



# Synthetic images as a regularity prior for image restoration neural networks

Raphaël Achddou, Yann Gousseau, Saïd Ladjal

## ► To cite this version:

Raphaël Achddou, Yann Gousseau, Saïd Ladjal. Synthetic images as a regularity prior for image restoration neural networks. Eighth International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)., May 2021, Cabourg (virtuel), France. hal-03186499

HAL Id: hal-03186499

<https://hal.science/hal-03186499>

Submitted on 31 Mar 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Synthetic images as a regularity prior for image restoration neural networks

Raphaël Achddou<sup>1</sup>, Yann Gousseau<sup>1</sup>, and Saïd Ladjal<sup>1</sup>

LTCI, Telecom Paris, Institut Polytechnique de Paris

**Abstract.** Deep neural networks have recently surpassed other image restoration methods which rely on hand-crafted priors. However, such networks usually require large databases and need to be retrained for each new modality. In this paper, we show that we can reach near-optimal performances by training them on a synthetic dataset made of realizations of a dead leaves model, both for image denoising and super-resolution. The simplicity of this model makes it possible to create large databases with only a few parameters. We also show that training a network with a mix of natural and synthetic images does not affect results on natural images while improving the results on dead leaves images, which are classically used for evaluating the preservation of textures. We thoroughly describe the image model and its implementation, before giving experimental results.

**Keywords:** Image restoration · Deep learning · Natural image models.

## 1 Introduction

A key ingredient of image restoration methods is an *a priori* hypothesis about image regularity. Methods based on total variation regularization terms [24] assume a Laplacian distribution for the image gradient. Methods involving wavelet shrinkage [11] are optimal in some Besov spaces. Non-local methods [4] rely on an auto-similarity hypothesis. More recently, deep neural networks have achieved impressive results in all fields of image restoration: denoising, single image super-resolution, deconvolution, etc. Although these models do not usually incorporate image regularity assumptions, it has been shown that networks themselves can be seen as regularization terms [27]. Instead of hand-crafted mathematical priors, these models necessitate a training on voluminous databases and necessitate to be retrained for each new modality or specific imaging device [8]. In this contribution, we show that these models can be efficiently trained from synthetic image databases based on a mathematical model grounded in physical priors and depending on few parameters. To the best of our knowledge, this is the first such result for networks aimed at restoring natural images.

Among available mathematical models for natural images, we show that an occlusion-based dead leaves model, equipped with a scaling size distribution, is sufficient to reach near state-of-the-art restoration performances, for both tasks of denoising and single image super-resolution. Moreover, we show that this

model can be efficiently combined with natural image databases to enhance the capacity of deep neural networks to preserve details, without impairing their classical performance evaluation.

We believe that such a study both sheds light on the way convolutional neural networks can address restoration problems and opens interesting perspectives. First, this result shows that the mere structure of such networks is adapted to image restoration tasks and that despite their huge number of parameters they can be made near-optimal from just a few principles and hyper-parameters. This result, and the fact that simpler, less structured models cannot achieve satisfying restoration performance, also highlights the type of geometric structures a neural networks needs to be efficiently trained. Second, the proposed learning database has the potential to be modified according to specific acquisition devices and in particular to their point spread function, dynamic range, noise modality, etc. This opens the way to flexible, generic and relatively light learning schemes.

## 2 Related works

**Image regularity and restoration priors.** A classical principled approach to image restauration is to assume some closed form statistical prior on the images distribution. Given an observation, one seeks the best explanation according to the model. Depending on the method the "best" may be the maximum a posteriori or the risk minimiser. Among the models that fall in this general framework let us cite the Wiener model, for which the distribution of images is assumed to be gaussian and translation invariant and the total variation [24] for which the log-likelihood is the  $l^1$  norm of the gradient vector field. In [11] the authors derive an algorithm for restoring signals under the assumption that the targeted signals are well approximated by a sparse representation in some wavelet decomposition. Later, a consequent body of literature discussed the implications of the sparsity assumption and a variety of algorithms where proposed to take advantage of this particular form of regularity ([5, 10]). The total variation model itself can be viewed as a form of sparsity.

