Innovating Real Fisheye Image Correction with Dual Diffusion Architecture

Shangrong Yang, Chunyu Lin*, Kang Liao, Yao Zhao
Institute of Information Science, Beijing Jiaotong University
Beijing Key Laboratory of Advanced Information Science and Network, Beijing, 100044, China
{sr_yang, cylin, kang_liao, yzhao}@bjtu.edu.cn

Abstract

Fisheye image rectification is hindered by synthetic models producing poor results for real-world correction. To address this, we propose a Dual Diffusion Architecture (DDA) for fisheye rectification that offers better practicality. The DDA leverages Denoising Diffusion Probabilistic Models (DDPMs) to gradually introduce bidirectional noise, allowing the synthesized and real images to develop into a consistent noise distribution. As a result, our network can perceive the distribution of unlabelled real fisheye images without relying on a transfer network, thus improving the performance of real fisheye correction. Additionally, we design an unsupervised one-pass network that generates a plausible new condition to strengthen guidance and address the non-negligible indeterminacy between the prior condition and the target. It can significantly affect the rectification task, especially in cases where radial distortion causes significant artifacts. This network can be regarded as an alternate scheme for fast producing reliable results without iterative inference. Compared to the state-of-the-art methods, our approach achieves superior performance in both synthetic and real fisheye image corrections.

1. Introduction

Many applications [1][2][3] have significant demands for large field-of-view environment information. Therefore, the fisheye camera is naturally taken into account. However, the images captured by fisheye cameras have structure distortion, which can significantly impact the performance of subsequent vision algorithms [4][5][6]. To retain the performance of downstream tasks, one can consider correcting distorted images or redesigning subsequent algorithms. Many individuals prefer the simple former.

Most existing methods determined distortion parameters by identifying relevant features. Non-automatic calibration methods [7][8][9][10] detect corners artificially using a checkerboard, while automatic methods [11][12] use an algorithm that recognizes distinctive curves automatically. However, faulty detecting characteristics significantly impact these methods. As a result, neural networks are utilized to extract features based on their stable properties. [13][14][15][16] use deep regression models to predict distortion parameters. [17][18][19] consider it simpler to transform the correction into an image-to-image generation solution. By learning the empirical distributions, they can obtain the corrected results directly. Despite deep learning methods achieving significant advances in distortion correction, their training heavily depends on synthetic datasets. It leads to poor performance on real-world fisheye correction.

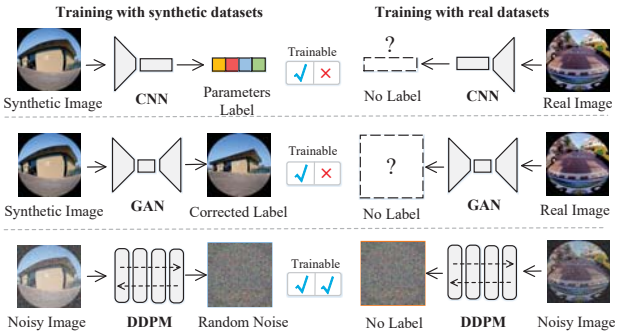

Figure 1. Most CNNs and GANs need labels for training. Without corresponding labels, DDPMs can perceive the real fisheye distribution during the training process by only supervising the noise.

One potential reason for poor results is due to the lack of labels, the real fisheye images cannot be used in training. Most convolutional neural networks (CNNs) and generative adversarial networks (GANs) require paired images for training, as illustrated in Figure 1. However, denoising diffusion probabilistic models (DDPMs) can be trained by supervising noise, which enables them to perceive the distribution of real fisheye images during training without requiring corresponding labels. Therefore, we utilize DDPMs [20][21] to explore real-world distortion correction and design a dual diffusion architecture (DDA) to handle the two available datasets. One part of our dual diffusion

*Corresponding author: cylin@bjtu.edu.cn

architecture is a conditional diffusion module, which learns the fundamental distribution of distortion from paired synthetic datasets in a supervised manner. The other part is an unconditional diffusion module, which leverages an unsupervised manner to perceive unlabeled real fisheye images during training. By gradually introducing noise in DDA, both synthetic and real fisheye images can gradually develop into a consistent noise distribution, as demonstrated in Figure 2. We can simultaneously train two modules by supervising the consistent noise. This approach eliminates the need for a specific network, like CycleGAN [22], to transform unlabeled real fisheye images explicitly for supervision. Our implicit transformation achieves good alignment for different datasets, allowing the knowledge learned from paired synthetic images to be utilized to enhance the perception of real fisheye images. As a result, the correction performance of real fisheye gains improvement. Due to the alignment in noise space, the trained model can be directly used for real fisheye correction.

As fisheye correction differs from other generation tasks, structural distortion causes notable disparities between the prior condition and target, which severely affects the generation quality of DDPMs. Therefore, we design a one-pass network embedded in the conditional diffusion module. It reduces the disparities by pre-correcting fisheye images, as illustrated in Figure 3. The corrected image can be used as a more plausible condition for DDPMs. Benefiting from the DDA, our embedded one-pass network learns in an unsupervised manner. After training, the one-pass network can be independently employed to rectify fisheye images without time-consuming inference.

Our contributions are summarized as follows:

- We propose a novel dual diffusion architecture that can simultaneously learn both synthetic and real image distributions through noise-space equivalence, thus improving the performance of real fisheye correction.
- To reduce the disparities between the prior condition and target, we design an unsupervised one-pass network to generate a plausible new condition. It can be used as an additional efficient correction approach.
- Distinguishing from previous methods, our approach pioneers to leverage unlabeled real fisheye images for training, achieving satisfactory results in both synthetic and real fisheye correction.

