

RUSID: ROBUST UNCERTAINTY-AWARE SINGLE IMAGE DERAINING BEYOND CERTAINTY

Anonymous authors

Paper under double-blind review

ABSTRACT

Rainy weather induces rain streaks, blurs details, and reduces contrast, impairing image quality, making single image deraining a classic research topic. However, existing learning-based image restoration methods fail to account for uncertainties in both data and model dimensions, thus being unable to produce satisfactory results. To address this challenge, we introduce a novel framework called the Uncertainty-aware Visual-priors Prompt-interaction Network (UVPNet). UVPNet comprises three key modules: the Distribution-aware Visual Priors Learning (DVPL) module, which aims at data-wise aleatoric uncertainties, the Certainty-Uncertainty Prompt Fusion (CUPF) module, which tackles model-wise epistemic uncertainties, and the Channel Spatial Uncertainty Weighting Block (CSUWB). UVPNet leverages uncertainty modeling through visual semantic and depth priors and distributionally representative prompts by integrating data-wise and model-wise uncertainty learning. To the best of our knowledge, our UVPNet first utilizes uncertainty modeling with visual priors for single image deraining. Extensive experiment results demonstrate that our UVPNet outperforms state-of-the-art methods on both public synthetic datasets and real-world images while maintaining low complexity.

1 INTRODUCTION

Adverse weather conditions present formidable obstacles to visual perception tasks across a wide spectrum of applications such as Autonomous Driving(AD). In the domain of AD, rain can cause raindrops to obscure the camera lens, leading to blurred and distorted images of the road, traffic signs, and other vehicles. This can severely impede the vehicle’s ability to accurately detect and classify objects, potentially resulting in dangerous driving situations. Fog, on the other hand, reduces visibility by scattering light, making it difficult for cameras to distinguish between different objects in the scene. In surveillance systems, fog can render the captured images nearly useless for identifying individuals or detecting suspicious activities. Rain, prevalent in arid and semi-arid regions, not only introduces a large amount of particulate matter into the air but also causes light attenuation and scattering. In outdoor photography, rainy weather can turn a clear and beautiful landscape into a hazy and indistinguishable scene. These challenging weather conditions have spurred extensive research in the field of image restoration. A multitude of studies, as referenced inJiang et al. (2020); Zamir et al. (2021; 2022), have been dedicated to this area. The overarching goal of this research is to enhance the clarity and quality of images under adverse weather conditions. By improving image quality, more reliable visual data can be provided for various applicationsValanarasu et al. (2022), such as surveillanceZheng et al. (2021b); Cui & Knoll (2023) and remote sensingHe et al. (2010); Song et al. (2023). In autonomous driving, high-quality images enable the vehicle’s perception system to make more informed decisions, ensuring safer driving.

However, the complex nature of rainy weather poses unique challenges for current image restoration methods. Current methods cannot solve the high uncertainty problem. Rain streaks and rain drops introduce multiple degradation types simultaneously. These models typically focus on a single aspect of degradation, such as dehazing or denoising, and struggle to capture a comprehensive set of features when dealing with the complex interplay of multiple degradations in rain-affected imagesZamir et al. (2021). This uncertainty makes it challenging for these models to accurately estimate the true nature of the degradation and restore the image to its original state.

The inherent uncertainties associated with these multiple degradation types are difficult for existing methods to handle effectively. The high uncertainty of rain particles in the air can cause light scattering, absorption, and occlusion. As a significant factor in rainy image degradations, uncertainty-aware probabilistic modeling becomes even more critical.

In response to challenges of dealing with the high uncertainty of rainy images, we propose the Uncertainty-aware Visual-priors Prompt-interaction Network (UVPNet) in this paper. UVPNet consists of our proposed Distribution-aware Visual Priors Learning (DVPL) module and Certainty-Uncertainty Prompt Fusion (CUPF) module, which can represent aleatoric and epistemic uncertainty, respectively. Specifically, for data-wise uncertainty, our proposed DVPL models the visual priors of segment and depth under uncertainty, enabling the model to extract uncertainty information to guide its training. For model-wise uncertainty, our proposed CUPF allows prompt learning to capture the uncertainty with the fusion of a certainty prompt, thereby enabling the decoder to probabilistically reconstruct clear images. Furthermore, to address the uncertainty inherent in the neural network while enhancing the quality of restored images, we propose the Channel Spatial Uncertainty Weighting Block (CSUWB). CSUWB enhances UVPNet’s performance through uncertainty-aware feature processing.

In summary, the primary contributions of this paper can be categorized into four main aspects:

- To the best of our knowledge, we are the first to utilize uncertainty modeling with the segment and depth priors for the single image deraining task.
- We propose the Certainty-Uncertainty Prompt Fusion module to handle epistemic uncertainty in model dimensions, effectively enhancing model robustness.
- We propose the Channel Spatial Uncertainty Weighting Block to guide the feature selection process of our proposed UVPNet, thereby boosting overall performance.
- Through comprehensive experiments conducted on several public deraining datasets and real-world images, we demonstrate that our proposed method achieves **State-Of-The-Art(SOTA)** performance.

2 METHODS

2.1 OVERALL FRAMEWORK

We propose UVPNet in this paper to address the lack of research on high uncertainty in the task of single-image rain removal. The overall structure of our proposed UVPNet is as depicted in Fig. 1. Our network adopts the multi-input multi-output U-shaped encoder-decoder structure due to its excellent performance. The encoder receives three inputs, and the decoder outputs restored images at three different scales. We use four NAFBlocksChen et al. (2022) as the backbone network embedded in each CSUWB. At the encoder end, we also utilize the prior information provided by SAMKirillov et al. (2023) and Depth Anything V2Yang et al. (2024) in an uncertainty-aware manner; meanwhile, at the decoder end, we also employ prompt learning modules with uncertainty-aware capabilities.