Other fruitful models do not translate easily into a restoration algorithm. Among these, the autosimilarity assumption assumes that the image possesses repeats of the same pattern or patch [4], [9]. Methods derived from this regularity assumption are algorithms that typically average similar patches in order to co-denoise those that are likely to represent the same ground truth.

On the other hand deep learning methods for restoration seek directly to build a machinery (the trained network) that minimises the reconstruction error. Typically, the loss function is taken to be the mean square error between the perfect image and the output of the network. Here the prior on images is represented by the training dataset which is believed to convey sufficient information about the distribution of images([30, 31]). The effort goes into the careful design of the network and its training.

In turn, a trained denoising network can be used as an implicit prior on the image distribution. This idea, named plug and play, consists in using the de-

noising network in place of a proximal operator during an iterative optimisation of a variational model [21]. Another approach to using a network as a prior is presented in [27] in which it is showed that the architecture of a network can serve as a regulariser.

**Synthesis models.** Some statistical priors on natural images have been turned into generative models that have the ability to synthesize images, mostly for the task of texture synthesis : Gaussian models [12], wavelet-based models [15, 22], Markov fields, dead leaves model [2, 18], neural networks [13]. In this work, we investigate the use of such generative models as a way to train restoration neural networks. We turn to the dead leaves model, because of its simplicity, limited number of parameters and ability to generate complex images with details at all scales.

Let us mention that the use of synthetic models to train neural networks is not new and has been extensively used for image analysis tasks such as segmentation, recognition, or detection, see e.g. [26]). To the best of our knowledge, no such strategy has been used for restoration tasks.

### 3 Dead leaves image generation

#### 3.1 The continuous dead leaves model

Informally, the dead leaves model is a random field obtained as the sequential superimposition of random shapes. It is defined (see [3]) from a set of random positions, times and shapes  $\{(x_i, t_i, X_i)\}_{i \in \mathbb{N}}$ , where  $\mathcal{P} = \sum \delta_{x_i, t_i}$  is a homogeneous Poisson process on  $\mathbb{R}^2 \times (-\infty, 0]$  and the  $X_i$  are random sets of  $\mathbb{R}^2$  that are independent of  $\mathcal{P}$ . The sets  $x_i + X_i$  are called *leaves* and for each  $i$ , the *visible part* of the leaf is defined as

$$V_i = (x_i + X_i) \setminus \bigcup_{t_j \in (t_i, 0)} (x_j + X_j),$$

that is, the visible part of the leaf  $(x_i, t_i, X_i)$  is obtained by removing from this leaf all leaves that are indexed by a time greater than  $t_i$  (that falls after it). The dead leaves model is then defined as the collection of all visible parts. A random image can be obtained by assigning a random gray level (or color) to each visible part. An example of a dead leaves model where the leaves are disks with a constant radius can be seen in Figure 2a.

A particular type of dead leaves model, where the leaves have a size with scaling properties, has been shown to reproduce many statistical properties of natural images [2, 18]. Such models are obtained by considering random leaves  $R.X$ , where  $X$  is a given shape and  $R$  is a real random variable with density  $f(r) = C.r^{-\alpha}$ , with  $C$  a normalizing constant. The case  $\alpha = 3$  corresponds to a scale invariant model [18]. In order for such models to be well defined, values of  $R$  have to be restricted to values in  $(r_{min}, r_{max})$ , see [14]. The resulting model therefore depends on 3 parameters:  $r_{min}$ ,  $r_{max}$  and  $\alpha$ . This model is especially appealing for natural images, because it incorporates two of their

most fundamental property, non Gaussianity (as a result of edges) and scaling properties [20], in a very simple setting. Because this model contains details and edges at all scales, potentially of arbitrary contrast, it has been proposed as a tool for the evaluation of the ability of imaging devices to respect textures [6, 7] and was recently retained as a standard for quality evaluation.

