SFTW & LV+SFTW 2024: Live Restoration of Weather-Degraded Images with Self-Guided Filter based on TransWeather

Zhang Li Fu
Wellington College International Shanghai
Shanghai, China
Work (Active): zhanglifu1mil@gmail.com
Organization: lifu.zhang.2026@wellingtoncollege.cn

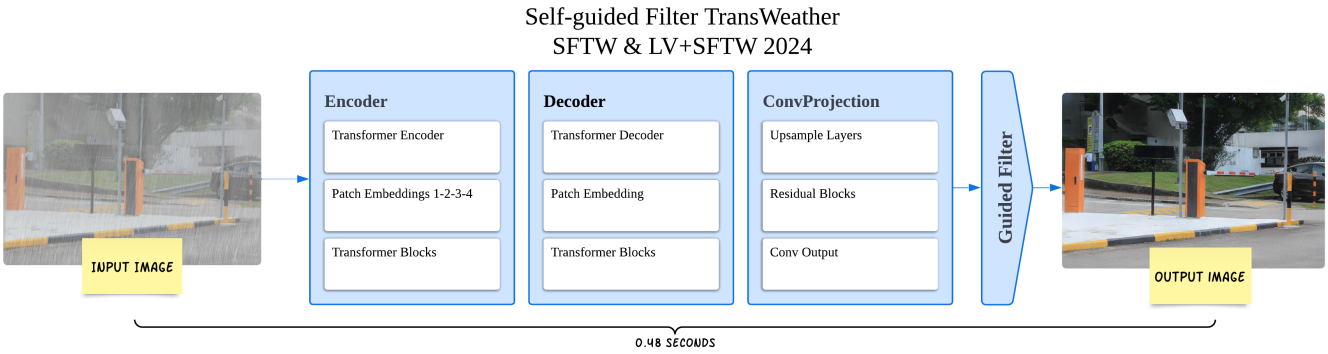

Fig. 1. Summary of the SFTW Framework

***Abstract*—SFTW and LV+SFTW 2024 (the basic: Self-guided Filter TransWeather - SFTW and Premium Live Video Plus - LV+SFTW) is a strategically evolved weather-degraded image restoration model established on the foundational TransWeather model[1] proposed in 2022, specifically targeting to enhance and significantly push the boundaries of the industry to more practicality and portability. The base, TransWeather[1], is a "transformer based end-to-end model" which implements a streamlined yet efficient "single encoder to decoder" network. The encoder utilizes "intra-patch attention" and the decoder with "learnable weather embeddings" to remove all weather degradations effectively and accurately. This enhanced model introduces several key innovations, including integrating a trainable self-guided filter layer, critical adjustments to the training framework, and a live cam and video processing function to its inference (processing output code), which is this industry's pioneer. The self-guided filter layer is designed and implemented to amplify the quality of the output images by effectively reducing noise artifacts and preserving the processed edges of the image, which is a common issue present in existing image restoration models, massively assisting other industries such as object detection algorithms. Despite minor enhancements in the quantitative analysis, such as PSNR and SSIM values, the introduction of a video and live cam processing usability and a guided filter demonstrates the significant potential of this model to advance in this field.**

***Index Terms*—Self-Guided Filter, TransWeather, Transformer, Encoder Decoder, weather-degraded image restoration, ResNet50, real-time processing, Allweather,**

I. Introduction

A. Background

Adverse weather conditions, including heavy rain, haze, and snow, significantly reduce the quality and visibility of images shot on a camera, which notably negatively impacts the performance and accuracy of all computer vision algorithms. Algorithms like object detection, semantic segmentation, and optical flow estimation are all examples that rely on apparent visual features and consistent quality for accurate analysis. Such required images are also critical for the safety and reliability of the operation of such systems, which makes the cooperation of the emerging concept of image restoration algorithms with such algorithms indispensable and critical for reliability for use in practice, inspiring innovations in this field.

Fig. 2. Comparison of an object detection model performance on a degraded and processed image through SFTW

These concerns about weather-degraded images' impact on computer vision algorithms can be visualized and emphasized through a simple demonstration through Roboflow Object Detection Playground [42], as shown in Figure 2. The model was tasked with labeling elements of 2 sets of images; the

first set, labeling a natural element, 'sky,' and a human element, 'warehouse,' performed accurately in clear (SFTW-processed) conditions by correctly labeling the two elements, but struggles under weather degradation, by failing to identify both elements. The second set tasks to label critical human elements, especially in automated driving systems, 'road' and 'cars.' Although accurately identifying both in both weather degraded conditions, the previous failure still reflects the inconsistency weather degradation brings to such algorithms, which ultimately risks the safety of passengers, pedestrians, or public property, further emphasizing the need for practical image restoration models.

Before the introduction of our base, TransWeather[1], in 2022, image restoration models specific to weather-degraded images were only designed to restore and tackle specific weather conditions. Such models include de-raining models[19][20][21][22], de-hazing[12][13][14][15] models, and de-snowing models[23][24][25], using either CNN, Transformer, or GAN architecture bases, but are obviously inadequate in adaptability and scalability and even in quantitative indicators, SSIM and PSNR. Subsequently, these models required separate training image sets and pipelines for different weather conditions, which proved its impracticality for common real-world weather scenarios. Though each method introduced advancements to its priors and achieved decent inference performances, the problem of lack of adaptability and complexity still restricts its effectiveness in practice; for instance, in real life, many adverse weather conditions include a mix of weather, such as haze with heavy rain. The complexity of training multiple models in production further limits their practical usability, which confines them only as a concept when this is a prudent and valuable approach. The only All-in-one model, initially proposed by Li et al. [3], which inspired TransWeather[1], effectively tackled the issue of limitations of the type of weather by implementing separate encoders for each type of weather with a single decoder based on the CNN network. However, the problem of complexity and lack of practicality still exists.[1]

Due to transformer-based models' exceeding ability to extract global features and TransWeather's proven high adaptability and notably simplified architecture, TransWeather[1] was selected as the most suitable foundation for this research after a comprehensive comparison with other models. TransWeather is mainly acknowledged for its ability to handle multiple weather conditions using a single encoder-decoder framework, unlike many existing models requiring separate architectures specific to different weather types (a visualized comparison of this is demonstrated in Figure 3). In TransWeather, the "multi-head self-attention mechanisms" will match the queries with the "keys and values" from the encoder. Then, the 'decoded features' and the 'hierarchical features' from the encoder are combined and projected through the convolutional block. [1] Furthermore, another main structural feature we based on is the Intra Patch Transformer blocks (Intra-PT). Intra-PT blocks use the features extracted from the smaller patches from the original path. Therefore, Intra-PT "focuses on attention inside the main patches" to restore various weather-degraded images under different scenarios.[1] Details of this will be further explored in the paper. These capabilities made TransWeather the ideal candidate for constructing a more advanced a practical image restoration model, Self-guided Filter TransWeather (SFTW).

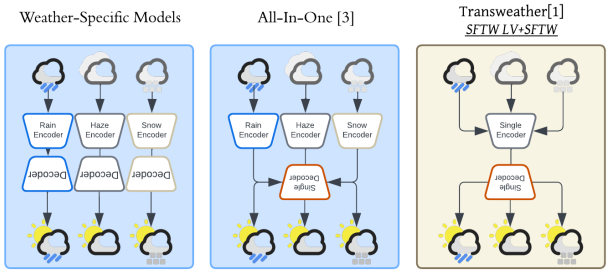

Fig. 3. Comparison of model structures, based on TransWeather's visualization [1]

Despite the pivotal advancements and advantages introduced by TransWeather[1], including its comprehensive ability to handle multiple weather scenarios in a simple network, there remain significant challenges when applying such models practically, especially in real-world, real-time situations. Further from TransWeather, the current 'State-of-Art' models all face limitations in terms of generalizability, computational efficiency for fast processing speed, and the ability to operate with real-time inputted data, such as live video streams or performance under complex and dynamic environmental conditions. Furthermore, while many models do indeed show superior performance in benchmark datasets, they fail to provide satisfactory performance when applied to live, real-world scenarios, mainly due to the noise artifacts, edge degradation, and the difficulty of training on large-scale, real-world datasets, due to the difficulty to create a dataset consisting of input-ground truth image pairs with a satisfactory amount for training.

B. Motivation

Given the limitations of this research context, this work aims to build upon the TransWeather model[1], and significantly push its practicality to a next level of this field. Self-guided Filter TransWeather (SFTW) and Live Video Plus SFTW (LV+ SFTW) aims to act as the pioneer to fill in the gap between the current stage of academic research of this emerging field of weather-degraded image restoration to a practical, appliable and useful model, that could actually benefit human informatics revolution, by introducing the innovations below:

Building on this foundation, SFTW is introduced with a trainable self-guided filter layer to reduce the noise artifacts and preserve edge details, as well as the replacement of the VGG 16 [41] pre-trained model with ResNet50 [40] to enhance the feature extraction ability. This model's practical functionality has also been extended with a live cam and video processing function, making it capable of real-world applications instead of simple images. This research mainly focuses on improving the output image quality, measured by PSNR and SSIM, while addressing the limitations of existing models, especially when processing complex weather scenarios. By utilizing TransWeather's efficient structure, SFTW's enhancements aim to improve the output image quality and the real-time performance and functionality in dynamic and complex environments.

Below is a summary of the key contributions:

1. *Trainable self-guided filter layer* addresses noise artifacts and enhances edge preservation. The original concept of a 'guided filter' [43] was used as a tool for edge-preserving smoothing through a guide image after processing. Although helpful, this concept also contradicts the ideology of practicality, as in practice, there would not be a clear image to act as a guide. Therefore, this concept is modified and trained in TransWeather's framework as a 'self-guided' filter.

2. *Live Video Plus additional function:* The addition of a live camera and video processing capabilities for real-time applications in the LV+ SFTW model, allows the user to be able to use this model to enhance reliability of images under real-world scenarios, when the time to upload images to the basic model is limited, or a video is required to be processed.

