HYPERNETWORK-BASED ADAPTIVE IMAGE RESTORATION

Shai Aharon *Gil Ben-Artzi*

Department of Computer Science
Ariel University, Israel

ABSTRACT

Adaptive image restoration models can restore images with different degradation levels at inference time without the need to retrain the model. We present an approach that is highly accurate and allows a significant reduction in the number of parameters. In contrast to existing methods, our approach can restore images using a single fixed-size model, regardless of the number of degradation levels. On popular datasets, our approach yields state-of-the-art results in terms of size and accuracy for a variety of image restoration tasks, including denoising, deJPEG, and super-resolution.

Index Terms— Image Denoising, DeJPEG, super-resolution

1. INTRODUCTION

The common approach in deep learning for image restoration tasks is to train the model in a supervised manner, optimizing the model for only a single degradation level. The exact degradation level of the degraded image is not known a priori, and treating all degradation levels the same lowers restoration quality. Recently, adaptive image restoration methods have gained popularity as an alternative. They enable image- and user-specific adaptation for all degradation levels with a single model without the need for retraining or deploying multiple models. At runtime, the user can adjust the restoration effect in order to generate a variety of output images and select one according to his preferences.

Various approaches have been presented for adaptive models. State-of-the-art approaches [1, 2, 3, 4] reduce the needed number of parameters with respect to multiple independent models by training the network to restore two degradation levels that span the desired range and interpolate the weights for any other degradation level within the range. However, to support a wide range of levels it is necessary to use several models or significantly increase the size of the basic model [5].

In this work, we present an approach that allows a standard single restoration model to achieve very high accuracy across a wide range of degradation levels without having to add any extra parameters. We introduce a hypernetwork that learns to generate the filter weights of an image restoration network conditionally based on the required restoration level given as an input parameter. As part of the training process, our hypernetwork is optimized with multiple main networks to simultaneously restore images with a variety of degradation levels. Based on the input degradation level, our hypernetwork generates a single network with the most accurate filter weights at runtime.

Contribution. Our architecture can restore images of different degrees of degradation with 26%-56% of the parameters and higher accuracy than existing adaptive image restoration methods.

2. PRIOR WORK

Recently, there has been a growing interest in constructing networks that can be continuously tuned at inference time. These can broadly be categorized into two categories, models which allow tuning different objectives at inference time and models which allow different restoration levels of the same objective, where our approach falls into the latter category. The typical approach is to train two related networks on different objectives and apply interpolation between their weights. The networks can be either the same or with additional tuning blocks. Dynamic-Net [3] adds specialized blocks directly after the convolution layers, which are optimized during the training for the additional objective. CFSNet [4] used branches, each based on a different objective. AdaFM [1] added modulation filters after each convolution layer. DNI [2] train the same network architecture on different objective and interpolates all the parameters. Son [6] extended the approach of [1] with an FTN module allowing better non-linear interpolation. [7] generates kernels for the super-resolution task. They employ an off-the-shelf SR network [8] as a backbone which is $\times 10$ bigger than our network, and their proposed solution is applicable only to the super-resolution task. [9] proposed solving the image restoration task as a multi-task problem. Their network, however, is specialized for restoring only a limited number of degradation levels, each of which must be trained individually. In our approach, the model is trained on a small set of degradation levels and can continuously restore any other level inside and even outside this range.

Learning to learn, or meta learner, uses meta networks to generate weights for the main network for various tasks [10].

Hypernetwork, introduced in [11], uses a small network with a reduced number of parameters to generate the weights for a larger target network. It often uses weight sharing across layers, while providing accurate results. [12] presented an image restoration hypernetwork with a single main network. In our approach, we employ a hypernetwork to generate the weights of kernels in multiple target networks simultaneously.