### 3.2 Implementation and discussion on the parameters

---

**Algorithm 1:** Dead leaves image generation algorithm

---

```

Parameters: ( $r_{min}, r_{max}, \alpha, w$ ), color_image
Output : X
mask = ones( $w, w$ );
X = zeros( $w, w, 3$ );
while ||mask|| > 0 do
    tmp =  $r_{max}^{1-\alpha} + (r_{min}^{1-\alpha} - r_{max}^{1-\alpha}) \times \text{random}()$ ;
    r =  $\text{tmp}^{-\frac{1}{\alpha-1}}$ ;
    x,y = randint(0,w), randint(0,w);
    color = color_image(randint(0,w), randint(0,w));
    new_disk = disk(r, color);
    X = update_image(X, x, y, new_disk);
    mask = update_mask(mask, r, x, y);
end
X = downscale(X,5).
```

---

We now detail how to generate digital versions of the dead leaves model, following the procedure summarized in Algorithm 1. At each step, a random discrete disk of radius  $r$  and center  $(x, y)$  is generated as the set of discrete positions satisfying the corresponding disk equation. Centers are uniformly distributed in the image domain and radiiuses are distributed according to a power law density with exponent  $\alpha$ , as discussed in the previous paragraph. Radiiuses are limited between  $r_{min}$  and  $r_{max}$ . To generate the image, we rely on a *perfect simulation* technique [17] and sequentially put the disks *below* the previously drawn disks, until the image domain has been fully covered. That is, at each step, pixels which have not been colored yet are given the color of the disk added at this step. The choice of the disk color will be shortly discussed. The used definition of discrete disk is crude and in particular does not include any anti-aliasing scheme. Therefore we first generate a large image that is then down-sampled by a factor 5 after convolution with a Gaussian filter with  $\sigma = 5/3$ . This step is a critical component of our algorithm. It allows for sub-pixel sized objects and for more natural boundaries. In Figure 1a, we display a full size (2000,2000) dead leaves image before down-scaling. A (20,20) crop on that same image (see Fig. 1b) exhibit very sharp boundaries and piecewise constant zones. A (20,20) crop on the down-scaled image has a much more realistic aspect (see Fig.1b). The whole procedure can be seen as a very simple simulation of the camera acquisition of a dead leaves model with tiny objects.

**Color sampling.** Colors of disks are picked randomly from the color histogram of a given natural image (different for each image generation). Another simpler

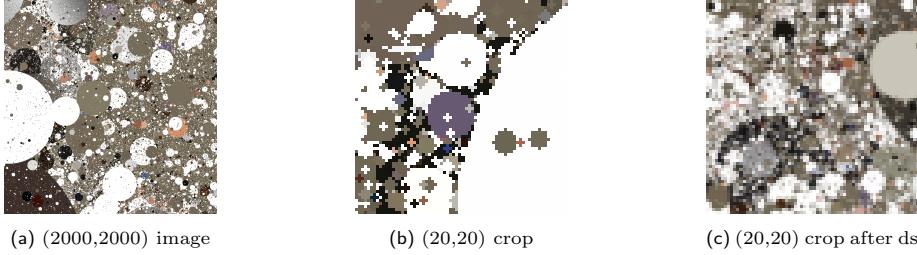


Fig. 1: Visual example of the downscaling step (ds)

option is to consider colors that are uniformly sampled from the RGB cube, but this lead to unnatural colors and no spatial color coherency, as well, as we shall see, to a loss in performance. We can see in Figure 2 that Fig. 2c,2d have more realistic colors than Fig. 2b.

$r_{min}, r_{max}, \alpha$ . Those parameters control the distribution of the disks size in the image. Images generated with different parameters can be seen in Figure 2. As we can see, at fixed  $\alpha = 3$  and  $r_{max} = 2000$ , the type of image strongly depends on the value of  $r_{min}$ . A large  $r_{min}$  yields images with clear edges and homogeneous zones, whereas smaller  $r_{min}$  yields micro-textures (see Fig. 2e,2f). Similar observations can be made when varying  $\alpha$  (see 2g,2h). In the rest of the paper, we chose to keep  $\alpha = 3.0$  (the scale invariant case) and to vary  $r_{min}$ .