3. *The Transition from VGG16[41] to ResNet50[40]* in the model's architecture is implemented to optimize its capacity and ability to extract more complex features from weather-degraded images, further enhancing its robustness. While obtaining real-world datasets remains challenging for both dataset creators and model developers, the project still aims to explore techniques to train model focusing on real-world images rather than synthetic ones, for a further enhanced practicality performance.

In summary, SFTW aims to enhance the quantitative performance benchmarks of PSNR and SSIM and takes the initiative to directly and explicitly make image-restoration models more practical, scalable, and deployable in real-world applications instead of massively enhancing benchmark performances.

C. *Related Works*

Prior to Transformer's introduction in 2022, previous efforts to attempt removing specific weather types has been acknowledged and used as inspiration in this literature to improve areas where the original TransWeather lacks.

De-Hazing

In 2017, Li et al. Proposed an All in one dehazing network, named AOD-Net based on the CNN structure based on a " reformulated atmospheric scattering model" [13]. In 2022, Zhou et al. Proposed the EHA-Transformer model, with additions to previous methods by implementing a haze detector to distinguish regions and a haze-adaptive loss to increase stability of training. [15] In 2024, Luo et al. Proposed a method 'Dehazeformer'. [12] To resolve the problem of loss of global structural information when only using CNN to dehaze, and the complex structure of transformer. Dehazeformer is composed of a "double-branch collaborative nonhomogeneous image dehazing network. [12] In May 2024, Zhao, Xu and Liu proposed the 'TransDehaze model', integrating the advantages of both Transformer, and U-net architecture to "better preserve image models" and resolve the problem of loss or distortion of texture features in previous methods.[14]

De-Raining

In 2017, Fu et al. Proposed "DerainNet" specifically designed for removing rain streaks from a single image, using the CNN network architecture. The method achieved successful performance in terms of reducing computational time [20] In 2019, Zhang et al. Proposed a model based on CGAN (Conditional generative adversarial networks) by enforcing the constraint of the similarity between de-reained images and the fround truth image. This implements a regularization to enhance de-rained image qualities. [19] In 2019, Quan (RuiJie) et al. Proposed a Complementary Cascaded Network Architecture (CCN) to remove Raindrops and Rain streaks using a unified framework in one go. Furthermore, Quan et al. Created their own real-world dataset 'RainDS' by filming in real life, solving the issue of overfitting synthetic images. [21] In 2019, Quan (YuHui) et al. Introduced a "Double double attention mechanism that concurrently guides the CNN using shape-driven attention and channel re-calibration" specifically designed to remove raindrops on glass.[22]

De-Snowing

In 2017, DesnowNet was developed by Liu et al. Specifically aiming to remove translucent and opaque snow particles on the camera, by designing the multistage network. Liu et al. Also proposed a new synthesized dataset named snow100k, which is used in our dataset. [23] In 2020, Chen et al. Proposed the JSTASR model, "Joint Size and Transparency-Aware Snow Removal Algorithm Based on Modified Partial Convolution and Veiling Effect Removal" [25] In 2022, Cheng et al. Proposed a model named Snow Mask Guided Adaptive Residual Network (SMGARN) consisting of "Mask-Net, Guidance-Fusion Network (GF-Net) and reconstruction net" to solve the common problem of most models: unable to detect the special distribution of snow streaks.[24]

All-in-One Models

In 2020, Li et al. Developed the first all-in-one image restoration model, consisting of separate encoders for the 3 most common types of bad weather: Rain, Fog and Snow, with a single decoder restore any type of weather degraded image. The model also includes a discriminator, to "simultaneously assess the correctness and classifies the degradation type of the restored image". The dataset used to train the model, "Allweather" is also used to train our model initially. [3] In 2022 Valanarasu et al. Proposed the transweather model, with significant advancements to previous models in terms of efficiency, simplicity and adaptability. Transweather is a "transformer based end to end model with just a single encoder and decoder that can restore an image degraded by any weather condition." [1]

Guided Filters and Self Guided Filters

He et al. (2010) originally developed the Guided Image Filter, which became foundational in edge-preserving smoothing techniques [43]. In 2017, Jiang et al. Proposed a vision enhancement algorithm, designed to remove haze and fog from a single image. The proposed algorithm has the ability to "automatically adjusts weight coefficient according to various structure images", including a self guided filter at the same stage. [29] In 2019, Zhu and Yu proposed a variant of guided filter, the self guided filter, designed for denoising and edge-preserving degraded images with the lack of clear images as a guide.[26] In 2019, Gu et al. Proposed and improved current methods of Self Guided Filters for more adaptability by simplifying the complex networks of common Self guided filters.

Gu et al. proposed a "self-guided network (SGN), which adopts a top- down self-guidance architecture to better exploit image multi- scale information." [28] Zhang et al. (2022) and Kumar et al. (2021) introduced innovations in Hybrid Guided Filters and Self-Adaptive Guided Filters, respectively, for low light image enhancement and medical image denoising [44][45]. In 2021, Li and Wang proposed a method of DFRCN (Deep Fully Regression Convolutional Network, to resolve the problem of refraction of light and the mask underwater. During comparison, the method added an additional guided filter layer, which significantly enhanced the PSNR values.[5] In October 2023, Karacan proposed a "new Generative Adversarial Network (GAN)–based MFIF model". The model includes an encoder and decoder network with a Trainable Self Guided Filter to address the limitations of MFIF (Multi-Focus Image Fusion) [27]

Key Research Milestones

The field of weather-degraded image restoration has been through significant advancements over the past decade, driven by innovations in deep learning and image processing for example most models introduced within the 3 years has all evolved to a single model for all weather scenarios, compared to the first DerainNet[20]. The SFTW & LV+SFTW model builds on key works proposed in this field, selectively aiming addressing previous limitations while introducing new functionality for real-time applications. Figure 4 includes the 5 most critical milestones that have shaped and inspired the development of SFTW & LV+SFTW, with explanations of each.

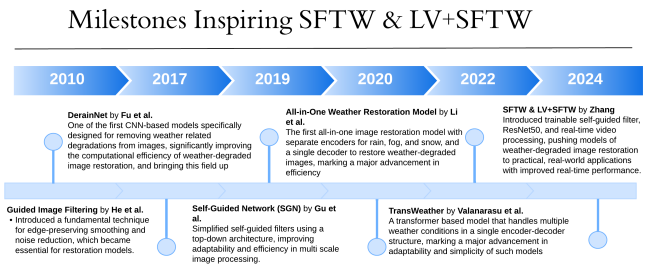

Fig. 4. Milestones Inspiring SFTW and LV+SFTW

D. *Hypotheses*

Null hypothesis and an Alternate hypothesis both provides the same concept of stating a hypothesis at start, but presented in a different method. Null Hypothesis is a statement which assumes there are no significant effect caused by a variable in an experiment, while an alternate hypothesis states a significant difference or effect, assuming the change does indeed have an impact. Since the core content provided on both hypotheses will be exactly identical, therefore ***I recommend only reading the Alternate Hypotheses.***

Alternate Hypotheses

1. The self-guided filter layer will significantly improve image quality, reducing noise artifacts and enhancing and smoothen edge preservation, as measured by PSNR and SSIM, compared to the original replicated TransWeather [1][6] model.

2. The replacement of VGG16 with ResNet50 will substantially enhance feature extraction, improving both image restoration accuracy and efficiency, particularly for weather-degraded images, as measured by PSNR and SSIM too.

3. The single encoder to decoder architecture based on TransWeather[1] will significantly improve performance in mixed weather scenarios, which will outperform the task-specific models in performance and inference speed.

4. The addition of real time video and camera processing capabilities will significantly enhance the model's practicality for complex environments such as autonomous driving and real-time surveillance algorithms, with afaster and higher quality image restoration output.

5. The overall upgrade, including the self-guided filter layer and the transition to ResNet50, will significantly reduce computational costs and improve inference speed, making the model more efficient.

Null Hypotheses

1.The self-guided filter layer will not significantly improve image quality, reducing noise artifacts and enhancing and smoothen edge preservation, as measured by PSNR and SSIM, compared to the original replicated TransWeather [1][6] model.

2.The replacement of VGG16 with ResNet50 will not substantially enhance feature extraction, with no impacts to both image restoration accuracy and efficiency, particularly for weather-degraded images, as measured by PSNR and SSIM too.

3.The single encoder to decoder architecture based on TransWeather[1] will not significantly improve performance in mixed weather scenarios, which will not be able to match the performance to the task-specific models in performance and inference speed.

4.The addition of real time video and camera processing capabilities will not significantly enhance the model's practicality for complex environments such as autonomous driving and real-time surveillance algorithms, with a slower and lower quality image restoration output.

5.The overall upgrade, including the self-guided filter layer and the transition to ResNet50, will not significantly reduce computational costs and improve inference speed, making the model less efficient.

II. Methodology

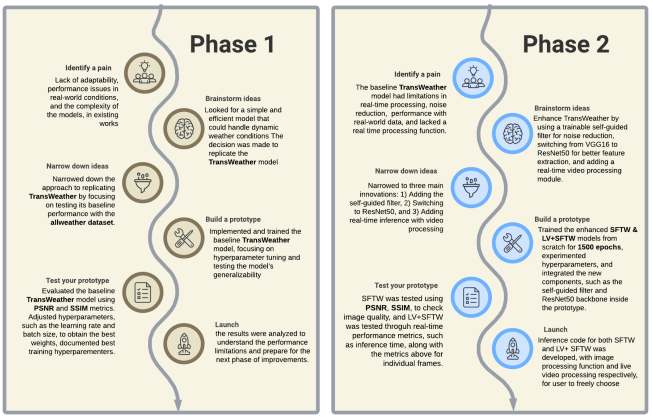

Fig. 5. Comparison of project phases

Project Procedure

The project procedure is split into two main phases: Replication and Innovation. Each stage is critical to the overall development and focuses on a distinct part of the project's evolution. It begins by replicating the original TransWeather model and then developing significant upgrades to improve its practicality and effectiveness. This section only covers the Experimental thinking procedure and process, results are separately documented in 'Results' section. A simple comparison of the project thought procedures are summarized in figure 5, below.

