity necessary for efficient training and testing of deep learning models, enabling the handling of large-scale data processing and complex model architectures.

4.1. Hyperparameters

To achieve a skin lesion segmentation model, hyperparameters shown in Table 1 are chosen to optimize performance and manage computational resources effectively. A learning rate of 5×10^{-6} is important as it allows for small steps to be taken by the optimizer while minimizing the loss function. It ensures the model converges slowly and steadily without overshooting the minimum loss function. Batch size 8 strikes a balance between memory efficiency and accurate gradient estimation.

To validate our skin lesion segmentation method, we carried out a 5-fold cross-validation. The steps involved partitioning a dataset into five equal sets, training on four, and validating against the fifth set. The results are shown in Table 2. The mean of the five-fold validation results is given in the last row of the table. The results show

Tab. 1. Hyperparameters for the proposed model.

Parameter Name	Parameter Value
Learning Rate	5×10^{-6}
Batch Size	8
Input Size	256, 256, 3
Optimizer	Adam Optimizer
Epoch	60

Tab. 2. Results of the proposed model with 5-fold cross validation (STD: standard deviation).

Folds	IoU	Dice Coefficient	Precision	Sensitivity	Specificity	Accuracy
1	0.8026	0.8980	0.8969	0.9086	0.9718	0.9657
2	0.8240	0.9242	0.9163	0.8969	0.9744	0.9616
3	0.8026	0.9051	0.9272	0.8890	0.9820	0.9611
4	0.8240	0.9022	0.9086	0.8725	0.9718	0.9631
5	0.8026	0.8965	0.9310	0.8619	0.9820	0.9611
Mean	**0.8111**	**0.9052**	**0.9160**	**0.8857**	**0.9764**	**0.96252**
STD	**0.0092**	**0.0045**	**0.0155**	**0.0172**	**0.0053**	**0.0026**

a high value of the Dice coefficient (0.9052) and segmentation accuracy (96%). The sensitivity of the proposed model is 0.8857.

Table 3 compares the results of our proposed model against state-of-the-art published results using the ISIC 2018 dataset. Our proposed model of skin lesion segmentation, tested on the ISIC 2018 dataset, shows significant improvements in segmentation. The primary comparison tools used to judge the outcomes are the Dice Coefficient and Binary Accuracy. The high value of the Dice Coefficient shows more similarity of the predicted results with the ground truth mask.

Adding spatial attention to the SegNet architecture achieves better skin lesion segmentation results. Spatial attention assigns weights to each pixel, highlighting the areas of interest and allowing the model to distinguish between lesion and non-lesion regions. This improves the segmentation accuracy as the model can focus on the spatial locations with features relevant to skin lesions, such as irregular shapes and varying pigmentation over background noise.

The SegNet with spatial attention model quantitatively improved the Dice similarity coefficient, IoU, and accuracy scores. These improvements are significant compared to the other segmentation models (Table 3). Figure 4 shows skin lesion segmentation results. The segmentation output looks better in the Figure 4, and lesion boundaries are more precise and consistent.

Tab. 3. Comparative analysis with state-of-the-art techniques.

Model	Dataset Split	Parameters [10^6]	Accuracy	Dice Coefficient
TMU Net [5]	70% training, 10% validation, and 20% testing	–	0.9603	0.905
UNeXt [33]	80% training, 20% testing	1.47	0.9586	0.8873
FAT-Net [34]	70% training, 10% validation, and 20% testing	30	0.9578	0.8903
CPFNET [10]	5-fold cross-validation	43	0.9496	0.8769
DAGAN [18]	2296 images for training, 300 images for testing.	54	0.9324	0.87707
CKDNet [15]	–	51	0.9492	0.8779
REDAUNet [20]	70% training, 10% validation, and 20% testing	47.77	0.9444	0.902
SA SegNet (Ours)	Five-Folds Cross-Validation.	29.6	**0.9625**	**0.9052**

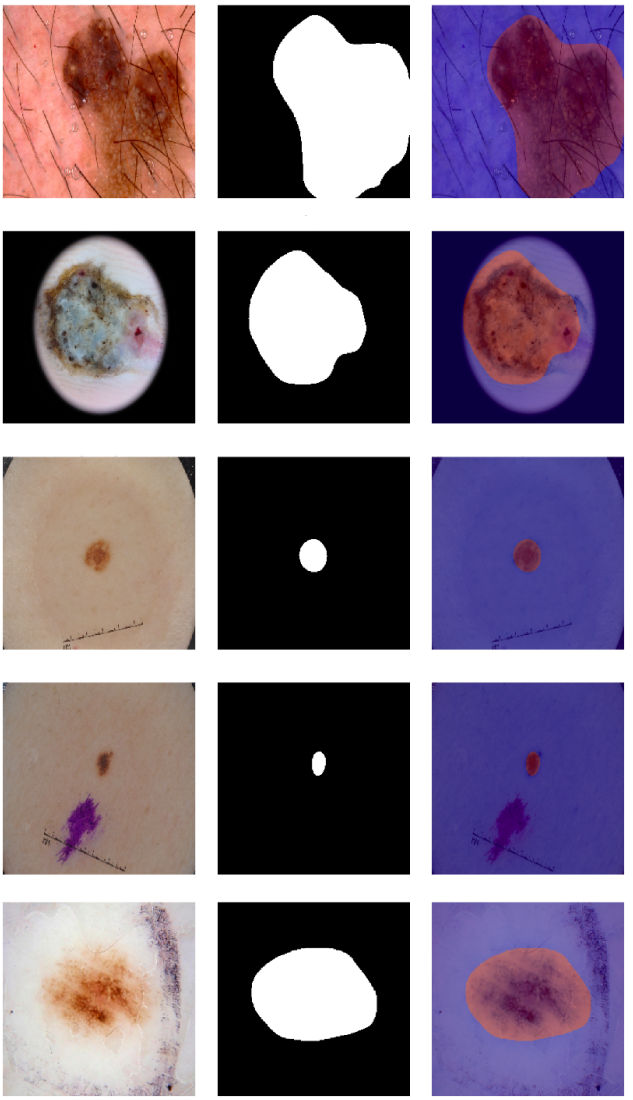

Fig. 4. Some of the segmented images. Vertically: five cases. Horizontally: (**a**) original image; (**b**) predicted mask, Dice = 0.85; (**c**) overlay image.

4.2. Impact of batch size and learning rate

In this experiment, we analyzed the influence of hyperparameters on the model performance. This analysis aims to understand the effect of batch size and learning rate on the Dice Coefficient and the Accuracy metrics. In these experiments, the whole dataset is divided in the ratio of 0.7:0.1:0.2 for training, validation, and testing, respectively.

The Table 4 presents results for the model at various initial learning rates and their effect on the Dice Coefficient and Accuracy. The learning rate of 1×10^{-6} is too low for the model to learn efficiently, as it yields the lowest performance, with a Dice coefficient of 0.8611 and an accuracy of 0.9395. A learning rate of 1×10^{-5} performs the best with the highest value of the Dice coefficient of 0.9053, an accuracy of 0.9626. Thus, it infers that this is the ideal rate of learning and generalization. If the learning rate is increased to 1×10^{-4}, the performance decreases slightly, as the Dice coefficient goes to 0.8794, and an accuracy of 0.9527 is achieved. This shows that although the model performs well, the learning rate is too large for optimal training. As the learning rate is increased to 1×10^{-3}, the model training is the worst, with a Dice coefficient of 0.8761 and an accuracy of 0.9436 on the testing dataset. It may indicate that the model is converging too fast and missing some finer details in the data. The learning rate of 1×10^{-5} is the most effective, being the best in segmentation and classification tasks, while increasing or decreasing the learning rate worsens the performance.

Batch size determines the number of samples in the training dataset to update the parameters. Increasing the batch size means fewer weight updates in an epoch. Hence, memory and computational requirements are lower for smaller batch sizes due to the smaller number of samples per update. However, for smaller batch sizes, the effect of noise and variance of the loss gradient will be more on the weight updates of the model. The Table 5 compares the model's performance across different batch sizes in

Tab. 4. Comparison of Dice Coefficient and Accuracy for different initial learning rates

Initial Learning Rate	Dice Coefficient	Accuracy
1×10^{-6}	0.8611	0.9395
1×10^{-5} **(Ours)**	0.9053	0.9626
1×10^{-4}	0.8794	0.9527
1×10^{-3}	0.8761	0.9436

Tab. 5. Comparison of model performance for different batch sizes.

Batch Size	Test Dice Coefficient	Test Accuracy
4	0.9005	0.9578
8 (Ours)	0.9053	0.9626
12	0.8967	0.9570
16	0.8733	0.9509

Tab. 6. Comparison of models based on model's variations.

Model	Test Dice Coefficient	Test Accuracy
SegNet only	0.8977	0.9581
Partial Removal of Spatial Attention	0.8986	0.9561
Removal of Batch Normalization	0.8827	0.9505
Spatial Attention SegNet (Ours)	0.9052	0.9625

terms of Dice Coefficient and Accuracy. Comparing the dice coefficient and accuracy for various batch sizes, a batch size of eight is optimal. The model performs best with a Dice coefficient of 0.9053 and an accuracy of 0.9626. The performance degrades as the batch size increases from eight, and the dice coefficient and accuracy decrease. Finally, with the batch size of 16, the performance significantly drops (Dice coefficient of 0.8733 and accuracy of 0.9509), which implies that the bigger batch sizes may preclude the model's ability to converge effectively and even generalize well. The overall results, however, indicate that batch size eight is more likely to bring equilibrium to the model's computing efficiency and practical utility. Therefore, it is the most suitable option for this experiment.