2. Related Work

The target of distortion correction is to restore the structure of the image before using downstream algorithms [23][24][25][26][5]. Early researchers [27][12][28] [29] noticed that straight lines captured with conventional lenses

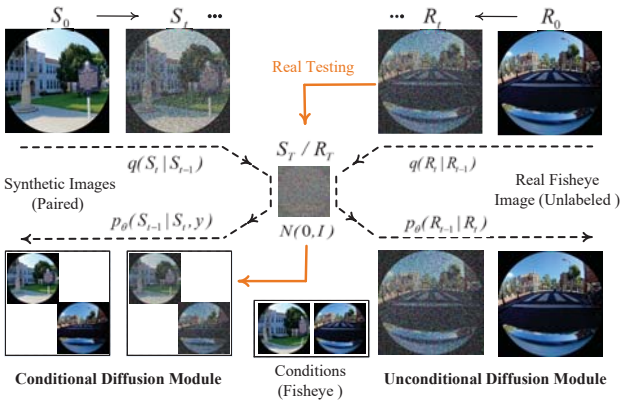

Figure 2. Dual diffusion architecture. Due to a consistent noise distribution, the knowledge learned from paired synthetic images improves the perception of unlabeled real fisheye images. Besides, the trained model can be directly used for real fisheye correction.

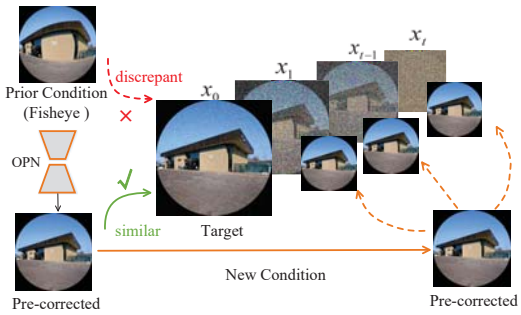

Figure 3. OPN pre-corrects the prior condition to alleviate the disparities and provide a new plausible condition.

appear curved in fisheye perspective. Therefore, it is necessary to locate the feature corners or lines. Mei et al. [27] developed a flexible calibration approach that uses corner points to calculate distortion parameters. However, it required additional standard planar grids and manual searching. Melo et al. [12] designed an unsupervised calibration method that employs an automatic detection algorithm to find 'a minimum of three lines' for calibration. Although the automatic method [12][29] is more flexible than the manual method [27], feature detection was susceptible to image content, thereby hampering accurate calculation.

Many researchers employed reliable neural networks to tackle the problem of distortion. Rong et al. [13] first predicted multiple distortion intervals using convolutional neural networks (CNNs). However, the initial correction is incomplete due to the limitations of the network and quantization intervals. [15][16] enhanced the regression network and integrated prior information such as semantics and edges for guidance. These methods significantly improved correction performance, but the image-parameters disparity limits the accurate prediction of all parameters. Therefore, [17][30][31] introduced generation-based methods to gen-

erate corrected images with learned empirical distributions. Liao et al. [17] utilized generative adversarial networks (GANs) to generate rectified results, but they noticed visible artifacts when using naive GANs. To enhance the quality of the corrected image further, [18][30][31] proposed a multi-stage generation to separate the structure correction from the content reconstruction. By learning the corresponding relationship between two explicit distributions, the correction performance was boosted.

Access to a large number of real fisheye images and corresponding labels can be challenging in practice. As a result, the aforementioned deep-learning methods only leveraged synthetic datasets for training. However, there is a discrepancy between the synthetic and the real datasets, the model trained on the synthetic images produces unacceptable results on the real images. Therefore, we propose a dual diffusion architecture. With the help of DDPMs, we can train with both paired synthetic datasets and unlabeled real fisheye datasets. The distribution of the real fisheye images is perceived in the training stage, thus improving the real image correction performance.

3. Preliminaries

3.1. Fisheye Model

The image was generated by projecting 3D space coordinates onto a 2D plane through a camera model. To capture a wider FoV and as much information as possible, the fisheye model alters the pinhole model $d = ftan\theta$ to a nonlinear relationship between the incidence angle θ and the emergence angle ρ :

$$\rho = k_1\theta + k_2\theta^3 + k_3\theta^5 + \cdots \tag{1}$$

However, this model involves precise angle calculation. To simplify this process, traditional methods [32][29][28] summarized two simple models: the polynomial model [32] and the division model [29]. They neglect the angle and directly perform the coordinate transformation from the perspective image to the fisheye image. Although the principles of both models are similar, the polynomial model does not require handling cases where the denominator is 0. Therefore, we apply the polynomial model to synthesize the fisheye dataset. It can be written as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} = (1 + \lambda_1(r')^2 + \lambda_2(r')^4 + \lambda_3(r')^6 + \cdots) \begin{bmatrix} x' \\ y' \end{bmatrix} \tag{2}$$

Where the coefficient of the polynomial λ_n reflects the distortion degree. (x', y') is an arbitrary point on the fisheye image, with its corresponding point on the perspective image being (x, y). r' is the distortion radius, which can be calculated by the Euclidean distance from (x', y') to the distortion center (x_d, y_d). Similarly, the undistorted radius r is the distance from (x, y) to the image center (x_0, y_0) on the perspective image.

3.2. Diffusion Model

DDPMs [20][21][33][34] are different from previous generation models [35] [36] [37]. It breaks down an image generation task into several subtasks, which include a forward process q with progressive noise addition and a reverse process p with iterative noise removal. Generally, the forward process can be represented as:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\boldsymbol{I}) \tag{3}$$

β_t represents the variance schedule utilized for generating Gaussian noise at each step. The data distribution x_t can be calculated from the previous distribution x_{t-1}. As opposed to the forward process, the reverse process needs to denoise beginning with $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In general, It can be written as neural network parameterization [38][39]:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{4}$$

The objective is to train networks $\mu_\theta(\cdot)$ and $\Sigma_\theta(\cdot)$ to minimize the distance D of the forward and backward distribution. D can be calculated according to KL-divergence:

$$D = D_{KL}(q(x_{t-1}|x_t)||p_\theta(x_{t-1}|x_t)) \tag{5}$$

Therefore, the optimization function of the unconditional diffusion model can be written as:

$$\mathcal{L}_u(\theta) = \mathbb{E}\left\|\epsilon_t - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, t)\right\|_1 \tag{6}$$

where ϵ_t is the noise ground truth. $\epsilon_\theta(\cdot)$ represents the diffusion model. $\bar{\alpha}_t$ can be calculated from $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$.

