Grounding Degradations in Natural Language for All-In-One Video Restoration

Supplementary Material

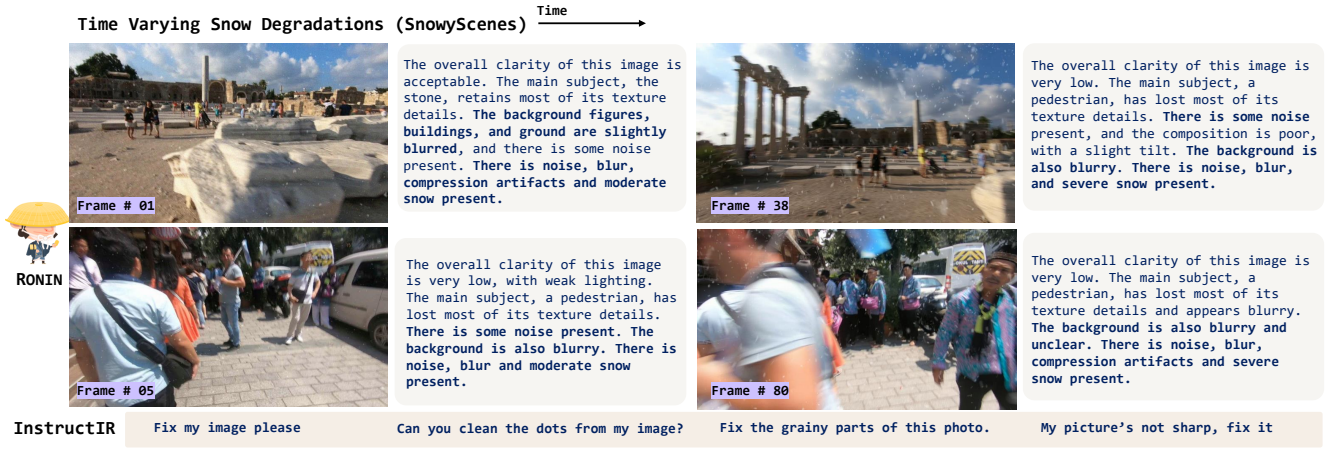

Figure 1. **Differences in Grounding Degradations and Instructions.** We sample two frames (at different timesteps) from two different videos of SnowyScenes benchmark and compare RONIN's language grounded descriptions with InstructIR [4]'s human-aligned instructions. Since InstructIR randomly samples instructions for each degradation, we show two samples (second and third) taken from noisy and blurry instructions, while first and third samples are taken from general instructions. It is evident that instructions are rigid and provide no meaningful clue without identifying the degradations. RONIN benefits from per-frame grounded degradations that also describe context.

Appendices

1. Additional Ablation Studies

We discuss the motivation behind grounding degradations, and present additional ablation studies to further understand different components of RONIN and the design choices made.

1.1. Motivation: Grounding Degradations

We posit that grounding the degradations in natural language to serve as a prior for the restoration algorithm offers flexible control along with interpretability. The instruction condition in methods such as InstructIR [4], although interpretable, requires that for each input, a random degradation-specific instruction is sampled and fed as input to the restoration method. While this is plausible in images, videos are much more challenging. Consider how restoring a 30fps 10 seconds video is dependent on 300 different calls to the text encoder in InstructIR [4], the VLM in [19] or the MLLM in [10]. We ablate this limitation in InstructIR [4] where we consider a single instruction variation i.e., we sample a degradation-dependent instruction once and reuse it for all the videos in the same degradation category and report results on the 3D benchmark in Tab. 1. Unsurprisingly, InstructIR [4] observes non-trivial performance drop.

In RONIN, however, no such limitation exists due to the proposed prompt approximation objective allowing the MLLM to be safely removed post-training. Further, grounded conditioning allows nuances in modulating channels since plain instructions can be rigid (e.g., *'clean up this image'*) and cannot handle composite degradations without complete knowledge of degradations at the inference time. Since the natural language grounding in RONIN also captures the context of the frame and offers fine-grained control, our proposed method is a positive step towards designing region-specific restoration methods (e.g., the sky has high noise due to flat texture, the building has ghosting artifacts due to repetitive patterns, etc.). We illustrate this further in Fig. 1 where we show that instructions that InstructIR [4] leverages are indeed rigid and fail to capture composite degradations meaningfully.

Method	Deblur (GoPro [20])		Denoise (DAVIS [22])		Derain (VRDS [30])	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
InstructIR [4]	30.93	0.94	31.25	0.92	31.10	0.95
Single Instruction	**30.90**	0.94	**31.17**	0.91	**31.03**	0.95
RONIN	**32.73**	**0.96**	**31.65**	**0.92**	**32.72**	**0.97**

Table 1. **Frequency of Instruction Sampling.** Results on 3D benchmark ablating the frequency of instruction sampling in InstructIR [4], and comparison with RONIN which does not need any instructions/text during inference.