A. *Phase 1: Preparation and Replication*

Preparation

Before beginning the model replication, extensive literature research and reviews of related works were conducted to understand the current state of weather-degraded image restoration models. The objective of this stage of this phase was to identify a baseline model that had proven successful in both Practicality, as considered in the ability to handle multiple complex weather scenarios through a single model, and portability, as measured in computational effectiveness and speed. As mentioned in section 1.3, TransWeather[1] was identified as the most suitable candidate as a baseline due to its superior ability to handle multiple dynamic weather scenarios with a single encoder-decoder architecture. This model also outperformed existing models through its remarkable balance between simplicity and adaptability, which makes it the foremost suitable foundation as a base for further innovations. [1]

During this stage, another research objective was to identify existing limitations in weather-restoration models and identify a base. Limitations were identified, including the lack of real-time processing capabilities and reduced performance in dynamic environments and real-world images. These identified challenges and limitations were critical in shaping the next phase of the innovation, as the central core focus of SFTW was reconfigured to optimize Practicality instead of further pursuing an upgrade of quantitative metrics. More research on several influential works that are not necessarily related to weather restoration further inspired improvement, particularly on incorporating self-guided filtering and enhanced feature extraction models [16][17][18][26][27][28][29][40][43].

Source Acquisition and Environment Set Up

To replicate the TransWeather model[1], the dataset used for training the model was kept consistent as the original, the 'Allweather' dataset [3][1], which is linked from the official TransWeather GitHub repository [1][6]. The dataset contains a mixture of real-world and synthetic weather-degraded images of rain, snow, and fog image pairs, corresponding with their clean images, which serve as ground truth (gt) - the dataset will be further explored in the 'Data' section. The original TransWeather source code was also downloaded and analyzed to ensure consistency in the initial replication experiments[1]. The utils module [6] was also incorporated into the framework, which had a critical role in data loading,

augmentation, and organizing the dataset for training.

An online server was rented and utilized for training and experimentation with the following hardware specifications: Miniconda with Python 3.1.0 running on Ubuntu 22.04 OS, equipped with an NVIDIA RTX 3090 GPU (24GB memory, CUDA 11.8). The CPU is a 14 vCPU Intel Xeon Platinum 8362, clock speed at 2.80GHz, with 45GB of RAM and a hard disk setup that includes a 30GB system disk and 50GB data disk, overall priced at1.58CNY per hour. The environment was initialized with dependencies such as timm 0.3.2, mmcv-full 1.2.7, torch 1.71, torch-vision 0.8.2, and OpenCV 4.5.1.48, consistent with the requirements of the TransWeather model [1][6]. These server and environment settings remained constant across the replication and innovation phases.

Training 1: Baseline Model Training (TransWeather) In the first stage, the baseline TransWeather model[1] was trained on the all-weather dataset[3] for three consecutive experiments with only minor changes to the training hyperparameters for each iteration. The objective of the training and experiments at this stage was to finetune the learning rate, batch size, and number of epochs to find the most suitable configuration and training settings for the baseline model to work at its optimum [1][6]. Another objective was to obtain the 'best' weights file for a preview of inference at the early stages, providing a visualization of the model's effects at an early stage for visual evaluation to determine future innovations. Experimenting with these parameters identified the 'best' model weights, providing a solid foundation for the next phase of innovations.

For each experiment, The default learning rate 1e-3 only varied slightly between 1e-4 and 1e-2 to observe how the model's convergence rate was affected. The batch size was only adjusted between 16 and 32 (powers of 2) to maintain computational effectiveness. The epochs decreased throughout the experiments since each new experiment trains the base of the previous experiment; therefore, reducing the epochs will massively reduce computational costs by reducing the rounds of useless training iterations. During these experiments, The batch size of 16 and a learning rate 1e-3 provided the most stable results, preventing overfitting while maintaining sufficient learning throughout the model; further details of this stage will also be further explored in the 'Experimental Results' section.

Additionally, it is essential to note that the inference (processing) code was not included in the source code, as each developer may prefer to customize the inference process for different purposes, ie. This is for optimized output quality or SFTW's video and live functionality. Therefore, for this stage, the inference was only kept as a default image processing function, used primarily to visualize outputs for visual evaluation without adding specific tasks related to real-world applications, as this baseline model will not be applied for the final SFTW model for utilization.

B. Phase 2: Innovation and Inference

Upgrades and Modifications The SFTW project entered the innovation stage after successfully replicating the TransWeather model[1][6]. This phase includes the introduction of significant upgrades, explicitly targeting and addressing the limitations identified earlier, such as a lack of real-time processing capabilities, noise reduction, and model generalization to real-world applications. The primary innovations introduced were the trainable self-guided filter into the framework, a switch from VGG16 to ResNet50, and real-time video processing capabilities.

Model Architecture Modification One of the key innovations introduced in this stage was the addition of a trainable self-guided filter layer to the TransWeather architecture. This filter is designed and implemented in this model to serve as a component specifically for reducing noise artifacts and preserving and smoothening important edge information, which is commonly degraded in weather-affected images. The self-guided filter enhances image clarity, especially in areas with densely packed edge information, improving PSNR and SSIM scores. Despite achieving decent quantitative benchmarks, existing works have not identified this component.

Another modification to the model architecture was substituting the model's feature extraction backbone from the VGG16 pre-trained model[41] to ResNet50[40]. ResNet50 was selected for its superior capability in handling and extracting features from complex images and improving the model's generalization, especially in weather-degraded images where fine details may be lost during degradation. This switch helped the model to extract deeper features from weather-affected images, thereby improving the overall quality of restored images through thorough training.

LV+ SFTW

A significant step towards practicality in this phase was the development of a real-time video processing function to the SFTW model inference, as the LV+SFTW model. This feature is critical for applications in real life, such as real time autonomous driving systems, and surveillance cameras, to process and restore the quality of a degraded video, where images must be processed in real time, under dynamic weather conditions. This function was integrated into the upgraded model, LV+SFTW's Inference pipeline, based on SFTW's inference, allowing the model to process live video inputs with minimal delays. Key details of the integration and utilization will be introduced in the 'Model' section. This upgraded model not only improved the model's practical applicability but also established it as one of the pioneering models in this field, capable of real-time weather-degraded image restoration.

Enhanced Model (SFTW) Training

With the architectural modifications successfully implemented in the model framework, the SFTW model was trained on the same all-weather dataset[3][1], ensuring consistency in the training dataset for comparison while focusing on improving performance with the innovations. The training process followed a similar protocol to the TransWeather base but with enhanced features, including a self-guided filter layer and ResNet50 backbone now incorporated. Minor changes to the hyperparameters were modified, similar to the first phase, including finetuning the learning rate, batch size, and epochs to adapt to the enhanced architecture.

Due to the shift from VGG16[41] to ResNet50[40], the model had to be trained from scratch, with no loaded weights, for a total of 1500 epochs, as the weights obtained from the original TransWeather[1] model in phase 1 training, with the VGG16 pre-trained model was not compatible with the new ResNet50 framework. The learning rate was still initially set to the default value of 2e-4 and was not manually changed during the training process, leaving it to adjust during the training process automatically. This method ensures that the model can effectively learn from the Allweather [3][1] dataset without being constrained by the pre-trained weights that do not align with the new architecture.

Compared to the training process in Phase 1, which aimed to fine-tune the hyperparameters and establish a working baseline, this phase targeted obtaining the final working model for direct usage and calling in the inference code. The objective was, therefore, to achieve the highest possible quantitative measures of PSNR and SSIM, ensuring that the model would be guaranteed high-level performance for real-time and practical applications once the inference objectives were called. This phase represents the culmination of the training process, where the objective was to refine model performance to its peak and, therefore, deploy it for the inference process.

Evaluation

Following the successful training process of the enhanced models, randomly selected examples of input images were passed through the inference pipeline, where in the basic model, SFTW, images would be separately processed and saved, while in LV+ SFTW the real-time video or uploaded video frames will be processed too. The results were used for visual evaluation, of the effect of the model, as seen through human eyes, compared with the PSNR and SSIM values obtained in the training log. Additionally, EVA (Exploratory Visual Analysis) was also conducted, to compare the clarity and edge preservation of the enhanced model outputs, with the input image, and the ground truth.

The LV+SFTW model, with the addition of real-time video processing was tested by evaluating the latency (delay) and processing speed of each separate frame, which can be seen as separate images processed in a batch, while quality is measured in the method above.

III. Proposed Model: SFTW

Overview of Model Architecture

As repeatedly mentioned, SFTW and LV+SFTW build on the original TransWeather architecture[1] as a base, with significant innovations designed to enhance quality and practicality with portability in weather-degraded scenarios. The original TransWeather model[1] utilizes a single encoder-decoder framework and uses VGG16 as its backbone for feature extraction. This baseline model has been upgraded and enhanced with the following essential modifications:

1.Integration of a trainable self-guided filter layer for noise reduction, edge preservation, and smoothing.

2.Switch from VGG16 to ResNet50 for a more profound and practical feature extraction ability.

3.The addition of a real-time video processing module allows real-time restoration of weather-degraded images for dynamic applications, such as autonomous driving and surveillance.

These improvements make the model more practical for real-world use, offering better performance in terms of image quality and speed. Below are explained the details of each component of the model, including the detailed flowchart diagram of the whole model architecture, as shown in Figure 6 (Next Page).

Self-Guided Filter: Definition and Implementation

The trainable self-guided filter layer is one of the primary innovations introduced in the SFTW and LV+SFTW models. This filter is critical under this context in the model, improving image restoration by reducing noise artifacts while preserving the essential structural details, such as edges and corners, which are commonly degraded by weather effects. Unlike traditional guided filters[43], which require a separate guidance image (a guide), the self-guided filter can learn the optimal filtering parameters directly from the input data in the whole model network and achieve a filtering effect in the output. The self-guided filter is defined mathematically as:

$$I' = A(I) \cdot I + B(I) \tag{1}$$

Where: I' is the output image, I is the input image, A(I) and B(I) are the learned coefficients to adjust the filtering process

In this implementation, the self-guided filter is applied within the model as a part of the overall network architecture instead of a separate component outside the whole (...

