International Journal of Computer Vision (2022) 130:1440–1458
https://doi.org/10.1007/s11263-022-01583-y

Dual Convolutional Neural Networks for Low-Level Vision

Jinshan Pan[1] · Deqing Sun[2] · Jiawei Zhang[3] · Jinhui Tang[1] · Jian Yang[1] · Yu-Wing Tai[4] · Ming-Hsuan Yang[5,6,7]

Received: 11 July 2021 / Accepted: 6 January 2022 / Published online: 6 April 2022

Abstract
We propose a general dual convolutional neural network (DualCNN) for low-level vision problems, e.g., super-resolution, edge-preserving filtering, deraining, and dehazing. These problems usually involve estimating two components of the target signals: structures and details. Motivated by this, we design the proposed DualCNN to have two parallel branches, which respectively recovers the structures and details in an end-to-end manner. The recovered structures and details can generate desired signals according to the formation model for each particular application. The DualCNN is a flexible framework for low-level vision tasks and can be easily incorporated into existing CNNs. Experimental results show that the DualCNN can be effectively applied to numerous low-level vision tasks with favorable performance against the state-of-the-art methods that have been specially designed for each individual task.

Keywords Low-level vision · Image restoration · Image filtering · Image enhancement · Dual convolutional neural network

1 Introduction

Stimulated by the success of deep learning for high-level vision tasks (Krizhevsky et al. 2012; Girshick 2015; He et al. 2016; Sun et al. 2014), numerous deep models have been developed to tackle low-level vision tasks, e.g., image super-resolution (Dong et al. 2014; Kim et al. 2016a; Dong et al. 2016a, b; Kim et al. 2016b; Liao et al. 2015), inpainting (Ren et al. 2015; Liu et al. 2016), noise removal (Dong et al. 2015; Jain and Seung 2008; Xie et al. 2012), image filtering (Xu et al. 2015; Liu et al. 2016), image deraining (Eigen et al. 2013; Zhang et al. 2020), and dehazing (Ren et al. 2016; Cai et al. 2016). Although achieving impressive performance, the network architectures of these models strongly resemble those developed for high-level classification tasks.

Recent methods for low-level vision tasks are mainly based on plain neural networks (where the architecture is a fully-connected feed-forward network without skip connections) or deeper neural networks with residual learning. As demonstrated in (Ren et al. 2015; Burger et al. 2012), plain neural networks based on reconstruction errors do not outperform the state-of-the-art statistical prior-based approaches on a number of low-level vision problems, e.g., super-resolution (Timofte et al. 2014). Low-level vision tasks usually involve the estimations of two components, low-frequency structures and high-frequency details. It is challenging for a single network to learn both components simultaneously. Consequently, going deeper with plain

Communicated by Subhransu Maji.

✉ Jinhui Tang
jinhuitang@njust.edu.cn

✉ Ming-Hsuan Yang
mhyang@ucmerced.edu

Jinshan Pan
jspan@njust.edu.cn

Deqing Sun
deqingsun@google.com

Jiawei Zhang
zhjw1988@gmail.com

Jian Yang
csjyang@njust.edu.cn

Yu-Wing Tai
yuwing@gmail.com

1 Nanjing University of Science and Technology, Nanjing 210094, China

2 Google, New York, USA

3 SenseTime Research, Shenzhen 518000, China

4 Kuaishou Technology, Shenzhen 518000, China

5 University of California at Merced, Merced, USA

6 Yonsei University, Seoul, South Korea

7 Google, Mountain View, USA

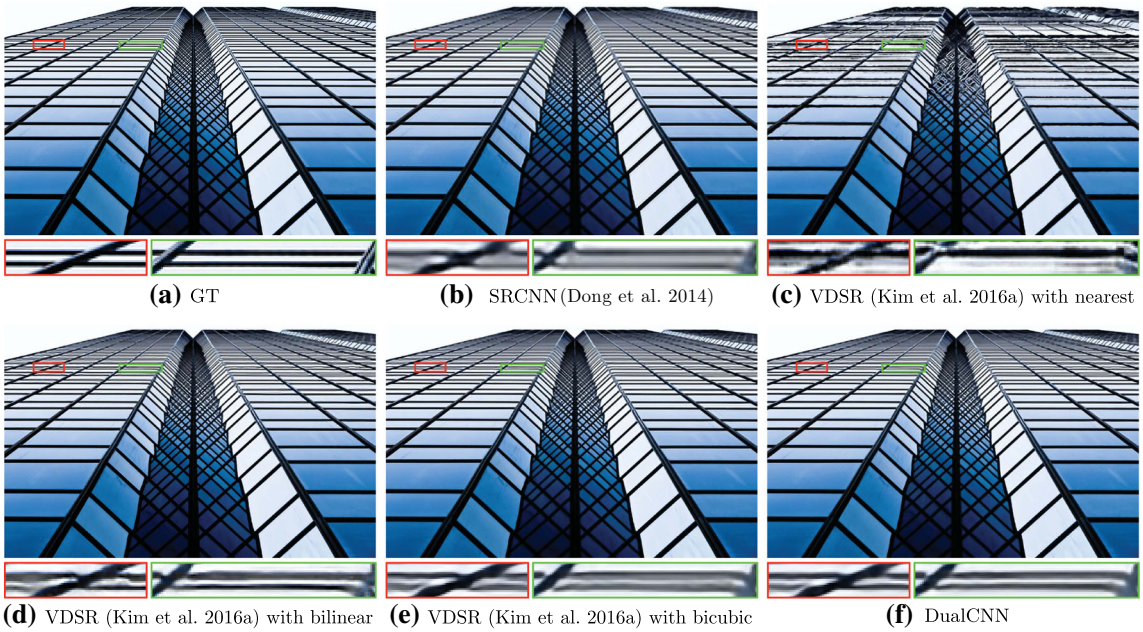


Fig. 1 Super-resolution results by the VDSR method (Kim et al. 2016a) ($\times 4$) with structures recovered by different methods, i.e., nearest neighbor, bilinear, and bicubic upsampling. Residual learning algorithms usually use upsampled images as the base structures and learn image details (i.e., the difference between the upsampled and ground truth images). However, residual learning is less effective in correcting low-frequency errors in the structures, e.g., the structure obtained by the nearest neighbor interpolation in (**c**). In contrast, our algorithm analyzes low and high frequency components to learn both image structures and details, and thus leads to better results

neural networks does not always lead to better performance (Dong et al. 2016a).

Residual learning has been shown to be an effective approach to achieve performance gain with a deeper neural network. The residual learning algorithms for low-level vision tasks (e.g., Kim et al. 2016a) assume that the main structure is given and mainly focus on estimating the residual (details) using a deep neural network. These methods perform well on the premise that the main structures can be properly recovered. Figure 1 shows the image super-resolution results by the VDSR method (Kim et al. 2016a) with structures recovered by different methods. The residual network cannot deal with low-frequency errors contained in the recovered structures (Fig. 1c).

To address this issue, we propose a dual convolutional neural network (DualCNN) to jointly estimate the structures and details. The DualCNN consists of two branches, one to estimate the structures and the other to estimate the details. The modular design of the DualCNN makes it a flexible framework for a variety of low-level vision problems. When trained end-to-end, DualCNN performs favorably against the state-of-the-art methods that are specially designed for each individual task.

We first proposed the DualCNN framework in two papers (Pan et al. 2018a; Yang et al. 2017), which have inspired the following work, including image dehazing (Zhu et al. 2018, 2021; Yang et al. 2019; Guo et al. 2019), image deraining (Li et al. 2019), image denoising (Tian et al. 2020), video super-resolution (Isobe et al. 2020), image super-resolution/deblurring (Singh et al. 2020), and general image restoration (Chen and Davies 2020), to name a few. In this journal version, we extend our preliminary work (Pan et al. 2018a; Yang et al. 2017) with the following notable improvements. First, we develop an alternative way to solve the proposed model and provide more analysis of the proposed network designs. Second, we analyze that the DualCNN model is not limited to the estimation of details and structures, and can be generalized to the image restoration problems (e.g., image dehazing, etc.) according to the corresponding physics models. Third, we show that the proposed DualCNN is a general framework that can be applied to image denoising and non-blind image deconvolution. In addition, we demonstrate that the proposed DualCNN can accommodate existing CNNs for better performance. Finally, we carry out more extensive experiments to demonstrate the effectiveness of the proposed algorithm.

2 Related Work

Numerous deep learning methods have been developed for low-level vision tasks. A comprehensive review is beyond the scope of this work, and we discuss the most related ones in this section.

2.1 Super-Resolution

Significant progress has been made in super-resolution with the advances of deep convolutional neural network (CNN) models (Dong et al. 2014; Kim et al. 2016a, b; Shi et al. 2016; Ledig et al. 2017; Zhang et al. 2018b; Dong et al. 2016b; Haris et al. 2018; Zhang et al. 2018a; Bulat et al. 2018). The SRCNN method (Dong et al. 2014) uses a three-layer CNN for super-resolution. As the SRCNN method is less effective in recovering image details, Kim et al. (2016a) propose a residual learning algorithm based on a deeper neural network, named as VDSR. The VDSR algorithm uses the bicubic interpolation of the low-resolution input as the structure of the high-resolution image and estimates the residual details using a 20-layer CNN. However, if the image structures are not well recovered, the generated results are likely to contain substantial artifacts, as shown in Fig. 1.

Instead of using the interpolated results as the main image structures, recent methods (Shi et al. 2016; Ledig et al. 2017; Zhang et al. 2018b; Dong et al. 2016b; Haris et al. 2018; Zhang et al. 2018a) directly estimate main high-resolution contents from low-resolution images based on deep neural networks. Different from these methods, we develop a framework that simultaneously estimates structures and details for image super-resolution.

2.2 Noise/Artifacts Removal

Numerous algorithms based on CNNs have been developed to remove noise/artifacts (Dong et al. 2015; Jain and Seung 2008; Xie et al. 2012) and unwanted components, e.g., dirty/rainy pixels (Eigen et al. 2013; Zhang et al. 2020). These methods are based on plain neural networks (Eigen et al. 2013), residual learning (Fu et al. 2017a; Zhang and Patel 2018b) or generative adversarial models (Zhang et al. 2020; Qian et al. 2018). However, plain neural networks cannot recover fine details (Kim et al. 2016a; He et al. 2016; Ren et al. 2015) and residual learning cannot correct structural errors as mentioned before. In contrast to existing methods, we formulate this problem as estimations of structures and details of clear images.

2.3 Edge-Preserving Filtering

Significant efforts have been made to approximate image filters using CNNs (Liu et al. 2016; Xu et al. 2015; Chen et al. 2017; Fan et al. 2018a, b). In Xu et al. (2015) develop an efficient CNN model to approximate a number of edge-preserving filters. Liu et al. (2016) use a hybrid network model to approximate a number of edge-preserving filters with favorable performance in terms of model parameter and run time. In Chen et al. (2017) develop a fully136 convolutional network to approximate a number of image processing operators. While these methods aim to preserve main image structures and remove details using a single network, this imposes a difficult learning task. In this work, we show that it is critical to accurately estimate both structures and details for low-level vision tasks.

2.4 Image Dehazing

Existing CNN-based methods for image dehazing (Ren et al. 2016; Cai et al. 2016) mainly focus on estimating the transmission map from an input. Given an estimated transmission map, the atmospheric light can be computed using the air light model. As such, errors in the transmission maps are propagated to the light estimation process. In contrast, recent algorithms directly estimate clear images from hazy images using a single deep neural network (Li et al. 2017; Zhang and Patel 2018a; Li et al. 2018). However, it is difficult to analyze the components of these methods that facilitate the dehazing task. To generate more realistic and accurate results, it is necessary to jointly estimate the transmission map and atmospheric light in one model, which the Dual-CNN is designed for.

A common theme is that we need to design a network based on the corresponding formation models for every low-level vision task. In this paper, we show that most low-level vision problems usually involve the estimation of two components: structures and details. Thus we develop the DualCNN that can be flexibly applied to a variety of low-level vision problems, including the four tasks discussed above.