4.3. Ablation experiments

In this section, we perform an ablation study on the proposed model. We have studied the effect of the spatial attention layer and batch normalization layer.

The Table 6 showcases the comparison of the performances of the different versions of the models. The core of the system is the SegNet architecture, and performance can be greatly enhanced by the introduction of some components, like batch normalization and spatial attention. The spatial attention technique is a method for improving segmentation accuracy, which allows the model to focus on relevant areas of the input.In our model we have included another highly significant layer, which is called batch normalization (BN). By doing BN the input to each layer, the result is the stabilization of the process of learning and reduction of internal co-variate shifts. An ablation study shows that when batch normalization is taken away, both the Dice coefficient and the accuracy fall drastically. The removal of batch normalization from the model greatly decreases the model's accuracy, thus proving its importance in ensuring a successful learning period.

Our model, proposed in this paper, combines spatial attention and batch normalization layers. Both layers in the model provides best Dice coefficient of 0.9052 and the highest accuracy of 0.9625. Hence, it is clear that spatial attention helps the model pick the salient parts of the image, while batch normalization ensures the model's training

runs smoothly and can generalize well, thereby enhancing both segmentation and classification performance. The implementing these layers is essential for the robustness, accuracy, and capacity to deal with the complexity of the patterns in the data.

5. Conclusion and future work

This paper provides a detailed framework for the proposed skin lesion Segmentation model. Our proposed approach uses SegNet architecture combined with spatial attention layers. Encoder layers are taken from the pre-trained model of VGG16. Our model showed better segmentation accuracy and improved lesion boundary delineation precision. It is evident from Table 3 that the proposed model performed better on the ISIC 2018 dataset than other published state-of-the-art models. We have achieved a dice coefficient of **0.9052** and a segmentation accuracy of **0.9625**.

An important aspect of future work is the incorporation of multimodal data. Combining dermoscopic images with clinical information provides a more holistic approach to analyzing skin lesions; thus, this approach may increase diagnostic performance by improving segmentation accuracy. This kind of approach uses different data strengths to give a more precise and reliable diagnosis. It is also essential to build real-time segmentation systems for clinical purposes. These systems need to work efficiently on edge devices or mobile platforms to be accessible for use in different clinical environments.

References

[1] N. Ahmed, X. Tan, and L. Ma. A new method proposed to melanoma-skin cancer lesion detection and segmentation based on hybrid convolutional neural network. *Multimedia Tools and Applications* 82(8):11873–11896, 2023. doi:10.1007/s11042-022-13618-0.

[2] Z. Apalla, A. Lallas, E. Sotiriou, E. Lazaridou, and D. Ioannides. Epidemiological trends in skin cancer. *Dermatology Practical & Conceptual* 7(2):1, 2017. doi:10.5826/dpc.0702a01.

[3] R. L. Araújo, F. H. D. de Araújo, and R. R. V. e. Silva. Automatic segmentation of melanoma skin cancer using transfer learning and fine-tuning. *Multimedia Systems* 28(4):1239–1250, 2022. doi:10.1007/s00530-021-00840-3.

[4] M. Attia, M. Hossny, H. Zhou, S. Nahavandi, H. Asadi, et al. Digital hair segmentation using hybrid convolutional and recurrent neural networks architecture. *Computer methods and programs in biomedicine* 177:17–30, 2019. doi:10.1016/j.cmpb.2019.05.010.

[5] R. Azad, M. Heidari, Y. Wu, and D. Merhof. Contextual Attention Network: Transformer meets U-Net. In: *International Workshop on Machine Learning in Medical Imaging (MLMI 2022)*, vol. 13583 of *Lecture Notes in Computer Science*, pp. 377–386. Springer, 2022. doi:10.1007/978-3-031-21014-3_39.

[6] Canadian Skin Cance Foundation. Skin cancer early detection. In: Skin Cancer, 2024. `https://www.canadianskincancerfoundation.com/early-detection/`. [Accessed: 2024-11-05].

[7] N. Codella, V. Rotemberg, P. Tschandl, M. E. Celebi, S. Dusza, et al. Skin lesion analysis toward

melanoma detection 2018: A challenge hosted by the International Skin Imaging Collaboration (ISIC). arXiv, arXiv:1902.03368, 2019. doi:10.48550/arXiv.1902.03368.

[8] J. Dinnes, J. J. Deeks, M. J. Grainge, N. Chuchu, L. Ferrante di Ruffano, et al. Visual inspection for diagnosing cutaneous melanoma in adults. *Cochrane Database of Systematic Reviews* 2018(12), 1996. doi:10.1002/14651858.CD013194.

[9] H. C. Engasser and E. M. Warshaw. Dermatoscopy use by US dermatologists: A cross-sectional survey. *Journal of the American Academy of Dermatology* 63(3):412–419, 2010. doi:10.1016/j.jaad.2009.09.050.

[10] S. Feng, H. Zhao, F. Shi, X. Cheng, M. Wang, et al. CPFNet: Context pyramid fusion network for medical image segmentation. *IEEE Transactions on Medical Imaging* 39(10):3008–3018, 2020. doi:10.1109/TMI.2020.2983721.

[11] Z. Ge, S. Demyanov, R. Chakravorty, A. Bowling, and R. Garnavi. Skin disease recognition using deep saliency features and multimodal learning of dermoscopy and clinical images. In: *Proc. 20th Int. Conf. Medical Image Computing and Computer Assisted Intervention (MICCAI 2017)*, vol. 10435 part III of *Lecture Notes in Computer Science*, pp. 250–258. Springer, 2017. doi:10.1007/978-3-319-66179-7_29.

[12] K. Hauser, A. Kurz, S. Haggenmüller, R. C. Maron, C. von Kalle, et al. Explainable artificial intelligence in skin cancer recognition: A systematic review. *European Journal of Cancer* 167:54–69, 2022. doi:10.1016/j.ejca.2022.02.025.

[13] W. Hu, L. Fang, R. Ni, H. Zhang, and G. Pan. Changing trends in the disease burden of non-melanoma skin cancer globally from 1990 to 2019 and its predicted level in 25 years. *BMC cancer* 22(1):836, 2022. doi:10.1186/s12885-022-09940-3.

[14] IARC. Skin cancer. In: International Agency for Research on Cancer (IARC), WHO, 2024. `https://www.iarc.who.int/cancer-type/skin-cancer/`. [Accessed: 05-11-2024].

[15] Q. Jin, H. Cui, C. Sun, Z. Meng, and R. Su. Cascade knowledge diffusion network for skin lesion diagnosis and segmentation. *Applied Soft Computing* 99:106881, 2021. doi:10.1016/j.asoc.2020.106881.

[16] M. A. Khan, M. Sharif, T. Akram, R. Damaševičius, and R. Maskeliūnas. Skin lesion segmentation and multiclass classification using deep learning features and improved moth flame optimization. *Diagnostics* 11(5):811, 2021. doi:10.3390/diagnostics11050811.

[17] N. Lama, R. Kasmi, J. R. Hagerty, R. J. Stanley, R. Young, et al. ChimeraNet: U-Net for hair detection in dermoscopic skin lesion images. *Journal of Digital Imaging* 36(2):526–535, 2023. doi:10.1007/s10278-022-00740-6.

[18] B. Lei, Z. Xia, F. Jiang, X. Jiang, Z. Ge, et al. Skin lesion segmentation via generative adversarial networks with dual discriminators. *Medical Image Analysis* 64:101716, 2020. doi:10.1016/j.media.2020.101716.

[19] W. Li, A. N. J. Raj, T. Tjahjadi, and Z. Zhuang. Digital hair removal by deep learning for skin lesion segmentation. *Pattern Recognition* 117:107994, 2021. doi:10.1016/j.patcog.2021.107994.

[20] L. Liu, X. Zhang, Y. Li, and Z. Xu. An improved multi-scale feature fusion for skin lesion segmentation. *Applied Sciences* 13(14), 2023. doi:10.3390/app13148512.

[21] P. A. Lyakhov, U. A. Lyakhova, and D. I. Kalita. Multimodal analysis of unbalanced dermatological data for skin cancer recognition. *IEEE Access* 11:131487–131507, 2023. doi:10.1109/ACCESS.2023.3336289.

[22] R. Maurya, N. N. Pandey, M. K. Dutta, and M. Karnati. FCCS-Net: Breast cancer classification using Multi-Level fully Convolutional-Channel and spatial attention-based transfer learning approach. *Biomedical Signal Processing and Control* 94:106258, 2024. doi:10.1016/j.bspc.2024.106258.

[23] A. Naveed, S. S. Naqvi, S. Iqbal, I. Razzak, H. A. Khan, et al. RA-Net: Region-Aware attention Network for skin lesion segmentation. *Cognitive Computation* 16:2279–2296, 2024. doi:10.1007/s12559-024-10304-1.

[24] S. Rajpar and J. Marsden. *ABC of Skin Cancer*. John Wiley & Sons, 2009.

[25] G. Ramella. Hair removal combining saliency, shape and color. *Applied Sciences* 11(1):447, 2021. doi:10.3390/app11010447.

[26] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE, 2007. doi:10.1109/CVPR.2007.383157.

[27] M. Rehman, M. Ali, M. Obayya, J. Asghar, L. Hussain, et al. Machine learning based skin lesion segmentation method with novel borders and hair removal techniques. *Plos one* 17(11):e0275781, 2022. doi:10.1371/journal.pone.0275781.

[28] M. A. Richard, C. Paul, T. Nijsten, P. Gisondi, C. Salavastru, et al. Prevalence of most common skin diseases in Europe: a population-based study. *Journal of the European Academy of Dermatology and Venereology* 36(7):1088–1096, 2022. doi:10.1111/jdv.18050.