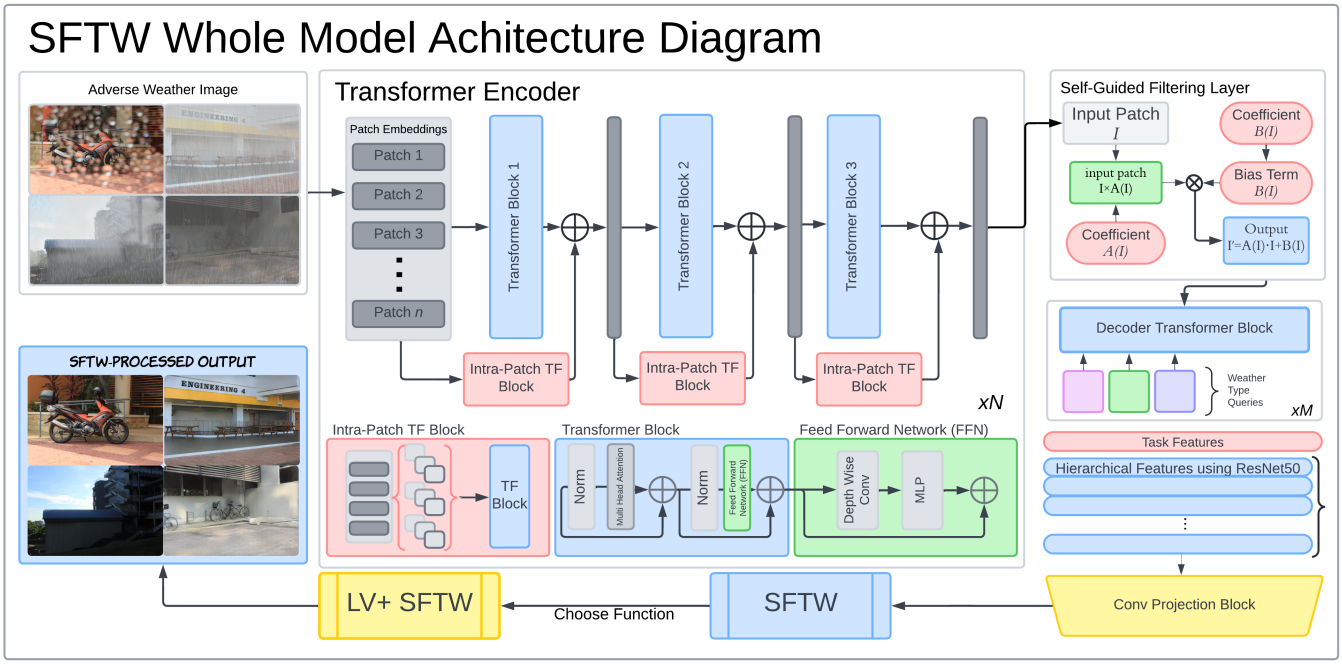

Fig. 6. SFTW Whole Model Architecture Diagram

...continued) model architecture before the output. Its parameters are also trained alongside the rest of the model through 'backpropagation.' The self-guided filter works by learning to smooth regions of the image where noises are present while preserving edges and textures, which is critical for accuracy and enhanced quality restoration of a weather-degraded image.

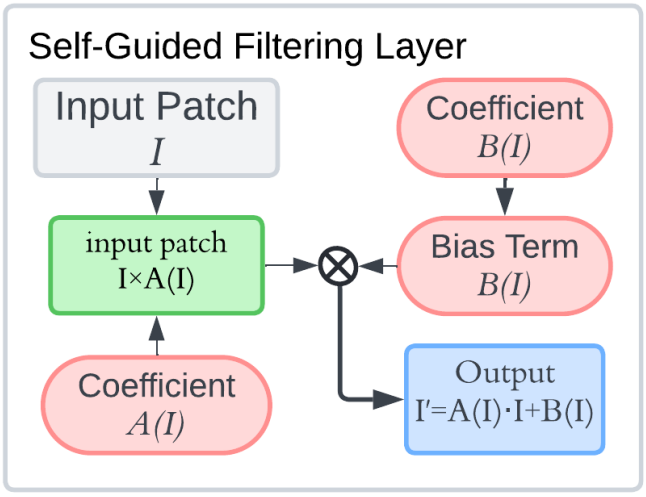

Fig. 7. Self-Guided Filter in SFTW

The self-guided filter operates at different levels within the network. It is beneficial when mitigating artifacts such as rain streaks, fog, or snow, which typically degrade the images' texture, contrast, and appropriate RGB values. The self-guided filter now helps ensure that the restored image maintains clarity and structure, even after the restoration under complex weather conditions. The diagram of the self guided filter is visualized, in figure 7.

A. *Transformer Encoder [1]*

The core component of the SFTW and LV+SFTW is the transformer-based encoder-decoder architecture, which is directly inherited from the original TransWeather model[1]. This Model architecture allows the model to process images 'hierarchically,' extracting all high-level and low-level global features to effectively restore the weather-degraded image through a single encoder-to-decoder network adaptable to all dynamic weather scenarios.

Transformer Encoder

Before the input is passed into the transformer encoder layer, the input image is pre-processed and divided into smaller patches called patch embeddings. The encoder processes these patches through multiple stages, each containing a transformer block that includes multi-head self-attention layers and feed-forward networks. The attention mechanism allows the model to focus on the essential features of the image while the feed-forward layers refine the extracted features.

The process within each transformer block is defined as:

$$T_i(I_i) = FFN(MSA(I_i) + I_i) \quad [1]$$

Where:

- $T_i(I_i)$ represents the transformer block,
- $MSA(I_i)$ is the multi-head self-attention applied to the input I_i,
- FFN is the feed-forward network used to refine the features.

In simple terms, the transformer encoder's primary use is to produce (extract) the hierarchical feature representations, critical for restoring the fine detail in the images affected by complex weather conditions, such as rain, fog or snow. The flowchart of components in the encoder, is shown in figure 8.

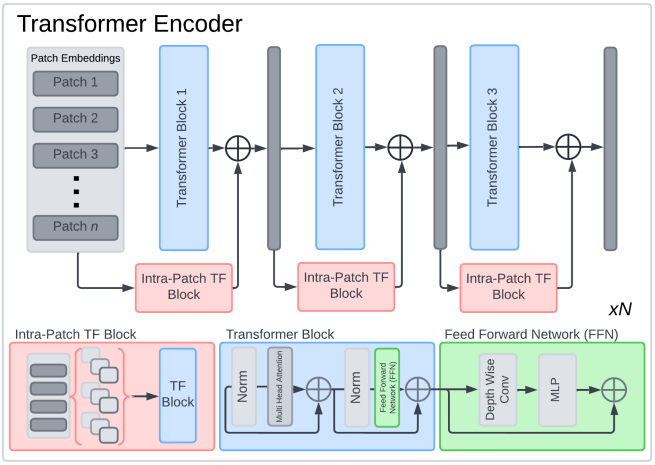

Fig. 8. Transformer Encoder in SFTW and TransWeather[1]

Transformer Blocks The transformer blocks in the transformer encoder are critical components in the model and are responsible for extracting the global features from the input image patches. Each of the Transformer Blocks consists of the following components: Multi-head self-attention; Norms, and a feed-forward network (FFN) [1] :

Multi-Head Attention (MSA) The multi head attention mechanism in the transformer blocks allows the mode to focus on different regions of the image patches, calculating the relationships between the different positions, as measured in pixels, in the input patches. The multi head attention mechanism helps to capture the 'long range' dependencies and important contextual information in the input image patch. This attention function is expressed mathematically as:

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad [1]$$

Where:

- Q, K, V represent the Query matrix, Key matrix, and Value matrix, respectively.
- d_k is the dimension of the keys.

The result is a weighted sum of the values, where the weights is the relevance between the query and the key.

Norm (Layer Normalization) and Feed Forward Network (FFN) After the multi-head attention component, a normalization layer stabilizes the training process. This additional layer ensures that the input to the next layer has a mean of zero and a standard deviation of one, which helps to enhance the training efficiency.

Each transformer block also includes a feed-forward network (FFN), which operates independently on each pixel of the individual inputted image patch. The FFN consists of 2 fully connected layers with a nonlinear activation function (ReLU) between each. The FFN's responsibility is to transform the features, initially weighted with the attention features, into more abstract representations, allowing the model to learn more complex relationships within the image data.

In ResNet Architectures [40], skip connections or residual connections are used in the transformer block, which allows the model to bypass the multi-head attention and feed-forward network (FFN) layers, making it easier to train deeper networks by mitigating the 'vanishing gradient' problem. The skip connection also adds the original input, I, to the output of the FFN, ensuring that the model keeps the original information while learning any additional transformations fed.

Intra-Patch Transformer Block Encoder

The Intra-Patch Transformer Block (intra-patch TF block, as in Figures 6 and 7) is applied at each encoder stage. The Intra Patch Transformer blocks will operate on the sub-patches from the original image patches, which helps capture even smaller and finer details and remove fine degradations, like small rain streaks, light snow, or translucent raindrops on the camera. Figures 6 and 8 above show the Intra Patch transformer blocks in red.

The Intra Patch Transformer block's operation can be defined mathematically as:

$$Y_i = MT_i(X_i) + \textit{IntraPT}_i(P(X_i))$$

Where:

- Y_i Is the output at stage i
- $MT_i(X_i)$ Is the main transformer block output
- $\textit{IntraPT}_i(P(X_i))$ is the *intra-patch* transformation applied to the sub-patches created from the input patches.

B. Transformer Decoder [1]

The transformer decoder is responsible for restoring the clean image based on the hierarchical features (Keys and Values, K, V) generated from the transformer encoder, filtered through the self-guided filter. The original TransWeather team introduced a revolutionary component in the transformer block of the decoder, weather type queries, which are the learnable embeddings representing different weather conditions, including rain, fog, and snow, represented as Q

[1]. The queries now allow the model to identify and directly address the specific weather degradation specified in the image, making the model adaptable to any complex weather conditions without requiring separate models or decoders for each one, achieving 'Portability.'