For the conditional diffusion model, we need to add additional condition y to the network [40][41][42] and replace t with continuous noise level [43][41][42]. Therefore, the optimization of the conditional diffusion model becomes:

$$\mathcal{L}_c(\theta) = \mathbb{E}\left\|\epsilon_t - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon_t, \bar{\alpha}_t, y)\right\|_1 \tag{7}$$

4. Architecture

Most generative-based correction methods [18][30][31] rely on paired synthetic datasets for training, leading to blurred effects on the real fisheye correction. Therefore, as shown in Figure 4, we propose a dual diffusion architecture (DDA) consisting of a conditional (CDM) and unconditional (UDM) diffusion module, as well as a one-pass network (OPN). In training, the OPN predicts flow and generates a coarse corrected image, which replace the original fisheye image as a new guidance. The CDM predicts the noise in the synthetic image guided by the new condition, while the UDM simultaneously predicts noise in the unlabeled real fisheye image. Our DDA supervises these two noises to map synthetic and real images to a consistent noise distribution. In testing, we optimize our network by solely utilizing the OPN and CDM to predict noise since the UDM cannot benefit from the guidance of conditions. The high-quality correction results of real and synthetic fisheye images are obtained after denoising.

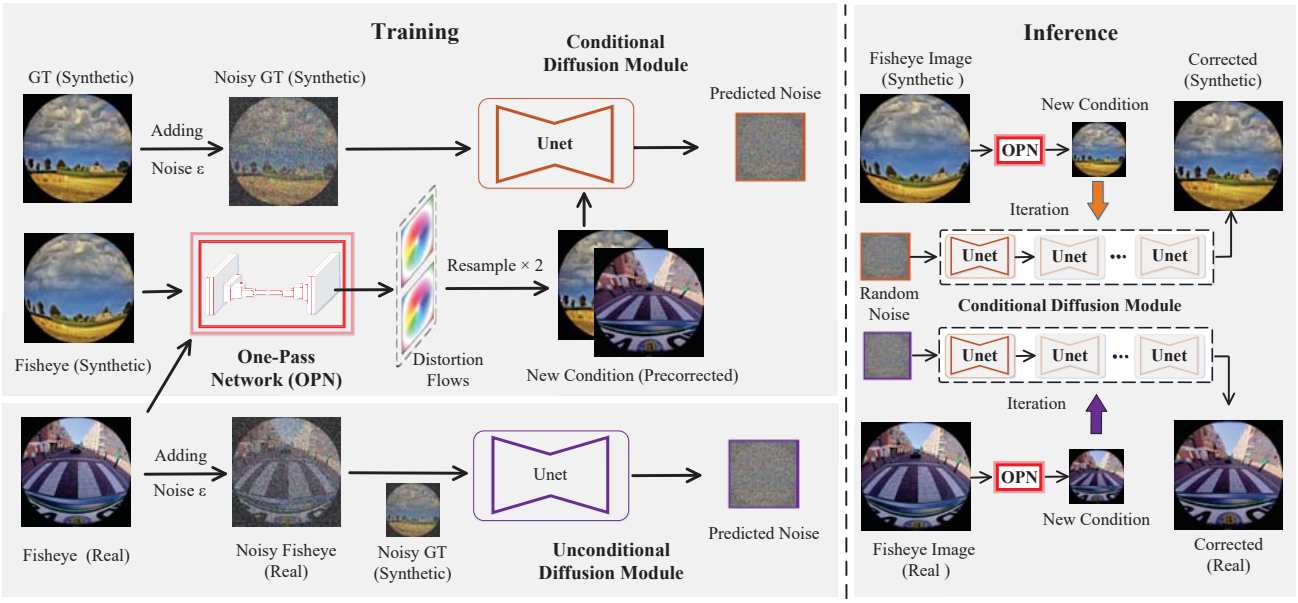

Figure 4. Our network consists of a conditional (CDM) and unconditional (UDM) diffusion module, as well as a one-pass network (OPN). During training, OPN pre-corrects fisheye images and provides new guidance for CDM. CDM and UDM perceive paired synthetic images and unlabeled real fisheye, mapping them to a consistent noise distribution via supervised noise. During inference, we optimize our network using only CDM to predict noise. Correction results of real and synthetic fisheye images are generated after denoising.

4.1. Conditional Diffusion Module (CDM)

Although it is difficult to obtain corresponding labels for real fisheye images, we can synthesize paired fisheye datasets. We fully leverage this paired resource by employing the conditional diffusion model (CDM) to perceive the fundamental distortion distribution. Typically, the input of CDM contains noisy target images and conditional images, which correspond to the noisy synthetic ground truth $\bar{S}_{gt}$ and synthetic fisheye image S_f in our correction task. The aim of the CDM is to use the fisheye image as a guide to gradually generate an image with a similar distribution to the synthetic gt S_{gt} during the denoising process. However, the distortion in the fisheye image can be extremely misleading for the generation. To address this, we use a one-pass network (OPN) to pre-correct the S_f and generate a coarse pre-correction S_p, which replaces the original fisheye image as more plausible guidance. Since we also need to correct the real fisheye image R_f, its pre-corrected result R_p should also be used as the CDM condition. Therefore, CDM takes both the $\bar{S}_{gt}$ and a new condition y_n concatenated by S_p and R_p to predict the noise $\epsilon_t{}'$ in the $\bar{S}_{gt}$. The CDM architecture consists of an encoder and a decoder, each with four scales output. It is a traditional Unet and the output channels are 64, 128, 256, and 512, respectively. Therefore, the optimization function of CDM is:

$$\mathcal{L}_{syn} = \mathbb{E}\left\|\epsilon - C_\theta(\sqrt{\bar{\alpha}_t}S_{gt} + \sqrt{1-\bar{\alpha}_t}\epsilon, \bar{\alpha}_t, y_n)\right\|_1 \quad (8)$$

where $C_\theta(\cdot)$ represents the conditional diffusion module.