3 Proposed Approach

As shown in Fig. 2, the proposed dual model consists of two branches, Net-S and Net-D, which respectively estimate the structure and detail components of the target signals from the input. We use super-resolution for illustration. Given a low-resolution image, we first use the bicubic upsampled image as the input. The dual network then learns details and structures according to the formulation model of the image decomposition.

Let X, S, and D denote the ground truth label, output of Net-S, and output of Net-D, respectively. The dual composition loss function enforces the recovered structure S and detail D can generate the ground truth label X using the given formation model:

$$\mathcal{L}_x(S, D) = \frac{1}{2}\|\phi(S) + \varphi(D) - X\|_2^2, \tag{1}$$

where the forms of the functions $\phi(\cdot)$ and $\varphi(\cdot)$ are known and depend on the domain knowledge of each task. For example, the functions $\phi(\cdot)$ and $\varphi(\cdot)$ are identity functions for image decomposition problems (e.g., filtering) and restoration prob-

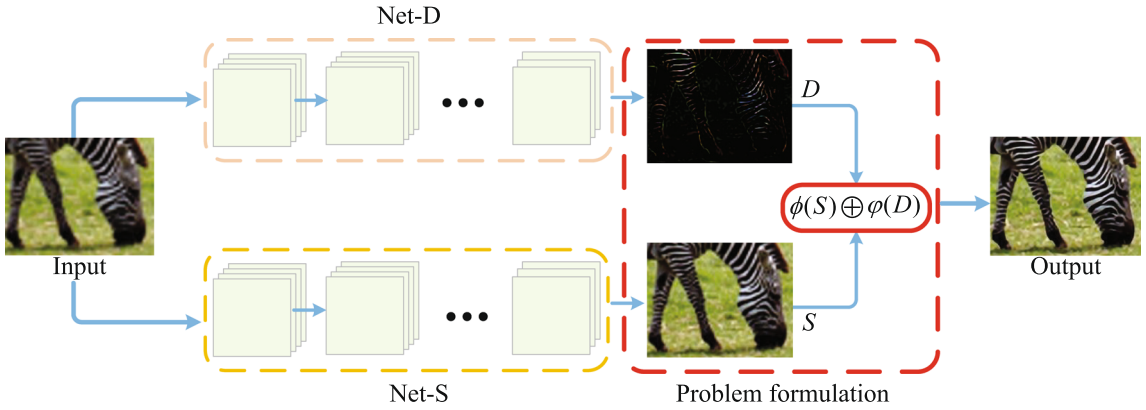


Fig. 2 Proposed DualCNN model. It contains two branches, Net-D and Net-S, and a formulation module. The DualCNN first estimates structures and details and then reconstructs the final results according to the formulation module. The whole network is end-to-end trainable

lems (e.g., super-resolution, denoising, and deraining). We will show that $\phi(\cdot)$ and $\varphi(\cdot)$ can take general forms to deal with specific problems.

3.1 Regularization of the DualCNN Model

The proposed DualCNN model consists of two branches, which may cause instability if only the composition loss (1) is used. For example, if Net-S and Net-D have the same structure, symmetrical solutions exist. To obtain a stable solution, we use individual loss functions to regularize two branches respectively. The loss functions for the Net-S and Net-D are:

$$\mathcal{L}_s(S) = \frac{1}{2}\|S - S_{gt}\|_2^2, \tag{2}$$

$$\mathcal{L}_d(D) = \frac{1}{2}\|D - D_{gt}\|_2^2, \tag{3}$$

where S_{gt} and D_{gt} are ground truths corresponding to the outputs of Net-S and Net-D. Consequently, the overall loss function to train DualCNN is:

$$\mathcal{L} = \alpha\mathcal{L}_x + \lambda\mathcal{L}_s + \gamma\mathcal{L}_d, \tag{4}$$

where α, λ and γ are non-negative trade-off weights. Our framework can also use other loss functions, e.g., perceptual loss for style transfer.

In the training stage, the gradients for Net-S and Net-D can be obtained by:

$$\frac{\partial\mathcal{L}}{\partial S} = \alpha\phi'(S)E + \lambda(S - S_{gt}), \tag{5a}$$

$$\frac{\partial\mathcal{L}}{\partial D} = \alpha\varphi'(D)E + \gamma(D - D_{gt}), \tag{5b}$$

where $E = \phi(S) + \varphi(D) - X$, $\phi'(S)$ and $\varphi'(D)$ are the derivatives with respect to S and D.

In the test stage, we compute the high-quality output X_{est} using the outputs of Net-S and Net-D according to the formation model,

$$X_{est} = \phi(S) + \varphi(D). \tag{6}$$

3.2 Beyond Details and Structures Learning

Aside from image decomposition and restoration problems, the proposed model can handle other low-level vision problems by modifying the composition loss function (1) according to the corresponding formation models.

3.2.1 Image Dehazing

The image dehazing model can be described using the air light model,

$$I = XD + S(1 - D), \tag{7}$$

where I is the hazy image, S is the atmospheric light, and D is the medium transmission map, which describes the portion of the light that reaches the camera from scene surfaces. For consistency, we still use X as the clear image in (7). With the formulation model (7), we can set $\phi(S) = S(1 - D)$ and $\varphi(D) = XD$ in (1) within the DualCNN framework.[1] As a result, the composition loss function (1) for image dehazing becomes

$$\mathcal{L}_x(S, D) = \frac{1}{2}\|XD + S(1 - D) - I\|_2^2. \tag{8}$$

The other two loss functions (2) and (3) remain the same. In the training phase, we use the same method (Ren et al. 2016) to generate the atmospheric light S, the transmission map

[1] As an extension of the details and structures learning, we do not assume that $\phi(\cdot)$ is independent of $\varphi(\cdot)$.

D and construct hazy/haze-free image pairs. The implementation details of the training stage are presented in Sect. 4.7.

In the test phase, the clear image X_{est} can be reconstructed by the outputs of Net-D and Net-S, i.e.,

$$X_{est} = \frac{I - S}{\max\{D, d_0\}} + S, \tag{9}$$

where d_0 is used to prevent division by zero and a typical value is 0.1.

From image dehazing, we note that the formation model of image dehazing actually constrains the DualCNN to ensure that it is able to estimate the key components for image dehazing. This indicates that we can use other formation models to constrain the DualCNN model to solve specific problems.

4 Experimental Results

We evaluate the DualCNN model on several low-level vision tasks including super-resolution, edge-preserving smoothing, deraining, and dehazing. More experimental results and findings can be found in the supplementary material. The source code and trained models are available at https://github.com/jspan/dualcnn.

4.1 Network Architectures

Motivated by the success of SRCNN and VDSR for super-resolution, we use three convolution layers followed by the ReLU function for the Net-S module. The filter sizes of each layer are 9×9, 1×1, and 5×5, respectively. The numbers of each layer are 64, 32, and 1, respectively. For the Net-D module, we use 20 convolution layers followed by the ReLU function. The filter size of each layer is 3×3, and the filter number in each layer is 64. The batch size is set to be 64 and the learning rate is 10^{-4}. Although each branch of the proposed model is similar to SRCNN or VDSR, both our analysis and experimental results show that the proposed model is significantly different from these methods and achieves better results.

4.2 Image Super-resolution

4.2.1 Training Data

For image super-resolution, we generate the training data by randomly sampling 250,000 patches with the size of 41×41 pixels from 291 natural images in the BSDS500 dataset (Martin et al. 2001). We apply the Gaussian filter to each ground truth label X to obtain S_{gt}. The ground truth D_{gt} is the difference between the ground truth label X and the structure S_{gt}.

For super-resolution, we set $\phi(S) = S$ and $\varphi(D) = D$. The weights α, λ and γ in the loss function (4) are set to be 1, 0.001 and 0.01, respectively. To achieve better results, we use the pre-trained models of SRCNN and VDSR as the initializations of Net-S and Net-D.

We present quantitative and qualitative comparisons against the state-of-the-art methods including A+ (Timofte et al. 2014), SelfEx (Huang et al. 2015), SRCNN (Dong et al. 2014), ESPCN (Shi et al. 2016), SRGAN (Ledig et al. 2017), and VDSR (Kim et al. 2016a). Table 1 shows quantitative evaluations on benchmark datasets. Overall, the proposed method performs favorably against state-of-the-art methods. The architecture of one branch in the DualCNN is either similar to SRCNN or VDSR. However, the results generated by the DualCNN have the highest average PSNR values, which demonstrate the effectiveness of the proposed dual model.

We note that Lai et al. (2019) propose an effective network that progressively restores the sub-band residuals of high-resolution images based on an image pyramid. Due to the pyramid structure, this method generates better results than the residual learning method (Kim et al. 2016a). In contrast, the proposed method develops two branches to estimate structures and details separately, which does not require the progressive restoration step and thus generates comparable results as shown in Table 1.

To better understand the sources of performance gains with respect to the VDSR method (Kim et al. 2016a), we improve the VDSR method by adding more convolutional layers to enlarge its model capacity so that it has similar model parameters to that of the proposed method (VDSR-M for short in Table 1). Table 1 shows that the proposed method still generates better results than VDSR-M even though the model parameter is fewer.

Figure 3 shows some super-resolution results by the evaluated methods. The proposed algorithm can better preserve the main structures than state-of-the-art methods.

We note that the aforementioned method needs to generate image details and structures to train DualCNN. However, separating image details and structures from clean images is challenging in most cases. To avoid this complex step, we adopt an alternative DualCNN-S model. We use S_{gt} as the ground truth label X. As the Net-S module is less effective to estimate details, we use the Net-D module to estimate the errors between the output of Net-S and ground truth label X. Table 1 shows that the DualCNN-S model achieves competitive performance against the DualCNN model. Note that the Net-D in DualCNN-S is used to learn the difference between the predicted results of Net-S and ground truths. In addition, given that the details are obtained by the difference (i.e., the residual) between the ground truth label X and structure S_{gt}, we refer to the difference learned by the Net-D in DualCNN-S as the pseudo detail. Table 2 summarizes the definitions of

Table 1 Quantitative evaluations for the state-of-the-art super-resolution methods on the benchmark datasets (Set5, Set14, B100, Urban100, and Manga109) in terms of PSNR and SSIM. "VDSR-M" denotes the improved VDSR method (Kim et al. 2016a) by using the similar model parameters to the proposed method