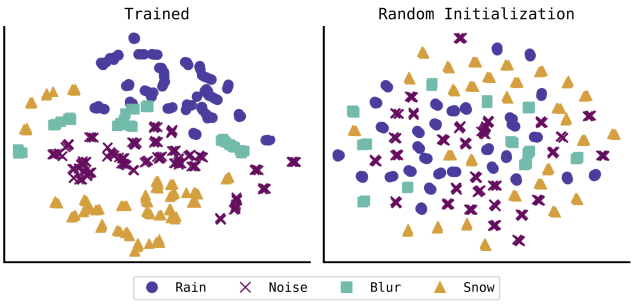

Figure 2. **tSNE Plot.** Visualization of learned and untrained prompts taken from the latent space of RONIN on 4D benchmark.

Methods	Denoise	Deblur	Derain	MACs (G)	Params
InstructIR [4]	0.1799	0.1444	0.0623	133.73*	73.9*M
PromptIR [23]	0.1793	0.1293	0.0578	158.49	35.6M
ViWSNet [34]	0.1734	0.1890	0.0902	88.93	57.7M
AverNet [38]	0.4277	0.1394	0.0640	127.72*	41.3*M
RONIN	**0.1713**	**0.1037**	**0.0463**	167.23	57M

Table 2. **Perceptual Results.** LPIPS scores on 3D benchmark (↓ is better), with MACs (G) and number of parameters Params (M). * indicates that optical flow network, while * indicates that the text encoder parameters were not included.

Are Learned Prompts Meaningful? To illustrate that the learned prompts are meaningful, we perturb the learned prompts with white Gaussian noise in inference and evaluate on 3D benchmark, see Tab. 3. We observe a significant drop in performance indicating that if wrong prompt information were propagated, RONIN would suffer. The drop in the performance illustrates that the learned prompts modulate the output and are necessary for the observed performance gains. We also visualize tSNE plots of learned and untrained prompts, showing that learned prompts effectively differentiate between degradations, see Fig. 2. Further, we also compute cosine similarity between the learned prompts and the raw text embedding taken from the text encoder and compare it with random prompts (untrained). We find that in the former case, trained prompts align closely with raw text embeddings (similarity scores in range of 0.9852–0.9914), while random prompts do not (similarity scores in range of -0.0393–0.0370).

Perceptual Results of RONIN On 3D benchmark, we present LPIPS [36] scores and compare it to prior methods. In line with the qualitative results, RONIN scores better on the metric (lower is better) indicating that the restored videos are pleasing to the human eye.

1.2. Additional Ablation Studies

Following ablation setup in **??**, we conduct a few more ablation studies with 3D as the benchmark.

Prompt Style	Deblur (GoPro [20])		Denoise (DAVIS [22])		Derain (VRDS [30])	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Perturbed Prompts	15.93	0.56	16.02	0.57	16.97	0.55
RONIN	**32.73**	**0.96**	**31.65**	**0.92**	**32.72**	**0.97**

Table 3. **Prompt Importance.** We perturb the prompts with white Gaussian noise and compute scores on the 3D benchmark dataset. The significant drop in performance illustrates that the learned prompts modulate the output and are necessary for the observed performance gains.

Prompt Location	Deblur (GoPro [20])		Denoise (DAVIS [22])		Derain (VRDS [30])	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Oracle	29.02	0.920	30.65	0.908	29.22	0.940
Only Degradation	28.88	0.915	30.60	0.906	29.05	0.934
RONIN	**28.99**	**0.919**	**30.65**	**0.908**	**29.18**	**0.937**

Table 4. **Utility of Learned Prompts.** We evaluate three different settings: an oracle setting where directly ground-truth descriptions are fed as prompts, only degradation and only scene prompts. RONIN's performance is similar to the oracle indicating the utility of learned prompts.

Prompt Location	Deblur (GoPro [20])		Denoise (DAVIS [22])		Derain (VRDS [30])	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Restormer [35]	28.65	0.906	30.32	0.900	28.70	0.920
Learned Prompt +Restormer [35]	**28.85**	**0.917**	**30.47**	**0.905**	**28.83**	**0.934**
Turtle [5]	28.68	0.914	30.59	0.906	29.01	0.934
Learned Prompt + Turtle [5]	**28.99**	**0.919**	**30.65**	**0.908**	**29.18**	**0.937**

Table 5. **Plug-and-Play RONIN.** We quantify the benefit of proposed learned prompt (RONIN) scheme in a fully Transformer based architecture: Restormer [35]. We see that in both Turtle [5] and Restormer [35], adding learned prompts boost the performance significantly.