4.2. One-Pass Network (OPN)

In CDM, fisheye images serve as a guiding condition. However, the significant distortion in fisheye images misleads image generation and causes severe artifacts. Therefore, the fisheye image is not suitable to be directly used as guidance. An intuitive idea is to alleviate the disparities between the prior condition and target by pre-correcting. We embedded a one-pass network (OPN) in the CDM to provide a more reasonable condition. The architecture of OPN is shown in Figure 5. OPN is an encoder and decoder structure, with each having six convolutional layers. The channels for each layer are 32, 32, 64, 128, 256, and 512, respectively. It takes the original fisheye image as input and generates a two-channel distortion flow W that reflects the image distortion degree. The W is used to resample the original fisheye images (S_f and R_f) and generate the pre-corrected images (S_p and R_p). We reference the TPS [44] to warp the image, as it is differentiable, which guarantees that the gradient can be backpropagated from the CDM to the OPN. Finally, we replace the fisheye image with the pre-corrected image as a new condition y_n to assist the network in predicting the noise $\epsilon_t{}'$. Since the distortion in y_n has been greatly reduced, the network can predict more accurately.

It is worth mentioning that our OPN is an unsupervised network that relies on CDM for training. Upon obtaining W from OPN, we use it directly without requiring flow labels for supervision. Because the entire aforementioned pro-

cess is differentiable, the input of CDM includes the output of OPN and noisy GT. Even though CDM supervises the noise, the gradient can still be backpropagated and accurately guide the flow prediction of OPN. Furthermore, OPN offers an alternative fast correction scheme. Due to the distortion flow, we can promptly correct the fisheye images, thereby avoiding the time-consuming inference in classic DDPMs. The experiments demonstrate that the corrected results from OPN also provide satisfactory performance as inference results.

4.3. Unconditional Diffusion Module (UDM)

The fundamental distortion distribution learned from paired synthetic datasets by OPN and CDM is not enough to correct real fisheye images. Because most methods solely rely on paired synthetic datasets for training, resulting in poor performance for real fisheye correction. To solve this problem, we attempt to learn the real image distribution directly from unlabeled real fisheye images via the unconditional diffusion module (UDM). The UDM structure is similar to the conditional diffusion module (CDM), but with a different input format. We concatenate the noisy synthetic $\bar{S}_{gt}$ and noisy real image $\bar{R}_f$ as input. $\bar{S}_{gt}$ are not used as conditions because UDM does not require input conditions. Besides, there is no OPN in the UDM, it is equivalent to performing a denoising task. Since noisy real fisheye images are unlabeled and can only be processed by UDM, CDM must employ UDM to learn the real distribution. To enhance the network's ability to learn the real and synthetic distribution, the same noise applied to the synthetic image is also applied to the real fisheye image. Subsequently, the UDM predicts their same noise $\epsilon_t{}'$. Therefore, the optimization target of the unconditional diffusion module can be represented as follows:

$$\mathcal{L}_{real} = \mathbb{E}\left\|\epsilon - U_\theta(\sqrt{\bar{\alpha}_t}R_f + \sqrt{1-\bar{\alpha}_t}\epsilon, \sqrt{\bar{\alpha}_t}S_{gt} + \sqrt{1-\bar{\alpha}_t}\epsilon, \bar{\alpha}_t)\right\|_1 \tag{9}$$

where $U_\theta(\cdot)$ refers to unconditional diffusion network.

4.4. Training strategy

In our work, we avoid relying on pre-trained networks (e.g. CycleGAN [22]) for bidirectional image transformation. This explicit transformation is inefficient for fisheye correction, which needs to correct the structure and reconstruct the content simultaneously. The pre-trained network cannot guarantee consistency between source and target distribution. Therefore, training a network by supervising transformed images is not convincing. In contrast, we use an implicit transformation that utilizes DDA to map data with different distributions onto a consistent noise space. In this noise space, different images can achieve good alignment. As a result, our DDA can complete efficient training

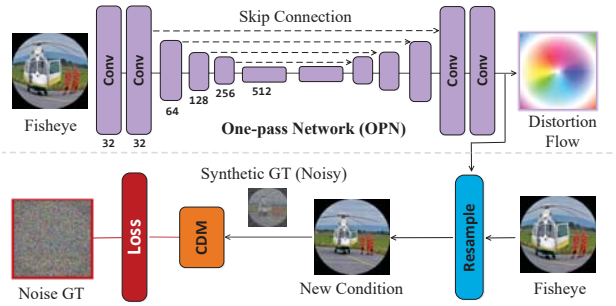

Figure 5. The architecture of OPN and training modality. It generates a new condition for CDM, allowing the gradient to be accurately backpropagated and guide the unsupervised flow prediction.

by simply supervising the predicted noise. Joint optimization is required for the CDM and UDM within our network. Notably, the disparity between the real and synthetic fisheye images remains substantial and has not been eliminated by the pre-trained network. Therefore, the weights between CDM and UDM are not shared. The final loss function is:

$$\mathcal{L} = \mathcal{L}_{syn} + \lambda_r \mathcal{L}_{real} \tag{10}$$

We leverage the tradeoff parameter λ_r to balance the training between modules. Through integrated supervision, our network can achieve end-to-end training.

4.5. Testing strategy

Benefiting from our DDA, we provide two optional test methods (one-pass correction and inference correction), which are a significant improvement over both the classic DDPMs methods [41][42] and existing fisheye correction methods [18][30]. For one-pass correction, we utilize the OPN obtained from our trained DDA to directly predict the distortion flow W of the fisheye image (real or synthetic). We then use W to warp the fisheye image and quickly obtain the correction result. This approach addresses the issue that diffusion models require long-time inference.