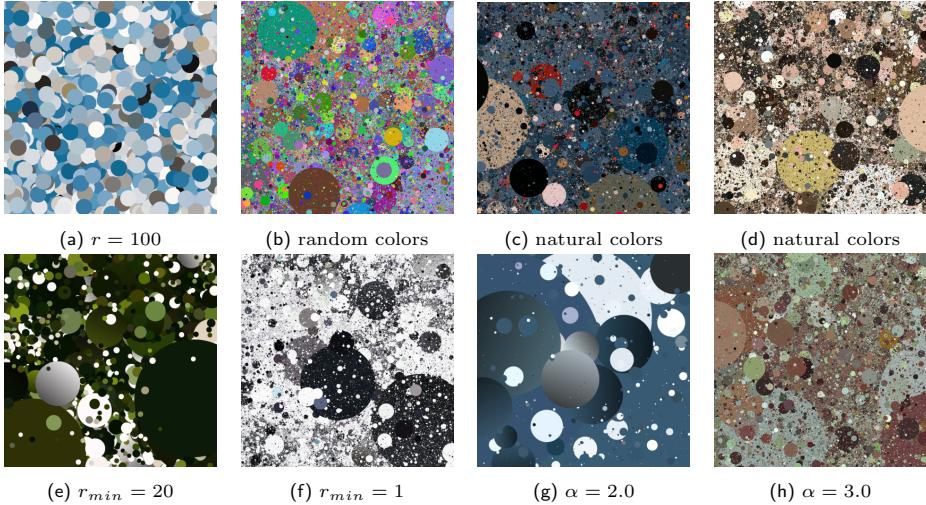


Fig. 2: Dead leaves images generated with different parameters.

## 4 Experimental results

In this experimental section, we first introduce the synthetic image dataset we consider. We then analyze and compare, numerically and qualitatively, the per-

formance obtained when using only synthetic images, only natural images or a mix of both. To assess the relevance of some important features of our generation algorithm, we perform an ablation study in which we train FFDNet on a dataset of dead leaves images generated without particular components of the synthesis method described in Section 3.2. Finally, we illustrate the versatility of the proposed dataset by training the super-resolution network RDN [32].

**Dead leaves dataset.** In order to account for both homogenous areas and micro-textures, we build a dataset made of images generated with either  $r_{min} = 1$  or  $r_{min} = 16$ , in both cases combined with parameters  $\alpha = 3.0$  and  $r_{max} = 2000$ . Micro-textures being harder to restore than homogeneous areas, we chose to have a 2 to 1 ratio between the two possible  $r_{min}$  values. The color distribution of the disks is given by the histograms of the natural images from the Waterloo database [19]. As shown previously, this leads to a more coherent color distribution than randomly sampling the RGB cube. Finally, we decided to apply a Gaussian blur to a 10th of the dataset, with a standard deviation uniformly sampled between 1 and 3. Indeed, most natural images tend to contain blurry zones due to the depth-of-field of cameras. By adding a very simple blur model to some of the images of the dataset, we expect blurry areas in natural images to be better restored.

#### 4.1 Denoising results

In order to assess the capacity of the proposed synthetic dataset to successfully train a denoising network, we consider the network FFDNet. It is a state-of-the-art image denoising CNN, which was introduced by Zhang et al. [31] and thoroughly examined in [25]. Its main specificity relies in the first layer of the network : to increase the receptive field and to handle a wide range of noise levels, the image is divided in four sub-images which are concatenated to a noise map indicating the local noise standard deviation. This tensor is then passed through a more classic network of batch normalized convolutional layers , with an architecture similar to that of DNCNN’s [30]. It then outputs the four denoised sub-images, which are reassembled to create the final denoised image.

**FFDNet results.** To compare different trainings fairly, we use the same optimization algorithm for all trainings. It consists of 80 epochs with the Adam optimizer and the L2 loss, starting with a  $10^{-3}$  learning rate. There is a decay of factor 10 at epoch 50, and another decay of factor 100 at epoch 60. For each training, we used 350k (50, 50, 3) patches, extracted from either the dead leaves dataset, or the natural image dataset, or a mix of both. The mixed dataset contains  $\frac{1}{3}$  dead leaves images, and  $\frac{2}{3}$  natural images. To show that scaling properties are needed to model natural images, we also trained FFDNet on dead leaves images generated from disks with a fixed radius of 100. In addition, we also consider two alternative training schemes from datasets of synthetic images : white noise images and Gaussian random fields [12]. Numerical evaluation is performed on 2 test sets of natural images (CBSD68, Kodak24) and one set of 24 dead leaves images, generated from the colors of Kodak24. For each test,

we compute the average PSNR, SSIM [28] and PieAPP metric [23], a recent perceptual metric based on human annotation, which tends to fit very well with human perception.