Algorithms	Scale	Set5 PSNR/SSIM	Set14 PSNR/SSIM	B100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM
Bicubic	×2	33.68/0.9303	30.33/0.8694	29.51/0.8425	26.86/0.8398	30.76/0.9344
A+	×2	36.58/0.9546	32.44/0.9060	30.67/0.8706	29.22/0.8935	35.29/0.9669
SelfEx	×2	36.60/0.9547	32.48/0.9060	31.14/0.8853	29.55/0.8980	35.82/0.9690
SRCNN	×2	36.36/0.9523	32.35/0.9045	31.09/0.8841	29.10/0.8893	34.96/0.9644
ESPCN	×2	36.73/0.9547	32.40/0.9056	31.45/0.9032	29.24/0.8920	35.01/0.9652
VDSR	×2	37.55/0.9588	33.21/0.9130	31.85/0.8954	30.76/0.9137	37.32/0.9731
SRGAN	×2	37.01/0.9548	32.69/0.9049	31.41/0.8892	29.44/0.8745	35.21/0.9662
LapSRN	×2	37.53/0.9591	33.15/0.9127	31.74/0.8942	30.40/0.9096	37.22/0.9739
VDSR-M	×2	37.54/0.9584	33.06/0.9126	31.87/0.8953	30.70/0.9131	37.35/0.9732
DualCNN	×2	**37.73/0.9589**	33.30/0.9131	31.92/0.8957	30.99/0.9157	**37.61**/0.9733
DualCNN-S	×2	**37.73/0.9589**	**33.31/0.9132**	**31.93/0.8959**	**31.01/0.9158**	**37.61/0.9735**
Bicubic	×3	30.42/0.8691	27.64/0.7753	27.17/0.7378	24.44/0.7344	26.91/0.8558
A+	×3	32.66/0.9094	29.26/0.8201	28.15/0.7767	26.03/0.7972	29.88/0.9100
SelfEx	×3	32.67/0.9106	29.34/0.8227	28.26/0.7838	26.45/0.8100	27.57/0.8210
SRCNN	×3	32.46/0.9039	29.16/0.8165	28.18/0.7795	25.87/0.7882	29.79/0.9032
ESPCN	×3	33.07/0.9134	29.54/0.8251	28.28/0.8052	25.92/0.7897	29.81/0.9033
VDSR	×3	33.70/0.9218	29.91/0.8326	28.79/0.7972	27.13/0.8275	32.15/0.9336
SRGAN	×3	33.54/0.9170	29.54/0.8227	28.68/0.8163	26.62/0.8159	29.98/0.9120
LapSRN	×3	33.85/0.9230	29.92/0.8327	28.78/0.7964	27.05/0.8266	32.17/0.9342
VDSR-M	×3	33.71/0.9225	29.88/0.8331	28.79/0.7972	27.13/0.8280	32.15/0.9336
DualCNN	×3	**33.90**/0.9233	**29.96/0.8334**	**28.82**/0.7975	**27.24/0.8296**	**32.22/0.9344**
DualCNN-S	×3	**33.90/0.9234**	**29.96/0.8334**	**28.82/0.7976**	27.23/0.8295	32.19/0.9342
Bicubic	×4	28.44/0.8114	26.10/0.7044	25.93/0.6670	23.13/0.6574	24.87/0.7868
A+	×4	30.35/0.8623	27.43/0.7514	26.74/0.7046	24.33/0.7186	27.00/0.8482
SelfEx	×4	30.36/0.8635	27.54/0.7550	26.81/0.7106	24.83/0.7400	27.83/0.8660
SRCNN	×4	30.15/0.8551	27.33/0.7441	26.69/0.7018	24.15/0.7057	26.88/0.8380
ESPCN	×4	30.27/0.8540	27.17/0.7401	26.68/0.7218	24.15/0.7031	26.89/0.8382
VDSR	×4	31.35/0.8838	28.06/0.7685	27.20/0.7236	25.14/0.7523	28.91/0.8859
SRGAN	×4	31.35/0.8797	27.84/0.7588	26.92/0.7426	24.38/0.7301	28.13/0.8636
LapSRN	×4	**31.57/0.8870**	**28.20/0.7707**	**27.29/0.7256**	25.20/**0.7547**	**29.07/0.8890**
VDSR-M	×4	31.38/0.8843	28.06/0.7687	27.21/0.7242	25.15/0.7528	28.87/0.8859
DualCNN	×4	31.55/0.8858	28.17/0.7699	27.26/0.7247	**25.25**/0.7546	29.01/0.8873
DualCNN-S	×4	31.55/0.8858	28.16/0.7697	27.26/0.7248	**25.25**/0.7545	29.00/0.8870

The results of the best performance are denoted in bold

$\phi(\cdot)$ and $\varphi(\cdot)$ in the DualCNN and DualCNN-S for different tasks.

4.2.2 Run Time and Model Parameter

We evaluate the run time and model parameter of the compared methods on a machine with an Intel Core i7-7700 CPU and an NVIDIA GTX 1080Ti GPU. Table 3 shows that the run time of the DualCNN model is comparable to VDSR and VDSR-M, but the proposed model achieves better results on the super-resolution benchmark datasets than VDSR and VDSR-M.

In addition, the proposed method has relatively fewer model parameters but with favorable performance than the LapSRN method (Lai et al. 2019), which demonstrates that the major performance is mainly due to the use of the DualCNN model instead of using large capacity models.

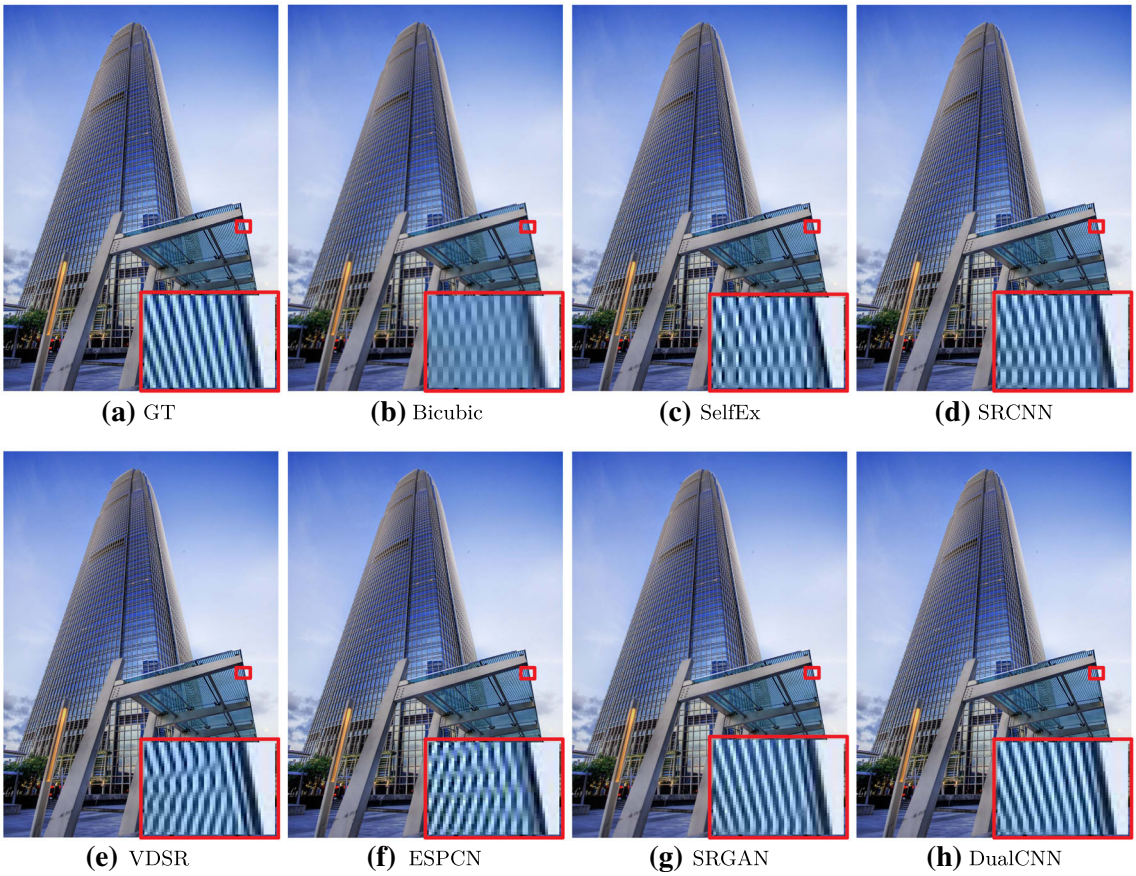


Fig. 3 Super-resolution (×2) results. While state-of-the-art methods do not preserve the main structures of the images, the proposed method is able to upsample this image well

Table 2 We summarize how $\phi(S)$ and $\varphi(D)$ are used for different vision tasks, where S and D denote the outputs of the Net-S and Net-D

	DualCNN		DualCNN-S	
	$\phi(S)$	$\varphi(D)$	$\phi(S)$	$\varphi(D)$
Super-resolution Denoising Non-blind Image Deconvolution Edge-preserving Filtering Deraining	S: structure	D: detail	S: pseudo structure	D: pseudo detail
Dehazing	$S(1-D)$ S: atmospheric light D: transmission map	XD X: clear image D: transmission map	–	–

Table 3 Average run time (seconds) and model parameter of the evaluated methods on image super-resolution (×4) using the Set5 test dataset

Methods	SRCNN	VDSR	VDSR-M	SRGAN	LapSRN	EDSR	RDN	DBPN	DualCNN (SRCNN+VDSR)
Average run time	0.0011	0.0045	0.0051	0.0120	0.0456	0.0190	0.0488	0.0235	0.0057
Model parameter	0.01M	0.66M	0.70M	1.55M	0.87M	43.09M	5.58M	10.43M	0.67M

4.3 Image Denoising

As image denoising methods are typically evaluated quantitatively with synthetically generated images, we generate noisy images from clear ones with additive noise for experiments. Similar to Zhang et al. (2017c), we use the training dataset from the BSDS dataset (Martin et al. 2001) to generate the training data.

In the training stage, we first generate the noisy images using three noise levels by setting the standard deviation of the Gaussian function to be 15, 25, and 50, respectively. Then we train three models based on the three noise level settings, respectively. In addition, we randomly add the Gaussian noise to each image, where the noise level ranges from 0 to 10% and train the proposed model on this training dataset with the mixed noise levels.

In the test stage, we use the test dataset with 200 clear images by Martin et al. (2001) to evaluate the proposed method. Similar to Zhang et al. (2017b), we use three noise levels by setting the standard deviation of the Gaussian function to be 15, 25, and 50, respectively. In addition, we add the noise with different noise levels to each test image, where the test images and training images do not overlap.

We quantitatively and qualitatively evaluate the proposed DualCNN model against the state-of-the-art denoising methods based on statistical priors (BM3D (Dabov et al. 2007), EPLL (Zoran and Weiss 2011), CSF (Schmidt and Roth 2014)) and deep neural networks (MLP (Burger et al. 2012), DNCNN (Zhang et al. 2017b), IRCNN (Zhang et al. 2017c)). Table 4 summarizes the denoising results on the BSDS test dataset. Although the proposed method is not designed for image denoising, it is able to remove noise and generates high-quality images compared to the state-of-the-art denoising methods.

In addition, we add the random noise with the noise levels of 0 to 10% to each test image and evaluate the proposed model trained on the images with mixed noise levels. The results in Table 4 (i.e., mixed) demonstrate that the proposed method performs well on the images with mixed noise levels.

Figure 4 shows denoised results from the evaluated methods. While state-of-the-art methods do not effectively restore the structural details, the proposed algorithm can accurately estimate both clear details and structures from the input image and generates a better-denoised image. We note that the Net-D module is able to separate image noise (Fig. 4f) from main structures (Fig. 4e) of the input.