For inference correction, we optimize the network using only OPN and CDM to predict the noise. Because CDM and UDM predicts the same noise, but UDM cannot use conditions to guide generation. Therefore, UDM is avoided in testing. We first use the OPN to predict the distortion flow W and pre-correct the synthetic or real fish images. Then we randomly sample a noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ as initial image. The initial image and the pre-corrected result are fed into the CDM to predict the noise for denoising. By repeatedly predicting the noise and recalculating new images, we can obtain high-quality results. This method significantly enhances the subjective visual effect of the images.

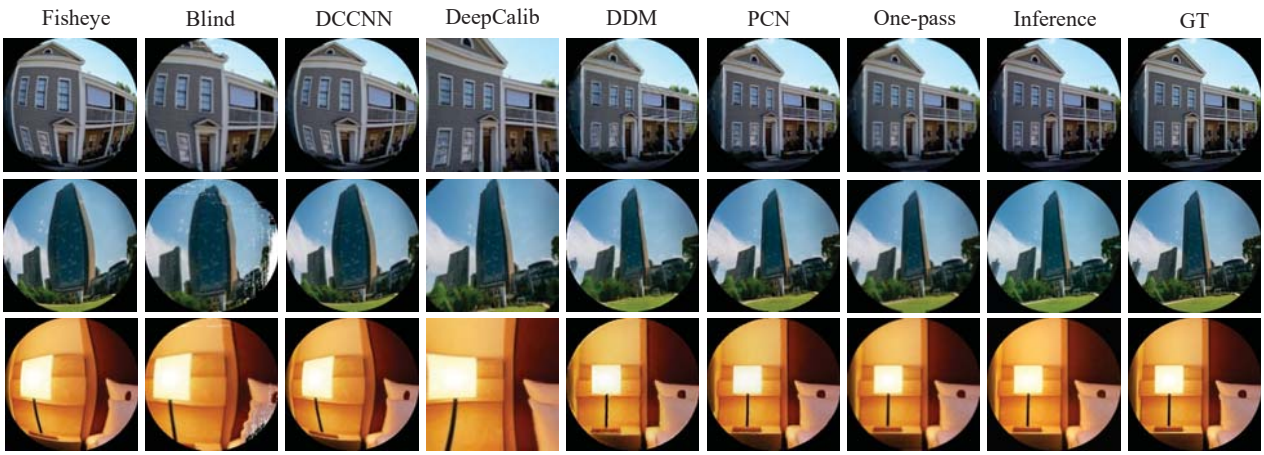

Figure 6. **Subjective comparison results on synthetic images.** We test synthetic fisheye images with random distortion using the state-of-the-art methods (Blind [19], DCCNN [13], DeepCalib [14], DDM [18], PCN [30]) and our methods (One-pass and Inferences).

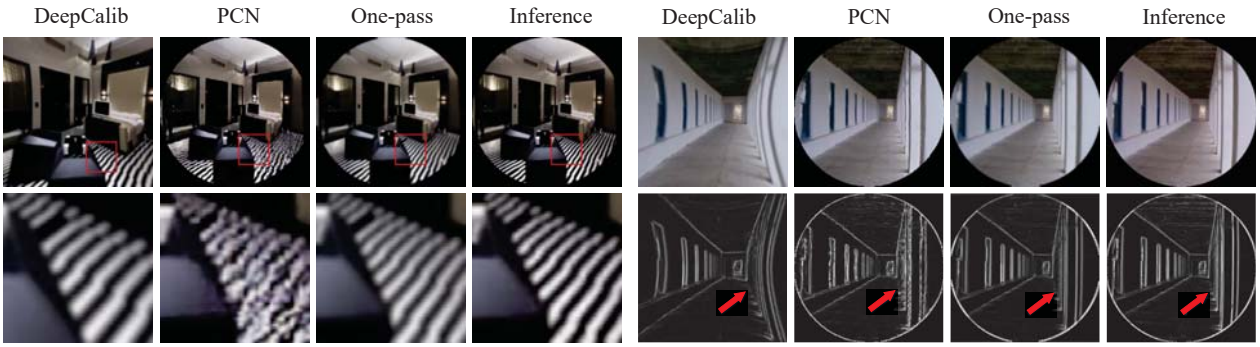

Figure 7. **Additional comparisons on some better performance methods.** We enlarged the local region (marked by red boxes on the left) to compare the image texture. Besides, we highlighted the structural differences (marked by red arrows on the right).

5. Experiments

5.1. Experiment Setting

For simultaneously perceiving the distribution of real and synthetic fisheye images, we need to use them for training. Initially, we refer to previous methods [16][18][30] and utilize a polynomial model with four parameters to generate the synthetic fisheye dataset. Our perspective image dataset is the Places2 dataset [45], which comprises 10 million perspective images. We randomly selected 44K images (40K for training, 4K for testing). We set the values of the four parameters randomly, based on [18][30], to generate distortion for each perspective image. As for real fisheye images, we use the Woodscape dataset [46], which is a popular dataset that contains over 8K real fisheye images of on-road driving. We randomly selected 8K images for training and 200 for testing. To address the problem of quantitative imbalance between real and synthetic fisheye images, we perform data augmentation on the real fisheye images. All images sent to the network are resized to 256×256. For our experiments, we set $\lambda_r = 1.0$ empirically. The initial learning rate and batch size are set to 1e-4 and 2, respectively. Finally, the network is trained for 50 epochs on eight NVIDIA RTX A4000.

5.2. Subjective and Objective Comparison

To evaluate the performance of our method, we retrained several mainstream correction methods using the same dataset. Specifically, we retrained Blind [19], DCCNN [13], DDM [18], PCN [30]. Additionally, we compared our results with DeepCalib [14], which employs a sphere model for correction. However, DeepCalib uses panorama images to synthesize its dataset, which is not available for the Places2 dataset [45]. As a result, we were unable to generate synthetic images using the same method as DeepCalib for direct comparison. Instead, we employed the pre-trained DeepCalib model and cropped its output to maximize resemblance to the ground truth. We visualize the correction results of each method and use common metrics, including PSNR, SSIM, FID [47], MS-SSIM [48], LPIPS-Alex [49],

Table 1. Performance comparison with the state-of-the-art methods.