Utility of Learned Prompts. We evaluate different prompt settings in Tab. 4 to establish the utility of learned prompts. The upper bound, i.e., the best performance possible, is directly feeding RONIN with the ground-truth embeddings, a setting termed as *oracle*. We see that the performance achieved by RONIN (using learned prompts) is quite similar to the oracle (row 1 and row 3), indicating that learned prompt indeed help disentangle the MLLM during inference. We also evaluate *Only Degradation* (row 2) setting where we remove any scene information from the prompt, leaving only degradation and severity informa-

tion. Notably, when scene information is removed, there little to distinguish between frames in a video since consecutive frames are usually quite similar. The results reflect that both distortion and scene information are important for fine-grained differentiation between degradations in video frames.

Plug-and-Play RONIN. In Tab. 5 we present results of the proposed prompt approximation setup with two different backbone architectures: Turtle [5], which is a U-Net style architecture with convolutional blocks in encoder, and Transformer blocks in decoder, and Restormer [35], which is a fully Transformer style architecture. In both cases, coupling the prompt approximation setting, proposed in this work, significantly boosts the performance of backbone architectures, indicating that with a single MLP block, the backbone can benefit from prior knowledge about degradations extracted from an MLLM.

2. Additional Related Work

Video restoration, in literature, is studied from several facets, mostly distributed in terms of how the motion is estimated and compensated for, and how the frames are processed in the learning procedure. Several methods employ optical flow to explicitly estimate motion, and devise a compensation strategy as part of the learning procedure, such as deformable convolutions [16, 17], or flow refinement [8]. On the other end, methods rely on the implicit learning of correspondences in the latent space across the temporal resolution of the video, a few strategies include temporal shift modules [12], or non-local search [13, 28, 41]. Further, similar differentiation exists in the manner a video is processed i.e., several methods opt for either recurrence in design [39, 40, 42] while others restore several frames at once [2, 7].

2.1. All-In-One Image Restoration

There have been several methods introduced in the literature for the purpose of all-in-one image restoration. All of these methods utilize backbone architectures which are constructed in either columnar [15] or UNet [25] fashion. Its extension to all-in-one tasks is aided by some conditioning on the restoration procedure, either only in the decoder (reconstruction), or conditioning at the latent stage. This condition is often realized in the form of some prior, either through degradation-aware feature injection, or through implicit (blackbox) or explicit (whitebox) prompts. However, all of the methods can be categorized into three different settings: contrastively learning the degradation information before restoring the input, implicitly injecting prompts to condition the restoration, or explicitly injecting prompts realized through degradation or textual features.

To the best of our knowledge, AirNet [11] proposed the first standardized baseline all-in-one method to recover images from a variety of degradation levels and corruptions. The authors proposed a contrastive learning based degradation encoder that learned to differentiate between the degradations in its latent space. The following architecture then learned to restore the frames conditioned on the contrastively learned representations of the degraded input. Another all-in-one restoration method for weather-specific degradations is TransWeather [29]. TransWeather proposed a Transformer-In-Transformer [6] style encoder to learn hierarchical features, followed by a weather degradation queries conditioned decoder to recover the clean image. In both of these methods, some degradation-specific guidance is provided–class labels for positive and negative sample mining in the case of AirNet, or weather-specific queries in TransWeather.

However, different from these, PromptIR [23] proposed to inject prompts in the decoder of the encoder-decoder style restoration architecture. The prompts were implicitly learned since they were input-conditioned, and the method required no supervision on the degradation. Henceforth, a series of all-in-one image restoration methods followed the baselines set by AirNet [11], and proposed different architectures for the task. However, most of these works differ in how the degradation information is injected in the learning procedure, either implicitly or otherwise. Prompt-In-Prompt (PIP) [14] proposed to fuse two prompts, i.e., degradation-aware prompt, and base restoration prompt, into a universal prompt. The resultant universal prompt is then fused with the input features through a feature-prompt mixing module for the restoration tasks.

Contemporary works such as InstructIR [4] proposed to inject human-aligned instructions into the restoration architecture's decoders through a prompt-feature mixing module. In practice, the instructions, generated through a multi-modal large language model, were first fed into a sentence transformer (pretrained on large textual data) to compute the instruction embeddings for the restoration procedure. One downside of such an approach is that on deployment, the sentence transformer can not be decoupled from the restoration architecture since the decoder is conditioned on the instruction embeddings obtained from the sentence transformer. Similarly, LLMRA [10] leveraged a multi-modal large language model (MLLM) to generate context descriptions, and a CLIP text encoder [24] to obtain embeddings of the context. These embeddings were then injected into the restoration procedure. LLMRA suffers from similar limitations as InstructIR i.e., both of these methods have to deploy the underlying procedure used to generate embeddings along with the restoration architecture. In line with language-guided restoration, several methods such as LanguageWeather [33], and TextIR [32] also leverage language

models (or vision-language models) to introduce degradation prior in the restoration procedure. In Perceiver-IR [37], a two-stage training is utilized wherein a prompt is learned in first stage, and then it guides the restoration process. Further, DINO-v2 is used to combine semantic information with the degradation representations. Similar to prior works, Perceiver-IR also is tightly coupled with pretrained modules such as prompts, and DINO-v2 encoder and cannot function standalone in inference unlike RONIN.