4.4 Non-blind Image Deconvolution

We apply the DualCNN model to image deconvolution (Levin et al. 2007; Zhang et al. 2017c, a; Dong et al. 2021) using the same training dataset as Zhang et al. (2017a). For this task, we first use the inverse filter which is implemented by the fast

Table 4 Quantitative evaluations on image denoising using the BSDS test dataset (Martin et al. 2001) in terms of PSNR, SSIM, and model parameter

Noise level	Input	BM3D (Dabov et al. 2007)	EPLL (Zoran and Weiss 2011)	CSF (Schmidt and Roth 2014)	MLP (Burger et al. 2012)	DNCNN (Zhang et al. 2017b)	IRCNN (Zhang et al. 2017c)	VDSR-M	DualCNN-S
15%	23.82/0.5193	30.40/0.8650	30.39/0.8717	30.52/0.8666	–/–	30.89/0.8810	31.17/0.8870	31.60/0.8889	**31.87/0.8952**
25%	19.79/0.3504	28.31/0.8008	28.33/0.8086	28.45/0.8026	28.46/0.8081	28.85/0.8244	29.04/0.8310	29.08/0.8255	**29.31/0.8334**
50%	14.77/0.1842	25.55/0.6936	25.57/0.6947	25.88/0.7098	25.86/0.7095	26.09/0.7213	26.15/0.7256	26.02/0.7128	**26.23/0.7238**
Mixed	27.23/0.6350	31.60/0.8765	26.08/0.7820	30.10/0.8164	28.91/0.7854	34.45/0.9193	34.38/0.9108	34.32/0.9028	**34.47/0.9198**
Model parameter	–	–	–	–	–	0.56M	0.19M	0.7M	0.67M

The results of the best performance are denoted in bold

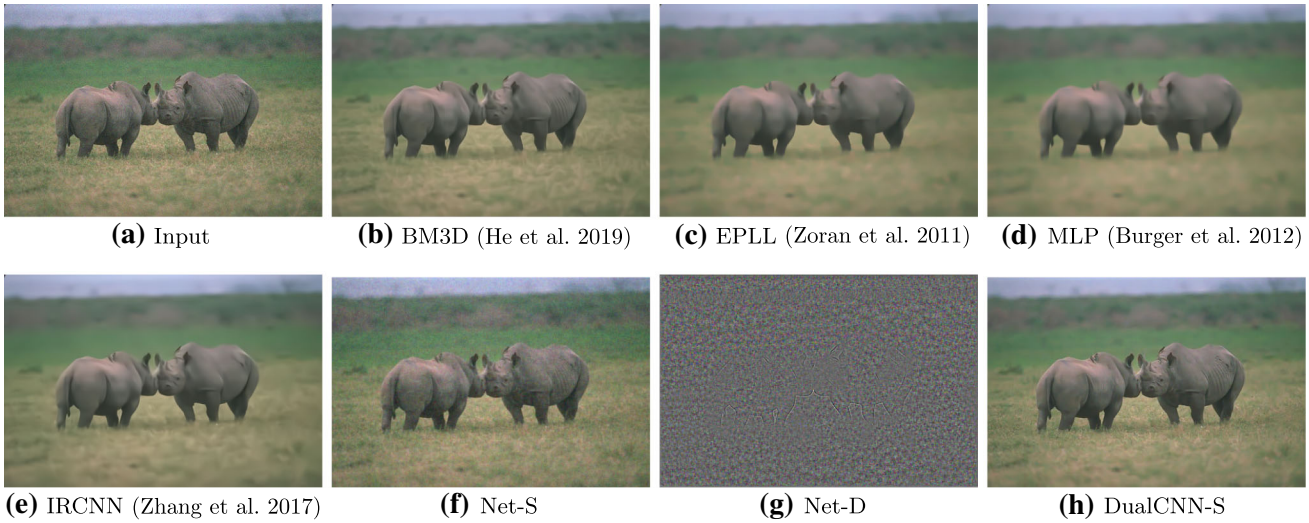


Fig. 4 Image denoising results. In contrast to the deep end-to-end trainable networks-based methods, the proposed DualCNN model simultaneously estimates structures and details according to the dual composition model. The denoised image in (h) contains fine details

Table 5 Quantitative evaluations on non-blind image deconvolution using the datasets (Levin et al. (2009) and Martin et al. (2001)) in terms of PSNR and SSIM

Methods	Blurred images	HL (Krishnan and Fergus 2009)	CSF (Schmidt and Roth 2014)	FCNN (Zhang et al. 2017a)	IRCNN (Zhang et al. 2017c)	VDSR-M	DualCNN-S
Dataset (Levin et al. 2009)	22.81/0.6895	28.43/0.8868	28.68/0.8826	24.29/0.7177	28.64/**0.9026**	28.39/0.8615	**29.20**/0.8912
Dataset (Martin et al. 2001)	22.19/0.5495	31.65/0.9034	31.65/0.9034	23.48/0.7683	**32.71/0.9200**	32.02/0.9018	32.21/**0.9200**

The results of the best performance are denoted in bold

Fourier transformation algorithm (Levin et al. 2007) to generate intermediate latent images and feed them as the inputs of the DualCNN model. Other settings are the same as those for image denoising.

We quantitatively evaluate the DualCNN model on the image deblurring dataset by Levin et al. (2009). We use the blind deblurring algorithm (Pan et al. 2018b) to generate blur kernels for test and then apply our DualCNN model to deblur images. The results in Table 5 demonstrate that the proposed DualCNN model generates competitive results against the state-of-the-art deep learning based methods (Zhang et al. 2017c, a).

4.5 Edge-Preserving Filtering

Similar to the methods in Liu et al. (2016) and Xu et al. (2015), we apply the DualCNN to learn edge preserving image filters including L_0 smoothing (Xu et al. 2011), relative total variation (RTV) (Xu et al. 2012), and weighted median filter (WMF) (Zhang et al. 2014). We generate the training data by randomly sampling 1 million patches (clear/filtered pairs) from 200 natural images in Martin et al. (2001). Each image patch is of 64×64 pixels, and other settings of generating training data are the same as those used in Xu et al. (2015).

We evaluate the proposed DualCNN model against methods (Liu et al. 2016; Xu et al. 2015) using the dataset from Xu et al. (2015). Table 6 summarizes the PSNR values of all evaluated methods. As Xu et al. (2015) use image gradients to train their model and the filtered results are reconstructed by solving a constrained optimization problem, it performs better for approximating L_0 smoothing. However, our method does not need additional steps and generates high quality filtered images with significant improvements over the state-of-the-art deep learning based methods, particularly on RTV and WMF.

We note that the architecture of Net-D is similar to that of VDSR. As such, we retrain the network of VDSR for these problems. The results in Table 6 show that only using residual learning does not always generate high-quality filtered images.

Figure 6 shows the filtering results of the approximating RTV (Xu et al. 2012). The state-of-the-art methods (Liu et al. 2016; Xu et al. 2015) fail to smooth the structures (e.g., the eyes in the green boxes) that are supposed to be removed

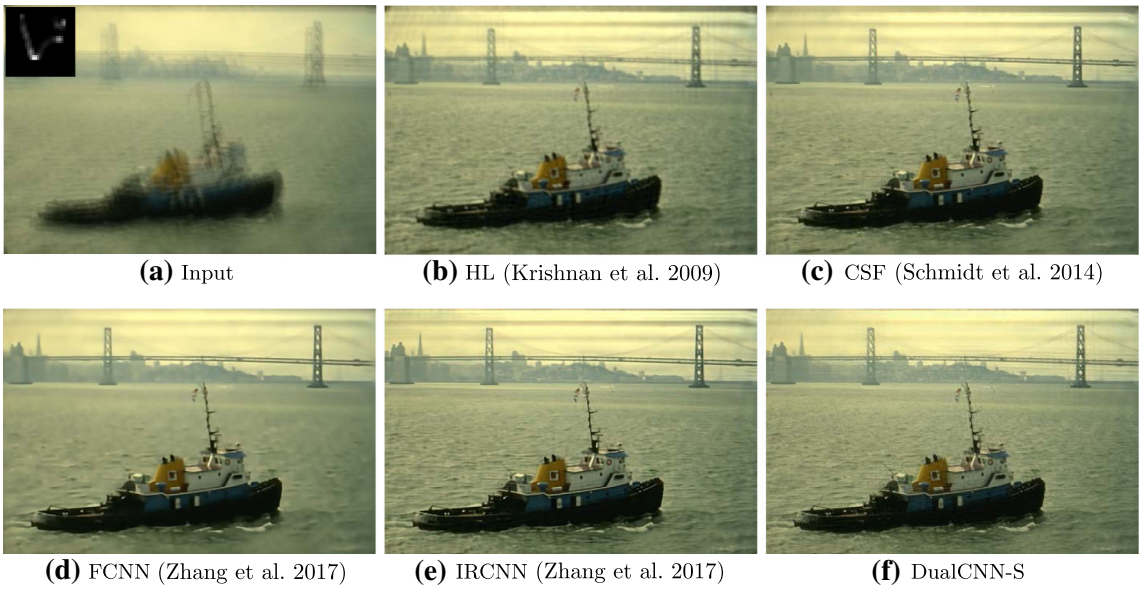


Fig. 5 Non-blind image deconvolution results. The proposed DualCNN model generates the deblurred image with few artifacts and fine details

Table 6 PSNR values for learning various image filters on the test dataset (Xu et al. 2014)

	Xu et al. (2015)	Liu et al. (2016)	VDSR (Kim et al. 2016a)	Net-S	DualCNN-S
L_0	**32.8**	30.9	31.5	28.0	31.4
WMF	31.4	34.0	38.5	29.2	**39.1**
RTV	32.1	37.1	41.6	32.0	**42.1**

The results of the best performance are denoted in bold

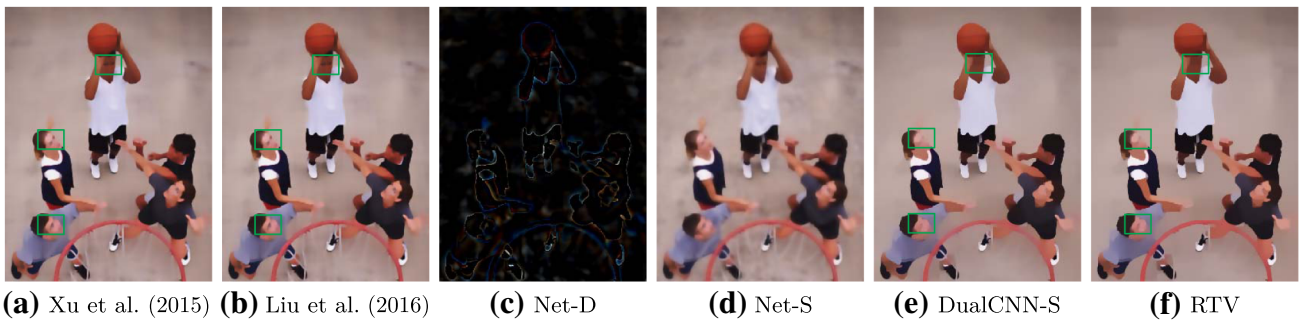


Fig. 6 Images generated by the learning-based relative total variation (RTV) filters. Existing deep learning based methods are not able to remove the details and structures that are supposed to be removed [the boxes in (**a, b**)]. **c, d** Outputs of the two branches of the proposed model. **f** Result by the original implementation of RTV. Better enlarge and view on a screen

using the RTV filter (Fig. 6f). In addition, the results with only one branch (i.e., Net-S) have lower PSNR values (Table 6) and some remaining tiny structures (Fig. 6d). In contrast, the proposed method with joint learning of structures and details preserves more accurate results, and the filtered images are significantly closer to the ground truth.

We further evaluate the run time of the proposed algorithm against state-of-the-art methods. Table 7 shows that the proposed algorithm is much more efficient than state-of-the-art methods.

4.6 Image Deraining

The goal of deraining is to recover clear contents from rainy images. This process can be regarded as recovering details (rainy streaks) and structures (clear images) from inputs.