Comparison		Metrics					
Methods	Type	PSNR ↑	SSIM ↑	MS-SSIM ↑	FID ↓	LPIPS-Alex ↓	LPIPS-Vgg ↓
Blind [19]	Regression	14.7	0.47	0.55	211.3	0.434	0.427
DCCNN [13]	Regression	15.2	0.48	0.37	190.8	0.289	0.345
DeepCalib [14]	Regression	20.8	0.69	0.77	69.7	0.136	0.195
DDM [18]	Generation	24.7	0.80	0.92	79.5	0.142	0.238
PCN [30]	Generation	25.1	0.82	0.92	65.8	0.106	0.165
Ours (one-pass)	Generation	**26.0**	**0.85**	**0.95**	57.8	0.149	0.123
Ours (inference)	Generation	24.6	0.76	0.92	**24.9**	**0.061**	**0.100**

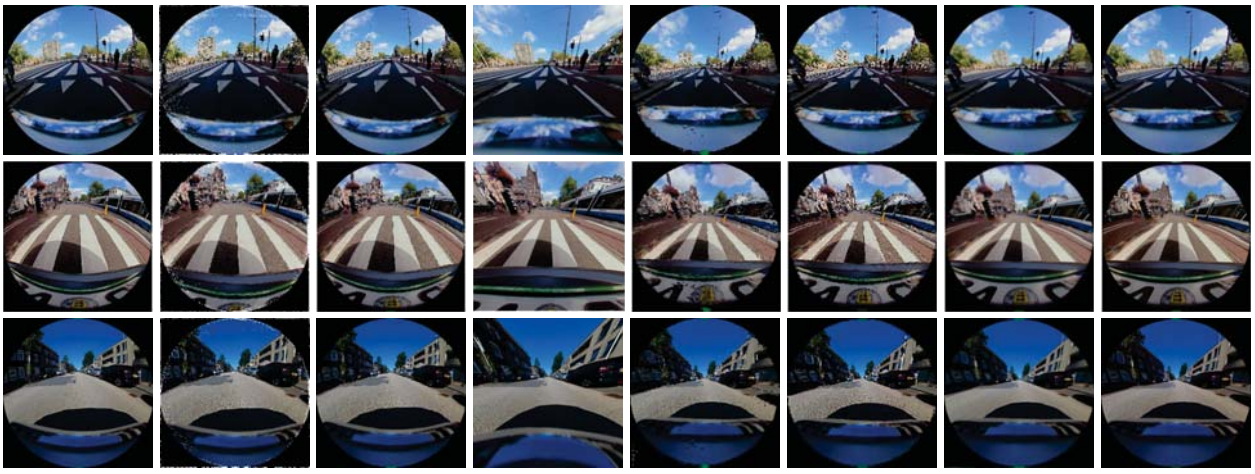

Figure 8. **Visualization results on the real fisheye correction.** We visualize the correction results of the mainstream methods and our methods on the Woodscape dataset [46].

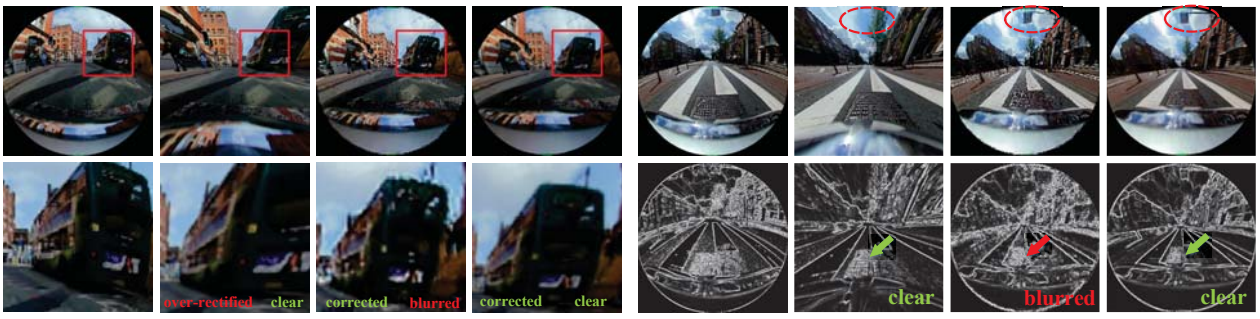

Figure 9. **Detailed comparison on real fisheye results.** We enlarge the local region (indicated by red boxes on the left) and highlight structural differences (indicated by arrows on the right). Additionally, other details require attention (indicated by the red circle on the right). We achieve complete correction with more accurate structure and realistic texture compared to other methods.

LPIPS-Vgg [49], to quantify the objective performance.

The subjective and objective results are demonstrated in Figure 6 and Table 1, respectively. Blind and DCCNN achieve incomplete corrections due to their simplistic model and limited distortion interval. DeepCalib achieves good results by leveraging a more realistic spherical model. However, DDM and PCN show more substantial progress with network improvements, as their recursive correction leads to better subjective and objective results.

By contrast, our method achieves more significan-

Table 2. Performance comparison on different architectures.