Table 1: Numerical comparisons of the different trainings of FFDNet. We evaluated the results on two benchmark datasets for image denoising (CBSD68 and Kodak24), and our dead leaves testset, at two noise levels. Each cell contains the triplet PSNR/SSIM/PieAPP. The best results are in **blue**, the second in **red**.

$\sigma$	Dataset	CBSD68	Kodak24	Dead leaves testset
$\sigma = 25$	White Noise	19.52/0.416/2.386	19.68/0.365/2.502	20.36/0.607/2.043
	Gaussian field	29.63/0.845/1.402	30.24/0.835/1.471	26.23/0.826/1.254
	DL $r = 100$	29.56/0.820/1.218	30.49/0.819/1.024	26.13/0.799/1.263
	Dead leaves	30.58/0.867/0.711	31.27/0.859/0.739	<b>27.46/0.865/0.573</b>
	Mix	<b>31.07/0.881/0.639</b>	<b>31.98/0.876/0.603</b>	<b>27.33/0.860/0.567</b>
	Natural Images	<b>31.09/0.882/0.629</b>	<b>32.00/0.878/0.599</b>	27.05/0.851/0.576
$\sigma = 50$	White Noise	15.58/0.247/4.682	15.71/0.209/4.785	16.24/0.387/2.932
	Gaussian field	26.68/0.738/2.203	27.41/0.737/2.353	23.31/0.694/2.158
	DL $r = 100$	26.85/0.720/1.563	27.91/0.739/1.314	23.24/0.654/2.005
	Dead leaves	27.40/0.762/1.088	28.21/0.765/1.154	<b>24.21/0.737/1.020</b>
	Mix	<b>27.86/0.782/0.997</b>	<b>28.86/0.789/0.985</b>	<b>24.12/0.732/1.015</b>
	Natural Images	<b>27.87/0.786/0.991</b>	<b>28.89/0.792/0.978</b>	23.90/0.722/1.053

On both natural image testsets and on the dead leaves testset, we observe that the model trained on dead leaves outperforms by a large margin all other models trained on alternative synthetic image datasets (0.9db for the Gaussian model and, without surprise, 11 dB for the white noise model), see Table 1. Visually, the Gaussian field model leads to denoised images still containing noise and grid-like artifacts, which severely impact the PieAPP metric. Observe that for both image models of white noise and Gaussian noise, the optimal solution is known and given by the Wiener filter (multiplication by a constant in the first case and linear filtering in the second). It is interesting to note that the network did not learn to apply this theoretical optimal solution to natural images in either cases. Confirming our intuition that an image model with scaling properties is needed, the dead leaves model with a fixed radius tends to strongly over-smooth the image, thus losing all texture information. This amounts to a loss of 0.65 dB on natural image testsets, and 1.2 dB on dead leaves images.

More surprisingly, the model trained exclusively on dead leaves images performs only 0.6dB lower than the model classically trained on natural images. Visually, the results are still almost as good, despite some limitations. In particular, the synthetically trained model has some difficulties with thin and low contrast lines, and occasionally creates dot artifacts. In other situations, the synthetic training improves the results, as can be seen in Figure 3, where the texture of the rusty artwork is quite well restored, with a better preservation of fine details than with the model trained on natural images. Another very interesting result is the fact that training on a mix of dead leaves images and natural images does not affect the result of the denoising model on testsets of natural images, the difference in PSNR being less than 0.02dB. Visually, the results are almost identical, with a slight advantage for the mixed trained model on texture areas. On the dead leaves test set, the mixed trained model clearly



Fig. 3: Denoising comparison with different FFDNet trainings. Top row from left to right : clean image, noisy image with  $\sigma = 25$ , model trained on white noise, model trained on Gaussian fields. Bottom row from left to right : model trained on dead leaves images with fixed radius  $r = 100$ , model trained on the dead leaves dataset, model trained on the mixed dataset, model trained on natural images.

outperforms the natural image trained model by 0.25dB. This result suggests that jointly optimizing the response to this kind of mixed datasets has the ability to increase some aspects on which imaging devices are evaluated. Indeed the scaling dead leaves model is classically used to evaluate the ability of imaging devices to preserve texture areas [6, 7] and the corresponding scale-invariant test chart has recently become an ISO standard (ISO/TS 19567-2:2019).