In the decoder transformer block, the multi-head attention mechanism (MSA) operates on these weather type queries (Q) and the features extracted from the transformer encoders as (Keys, K and Values, V). The attention mechanism also calculates the relevance of each different part (pixel) of the image based on the weather type identified. The newly drawn diagram of the decoder flow is shown in Figure 9, based on the diagram in TransWeather[1], with a self-guided filter component added.

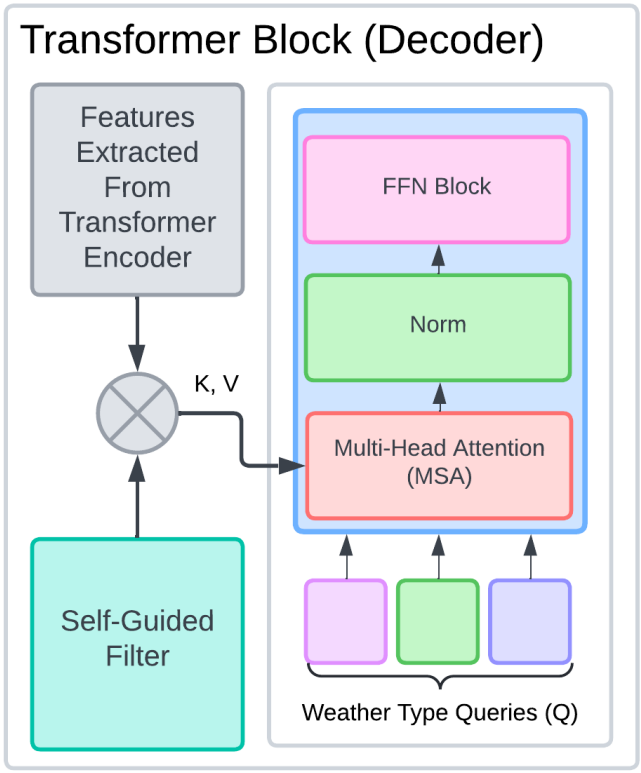

Fig. 9. Decoder Transformer Block flow, with QKV

After applying the multi-head attention and passing through normalization and a feed-forward network(FFN), the hierarchical features from the transformer encoder and decoded features from the transformer decoder are combined to an abstracted output and then passed to the convolutional projection block to produce and output the final processed image.

C. *Convolutional Projection Block*

The convolutional projection block's responsibility is to transform the task-specific features generated and fed by the decoder into the final restored image for presentation to the user. This block will receive two inputs in total: the hierarchical features extracted by the encoder and the weather-specific features decoded by the transformer decoder, and it operates on the sum of these.

The critical functions of the convolutional projection block include: 1.Combining the task-specific hierarchical features from the encoder and the decoder by merging the output from the decoder with the hierarchical features from the encoder to improve the quality of the restored image.

2. Upsampling layers are used to resize (post-process post-process) and normalize the image back to its original dimensions for a presentation from the preprocessed size of the image. Furthermore, some of the transformer layers earlier reduced the image resolution, too, to speed up the processing efficiency.

3. Convolutional layers (3) are applied to the combined hierarchical features, restoring the final image quality.

4. The final convolutional layers use the Tanh activation function to output the final clean image, which is HxWx3 and has three dimensions, with the three representing the three color channels: R, G, and B.

The projection block also uses skip connections, as explained in the encoder section, to link the encoder and decoder stages, which overall helps refine details and enhance the quality of the output image. The process of this layer is visualized in Figure 10, whereas in Figure 6, it is represented by the yellow trapezium.

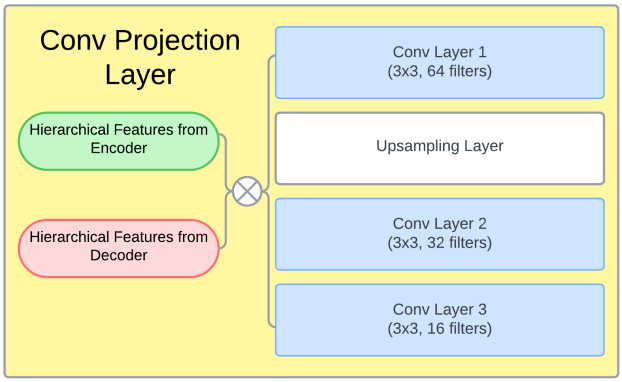

Fig. 10. Conv Projection Layer

Training and Loss Functions

The SFTW model was trained from scratch in phase 2, after phase 1 of the replication simulation. In phase 2, the model was trained over 1500 epochs using the same allweather dataset [1][3], which includes both synthetic images and real

life images representing various complex weather degraded image and ground truth image pairs. The initial learning rate was set at default of 2e-4, and was automatically dynamically adjusted during the training process, this will be further explored in the 'Experiments' section.

Loss Functions:
The training process' evaluation process used a combination of two loss functions, 1. Smooth L1 loss and 2. Perceptual Loss.

Smooth L1 Loss: This loss function's primary objective is to measure and aim to minimize the pixel differences between the restored image and the ground truth image during training. The smooth L1 loss function is defined mathematically as:

$$L_{\text{smooth}L_1} = \begin{cases} 0.5E^2 & \text{if } |E| < 1 \\ |E| - 0.5 & \text{otherwise} \end{cases}, \quad [1]$$

Where $E = \hat{I} - G$ represents the error between the predicted image $\hat{I}$ and the ground truth image G.

Perceptual Loss: The perceptual loss function is used to preserve the structural similarity of the image, by comparing the high-level features between the predicted and ground truth images. The features are extracted using the pretrained ResNet50 model (originally VGG16), and is expressed mathematically as:

$$L_{\text{perceptual}} = L_{\text{MSE}} \left(\text{ResNet50}(\hat{I}), \text{ResNet50}(G) \right)$$

Where L_{MSE} is the mean squared error of the features between the predicted image $\hat{I}$ and the ground truth G.

The total loss function used during training is the weighted sum of both the Smooth L1 loss and Perceptual Loss together, expressed mathematically as:

$$L_{\text{total}} = L_{\text{smooth}L_1} + \lambda \cdot L_{\text{perceptual}}$$

Where the variable λ is used to control the balance between the two losses, by changing its weight.

VGG16 to ResNet50
In the original TransWeather model[1], VGG16 was used as the pre-trained model for feature extraction. VGG16[41] is a convolutional neural network (CNN) consisting of 16 layers for feature extraction; TransWeather used the 3rd, 8th, and 15th layers. It is known for extracting low-level features, such as edges and textures, instead of fine weather details. However, VGG16 is limited in its ability and capacity to extract complex high-level features, making it challenging to restore weather-degraded images.

Considering the identified limitations, the SFTW model replaced the VGG16 pre-trained model with ResNet50 [40], a more advanced network with new features such as residual connections. The residual connection now helps to mitigate the 'vanishing gradient' problem [40], which allows the deeper networks to learn more effectively and deeper features from the input image batches. The residual connections can be mathematically expressed as [40]:

$$F(x) = H(x) + x$$

Where:
- $F(x)$ is the final output.
- $H(x)$ is the residual function learned by the model.
- x is the input to the residual block.

After the implementation of ResNet50, the model is now expected to extract more complex and detailed features from the weather-degraded images, which improves the overall image restoration quality, especially in difficult conditions like heavy rain, fog, or snow.

SFTW - Image Inference
Image inference is relatively simple, the model processes the weather degraded image inputs directly and restores them, and saves them into the directory the user would like. The input image is first resized and preprocessed into appropriate dimensions, then passed through the model's main network (encoder and decoder), where the specific weather features are processed and extracted, and refined by reducing noise and enhancing edge details through the self guided filter layer. The model then outputs a clean and restored image, which is normalized and resized to match the size of the original input image. Processing an image is computationally efficient, with a very small delay of 0.48 seconds with self guided filter, and 0.14 without[1], which makes this suitable for high resolution, single images, most likely for enthusiasts, or for users that only require fast inference of a single image.

LV+ SFTW - Video and Live Video Inferencing
If the input video is a pre-recorded video, the video is first loaded and each frame is extracted. For a live video, frames are captured directly from the camera's continuous input of frames captured. Each frame is then pre-processed and passed through the model, just like a normal image inference, through the Transformer Encoder, Decoder, and Self-Guided Filter Layer. Once the frames are processed, they are either recombined to a continuous video, and saved to the directory the user would like, or as individually restored frames. The final output will match the original resolution and frame rate, with a weather-free video for real-time applications.

IV. Data

A. *Allweather*

In SFTW's training process, the Allweather dataset[3][1] was the primary dataset to train the model to restore weather-degraded images. The Allweather dataset is designed explicitly for all-in-one types of related image restoration

models, where all types of dynamic and complex weather scenarios are included, for models like All-in-One[3], TransWeather[1], consisting of a total of 18,069 image pairs. These pairs include both clear types of images, named the ground truth (or gt), with their corresponding degraded images, at the same angle shot, named the input, representing various extreme weather types of rain, snow, haze, and their mixtures. The dataset consists of synthesized (Outdoor Rain and Snow 100K) and real world images (RainDrop). For some images, the weather degradation is artificially applied to the explicit images to achieve a degrading effect manually. It is essential to note the difficulty in obtaining and creating a full real-world dataset; such difficulties will be explored later in this section.

Specifically, the Allweather dataset[3][1] is composed of the following subsets:

1. **Outdoor Rain**: This subset includes *9000* image pairs, where the clear outdoor images (ground truth) are realistically shot in random scenarios and are degraded by manually synthesizing various intensities of rain and haze onto the image. These synthesized rain and haze effects are designed to mimic extreme weather conditions in real life, ranging from light rain streaks to heavy haze covering the whole image, meaning there are fewer diverse scenarios than 9000.

2. **Snow100K [23]**: Another *8000* image pairs were copied from the Snow100k dataset [23], which also contains realistically shot outdoor image pairs, where the degraded images are degraded from synthesized snow but don't range within different densities. Instead, they contain 8000 diverse scenarios with completely different snowfalls. The snow effects can also randomly cover elements from the image, realistically replicating snowfall conditions.