Architecture	Syn	Real	One-Pass				
			PSNR	SSIM	MS-SSIM	FID	LPIPS
CDM	✓	✗	–	–	–	–	–
CDM+OPN	✓	✗	25.9	0.85	0.94	**56.4**	**0.146**
DDA	✓	✓	**26.0**	**0.85**	**0.95**	57.8	0.149
			Inference				
CDM	✓	✗	18.6	0.55	0.72	127.9	0.191
CDM+OPN	✓	✗	22.1	0.71	0.84	55.1	0.101
DDA	✓	✓	**24.6**	**0.76**	**0.92**	**24.9**	**0.061**

t progress with a novel dual diffusion architecture, which combines the strengths of GANs and DDPMs. For one-pass correction, our unsupervised OPN outperforms the state-of-the-art, mainly due to the effective backpropagation gradient from CDM. CDM uses noisy data as input, increasing the network's robustness. Additionally, CDM is trained by supervising noise instead of images, enabling easier convergence and more accurate distortion flow prediction in unsupervised OPN. For inference correction, benefits from DDPMs' iterative calculation, the results achieve greater clarity based on accurate structure, as shown in Figure 7. Consequently, our one-pass correction and inference correction yield optimal objective performance in distortion metrics (PSNR, SSIM, MS-SSIM) and perception metrics (FID, LPIPS-Alex, LPIPS-Vgg), respectively.

It is worth noting that even though the distortion metrics of one-pass correction are superior to inference correction, it does not imply that the inference process is futile. In the correction task, PSNR and SSIM cannot effectively evaluate the correction effect since they only reflect the degree of pixel alignment between the corrected image and the ground truth. However, achieving the highest visual similarity (FID and LPIPS) with the ground truth without relying on pixel alignment is also a viable solution for correction.

5.3. Comparison on Real Fisheye Image Correction

To compare the effectiveness of correcting real fisheye images, we visualize the real correction results in Figure 8. We observe that Blind and DCCNN fail to achieve complete correction. DDM and PCN can correct real fisheye images, but the results display obvious artifacts, indicating that models trained on synthetic datasets can only effectively correct synthetic fisheye images. DeepCalib performs well because the sphere model is more consistent with the real fisheye distribution. However, it crops the image boundaries, resulting in a loss of information. In contrast, both our one-pass and inference correction achieve outstanding results by preserving all boundaries, generating accurate structure and clear content. To enhance the clarity of observations, we conducted a specific comparison, which is depicted in Figure 9. It further proves that our real correction effect outperforms other methods.

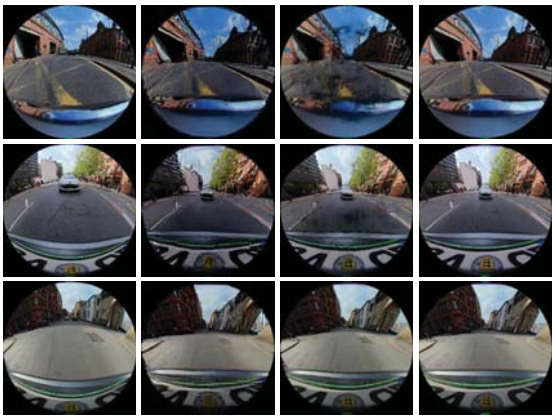

Figure 10. Visualization results of different architectures. The results obtained by DDA are the clearest and most accurate.

5.4. Ablation Study

The conditional diffusion module (CDM), the one-pass network (OPN), and the unconditional diffusion module (UDM) are the major components of our dual diffusion architecture (DDA). To verify the effectiveness of each module, we start with the original CDM, then add each module and evaluate the improvement. The results are presented in Figure 10 and Table 2. First, we only use CDM with the original synthetic fisheye as the condition to correct our fisheye image. It can be seen that the performance is poor, and the correction results exhibit significant blurring. Subsequently, we increase OPN (CDM + OPN). The artifacts of the image are eliminated to a certain extent, and the quantitative results of inference correction for synthetic images have significant improvement. It indicates that it is effective to replace the original image with more reasonable conditions generated by OPN. However, this scheme generates obvious artifacts in the correction of real fisheye images. Thus, we further increase the UDM and introduce unlabeled real fisheye images for training. With this approach, the network achieved satisfactory performance for both synthetic and real fisheye images. This result confirms that even in the absence of corresponding labels, UDM can learn the distortion rules of real fisheye images, thus enhancing the ability of CDM to correct both synthetic and real fisheye images.

6. Conclusion

In this paper, we propose a novel dual diffusion architecture (DDA) that addresses the low applicability of the synthetic fisheye image model in real fisheye correction. Our DDA combines both conditional and unconditional diffusion modules and leverages both paired synthetic fisheye

images and unlabeled real fisheye images for training. By progressively adding noise to the two source images, we can transform their inconsistent distributions into a consistent noise distribution, enabling the network to improve the correction performance on real fisheye images without corresponding labels. Different from previous diffusion methods, we introduce a one-pass network (OPN) in the conditional diffusion module to provide new reasonable guidance. OPN achieves unsupervised training and provides a fast correction scheme. Experiments demonstrate our one-pass and inference results outperform all comparisons.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62172032, 62120106009).

References

[1] R. Rajesh Sharma and A. Sungheetha. An efficient dimension reduction based fusion of cnn and svm model for detection of abnormal incident in video surveillance. *Journal of Social and Clinical Psychology*, 3:55–69, 2021. 1

[2] K. Zhu and T. Zhang. Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Science & Technology*, 26:674–691, 2021. 1

[3] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 1

[4] J. Redmon, S. Kumar Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016. 1

[5] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:640–651, 2017. 1, 2

[6] A. A. Mohamed, K. Qian, M. Elhoseiny, and C. G. Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14412–14420, 2020. 1

[7] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2301–2306. IEEE, 2004. 1

[8] A. Geiger, F. Moosmann, Ö. Car, and B. Schuster. Automatic camera and range sensor calibration using a single shot. In *2012 IEEE international conference on robotics and automation*, pages 3936–3943. IEEE, 2012. 1

[9] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. *ICCV*, 1:666–673 vol.1, 1999. 1

[10] V. C. Usenko, N. Demmel, and D. Cremers. The double sphere camera model. *2018 International Conference on 3D Vision (3DV)*, pages 552–560, 2018. 1

[11] D. Dansereau, O. Pizarro, and S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1027–1034, 2013. 1

[12] R. Melo, M. Antunes, J. Pedro Barreto, G. Falcão Paiva Fernandes, and N. Gonalves. Unsupervised intrinsic calibration from a single frame using a "plumb-line" approach. *ICCV*, pages 537–544, 2013. 1, 2