2.2. All-In-One Video Restoration

All of the image methods discussed above are comparable to each other given consistent evaluation on similar all-in-one restoration datasets and tasks. However, the all-in-one video restoration progress is siloed, and the attempts made in literature are disparate in nature. VJT [9] proposed a multi-degradation restoration architecture for low-light enhancement, deblurring and denoising tasks. The proposed Transformer-based architecture employed a multi-tier setup wherein each tier utilized a different level of degraded video as a target for feature learning process. Further, they also introduced a new Multi-scenes Lowlight-Blur-Noise (MLBN) dataset for the restoration task. However, the dataset was not publicly released for any follow-up methods to train and evaluate their methods on. Similarly, another work [26] introduced joined deblurring and denoising method, and proposed a new dataset for the task. The proposed method departed from conventional architecture design in all-in-one restoration literature by introducing separate encoders for each task. However, similar to VJT, the dataset was not publicly released. Before VJT, another method CDUN [3] proposed an all-in-one video restoration architecture targeting deraining, dehazing, desnowing and low-light enhancement tasks. Although similar in a few tasks to VJT [9], CDUN utilized different datasets, while synthesizing own video desnowing dataset due to, then, a lack of any video desnowing dataset. More recently, ViWS-Net [34] proposed all-in-one video restoration architecture for weather degradation removal, namely for desnowing, dehazing and deraining tasks. However, since CDUN [3] did not publicly release the desnowing dataset that they reported scores on, ViWS-Net synthesized another desnowing dataset, referred to as KITTI-Snow based on the KITTI dataset [18][1]. More recently, AverNet [38] proposed time-varying degradation dataset where every fixed interval (a predefined frame, e.g., every sixth frame), the degradation changed simulating varying corruption in a video. The authors argue that this setting is more natural to videos. However, the degradations considered are limited to variations in noise, Gaussian blur and compression.

[1] https://github.com/scott-yjyang/ViWS-Net KITTI-Snow was publicly released.

3. Dataset Details

All of the benchmarks considered in this work are created through standard datasets in video restoration literature and are available open-source for academic research purposes, except our proposed `SnowyScenes` benchmark, which will be open-sourced and released publicly for future research work.

3.1. 3D Benchmark

As discussed earlier, we consider three different video restoration tasks to form the 3D benchmark, namely video denoising, video deraining, and video deblurring. In video denoising, following [27], we employ the DAVIS [22] dataset which consists of 60 videos in the training set and 30 videos in the held-out test set. We add white Gaussian noise with $\sigma \in \mathcal{U}[20, 50]$, and test with $\sigma = 50$ Gaussian noise. In video deraining, we use the video raindrop and rain streak removal (VRDS) dataset introduced in [30]. The dataset comprises videos captured in diverse scenarios in both daytime and nighttime settings corrupted by both rain streaks and raindrops. There are a total of 102 videos at a resolution of 1280×720 with 100 frames per video in the dataset, and 72 are in training set while 30 are in the held-out test set. In video deblurring, we employ the GoPro dataset introduced in [20] which contains videos captured from the GOPRO4 Hero consumer camera at a resolution of 1280×720. The dataset contains 3214 pairs of blurry and sharp images, with 2103 pairs in the training set and 1111 pairs in the test set. GoPro dataset is formed by integrating sharp information over time for blur image generation, instead of modeling a kernel to convolve on the sharp image [20].

3.2. 4D Benchmark

The 4D benchmark considers four different video restoration tasks, with three being similar to the ones in 3D benchmark. The additional restoration task is video desnowing and dehazing. In [1], the authors introduced a video desnowing and dehazing dataset, RVSD. The dataset consists of 110 videos at varying resolutions from 480p to 4k, with 80 videos in the training set and 30 videos in the held-out test set. RVSD contains dynamic scenes in varied lighting conditions, both in night and daytime, and has realistic and dynamic snow and haze rendered in Unreal Engine.

3.3. SnowyScenes Benchmark

In both 3D and 4D benchmarks, a single degradation affects a video, i.e., there are no videos with composite degradations. However, in many cases, degradations affect videos in a time-varying fashion. In other words, degradations change in intensity or even type as more frames are sampled/observed. To simulate such a setting, a new dataset called time-varying degradations, TUD, was introduced in a recent work [38]. In TUD, the authors con-

Figure 3. **Samples of Degradations Descriptions.** A few samples of frames and their respective grounded degradation prompts taken from different benchmarks. In the first column, from top to bottom, the frames are taken from SnowyScenes (moderate snow), SnowyScenes (severe snow), 3D (denoise). In the second column, from top to bottom, the frames are taken from 3D (derain), 4D (desnow), and 3D (deblur) benchmarks, respectively.