3. METHOD

3.1. Formulation

Our model consists of a hypernetwork h and main restoration networks n_i. The weights of our hypernetwork, θ^h, are learned during the training process and fixed during inference time. The input to hypernetwork h is a degradation level $c_i \in \mathbb{R}$ and the output is θ^{n_i}, the kernels' weights for the main restoration networks n_i, $\theta^{n_i} = h(c_i; \theta^h)$. The input for each main network n_i is a degraded image $\mathbf{I}^{c_i} \in \mathbb{R}^{3\times H\times W}$ with a degradation level c_i, H, W are the height and width of the image and the output is the restored image with the same dimensions, $n_i(\mathbf{I}^{c_i}; \theta^{n_i})$. The goal is to learn θ^h from $\mathbf{I}^{c_i}, c_i$ so that h can generate the optimal weights for the corresponding restoration network θ^{n_i} in order to restore the degraded image. The optimization problem associated with our model is formulated as:

$$\arg\min_{\theta^h} \sum_{i=1}^{k} \mathbb{E}_{\mathbf{I}^{c_i},\mathbf{I}} \left[\mathcal{L}\big(n_i(\mathbf{I}^{c_i}; h(c_i; \theta^h)), \mathbf{I}\big) \right] \tag{1}$$

where k is the number of main networks, $\mathcal{L} : \mathbb{R}^{3\times H\times W} \times \mathbb{R}^{3\times H\times W} \rightarrow \mathbb{R}_+ \cup \{0\}$ is our restoration loss for each main network. We train our hypernetwork, h, by simultaneously generating the weights for all the k main networks (and degradation levels), and optimizing them together. We demonstrate that this enables our hypernetwork to generate optimal weights for all other degradation levels within the continuous range that are not included in the k levels.

3.2. The Network

Parametrization of Main Network. The hypernetwork h consists of l meta blocks, where l is the number of kernels in the main network's residual blocks. Each meta block is a fully connected layer constructed out of weights and biases, $\mathbf{w}^j, \mathbf{b}^j \in \mathbb{R}^{(C_{out}\times C_{in}\times K\times K)\times 1}$, where C_{in} and C_{out} are the number of input and output channels, respectively, and $K \times K$ is the kernel's size. The j^{th} kernel of main network n_i is $\mathbf{k}_i^j$, of the same dimensions as the meta block. The parameters of the hypernetwork and each main network are $\theta^h = \{(\mathbf{w}^j, \mathbf{b}^j)\}_{j=1}^{l}$, $\theta^{n_i} = \{(\mathbf{k}_i^j,\}_{j=1}^{l}$.

We parameterize the kernels of the main network as a linear combination of our hypernetwork's weights and biases with the degradation level as a scaling parameter. Thus, our

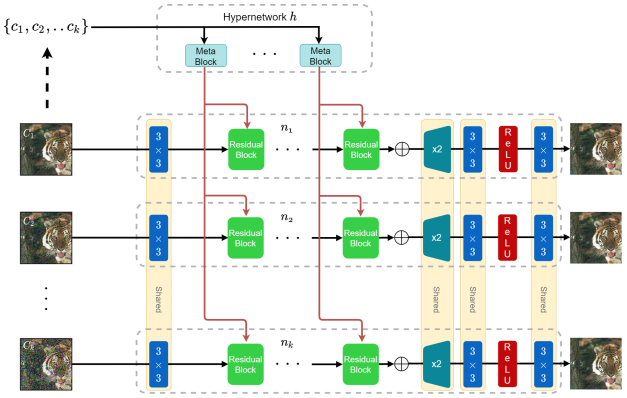

Fig. 1: The training framework. k main networks are generated by the hypernetwork using the same kernels. Each is fed with a corresponding degraded image, and the losses are summed and back-propagated to the shared and hypernetwork weights. At runtime, a single main network is generated based on a single input parameter.

hypernetwork h generates the weights of the j^{th} kernel of main network n_i for degradation level c_i by:

$$\mathbf{k}_i^j = c_i \mathbf{w}^j + \mathbf{b}^j. \tag{2}$$

This allows a highly efficient representation as we only need to store the weights and biases of the hypernetwork in order to generate the corresponding kernels for each possible restoration network given the degradation level. We demonstrate that, despite the compact representation, our model is highly accurate. Note that, unlike hypernetworks [11], our method assigns each meta block to generate weights for a specific main network layer with one common input scalar. Due to the bias term, the output convolutional kernels are not identical throughout the various main networks up to the input scalar.