To train the proposed DualCNN for image deraining, we generate the training data by randomly sampling 1 million patches (rainy/clear pairs) from the rainy image dataset used in Zhang et al. (2020). The size of each image patch used in the training stage is 64 × 64 pixels. We use the test

Table 7 Average run time (seconds) for learning various image filters on the images with size of 1920 × 1080 pixels

	Original implementation	Xu et al. (2015)	Liu et al. (2016)	VDSR (Kim et al. 2016a)	DualCNN-S
L_0	8.4478				
WMF	2.7079	5.3996	0.9514	0.0162	0.0198
RTV	9.1364				

Table 8 Quantitative evaluations using the synthetic rainy dataset (Zhang et al. 2020) in terms of PSNR, SSIM, and model parameter

Methods	SPM (Kang et al. 2012)	PRM (Chen and Hsu 2013)	CNN (Fu et al. 2017a)	GMM (Li et al. 2016)	ID-CGAN (Zhang et al. 2020)	DID-MDN (Zhang and Patel 2018b)	Net-S	DualCNN-S
Avg. PSNR	18.88	20.46	19.12	22.27	22.73	21.26	22.75	**24.60**
Avg. SSIM	0.5832	0.7297	0.6013	0.7413	0.8133	0.7632	0.7781	**0.8190**
Model parameter	–	–	0.75M	–	1.82M	0.56M	0.01M	0.67M

The results of the best performance are denoted in bold

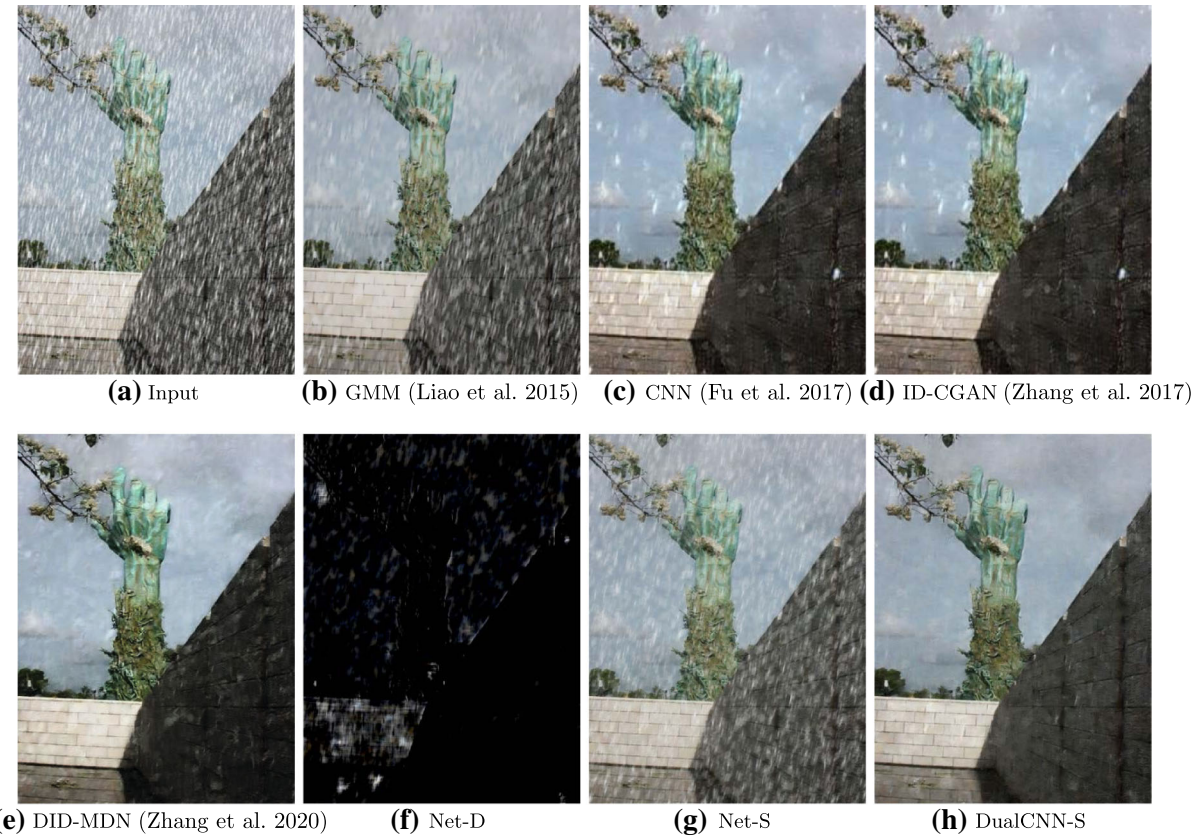


Fig. 7 Image deraining results. The proposed method is able to remove rainy streaks from the input image

dataset (Zhang et al. 2020) for evaluation where the test images and training images do not overlap.

The evaluated state-of-the-art methods are based on statistical priors (SPM (Kang et al. 2012), PRM (Chen and Hsu 2013), GMM (Li et al. 2016)) and deep neural networks (i.e., CNN (Fu et al. 2017a) , ID-CGAN (Zhang et al. 2020), DID-MDN (Zhang and Patel 2018b)). Table 8 shows the average PSNR values of restored images on the test dataset (Zhang et al. 2020). Overall, the proposed method generates the results with the highest PSNR and SSIM values.

Figure 7 shows derained results by the evaluated methods. The proposed algorithm is able to estimate two key components (i.e., $\varphi(D)$ and $\phi(S)$ in Fig. 7f, g) to better facilitate image deraining. We note that the two estimated key

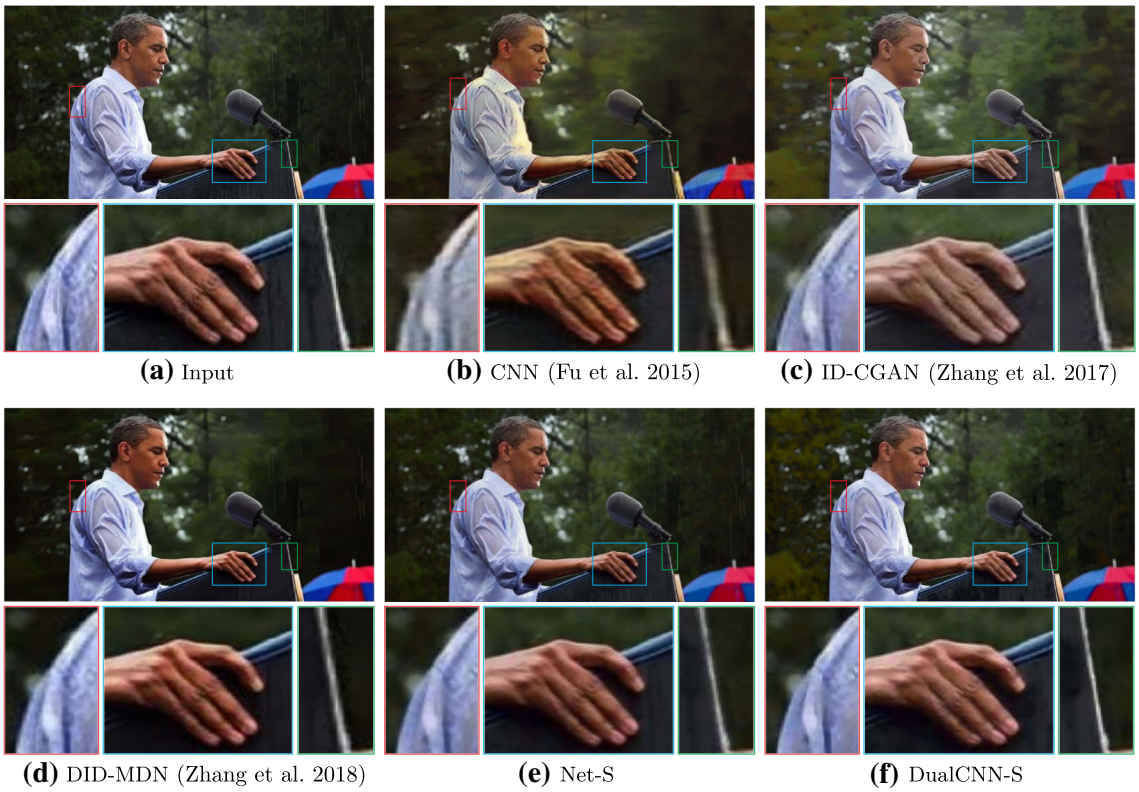


Fig. 8 Image deraining results on real examples by deep learning-based methods. The proposed method is able to remove rainy streaks from the input image and generates clearer images with fine details

components (i.e., $\varphi(D)$ and $\phi(S)$ in Fig. 7f, g) do not exactly correspond to the image details and structures. This is mainly because the Net-D in the proposed DualCNN-S model is not required to estimate the details of the clear images. It is used to estimate the errors between the output of the Net-S and ground truth clear images. Thus, the learned errors by the Net-D with the output of the Net-S can facilitate better image restoration.

Figure 8 shows the derained results of the evaluated methods on a real image. We note that the algorithm in Fu et al. (2017b) removes rain streaks by a deep detail network. However, this method depends on whether the image decomposition method is able to extract details or not. The results in Fig. 8c demonstrate the algorithm by Fu et al. (2017b) is less effective in removing rain streaks in the real image. In contrast, our method generates much clearer images compared to state-of-the-art algorithms.

4.7 Image Dehazing

As discussed in Sect. 3.2, the proposed method can be applied to the image dehazing. Similar to the method in Ren et al. (2016), we synthesize the hazy image dataset using the NYU depth dataset (Silberman et al. 2012) and Make3D dataset (Saxena et al. 2009) and generate the training data including hazy/clear pairs (I/J), atmospheric light (S), and transmission map (D). The size of each image patch used in the training stage is 64×64 pixels. The weights α, λ and γ are set to be 0.1, 0.9, and 0.9, respectively. In the test stage, we randomly choose 64 hazy images from the synthetic dataset for evaluations, where the test images and the training images do not overlap.

We quantitatively evaluate our method on the above synthetic hazy test images. As summarized in Table 9, the proposed method performs favorably against the state-of-the-art methods for image dehazing.

Figure 9 shows dehazed results from the test dataset. The proposed DualCNN model is able to remove haze and generate better dehazed results.

The dehazed results on real images in Fig. 10 show that the proposed method can recover the atmospheric light (Fig. 10e) and transmission map (Fig. 10f) well, thereby facilitating to recover the clear image (Fig. 10g).

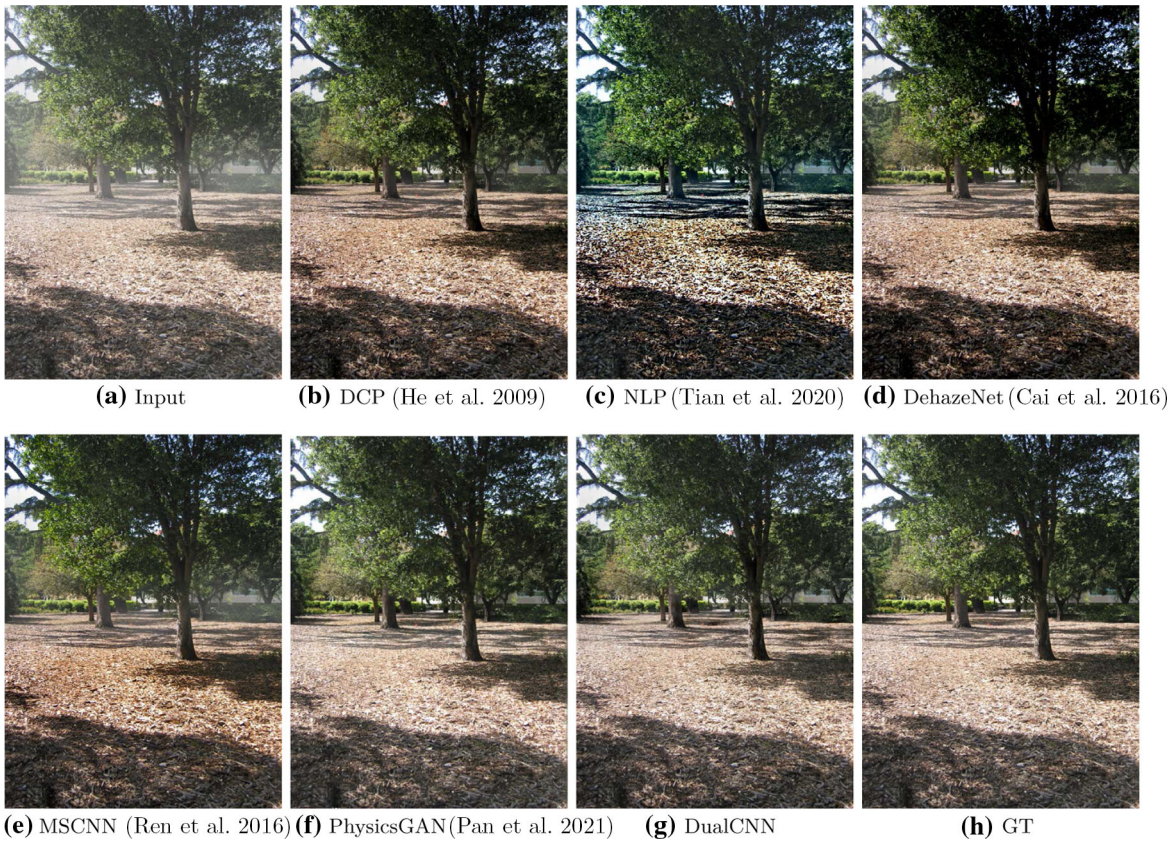
5 Analysis and Discussion

In this section, we analyze the DualCNN model and compare it with the most related methods.

