Self-Supervised Denoising Transformer with Gaussian Process

Rajeev Yasarla Jeya Maria Jose Valanarasu Vishwanath S Vishal M. Patel*
Johns Hopkins University
Department of Electrical and Computer Engineering, Baltimore, MD 21218, USA
{ryasarl1, jvalana1, vpatel36}@jhu.edu

Abstract

Convolutional neural network (CNN) based methods have been the main focus of recent developments for image denoising. However, these methods lack majorly in two ways: 1) They require a large amount of labeled data to perform well. 2) They do not have a good global understanding due to convolutional inductive biases. Recent emergence of Transformers and self-supervised learning methods have focused on tackling these issues. In this work, we address both these issues for image denoising and propose a new method: Self-Supervised denoising Transformer (SST-GP) with Gaussian Process. Our novelties are two fold: First, we propose a new way of doing self-supervision by incorporating Gaussian Processes (GP). Given a noisy image, we generate multiple noisy down-sampled images with random cyclic shifts. Using GP, we formulate a joint Gaussian distribution between these down-sampled images and learn the relation between their corresponding denoising function mappings to predict the pseudo-Ground truth (pseudo-GT) for each of the down-sampled images. This enables the network to learn noise present in the down-sampled images and achieve better denoising performance by using the joint relationship between down-sampled images with help of GP. Second, we propose a new transformer architecture - Denoising Transformer (Den-T) which is tailor-made for denoising application. Den-T has two transformer encoder branches - one which focuses on extracting fine context details and another to extract coarse context details. This helps Den-T to attend to both local and global information to effectively denoise the image. Finally, we train Den-T using the proposed self-supervised strategy using GP and achieve a better performance over recent unsupervised/self-supervised denoising approaches when validated on various denoising datasets like Kodak, BSD, Set-14 and SIDD.

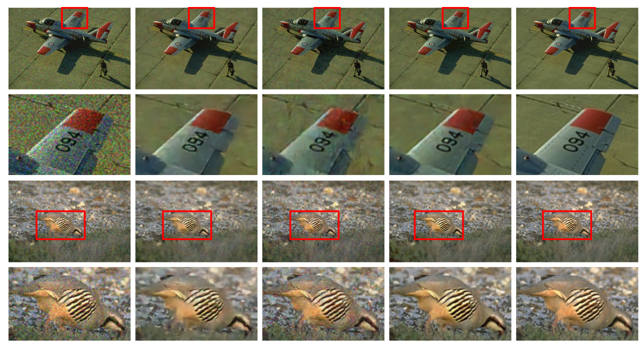

Figure 1. **Visual Quality comparison.** Rows 1-2: Comparisons on noisy image with Poisson noise $\sigma = 30$. Rows 3-4: Comparisons on noisy image with Gaussian noise $\sigma = 25$. Red box corresponds to the zoomed-in region. Our method (SST-GP) achieves better performance than recent methods.

1. Introduction

Noise adversely affects the visual quality of images captured by camera sensor and thus has a detrimental impact on the performance of downstream computer vision tasks like classification, detection and segmentation. Hence, image denoising is an important pre-processing task in many computer vision applications. Denoising is classically formulated as follows: Given a noisy image y, which is a corrupted version of the clean image x with known or unknown noise distribution n, the goal of denoising is to recover the clean image x from y.

Denoising has been extensively studied in the literature because of its importance in several applications. Some of the early methods like BM3D [11], WNNM [16], *etc.* do not require clean ground-truth images. These traditional approaches are computationally efficient, do not involve any learning, and are based on natural image priors. However, they require knowledge of the noise levels making it difficult to use them in the wild. Emergence of CNNs in addressing image denoising significantly improved the quality of restored images. Many CNN methods like, RED30 [33], U-Net [39], DnCNN [55], MemNet [42], N3Net [36], and NLRN [28] address image denoising in a supervised fash-

*This work was supported by NSF CAREER award 2045489.

ion. Since these are data driven approaches, they need large amounts of paired noisy-clean images to train the network.

The In-camera Signal Processing (ISP) pipeline in modern sensors are complicated which makes the noise in the real-world difficult to model. This makes it really hard and expensive to obtain labeled pairs of noisy and corresponding ground-truth images which are essential for supervised learning based methods. Hence, most of the existing fully-supervised approaches [18,28,36,55] synthetically generate the noisy images and train their network on these synthetic data pairs. However, as discussed in [20, 21], when these fully-supervised methods are tested on real-world noisy images, they tend to perform poorly because of the domain gap between synthetic and real world noise.