[13] J. Rong, S. Huang, Z. Shang, and X. Ying. Radial lens distortion correction using convolutional neural networks trained with synthesized images. In *ACCV*, 2016. 1, 2, 6, 7

[14] O. Bogdan, V. Eckstein, F. Rameau, and J. Bazin. Deepcalib: a deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In *CVMP*, 2018. 1, 6, 7

[15] X.Yin, X. Wang, J. Yu, M. Zhang, P. Fua, and D. Tao. Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification. In *ECCV*, pages 475–490, 2018. 1, 2

[16] Z. Xue, N. Xue, GS. Xia, and W. Shen. Learning to calibrate straight lines for fisheye image rectification. *CVPR*, pages 1643–1651, 2019. 1, 2, 6

[17] K. Liao, C. Lin, Y. Zhao, and M. Gabbouj. DR-GAN: Automatic radial distortion rectification using conditional GAN in real-time. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019. 1, 2, 3

[18] K. Liao, C. Lin, Y. Zhao, and M. Xu. Model-free distortion rectification framework bridged by distortion distribution map. *IEEE Transactions on Image Processing*, 29:3707–3718, 2020. 1, 3, 5, 6, 7

[19] X. Li, B. Zhang, Pedro V. Sander, and J. Liao. Blind geometric distortion correction on images through deep learning. In *CVPR*, pages 4855–4864, 2019. 1, 6, 7

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *CoRR*, abs/2006.11239, 2020. 1, 3

[21] A. Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In Marina Meila and Tong Zhang, editors, *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8162–8171. PMLR, 18–24 Jul 2021. 1, 3

[22] JY. Zhu, T. Park, P. Isola, and AA. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2, 5

[23] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:318–327, 2020. 2

[24] S. Zhang, H. Yao, X. Sun, and X. Lu. Sparse coding based visual tracking: Review and experimental comparison. *Pattern Recognition*, 46:1772–1788, 2013. 2

[25] X. Li, C. Ma, B. Wu, Z. He, and M. Yang. Target-aware deep tracking. *CVPR*, pages 1369–1378, 2019. 2

[26] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 2

[27] C. Mei and P. Rives. Single view point omnidirectional camera calibration from planar grids. *IEEE International Conference on Robotics and Automation*, pages 3945–3950, 2007. 2

[28] F. Bukhari and M. N. Dailey. Automatic radial distortion estimation from a single image. *Journal of Mathematical Imaging & Vision*, 45(1):31–45, 2013. 2, 3

[29] M. Alemánflores, L. Álvarez, L. Gómez, and D.SantanaCedrés. Automatic lens distortion correction using one-parameter division models. *IPOL*, 4:327–343, 2014. 2, 3

[30] S. Yang, C. Lin, K. Liao, C. Zhang, and Y. Zhao. Progressively complementary network for fisheye image rectification using appearance flow. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6344–6353, 2021. 2, 3, 5, 6, 7

[31] K. Zhao, C. Lin, K. Liao, S. Yang, and Y. Zhao. Revisiting radial distortion rectification in polar-coordinates: A new and efficient learning perspective. *IEEE Transactions on Circuits and Systems for Video Technology*, 2021. 2, 3

[32] A. Basu and S. Licardie. Alternative models for fish-eye lenses. *Pattern Recognition Letters*, 16:433–441, 1995. 3

[33] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis R. Bach and David M. Blei, editors, *International Conference on Machine Learning, ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2256–2265. JMLR.org, 2015. 3

[34] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations, ICLR*. OpenReview.net, 2021. 3

[35] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations, ICLR*, 2016. 3

[36] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations, ICLR 2014*, 2014. 3

[37] D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Neural Information Processing Systems*, pages 10236–10245, 2018. 3

[38] X. Yang, S. Shih, Y. Fu, X. Zhao, and S. Ji. Your vit is secretly a hybrid discriminative-generative diffusion model. *CoRR*, abs/2208.07791, 2022. 3

[39] W. Wang, J. Bao, W. Zhou, D. Chen, D. Chen, L. Yuan, and H. Li. Semantic image synthesis via diffusion models. *CoRR*, abs/2207.00050, 2022. 3

[40] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, N. Dehak, and W. Chan. Wavegrad 2: Iterative refinement for text-to-speech synthesis. In Hynek Hermansky, Honza Cernocký, Lukás Burget, Lori Lamel, Odette Scharenborg, and Petr Motlícek, editors, *Annual Conference of the International Speech Communication Association*, pages 3765–3769. ISCA, 2021. 3

[41] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2022. 3, 5

[42] J. Whang, M. Delbracio, H. Talebi, C. Saharia, A. G. Dimakis, and P. Milanfar. Deblurring via stochastic refinement. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16272–16282, 2022. 3, 5

[43] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan. Wavegrad: Estimating gradients for waveform generation. In *International Conference on Learning Representations, ICLR*. OpenReview.net, 2021. 3

[44] J. Duchon. Splines minimizing rotation-invariant semi-norms in sobolev spaces. In *Constructive Theory of Functions of Several Variables: Proceedings of a Conference Held at Oberwolfach April 25–May 1, 1976*, pages 85–100. Springer, 1977. 4

[45] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1452–1464, 2018. 6

[46] S.Yogamani, C.Witt, H.Rashed, S.Nayak, S.Mansoor, P.Varley, X.Perrotton, D.Odea, P.Perez, C.Hughes, J.Horgan, G.Sistu, S.Chennupati, M.Uricar, S.Milz, M.Simon, and K.Amende. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9308–9318, 2019. 6, 7

[47] H. Martin, R. Hubert, U. Thomas, N. Bernhard, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. 6

[48] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multi-scale structural similarity for image quality assessment. In *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1398–1402, 2003. 6

[49] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Conference on Computer Vision and Pattern Recognition, CVPR*, pages 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. 6, 7