[29] M. Sharafudeen and S. S. V. Chandra. Detecting skin lesions fusing handcrafted features in image network ensembles. *Multimedia Tools and Applications* 82:3155–3175, 2022. doi:0.1007/s11042-022-13046-0.

[30] S. Tehsin, I. M. Nasir, R. Damaševičius, and R. Maskeliūnas. DaSAM: Disease and spatial attention module-based explainable model for brain tumor detection. *Big Data and Cognitive Computing* 8(9), 2024. doi:10.3390/bdcc8090097.

[31] The American Cancer Society medical and editorial content team. Key Statistics for Melanoma Skin Cancer. In: Melanoma Skin Cancer, 2025. `https://www.cancer.net/cancer-types/melanoma/statistics`. [Accessed: 2024-11-05].

[32] The International Skin Imaging Collaboration. ISIC CHALLENGE, 2018-2024. `https://challenge.isic-archive.com/data/`.

[33] J. M. J. Valanarasu and V. M. Patel. UNeXt: MLP-based rapid medical image segmentation network. In: *Medical Image Computing and Computer Assisted Intervention (MICCAI 2022)*, vol. 13435 of *Lecture Notes in Computer Science*, pp. 23–33. Springer Nature Switzerland, 2022. doi:10.1007/978-3-031-16443-9_3.

[34] H. Wu, S. Chen, G. Chen, W. Wang, B. Lei, et al. FAT-Net: Feature adaptive transformers for automated skin lesion segmentation. *Medical Image Analysis* 76:102327, 2022. doi:https://doi.org/10.1016/j.media.2021.102327.

[35] Z. Yu, L. Yu, W. Zheng, and S. Wang. EIU-Net: Enhanced feature extraction and improved skip connections in U-Net for skin lesion segmentation. *Computers in Biology and Medicine* 162:107081, 2023. doi:10.1016/j.compbiomed.2023.107081.

[36] Y. Zhong, Z. Shi, Y. Zhang, Y. Zhang, and H. Li. CSAN-UNet: Channel spatial attention nested UNet for infrared small target detection. *Remote Sensing* 16(11):1894, 2024. doi:10.3390/rs16111894.

Perceptually Optimised Swin-Unet for Low-Light Image Enhancement

Tomasz M. Lehmann* and Przemysław Rokita
Warsaw University of Technology, Warsaw, Poland
*Corresponding author: Tomasz M. Lehmann (tomasz.lehmann.dokt@pw.edu.pl)

Submitted: May 27, 2025 **Accepted:** Oct 12, 2025 **Published:** Nov 12, 2025

Abstract In this paper we propose a novel approach to low-light image enhancement using a transformer-based Swin-Unet and a perceptually driven loss that incorporates Learned Perceptual Image Patch Similarity (LPIPS), a deep-feature distance aligned with human visual judgements.

Specifically, our U-shaped Swin-Unet applies shifted-window self-attention across scales with skip connections and multi-scale fusion, mapping a low-light RGB image to its enhanced version in one pass. Training uses a compact objective – Smooth-L_1, LPIPS (AlexNet), MS-SSIM (detached), inverted PSNR, channel-wise colour consistency, and Sobel-gradient terms – with a small LPIPS weight chosen via ablation.

Our work addresses the limits of purely pixel-wise losses by integrating perceptual and structural components to produce visually superior results. Experiments on LOL-v1, LOL-v2, and SID show that while our Swin-Unet does not surpass current state-of-the-art on standard metrics, the LPIPS-based loss significantly improves perceptual quality and visual fidelity.

These results confirm the viability of transformer-based U-Net architectures for low-light enhancement, particularly in resource-constrained settings, and suggest exploring larger variants and further tuning of loss parameters in future work.

Keywords: low-light image enhancement, U-Net, mean opinion score, LPIPS.

1. Introduction

As shown below, numerous software frameworks, models, and methodologies have been proposed for the low-light enhancement task. Nevertheless, we extend this research by examining three persistent gaps – architecture, efficiency, and perception. Pure transformer U-Nets such as Swin-Unet [3] have been scarcely explored in this context, yet their hierarchical shifted-window attention is well suited to the joint global–local reasoning required by complex illumination. Moreover, state-of-the-art models almost exclusively optimise pixel-level errors, which correlate poorly with human judgement; colour shifts and texture flattening therefore persist. A composite loss that blends classic terms with a perceptual metric (LPIPS) [48] is needed to align optimisation with visual quality. In addition, many high-performing pipelines rely on heavy diffusion stages or multi-branch designs, whereas a lightweight, single-stage Swin-Unet promises a superior accuracy-efficiency trade-off – crucial for real-time or mobile applications.

These observations motivate our investigation of a perceptually optimised Swin-Unet

that couples the representational power of hierarchical transformers with an LPIPS-augmented composite loss, aiming to reduce residual artefacts while retaining computational frugality.

1.1. Related Works

Enhancing photographs captured in severe darkness has matured from handcrafted tone-mappers to sophisticated learning pipelines, yet every generation still negotiates its own trade-offs between fidelity, robustness, and speed. Early grey-level transformations and Retinex-based formulations [9, 10, 13, 14, 17, 27, 44] adjust global brightness through fixed, analytical rules that remain attractive for real-time use but inevitably falter when illumination varies across a scene, leaving local noise and colour bias unresolved. Retinex theory itself – explicitly separating reflectance from illumination – continues to underpin most modern networks: Retinex-Net [37] dissects, corrects, and re-merges the two layers in three consecutive modules, achieving joint denoising and brightening, although its separate branches occasionally amplify artefacts if any module under-fits. Diff-Retinex [43] replaces convolutions with Transformer Decomposition Networks (TDN) and diffusion-style adjusters that offer smoother global illumination at the cost of substantial inference latency introduced by the diffusion iterations. Alternative encoder–decoder designs regress a coarse illumination map and refine it in a single pass; their simplicity improves throughput but risks oversmoothing high-frequency detail. Two-stream recurrent models mitigate this blur by letting a secondary branch track salient textures, yet the recurrent roll-out lengthens both memory use and training time.

To preserve the fine structure of the image, in the subsequent work the multi-scale processing and attention was introduced. Unrolled optimisation with residual blocks and parallel multi-resolution streams [19, 45] retains context over very large receptive fields, but the extra resolution hierarchy enlarges GPU memory consumption. CDAN [31] adds dense connectivity and channel-attention to a U-Net skeleton, improving colour consistency and perceptual sharpness while inflating parameter count. SNR-aware attention [40] and residual dense attention units [50] explicitly weight features by estimated noise statistics, reducing information loss on consumer cameras, yet the reliance on a reliable SNR estimate can degrade accuracy when sensor characteristics change. Laplacian-pyramid diffusion in PyDiff [52] progressively samples higher resolutions so as to suppress global RGB shifts with fewer parameters than classic diffusion; nevertheless, its iterative denoiser remains too heavy for battery-powered hardware.

The field is therefore witnessing a parallel push toward lightweight yet perceptually solid designs. LYT-Net [1] splits the Y and UV channels into separate paths with a Channel-Wise Denoiser and a ViT-based fusion block, reaching mobile-class throughput; its dependence on an explicit YUV conversion, however, complicates end-to-end RAW processing pipelines. Self-DACE [38] alternates Adaptive Adjustment Curves with a

CNN-based denoiser in a two-stage loop and learns solely from unpaired data, generalising across cameras while effectively doubling runtime. Other lightweight attempts compress feature maps aggressively but tend to underperform on real photographs where noise, colour cast, and motion blur co-occur.

Collectively, these developments yield a toolbox that can brighten images, suppress grain, and restore colour, yet three persistent challenges remain. First, colour distortion survives in regions where statistical priors deviate from the true illumination spectrum. Second, texture fidelity still drops whenever a network relies exclusively on pixel-wise losses such as L_1 or MSE, encouraging overly smooth outputs. Third, computational overhead – either from deep cascades, recurrent loops, or diffusion steps – prevents many state-of-the-art models from running interactively on edge devices.

Transformers equipped with windowed self-attention offer a plausible route toward closing these gaps. The Swin Transformer family [21] combines convolution-like locality with long-range context in a hierarchical fashion that scales linearly with image size, and thus promises a more favourable accuracy–efficiency balance than global-attention ViTs. Embedding Swin blocks in an encoder–decoder topology inherits the strong reconstruction ability of U-Nets while eliminating the multi-branch overhead common in Retinex cascades or the multi-step burden of diffusion. Such a design can devote its full capacity to suppressing colour shifts and preserving texture within a single pass, potentially delivering competitive perceptual quality at a fraction of the compute budget. The present work therefore positions a Swin-based U-Net at the centre of the low-light enhancement landscape, evaluating it against both heavyweight perceptual optimisers and recent lightweight specialists, and highlighting where transformer attention can bridge the longstanding trade-off between fidelity, robustness, and real-time performance.

2. Experimental setup

2.1. Datasets

To comprehensively evaluate our proposed method for low-light image enhancement, we utilized two prominent benchmark datasets specifically designed for addressing challenges associated with underexposed photography: the LOL and SID datasets. These datasets provide paired low-light and normal-light images, enabling supervised learning and detailed performance assessments. Additionally, to determine the most effective approach to data integration, we explored various dataset combinations, consistently using LOL for training, while systematically varying the inclusion and selection strategy of SID images (single darkest, three darkest, random selection, or none).

2.1.1. LOL Dataset

The LOL dataset [37] consists of pairs of images captured under low-light and normal-light conditions, primarily designed to support research focused on image enhancement

techniques. It includes 500 image pairs, of which 485 are used for training and 15 for testing. Most images in this dataset depict indoor scenes and maintain a uniform resolution of 400×600 pixels. Additionally, we employed an expanded version, known as LOL-v2, which provides 689 training and 100 testing image pairs. LOL-v2 notably enhances dataset variability by incorporating both synthetic and real-world low-light scenarios, allowing for more robust evaluations of algorithmic performance under diverse conditions.