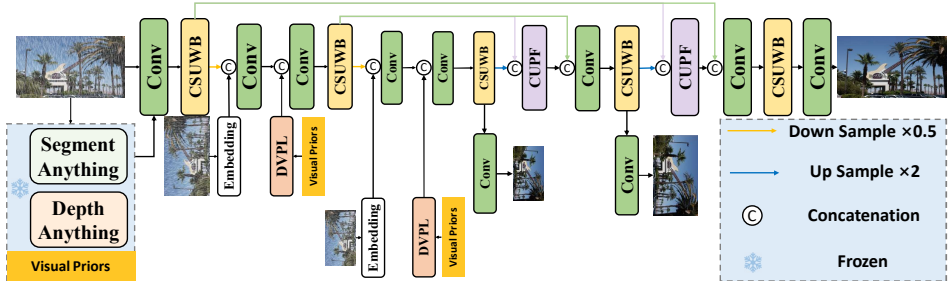


Figure 1: Overall framework of our proposed UVPNet.

2.2 DISTRIBUTION-AWARE VISUAL PRIORS LEARNING

To address the uncertainty in data dimensions for single image rain removal, we integrate uncertainty-aware visual priors provided by the Distribution-aware Visual Priors Learning (DVPL) into our UVPNet, thereby achieving an accurate representation for aleatoric uncertainties. We utilize a learnable distribution in the feature space to introduce uncertainty modeling into the rain streak and rain drop removal task. We use two independent convolutional layers f_μ and f_σ to achieve the feature distribution, and we model the distribution of DVPL at each pixel as Gaussian following Yang et al. (2021). The results are denoted as the mean μ and the variance σ of these two layers. These two convolutional layers learn a distribution $N(\mu, \sigma)$ parameterized by mean μ and variance σ . This approach allows the model to focus on the most valuable aspects inside the features while being aware of the inherent high uncertainty region in the feature space. The detailed formulation of this uncertainty-aware distribution modeling can be shown as:

$$Y \sim N(f_u(x), \epsilon \odot \sigma(x)) \quad (1)$$

where

$$\sigma(x) = \text{Softplus}(f_\sigma(x)) + \theta \quad (2)$$

where x and Y are the input feature and output result, *Softplus* denotes the Softplus activation function, θ is set to 0.001, \odot denotes the element-wise multiplication, and $\epsilon \sim N(0, 1)$. As shown in Fig.2(b), we apply the aforementioned distribution modeling process to extract the uncertainty within the segment and depth priors provided by SAMKirillov et al. (2023) and Depth Anything V2Yang et al. (2024), using it to characterize the data-wise aleatoric uncertainty. Furthermore, we employ the affine transformation to refine the features, leveraging this approach to enhance the representation of the data-wise inherent uncertainty. Finally, we couple these selected features with the original feature. This process can be described by the following Eq. 3:

$$\begin{aligned} P_S &= \text{Conv}([S_w * x + S_b, x]), \\ P_D &= \text{Conv}([D_w * x + D_b, x]) \end{aligned} \quad (3)$$

where S_w , S_b , D_w , and D_b are obtained by modeling the uncertainty of the segment priors and depth priors in an affine transformation way, P_S and P_D are the learned priors. $[\cdot]$ indicates the concatenation operation. Further, we designed a cross-branch weighted interaction to facilitate the cross-interaction between segmentation priors and depth features, as shown in the right part of Fig. 2(a). The goal is to cross-complement one type of visual prior feature with another. Segment prior features include details of edges and fine texture categories, so we utilize this information to enrich the exploration of depth visual prior features through a lightweight spatial feature weighting unit (Segment to Depth, S→D), as shown in Fig. 2(c). Similarly, the global information in depth prior features is passed to the segmentation prior uncertainty-aware distribution learning branch through a channel feature weighting unit (Depth to Segment, D→S), as shown in Fig. 2(d).

S→D: This module calculates the weight map of the spatial dimension based on the prior features of semantic segmentation, and then uses it to supplement the features of the depth prior branch, thereby improving the overall model feature extraction ability. The S→D unit utilizes the average pooling and max pooling of channel dimensions in parallel to generate two independent spatial feature maps, each with a size of $H \times W \times 1$. Then, these two feature maps are concatenated in the channel dimension, and the concatenated features are further refined through a convolution layer with a kernel size of 7×7 . Finally, the sigmoid activation function is used to generate the final spatial attention map. Overall, the detailed process of the S→D module can be formulated as:

$$A_{S \rightarrow D} = \delta(f_{7 \times 7}^{2 \rightarrow 1}([GAP_c(x), GMP_c(x)])) \quad (4)$$

where $f_{7 \times 7}^{2 \rightarrow 1}$ is the convolution layer with a kernel size of 7×7 and changes the channel of input features from 2 to 1; δ is the sigmoid function, GAP_c and GMP_c are the channel-wise global average pooling and max pooling, respectively.