Table 9 Quantitative evaluations using the proposed synthetic hazy test images in terms of PSNR, SSIM, and model parameter

Methods	DCP (He et al. 2009)	Meng et al. (Meng et al. 2013)	NLP (Berman et al. 2016)	DehazeNet (Cai et al. 2016)	MSCNN (Ren et al. 2016)	PDN (Zhang and Patel 2018a)	PhysicsGAN (Pan et al. 2021)	DualCNN
Avg. PSNR	21.52	16.62	15.91	19.22	19.99	27.18	**31.94**	27.28
Avg. SSIM	0.9149	0.8805	0.8196	0.8368	0.9102	0.8703	0.9369	**0.9526**
Model parameter	–	–	–	0.01M	0.01M	69.66M	16.91M	0.67M

The results of the best performance are denoted in bold



(a) Input **(b)** DCP (He et al. 2009) **(c)** NLP (Tian et al. 2020) **(d)** DehazeNet (Cai et al. 2016)

(e) MSCNN (Ren et al. 2016) **(f)** PhysicsGAN (Pan et al. 2021) **(g)** DualCNN **(h)** GT

Fig. 9 Image dehazing results on a synthetic image. Constrained by the formulation model of image dehazing, the proposed DualCNN model simultaneously estimates atmosphere light and transmission maps and thus performs comparably to the state-of-the-art methods

5.1 Effect of the DualCNN Architecture

Lin et al. (2015) develop a bilinear model to extract complementary features for fine-grained visual recognition. In contrast, the proposed DualCNN is motivated by decomposing signals into structures and details. More importantly, the formulation of the proposed model facilitates incorporating the domain knowledge of each individual application. Thus, the DualCNN model can be effectively applied to numerous low-level vision problems, e.g., super-resolution, image filtering, deraining, and dehazing.

Numerous deep learning methods have been developed based on a single branch for low-level vision problems, e.g., SRCNN (Dong et al. 2014) and VDSR (Kim et al. 2016a). One natural question is why deeper architectures do not necessarily lead to better performance. In principle, a sufficiently deep neural network has sufficient capacity to solve any problem given enough training data. However, it is difficult to learn very deep CNN models for these problems while ensuring high efficiency and simplicity.

For experimental validation, we use the SRCNN and a deeper model, i.e., VDSR, for image filtering and deraining. The experimental settings are discussed the same as discussed in Sect. 4.

Sample results using the VDSR model are shown in Fig. 11. While the residual learning (i.e., VDSR) approach performs better than the SRCNN, the generated images with

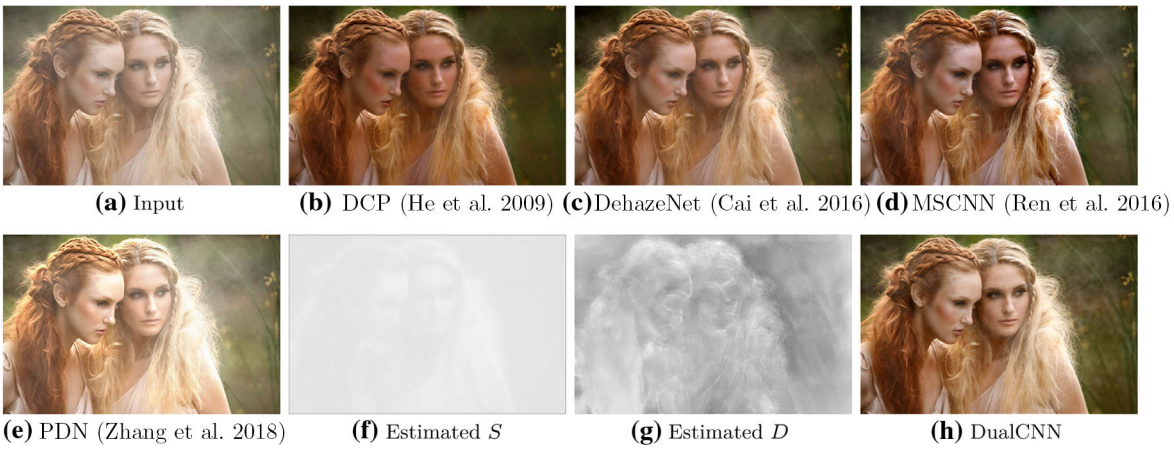


Fig. 10 Image dehazing results on a real image. Constrained by the formation model of image dehazing, the proposed algorithm simultaneously estimates the atmosphere light in (**e**) and transmission map in (**f**) and thus performs comparably to the state-of-the-art methods

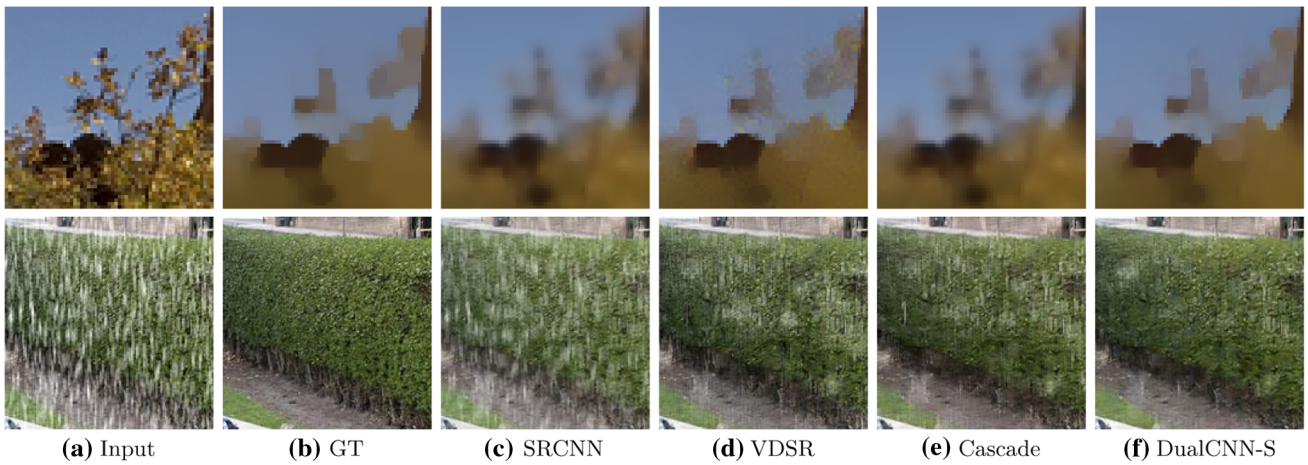


Fig. 11 Effectiveness of the proposed DualCNN model. **c**–**f** Comparisons between existing CNNs (including plain net and ResNet) and the proposed net in edge-preserving filtering and image deraining. The plain net (i.e., **c**), ResNet and its deeper version (i.e., **d, e**) generate results with significant artifacts. Quantitative evaluations are included in Table 12

Table 10 Quantitative evaluations of different network architectures on the image super-resolution ($\times 2$) datasets in terms of PSNR, SSIM, and model parameter. The "Cascade-feature-input" denotes that the intermediate output of the network is {input image, output features of the Net-S, and output image of the Net-S}. Although using more features in "Cascade-feature-input" leads to the performance improvement compared to "Cascade", it leads to a larger deep model. Moreover, it does not perform well compared to the proposed method

Different nets	Set5	Set14	B100	Urban100	Manga109	Model parameter
Cascade	37.51/0.9581	33.13/0.9123	31.82/0.8946	30.75/0.9130	37.27/0.9731	0.67M
Cascade-feature-input	37.73/0.9589	33.25/0.9129	31.91/0.8954	31.01/0.9161	37.55/0.9732	0.85M
DualCNN-S	37.73/0.9589	33.31/0.9132	31.93/0.8959	31.01/0.9158	37.61/0.9735	0.67M

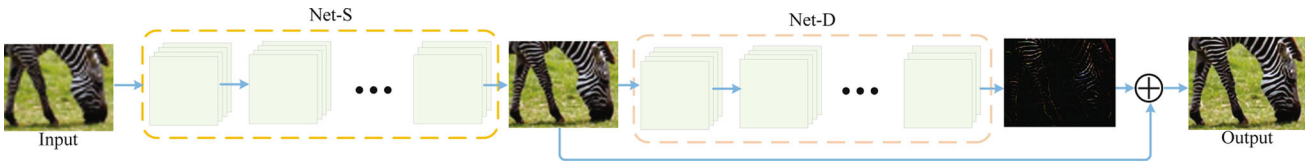


Fig. 12 An alternative cascaded architecture that estimates the structure and details sequentially

Table 11 Quantitative evaluations of single branch networks on the image super-resolution (×2) datasets in terms of PSNR and SSIM. The "SRCNN-Split" denotes the single branch using the SRCNN network and "VDSR-Split" denotes the single branch using the VDSR network. The "DualCNN w/all GTs" denotes that each branch in the proposed DualCNN model uses GT images as the constraint

Different nets	Set5	Set14	B100	Urban100	Manga109
SRCNN-Split	36.58/0.9528	32.47/0.9051	31.24/0.8859	29.26/0.8910	35.34/0.9649
VDSR-Split	37.68/0.9585	33.22/0.9126	31.88/0.8947	30.88/0.9147	37.33/0.9729
DualCNN w/all GTs	37.65/0.9588	33.25/0.9131	31.90/0.8956	30.92/0.9149	37.47/0.9731
DualCNN-S	**37.73/0.9589**	**33.31/0.9132**	**31.93/0.8959**	**31.01/0.9158**	**37.61/0.9735**

The results of the best performance are denoted in bold

the plain CNN model (Dong et al. 2014) contain blurry boundaries or rainy streaks (Fig. 11d).

Although the proposed DualCNN consists of two branches, an alternative is to combine the Net-S and Net-D in a cascaded manner as shown in Fig. 12. In this cascaded model, the first stage estimates the main structure while the second stage estimates details. This network architecture is similar to that of the ResNet (He et al. 2016). However, this cascaded architecture does not generate high-quality results compared to the proposed DualCNN (Fig. 11e and Table 12).

We note that the network based on the cascaded architecture outputs an image in the intermediate layer (Fig. 12). We further evaluate the network based on a cascaded architecture, where the outputs of the intermediate layer are the feature maps. To this end, we set the number of features in the intermediate convolutional layer to be 64. In addition, we concatenate the input image and the output image of the Net-S and use the concatenated result as the input of the Net-D ("Cascade-feature-input" for short in Table 10). We train this baseline using the same settings as the proposed method and evaluate it on image super-resolution. Because the outputs of the intermediate layer are features, we cannot constrain the intermediate features using (2) as in the cascaded model (i.e., "Cascade" in Table 10). Compared to the "Cascade" model, the "Cascade-feature-input" uses additional features and its model parameter is more than that of the "Cascaded" model. Thus, the "Cascade-feature-input" model generates better results as shown in Table 10. However, the performance gains by the "Cascade-feature-input" model are likely due to the use of a larger capacity network model. In contrast, the proposed model generates better results compared to the "Cascade-feature-input" even though the model parameter of the proposed method is fewer than that of "Cascade-feature-input" as shown in Table 10.

We clarify why the proposed network design performs better than the cascaded network designs as follows. For the proposed DualCNN model, the output of Net-D and ground truth images will affect the parameter updating process of Net-S. However, for the cascaded architecture, the parameter updating process of Net-S is not only affected by the output of Net-D and ground truth images but also the intermediate network parameters of Net-D. Thus, the back-propagation process of the cascaded architecture is much longer than that of the DualCNN model. This leads to a more complex optimization process, where small errors in the intermediate layers may significantly affect the final estimation.