3. **Raindrop**: This is the smallest subset of the Allweather dataset; it only contains *1069* real-world image pairs. It explicitly focuses on the scenario with raindrops in the camera shot under clear weather. Unlike the other subsets, where the degraded shots are entirely artificial, this subset includes authentic images in which raindrops cover the camera lens. However, due to its limited size, this subset is significantly less impactful in training the model on performance for this real-world scenario. Reasons will be further analyzed in 'Results Analysis.'

When the dataset is implemented in the training process, the dataset is manually split into training and evaluation sets into the ratio of 9:1 [6]. Of the 18069 image pairs, 16263 were used for training and 1806 for evaluation (testing). This ratio is specifically designed so the model is trained on most of the data, ensuring the amount is specific enough to cover most scenarios to prevent underfitting. The remaining 10 percent of the testing data also allows for relatively thorough and accurate testing and evaluation of unseen data to prevent overfitting of the seen data. The dataset's synthetic element makes it easier to manipulate and standardize the weather conditions applied to the images, simplifying the overall training process.

All the information about the Allweather dataset above is visualized in Figure 11 below.

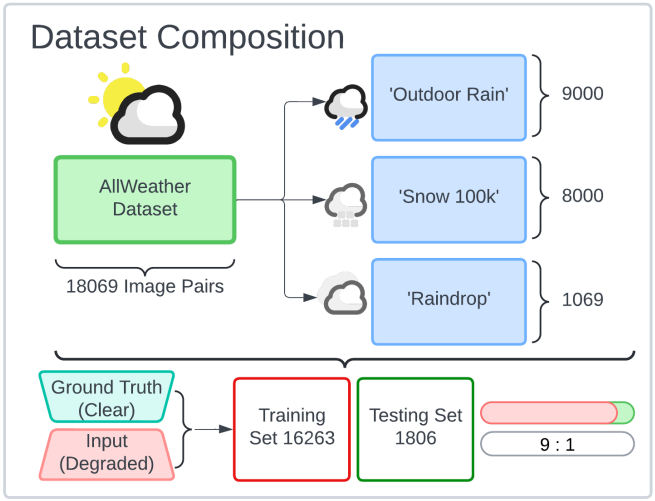

Fig. 11. Dataset Composition

B. *Limitations*

While the Allweather dataset provides a valuable and thorough database for training weather restoration models, it must fully cover authentic, real-world graded images' dynamics, complexities, or realistic feature patterns. A significant challenge in this research and this entire field is the lack of large-scale real-world datasets that consist of clear (ground truth) and degraded (input) images. Ideally, the image pairs must be shot under the same scenario. For example, to create a full-size real-world dataset, the developer would be required to shoot the same scene in both clear and adverse conditions without any slight changes to the camera's position, angle, or any external factors, such as the position of any artifacts, while having to wait for extreme weather to come, or go. Objects may move, lighting may change, and even minor camera vibrations can alter images, introducing artifacts that hinder the model's ability to learn the correct features between degraded and clear images, reducing the accuracy of the overall training process.

The difficulty in maintaining the exact consistency between the clear and degraded images under actual real-world conditions explains why there are currently no large datasets of real-world images. Therefore, most available datasets, including Allweather, only rely on synthetic images to generate degraded images from clear images with artificial features. While these synthetic datasets are helpful for training and preventing underfitting, they cannot fully cover the dynamic unpredictability and variability of real-world

weather conditions. This leads to the model's limited performance when actually implemented due to its overfitting on synthetic datasets. In the future, developing large-scale real-world datasets will be the most crucial step in promoting practicality and advancement in this specific field.

C. *Evaluation Metrics*

Before the Results section, it is essential to understand the two key metrics used to evaluate the output image quality, PSNR (Peak Signal to Noise Ratio) and SSIM (Structural Similarity Index Measure), which were briefly mentioned previously.

Peak Signal to Noise Ratio (PSNR)

· **Purpose**: The metric PSNR is used to measure the ratio between the maximum signal level (Clear Image) to the noise present in the image, which affects the image quality. This metric is crucial in this research to evaluate how well the model restores weather-degraded images to its original best quality.

· **Explanation**: PSNR is expressed as decibels (dB) and is calculated using the mean squared error (MSE) between the original and ground truth image (gt) and the restored image. A higher PSNR value would indicate better image quality, which indicates that the restored image is closer to the original image, with minimal noise or errors present. This metric's calculation is expressed as below:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right)$$

Where: MSE represents the Mean Squared Error between the output and ground truth.
255 is the maximum amount of pixel value of an 8-bit image

SSIM (Structural Similarity Index Measure)
· Purpose: The Structural Similarity Index Measure (SSIM) is used to measure the similarity of structure, luminosity, and contrast between the original (ground truth) and the restored image. It is also used to evaluate the perceptual quality of the image restoration, and measures structural degradation or differences more effectively than PSNR.

· Explanation: SSIM will provide a value between 0 or 1, where 1 indicates a perfect structural similarity score to the original image, normally meaning an identical image. It also considers the differences in texture or contrast, which makes it much more thorough and sensitive than the PSNR metric.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Where:

- μ_x and μ_y are the means of images x and y.
- σ_x^2 and σ_y^2 are the variances of images x and y.
- σ_{xy} is the covariance of x and y.
- c_1 and c_2 are constants to stabilize the division.

V. RESULTS

This section will document the experimental results recorded during the two research phases mentioned in the "Methodology" section, with an analysis. The first phase replicates the original TransWeather model as a baseline, and the second phase introduces innovations such as the Self-Guided Filter TransWeather model (SFTW). Both are evaluated using the metrics introduced above. The obtained results will also be compared with other existing methods mentioned in the TransWeather research [1].

A. *Phase One: Baseline Model Results*

Objective: The purpose of this phase was to fine-tune hyperparameters, such as batch size and learning rate, to optimize the model's final results, stability, and efficiency. The model was trained for a thorough number of epochs until the PSNR and SSIM scores on the validation set had stabilized at their best values. The goal of this phase was also to ensure that the model achieved higher scores than existing methods and maintained its original results.

The original TransWeather [1][6] model was used in Phase One, trained on the original *Allweather* [3][1] dataset. The main objective of this phase was also to *optimize hyperparameters and acquire a reliable hyperparameter configuration for Phase Two training and future research.*

The results from the first phase, replicating the TransWeather model, achieved a highest **PSNR of 29.52 dB** and an **SSIM of 0.8965** on the validation set (with the best result from Experiment 2 in three trials).

B. *Phase Two: SFTW Model Results*

The objective of Phase Two was to introduce key innovations to the architecture to improve the results of the model replicated in the previous phase and enhance the comprehensiveness of the model. The Self-Guided Filter TransWeather (SFTW) introduces a self-guided filter layer and substitutes the feature extraction base from VGG16 to ResNet50. The results of this phase show progress and further justify the importance of these innovations.

PSNR and SSIM: The SFTW model achieved a PSNR of 29.71 dB, an improvement over the 29.52 dB from Phase One, with an SSIM of 0.8962, which is almost identical to the baseline. Although the differences are small, the context of these results is significant. The improvement in PSNR with the same SSIM as in Phase One suggests that the SFTW model not only matched the baseline (still significantly higher than other existing models) in processing key features, but

also surpassed it in certain aspects, especially noise removal.

The best result recorded during training was obtained on September 26, 2024, at 00:50:11. Each training epoch took 283 seconds and a total of 1500 epochs were completed. The highest **PSNR achieved was 29.71 dB**, and an **SSIM of 0.8962.** The initial learning rate was set to the default value and automatically optimized during the training process.

Phase	Method	PSNR	SSIM
Phase 1	Replication	29.52	0.8965
Phase 2	Innovation	29.71	0.8962

As mentioned above, the aim of the SFTW model was to improve overall generalization, comprehensiveness, and practical performance of the model, rather than simply optimizing quantitative metrics. Although the training efficiency was slightly lower due to ResNet's deeper architecture (50 layers compared to VGG16's 16 layers), the deeper architecture of ResNet allowed the model to learn more complex features, which are clearly beneficial in practical complex weather conditions.

C. Exploratory Visual Analysis

Due to the large volume of data within the dataset, a complete analysis of all processed output would be unfeasible; this section randomly selected one processed image for Exploratory Data Analysis (EDA) and Exploratory Visual Analysis (EVA), comparing the degraded input image, the restored image, and the corresponding ground truth. The selected image is a piece of the set in dense fog conditions. It is analyzed using RGB histograms, HSV color space, LBP texture analysis, and two edge detection methods: Roberts and Susan edge detection. All analytical methods are used to demonstrate the effectiveness of the SFTW model in denoising, edge preservation, texture recovery, and color/brightness restoration.

Raw Output Image Comparison

Haze + Rain	Input	Ground Truth (gt)	SFTW
Images			

Fig. 12. Raw Output Image Comparison

The input image is degraded from fog and rain, which obscured the details, reduced contrast, and blurred edges of the object. The gt image is clear and vivid, with sharp edges and clearly defined structures. After processing with SFTW, the restored image based on naked eye observation is almost the same as the ground truth. Details are recovered, contrast is restored, and colors are saturated and natural. The results of the analysis are shown in Figure 12; subsequent images follow the same layout.

RGB Histogram

RGB color spaces visualize images of three color components: Red, Green, and Blue. Each color channel is represented by a value from 0 to 255, indicating its color intensity. [33] In the context of weather removal, this index describes how weather conditions may affect the color intensities of each channel. It visualizes how restoring the image can improve/restore the image's contrast and adequate visibility.

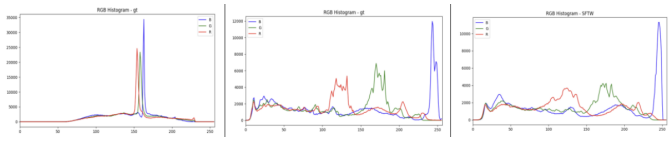

Fig. 13. RGB Histogram Analysis

Compressed distributions are seen in the input image for all 3 channels, especially at low and high brightness regions, which indicates a lack of diversity and dullness due to fog and rain features. The ground truth histogram indicates a balanced distribution with multiple peaks, reflecting vivid and diverse colors with clear contrasts. The SFTW restored results also indicate a close restoration with regard to the original image: the peak values increased, showing the restoration of brightness, while the color spreads widen, outwards, reflecting the improved saturation and contrast.