**Ablation study.** To confirm the choices made to build the synthetic dataset, we compare different trainings performed with different parameters or design choices, both visually and numerically.

We first illustrate the impact of  $r_{min}$  on the denoising results. As shown in Figure 4, the smaller the  $r_{min}$ , the better micro-textures are restored. Conversely, they are smoothed when  $r_{min}$  gets larger. On the other hand, homogeneous zones contain artifacts when  $r_{min}$  is too small, and are well restored when  $r_{min}$  is larger. This behaviour is expected since a large  $r_{min}$  leads to dead leaves images with homogeneous zones, and a small  $r_{min}$  to more micro-texture zones. Referring to Table 2, the optimal  $r_{min}$  seems to be between 4 and 8. However, by mixing images generated with  $r_{min} = 1$  and  $r_{min} = 16$ , we get a noticeable improvement in PSNR (0.17dB) and in image quality, as can be seen in Figure 4.

Other important features of our algorithm are : the color distribution, the downscaling step, and the blur. As we can see in Figure 4, when we sample the disks colors uniformly in the RGB cube, the denoised images show many color artifacts. The additive Gaussian noise creates unnatural colors that the network doesn't identify as such, since it has not been trained on images with natural colors. This leads to a performance gap of more than 1 dB in PSNR. The

Table 2: Impact of the parameters and ablation study. In the first 5 columns(DL1 to DL16), we fix the parameters to  $r_{max} = 2000$ ,  $\alpha = 3.0$ , with natural colors and the downscaling step. From column 6 to 8, we keep the same parameters as in the final dataset, but we remove some important features of the generation. The last column corresponds to the final result.

$\sigma$	DL-1	DL-2	DL-4	DL-8	DL-16	Rand. col	No sub	No blur	Final
25	31.03	31.03	31.09	31.07	30.98	29.99	30.79	31.25	<b>31.27</b>
50	27.98	27.96	28.04	28.06	28.05	27.16	27.74	28.20	<b>28.21</b>

downscaling step is also critical, as it allows sub-pixel sized objects and more natural boundaries. In Figure 4, we can see that the network trained on the dead leaves dataset without subsampling tends to over-smooth texture areas, and to produce stair-casing artifacts. We can also identify some disk-like objects with hard boundaries in the images, creating an unnatural aspect. In terms of PSNR, this amounts to a loss of 0.5 dB compared to the training on our final dead leaves dataset. For the final synthetic dataset, we decided to blur 10 % of images. As we can see in Table 2, removing this step has almost no impact on the PSNR. Nonetheless, we observe in Figure 4 that removing this step makes blurry zones look sharper than they really are. Overall, the final dead leaves dataset yields better results, numerically and visually, than the ablated datasets.



Fig. 4: Visual illustration of the ablation study. From left to right : top line : clean image, noisy image,  $r_{min} = 1$ ,  $r_{min} = 2$ ,  $r_{min} = 4$ , / bottom line :  $r_{min} = 16$ , No subsampling, Uniform color distribution, No blur, Final result.

#### 4.2 Single-image super-resolution

To assess our dataset’s versatility, we consider the task of single-image super-resolution (SISR). We chose to retrain the Residual Dense Network (RDN)[32], a state-of-the-art super-resolution network. Its architecture is based on residual dense blocks, a combination of dense blocks introduced in [16] and residual connections.

The numerical evaluation shows a similar behaviour to the one observed in image denoising in Section 4.1. The gaps are of 1.2dB and 0.6dB for a super-resolution of scale 2 and 3 respectively. The results on the Set5 and Set14

Table 3: Numerical evaluation of our Super-resolution results. We report the PSNR of RDN trained either on the dead leaves dataset or on the DIV2K dataset [1].