To overcome this problem, especially in cases where we do not have access to real-world ground-truth, Lehtinen *et al.* [25] applied statistical reasoning in signal reconstruction to CNNs to perform denoising. They demonstrate that it is possible to learn to restore images by using only the corrupted examples. However, they require multiple independent noisy observation of a scene to train the network. This requirement is not practical, since capturing multiple observations of the same scenes is quite challenging when there are movements in the scene. Subsequently, approaches like [21, 23, 48] were developed using a blind-spot network (BSN) structures for learning a self-supervised model. Additionally, [23, 48] employed Gaussian-Poisson noise models to further improve the performance. The main limitation for these methods is that BSN is computationally expensive and suffers from relatively low accuracy. Moran *et al.* [35] proposed a method that uses noiser-noisy pairs to train the network, where they assume the prior information about the noise model to obtain the denoised image. These self-supervised methods assume prior information about the noise model and although they perform well on synthetic noise, they tend to under perform on real-world noisy images. Recently, Huang *et al.* [20] proposed Neighbor2Neighbor, where they down-sample the noisy image into pairs to train the network. An additional regularizer is used in the loss function to account for the differences in the ground-truth of down-sampled images, and might not exploit the joint relationship between the down-sampled images. On the other hand, traditional approaches like SS-GMM [29] proposed a parametric approach to generate image prior using Gaussian mixture model (GMM) that models the relationship between patches to estimate noise characteristics like variance.

To this end, we propose a novel self-supervised technique based on Gaussian Process (GP) (note GP is a non-parametric approach). In our proposed method, we first obtain down-sampled images from the noisy image. Then we perform random cyclical shift to these down-sampled images in order to increase the number of down-sampled images. Random cyclical shifts [10] are found to minimize artifacts in denoised images helping us to generate better quality pseudo-GTs. Further, based on the consideration that these down-sampled images have the same noise characteristics and image properties [20], we propose a pseudo-GT generation approach using a Gaussian processes (GP) to model a learnable joint distribution of the down-sampled images. Note unlike, GMM based approaches GP is non-parametric based approach that can formulate joint distribution between infinitly many random variables. Specifically, we formulate a joint Gaussian distribution between down-sampled images that learns joint relation of the denoising function mappings of the down-sampled images to generate pseudo-GT for every down-sampled image. In other words, the learnable joint distribution between down-sampled images using GP, tries to model similar properties among down-sampled images, and also accounts for the difference between down-sampled images by learning covariance relation between the down-sampled images. Additionally, by predicting pseudo-GT for given down-sample image using other down-sampled images and their corresponding denoised clean images, GP is modelling the joint relation between the denoise function mappings of down-sampled images to learn noise properties in the noisy image. Hence, supervising the network weights using the pseudo-GT obtained by GP, helps the network to learn the joint relation between the down-sampled images and leverage the noise characteristic information from the other down-sampled images. In this way, network is trained in a self-supervised way using GP to exploit the real noise distribution, and achieve a better denoising performance.

Transformers are currently being widely adopted for various computer vision tasks [13,17,31,44,52,58]. The major improvements of transformers come from the lack of using convolutions thus not inducing any convolutional inductive biases [38]. This enables transformers to have a global understanding of the input. Recently, transformers have also been used for many low-level vision tasks [6, 27, 46, 57]. In this work, we propose a new transformer architecture-Denoising Transformer (Den-T) tailor-made for denoising application. We note that for denoising we need a global understanding as well as attention to fine details to get the best prediction. To this end, we propose having two branches in the transformer encoder: one focusing to extract fine-context information and another to extract coarse-context information. The coarse context branch is built in a fine-to-coarse way where the feature maps are taken to a lower spatial resolution in the latent space. The fine context branch is built in a coarse-to-fine way where the feature maps are taken to a higher spatial resolution in the latent space. From our experiments, we find that this design helps in improving the denoising performance. More details on why this design works can be found in Sec 3. We train Den-T using the proposed self-supervised technique using GP and run experiments on multiple denoising datasets like Kodak, BSD, Set-14 and SIDD where we achieve better performance than previous unsupervised/self-supervised denoising methods. Figure 1 demonstrates that with the help of multiple down-sampled images and the joint distribution modeling, the proposed method is able to produce clearer and sharper outputs as compared to [20, 25].

The key contributions of this paper are as follows:

- We propose a new self-supervised image denoising approach by modelling the joint distribution between down-sampled images using Gaussian processes. This helps the network to explicitly model the real noise distribution and achieve a better denoising performance
- We propose Denoising Transformer (Den-T), a dual-branch transformer based denoising network which extracts both coarse and fine details to perform denoising.
- We demonstrate the superiority of our proposed method by conducting experiments on multiple synthetic denoising datasets generated using Kodak, BSD, Set-14, and real-world denoising dataset SIDD.

2. Related work

2.1. Supervised Denoising

Compared to the traditional approaches [7, 11, 16, 40], CNN-based methods [5, 8, 28, 33, 36, 55] have achieved superior performance for image denoising. Zhang *et al.* [55] was among the first CNN-based approach and they employed a residual learning mechanism for effective denoising. Later, methods like [2, 15, 18, 24, 42, 56] were proposed that introduced either efficient training or novel architectural modifications. These approaches follow a fully-supervised paradigm and require large amounts of paired noisy-clean images to train the network. However, it is extremely challenging and expensive to collect real-world paired noisy-clean images. This limits the use of supervised methods on real images with unknown noise models.

2.2. Unsupervised and Self-supervised Denoising

Over the past years, image denoising algorithms like NLM [4], BM3D [11], and WNNM [16] have been proposed which make use of local or non-local structures of the images. However, these methods require knowledge of the noise levels. Soltanayev *et al.* [41] proposed a image denoising method for AWGN noise models using Steins unbiased risk estimator (SURE) based method on noisy images. Zhussip *et al.* [59] extended SURE further by training the network using correlated pairs of noisy images.