2.1.2. SID Dataset

The See-in-the-Dark (SID) dataset [4] is a comprehensive collection of raw, short-exposure images accompanied by corresponding long-exposure reference images, tailored specifically for low-light enhancement studies. It comprises 5094 image pairs captured under various illumination conditions using two different professional-grade camera systems. This dataset uniquely offers multiple exposure levels per scene, providing valuable insights into the effectiveness of enhancement methods across varying degrees of darkness. In our experiments, we specifically evaluated multiple strategies for incorporating SID data into the training process. These strategies included selecting only the darkest exposure per scene, the three darkest exposures, random exposure selection, and excluding SID data entirely. This allowed us to rigorously investigate the impact of different dataset configurations on model performance and generalizability.

2.2. Proposed method

The goal of this work is to investigate whether a carefully tuned and loss-optimised lightweight architecture based on Swin-Unet [3] can achieve performance competitive with current state-of-the-art models for low-light image enhancement. In contrast to many recent approaches that incorporate multiple complex modules or multi-stage designs [1, 31, 52], we focus on a streamlined and efficient model that leverages the global context modelling capabilities of Vision Transformers while maintaining the desirable properties of U-Net's encoder-decoder structure.

We hypothesize that, with the right combination of architectural design and a composite loss function tailored to perceptual and structural fidelity, a pure transformer-based model can deliver good results on both synthetic and real-world low-light datasets.

2.2.1. Model Architecture

Our proposed model builds upon Swin-Unet [3], a pure Transformer architecture originally developed for medical image segmentation. The architecture follows a symmetric U-shaped design composed entirely of Swin Transformer blocks [21], organized into an encoder, bottleneck, and decoder, interconnected through skip connections.

The encoder consists of a patch embedding layer followed by four hierarchical stages of Swin Transformer blocks and patch merging layers, progressively reducing spatial

resolution while increasing feature dimensionality. The bottleneck module operates at the lowest resolution, capturing deep contextual features.

The decoder mirrors the encoder structure, utilizing patch expanding layers and Swin Transformer blocks to restore spatial resolution and refine the feature representations. Skip connections are introduced at each level to recover fine-grained spatial information lost during downsampling.

Unlike traditional CNN-based U-Nets, Swin-Unet replaces convolutional layers with self-attention mechanisms using shifted windows. This allows the model to efficiently capture both local details and long-range dependencies without excessive computational overhead. A final upsampling module brings the output back to the original image resolution, followed by a 1×1 convolution to produce the enhanced image.

2.2.2. Loss function

The most commonly used loss functions in low-light image enhancement tasks are the Mean Absolute Error (MAE), often referred to as L_1-loss, and the Mean Squared Error (MSE), also known as L_2-loss. These functions have been widely adopted due to their simplicity and effectiveness in pixel-wise intensity comparison.

Recent top-tier works, such as [52] and [2], prominently utilize the L_1-loss, highlighting its continued relevance in state-of-the-art models. The formula for L_1-loss is given by:

$$L_1 = \frac{1}{N}\sum_{i=1}^{N} |\hat{y}_i - y_i| \ , \tag{1}$$

where $\hat{y}_i$ denotes the predicted pixel value, y_i is the corresponding ground-truth value, and N is the total number of pixels. For comparison, the L_2 loss (mean squared error, MSE) is defined as:

$$L_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \ . \tag{2}$$

While L_2-loss penalizes large deviations more heavily, leading to smoother outputs, L_1-loss is less sensitive to outliers and often results in sharper reconstructions. This distinction makes L_1-loss preferable in tasks requiring better preservation of image details.

In addition to pixel-wise losses, perceptual losses have gained popularity for improving the visual quality of enhanced images. In [31], the authors utilize a combination of MSE and perceptual loss based on a pre-trained VGG19 network. The perceptual loss compares feature maps from different layers of the VGG19 network for both generated and reference images, ensuring better high-level feature alignment. The perceptual loss is formulated as:

$$L_{\text{VGG}} = \frac{1}{N}\sum_{i=1}^{N} \|\text{VGG}(\hat{I}_i) - \text{VGG}(I_i)\|_2^2 \, , \tag{3}$$

where $\hat{I}_i$ and I_i represent the predicted and ground truth images, respectively, and VGG denotes the feature extraction function using the VGG19 network.

The composite loss function used in this work combines MSE and perceptual loss as follows:

$$L_{\text{composite}} = L_{\text{MSE}} + \lambda L_{\text{VGG}}\,, \tag{4}$$

where λ is a hyperparameter balancing the contributions of the two components. According to the authors, $\lambda = 0.25$ yields optimal results.

Similarly, [8] proposes a loss function designed for low-light image enhancement in both HVI and sRGB colour spaces; we will refer to it as *FN-loss* in the remainder of this paper to simplify the nomenclature. The total loss L is defined as:

$$L = \lambda_c \cdot l(\hat{I}_{\text{HVI}}, I_{\text{HVI}}) + l(\hat{I}, I)\,, \tag{5}$$

where $\hat{I}_{\text{HVI}}$ and I_{HVI} are the predicted and ground truth images in the HVI colour space, $\hat{I}$ and I are the predicted and ground truth images in the sRGB colour space, and λ_c is a weight balancing the two losses.

The loss function l for each colour space consists of multiple components:

$$l(\hat{X}, X) = \lambda_1 L_1(\hat{X}, X) + \lambda_e L_e(\hat{X}, X) + \lambda_p L_p(\hat{X}, X)\,, \tag{6}$$

where: L_1 loss denotes the pixel-wise L_1 loss, L_e is the edge loss encouraging edge preservation in the enhanced image, and L_p is the perceptual loss, ensuring perceptual similarity by comparing features extracted by a pre-trained network (e.g., VGG19). λ_1, λ_e, and λ_p are weights controlling the contributions of the respective loss components.

The proposed approaches demonstrate the efficacy of combining multiple loss components, including pixel-wise, edge, and perceptual losses, to achieve enhanced brightness, colour accuracy, and edge sharpness in low-light image enhancement tasks.

A notable example of an advanced loss function design is presented in [1]. The authors of LYT-Net used a hybrid loss function that combines multiple components to jointly optimise image brightness, perceptual quality, structural similarity, and colour fidelity. Their loss function can be expressed as:

$$L_{\text{total}} = L_S + \alpha_1 L_{\text{Perc}} + \alpha_2 L_{\text{Hist}} + \alpha_3 L_{\text{PSNR}} + \alpha_4 L_{\text{colour}} + \alpha_5 L_{\text{MS-SSIM}}\,, \tag{7}$$

where: L_{S} denotes the Smooth L_1 loss, applying a linear or quadratic penalty depending on the error magnitude to handle outliers effectively, L_{Perc} is the perceptual loss enforcing high-level feature consistency via VGG feature maps, L_{Hist} is the histogram loss aligning intensity distributions of prediction and ground truth, L_{PSNR} is the PSNR-based loss penalizing deviations in peak signal-to-noise terms, L_{colour} is the colour fidelity loss minimizing channel-wise mean differences, and $L_{\text{MS-SSIM}}$ is the multiscale structural similarity loss preserving structure across scales.

Each component in this hybrid loss function addresses a specific aspect of the enhancement problem, ensuring a balanced optimization process. This approach demonstrates how combining multiple loss terms can lead to excellent results in low-light image enhancement.

Both methods, [1] and [20], achieve excellent performance, particularly on synthetic datasets like LOLv2. However, models trained with simpler loss functions, such as the L_1-loss used in [52], tend to perform better on real-world datasets. This suggests that while advanced hybrid loss functions can improve performance on controlled datasets, simpler losses might generalize better in real-world scenarios. The superior real-world performance of [52] is likely influenced by the entire network architecture and training optimization strategy, including the choice of loss function.

In [20], the authors employ a vector quantization-based method for low-light image enhancement and define separate loss functions across three stages:

Stage I Loss: The goal is to train a normal-light encoder, decoder, and codebook using a combination of:

$$L_{\text{Stage I}} = L_{\text{recon}} + \beta L_{\text{vq}} \,, \tag{8}$$

where L_{recon} is the L_2-loss (Mean Squared Error) ensuring pixel-wise reconstruction accuracy, and L_{vq} is the vector quantization loss, which penalizes differences between the encoded and quantized features.

Stage II Loss: To bridge the gap between low-light and normal-light feature spaces, a distillation loss is introduced, alongside a query loss that optimises the matching process:

$$L_{\text{Stage II}} = L_{\text{distill}} + L_{\text{query}} \,, \tag{9}$$

Here, L_{distill} minimizes the feature-level discrepancy using L_1-loss, while L_{query} ensures accurate codebook item selection by aligning distance maps between features and codebook/query items.

Stage III Loss: In the final stage, a fusion branch combines features from different scales, and a brightness-aware attention module is employed to refine the enhanced image. The total loss in this stage is an L_1-loss defined as:

$$L_{\text{Stage III}} = \|I_{\text{rec}} - I_N\|_1 \tag{10}$$

where I_{rec} is the reconstructed image, and I_N is the ground truth normal-light image. Influence when parameters change: Eq. (10) has no explicit hyperparameters; if weighted by λ_3 in the total loss, increasing λ_3 scales the gradient $\partial L/\partial I_{\text{rec}} = \lambda_3 \operatorname{sign}(I_{\text{rec}} - I_N)$ and enforces pixel fidelity (typically higher PSNR/SSIM, smoother textures), while decreasing λ_3 lets perceptual/structural terms dominate (often sharper appearance with slight PSNR/SSIM trade-off). Replacing $\|\cdot\|_1$ with $\|\cdot\|_2^2$ would penalize large residuals more (more denoising/smoothness, potential edge blurring); keeping L_1 preserves edges and is outlier-robust. Stronger brightness-aware attention

concentrates updates in dark regions (better shadow recovery, risk of halos if excessive); weaker attention spreads updates (fewer artifacts, possible residual shadow noise). We use plain L_1 ($\lambda_3 = 1$ unless stated) and control the overall balance via Eq. (11).