Training. During training, the hypernetwork generates multiple main networks, each main network n_i is optimized to restore a degraded image with a corresponding degradation level c_i. The number of main networks (k) is fixed during the training process. The main network is a standard image restoration network [13]. It consists of a downsampling layer using convolution with a stride of 2, 16 residual [14] blocks and upsampling layers using pixelshuffle [15] and a skip-connection over the residual blocks. The weights of each main network are the weights of the residual block's kernels generated by the hypernetwork (Fig. 1, green background) and the weights of the head and tail of the network which are shared among all the main networks (Fig. 1, yellow background).

Each image in the training set $\mathcal{D} = \{\mathbf{I_1}, \mathbf{I_2}, \ldots, \mathbf{I_n}\}$ is degraded with k degradation levels $\{c_1, c_2, \ldots, c_k\}$ and fed into the corresponding main network $\{n_1, n_2, \ldots, n_k\}$. Each

Table 1: Results for DeJPEG artifacts removal task.

	$PSNR$					
	10	30	50	70	80	Mean
Baseline	28.82	32.57	34.40	36.40	38.14	34.06
Ours	28.81	32.56	34.39	36.38	38.09	34.04

	$SSIM$				
	10	30	50	70	80
Baseline	0.82	0.91	0.94	0.96	0.97
Ours	0.82	0.91	0.94	0.96	0.97

Table 2: Results for image denoising.

	$PSNR$					
	5	25	45	65	90	Mean
Baseline	40.48	31.42	28.64	27.06	25.73	30.66
Ours	40.39	31.40	28.51	27.06	25.73	30.61

	$SSIM$				
	5	25	45	65	85
Baseline	0.98	0.89	0.81	0.76	0.71
Ours	0.98	0.89	0.81	0.76	0.71

main network is generated according to the degradation level c_i and meta blocks. Our goal is to optimize the overall restoration accuracy under the different degradation levels. Therefore, no degradation level is privileged and our total loss is the unweighted sum of individual $\mathcal{L}_1$ losses. Since the aforementioned weight generation operations are completely differentiable, the parameters in our hypernetwork h are optimized simultaneously following the chain rule. The $L1$ loss is used throughout all the experiments. The training process is illustrated in Fig. 1.

Inference. Given a degraded image and an input degradation level c_i, we employ the learned weights of the hypernetwork θ^h to generate the weights of a restoration network θ^{n_i}. Each meta block generates the weights according to Eq. 2. A simple user interface enables the user to interact with the system in real-time, selecting the input value and, as a result, the desired restoration outcome. The restoration network generation is efficient due to the multiplication of the same single scalar for all the residual blocks' kernels of the main network.

4. EXPERIMENTS

We demonstrate that our approach with a single network is equivalent to deploying multiple networks (5-11), each is optimized for a different degradation level. We also achieve the same or higher accuracy with a significant reduction in the number of parameters with respect to state-of-the-art adaptive models.

Table 3: Results for super-resolution task.

	$PSNR$					
	2	3	4	5	6	Mean
Baseline	36.95	29.86	29.54	25.67	25.06	29.41
Ours	36.71	29.77	29.48	25.63	24.92	29.30

	$SSIM$				
	2	3	4	5	6
Baseline	0.94	0.84	0.84	0.74	0.71
Ours	0.94	0.84	0.83	0.74	0.71

We evaluate our approach with the following tasks - denoising, DeJPEG and super-resolution, based on popular benchmarks. The DIV2K[16] dataset was used to train the models for all tasks. For evaluation, we use the CBSD68 dataset [17] for denoising, the LIVE1 [18] for DeJPEG and the Set5 [19] for super-resolution.

4.1. Comparison with optimal accuracy

We deploy five (super-resolution), eight (DeJPEG) and eleven (denoising) dedicated restoration models, each specifically trained to restore a single degradation level. The basic restoration network, both ours and for each of the dedicated models, includes 16 residual blocks and is based on [13].