HSV Color Space Analysis

The HSV color space separately outputs the image's 3 other channels besides RGB, represented by three key terms: H for Hue, S for Saturation, and V for Value. In this index, Hue represents the color type, measured in degrees; Saturation indicates the contrast of the color; and the Value to measure the brightness. [35]

Fig. 14. HSV Color Space Analysis

The input image has low brightness and saturation, as it appears dark and washed out. In contrast, the ground truth has high brightness, vivid hue, and strong saturation. The SFTW-restored image shows a close restoration of all 3 brightness, hue, and saturation to the ground truth. SFTW effectively removes light scattering due to weather and saturation loss, generating a visually appealing and realistic output.

LBP Texture

LBP(Local Binary Pattern) is a texture visualization index that

visualizes the local textures of the image by comparing each pixel with each neighboring pixel. [36] LBP converts the local texture into a binary number by thresholding all the adjacent pixels against the center pixel's value.

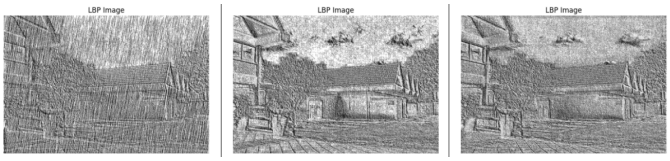

Fig. 15. LBP Texture Analysis

The input image shows blurred edge textures due to weather degradation, particularly affecting intricate and high detail-density regions like human faces or trees. The ground truth has sharp and clear textures. The SFTW restored texture map reflects accurate restoration compared to the ground truth, where critical structural details are recovered while reducing noise simultaneously, which is critical for applications like object recognition and scene analysis.

Roberts' Edge and Susan's Corner Detection

Roberts Edge Detection identifies object boundaries by calculating pixel gradients. It calculates and identifies transitions in intensity to detect sharp edges. Susan Corner Detection emphasizes delicate details and corner structures in the image.

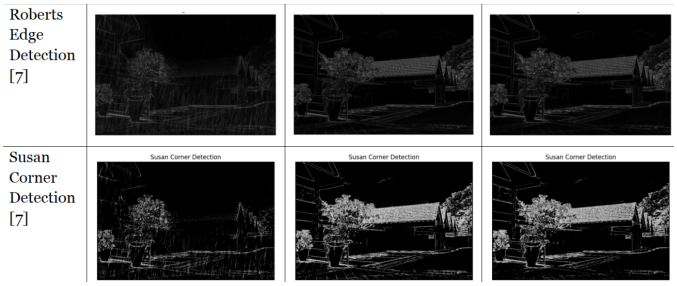

Fig. 16. Roberts and Susan's Edge and Corner Detection

The degraded input image through the detection algorithms shows blurred edges and missing details, while the ground truth image still reflects structural clarity by retaining the sharp edges and corners. The SFTW-restored image achieves significant improvements: Roberts edge detection shows recovered edge boundaries, especially in object edges and buildings. Susan corner detection reflects the restoration of finer details in complex regions, such as leaves and rooftops.

Summary of EVA

Combining the prior analytical results, it is concluded that SFTW:

1. Effectively **denoises** while **preserving structural details**.
2. Accurately **recovers saturation, hues, and contrast**.
3. Significantly improves **boundary clarity** and **fine details**.
4. Precisely restores **critical textures** in **complex** regions.

VI. Discussion

A. *Overview of SFTW Comparison with Existing Models*

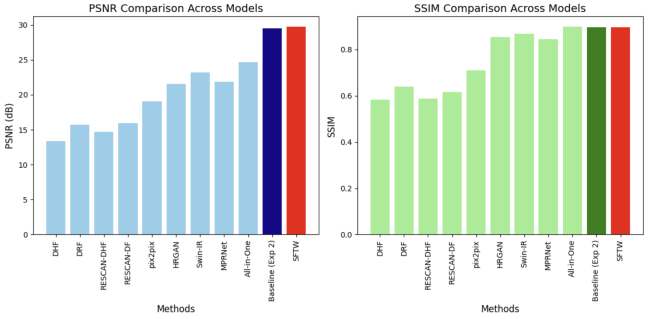

Fig. 17. SFTW Results Compared with Existing Methods

The results from both phases of this research were compared to some of the existing weather restoration models mentioned in the TransWeather research [1]: DetailsNet+Dehaze (DHF/DRF) [46], RESCAN + Dehaze (DHF/DF) [47], pix2pix [48], HRGAN [49], Swin-IR [50] and MPRNet [51]. The performance comparison between mentioned models regarding PSNR and SSIM evaluation metrics is presented in the bar chart (Figure 17) and depicted in the comparison table after the Phase Two results in the 'Results' section.

In figure 17, the SFTW model is highlighted in red to emphasize its contrasting performance with the other existing models and the Phase One baseline TransWeather model. The replicated TransWeather model tested in this research significantly outperformed the existing methods, especially in PSNR, achieving 29.52 dB, which is notably higher than HRGAN (21.56 dB) and MPRNet (21.90 dB). Swin-IR and HRGAN models, achieving PSNR values of 23.23 dB and 21.56 dB, respectively, are significantly lower than the SFTW model as well.

Apart from the original TransWeather model [1], the All-in-One model [3] also achieved relatively decent results, with a PSNR of 24.71 and an SSIM of 0.8980, slightly higher than the SSIM of the baseline TransWeather. Though the Phase One experiment achieved higher PSNR scores, the lower SSIM score indicates the room for improvement in terms of maintaining and restoring structural similarity, which could be future experiments to address these limitations in the SFTW model. Although the All-in-One model achieved a slightly higher SSIM value of 0.8980, the SFTW model also achieved competitive results and further focuses on enhancing stability and optimizing deeper feature extraction capabilities.

The following subsections provide detailed horizontal (with other existing methods) and vertical (with own research) comparisons along with a discussion of the effectiveness and results.

B. Detailed Horizontal Comparison

1. Comparison: Phase One Replicated TransWeather Model vs. SFTW
PSNR: 29.52 dB, SSIM: 0.8965 (TransWeather); PSNR: 29.71 dB, SSIM: 0.8962 (SFTW). Compared to the replicated TransWeather model in phase 1, the SFTW model achieved a slight improvement of PSNR, reaching 29.71 dB, and the SSIM value remained nearly the same at 0.8962. Although there was a minor decline in SSIM, the improvement in PSNR indicates that SFTW demonstrates an overall better capability to reduce noise; a decline of 0.0003 is also minor enough to neglect based on naked eye experience.

2. Comparison: All-in-One Model vs. SFTW
Summary of Results: PSNR: 24.71 dB, SSIM: 0.8980 (All-in-One).
the All-in-One model uses multiple encoders each specialized for processing a specific type of extreme weather and uses a single decoder for all weather-degraded images. This design gives the All-in-One model strong adaptability. Although it achieved a higher SSIM value than that of SFTW - 0.8980 against 0.8962, its PSNR was far lower than that achieved by SFTW at 24.71 dB. This indicates the limitations of the All-in-One model in terms of reducing noise, reflecting the critical advantages of the self guided filtering mechanism introduced in SFTW. The All-in-One model's architecture is based on multiple encoders, which increases its architectural complexity. Though performing well for specific weather types, it has significantly weaker performance under mixed or dynamic weather conditions due to the increased architectural complexity and limited feature extraction depth. In contrast, SFTW maintains superior overall performance under such scenarios.

Edge and Texture Preservation Comparison
In terms of edge and texture preservation, SFTW significantly outperformed other models. The LBP texture analysis and edge detection results reflect SFTW's advantage at preserving image details and edges. Other models, such as HRGAN and MPRNet, had a poor performance in the recovery of edges and textures, as reflected through their low SSIM scores with HRGAN often resulting in over-smoothed images that led to the loss of important edge and detail features. The self-guided filter layer in the SFTW model is superior not only in denoising but also in preserving key structural information within the image, especially for edge and corner detection, where the SFTW model shows an excellent recovery of original details. In contrast, more complex architectures, such as All-in-One and Swin-IR lose edge details during feature extraction.

Practical Applicability Comparison
The enhanced LV+SFTW model's ability to process real-time video inputs provides the model greater flexibility and adaptability for practical applications. This makes it adapted for practical scenarios, where fast and stable output is critical. Models like HRGAN and MPRNet are better for static image restoration, as they lack the capability to handle video or more dynamic weather conditions effectively and efficiently. For basic applications, where the static output images are dedicated for computer algorithms only, such models are sufficient enough, as quality is not important.

Better performing models such as Swin-IR and All-in-One, the models perform comparably well in some scenarios, but their complex architectures will require significantly more computational power, which further limits their practicality in real-time applications. In contrast, SFTW's utilizes TransWeather's optimized architecture which leads to faster computation, enabling faster inference speeds on the same hardware, further enhancing its practicality beyond the quantitative metrics.

Versatility and Portability Comparison
As mentioned previously, SFTW is able to handle complex or mixed weather conditions compared to existing models. SFTW uses a single encoder-decoder structure combined with self-guided filtering and ResNet50-based deep residual feature learning compared to All-in-One's multiple encoders. This enables the model to adapt and process a variety of mixed and dynamic extreme weather scenarios.

For real world utilization, SFTW is able to process diverse weather conditions without requiring separate training or unique architectures/methods for each weather type, which gives SFTW an advantage in portability. This portability feature makes SFTW a highly practical method for adverse weather restoration. Its stable performance and efficiency across various conditions also establishes SFTW as the most versatile and effective model for extreme weather restoration currently available.

C. Detailed Vertical Comparison

Phase	Method	PSNR	SSIM
Phase 1	Replicated TransWeather	29.52	0.8965
Phase 2	Innovated SFTW	29.71	0.8962

To remind, in the SFTW model, the PSNR value improved to 29.71 dB, which reflects that the introduction of innovative components enhanced both the results and model performance. The decrease of SSIM can be attributed to the upgraded model's utilization of a deeper feature extraction network (ResNet50) instead of a broader VGG16. Although this deeper architecture allows the extraction of more complex features, it also introduces structural incompatibilities in certain areas, which may lead to minor structural degradation.