To better align the network output with human visual perception, we augment classic pixel-wise objectives with a deep-feature component based on LPIPS [48]. The total training signal is defined as:

$$L_{\text{total}} = \alpha_{\text{S}} L_S + \alpha_P L_{\text{LPIPS}} + \alpha_M L_{\text{MS-SSIM}} + \alpha_N L_{\text{PSNR}} + \alpha_C L_{\text{colour}} + \alpha_G L_{\text{Grad}}\,, \quad (11)$$

where L_S is the Smooth-L_1 loss, $L_{\text{MS-SSIM}}$ is the multi-scale structural similarity loss (computed with detached gradients), L_{PSNR} is the inverted PSNR loss, L_{colour} penalizes differences in channel-wise mean values, and L_{Grad} enforces edge consistency using Sobel-based gradients. The perceptual term L_{LPIPS} uses the metric introduced by Zhang et al. [48], based on a frozen AlexNet backbone [16]. During training, both prediction and ground-truth images are forwarded through the LPIPS network in `no_grad` mode, after being rescaled from $[0, 1]$ to $[-1, 1]$, as required by the implementation. The choice of the LPIPS loss weight α_P was also subject to ablation, as we evaluated different values to balance perceptual quality and training stability. A comprehensive comparison of alternative loss functions and weight configurations is presented later in the paper.

2.2.3. Training setup

The complete pipeline is implemented in PyTorch 2.3 [28] with native AMP (Automatic Mixed Precision), uDNN (CUDA Deep Neural Network library) [26], benchmarking enabled, weight-initialization utilities from *timm* [25], and tensor rearrangements from *einops* [29, 30]. The Swin-Unet backbone is realised as a pure-attention U-Net: a patch-embedding stem feeds four encoder stages that alternate shifted-window multi-head self-attention, MLPs and residual connections, each stage halving the spatial resolution through patch merging; a bottleneck attends at the coarsest scale; four symmetric decoder stages then perform patch expansion while concatenating the corresponding encoder activations; an expand-by-four layer followed by a 1×1 projection produces the RGB output. Three capacities are explored by crossing initial widths $256, 384, 512$ with depth patterns 2–4–6–2, 2–4–8–2 and 2–6–12–4, giving nine architectural variants.

Training uses the LOL-v1 split, both LOL-v2 subsets and the SID corpus; for SID only the darkest exposure of every scene is paired with its long-exposure reference and the official Part-1 / Part-2 division is kept for training and validation. All images are converted to linear $[0, 1]$, randomly flipped and rotated by multiples of 90°, then partitioned into non-overlapping 256×256 crops that serve as individual samples; evaluation runs on a single uncropped patch without test-time augmentation. Four supervision regimes are tested: the hybrid LYT objective, the six-term LPIPS-augmented loss of Eq. (11) with $\alpha_P \in 0.1, 0.2, 0.5$, pure MSE and the colour-space FN-loss of Feng et al [8]. In every

case AdamW starts at 1×10^{-4}, warms up linearly for five epochs, decays cosinely to 1×10^{-6}, applies weight-decay of 10^{-4}, clips the gradient norm to 1.0 and accumulates two mixed-precision micro-batches, yielding an effective batch of sixteen patches. Each run spans one hundred epochs and the checkpoint with the lowest mean validation loss over LOL-v1, LOL-v2-real and LOL-v2-synthetic is retained.

All experiments were run on a single NVIDIA RTX 4090. Mini-batch size was adjusted per model to saturate GPU memory; for the 512-channel backbone this meant a batch size of 1, which noticeably slowed iterative testing. Given the tight hardware and time budget – and the wish to cover nine capacities and four loss functions – some hyper-parameters (e.g. the LPIPS multiplier) were fixed to representative values instead of being exhaustively tuned. Access to stronger hardware would allow a broader sweep over embed width, window size and loss weights, leading to a more thoroughly optimised model.

3. Experimental results

In this section, we present extensive experimental validation of our proposed Swin-Unet-based method for low-light image enhancement. We systematically evaluated the performance impact of key architectural choices, different strategies for incorporating supplementary datasets, and various loss functions. To directly address the reviewer's concern and isolate sources of improvement, we conducted two complementary ablations: (i) with the architecture and data held fixed, we varied only the loss (MSE, FN-loss, LYT, and LPIPS-weighted variants); and (ii) with the loss and data held fixed, we varied only the architecture (embedding dimensions and transformer depths). The baseline for all comparisons was the original Swin-Unet model configuration with embedding dimension 512 and hierarchical depths of 2-4-8-2, which previously demonstrated promising results in similar vision tasks. The LOL-v1 and LOL-v2 datasets (both synthetic and real subsets) were utilized as primary benchmarks. We specifically investigated the impact of embedding dimensions and transformer depths, dataset integration strategies (particularly regarding the SID dataset), and diverse loss function formulations, including Mean Squared Error (MSE), FN-loss, LYT loss, and our proposed LPIPS-based perceptual loss function. The evaluation metrics used were Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR), commonly adopted standards for image enhancement assessment.

3.1. Comparative analysis

Initially, we focused on the effective use of the SID dataset within the training pipeline. Four distinct approaches were tested using the optimal Swin-Unet architecture (embedding dimension 512, depths 2-4-8-6) and LYT loss: (1) selecting the single darkest image per scene from SID, (2) selecting the three darkest images, (3) randomly choosing SID

Tab. 1. Comparison of SID dataset integration strategies using LYT loss.

SID Strategy	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
Single darkest	**0.829**	21.43	**0.834**	22.55	0.897	22.46
Three darkest	0.799	**23.16**	0.825	22.63	0.897	22.40
Random	0.772	22.53	0.810	22.58	**0.904**	**23.30**
No SID	0.780	22.13	0.826	**23.67**	0.902	22.61

Tab. 2. Effect of embedding dimensions and depths (LYT loss, single darkest SID).

Embed dim / depths	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
512 / 2-4-8-2	**0.829**	21.43	**0.834**	**22.55**	0.897	**22.46**
512 / 2-4-6-2	0.762	20.00	0.803	21.56	0.883	20.84
384 / 2-4-6-2	0.759	20.91	0.792	21.10	0.872	20.52
384 / 2-6-12-4	0.784	**21.47**	0.806	21.74	**0.895**	22.52

images, and (4) completely excluding SID. Table 1 summarizes these experiments, clearly indicating that leveraging the single darkest SID image achieved consistently superior results. This strategy yielded an SSIM = 0.829 and PSNR = 21.43 for LOL-v1, and SSIM = 0.834 and PSNR = 22.55 for LOL-v2-real, significantly outperforming alternative approaches.

The observed differences between SID usage strategies highlight that carefully selecting SID images based on luminance intensity notably improves performance and training stability. Because the loss and architecture were held fixed here, these gains are attributable to the data integration strategy rather than the perceptual loss choice. Random SID selection, although performing well on synthetic datasets, showed reduced consistency across real-world benchmarks.

We then explored varying model configurations by adjusting the embedding dimensions and transformer depths, again utilizing the optimal SID selection (single darkest image). We compared embedding dimensions of 384 and 512, and various depth configurations, specifically 2-4-6-2 and 2-6-12-4. As Table 2 demonstrates, significantly lower embedding dimensions (384) substantially decreased SSIM and PSNR values, indicating insufficient representational capacity. Thus, such configurations were excluded from further experiments.

Under a fixed loss (LYT) and data strategy, increasing architectural capacity from depths 2-4-6-2 to 2-4-8-2 at embed = 512 improved SSIM/PSNR by +0.067/+1.43 (LOL-v1), +0.031/+0.99 (LOL-v2-real), and +0.014/+1.62 (LOL-v2-synth). These deltas are larger than those observed when swapping perceptual losses under a fixed architecture (see below), indicating that most SSIM/PSNR gains stem from the architecture.

Next, we assessed several loss functions to determine their efficacy. Specifically, we compared MSE, FN-loss, LYT loss, and our perceptual LPIPS-based loss with varying LPIPS multipliers (0.1, 0.5, and 1.0). Results summarized in Table 3 illustrate that simpler loss functions such as MSE and FN-loss underperformed notably, with MSE consistently lowest due to its exclusive pixel-level error penalization, which leads to

Tab. 3. Performance comparison of different loss functions.

Loss Function	SSIM LOL-v1	PSNR LOL-v1	SSIM LOL-v2-real	PSNR LOL-v2-real	SSIM LOL-v2-synth	PSNR LOL-v2-synth
LYT	**0.829**	21.43	**0.834**	22.55	**0.897**	22.46
LPIPS (0.1)	0.827	**21.77**	0.826	**22.60**	**0.897**	22.42
LPIPS (0.5)	0.789	21.13	0.827	22.32	0.895	**22.58**
LPIPS (1.0)	0.789	21.05	0.799	20.46	0.871	21.72
FN-loss	0.798	21.41	0.809	21.09	0.882	22.11
MSE	0.675	19.27	0.722	18.12	0.832	19.00

Tab. 4. NIQE and BRISQUE scores for the four loss functions (lower is better).