For DeJPEG, we evaluate our model at eight different compression levels. Table 1 shows our results with respect to optimal accuracy obtained by training the independent models to restore each compression level. In our approach, we achieve very high PSNR and SSIM, with a negligible PSNR distance from optimal accuracy. For denoising, we evaluate our model with respect to all noise levels from 5 to 110 with intervals of 5. Table 2 shows our results for the denoising task. Our model achieves restoration accuracy equivalent to that of eleven dedicated models using only a single network. For super-resolution, we evaluate our model using five upscaling factors, $\times 2, \times 3, \times 4, \times 5, \times 6$. Table 3 shows our results. Similar to the other tasks, our method achieves state-of-the-art accuracy with only one basic restoration network.

4.2. Comparison with small size adaptive models

We compare our accuracy with respect to existing state-of-the-art smaller adaptive models, AdaFM [1] and CFSNet [4]. Despite not being optimal, these models achieve very high accuracy with a small number of parameters with respect to multiple networks.

Our evaluation is the same as before. Fig. 2 shows our results for the DeJPEG task. Our model includes only 1 resblock with 0.37*1e6 parameters, AdaFM has 1.41*1e6 parameters, and CFSNet includes 1.73*1e6 parameters. Our model achieves slightly better accuracy with only 22%-26% of the parameters. For denoising (Fig. 3) our model achieves slightly better accuracy than other methods with a significant reduction of 44%-54% in the size of the network. For super-resolution (Fig. 4), our model yields comparable accuracy to AdaFM with only 36% of the parameters and better accuracy than CFSNet with 29% of the parameters.

Overall, with respect to adaptive models that aim to reduce the number of parameters, our approach presents a significant saving in the number of parameters of up to 78% with comparable accuracy.

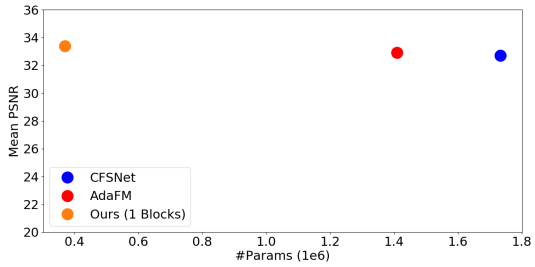

Fig. 2: Results for DeJPEG

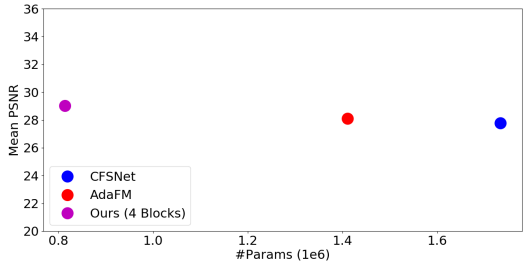

Fig. 3: Results for denoising

4.3. Comparison with large size adaptive models

We compare our results to those of the large size adaptive model CResMD [5]. CResMD includes 32 residual blocks, 16 more than in our network. It has been shown that with these additional layers, CResMD outperforms other smaller size adaptive methods. Table 4 presents the mean PSNR for both our approach and CResMD. Overall, our approach achieves better accuracy.

4.4. Input parameter tuning

In the following, we explore the ability to accurately set the input parameters in our approach.

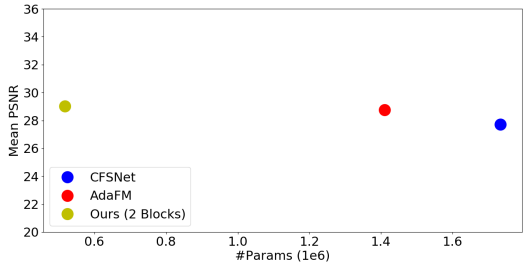

Fig. 4: Results for super-resolution

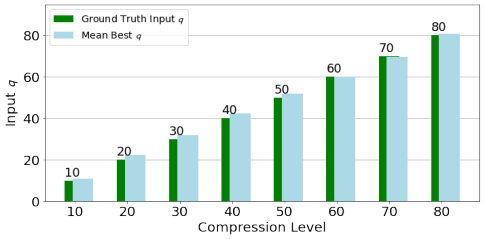

Fig. 5: Parameter accuracy for DeJPEG. The dark green bars represent the ground truth levels while the light green bars represent the best input parameter that achieves the highest restoration accuracy.

