# **Available noise reduction models**

Please use the following models in project KAIR (integrated and updated): <u>https://github.com/</u> <u>cszn/KAIR?tab=readme-ov-file</u>

- 1. DnCNN: <u>https://github.com/cszn/DnCNN?tab=readme-ov-file</u>
- 2. FFDNet: https://github.com/cszn/FFDNet?tab=readme-ov-file
- 3. DPIR: https://github.com/cszn/DPIR
- 4. IRCNN Noise reducer: https://github.com/cszn/KAIR?tab=readme-ov-file

## Instructions

### -.DnCnn(Using Gaussian denoising)

Environment configuration, for using DnCNN, use python version 3.8.10 portable

• Install dependencies (please execute in terminal). Please don't worry. The installation instructions are executed in one line. The first instruction installs multiple dependencies. After the installation is completed, wait a moment. Pycharm takes time to load.

```
1 pip install -r requirement.txt
```

- 2 pip install requests
- 3 pip install matplotlib



When executing in the terminal, the path will be displayed when each installation instruction is completed. If it is not displayed, it means it is loading.

1. Open the project KAIR, open the model\_zoo folder, and view the documentation inside

	🍦 main_te	st_dncnn.py	🕙 01.png	🗬 main_test_ircnn_denoiser.py	M↓ README.md	×	~ ≣ ≣0 ⊠
		mố	model_zoo				
l			download all the r	nodels:			
l		dov	python main_dow	nload_pretrained_models.py -	-models "all"	model_dir "model_zo	o''
l		nutl		u modelc:			
		 		n mouels.			
		dov	python main_dow	nload_pretrained_models.py -	-models "DnCNN	"model_dir "model_z	00"
		pytł	download SRMD	models:			

- 2. You can execute the code above in the terminal to download the model required by DnCNN, or download it from the download URL (model\_zoo folder, the download URL is provided at the bottom of the documentation inside, and all files starting with dncnn are included). The downloaded model files are placed in the model\_zoo folder
- 3. Set the pictures that need to be denoised: Open the folder under the KAIR project, then open set12, and place the pictures that need to be denoised inside.

,	
;	> 🗈 scripts
\ \	testsets
	> 🛅 real_faces
	> 🛅 set5
	> 🗋 set12
	M↓ README.md
;	> 🗋 trainsets
;	> 🗅 utils
	Ӛ main challenge sr.py

4. Under the KAIR project, find main\_test\_dncnn.py, click on this file, scroll down to the bottom of the page, see the main method, click to run



5. The denoised image results are placed in the results folder under the KAIR project. The first half of the generated folder (set12) represents the data source, and the second half (dnncn\_25) represents the model used.

<ul> <li>results</li> </ul>
🔁 01.png
🔁 02.png
🔁 03.png
🔁 04.png
🔁 05.png
🔁 06.png
🔁 07.png
🔁 08.png
🔁 09.png
🔁 10.png
🔁 11.png
🔁 12.png
🔀 6666.png
aset12_dncnn_25
> 🗋 retinaface
> 🗅 scripts

6. The generated log file displays key information about the image data used..

- 1. **Average PSNR/SSIM(RGB)**: This means that what is recorded next is the average PSNR and SSIM values based on the RGB image. PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index) are common indicators for image quality evaluation.
- 2. **set12\_dncnn\_25**: This means that the evaluation is based on the **set12** dataset and uses the **DnCNN** algorithm with a noise level of 25 (usually meaning Gaussian noise with a standard deviation of 25). **DnCNN** is a deep convolutional neural network commonly used for image denoising tasks.
- 3. **PSNR: 31.04 dB**: This is the average Peak Signal-to-Noise Ratio value. PSNR is an important indicator of image quality after denoising. The higher the value, the better the image quality. 31.04 dB indicates that the average quality of the denoised image is relatively high.
- 4. **SSIM: 0.8496**: This is the average Structural Similarity Index value. SSIM is used to measure the similarity of two images in terms of structure, brightness and contrast. The value ranges from 0 to 1. The closer to 1, the more similar the image structure is. 0.8496 indicates that the structural similarity between the denoised image and the original image is high.



补充, The model file can be changed. For example, dncnn\_25 is a model file trained for average noise 25. Can be replaced with 3, 15, 50. **dncnn\_gray\_blind\_blind** removes gray image noise, **dncnn\_color\_blind** removes color image noise

• Enter in the terminal. This allows you to use another model dncnn\_50

```
1 python main_test_dncnn.py --model_name dncnn_50
```

#### **\_**.FFDET

Fast and flexible denoising convolutional neural network, providing a fast and flexible solution for CNN-based image denoising.

Environment configuration, for using FFDET, it is feasible to use python version 3.8.10.