Comparison of Feature Extraction and Deep Learning Capabilities

In contrast, the enhanced Phase Two SFTW model replaced VGG16 with the deeper ResNet50. The improved PSNR performance clearly shows the advantage of this change. The residual connections in ResNet50 address the vanishing gradient problem often encountered in deep networks, which enables the model to maintain stability and extract deeper features.

The enhanced phase 2 SFTW model replaced VGG16 with the deeper ResNet50. The advantage brought from this substitution is reflected in the improved PSNR performance. The residual connections in ResNet50 address the vanishing gradient problem often encountered in deep networks.

The replicated TransWeather model in phase 1 also performed well enough in terms of preserving details and edges as indicated through SSIM. However, SFTW introduced a trainable self-guided filter layer, which not only further improved noise removal but also restored edges and detailed features within the image.

Comparison of Computational Efficiency

Throughout both phases, the base model was kept the same with the original single encoder-decoder architecture, which ensures the basic efficiency. Compared to other models, this architecture from TransWeather will provide a faster inference process. However, the enhanced SFTW model in Phase 2 introduced additional components and adopted a deeper feature extraction pre-trained model, which increases computational power in both training and inference processes. This is the only aspect where Phase Two had a downgrade compared to Phase One.

For real-time video processing, the phase one replicated-TransWeather model was only designed for processing single images and lacks the functionality to process real-time videos. However, the inference capabilities of the phase 2 SFTW model were enhanced to include real-time video processing. The minor delays of roughly average 0.01 seconds make the computational efficiency of the SFTW model fast and capable enough to process frames of video feeds with no significant lag.

VII. Conclusion

This research successfully proposed the upgraded SFTW model by replicating and enhancing the chosen base model, TransWeather[1] model. The innovations includes the self guided filter layer, replacement of VGG16 to ResNet50, and introducing a real-time video processing inference to the model. Such innovations are all important experimental achievements in attempting to advance the field of weather degraded image restoration models to practicality. These have allowed SFTW to achieve an overall improvement in terms of denoising, detail restoration, generalization under complex weather scenarios, and practical applicability.

Experimental results also reflects the improvements in image quality through the improvements in PSNR and that of visual metrics, which indicates the enhanced accuracy and effectiveness, especially in noise removal and key features' restoration. Furthermore SFTW further exceeds existing models in restoring edge and details, which indicates its ability to recover structural features in degraded images. The structural evaluation index, SSIM has a minor decline, but the model still maintained a high standard to process images sufficient enough for both machines detection and perceptual visual quality, and also maintaining effectiveness in practice.

The research also validates the effectiveness of the introduced innovations for improvement by quantified metrics, which aims to contribute significantly in the field of computer vision, by developing a practical and stable side tool, that is able to push AI into people's daily safely. The versatility of SFTW in processing diverse weather scenarios, with its portability and efficiency, allows SFTW be positioned as a frontier of modern solutions for adverse weather restoration, ensuring both SFTW's and such related weather restoration algorithms' necessity in the evolving scope of technological advancement of practicality in real life.

VIII. References

[1] Valanarasu, Jeya Maria Jose, Rajeev Yasarla, and Vishal M. Patel. "Transweather: Transformer-based restoration of images degraded by adverse weather conditions." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[2] Zhu, Yurui, et al. "Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.

[3] Li, Ruoteng, Robby T Tan, and Loong-Fah Cheong. "All in one bad weather removal using architectural search." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[4] Liang, Jingyun, et al. "Swinir: Image restoration using swin transformer." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

[5] "Analysis of Image Quality Assessment Methods Based on FSIM, SSIM, MSE, and PSNR." Weixin Official Account. Accessed July 2023.

[6] "Restoration of Images Degraded by Adverse Weather Conditions." GitHub Repository. Accessed July 2023.

[7] "Image Restoration under Adverse Weather Conditions." CSDN Blog. Accessed July 2023.

[8] "Image Dehazing Methods: A Comprehensive Review." CSDN Blog. Accessed July 2023.

[9] Ajmera, Girish. "Feature Extraction of Images Using GLCM (Gray Level Co-occurrence Matrix)." Medium Blog. Accessed July 2023.

[10] "Create Local Binary Pattern of an Image Using OpenCV Python." GeeksforGeeks. Accessed July 2023.

[11] Sara, Umme, et al. "Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study." Journal of Computer and Communications, vol. 07, no. 03, Jan. 2019, pp. 8–18.

[12] , et al. "." , vol. 50, no. 7, 2024, pp. 1-12.

[13] Li, Boyi, et al. "Aod-net: All-in-one dehazing network." Proceedings of the IEEE International Conference on Computer Vision. 2017.

[14] Zhao, Xun, Feiyun Xu, and Zheng Liu. "TransDehaze: transformer-enhanced texture attention for end-to-end single image dehaze." The Visual Computer, 2024, pp. 1-15.

[15] Zhou, Yu, et al. "Eha-transformer: efficient and haze-adaptive transformer for single image dehazing." Proceedings of the 18th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry. 2022.

[16] Yasarla, Rajeev, Vishwanath A. Sindagi, and Vishal M. Patel. "Syn2real transfer learning for image deraining using gaussian processes." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[17] Yasarla, Rajeev, Carey E. Priebe, and Vishal M. Patel. "Art-ss: An adaptive rejection technique for semi-supervised restoration for adverse weather-affected images." European Conference on Computer Vision. Springer Nature Switzerland, 2022.

[18] Lehtinen, Jaakko, et al. "Noise2Noise: Learning image restoration without clean data." arXiv preprint arXiv:1803.04189. 2018.

[19] Zhang, He, Vishwanath Sindagi, and Vishal M. Patel. "Image de-raining using a conditional generative adversarial network." IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 11, 2019, pp. 3943-3956.

[20] Fu, Xueyang, et al. "Clearing the skies: A deep network architecture for single-image rain removal." IEEE Transactions on Image Processing, vol. 26, no. 6, 2017, pp. 2944-2956.

[21] Quan, Ruijie, et al. "Rethinking Image Restoration with Transformer-based Models." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[22] Quan, Yuhui, et al. "Deep learning for seeing through window with raindrops." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[23] Liu, Yun-Fu, et al. "Desnownet: Context-aware deep network for snow removal." IEEE Transactions on Image Processing, vol. 27, no. 6, 2018, pp. 3064-3073.

[24] Cheng, Bodong, et al. "Snow mask guided adaptive residual network for image snow removal." Computer Vision and Image Understanding, vol. 236, 2023, p. 103819.

[25] Chen, Wei-Ting, et al. "JSTASR: Joint size and transparency-aware snow removal algorithm based on modified partial convolution and veiling effect removal." Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 2020.

[26] Zhu, Shujin, and Zekuan Yu. "Self-guided filter for image denoising." IET Image Processing, vol. 14, no. 11, 2020, pp. 2561-2566.

[27] Karacan, Levent. "Trainable Self-Guided Filter for Multi-Focus Image Fusion." IEEE Access, 2023.

[28] Gu, Shuhang, et al. "Self-guided network for fast image denoising." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[29] Jiang, Bo, et al. "Single image fog and haze removal based on self-adaptive guided image filter and color channel information of sky region." Multimedia Tools and Applications, vol. 77, 2018, pp. 13513-13530.

[30] "RainDS CCNet A Dataset and Benchmark for Single Image Rain Removal." GitHub Repository. Accessed July 2023.

[31] "WGWS-Net: Weather-General and Weather-Specific Image Restoration Network." GitHub Repository. Accessed July 2023.

[32] "RESIDE Dehaze Datasets." Google Sites. Accessed July 2023.

[33] Gonzalez, R. C., Woods, R. E. Digital Image Processing. 2nd ed., Prentice Hall, 2002.

[34] Forsyth, D. A., Ponce, J. Computer Vision: A Modern Approach. Prentice Hall, 2002.

[35] Smith, A. R. "Color Gamut Transform Pairs." SIGGRAPH Comput. Graph., vol. 12, no. 3, 1978, pp. 12-19.

[36] Ojala, T., Pietikäinen, M., Mäenpää, T. "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, 2002, pp. 971-987.

[37] Xie, Enze, et al. "Segformer: Simple and efficient design for semantic segmentation with transformers." arXiv preprint arXiv:2105.15203. 2021.

[38] Vaswani, Ashish, et al. "Attention is all you need." Advances in Neural Information Processing Systems, 2017, pp. 5998-6008.

[39] Carion, Nicolas, et al. "End-to-end object detection with transformers." European Conference on Computer Vision, 2020, pp. 213-229. Springer.

[40] He, K., Zhang, X., Ren, S., Sun, J. "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778. 2016.

[41] Simonyan, K., Zisserman, A. "Very Deep Convolutional Networks for Large-Scale Image Recognition." Proceedings of the International Conference on Learning Representations (ICLR). 2015.

[42] "Object Detection Playground." Roboflow Universe. Accessed 1 Oct. 2024.

[43] He, Kaiming, Jian Sun, and Xiaoou Tang. "Guided Image Filtering." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 6, 2013, pp. 1397-1409.

[44] Zhang, Y., Liu, Z., and Wu, X. "Hybrid Guided Filter for Low-Light Image Enhancement." Journal of Visual Communication and Image Representation, vol. 83, 2022, pp. 103371.

[45] Kumar, P., Singh, A., and Saini, R. ”Self-Adaptive Guided Filter for Denoising Medical Images.” IEEE Access, vol. 9, 2021, pp. 83621-83633.

[46] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. ”DetailsNet + Dehaze (DHF) (DRF): Removing rain from single images via a deep detail network.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[47] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. ”RESCAN + Dehaze (DHF): Recurrent squeeze-and-excitation context aggregation net for single image deraining.” Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[48] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. ”pix2pix: Image-to-image translation with conditional adversarial networks.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[49] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. ”HRGAN: Heavy rain image restoration: Integrating physics model and conditional adversarial learning.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[50] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. ”Swinir: Image restoration using swin transformer.” Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.

[51] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. ”MPRNet: Multi-stage progressive image restoration.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.