5.2 Difference from the Single Branch Network with Two Outputs

As the proposed DualCNN involves two branches to estimate key components for image restoration, it may not be clear whether the proposed model is a special case of the single branch network with two outputs. To answer this question, we compare the proposed method with the single branch network with two outputs on the image super-resolution task using the same experimental settings as the proposed method. Specifically, the single branch splits the output as the structures and details and generates the final output using the structures and details. Table 11 shows the quantitative evaluations on the image super-resolution task, where "SRCNN-Split" denotes the single branch using the SRCNN network and "VDSR-Split" denotes the single branch using the VDSR network. Table 11 shows that using a single branch by splitting it into two branches according to channel dimension does not generate good results. As the details usually correspond to the high-frequency information while the structures correspond to the low-frequency information, it is necessary

Table 12 Quantitative evaluations of different network architectures on the image filtering (Xu et al. 2015) and deraining (Zhang et al. 2020) datasets in terms of PSNR

Different nets	SRCNN	VDSR	Cascade	DualCNN-S
Filtering	32.0	41.6	42.0	**42.1**
Deraining	22.3	23.9	23.5	**24.1**

The results of the best performance are denoted in bold

Table 13 Quantitative evaluations of the proposed dual composition loss function on the proposed image dehazing test dataset in terms of PSNR and SSIM

$(\lambda/\alpha, \gamma/\alpha)$	(0, 0)	(1, 0)	(0, 1)	(9, 9)
Avg. PSNR	24.66	26.65	27.12	**27.28**
Avg. SSIM	0.8979	0.9265	0.9512	**0.9526**

The results of the best performance are denoted in bold

to use different branches to jointly estimate them for image super-resolution and related low-level vision problems. Thus, the proposed DualCNN model is not a special case of one branch network with two outputs and generates better super-resolution results.

In addition, as the commonly used single branch networks, e.g., SRCNN, VDSR, usually adopt ground truth images to constrain the network training, we further evaluate the proposed DualCNN model when the Net-S and the Net-D are both constrained by the ground truth images. Table 11 shows that using GT images to regularize both the Net-S and Net-D does not generate better results. This can be attributed to that if the two branches are both constrained by the ground truth images, the image details may not be learned well thus affecting the final image restoration.

5.3 Effect of the Loss Functions in DualCNN

We evaluate the effect of different loss functions on image dehazing. Table 13 shows that adding two regularization losses $\mathcal{L}_s$ in (2) and $\mathcal{L}_d$ in (3) significantly improves the performance.

5.4 Different Architectures of Two Branches in DualCNN

We use different network structures for two branches of DualCNN in the experiments in Sect. 4. It is of interest to analyze the performance by using the same structures for both branches. To this end, we set the two branches in the DualCNN using the network structures of SRCNN (Dong et al. 2014) (DualCNN2SRCNN for short) or VDSR (Kim et al. 2016a) (DualCNN2VDSR for short) and train the DualCNN according to the same settings used in the image super-resolution experiment. Table 14 shows that the DualCNN2SRCNN method does not generate better results than those by DualCNN2VDSR, which demonstrates that DualCNN with a deeper model generates better results when the architectures of two branches are the same. However, the DualCNN where one branch is SRCNN and the other one is VDSR performs better. In addition, we note that the DualCNN whose one branch is SRCNN and the other one is VDSR has fewer model parameter compared to DualCNN2VDSR. This further demonstrates that the performance gains are not due to the use of the large capacity models.

We quantitatively evaluate the DualCNN when the two branches are the same on image deraining using synthetic rainy dataset (Zhang et al. 2020). Similar to the image super-resolution experimental settings, the two branches in the DualCNN are set to be the network structures of SRCNN (Dong et al. 2014) and the network structures of VDSR (Kim et al. 2016a), respectively. The results in Table 15 also demonstrate that the DualCNN where one branch is SRCNN and the other one is VDSR performs better.

In addition, we note that the proposed DualCNN model can accommodate other CNNs (e.g., Haris et al. 2018; Lim et al. 2017) for image super-resolution. To validate

Table 15 Quantitative evaluations of two branches in DualCNN using the synthetic rainy dataset (Zhang et al. 2020) in terms of PSNR and model parameter

	DualCNN2SRCNN	DualCNN2VDSR	DualCNN-S
Avg. PSNR	22.42	23.58	**24.11**
Model parameter	0.02M	1.32M	0.67M

The results of the best performance are denoted in bold

Table 14 Quantitative evaluations of two branches in DualCNN on image super-resolution (×2) in terms of PSNR, SSIM, and model parameter

Different nets	Set5	Set14	B100	Urban100	Manga109	Model parameter
DualCNN2SRCNN	36.58/0.9528	32.48/0.9050	31.24/0.8855	29.27/0.8909	35.32/0.9649	0.02M
DualCNN2VDSR	**37.74/0.9589**	33.26/0.9131	31.92/0.8958	**31.01/0.9160**	37.47/0.9734	1.32M
DualCNN-S	37.73/**0.9589**	**33.31/0.9132**	**31.93/0.8959**	**31.01**/0.9158	**37.61/0.9735**	0.67M

The results of the best performance are denoted in bold

Table 16 Quantitative evaluations of the flexibility of the proposed DualCNN on the image super-resolution (Set 5, ×2) datasets in terms of PSNR and SSIM. The "EDSR+VDSR" denotes that the proposed DualCNN model uses EDSR as the Net-S and VDSR as Net-D. The "EDSR+16ResBlocks" denotes that the proposed DualCNN model uses EDSR as the Net-S and 16ResBlocks as Net-D

	EDSR	EDSR+VDSR	EDSR+16ResBlocks
Avg. PSNR	38.11	38.21	**38.25**
Model parameter	0.9602	0.9614	**0.9615**

The results of the best performance are denoted in bold

the effectiveness of the proposed method, we use the EDSR method (Lim et al. 2017) as the Net-S (DualCNN (EDSR+VDSR) in Table 16) and train the proposed method according to the same protocols of Lim et al. (2017). Table 16 shows that using the proposed DualCNN model can improve the performance of the EDSR method on image super-resolution, where the PSNR value of the proposed method is at least 0.1dB higher than that of the EDSR method. Moreover, we use the network with 16 residual blocks (Lim et al. 2017) as the Net-D. Table 16 shows the proposed DualCNN model still performs better than the EDSR method, suggesting that the proposed DualCNN model is able to accommodate other CNNs for the performance gain.

5.5 Intermediate Results by the Proposed DualCNN and DualCNN-S

To better understand what the DualCNN and DualCNN-S models can learn, we show the intermediate results by the DualCNN and DualCNN-S on the image super-resolution task. Figure 13b, g show the estimated details and structures by the DualCNN are visually similar to the ground truths. Figure 13c, h show two key components by the DualCNN-S. We note that DualCNN-S is able to learn the pseudo details which help restore high-quality image details. Although there are some differences in the intermediate results by DualCNN and DualCNN-S (e.g., details by the DualCNN and the pseudo ones by the DualCNN-S), the final restored images are almost the same based on the decomposition model (6).

5.6 Convergence Property

We evaluate the convergence properties of our method on the Set5 dataset for super-resolution. Although the proposed network contains two branches, Fig. 14 shows that it has the similar convergence property to the SRCNN (Dong et al. 2014) and VDSR (Kim et al. 2016a).

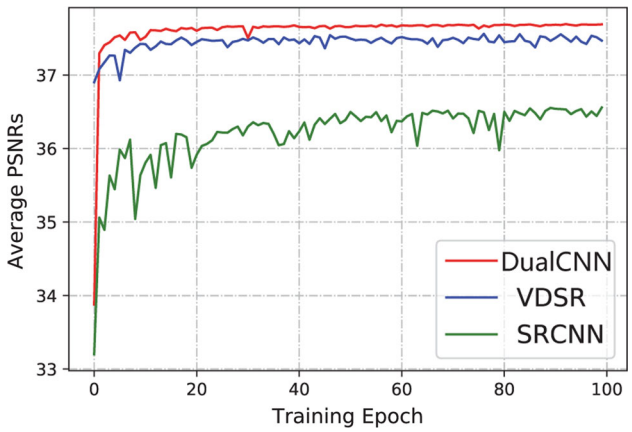


Fig. 14 Quantitative evaluations of the convergence property on the super-resolution dataset (Set5, ×2)

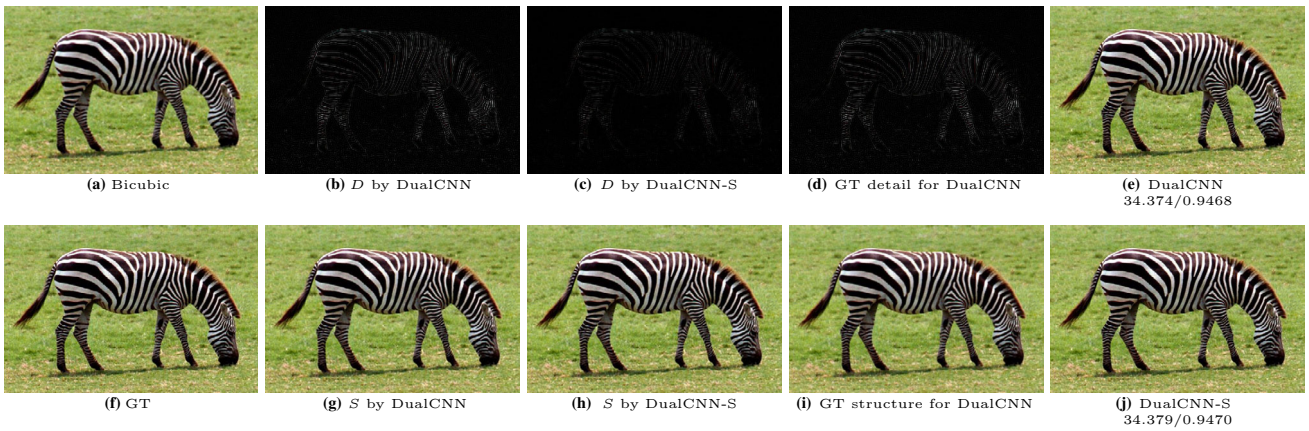


Fig. 13 Intermediate results by the DualCNN and DualCNN-S on the image super-resolution problem (×2). As both the proposed DualCNN and DualCNN-S can estimate two key components from the input images, the generated images are of high quality with finer details (best viewed on high-resolution displays). The pixel values of the details are re-scaled for visualization purpose

6 Concluding Remarks

In this paper, we propose a DualCNN model for low-level vision tasks. The DualCNN extracts both structures and details from inputs with minimum reconstruction errors for a specific task. We analyze the effectiveness of the DualCNN and demonstrate that it is a generic framework and can be effectively and efficiently applied to numerous low-level vision tasks, including image super-resolution, image denoising, image deconvolution, edge-preserving filtering, image deraining, and image dehazing. Experimental results show that the DualCNN model performs favorably against the state-of-the-art methods that have been specially designed for each task.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11263-022-01583-y.

Acknowledgements This work is supported in part by the National Key Research and Development Program of China under Grant 2018AAA0102001, the National Natural Science Foundation of China under Grants 61872421, 61922043, and 61925204, the Fundamental Research Funds for the Central Universities under Grant 30920041109, and NSF CAREER under Grant 1149783.

References

Berman, D., Treibitz, T., & Avidan, S. (2016). Non-local image dehazing. In *CVPR* (pp. 1674–1682).

Bulat, A., Yang, J., & Tzimiropoulos, G. (2018). To learn image super-resolution, use a GAN to learn how to do image degradation first. In *ECCV* (pp. 187–202).

Burger, H. C., Schuler, C. J., & Harmeling, S. (2012). Image denoising: Can plain neural networks compete with bm3d? In *CVPR* (pp. 2392–2399).

Burger, H., Schuler, C., & Harmeling, S. (2012). Image denosing: Can plain neural networks compete with BM3D. In *CVPR*.