Loss	Dataset	NIQE	BRISQUE
MSE	LOL-v1	5.20	19.26
MSE	LOL-v2-real	5.46	20.90
MSE	LOL-v2-synth	5.02	15.84
FN-Loss	LOL-v1	7.14	22.56
FN-Loss	LOL-v2-real	7.36	25.78
FN-Loss	LOL-v2-synth	6.30	17.36
LYT	LOL-v1	5.79	15.36
LYT	LOL-v2-real	6.16	18.00
LYT	LOL-v2-synth	5.85	16.42
LPIPS	LOL-v1	5.55	17.18
LPIPS	LOL-v2-real	5.97	19.23
LPIPS	LOL-v2-synth	5.58	16.08

overly smooth and detail-deficient images. Conversely, LYT and LPIPS-based losses yielded the highest results, largely attributed to their composite nature – incorporating pixel-wise accuracy, perceptual quality, structural similarity, and colour fidelity, thus better aligning with human visual preferences.

With the architecture held constant (embed = 512, depths 2-4-8-2) and the same data strategy, LPIPS at a small weight (0.1) slightly increased PSNR relative to LYT while keeping SSIM essentially unchanged: +0.34 dB / −0.002 (LOL-v1) and +0.05 dB / −0.008 (LOL-v2-real); results on LOL-v2-synth were virtually tied (−0.04 dB / 0.000). Heavier LPIPS weights (0.5–1.0) reduced effectiveness, emphasizing the importance of balancing perceptual and pixel-level constraints. These comparisons show that while architectural capacity dominates fidelity (SSIM/PSNR), a lightly weighted LPIPS term can nudge optimization toward slightly better PSNR without sacrificing SSIM.

The four representative checkpoints were re-evaluated with the no-reference perceptual metrics NIQE [23] and BRISQUE [22] (Tab. 4). NIQE measures the deviation of an image's natural-scene statistics from a model learned on pristine photographs, whereas BRISQUE regresses locally normalized luminance and contrast statistics to subjective quality scores. Lower values in both cases correspond to higher perceptual quality.

Across the entire evaluation spectrum, LYT and LPIPS deliver noticeably better NIQE and BRISQUE scores than the multi-component FN-Loss of Feng et al., combining L_1, edge, and perceptual terms in both sRGB and HVI colour spaces. LPIPS attains the lowest NIQE values among the perceptual objectives, whereas LYT secures the best BRISQUE on LOL-v1 and LOL-v2-real, with LPIPS edging ahead on the synthetic subset. Because the architecture was fixed in these comparisons, these perceptual gains can be attributed primarily to the loss design.

Surprisingly, the plain MSE loss performs very competitively – particularly on LOL-v2-synth, where it records the overall best NIQE of 5.02. This suggests that strict pixel fidelity can suppress subtle non-linear artefacts sometimes introduced by perceptual losses; such artefacts are often imperceptible to the human eye yet penalised by statistical quality metrics. In summary, perceptually driven losses (LYT and LPIPS) still provide clear gains over FN-Loss, but a well-tuned MSE baseline remains a strong contender when judged solely by no-reference measures.

Detailed training convergence (Fig. 1) shows that, under the same architecture, the LYT loss and LPIPS with weight 0.1 both stabilize training and maintain superior PSNR/SSIM across epochs, with LPIPS slightly stronger in later epochs. Increasing the LPIPS weight reduces effectiveness, underscoring the need to balance perceptual and pixel-level terms. FN-Loss converges more gradually but remains competitive, whereas MSE lags throughout. Convergence plateaus appear around epoch 90.

Taken together, the ablations make the source of possible improvements explicit: most SSIM/PSNR gains come from scaling the Swin-Unet architecture (e.g., up to +1.62 dB PSNR when increasing depth at embed = 512), while perceptual gains (NIQE) are predominantly induced by the LPIPS-based loss when the architecture is fixed. The best

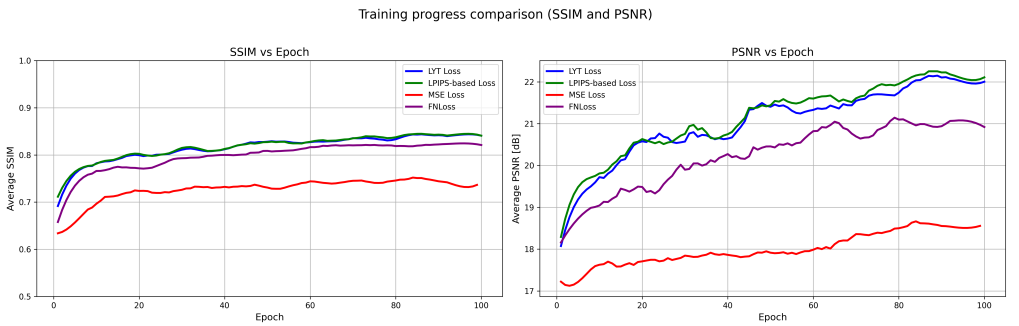

Fig. 1. Validation PSNR and SSIM versus training epochs. Legend: LYT Loss (blue), LPIPS-based loss with weight 0.1 (green), FN-Loss (purple), and MSE Loss (red). Curves are smoothed; metrics are computed after each epoch on the full validation set comprising SID (darkest exposure), LOL-v1, and the real and synthetic subsets of LOL-v2.

results arise from their combination – adequate model capacity paired with a modest LPIPS weight – yielding images that are both faithful and perceptually convincing.

These comprehensive results underscore the importance of model capacity, appropriate dataset integration, and carefully chosen composite loss functions in achieving high-quality, perceptually convincing low-light image enhancement; a visual comparison of our model's outputs with the reference images is provided in Figure 2.

On a per-dataset basis, holding the loss fixed (LYT) and increasing capacity from depths 2-4-6-2 to 2-4-8-2 at embed = 512 yields ΔPSNR/ΔSSIM of $+1.43/+0.067$ (LOL-v1), $+0.99/+0.031$ (LOL-v2-real), and $+1.62/+0.014$ (LOL-v2-synth). With the architecture fixed, LPIPS(0.1) improves NIQE vs. LYT by 0.24 (5.55 vs. 5.79, LOL-v1), 0.19 (5.97 vs. 6.16, LOL-v2-real), and 0.27 (5.58 vs. 5.85, LOL-v2-synth); BRISQUE favors LYT on real images (15.36 vs. 17.18; 18.00 vs. 19.23), while LPIPS is slightly better on synthetic (16.08 vs. 16.42). Although MSE attains a strong NIQE on LOL-v2-synth (5.02), it lags markedly in SSIM/PSNR across datasets. For data integration, selecting the single darkest SID exposure per scene is the most consistent strategy on real benchmarks; random selection can score higher on synthetic data but is less stable overall.

In practice, a compact recipe emerges: embed = 512 with depths 2-4-8-2, training on SID (single darkest) and a light LPIPS weight (0.1). Heavier LPIPS weights (0.5–1.0) reduce fidelity and stability, and convergence plateaus around epoch 90, after which early stopping is beneficial. Qualitatively (Fig. 2), this setting mitigates colour shifts and preserves edges, with only minor brightness deviations relative to ground truth.

3.2. Comparison with other algorithms

The quantitative comparison of our best-performing model – Swin-Unet trained with the proposed LPIPS-based loss function – is presented in Table 5. Although the model employing the LYT loss achieved similar performance, we prioritize the LPIPS-based approach as it introduces a novel perceptual component specifically tailored to low-light image enhancement. Furthermore, since the LPIPS-based loss was explicitly designed and proposed within this work, it more clearly represents our contributions.

From the results, it is evident that our Swin-Unet architecture achieves competitive but somewhat lower quantitative performance compared to state-of-the-art methods on all considered LOL datasets. Specifically, our best model achieved PSNR and SSIM of 21.77 dB and 0.827 on LOL-v1, 22.60 dB and 0.826 on LOL-v2-real, and 22.42 dB and 0.897 on LOL-v2-synthetic. In contrast, leading architectures such as CIDNet-oP [8], RetinexFormer [2], and LYT-Net [1] consistently surpass these metrics across all benchmarks, reaching PSNR values around 28 dB and SSIM over 0.88 in many cases.

These observed discrepancies may suggest that the Swin-Unet architecture – originally proposed for medical image segmentation – might not be optimal in capturing the

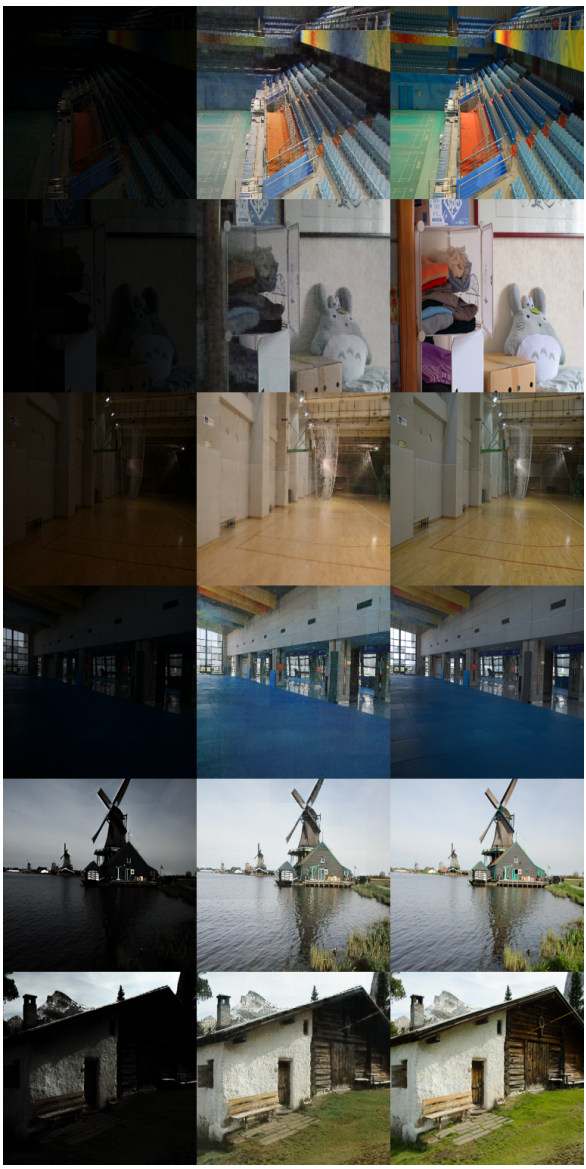

Fig. 2. Qualitative comparison layout and data sources. Columns: left – low-light inputs; centre – outputs from the model trained with an LPIPS-weighted loss; right – corresponding well-exposed ground-truth images. Rows: 1–2 from LOL-v1; 3–4 from LOL-v2-real; 5–6 from LOL-v2-synth. Images are randomly selected examples from the LOL family.