D→S: D→S is an approximately symmetrical dual-branch module that processes the input depth visual prior features, generates a weight, and then uses it to process the segmentation prior features of uncertainty perception. Specifically, given the input depth prior feature $x \in \mathbb{R}^{H \times W \times C}$, the first branch of the D→S module applies global average pooling along the spatial dimension to obtain feature vectors of size $1 \times 1 \times C$, followed by two convolutional layers with an intermediate ReLU

162 activation. The other branch of the $D \rightarrow S$ module adopts the same structure, with the only difference
 163 being the use of max pooling to obtain features. Finally, add the results of the two branches and apply
 164 the sigmoid function to generate the final weight, which is used to modulate the uncertainty-aware
 165 segmentation prior. The detailed process of the $D \rightarrow S$ module can be expressed as:

$$166$$

$$167$$

$$168 A_{D \rightarrow S} = \delta(f_{1 \times 1}^{\frac{c}{r} \rightarrow c}(\gamma(f_{1 \times 1}^{c \rightarrow \frac{c}{r}}(GAP_s(x))))$$

$$169 + f_{1 \times 1}^{\frac{c}{r} \rightarrow c}(\gamma(f_{1 \times 1}^{c \rightarrow \frac{c}{r}}(GMP_s(x))))$$

$$170 \quad (5)$$

171 where GAP_s and GMP_s are the spatial-wise global average pooling and max pooling, respectively;
 172 γ denotes the ReLU activation function. $f_{1 \times 1}^{c \rightarrow \frac{c}{r}}$ has a reduction ratio r for the channel adjustment,
 173 while $f_{1 \times 1}^{\frac{c}{r} \rightarrow c}$ has an increasing ratio r .
 174

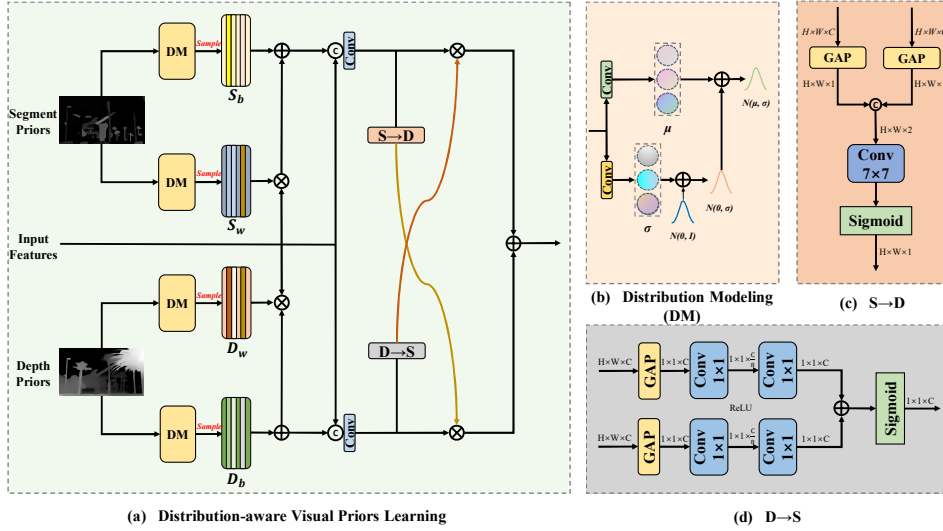


Figure 2: Detailed structure of our proposed DVPL.

2.3 CERTAINTY-UNCERTAINTY PROMPT FUSION

197 We introduce the Certainty-Uncertainty Prompt Fusion (CUPF) module engineered to mitigate the
 198 epistemic uncertainty intrinsic to neural networks. The inherent uncertainty within a model is termed
 199 epistemic uncertainty, and neural network models designed to learn fixed mappings lack the capa-
 200 bility to capture this uncertainty in the model dimension. To address this issue, CUPF integrates
 201 prompt learning with uncertainty-aware distribution modeling, enabling the model to represent its
 202 own inherent epistemic uncertainties with the fusion of certainty prompts.

203 Prompt-based techniques have been investigated for the parameter-efficient fine-tuning of large,
 204 frozen models trained on a source task to adapt them to a target task within the domains of
 205 NLP Brown et al. (2020); Houlsby et al. (2019) and vision Jia et al. (2022); Khattak et al. (2023).
 206 However, prompt learning lacks the capability to address uncertainties in the model dimensions.
 207 Consequently, we integrate prompt learning with uncertainty-aware distributional representations,
 208 thereby transforming prompt learning into a mitigator of model dimension epistemic uncer-
 209 tainties. In our proposed CUPF module, the prompt components serve as learnable parameters that
 210 interact with input features, thereby enriching the input features with information regarding the
 211 degree of rainy degradation. Given input features $x \in \mathbb{R}^{H \times W \times C}$ and N certainty prompt com-
 212 ponents $P_{cer} \in \mathbb{R}^{N \times H \times W \times C}$, and N uncertainty prompt components $P_{unc} \in \mathbb{R}^{N \times H \times W \times C}$, the
 distribution-aware prompt components can be obtained by:

$$213 P_{dis} \sim P_{cer} + \epsilon \odot \sigma_p(P_{unc}) \quad (6)$$

214 where

$$215 \sigma_p(P_{unc}) = \text{Softplus}(P_{unc}) + \theta \quad (7)$$

then the final fusion prompt components P are the sum of $W_c \odot P_{cer}$ and $W_d \odot P_{dis}$, where W_c and W_d are learnable parameters used for weighting this two branch. So the detailed formulation of the output Y_{CUPF} of our CUPF can be defined as:

$$Y_{CUPF} = PIM(PGM(P, x), x) \quad (8)$$

where PGM and PIM are the prompt generation module and prompt interaction module Potlapalli et al. (2023), respectively.