Lehtinen *et al.* [25] proposed a self-supervised solution which avoids paired noisy-clean data, and instead uses paired noisy-noisy images of the same scene to train the network. Thereafter, in the self-supervised image denoising, Noise2Void (N2V) [21], Noise2Self [3], Noise2Same [50], Self2Self [37] and Noisier2Noise [35] are proposed that uses only one noisy image per scene to train the network. Methods like Probabilistic N2V [22], Laine *et al.* [23], and MWCNN [48] propose an elegant way of modeling noise and probabilistic inference to further improve the denoising performance. Noise-as-clean (NAC) [51] addressed the image denoising task by focusing on the cases where noise is weak. Huang *et al.* [20] down-sampled the noisy image into neighboring pairs of down-sampled images, and used them to train the network, where the proposed loss accounts for the difference in the ground-truth of the neighboring down-sampled images.

2.3. Transformers for low-level vision

After Vision Transformer (ViT) [13] was shown to perform well for visual recognition tasks, transformers have been widely adopted for various other computer vision applications [17, 31, 44, 52, 58]. Especially for low-level vision, Image processing transformer [6] shows how pre-training a transformer on large-scale datasets can help in obtaining a better performance for low-level applications. U-former [46] proposed a U-Net based transformer architecture for restoration problems. Recently, Swin-IR [27] adopted Swin Transformer [30] for image restoration.

3. Preliminaries

Problem setting. Given a set of only noisy images $\mathcal{D} = \{y^i\}_{i=1}^{M}$, our objective is to train Den-T $f_\theta(.)$ and learn the network weights θ to perform image denoising. We follow Huang*et al.* [20] where only noisy images are used to train the network in a self-supervised fashion. Given a noisy image $y \in \mathcal{D}$, we generate down-sampled images with cell-size 2×2 (for more details about down-sampling please refer [20]) and randomly shift them to obtain more down-sampled images for y. Finally, using the proposed method we compute pseudo-GTs for these down-sampled images, and use them for training the network.

Motivation for Self-supervision with GP. Just minimizing L2-Norm between noisy image pairs (in case of N2N [25]) or minimizing L2-Norm between down-sampled images with additional regularizer (in case of Neighbor2Neighbor [20]) might not be beneficial for network in learning the noise model. The additional regularizer [20] accounts for the difference in the ground-truth of down-sampled images but doesn't help the network learn the relationship between the down-sampled images or the noise model. In contrast to [20], we believe that learning joint relation between the down-sampled images is beneficial for a self-supervised method to achieve better performance, since the joint relationship between the down-sampled images leverages the noise information present in the down-sampled images. In other words, formulating joint relationship between the denoising function $f(.)$ mappings of down-sampled images using GP, we can learn the noise

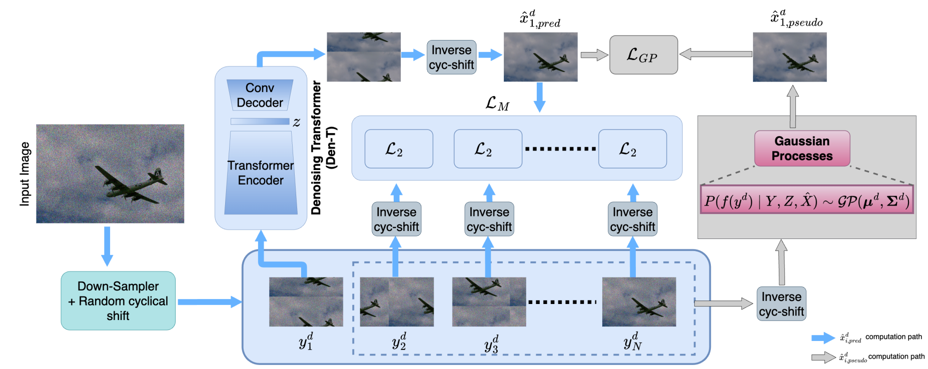

Figure 2. **Overview of the proposed method SST-GP.** Given a noisy image y, we generate down-sampled images $\{y^d_i\}_{i=1}^N (= Y)$, and pass them through Den-T to obtain Z and $\hat{X}$. Later, we model joint distribution between down-sampled images (Y) using GP to compute pesudo-GTs for each of the down-sampled image y^d_i. We then train SST-GP using the proposed loss $\mathcal{L}_{GP}$ and $\mathcal{L}_M$. $\mathcal{L}_2$ represents L2-norm. Down-Sampler represents the down-sampling technique used in [20]. blue arrow denotes the path network denoised image prediction ($\hat{x}^d_{i,pred}$), and grey arrow denotes the path for pseudo-Gt ($\hat{x}^d_{i,pseudo}$) prediction using Gaussian process.

information present in denoised images. To this end, we propose a self-supervised technique based on Gaussian process (GP) to learn pseudo-GT for each down-sampled image while not requiring any paired noisy or clean images to update the network weights.