Tab. 5. Quantitative results on LOL datasets.

Methods	PSNR (LOL-v1)	SSIM (LOL-v1)	PSNR (LOL-v2-real)	SSIM (LOL-v2-real)	PSNR (LOL-v2-syn)	SSIM (LOL-v2-syn)
SID [4]	14.35	0.436	13.24	0.442	15.04	0.610
3DLUT [47]	21.35	0.585	20.19	0.745	22.17	0.854
Zero-DCE [11]	14.86	0.540	13.65	0.246	21.46	0.848
EnlightenGAN [15]	17.48	0.650	18.23	0.617	—	—
KinD [51]	20.87	0.800	20.40	0.652	16.26	0.591
KinD++ [49]	21.30	0.820	20.15	0.678	19.44	0.830
Bread [12]	22.96	0.840	22.54	0.762	19.28	0.831
IAT [6]	23.38	0.810	21.43	0.638	19.18	0.813
HWMNet [7]	24.24	0.850	22.40	0.622	18.79	0.817
LLFlow [35]	24.99	0.920	21.60	0.643	19.15	0.860
DeepUPE [33]	14.38	0.446	13.27	0.452	15.08	0.623
DeepLPF [24]	15.28	0.473	14.10	0.480	16.02	0.587
UFormer [36]	16.36	0.771	18.82	0.771	19.66	0.871
RetinexNet [37]	18.92	0.427	18.32	0.447	19.09	0.774
Sparse [42]	17.20	0.640	20.06	0.816	22.05	0.905
EnGAN [15]	20.00	0.691	18.23	0.617	16.57	0.734
FIDE [39]	18.27	0.665	16.85	0.678	15.20	0.612
Restormer [46]	26.68	0.853	26.12	0.853	25.43	0.859
LEDNet [53]	25.47	0.846	27.81	0.870	27.37	0.928
SNR-Aware [40]	26.72	0.851	27.21	0.871	27.79	0.941
LLFormer [34]	25.76	0.823	26.20	0.819	28.01	0.927
RetinexFormer [2]	27.14	0.850	27.69	0.856	28.99	0.939
CIDNet-wP [8]	27.72	0.876	**28.13**	**0.892**	29.37	**0.950**
CIDNet-oP [8]	**28.14**	0.889	27.76	0.881	**29.57**	**0.950**
A3DLUT [32]	14.77	0.458	18.19	0.745	18.92	0.838
IPT [5]	16.27	0.504	19.80	0.813	18.30	0.811
Band [41]	20.13	0.830	20.29	0.831	23.22	0.927
LPNet [18]	21.46	0.802	17.80	0.792	19.51	0.846
SNR [40]	24.61	0.842	21.48	0.849	24.14	0.928
LLIE [20]	25.24	0.855	25.94	0.854	27.79	0.941
PyDiff [52]	27.09	**0.930**	24.01	0.876	19.60	0.878
MIRNet [45]	26.52	0.856	27.17	0.865	25.96	0.898
LYT-Net [1]	27.23	0.853	27.80	0.873	29.38	0.940
Ours Swin-Unet (LPIPS-based)	21.77	0.827	22.60	0.826	22.42	0.897

specific features necessary for low-light image enhancement. However, despite somewhat lower quantitative results, the Swin-Unet architecture presents certain distinct advantages. Its pure transformer-based design effectively leverages global context modelling through self-attention mechanisms, enabling a strong representation of both local details and long-range dependencies simultaneously. Moreover, the architecture is relatively straightforward, highly modular, and significantly easier to train and fine-tune compared to more complex multi-stage architectures, such as those incorporating diffusion models or hybrid convolution-transformer networks.

Another key advantage of our model is computational efficiency and flexibility. While it is plausible that utilizing a larger-scale Swin-Unet network (e.g., deeper or wider variants) could potentially yield better quantitative performance, our experimental investigation was limited by available computational resources and time constraints. Therefore, an extensive exploration of larger models was beyond the scope of this work.

Nonetheless, the performance achieved demonstrates the viability and potential of the Swin-Unet approach – especially when paired with novel perceptual losses such as our LPIPS-based formulation. Given its favorable balance between complexity, computational efficiency, and respectable image enhancement quality, Swin-Unet remains an attractive candidate for further exploration, potentially yielding improved performance if scaled appropriately.

4. Conclusions and contributions

This study set out to verify whether a compact, single-stage Swin-Unet can remain competitive in extremely low-light conditions once supervision is shifted from purely pixel-based criteria to a perceptually oriented objective. The network we employed – an off-the-shelf Swin-Unet restricted to an embedding width of 512 and an encoder–decoder depth pattern of 2-4-8-2 – was purposefully kept small: with batch size one it already saturates the memory of a single RTX 4090, and shortening turnaround times was essential for running the nine-by-four grid of capacity-and-loss experiments reported throughout the paper. Within these resource limits several contributions emerge.

First, the composite loss that blends LPIPS, Smooth-L_1, MS-SSIM, inverted PSNR, colour mean and gradient consistency proves almost as effective as the far more elaborate LYT objective when both are applied to the same Swin-Unet backbone; on LOL-v1 and LOL-v2-real the two formulations reach virtually identical SSIM, while the LPIPS variant shows a slight PSNR advantage on two of the three benchmark splits. This confirms that loss design can close much of the perceptual gap even when architectural capacity is modest.

Second, the paper offers what is, to our knowledge, the first transformer-only baseline that covers LOL-v1, LOL-v2-real, LOL-v2-synthetic and SID under a single, fully documented training protocol; future work can therefore compare new transformer variants against numbers that are not confounded by convolutional extras or multi-branch tricks.

Third, the SID ablation confirms that keeping only the darkest exposure of each scene yields more dependable generalisation than either random or multi-exposure sampling – an observation that simplifies dataset preparation and, to our knowledge, had not been quantified before. The study also clarifies the limitations of our approach. Even the strongest configuration trails recent diffusion or multi-branch systems by roughly 5–6 dB in PSNR and a few hundredths in SSIM; visual inspection further reveals occasional smoothing of fine texture, most notably in areas dominated by read-noise. These deficits likely stem from choices that remained arbitrary because of limited time and compute – for example, the fixed LPIPS weight, the 7×7 shifted-window size, and the cap on embedding width. A wider sweep over those hyper-parameters, combined with experiments on deeper or broader Swin backbones, appears the most direct route to closing the remaining performance gap.

In short, although the model remains below the current state of the art, the study shows that a judiciously balanced perceptual loss can bring a compact Swin-Unet within striking distance of results obtained with far more elaborate objectives, establishes a clean transformer-only benchmark for future scaling studies, and uncovers a simple luminance-based strategy for sampling SID that reliably improves generalisation – insights that will help subsequent research allocate computational resources where they matter most.

References

[1] A. Brateanu, R. Balmez, A. Avram, C. Orhei, and C. Ancuti. LYT-NET: Lightweight YUV transformer-based network for low-light image enhancement. *IEEE Signal Processing Letters* 32:2065–2069, 2025. doi:10.1109/LSP.2025.3563125.

[2] Y. Cai, H. Bian, J. Lin, H. Wang, R. Timofte, et al. Retinexformer: One-stage Retinex-based transformer for low-light image enhancement. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12504–12513, 2023. doi:10.1109/ICCV51070.2023.01149.

[3] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, et al. Swin-Unet: Unet-like pure transformer for medical image segmentation. In: *Computer Vision – ECCV 2022 Workshops*, vol. 13803 of *Lecture Notes in Computer Science*, 2023. doi:10.1007/978-3-031-25066-8_9.

[4] C. Chen, Q. Chen, J. Xu, and V. Koltun. Learning to see in the dark. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 3291–3300, 2018. doi:10.1109/CVPR.2018.00347.

[5] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, et al. Pre-trained image processing transformer. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, pp. 12299–12310, 2021. doi:10.1109/CVPR46437.2021.01212.

[6] Z. Cui, K. Li, L. Gu, S. Su, P. Gao, et al. You only need 90k parameters to adapt light: a light weight transformer for image enhancement and exposure correction. In: *33rd British Machine Vision Conference (BMVC 2022)*, 2022. https://bmvc2022.mpi-inf.mpg.de/238/.

[7] C.-M. Fan, T.-J. Liu, and K.-H. Liu. Half wavelet attention on M-Net+ for low-light image enhancement. In: *2022 IEEE International Conference on Image Processing (ICIP 2022)*, pp. 3878–3882, 2022. doi:10.1109/ICIP46576.2022.9897503.