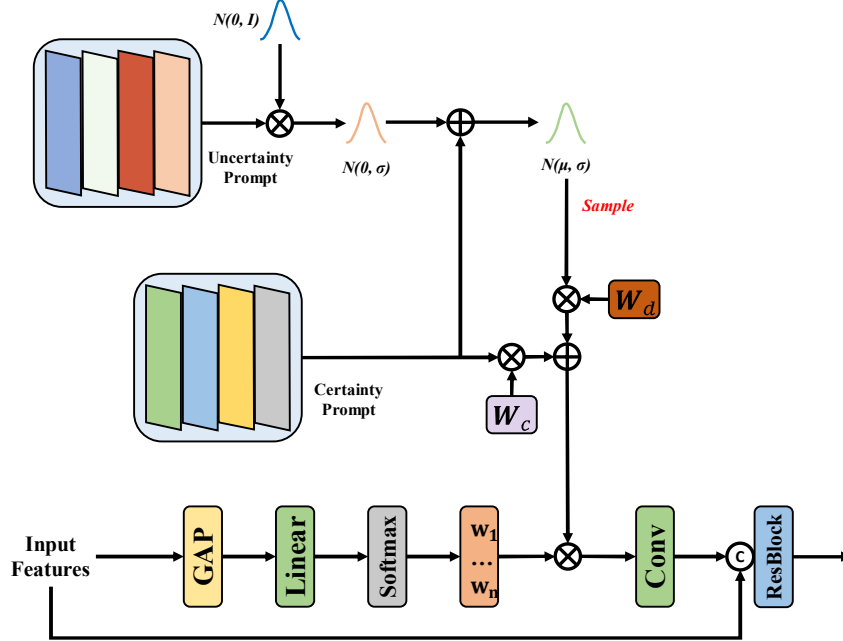


Figure 3: Detailed structure of our proposed CUPF.

2.4 CHANNEL SPATIAL UNCERTAINTY WEIGHTING BLOCK

This section introduces a novel feature representation method proposed in this paper, called Channel Spatial Uncertainty Weighting Block (CSUWB). In common standard convolutional neural networks, channel dimension mapping is performed on input features. There are many studies that weigh and refine the features among channels to improve the feature representation ability of neural networks. However, the weights corresponding to each channel in a convolutional neural network are randomly initialized, and the correspondence between multiple channels is also random, which results in significant uncertainty in the channel dimension of the convolutional neural network. To address this issue and characterize the uncertainty of channel dimension changes in convolutional neural networks, we propose Channel-wise Uncertainty-aware Weighting Block (CUWB), as shown in Fig. 4(a) and (b). Specifically, given input features x , we first project the input features through convolutional layers, while also using convolutional layers to project the features that have undergone channel shuffle operations; Then, the shuffled features are sent to an unshuffle operation, and the two projected features are subtracted element by element to obtain the uncertainty features U_c of the channel dimension, this process can be formulated as:

$$U_c = |Conv(x) - UShu(Conv(Shu(x)))| \quad (9)$$

where $Conv$ denotes the convolution layer for projection, $UShu$ and Shu are the channel-wise unshuffle operation and shuffle operation, respectively. Then, the obtained uncertainty features are subjected to global average pooling and global maximum pooling in spatial dimensions, and the feature information is aggregated into channel statistics. These statistics are input into an MLP with a reduction rate to generate channel attention weights. This process can be formulated as follows:

$$W_c = \delta(MLP(GAP_s(U_c) + GMP_s(U_c))) \quad (10)$$

270 These weights are used to scale the original features, highlighting channels related to uncertainty
 271 perception while suppressing irrelevant channels. Finally, the weighted uncertainty-aware features
 272 are concatenated with the original deterministic features, and the final mapping is performed by a
 273 convolution layer.

274 Similar to CUWB, we propose a novel module for measuring the uncertainty caused by feature
 275 changes in spatial dimensions, called the Spatial-wise Uncertainty Weighting Module (SUWM).
 276 Most neural networks used for image restoration are based on UNet architecture or Encoder-Decoder
 277 architecture, which inevitably require frequent sampling operations of features to reduce computa-
 278 tional complexity. Although many studies have investigated how to weigh spatial features, these
 279 studies have overlooked the impact of uncertainty caused by changes in feature maps on model
 280 performance. Meanwhile, since the model is generally trained using a fixed image size, the actual
 281 application of the model involves significant changes in image size, which introduces considerable
 282 uncertainty. So it is necessary to model the uncertainty caused by size changes in the spatial dimen-
 283 sion of features. To address this issue and characterize the uncertainty of spatial dimension changes
 284 in UNet or Encoder-Decoder architecture, we propose our novel SUWM in this section, as shown
 285 in Fig. 4(c) and (d). Unlike CUWB, we use convolutional layers to project both the original input
 286 features and the downsampled spatially transformed features, specifically. Then, the downsampled
 287 projected features are upsampled to restore their original spatial size, and then subtracted element
 288 by element from the original features to obtain uncertain spatial features U_s . This process can be
 formulated as:

$$289 U_s = |Conv(x) - Up(Conv(Down(x)))| \quad (11)$$

290 where Up and $Down$ are upsampling operation and downsampling operation, respectively. Then,
 291 the obtained uncertainty features are characterized using global average pooling and global max-
 292 imum pooling in the channel dimension, and are input into a convolutional layer for refinement.
 293 Finally, the spatial uncertainty weights are obtained through a sigmoid layer. This process can be
 294 formulated as:

$$295 W_s = \delta(Conv[GAP_c(U_s), GMP_c(U_s)]) \quad (12)$$

296 Finally, we fuse the features refined by CUWB and SUWM with the original features x represented
 297 by 4 NAFBlocksChen et al. (2022) using a residual connection. This process can be expressed as
 298 follows:

$$299 Y_{CSUWB} = x + x * W_c * W_s \quad (13)$$

300 We use CSUWB to weight the uncertainty perception of the original features. After refinement by
 301 CSUWB, the original features have the ability to sense uncertainty, greatly improving the perfor-
 302 mance of single image rain removal.