Let y and s be two independent noisy images conditioned on x, such that $\mathbb{E}_{y|x}(y) = x$ and $\mathbb{E}_{z|x}(z) = x + \varepsilon$ where $\varepsilon \neq 0$ and small. Thus, $y = x + n_1$, $s = x + \varepsilon + n_2$, where n_1 and n_2 are additive zero mean noises with variance σ_y^2 and σ_s^2. If we approximate ε with a Gaussain distribution, *i.e.* $\varepsilon \sim \mathcal{N}\left(0, \sigma_\varepsilon^2\right)$. Let $\tilde{n}_2 = n_2 + \varepsilon$, then,

$$
\begin{gathered}
y - x = n_1, \quad s - x = \tilde{n}_2 \\
P(n_1, \tilde{n}_2) = \mathcal{N}\left(0, \Sigma^2\right), \quad \Sigma^2 = \begin{bmatrix} \sigma_y^2 & 0 \\ 0 & \sigma_z^2 + \sigma_\varepsilon^2 \end{bmatrix} \\
\implies P(y - x, s - x) = \mathcal{N}\left(0, \Sigma^2\right) \implies P(y - f(y), s - f(s)) = \mathcal{N}\left(0, \Sigma^2\right) \\
\implies P(f(y) \,|s, f(s)) = \mathcal{N}\left(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}^2\right)
\end{gathered} \tag{1}
$$

Since in optimal (ideal denoise network) case $x \approx f(y)$, where $f(.)$ represents the function for denoising network. This allows us to formulate learnable joint distribution between function mappings($f(.)$) of y and s with help of GP where in learning this relation between $f(y)$ and $f(s)$, GP learns the noise information present in y and s. By conditioning this joint distribution between $f(y)$ and $f(s)$ (in Eq. 1) with $f(s)$ we can predict the denoised image for y as $\boldsymbol{\mu}_y$. We can define $\boldsymbol{\mu}_y$ in Eq. 1 as pseudo-GT for y and learn the networks weights θ by minimizing the negative log-likelihood of the conditional distribution as follows,

$$
\mathcal{L}_{GP} = -\log P\left(\boldsymbol{\mu}_y - f(y) \mid s, f(s)\right) \tag{2}
$$

In this way, we can learn the joint relation in y and s using GP with help of learnable kernel functions which is beneficial in modelling the similar properties y annd s and account also for differences between them. Updating the network weights using $\mathcal{L}_{GP}$ using $\boldsymbol{\mu}_y$ helps the network to leverage noise present in s. We can extend this to multiple noisy observations $\{y_i\}$ (where $\mathbb{E}_{y_i|x}(y_i) = x + \varepsilon_i$, and ε_i's are small), and formulate joint Gaussian distribution using GP to leverage noise information in $\{y_i\}$'s and update the network using following optimization:

$$
\begin{aligned}
P\left(f(y_i) \mid \{y_j\}_{j\neq i}, \{f(y_j)\}_{j\neq i}\right) &= \mathcal{N}\left(\boldsymbol{\mu}_{y_i}, \boldsymbol{\Sigma}_i^2\right) \\
\mathcal{L}_{GP} = -\log P\left(\boldsymbol{\mu}_{y_i} - f(y_i) \mid \{y_j\}_{j\neq i}, \{f(y_j)\}_{j\neq i}\right)&
\end{aligned} \tag{3}
$$

4. Proposed Method

Given a noisy image y, following Huang *et al.* [20] we obtain neighboring down-sampled images. Then we perform cyclical random shifts to these down-sampled images in order to obtain more down-sampled images for y. Note that [10] explained that random cyclical shifts minimizes the artifacts and aliasing effects introduced during down-sampling. Thus, for noisy image y, we obtain a set of N down-sampled cyclically-shifted images, $\{y^d_1\, y^d_2, y^d_3, \dots, y^d_N\}$. Next, we forward these down-sampled images, $\{y^d_1\, y^d_2, y^d_3, \dots, y^d_N\}$ through the denoising network and inverse-shift them to obtain the corresponding denoised down-sampled images, $\{\hat{x}^d_1\, \hat{x}^d_2, \hat{x}^d_3, \dots, \hat{x}^d_N\}$. Figure 2 gives an overview of the proposed method where each down-sampled image y^d_i is passed through the encoder to obtain intermediate vector $z^d_i = g(y^d_i, \theta_e)$. The vector z^d_i is then forwarded to a decoder followed by a inverse-cyclical shift to obtain the corresponding denoised down-sampled image, *i.e* $\hat{x}^d_i = Inv(h(z^d_i, \theta_d))$. Here, $Inv(.)$ represents inverse-cyclical shift function. SST-GP is trained with two losses: (i) $\mathcal{L}_M$, (minimizing the L2-norm between down-sampled images), and (ii) $\mathcal{L}_{GP}$. The latter loss is constructed based on pseudo-GT predicted by the joint distribution modeled with $\{\hat{x}^d_1\, \hat{x}^d_2, \hat{x}^d_3, \dots, \hat{x}^d_N\}$ using Gaussian processes. First, we explain the details of our transformer network Den-T and then explain how we train it using our proposed GP based self-supervised approach.

4.1. Denoising Transformer (Den-T)

We use a dual branch transformer based encoder and a convolutional decoder for Den-T. The two branches of our encoder are: 1) Fine Context Transformer Branch (FTB) and 2) Coarse Context Transformer Branch (CTB).