Dataset	Set 5		Set 14	
	$\times 2$	$\times 3$	$\times 2$	$\times 3$
Dead leaves	36.76	33.82	32.93	30.42
Natural Images	38.18	34.71	33.88	30.73

datasets, which are common benchmarks for super-resolution, are given in Table 3. Visually, the super-resolution results are very similar as we can see in Figure 5. However, if we look closely to Fig. 5g and Fig. 5h, we see in Fig. 5g that some small white spots in the top black region are better restored. Conversely, thin lines in yellow regions have a "dotted" aspect in Fig. 5g, which disappears in Fig. 5h. The dead leaves model does not contain any straight and thin lines, making it harder for this model to retrieve them. A lead to tackle this problem would be to complete our database with patches generated from a sinusoidal basis, as was done in [29].

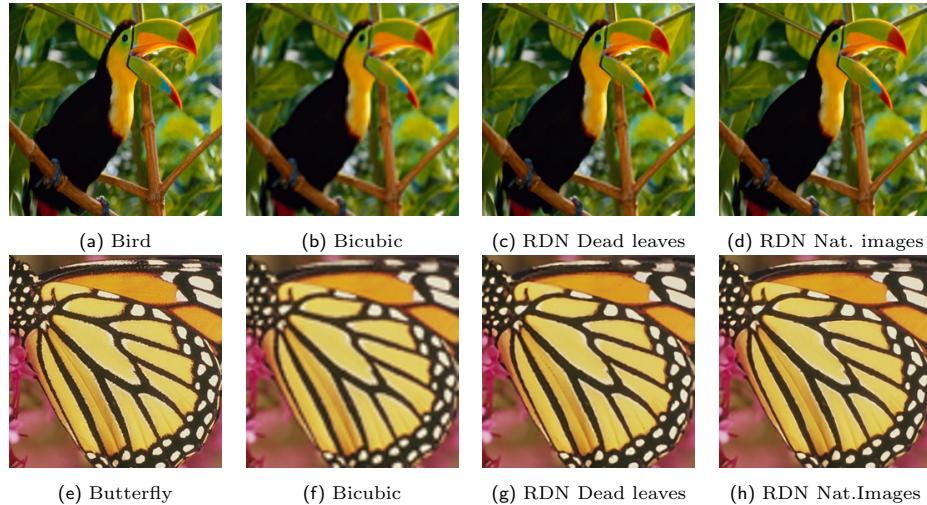


Fig. 5: Visual results of our trainings of RDN at scale 3. From left to right : High resolution image, Bicubic interpolation, RDN trained on dead leaves images, and RDN trained on natural images.

## 5 Conclusion and future works

To the best of our knowledge, this work is the first effort to train an image restoration network on synthetic images. After introducing the dead leaves model and its digital implementation, we carefully studied the role of each component of the image generation method, and their impact on the restoration performances. Both for denoising and super-resolution, models trained on our dead leaves dataset are surprisingly close to those trained on natural images. When mixing natural and synthetic images in the training, the results reach performances on par with the model trained on natural images only. Both results

indicate that the dead leaves model with scaling properties is a good candidate to replace natural images for training, with only a few parameters. Indeed, the geometry of the model only depends on three parameters :  $\alpha, r_{min}, r_{max}$ . Even though the color parameters are still relatively numerous, we plan to investigate simple samplings of the horse-shoe color space in order to avoid unnatural colors. Another perspective would be to complement our dataset with sinusoidal patches to better restore oscillating patches and straight lines. Eventually we believe, as already explained, that a synthetic dataset can be a simple way to avoid retraining new imaging devices with relatively heavy acquisition campaigns [8] and we plan to investigate this ability further.