Figure 4. **Illustration of Limitation in Grounded Degradations.** Two samples of language descriptions where extraneous degradations are present. The first frame is taken from a desnowing task video, but the prompt describes *noise and blur*. Although the frame has slight blur and arguably even noise, the ground truth is only free of snow degradation. The second frame is taken from a deblurring video, but there is mention of *some noise* in the description.

sidered degradations introduced by Gaussian, Poisson and Speckle noise, kernel-based blur, and video/JPEG compression. In this work, we propose a harder time-varying setting, SnowyScenes, with realistic blur and varying snow intensity. We pick 56 blurry videos from widely used GoPro [20] and REDS [21] datasets, with 42 videos in the training set and 14 in the held-out test set. We borrow Gaussian, Poisson and Speckle noise and compression degradations, but synthesize snow with two intensity levels moderate and severe. For Gaussian and Speckle noise, the noise levels are sampled uniformly from $[10, 15]$, while the Poisson noise α is sampled from $[2, 4]$ following the Poisson noise mathematical model $\mathcal{P}(10^{\alpha} \times x)/10^{\alpha} - x$. Further, in the case of compression, the quality factor in JPEG compression is randomly chosen from $\{20, 30, 40\}$, while in video compression the codecs are randomly chosen from $\{\text{libx264}, \text{h264}, \text{mpeg4}\}$, following [38]. Since the videos already have dynamic blur which is kernel-free, we do not further add Gaussian or resize blur. To generate a corrupted video, degradations are sampled with a probability of 0.55. We summarize the statistics of our proposed benchmark in Tab. 6. The benchmark will be released along with the necessary codebase for reproducibility and future research.

SnowyScenes Statistics	**GoPro** [20]		**REDS** [21]	
	Train	**Test**	**Train**	**Test**
Total Videos	22	11	20	3
Total Frames	2103	1111	2000	300
Resolution	1280×720			

Table 6. **Statistics of SnowyScenes Benchmark.** We present a summary of total videos, frames and resolution in the proposed SnowyScenes benchmark.