304 3 EXPERIMENTS

306 3.1 IMPLEMENTATION DETAILS AND DATASETS

308 We implement our framework and other methods using PyTorch. We train all the models for 200
 309 epochs with the AdamW optimizer from scratch for fairness. The initial learning rate is set to
 310 1×10^{-4} , and we employ CosineAnnealingLR to adjust the learning rate. For data augmentation,
 311 we apply horizontal flipping and randomly rotate the image by 0, 90, 180, 270 degrees. The batch
 312 size is set to 4. Computing complexity is computed on a patch size of 128×128 .

314 3.2 RESULTS

315 For datasets used for training and testing on the image rain streak removal task, we use 13,712
 316 image pairs collected from multiple datasetsLi et al. (2016); Yang et al. (2017); Zhang et al. (2019);
 317 Zhang & Patel (2018); Fu et al. (2017) for training by following previous methodsZamir et al.
 318 (2021; 2022); Cui et al. (2023). For testing the performance of rain streak removal, we use five
 319 synthetic datasets, Test100Zhang et al. (2019), Rain100LYang et al. (2017), Rain100HYang et al.
 320 (2017), Test1200Zhang & Patel (2018), Test2800Fu et al. (2017), and a public real-world rainy
 321 image datasetWang et al. (2019). Furthermore, for evaluating the performance of rain drop removal,
 322 we use the public datasetQian et al. (2018) for training and testing. In the comparative experiments
 323 with other methods, we compare our proposed method with various cutting-edge approaches and
 compute PSNR and SSIM scores following Li et al. (2025). For real-world rainy images, we use

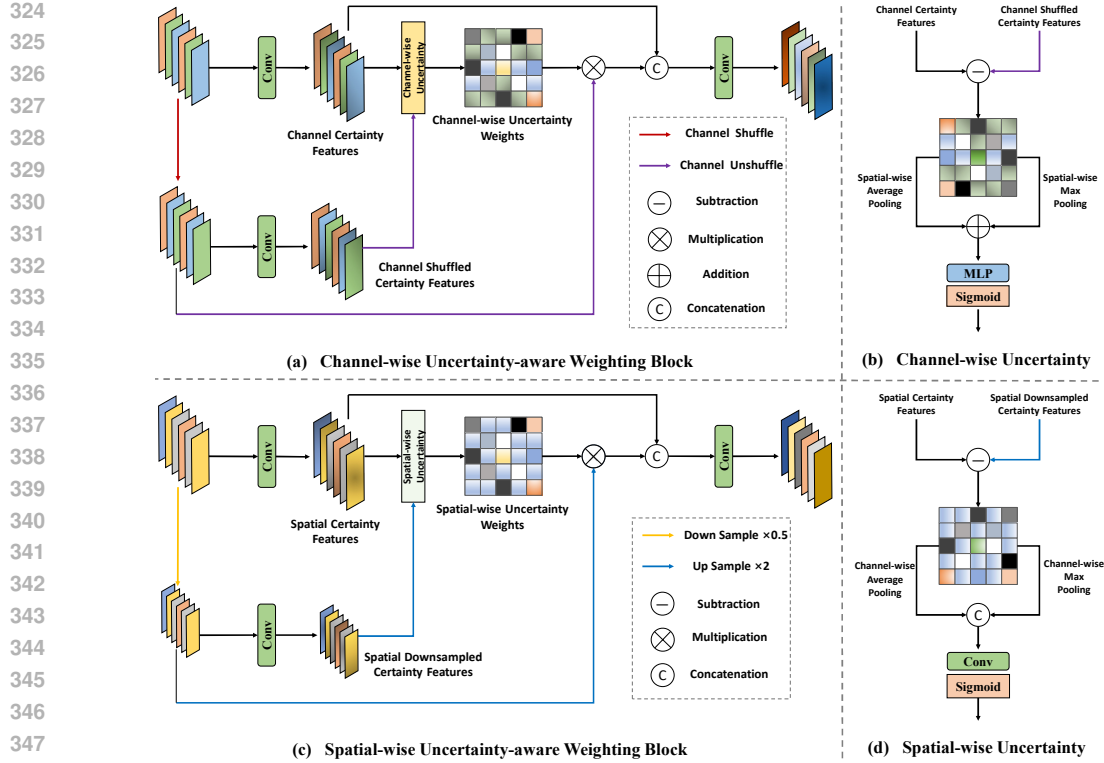


Figure 4: Detailed structure of our proposed CSUWB.

NIQEMittal et al. (2012b), PIQEVenkatanath et al. (2015), and BRISQUEMittal et al. (2012a) scores for evaluation. For the MaRLi et al. (2025) method, we utilize the UNet variant model specifically designed for image dehazing in their proposed approach.

For the rain streak removal task and rain drop removal task on synthetic images, as shown in Table 1 and Table 3, our method achieves significant performance gains more than other methods while maintaining fewer parameters and computational complexity as shown in Table 1. Red/blue text indicates best/second-best comparative results, the third-best comparative results are underlined. Furthermore, the visual results shown in Fig. 5 are in close alignment with the quantitative findings, substantiating the superior image rain streak removal capabilities of our proposed method. Various methods produced relatively similar restored images on synthetic data, making it difficult to distinguish differences visually. So, we use error maps Zheng et al. (2021a) between restoration results and ground-truth clear images to demonstrate the effectiveness of each method in restoring images. Our method produces the clearest error map, while others like MaRLi et al. (2025), MFSNetGao & Dang (2025), and PromptIRPotlapalli et al. (2023) show many errors in regions with sparser features, such as buildings and skies. Other methods, such as FMRNetJiang et al. (2024), MHNetGao et al. (2025), and CPRAFormerZou et al. (2025), perform poorly in the rain removal task, thus exhibiting larger errors in the error maps.