- 1. Same as above
- 2. Same as above, Just download two models ffdnet\_color.pth and ffdnet\_gray.pth, the former is for color and the latter is for grayscale. A single FFDET model can target the noise of [0, 75].
- 3. Set the pictures that need to be denoised: Open the folder under the KAIR project, create the file bsd68, and place the pictures that need to be denoised in it.

testsets	
🗸 🖿 bsd68	
🗟 05.png	
🔁 07.png	
🔁 09.png	
🗟 6666.png	

4. Under the KAIR project, find main\_test\_ffdnet.py, click on this file, scroll down to the bottom of the page, see the main method, click to run (grayscale is used by default), you can use the code below to switch (console execution)

1 python main\_test\_ffdnet.py --model\_name ffdnet\_color.pth

5. The result output is as follows

results	Log format not recognized Configure log formats Hide				
✓ ☐ bsd68_ffdnet_gray	1 24-05-14 14:49:09.082 : Model path: model zoo\ffdnet gray.pth				
	2 29-05-14 14:49:09.082 : model_name:ffdnet_gray, model sigma:15, image sigma:15				
🕙 09.png	3 24-05-14 14:49:09.083 : testsets\bsd68				
🖾 6666.png	4 24-05-14 14:49:09.215 : 05.png - PSNR: 32.92 dB; SSIM: 0.9491.				
🝰 bsd68_ffdnet_gray.log	5 24-05-14 14:49:09 322 · 07 ppg - PSNR· 31 79 dB· SSIM· 0 9045				
> 🗅 set12_dncnn_25	6 24 05 14 14:49:09 717 : 09 ppg PSNR: 22 48 dB; SSIM: 0.9198				
> 🗅 retinaface	24-03-14 14.49.09.717 . 09.phg - FSIM. 52.40 db, 531W. 0.9190.				
> 🗅 scripts	7 24-05-14 14:49:09.886 : 6666.png - PSNR: 39.66 dB; SSIM: 0.9731.				
	8 24-05-14 14:49:09.889 Average PSNR/SSIM(RGB) - bsd68_ffdnet_gray - PSNI	R: 34.21 dB; SSIM: 0.9366			
> 🛅 trainsets					
> Do utile					

#### <u></u>**Ξ**.DPIR

Go to the project address to research and use. The other three can be used directly in KAIR.

#### 四.IRCNN

Environment configuration, it is feasible to use python version 3.8.10.

Dependency version changes

• To lower the numpy version, first uninstall the associated dependencies, then install the 1.19.5 version of numpy, and finally install the uninstalled dependencies.

```
1 pip uninstall matplotlib
2 pip uninstall scikit-image
3 pip uninstall scipy
4 pip install numpy==1.19.5
5 5
6 pip install matplotlib==3.3.3
7 pip install scikit-image==0.17.2
8 pip install scipy==1.5.4
```

- 1. Same as above
- 2. Same as above, Just download two models ircnn\_gray.pth and ircnn\_color.pth, the former is for grayscale and the latter is for color.
- 3. Set the pictures that need to be denoised: Open the folder under the KAIR project, then open set12, and place the pictures that need to be denoised in it.
- 4. Under the KAIR project, find main\_test\_ircnn\_denoiser.py, click on this file, scroll down to the bottom of the page, see the main method, click to run (grayscale is used by default), you can use the code below to switch (console execution)

1 python main\_test\_ircnn\_denoiser.py --model\_name ircnn\_color.pth

5.Output results

✓ □ results	24-05-14 15:24:47.921	: 03.png - PSNR: 27.33 dB; SSIM:	0.8044.	
> 🗅 bsd68_ffdnet_gray	24-05-14 15:24:48 100	): 04 png - PSNR: 25 48 dB: SSIM:	0 7675	
> 🗅 set12_dncnn_25	24 05 14 15 24 49 27		0.0454	
V D set12_ircnn_gray	24-05-14 15:24:46.272	- 05.prig - PSINK: 20.00 dB; SSIIVI:	0.0454.	
🔁 01.png	24-05-14 15:24:48.463	3 : 06.png - PSNR: 25.78 dB; SSIM:	0.7953.	
🔁 02.png	24-05-14 15:24:48.647	' : 07.png - PSNR: 26.48 dB; SSIM:	0.7953.	
🕙 03.png	24-05-14 15:24:49.361	: 08.png - PSNR: 29.36 dB; SSIM:	0.8096.	
🕙 04.png	24-05-14 15:24:50 057	· 09 ppg - PSNR 26 17 dB SSIM	0 7704	
🕙 05.png			0.7704.	
🔁 06.png	24-05-14 15:24:50.78	' : 10.png - PSNR: 27.17 dB; SSIM:	0.7155.	
🔁 07.png	24-05-14 15:24:51.476	5 : 11.png - PSNR: 27.14 dB; SSIM:	0.7193.	
🔀 08.png	24-05-14 15:24:52.158	3 : 12.png - PSNR: 26.86 dB; SSIM:	0.7238.	
🔁 09.png	24-05-14 15.24.52 446	5 · 6666 png - PSNR · 31 77 dB: SSI	M· 0.8872	
🔁 10.png	24 05 14 15:24:52 45		DEND: 2	7 40 JD. CCINA. 0 700C
🔁 11.png	24-05-14 15:24:52.450	- Set I	$2_{\rm lrcnn}$ gray - PSNR: 21	.40 UB; SSIIVI: 0.7000
🔁 12.png				
🔀 6666.png				
🛃 set12_ircnn_gray.log				
> D rotinsface				