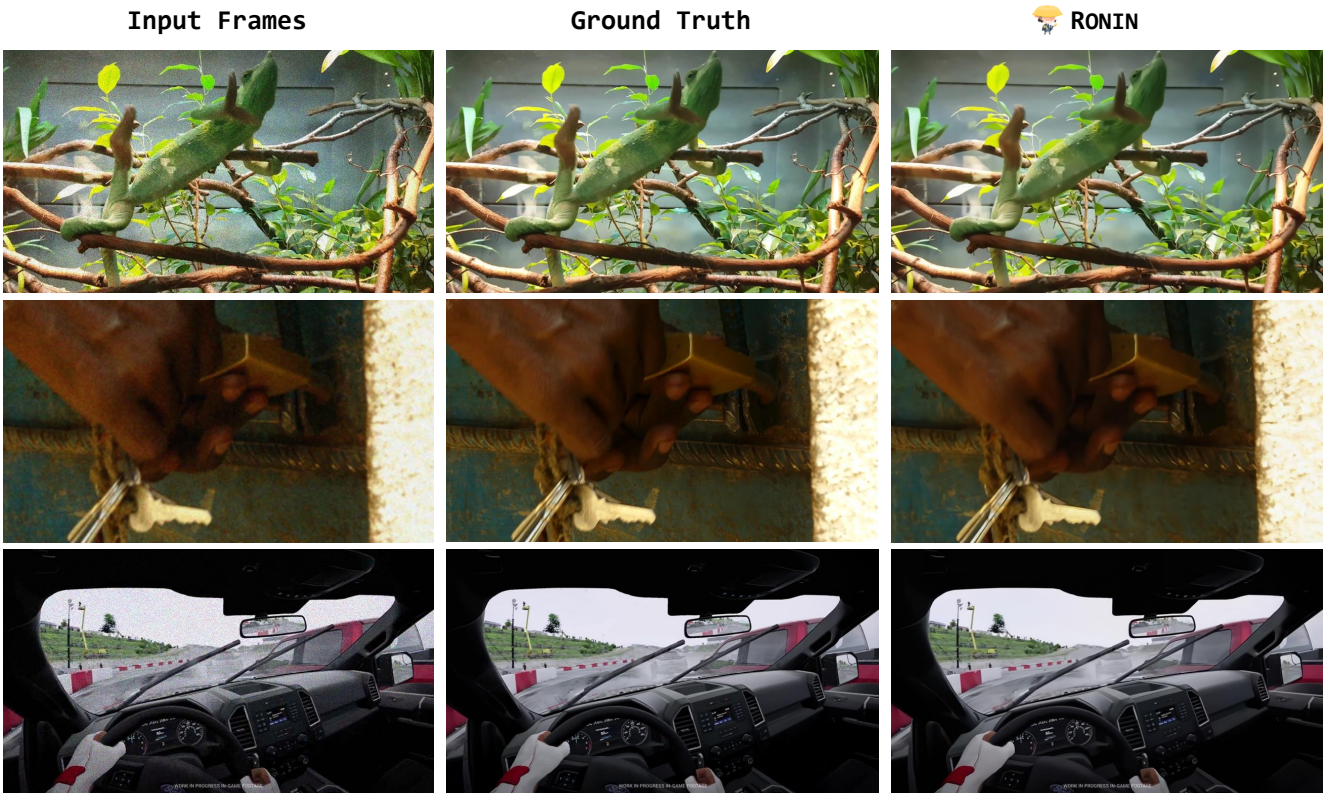

Figure 5. **TUD Benchmark Visual Results.** Qualitative results of RONIN on the TUD benchmark on three different settings. The first row contains frames from $t = 6$ test set, while second and third row contains frames from $t = 12$ and $t = 24$ test sets, respectively. RONIN's outputs are natural and faithful to the ground truth.

Deg.	'Snow'	'Noise'	'Rain'	'Haze'	'Blur'
Deblur	0	1328	0	0	2103
Derain	0	5518	7200	0	1669
Denoise	6	6208	80	0	6117
Desnow	26516	13471	2	2163	10549

Table 7. **Robustness Analysis.** Count of degradations in the grounded degradation text from Q-Instruct [31] for the 4D benchmark. The numbers represent correctly classified degradations, while others are misclassifications.

4. Prompt Details

Recall that the basic prompt to query Q-Instruct [31] to assess the degradation in the image is *'Rate the quality of the image. Think step by step.'*. While this works in most cases where the degradation matches the synthetic degradations Q-Instruct has been fine-tuned on e.g., noise, blur, brightness, clarity, it struggles to understand degradations like snow, rain, compression, and the intensity of these degradations. Therefore, we explicitly query the VLM and inquire regarding each of the candidate degradations, i.e., noise, blur, rain, compression, snow, and their appropriate combinations in the case of TUD and `SnowyScenes` benchmarks, with the answer being in a Yes/No format, while it is a multiple choice answer in the case of intensity of degradations question. A bare-bones sketch of the prompt algorithm is presented in Algorithm 1. In Algorithm 1, we loop through different degradations for each image $\mathbf{I}$ and concatenate the degradations Q-Instruct [31] flags as being present in the image. Each degradation is added as: *there is d in the image, and the intensity of d is $s_{\mathbf{I}}$* in the description. Consider a few prompt samples in Fig. 3, where the first two images in the first column have moderate and severe snow, respectively, while the third image has severe noise. Also, the first image in second column has severe rain.

4.1. Robustness of RONIN

We evaluate the robustness of our proposed method, RONIN, to misclassifications of Q-Instruct [31]. In Tab. 7, we show the count of degradations accurately identified by the MLLM and misclassifications. Since the dataset is video-based, naturally blur and noise (e.g., motion blur) occur, and as we lack appropriate ground truth (e.g., no blur but only snow in desnow data), we do not clean the prompts. We find that RONIN is robust and handles these

Algorithm 1 Prompt Algorithm

Require: Image $\mathbf{I}$
Require: Vision-Language Model $\mathbf{Q}_\theta$ ▷ e.g., Q-Instruct
  $b_p \leftarrow$ Rate the quality of the image. Think step by step.
  $d_1 \leftarrow \mathbf{Q}_\theta(\mathbf{I}, b_p)$ ▷ Initial Description
  desc $\leftarrow \emptyset$
  for $d \in \{\text{noise}, \text{rain}, ...\}$ **do** ▷ Candidate Degradations
    $f_\mathbf{I} \leftarrow$ Is there d degradation present in the image?
      Answer Yes or No. ▷ Fine-grained Query
    if $f_\mathbf{I}$ is Yes **then**
      $t_s \leftarrow$ Rate the intensity of degradation d?
        Choose either severe or moderate.
      $s_\mathbf{I} \leftarrow \mathbf{Q}_\theta(\mathbf{I}, t_s)$ ▷ Evaluate
      $d_2 \leftarrow$ There is d in the image,
        and the intensity of d is $s_\mathbf{I}$
      desc $\leftarrow \mathsf{concat}(d_1, d_2)$ ▷ Grounded Degradation
    end if
  end for

cases well due to degradation information from the first encoder (see **??**), and learnable prompts initialized from latent features. Notably, degradations like snow, rain, and haze, which are not caused by camera equipment, have minimal misclassifications. For example, only 80 out of 6208 frames in the noise dataset were misidentified as rain. In the desnow data, haze was occasionally flagged, but the authors of desnow dataset [1] consider snow+haze as one degradation, so we do not consider haze separately.