Table 4: Mean PSNR

	DeJPEG	denoise	super-resolution
CResMD	32.9	28.80	28.71
Ours	34.04	30.61	29.30

Blind Setting. We train a simple CNN (five convolutional layers and three fully connected layers) to estimate the degradation level of a noisy image. Based on our trained network, we can estimate the input degradation level and set the input scalar accordingly. On average, the degradation level estimation network achieves an accuracy of 98.23%. Overall, the estimation of noise level results in accurate restoration and can be advantageous in cases where the actual degradation level is unknown.

Manual Setting. We experiment with the ability of the hypernetwork at inference time to generate, as trained, the optimal weights for the corresponding input degradation level. For each image in the test set, we degrade the image with a specific degradation level, e.g. $\sigma = 15$ for the denoising task. For the degraded image, we measure the best input parameter that yields the network with the highest restoration accuracy in terms of PSNR and compare the value of the input parameter with the ground truth level. Figure 5 presents the results, showing a negligible difference between the optimal and actual input.

5. CONCLUSION

We presented an efficient approach that can restore images with multiple degradation levels at runtime without the need to retrain the model. In order to increase efficiency, we propose using a hypernetwork-based model and simultaneously training several main networks. We demonstrate that our method achieves state-of-the-art accuracy while significantly reducing network size.

6. REFERENCES

[1] Jingwen He, Chao Dong, and Yu Qiao, "Modulating image restoration with continual levels via adaptive feature modification layers," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[2] Xintao Wang, Ke Yu, Chao Dong, Xiaoou Tang, and Chen Change Loy, "Deep network interpolation for continuous imagery effect transition," 2018.

[3] Alon Shoshan, Roey Mechrez, and Lihi Zelnik-Manor, "Dynamic-net: Tuning the objective without re-training for synthesis tasks," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[4] Wei Wang, Ruiming Guo, Yapeng Tian, and Wenming Yang, "Cfsnet: Toward a controllable feature space for image restoration," 2019.

[5] Jingwen He, Chao Dong, and Yu Qiao, "Interactive multi-dimension modulation with dynamic controllable residual learning for image restoration," *arXiv preprint arXiv:1912.05293*, 2019.

[6] Hyeongmin Lee, Taeoh Kim, Hanbin Son, Sangwook Baek, Minsu Cheon, and Sangyoun Lee, "Smoother network tuning and interpolation for continuous-level image processing," *arXiv preprint arXiv:2010.02270*, 2020.

[7] Xuecai Hu, Haoyuan Mu, Xiangyu Zhang, Zilei Wang, Tieniu Tan, and Jian Sun, "Meta-sr: A magnification-arbitrary network for super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1575–1584.

[8] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu, "Residual dense network for image restoration," 2018.

[9] Wei Jiang, Wei Wang, Shan Liu, and Songnan Li, "Png: Micro-structured prune-and-grow networks for flexible image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 756–765.

[10] Benlin Liu, Yongming Rao, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh, "Metadistiller: Network self-boosting via meta-learned top-down distillation," in *European Conference on Computer Vision*. Springer, 2020, pp. 694–709.

[11] David Ha, Andrew Dai, and Quoc V. Le, "Hypernetworks," 2016.

[12] Qingnan Fan, Dongdong Chen, Lu Yuan, Gang Hua, Nenghai Yu, and Baoquan Chen, "Decouple learning for parameterized image operators," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 442–458.

[13] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang, "Compression artifacts reduction by a deep convolutional network," 2015.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[15] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[16] Eirikur Agustsson and Radu Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[17] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. IEEE, 2001, vol. 2, pp. 416–423.

[18] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[19] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie line Alberi Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proceedings of the British Machine Vision Conference*. 2012, pp. 135.1–135.10, BMVA Press.