For real-world rainy images, as shown in the bottom two rows of Fig. 5, except for our method, which effectively removes rainy streaks from images, all other methods fail to do so in real-world scenarios. Additionally, no reference image quality metrics shown in Table 2 denote that our method can removal real-world rain streaks effectively.

3.3 DISCUSSION ON COMPUTING COMPLEXITY

In this part, we have listed relevant complexity measurement metrics, including Inference times (in ms), and Memory cost (in Mb), measured using a NVIDIA RTX4090 GPU. As shown in Table 1, our method maintains low Flops and parameters while achieving SOTA performance, while only occupying low memory at runtime and achieving fast inference time. Other methods, such as

MIMOUNet, NAFNet, and FMRNet, although simpler than our proposed method in a certain complexity metric, have poor image rain removal performance and cannot meet the quality requirements of actual image restoration. Some methods, such as PromptIR, MHNet, and MFSNet, significantly exceed the proposed method in terms of computational complexity metrics, and have shortcomings in image rain removal performance, making them cumbersome. Finally, the MaIR method is based on Mamba, and due to the complex image scanning operations involved, although it can maintain low Flops and Params, it consumes the most inference time.

Furthermore, due to the use of vision large models (i.e., SAM and Depth Anything V2) in our UVPNet to obtain priors, we also conducted research on scenarios where vision large models cannot be used. As shown in Table 5. The 0S, 0D, wS, and wD represent segmentation priors with tensor values all 0, depth priors with tensor values all 0, normal SAM priors, and Depth Anything V2 priors, respectively. From Table 5, it can be seen that without relying on SAM and Depth Anything V2, that is, when inputting all 0 tensors, although our method has a performance decrease, it is not significant and does not affect SOTA performance. This indicates that our method has the potential for real-time applications and can handle real-time requirements in application scenarios.

3.4 ABLATION STUDIES

For simplicity, we randomly selected 4,000 images from all the previous dataset used for training. As shown in Table 4, the ablation study of the above-mentioned modules proves the effectiveness of our proposed method. For example, using segment priors and depth priors in DVPL can improve performance. Meanwhile, fusing the certainty prompt and uncertainty prompt in the CUPF module can significantly enhance network performance. Finally, by embedding CUWB and SUWM in CSUWB can truly improve the image quality than using CBAMWoo et al. (2018) or Transformer blockZamir et al. (2022).

Table 1: Comparative results on Raindrop removal dataset.

Method	Venue	Raindrop-A		Raindrop-B		Inf. times ms	Mem. cost Mb
		PSNR↑	SSIM↑	PSNR↑	SSIM↑		
MIMOUNetCho et al. (2021)	ICCV'21	26.70	0.906	24.52	0.871	2.260	25.97
NAFNetChen et al. (2022)	ECCV'22	26.04	0.893	24.10	0.860	6.493	65.28
PromptIRPotlapalli et al. (2023)	Nips'23	<u>28.09</u>	<u>0.931</u>	<u>25.12</u>	<u>0.887</u>	16.453	135.77
FMRNetJiang et al. (2024)	AAAI'24	23.40	0.815	22.11	0.796	17.562	6.63
MHNetGao et al. (2025)	PR'25	25.31	0.882	23.64	0.851	13.476	257.66
MFSNetGao & Dang (2025)	TCSVT'25	27.01	0.910	24.62	0.873	14.601	351.54
CPRAFormerZou et al. (2025)	ACMMM'25	<u>28.13</u>	<u>0.932</u>	<u>25.14</u>	<u>0.887</u>	32.548	111.68
MaIRLi et al. (2025)	CVPR'25	27.24	0.917	24.65	<u>0.875</u>	53.165	12.99
Ours	-	28.36	0.935	25.28	0.892	8.580	49.34

Table 2: Comparative results on real_test_1000 datasetWang et al. (2019).

Method	PromptIR	FMRNet	MHNet	MFSNet	CPRAFormer	MaIR	Ours
NIQE↓	<u>6.00</u>	7.65	8.20	8.01	<u>5.96</u>	7.32	5.86
BRISQUE↓	40.70	<u>39.98</u>	39.64	40.32	41.34	<u>41.95</u>	39.82
PIQE↓	<u>58.18</u>	64.48	64.58	66.94	<u>58.80</u>	71.58	56.85

Table 3: Results of synthetic rainy images on several public datasets.

Method	Venue	Rain100L		Rain100H		Test1200		Test100		Test2800		Average	
		PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
MIMOUNetCho et al. (2021)	ICCV'21	29.68	0.931	27.47	0.899	<u>29.48</u>	<u>0.934</u>	<u>29.04</u>	<u>0.931</u>	<u>28.86</u>	<u>0.937</u>	<u>28.90</u>	0.926
NAFNetChen et al. (2022)	ECCV'22	30.73	0.950	27.62	0.898	28.66	<u>0.930</u>	28.41	0.928	28.50	0.934	28.78	0.928
PromptIRPotlapalli et al. (2023)	Nips'23	31.08	0.938	28.01	0.900	28.42	0.928	28.15	0.926	28.35	<u>0.935</u>	28.80	0.925
FMRNet Jiang et al. (2024)	AAAI'24	28.17	0.903	25.91	0.859	26.94	0.921	26.72	0.919	27.33	0.926	27.01	0.905
MHNetGao et al. (2025)	PR'25	<u>31.12</u>	<u>0.951</u>	27.67	0.896	28.01	0.925	27.77	0.923	28.17	0.932	28.54	0.925
MFSNetGao & Dang (2025)	TCSVT'25	30.68	0.946	27.89	0.904	28.01	0.926	27.76	0.924	28.12	0.933	28.49	0.926
CPRAFormerZou et al. (2025)	ACMMM'25	32.73	0.968	29.03	0.920	27.74	0.923	27.49	0.922	27.74	0.930	<u>28.94</u>	<u>0.932</u>
MaIRLi et al. (2025)	CVPR'25	31.05	0.949	<u>28.24</u>	<u>0.905</u>	<u>29.46</u>	<u>0.934</u>	<u>29.15</u>	<u>0.932</u>	<u>28.93</u>	<u>0.937</u>	<u>29.36</u>	<u>0.931</u>
Ours	-	31.72	0.955	28.69	0.913	29.50	0.938	29.21	0.936	29.00	0.939	29.62	0.936