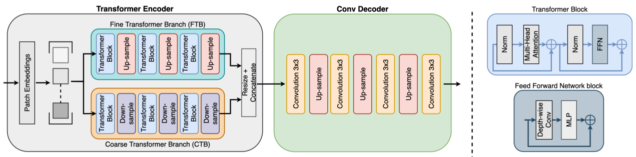

Figure 3. **Overview of our proposed Den-T architecture.** We use two branches (FTB and CTB) in the transformer encoder to extract both coarse and fine information to facilitate efficient denoising. We use a convolutional decoder to get the final prediction.

Fine Context Transformer Branch: To extract fine-detailed information from the input image, CNN-based methods like [45, 53] project the features to a high spatial resolution. Inspired by these works, we apply the same process on the self-attention features to extract fine-details. We use three transformer blocks in this branch with upsampling in between every transformer block. Performing self attention in a high spatial resolution latent space helps in attending to smaller information as the feature space . Upsampling here is done using bilinear interpolation.

Coarse Context Transformer Branch: We use a generic fine-to-coarse transformer branch to extract global features. In this branch, we forward the input image through a series of transformer and downsampling blocks.

Transformer Block: Each transformer block is equipped with multi-head self-attention layers and feed forward networks to calculate the self-attention features. The feed forward process inside a transformer block can be summarized as, $T(I) = FFN(MSA(I) + I)$, where $T()$ represents the transformer block, $FFN()$ represents the feed forward network block, $MSA()$ represents multi head self-attention, I is the input. Similar to the original self-attention network, the heads of queries (Q), keys (K) and values (V) have same dimensions and the self-attention is calculated as:

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V, \tag{4}$$

where d represents the dimensionality. We use multiple attention heads in each transformer block and that number is a hyper-parameter which we vary across each stage in the transformer encoder. More details regarding the hyper-parameter settings can be found in the supplementary document. The self-attention features are then passed to a FFN block. In the FFN block, we use depth-wise convolution to MLP inspired from [26, 47, 49]. Using depth-wise convolution here helps bring locality information and provides positional information for transformers as shown in [49]. The computation in the FFN block can be summarized as follows:

$$FFN(A) = MLP(GELU(DWC(MLP(A)))) + A,$$

where A corresponds to the self-attention features, DWC is depth-wise convolution [9], $GELU$ is Gaussian error linear units [19], and MLP is multi-layer perceptron.

Decoder: We use a convolutional decoder with a series of convolutional and upsampling layers to output the denoised image. An overview of Den-T can be found in Figure 3.

4.2. Self-Supervision using GP

As we do not have the corresponding ground-truths for the down-sampled images $\{y_1^d\, y_2^d, y_3^d, \ldots, y_N^d\}$, we use GP to model the noise information between the noisy down-sampled images. Specifically, we use GP to generate the pseudo-GT's and use them for supervision. The primary intuition behind the pseudo-GT generation is to formulate a joint relation between $\{y_1^d\, y_2^d, y_3^d, \ldots, y_N^d\}$, as they share same image properties and the corresponding input down-sampled images share the same noise distribution. This motivates us to formulate a learnable joint Gaussian distribution between $\{\hat{y}_i^d\}_{i=1}^N$, and predict pseudo-GT for every down-sampled image y_i^d using the denoised images of other down-sampled images $\{\hat{x}_j^d\}_{i\neq j, j=1}^N$. In this way, we are learning a covariance relation and also noise present in the down-sampled images $\{\hat{y}_i^d\}_{i=1}^N$, to train the denoising network in a self-supervised fasion.

Pseudo-GT: Given $\{y_1^d\, y_2^d, y_3^d, \ldots, y_N^d\}$, we forward them through Den-T to obtain the corresponding intermediate vectors $\{z_1^d\, z_2^d, z_3^d, \ldots, z_N^d\}$. These intermediate vectors are then passed through a decoder network and inverse-shifted to obtain the corresponding denoised images $\{\hat{x}_1^d\, \hat{x}_2^d, \hat{x}_3^d, \ldots, \hat{x}_N^d\}$. The denoise function mappings between y_i^d and $\hat{x}_i^d$, *i.e* $\hat{x}_i^d = f(y_i^d), \quad \forall i = 1, 2, 3, \ldots, N$ can be modelled using GP by formulating a joint Gaussian distribution between these function mappings of down-sampled images. Assuming these function mapping $f(.)$ form a Gaussian process (GP) which is an infinite collection of functions of which any finite subset of these function mappings form a jointly Gaussian distribution. Then joint Gaussian distribution for function $f(.)$ mappings of down-sampled images is formulated as follows:

$$\begin{bmatrix} f(y_1^d) \\ f(y_2^d) \\ \ldots \\ f(y_N^d) \end{bmatrix} \sim GP\left(\begin{bmatrix} \mu_1^d \\ \mu_2^d \\ \ldots \\ \mu_N^d \end{bmatrix}, \begin{bmatrix} \kappa\left(z_1^d, z_1^d\right) & \kappa\left(z_1^d, z_2^d\right) & \ldots & \kappa\left(z_1^d, z_N^d\right) \\ \kappa\left(z_2^d, z_1^d\right) & \kappa\left(z_2^d, z_2^d\right) & \ldots & \kappa\left(z_2^d, z_N^d\right) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa\left(z_N^d, z_1^d\right) & \kappa\left(z_N^d, z_2^d\right) & \ldots & \kappa\left(z_N^d, z_N^d\right) \end{bmatrix} + \sigma_\epsilon^2 \mathbb{I}\right). \tag{5}$$