## References

1. Agustsson, E., Timofte, R.: 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)
2. Alvarez, L., Gousseau, Y., Morel, J.M.: The size of objects in natural and artificial images. In: Advances in Imaging and Electron Physics, vol. 111, pp. 167–242. Elsevier (1999)
3. Bordenave, C., Gousseau, Y., Roueff, F.: The dead leaves model: a general tessellation modeling occlusion. *Advances in applied probability* **38**(1), 31–46 (2006)
4. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation* **4**(2), 490–530 (2005)
5. Candes, E.J., Tao, T.: Decoding by linear programming. *IEEE Transactions on Information Theory* **51**(12), 4203–4215 (2005). <https://doi.org/10.1109/TIT.2005.858979>
6. Cao, F., Guichard, F., Hornung, H.: Measuring texture sharpness of a digital camera. In: Digital Photography V. vol. 7250, p. 72500H. International Society for Optics and Photonics (2009)
7. Cao, F., Guichard, F., Hornung, H.: Dead leaves model for measuring texture quality on a digital camera. In: Digital Photography VI. vol. 7537, p. 75370E. International Society for Optics and Photonics (2010)
8. Chen, C., Chen, Q., Xu, J., Koltun, V.: Learning to see in the dark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3291–3300 (2018)
9. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image restoration by sparse 3D transform-domain collaborative filtering. In: Astola, J.T., Egiazarian, K.O., Dougherty, E.R. (eds.) *Image Processing: Algorithms and Systems VI*. vol. 6812, pp. 62 – 73. International Society for Optics and Photonics, SPIE (2008). <https://doi.org/10.1117/12.766355>, <https://doi.org/10.1117/12.766355>
10. Donoho, D.L.: Compressed sensing. *IEEE Transactions on Information Theory* **52**(4), 1289–1306 (2006). <https://doi.org/10.1109/TIT.2006.871582>
11. Donoho, D.L., Johnstone, I.M., et al.: Minimax estimation via wavelet shrinkage. *The annals of Statistics* **26**(3), 879–921 (1998)
12. Galerne, B., Gousseau, Y., Morel, J.M.: Micro-texture synthesis by phase randomization. *Image Processing On Line* **1**, 213–237 (2011)
13. Gatys, L.A., Ecker, A.S., Bethge, M.: Texture synthesis using convolutional neural networks. arXiv preprint arXiv:1505.07376 (2015)
14. Gousseau, Y., Roueff, F.: Modeling occlusion and scaling in natural images. *Multiscale Modeling & Simulation* **6**(1), 105–134 (2007)

15. Heeger, D.J., Bergen, J.R.: Pyramid-based texture analysis/synthesis. In: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. pp. 229–238 (1995)
16. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
17. Kendall, W.S., Thöennes, E.: Perfect simulation in stochastic geometry. *Pattern Recognition* **32**(9), 1569–1586 (1999)
18. Lee, A.B., Mumford, D., Huang, J.: Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *International Journal of Computer Vision* **41**(1-2), 35–59 (2001)
19. Ma, K., Duanmu, Z., Wu, Q., Wang, Z., Yong, H., Li, H., Zhang, L.: Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing* **26**(2), 1004–1016 (2016)
20. Mumford, D., Gidas, B.: Stochastic models for generic images. *Quarterly of applied mathematics* **59**(1), 85–111 (2001)
21. Ono, S.: Primal-dual plug-and-play image restoration. *IEEE Signal Processing Letters* **24**(8), 1108–1112 (2017). <https://doi.org/10.1109/LSP.2017.2710233>
22. Portilla, J., Simoncelli, E.P.: A parametric texture model based on joint statistics of complex wavelet coefficients. *International journal of computer vision* **40**(1), 49–70 (2000)
23. Prashnani, E., Cai, H., Mostofi, Y., Sen, P.: Pieapp: Perceptual image-error assessment through pairwise preference. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1808–1817 (2018)
24. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena* **60**(1-4), 259–268 (1992)
25. Tassano, M., Delon, J., Veit, T.: An analysis and implementation of the ffdnet image denoising method. *Image Processing On Line* **9**, 1–25 (2019)
26. Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., Birchfield, S.: Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 969–977 (2018)
27. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9446–9454 (2018)
28. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004)
29. Yu, G., Sapiro, G., Mallat, S.: Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing* **21**(5), 2481–2499 (2011)
30. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing* **26**(7), 3142–3155 (2017)
31. Zhang, K., Zuo, W., Zhang, L.: Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing* **27**(9), 4608–4622 (2018)
32. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2472–2481 (2018)