4 CONCLUSION

In this paper, we propose Uncertainty-aware Visual-priors Prompt-interaction Network(UVPNet) to address the impact of high-uncertainty in rainy scenes on visual perception data. For the first

Table 4: Ablation studies of components proposed on Test1200 datasetZhang & Patel (2018).

Configuration	Experiment setting	PSNR \uparrow	SSIM \uparrow
Baseline	-	28.34	0.887
DVPL	w/o Segment priors	28.56	0.888
	w/o Depth priors	28.60	0.891
	w/o uncertainty modeling	28.76	0.891
	w/o attentive weighting	28.62	0.891
	Ours	28.85	0.892
CUPF	fixed all prompt	28.60	0.890
	fixed only certainty prompt	28.63	0.889
	w/o certainty prompt	28.70	0.893
	w/o uncertainty promptPotlapalli et al. (2023)	28.71	0.891
	Ours	28.73	0.893
CSUWB	w/o CUWB in CSUWB	28.55	0.891
	w/o SUWM in CSUWB	28.59	0.891
	w/ CSUWB \leftarrow CBAMWoo et al. (2018)	28.67	0.892
	w/ CSUWB \leftarrow TransformerZamir et al. (2022)	28.67	0.890
	Ours	28.69	0.894

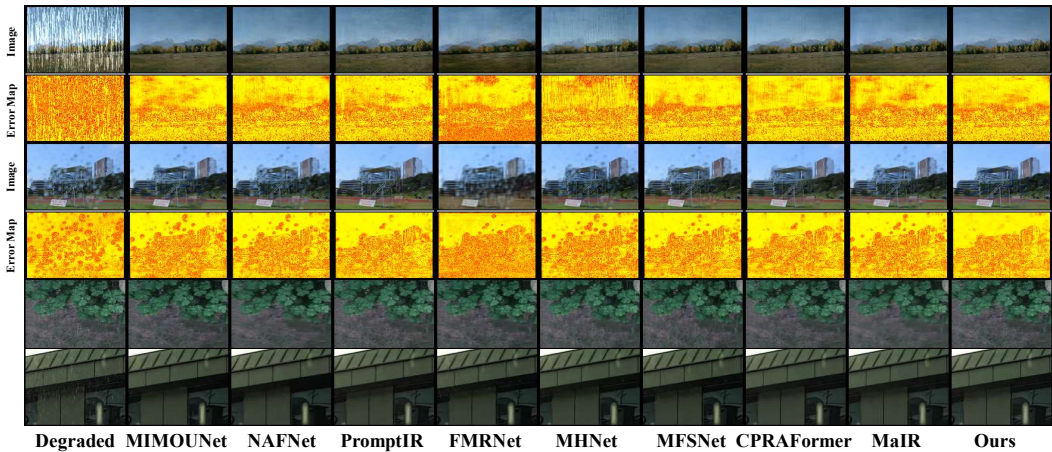


Figure 5: Visual results of comparative methods and our UVPNet on synthetic rainy images and real-world rainy images.

Table 5: Comparative results on the assistance of segment priors and depth priors.

Method	Venue	Rain100H		Test100		Raindrop-B	
		PSNR \uparrow	SSIM \uparrow	PSNR \uparrow	SSIM \uparrow	PSNR \uparrow	SSIM \uparrow
MFSNetGao & Dang (2025)	TCSVT'25	27.89	0.904	27.76	0.924	24.62	0.873
MaIRLi et al. (2025)	CVPR'25	28.49	0.905	29.15	0.932	24.65	0.875
OR&OD	-	28.60	0.912	29.07	0.936	25.22	0.890
wR&wD	-	28.69	0.913	29.21	0.936	25.28	0.892

time, UVPNet combines the segment priors provided by SAM and depth priors provided by Depth Anything V2 with distribution modeling, effectively extracting the visual priors in an uncertainty-aware manner. Furthermore, with the integration of certainty fusing uncertainty prompt learning, UVPNet excellently handles the model-wise epistemic uncertainty in the high-uncertainty rain streak and rain drop removal task. Comparative experiments on both publicly synthesized datasets for rain streak and rain drop removal and real-world rainy images, along with computing complexity and ablation studies, demonstrate the effectiveness and rationality of our method with and without the assistance of vision large models.