Cai, B., Xu, X., Jia, K., Qing, C., & Tao, D. (2016). Dehazenet: An end-to-end system for single image haze removal. *IEEE TIP, 25*(11), 5187–5198.

Chen, D., & Davies, M. E. (2020). Deep decomposition learning for inverse imaging problems. In *ECCV* (pp. 510–526).

Chen, Y. L., & Hsu, C. T. (2013). A generalized low-rank appearance model for spatio-temporally correlated rain streaks. In *ICCV* (pp. 1968–1975).

Chen, Q., Xu, J., & Koltun, V. (2017). Fast image processing with fully-convolutional networks. In *ICCV* (pp. 2516–2525).

Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. O. (2007). Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE TIP, 16*(8), 2080–2095.

Dong, C., Deng, Y., Loy, C. C., & Tang, X. (2015). Compression artifacts reduction by a deep convolutional network. In *ICCV* (pp. 576–584).

Dong, C., Loy, C. C., & Tang, X. (2016). Accelerating the super-resolution convolutional neural network. In *ECCV* (pp. 391–407).

Dong, C., Loy, C. C., He, K., & Tang, X. (2014). Learning a deep convolutional network for image super-resolution. In *ECCV* (pp. 184–199).

Dong, C., Loy, C. C., He, K., & Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE TPAMI, 38*(2), 295–307.

Dong, J., Roth, S., & Schiele, B. (2021). DWDN: Deep wiener deconvolution network for non-blind image deblurring. *IEEE TPAMI, 52,* 1. https://doi.org/10.1109/TPAMI.2021.3138787.

Eigen, D., Krishnan, D., & Fergus, R. (2013). Restoring an image taken through a window covered with dirt or rain. In *ICCV* (pp. 633–640).

Fan, Q., Chen, D., Yuan, L., Hua, G., Yu, N., & Chen, B. (2018a). Decouple learning for parameterized image operators. In *ECCV* (pp. 455–471).

Fan, Q., Yang, J., Wipf, D. P., Chen, B., & Tong, X. (2018b). Image smoothing via unsupervised learning. *ACM TOG, 37*(6), 259:1-259:14.

Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., & Paisley, J. (2017). Removing rain from single images via a deep detail network. In *CVPR* (pp. 3855–3863).

Fu, X., Huang, J., Ding, X., Liao, Y., & Paisley, J. (2017). Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing, 26*(6), 2944–2956.

Girshick, R. B. (2015). Fast R-CNN. In *ICCV* (pp. 1440–1448).

Guo, T., Li, X., Cherukuri, V., & Monga, V. (2019). Dense scene information estimation network for dehazing. In *CVPR workshops* (pp. 2122–2130).

Haris, M., Shakhnarovich, G., & Ukita, N. (2018). Deep back-projection networks for super-resolution. In *CVPR* (pp. 1664–1673).

He, K., Sun, J., & Tang, X. (2009). Single image haze removal using dark channel prior. In *CVPR* (pp. 1956–1963).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR* (pp. 770–778).

Huang, J. B., Singh, A., & Ahuja, N. (2015). Single image super-resolution from transformed self-exemplars. In *CVPR* (pp. 5197–5206).

Isobe, T., Jia, X., Gu, S., Li, S., Wang, S., & Tian, Q. (2020). Video super-resolution with recurrent structure-detail network. In *ECCV* (pp. 645–660).

Jain, V., & Seung, H. S. (2008). Natural image denoising with convolutional networks. In *NIPS* (pp. 769–776).

Kang, L. W., Lin, C. W., & Fu, Y. H. (2012). Automatic single-image-based rain streaks removal via image decomposition. *IEEE TIP, 21*(4), 1742–1755.

Kim, J., Lee, J. K., & Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. In *CVPR* (pp. 1646–1654).

Kim, J., Lee, J. K., & Lee, K. M. (2016). Deeply-recursive convolutional network for image super-resolution. In *CVPR* (pp. 1637–1645).

Krishnan, D., & Fergus, R. (2009). Fast image deconvolution using hyper-Laplacian priors. In *NIPS* (pp. 1033–1041).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS* (pp. 1106–1114).

Lai, W. S., Huang, J. B., Ahuja, N., & Yang, M. H. (2019). Fast and accurate image super-resolution with deep Laplacian pyramid networks. *IEEE TPAMI, 41*(11), 2599–2613.

Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., & Shi, W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR* (pp. 4681–4690).

Levin, A., Weiss, Y., Durand, F., & Freeman, W. T. (2009). Understanding and evaluating blind deconvolution algorithms. In *CVPR* (pp. 1964–1971).

Levin, A., Fergus, R., Durand, F., & Freeman, W. T. (2007). Image and depth from a conventional camera with a coded aperture. *ACM TOG, 26*(3), 70.

Li, R., Pan, J., Li, Z., & Tang, J. (2018). Single image dehazing via conditional generative adversarial network. In *CVPR* (pp. 8202–8211).
Li, B., Peng, X., Wang, Z., Xu, J., & Feng, D. (2017). Aod-net: all-in-one dehazing network. In *ICCV* (pp. 4780–4788).
Li, Y., Tan, R.T., Guo, X., Lu, J., & Brown, M. S. (2016). Rain streak removal using layer priors. In *CVPR* (pp. 2736–2744).
Liao, R., Tao, X., Li, R., Ma, Z., & Jia, J. (2015). Video super-resolution via deep draft-ensemble learning. In *ICCV* (pp. 531–539).
Lim, B., Son, S., Kim, H., Nah, S., & Lee, K. M. (2017). Enhanced deep residual networks for single image super-resolution. In *CVPR workshop* (pp. 1132–1140).
Lin, T. Y., Roy Chowdhury, A., & Maji, S. (2015). Bilinear CNN models for fine-grained visual recognition. In *ICCV* (pp. 1449–1457).
Li, S., Ren, W., Zhang, J., Yu, J., & Guo, X. (2019). Single image rain removal via a deep decomposition-composition network. *Computer Vision Image Understanding, 186,* 48–57.
Liu, S., Pan, J., & Yang, M. H. (2016). Learning recursive filters for low-level vision via a hybrid neural network. In *ECCV* (pp. 560–576).
Martin, D. R., Fowlkes, C. C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV* (pp. 416–425).
Meng, G., Wang, Y., Duan, J., Xiang, S., & Pan, C. (2013). Efficient image dehazing with boundary constraint and contextual regularization. In *ICCV* (pp. 617–624).
Pan, J., Liu, S., Sun, D., Zhang, J., Liu, Y., Ren, J., Li, Z., Tang, J., Lu, H., Tai, Y. W., & Yang, M. H. (2018). Learning dual convolutional neural networks for low-level vision. In *CVPR* (pp. 3070–3079).
Pan, J., Dong, J., Liu, Y., Zhang, J., Ren, J. S. J., Tang, J., et al. (2021). Physics-based generative adversarial models for image restoration and beyond. *IEEE TPAMI, 43*(7), 2449–2462.
Pan, J., Sun, D., Pfister, H., & Yang, M. (2018). Deblurring images via dark channel prior. *IEEE TPAMI, 40*(10), 2315–2328.
Qian, R., Tan, R. T., Yang, W., Su, J., & Liu, J. (2018). Attentive generative adversarial network for raindrop removal from a single image. In *CVPR* (pp. 2482–2491).
Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., & Yang, M. H. (2016). Single image dehazing via multi-scale convolutional neural networks. In *ECCV* (pp. 154–169).
Ren, J. S. J., Xu, L., Yan, Q., & Sun, W. (2015). Shepard convolutional neural networks. In *NIPS* (pp. 901–909).
Saxena, A., Sun, M., & Ng, A. Y. (2009). Make3d: Learning 3d scene structure from a single still image. *IEEE TPAMI, 31*(5), 824–840.
Schmidt, U., & Roth, S. (2014). Shrinkage fields for effective image restoration. In *CVPR* (pp. 2774–2781).
Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., & Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR* (pp. 1874–1883).
Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor segmentation and support inference from RGBD images. In *ECCV* (pp. 746–760).
Singh, V., Ramnath, K., & Mittal, A. (2020). Refining high-frequencies for sharper super-resolution and deblurring. *Computer Vision Image Understanding, 199,* 103034.
Sun, Y., Chen, Y., Wang, X., & Tang, X. (2014). Deep learning face representation by joint identification-verification. In *NIPS* (pp. 1988–1996).
Tarel, J., Hautière, N., Caraffa, L., Cord, A., Halmaoui, H., & Gruyer, D. (2012). Vision enhancement in homogeneous and heterogeneous fog. *IEEE Intelligent Transportation Systems Magazine, 4*(2), 6–20.
Tian, C., Xu, Y., Zuo, W., Du, B., Lin, C. W., & Zhang, D. (2020). Designing and training of A dual CNN for image denoising. CoRR arXiv:2007.03951
Timofte, R., Smet, V. D., & Gool, L. J. V. (2014). A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV* (pp. 111–126).
Xie, J., Xu, L., & Chen, E. (2012). Image denoising and inpainting with deep neural networks. In *NIPS* (pp. 350–358).
Xu, L., Ren, J. S. J., Liu, C., & Jia, J. (2014). Deep convolutional neural network for image deconvolution. In *NIPS* (pp. 1790–1798).
Xu, L., Ren, J.S.J., Yan, Q., Liao, R., & Jia, J. (2015). Deep edge-aware filters. In *ICML* (pp. 1669–1678).
Xu, L., Lu, C., Xu, Y., & Jia, J. (2011). Image smoothing via L_0 gradient minimization. *ACM TOG, 30*(6), 174:1-174:12.
Xu, L., Yan, Q., Xia, Y., & Jia, J. (2012). Structure extraction from texture via relative total variation. *ACM TOG, 31*(6), 139:1-139:10.
Yang, H., Pan, J., Yan, Q., Sun, W., Ren, J. S. J., & Tai, Y. W. (2017). Image dehazing using bilinear composition loss function. CoRR arXiv:1710.00279
Yang, A., Wang, H., Ji, Z., Pang, Y., & Shao, L. (2019). Dual-path in dual-path network for single image dehazing. In *IJCAI* (pp. 4627–4634).
Zhang, H., & Patel, V. M. (2018). Densely connected pyramid dehazing network. In *CVPR* (pp. 3194–3203).
Zhang, H., & Patel, V. M. (2018). Density-aware single image de-raining using a multi-stream dense network. In *CVPR* (pp. 695–704).
Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., & Fu, Y. (2018). Image super-resolution using very deep residual channel attention networks. In *ECCV* (pp. 294–310).
Zhang, J., Pan, J., Lai, W. S., Lau, R. W. H., & Yang, M. H. (2017). Learning fully convolutional networks for iterative non-blind deconvolution. In *CVPR* (pp. 6969–6977).
Zhang, Y., Tian, Y., Kong, Y., Zhong, B., & Fu, Y. (2018). Residual dense network for image super-resolution. In *CVPR* (pp. 2472–2481).
Zhang, Q., Xu, L., & Jia, J. (2014). 100+ times faster weighted median filter (WMF). In *CVPR* (pp. 2830–2837).
Zhang, K., Zuo, W., Gu, S., & Zhang, L. (2017). Learning deep CNN denoiser prior for image restoration. In *CVPR* (pp. 2808–2817).
Zhang, H., Sindagi, V., & Patel, V. M. (2020). Image de-raining using a conditional generative adversarial network. *IEEE TCSVT, 30*(11), 3943–3956.
Zhang, K., Zuo, W., Chen, Y., Meng, D., & Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE TIP, 26*(7), 3142–3155.
Zhu, H., Peng, X., Chandrasekhar, V., Li, L., & Lim, J. H. (2018). Dehazegan: When image dehazing meets differential programming. In *IJCAI* (pp. 1234–1240).
Zhu, H., Cheng, Y., Peng, X., Zhou, J. T., Kang, Z., Lu, S., et al. (2021). Single-image dehazing via compositional adversarial network. *IEEE Transactions on Cybernetics, 51*(2), 829–838.
Zoran, D., & Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In *ICCV* (pp. 479–486).