4.2. Cost of Querying MLLM

We report and discuss the cost of querying Q-Instruct [31] for grounding the degradations in natural language. These costs are independent of number of degradations since we check for each degradation asking for a Yes/No answer from Q-Instruct. So, even if 3D dataset is used, we cycle through a list of candidate degradations which is a union of all degradations present across all benchmarks discussed in this manuscript. Q-Instruct takes about 24GB of memory to run for the entire dataset, with a frame of size $1280\times720\times3$. It takes around 13 seconds per frame to get the description, which includes querying Q-Instruct twice once for general description and then looping through degradation types (see Algorithm 1). So, for the entire dataset of GoPro, the cost to generate descriptions is 3214×13 seconds or 11.6 hours. This is on a consumer grade GPU, but with better GPUs, distributed inference, and flash attention, the time would reduce significantly. It is important to emphasize that all of this cost incurs only once prior to training, since we store the prompts offline, and only do $\mathcal{O}(1)$ look ups during training.

5. Limitations, Future Work, and Impact

The descriptions may occasionally include more degradations than are present in the video, such as the mention of noise in a frame which is a part of a video in the deblurring task. Although this rarely happens, as Q-Instruct [31] when prompted appropriately is adept at grounding degradations, we hypothesize that as such models improve, RONIN will directly benefit from their advancements. We do not correct such descriptions due to the assumption of no access to individual degradations, but improving the prompt template should also benefit RONIN which we leave for future work, see Fig. 4 for few examples of such cases.

5.1. Ethics and Societal Impact

This work introduces a method, RONIN, and a benchmark dataset, `SnowyScenes`, to help advance the study of machine learning, particularly for video restoration. While the proposed method effectively restores the degraded videos, we recommend expert supervision in sensitive applications. Further, our proposed benchmark is constructed from two publicly available datasets, namely GoPro [20] and REDS [21]. The snow is synthesized using assets of two different types of snows (for moderate and severe snow). All of the assets and both the datasets are distributed under the Creative Commons Attribution 4.0 International (CC BY 4.0) license[2]. Therefore, `SnowyScenes` will also be distributed under the same CC BY 4.0 license.

References

[1] Haoyu Chen, Jingjing Ren, Jinjin Gu, Hongtao Wu, Xuequan Lu, Haoming Cai, and Lei Zhu. Snow removal in video: A new dataset and a novel method. in 2023 ieee. In *CVF International Conference on Computer Vision (ICCV)*, pages 13165–13176. 4, 7

[2] Huaian Chen, Yi Jin, Kai Xu, Yuxuan Chen, and Changan Zhu. Multiframe-to-multiframe network for video denoising. *IEEE Transactions on Multimedia*, 24:2164–2178, 2021. 3

[3] Yuanshuo Cheng, Mingwen Shao, Yecong Wan, Lixu Zhang, Wangmeng Zuo, and Deyu Meng. Cross-consistent deep unfolding network for adaptive all-in-one video restoration. *arXiv preprint arXiv:2309.01627*, 2023. 4

[4] Marcos V Conde, Gregor Geigle, and Radu Timofte. High-quality image restoration following human instructions. *arXiv preprint arXiv:2401.16468*, 2024. 1, 2, 3

[5] Amirhosein Ghasemabadi, Muhammad Kamran Janjua, Mohammad Salameh, and Di Niu. Learning truncated causal history model for video restoration. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2, 3

[6] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in neural information processing systems*, 34:15908–15919, 2021. 3

[2] https://creativecommons.org/licenses/by/4.0/

[7] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3897–3906, 2019. 3

[8] Cong Huang, Jiahao Li, Bin Li, Dong Liu, and Yan Lu. Neural compression-based feature learning for video restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5872–5881, 2022. 3

[9] Yuxiang Hui, Yang Liu, Yaofang Liu, Fan Jia, Jinshan Pan, Raymond Chan, and Tieyong Zeng. Vjt: A video transformer on joint tasks of deblurring, low-light enhancement and denoising. *arXiv preprint arXiv:2401.14754*, 2024. 4

[10] Xiaoyu Jin, Yuan Shi, Bin Xia, and Wenming Yang. Llmra: Multi-modal large language model based restoration assistant. *arXiv preprint arXiv:2401.11401*, 2024. 1, 3

[11] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17452–17462, 2022. 3

[12] Dasong Li, Xiaoyu Shi, Yi Zhang, Ka Chun Cheung, Simon See, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. A simple baseline for video restoration with grouped spatial-temporal shift. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9822–9832, 2023. 3