REFERENCES

- 486
487
488 Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal,
489 Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are
490 few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- 491 Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration.
492 In *European Conference on Computer Vision*, pp. 17–33. Springer, 2022.
- 493
494 Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-
495 to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international con-
496 ference on computer vision*, pp. 4641–4650, 2021.
- 497 Yuning Cui and Alois Knoll. Exploring the potential of channel interactions for image restoration.
498 *Knowledge-Based Systems*, 282:111156, 2023.
- 499
500 Yuning Cui, Wenqi Ren, Xiaochun Cao, and Alois Knoll. Image restoration via frequency selection.
501 *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- 502
503 Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing
504 rain from single images via a deep detail network. In *Proceedings of the IEEE conference on
505 computer vision and pattern recognition*, pp. 3855–3863, 2017.
- 506
507 Hu Gao and Depeng Dang. Exploring richer and more accurate information via frequency selection
508 for image restoration. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.
- 509
510 Hu Gao, Ying Zhang, Jing Yang, and Depeng Dang. Mixed hierarchy network for image restoration.
511 *Pattern Recognition*, 161:111313, 2025.
- 512
513 Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE
514 transactions on pattern analysis and machine intelligence*, 33(12):2341–2353, 2010.
- 515
516 Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, An-
517 drea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp.
518 In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.
- 519
520 Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and
521 Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727.
522 Springer, 2022.
- 523
524 Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun
525 Jiang. Multi-scale progressive fusion network for single image deraining. In *Proceedings of the
526 IEEE/CVF conference on computer vision and pattern recognition*, pp. 8346–8355, 2020.
- 527
528 Kui Jiang, Junjun Jiang, Xianming Liu, Xin Xu, and Xianzheng Ma. Fmrnet: Image deraining
529 via frequency mutual revision. In *Proceedings of the AAAI Conference on Artificial Intelligence*,
530 volume 38, pp. 12892–12900, 2024.
- 531
532 Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shah-
533 baz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference
534 on Computer Vision and Pattern Recognition*, pp. 19113–19122, 2023.
- 535
536 Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete
537 Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceed-
538 ings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.
- 539
540 Boyun Li, Haiyu Zhao, Wenxin Wang, Peng Hu, Yuanbiao Gou, and Xi Peng. Mair: A locality- and
541 continuity-preserving mamba for image restoration. In *IEEE Conference on Computer Vision and
542 Pattern Recognition*, Nashville, TN, June 2025.
- 543
544 Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using
545 layer priors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,
546 pp. 2736–2744, 2016.

- 540 Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assess-
541 ment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012a.
542
- 543 Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a “completely blind” image quality
544 analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012b.
545
- 546 Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. Promptir:
547 Prompting for all-in-one image restoration. *Advances in Neural Information Processing Systems*,
548 36:71275–71293, 2023.
- 549 Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial
550 network for raindrop removal from a single image. In *Proceedings of the IEEE conference on*
551 *computer vision and pattern recognition*, pp. 2482–2491, 2018.
552
- 553 Yuda Song, Zhuqing He, Hui Qian, and Xin Du. Vision transformers for single image dehazing.
554 *IEEE Transactions on Image Processing*, 32:1927–1941, 2023.
- 555 Jeya Maria Jose Valanarasu, Rajeev Yasarla, and Vishal M Patel. Transweather: Transformer-based
556 restoration of images degraded by adverse weather conditions. In *Proceedings of the IEEE/CVF*
557 *Conference on Computer Vision and Pattern Recognition*, pp. 2353–2363, 2022.
558
- 559 Narasimhan Venkatanath, D Praneeth, S Channappayya Sumohana, S Medasani Swarup, et al. Blind
560 image quality evaluation using perception based features. In *2015 twenty first national conference*
561 *on communications (NCC)*, pp. 1–6. IEEE, 2015.
562
- 563 Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. Spatial atten-
564 tive single-image deraining with a high quality real rain dataset. In *Proceedings of the IEEE/CVF*
565 *conference on computer vision and pattern recognition*, pp. 12270–12279, 2019.
- 566 Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block
567 attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pp.
568 3–19, 2018.
- 569 Fan Yang, Qiang Zhai, Xin Li, Rui Huang, Ao Luo, Hong Cheng, and Deng-Ping Fan. Uncertainty-
570 guided transformer reasoning for camouflaged object detection. In *Proceedings of the IEEE/CVF*
571 *international conference on computer vision*, pp. 4146–4155, 2021.
572
- 573 Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang
574 Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911,
575 2024.
576
- 577 Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep
578 joint rain detection and removal from a single image. In *Proceedings of the IEEE conference on*
579 *computer vision and pattern recognition*, pp. 1357–1366, 2017.
- 580 Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-
581 Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the*
582 *IEEE/CVF conference on computer vision and pattern recognition*, pp. 14821–14831, 2021.
583
- 584 Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-
585 Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceed-*
586 *ings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5728–5739,
587 2022.
- 588 He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense
589 network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.
590 695–704, 2018.
591
- 592 He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image de-raining using a conditional generative
593 adversarial network. *IEEE transactions on circuits and systems for video technology*, 30(11):
3943–3956, 2019.

- 594 Chuanjun Zheng, Daming Shi, and Yukun Liu. Windowing decomposition convolutional neural
595 network for image enhancement. In *Proceedings of the 29th ACM International Conference on*
596 *Multimedia*, pp. 424–432, 2021a.
- 597
598 Zhuoran Zheng, Wenqi Ren, Xiaochun Cao, Xiaobin Hu, Tao Wang, Fenglong Song, and Xiuyi Jia.
599 Ultra-high-definition image dehazing via multi-guided bilateral learning. In *2021 IEEE/CVF Con-*
600 *ference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16180–16189. IEEE, 2021b.
- 601
602 Shun Zou, Yi Zou, Juncheng Li, Guangwei Gao, and Guojun Qi. Cross paradigm representation and
603 alignment transformer for image deraining. In *ACM MM*, 2025.

604 605 A APPENDIX

606 607 B THE USE OF LARGE LANGUAGE MODELS (LLMs)

608
609 The author(s) acknowledge that LLMs have been used to polish the language of this paper.
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647