Here, $\mathbb{I}$ denotes identity matrix and σ_ϵ^2 denotes the learnable additive variance. We denote this joint distribution as:

$$P(f(y^d)) \sim \mathcal{GP}(\boldsymbol{\mu}^d, K(Z^d, Z^d) + \sigma_\epsilon^2 \mathbb{I}), \tag{6}$$

where, $\boldsymbol{\mu}^d$ function value obtained using GP, and $K(.,.)$ is the learnable kernel function that defines the covariance relation among down-sampled images. $K(.,.)$ is Rational quadratic (RQ[.]) based kernel function defined as follows,

$$K(Z^d, Z^d)_{p,q} = \kappa\left(z_p^d, z_q^d\right) = \alpha^2 \left(1 + \frac{||z_p^d - z_q^d||_2^2}{\beta^2}\right)^{-0.5} \tag{7}$$

Note that α, β, and σ_ϵ are learnable parameters which help in learning the covariance relation among the down-sampled images.

Here, Z is constructed using the intermediate latent vectors, *i.e* $Z = \{z_i^d\}_{i=1}^N$. We use Z in order to compute covariance since intermediate latent vectors z^d's are more informative than y^d's. Let, Y be a set of all down-sampled images generated from y, *i.e* $Y = \{y_i^d\}_{i=1}^N$, and $\hat{X}$ be a set of the corresponding function values, *i.e* $\hat{X} = \{\hat{x}_i^d\}_{i=1}^N$. We define Y_j^c as a set of all down-sampled image excluding y_j^d, *i.e* $Y_j^c = \{y_i^d : i = [1, N]$ and $i \neq j\}$, similarly $\hat{X}_j^c = \{\hat{x}_i^d : i = [1, N]$ and $i \neq j\}$. Likewise, we define Z_j^c as a set of all intermediate vectors of the down-sampled images excluding z_j^d, *i.e* $Z_j^c = \{z_i^d : i = [1, N]$ and $i \neq j\}$. Using the joint distribution in Eq. 6, we can obtain conditional distribution for $f(y_j^d)$ as the following Gaussain distribution given Y, Z and $\hat{X}_j^c$,

$$P(f(y_j^d)|Y, Z, \hat{X}_j^c) = \mathcal{N}(\boldsymbol{\mu}_j^d, \boldsymbol{\Sigma}_j^d), \tag{8}$$

where

$$\begin{aligned} \boldsymbol{\mu}_j^d &= K\left(z_j^d, Z_j^c\right)\left[K\left(Z_j^c, Z_j^c\right) + \sigma_\epsilon^2 \mathbb{I}\right]^{-1} \hat{X}_j^c, \\ \boldsymbol{\Sigma}_j^d &= K\left(z_j^d, z_j^d\right) - K\left(z_j^d, Z_j^c\right)\left[K\left(Z_j^c, Z_j^c\right) + \sigma_\epsilon^2 \mathbb{I}\right]^{-1} K\left(Z_j^c, z_j^d\right) + \sigma_\epsilon^2 \mathbb{I}. \end{aligned} \tag{9}$$

We use $\boldsymbol{\mu}_j^d$ computed using GP in Eq. 9 as pseudo-GT ($\hat{x}_{j,pseudo}^d$) for the down-sampled image y_j^d. For every down-sampled image generated using input image y, we compute network's denoised down-sampled image $\hat{x}_{j,pred}^d = Inv(h(g(y_j^d, \theta_e), \theta_d)) = f(y_j^d, \theta)$ and pseudo-GT ($\hat{x}_{j,pseudo}^d$) computed using GP (here $Inv(.)$ represents the inverse-cyclical shifting fuction). Finally, we minimize the L2-error between $\hat{x}_{j,pred}^d$ and $\hat{x}_{j,pseudo}^d$ to update the network weights (θ), hence incorporating the modeled joint distribution between down-sampled images that helps learning the noise information to perform image denoising. Further, we gate the L2-error between $\hat{x}_{j,pred}^d$ and $\hat{x}_{j,pseudo}^d$ with the inverse of the computed variance $\boldsymbol{\Sigma}_j^d$ in order to obtain more accurate predictions. This gating ensures that lesser importance is given to the uncertain predictions while learning the network weights. Additionally, we minimize the variance that helps GP model to learn the joint distribution more accurately, and obtain accurate pseudo-GT labels. The proposed GP based loss on the down-sampled images is as follows,

$$\mathcal{L}_{GP} = -\log P\left(\boldsymbol{\mu}_{y_j}^d - f(y_j^d) \mid Y, Z, \hat{X}_j^c\right) = \frac{1}{N}\sum_{j=1}^N \frac{1}{|\boldsymbol{\Sigma}_j^d|}\left\|\hat{x}_{j,pred}^d - \hat{x}_{j,pseudo}^d\right\|_2^2 + \log\left|\boldsymbol{\Sigma}_j^d\right|. \tag{10}$$