[8] Y. Feng, C. Zhang, P. Wang, P. Wu, Q. Yan, et al. You only need one color space: An efficient network for low-light image enhancement. arXiv, arXiv:2402.05809, 2024. doi:10.48550/arXiv.2402.05809.

[9] X. Fu, Y. Liao, D. Zeng, Y. Huang, X.-P. Zhang, et al. A probabilistic method for image enhancement with simultaneous illumination and reflectance estimation. *IEEE Transactions on Image Processing* 24(12):4965–4977, 2015. doi:10.1109/TIP.2015.2474701.

[10] Z. Gu, F. Li, F. Fang, and G. Zhang. A novel Retinex-based fractional-order variational model for images with severely low light. *IEEE Transactions on Image Processing* 29:7233–7247, 2020. doi:10.1109/TIP.2019.2958144.

[11] C. Guo, C. Li, J. Guo, C. C. Loy, J. Hou, et al. Zero-reference deep curve estimation for low-light image enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 1780–1789, 2020. doi:10.1109/CVPR42600.2020.00185.

[12] X. Guo and Q. Hu. Low-light image enhancement via breaking down the darkness. *International Journal of Computer Vision* 131:48–66, 2023. doi:10.1007/s11263-022-01667-9.

[13] H. Hou, Y. Hou, Y. Shi, B. Wei, and J. Xu. NLHD: A pixel-level non-local Retinex model for low-light image enhancement. arXiv, arXiv:2106.06971, 2021. doi:10.48550/arXiv.2106.06971.

[14] J. H. Jang, Y. Bae, and J. B. Ra. Contrast-enhanced fusion of multisensor images using subband-decomposed multiscale Retinex. *IEEE Transactions on Image Processing* 21(8):3479–3490, 2012. doi:10.1109/TIP.2012.2197014.

[15] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, et al. EnlightenGAN: Deep light enhancement without paired supervision. *IEEE Transactions on Image Processing* 30:2340–2349, 2021. doi:10.1109/TIP.2021.3051462.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, vol. 25, pp. 1097–1105, 2012. https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

[17] E. H. Land. The Retinex theory of color vision. *Scientific American* 237(6):108–128, 1977. doi:10.1038/scientificamerican1277-108.

[18] J. Li, J. Li, F. Fang, F. Li, and G. Zhang. Luminance-aware pyramid network for low-light image enhancement. *IEEE Transactions on Multimedia* 23:3153–3165, 2021. doi:10.1109/TMM.2020.3021243.

[19] R. Liu, L. Ma, J. Zhang, X. Fan, and Z. Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, pp. 10561–10570, 2021. doi:10.1109/CVPR46437.2021.01042.

[20] Y. Liu, T. Huang, W. Dong, F. Wu, X. Li, et al. Low-light image enhancement with multi-stage residue quantization and brightness-aware attention. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV 2023)*, pp. 12106–12115, 2023. doi:10.1109/ICCV51070.2023.01115.

[21] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, et al. Swin transformer V2: Scaling up capacity and resolution. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 11999–12009, 2022. doi:10.1109/CVPR52688.2022.01170.

[22] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing* 21(12):4695–4708, 2012. doi:10.1109/TIP.2012.2214050.

[23] A. Mittal, R. Soundararajan, and A. C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters* 20(3):209–212, 2013. doi:10.1109/LSP.2012.2227726.

[24] S. Moran, P. Marza, S. McDonagh, S. Parisot, and G. Slabaugh. DeepLPF: Deep Local Parametric Filters for image enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 12823–12832, 2020. doi:10.1109/CVPR42600.2020.01284.

[25] M. Noyan, A. R. Gostipathy, R. Wightman, and P. Cuenca. timm PyTorch Image Models. In: Hugging Face, 2025. https://huggingface.co/timm.

[26] NVIDIA Corporation. NVIDIA cuDNN. In: NVIDIA DEVELOPER, 2025. https://developer.nvidia.com/cudnn.

[27] S. Park, S. Yu, B. Moon, S. Ko, and J.-I. Paik. Low-light image enhancement using variational optimization-based Retinex model. *IEEE Transactions on Consumer Electronics* 63(2):178–184, 2017. doi:10.1109/TCE.2017.014847.

[28] PyTorch. Previous PyTorch Versions, 2025. https://pytorch.org/get-started/previous-versions/.

[29] A. Rogozhnikov. Einops: Clear and reliable tensor manipulations with Einstein-like notation. In: *International Conference on Learning Representations (ICLR 2022)*, 2022. https://openreview.net/forum?id=oapKSVM2bcj.

[30] A. Rogozhnikov. einops, 2025. https://einops.rocks/.

[31] H. Shakibania, S. Raoufi, and H. Khotanlou. CDAN: Convolutional dense attention-guided network for low-light image enhancement. *Digital Signal Processing* 156:104802, 2025. doi:10.1016/j.dsp.2024.104802.

[32] A. Wang, Y. Li, J. Peng, Y. Ma, X. Wang, et al. Real-time image enhancer via learnable spatial-aware 3D lookup tables. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV 2021)*, pp. 2451–2460, 2021. doi:10.1109/ICCV48922.2021.00247.

[33] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, et al. Underexposed photo enhancement using deep illumination estimation. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, pp. 6842–6850, 2019. doi:10.1109/CVPR.2019.00701.

[34] T. Wang, K. Zhang, T. Shen, W. Luo, B. Stenger, et al. Ultra-high-definition low-light image enhancement: A benchmark and transformer-based method. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2023)*, vol. 37 no. 3, pp. 2654–2662, 2023. doi:10.1609/aaai.v37i3.25364.

[35] Y. Wang, R. Wan, W. Yang, H. Li, L.-P. Chau, et al. Low-light image enhancement with normalizing flow. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2022)*, vol. 36 no. 3, pp. 2604–2612, 2022. doi:10.1609/aaai.v36i3.20162.

[36] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, et al. Uformer: A general U-shaped transformer for image restoration. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 17662–17672, 2022. doi:10.1109/CVPR52688.2022.01716.

[37] C. Wei, W. Wang, W. Yang, and J. Liu. Deep Retinex decomposition for low-light enhancement. In: *Proceedings of the British Machine Vision Conference (BMVC 2018)*, 2018. https://bmva-archive.org.uk/bmvc/2018/contents/papers/0451.pdf.

[38] J. Wen, C. Wu, T. Zhang, Y. Yu, and P. Swierczynski. Self-reference deep adaptive curve estimation for low-light image enhancement. arXiv, arXiv:2308.08197, 2023. doi:10.48550/arXiv.2308.08197.

[39] K. Xu, X. Yang, B. Yin, and R. W. H. Lau. Learning to restore low-light images via decomposition-and-enhancement. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020)*, pp. 2278–2287, 2020. doi:10.1109/CVPR42600.2020.00235.

[40] X. Xu, R. Wang, C.-W. Fu, and J. Jia. Snr-aware low-light image enhancement. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 17737–17747, 2022. doi:10.1109/CVPR52688.2022.01719.

[41] W. Yang, S. Wang, Y. Fang, Y. Wang, and J. Liu. Band representation-based semi-supervised low-light image enhancement: Bridging the gap between signal fidelity and perceptual quality. *IEEE Transactions on Image Processing* 30:3461–3473, 2021. doi:10.1109/TIP.2021.3062184.

[42] W. Yang, W. Wang, H. Huang, S. Wang, and J. Liu. Sparse gradient regularized deep Retinex network for robust low-light image enhancement. *IEEE Transactions on Image Processing* 30:2072–2086, 2021. doi:10.1109/TIP.2021.3050850.

[43] X. Yi, H. Xu, H. Zhang, L. Tang, and J. Ma. Diff-Retinex: Rethinking low-light image enhancement with a generative diffusion model. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6725–6735, 2023. doi:10.1109/ICCV51070.2023.01130.

[44] D. You, J. Tao, Y. Zhang, and M. Zhang. Low-light image enhancement based on gray scale transformation and improved Retinex. *Infrared Technology (Hongwai Jishu)* 45(2):161–170, 2023.

[45] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, et al. Learning enriched features for real image restoration and enhancement. In: *European Conference on Computer Vision (ECCV 2020)*, vol. 12370 of *Lecture Notes in Computer Science*, pp. 492–511, 2020. doi:10.1007/978-3-030-58595-2_30.

[46] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, et al. Restormer: Efficient transformer for high-resolution image restoration. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 5718–5729, 2022. doi:10.1109/CVPR52688.2022.00564.

[47] H. Zeng, J. Cai, L. Li, Z. Cao, and L. Zhang. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44(4):2058–2073, 2022. doi:10.1109/TPAMI.2020.3005590.

[48] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pp. 586–595, 2018. doi:10.1109/CVPR.2018.00068.

[49] Y. Zhang, X. Guo, J. Ma, W. Liu, and J. Zhang. Beyond brightening low-light images. *International Journal of Computer Vision* 129(4):1013–1037, 2021. doi:10.1007/s11263-020-01407-x.

[50] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(7):2480–2495, 2021. doi:arXiv:1812.10477.

[51] Y. Zhang, J. Zhang, and X. Guo. Kindling the darkness: A practical low-light image enhancer. In: *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, pp. 1632–1640, 2019. doi:10.1145/3343031.3350926.

[52] D. Zhou, Z. Yang, and Y. Yang. Pyramid diffusion models for low-light image enhancement. arXiv, arXiv:2305.10028, 2023. doi:10.48550/arXiv.2305.10028.

[53] S. Zhou, C. Li, and C. C. Loy. LEDNet: Joint low-light enhancement and deblurring in the dark. In: *European Conference on Computer Vision (ECCV)*, vol. 13666 of *Lecture Notes in Computer Science*, pp. 573–589, 2022. doi:10.1007/978-3-031-20068-7_33.