[13] Wenbo Li, Xin Tao, Taian Guo, Lu Qi, Jiangbo Lu, and Jiaya Jia. Mucan: Multi-correspondence aggregation network for video super-resolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 335–351. Springer, 2020. 3

[14] Zilong Li, Yiming Lei, Chenglong Ma, Junping Zhang, and Hongming Shan. Prompt-in-prompt learning for universal image restoration. *arXiv preprint arXiv:2312.05038*, 2023. 3

[15] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021. 3

[16] Jingyun Liang, Yuchen Fan, Xiaoyu Xiang, Rakesh Ranjan, Eddy Ilg, Simon Green, Jiezhang Cao, Kai Zhang, Radu Timofte, and Luc V Gool. Recurrent video restoration transformer with guided deformable attention. *Advances in Neural Information Processing Systems*, 35:378–393, 2022. 3

[17] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. *IEEE Transactions on Image Processing*, 2024. 3

[18] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2022. 4

[19] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Controlling vision-language models for universal image restoration. *arXiv preprint arXiv:2310.01018*, 2023. 1

[20] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 1, 2, 4, 5, 7

[21] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *CVPR Workshops*, 2019. 5, 7

[22] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. 1, 2, 4

[23] Vaishnav Potlapalli, Syed Waqas Zamir, Salman Khan, and Fahad Khan. Promptir: Prompting for all-in-one image restoration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 2, 3

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3

[25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 3

[26] Shayan Shekarforoush, Amanpreet Walia, Marcus A Brubaker, Konstantinos G Derpanis, and Alex Levinshtein. Dual-camera joint deblurring-denoising. *arXiv preprint arXiv:2309.08826*, 2023. 4

[27] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1354–1363, 2020. 4

[28] Gregory Vaksman, Michael Elad, and Peyman Milanfar. Patch craft: Video denoising by deep modeling and patch matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2157–2166, 2021. 3

[29] Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M Patel. Transweather: Transformer-based restoration of images degraded by adverse weather conditions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2353–2363, 2022. 3

[30] Hongtao Wu, Yijun Yang, Haoyu Chen, Jingjing Ren, and Lei Zhu. Mask-guided progressive network for joint raindrop and rain streak removal in videos. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7216–7225, 2023. 1, 2, 4

[31] Haoning Wu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Annan Wang, Kaixin Xu, Chunyi Li, Jingwen Hou, Guangtao Zhai, et al. Q-instruct: Improving low-level visual abilities for multi-modality foundation models. *arXiv preprint arXiv:2311.06783*, 2023. 6, 7

[32] Qiuhai Yan, Aiwen Jiang, Kang Chen, Long Peng, Qiaosi Yi, and Chunjie Zhang. Textual prompt guided image restoration. *arXiv preprint arXiv:2312.06162*, 2023. 3

[33] Hao Yang, Liyuan Pan, Yan Yang, and Wei Liang. Language-driven all-in-one adverse weather removal. *arXiv preprint arXiv:2312.01381*, 2023. 3

[34] Yijun Yang, Angelica I Aviles-Rivero, Huazhu Fu, Ye Liu, Weiming Wang, and Lei Zhu. Video adverse-weather-component suppression network via weather messenger and adversarial backpropagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13200–13210, 2023. 2, 4

[35] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 2, 3

[36] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 2

[37] Xu Zhang, Jiaqi Ma, Guoli Wang, Qian Zhang, Huan Zhang, and Lefei Zhang. Perceive-ir: Learning to perceive degradation better for all-in-one image restoration. *IEEE Transactions on Image Processing*, 2025. 4

[38] Haiyu Zhao, Lei Tian, Xinyan Xiao, Peng Hu, Yuanbiao Gou, and Xi Peng. Avernet: All-in-one video restoration for time-varying unknown degradations. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2, 4, 5

[39] Minyi Zhao, Yi Xu, and Shuigeng Zhou. Recursive fusion and deformable spatiotemporal attention for video compression artifact reduction. In *Proceedings of the 29th ACM international conference on multimedia*, pages 5646–5654, 2021. 3

[40] Zhihang Zhong, Ye Gao, Yinqiang Zheng, and Bo Zheng. Efficient spatio-temporal recurrent neural network for video deblurring. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 191–207. Springer, 2020. 3

[41] Kun Zhou, Wenbo Li, Xiaoguang Han, and Jiangbo Lu. Exploring motion ambiguity and alignment for high-quality video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22169–22179, 2023. 3

[42] Chao Zhu, Hang Dong, Jinshan Pan, Boyang Liang, Yuhao Huang, Lean Fu, and Fei Wang. Deep recurrent neural network with multi-scale bi-directional propagation for video deblurring. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3598–3607, 2022. 3