L2-norm loss: Motivated by the loss proposed in Noise2Noise [25] and Haug *et al.* [20], we use the following objective function $\mathcal{L}_M$ to exploit the down-sampled image pairs:

$$\mathcal{L}_M = \frac{1}{N(N-1)}\sum_{j=1}^N \sum_{i=[1,N], i\neq j}\left\|\left(\hat{x}_{j,pred}^d - Inv(y_i^d)\right)\right\|_2^2 \tag{11}$$

here, $Inv(.)$ represents inverse-cyclical shift function.

Total loss: The overall loss function used for training the SST-GP is defined as follows,

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{M}} + \lambda_{\text{GP}}\mathcal{L}_{\text{GP}}, \tag{12}$$

where λ_{GP} is a predefined weight that is set equal to 0.03. We provide an ablation study for λ_{GP} in the supplementary document. In our experiments, we use values for $\mathcal{L}_{\text{M}}$ in the order of 10^{-3} and the values of $\mathcal{L}_{GP}$ in the order of 10^{-1}.

4.3. Implementation details

We train our SST-GP network using $\mathcal{L}_{total}$ with Den-T as base denoising network. We use Adam optimizer with a learning rate of 0.0002 and batch-size of 4 to train SST-GP for a total of 60 epochs. We decrease the learning rate by a factor of 0.5 for every 25 epochs. During training, the images are randomly cropped to the size of 256×256. We set $\lambda_{\text{GP}} = 0.03$, cell size $k = 2$ in generating down-sampled images using [20]. We shift each down-sampled cyclical for 4 times, so $N = 8$ for every noisy image y. Pseudo algorithm for training the SST-GP are provided in the supplementary document.

5. Experiments and Results

In this section, we provide the results of various experiments conducted to demonstrate the effectiveness of the proposed approach. In addition, we also provide a comparison of the proposed method with existing methods on both synthetic and real-world noisy datasets.

5.1. Dataset details

Synthetic datasets: For training SST-GP to perform experiments using synthetic sRGB space, we use 50k clean images from the validation dataset of ImageNet [12]. Crops of 256×256 are obtained from these 50k clean images and used to generate noisy images by adding the following 4 different noise levels: (i) Gaussian noise with fixed standard deviation $\sigma = 25$, (ii) Gaussian noise with varied noise level, $\sigma = [5, 50]$, (iii) Poisson noise with fixed $\lambda = 30$, and (iv) Poisson noise with $\lambda = [5, 50]$. Note that these σ, λ values correspond to pixel intensities in the range of $[0, 255]$. Synthetic test sets are created using the clean images from Kodak [14], BSD [34], and Set-14 [54] datasets.

Real datasets: Authors of SIDD [1] collected real-world noisy images of 10 static scenes using 5 smart phone cameras in different lighting conditions. The authors grouped the collected images into SIDD Medium Dataset for training, and use SIDD Validation and Benchmark Dataset in RAW formats. Following the same protocol, we use the SIDD Medium training Dataset to train SST-GP, and use the Validation and Benchmark Datasets for evaluation and comparisons.

5.2. Comparisons on synthetic test data

We use PSNR and SSIM to compare SST-GP against the state-of-the-art (SOTA) methods. We train all the networks using ImageNet [12] following the steps mentioned in the respective SOTA methods. We denote Laine19 [23] with probabilistic post-processing as Laine-pme, and without as Laine-mu. Table 1 shows comparisons on synthetic Gaussian noise test sets, where our proposed method significantly outperforms the previous methods. Table 2 shows

Table 1. PSNR/SSIM comparisons on synthetic test sets created using Gaussian noise. Higher number represents better performance.

Type of Noise	Dataset	N2C [39]	N2N [25]	CBM3D [11]	DIP [43]	N2V [21]	Laine19-mu [23]	Laine19-pme [23]	DBSN [48]	Huang *et al.* [20]	SST-GP (ours)	Den-T w/ GP oracle (ours)
Gaussian $\sigma = 25$	Kodak	32.43/0.884	32.41/0.884	31.87/0.868	27.20/0.720	30.32/0.821	30.62/0.840	32.40/0.883	31.64/0.856	32.08/0.879	**32.75/0.898**	32.98/0.910
	BSD	31.05/0.879	31.04/0.878	30.48/0.861	26.38/0.708	29.34/0.824	28.62/0.803	30.99/0.877	29.80/0.839	30.79/0.873	**31.18/0.880**	31.44/0.900
	Set-14	31.40/0.869	31.37/0.868	30.88/0.854	27.16/0.758	28.84/0.802	29.93/0.830	31.36/0.866	30.63/0.846	31.09/0.864	**31.68/0.872**	31.96/0.896
Gaussian $\sigma = [5, 50]$	Kodak	32.51/0.875	32.50/0.875	32.02/0.860	26.97/0.713	30.44/0.806	30.52/0.833	32.40/0.870	30.38/0.826	32.10/0.870	**31.78/0.880**	32.01/0.913
	BSD	31.07/0.866	31.07/0.866	30.56/0.847	25.89/0.687	29.31/0.801	28.43/0.794	30.95/0.861	28.43/0.788	30.73/0.861	**31.12/0.869**	31.36/0.876
	Set-14	31.41/0.863	31.39/0.863	30.94/0.849	26.61/0.738	29.01/0.792	29.71/0.822	31.21/0.855	29.49/0.814	31.05/0.858	**31.38/0.871**	31.56/0.886

Table 2. PSNR/SSIM comparisons on synthetic test sets created using Poisson noise. Higher number represents better performance.

Type of Noise	Dataset	N2C [39]	N2N [25]	Anscombe [32]	DIP [43]	N2V [21]	Laine19-mu [23]	Laine19-pme [23]	DBSN [48]	Huang *et al.* [20]	SST-GP (ours)	Den-T w/ GP oracle(ours)
Poisson $\lambda = 30$	Kodak	31.78/0.876	31.77/0.876	30.53/0.856	27.01/0.716	28.90/0.788	30.19/0.833	31.67/0.874	30.07/0.827	31.44/0.870	**31.99/0.879**	32.16/0.884
	BSD	30.36/0.868	30.35/0.868	29.18/0.842	26.07/0.698	28.46/0.798	28.25/0.794	30.25/0.866	28.19/0.790	30.10/0.863	**30.84/0.897**	31.04/0.910
	Set-14	30.57/0.858	30.56/0.857	29.44/0.837	26.58/0.739	27.73/0.774	29.35/0.820	30.47/0.855	29.16/0.814	30.29/0.853	**30.87/0.867**	31.14/0.881
Posisson $\lambda = [5, 50]$	Kodak	31.19/0.861	31.18/0.861	29.40/0.836	26.56/0.710	28.78/0.758	29.76/0.820	30.88/0.850	29.60/0.811	30.86/0.855	**31.39/0.872**	31.61/0.897
	BSD	29.79/0.848	29.56/0.848	28.22/0.815	25.44/0.671	27.92/0.766	27.89/0.778	29.57/0.841	27.81/0.771	29.54/0.843	**29.96/0.853**	30.22/0.871
	Set-14	30.02/0.842	30.02/0.842	28.51/0.817	25.72/0.683	27.43/0.745	28.94/0.808	28.65/0.785	28.72/0.800	29.79/0.838	**30.22/0.848**	30.56/0.867

comparison of the proposed method with several recent image denoising approaches [20,23,25,39,43,48] on synthetic Posisson noise test sets. Since the proposed method relies on multiple down-sampled images and uses GP to perform pseudo-label based supervision, it is able to achieve better results as compared to the other methods by a significant margin. Note that in Table 1 and Table 2, we also include the oracle performance i.e. when Den-T trained in a fully-supervised manner with pairs noisy-clean images along with proposed GP loss $\mathcal{L}_{GP}$. Figure 4 illustrates sample denoising results of SST-GP along with recent methods. It can be observed that the results of our method is more clearer and sharper compared to the predictions of other methods [20, 23, 25, 39]. More quantitative comparisons on other self-supervised methods [37, 50] are provided in supplementary material.

5.3. Comparisons on real test data

We use SIDD [1] dataset to compare the performance of SST-GP against other methods. We train all the networks using SIDD Medium training dataset images, and follow the steps mentioned in the respective SOTA methods. As BM3D [11] requires prior information to denoise, we use Anscombe for Poisson to estimate the priors. Results corresponding to this experiment are shown in Table 3 and Figure 5 where we obtain a better performance compared to other methods. In contrast to other methods [20, 23, 25, 39], we used down-sampled images and modelled joint distribution using GP, that helped the proposed SST-GP outperform the other methods by a significant margin and it is able to produce sharper images than the other methods. Note that in Table 3, we also present the oracle performance i.e. when Den-T trained in fully-supervised manner with pairs noisy-clean images and GP loss $\mathcal{L}_{GP}$. Additionally, we compare our method with SS-GMM, that computes noise characteristics in self-supervised way and uses EPLL [60] to denoise the image.

5.4. Ablation Study

Impact of using Den-T: To prove that Den-T is better than CNN-based architectures, we train both U-Net and Den-T in a fully-supervised way using the pairs of noisy-clean images with same losses (L2 and the proposed GP based loss $\mathcal{L}_{GP}$). In Table 4, we can see that Den-T outperforms U-Net even while trained in a similar fully-supervised fashion with comparably less number of parameters. Additionally in Table 4, we compare computational complexity of Den-T using Giga Multiply Accumulate(GMacs) operations per second.

Impact of $\mathbf{L_{GP}}$: In Table 4, it can be observed that using $\mathcal{L}_{GP}$ significantly improved the performance of both U-Net and Den-T by $\sim$ 0.4dB while trained in a fully-supervised. The main reason for this improvement is that proposed pseudo-GT based GP approach learns the relation between the down-sampled images and updates the networks using $\mathcal{L}_{GP}$.

Impact of GP based self-supervision: We train both U-Net and Den-T in self-supervised manner using only noisy images with $\mathcal{L}_M$, we achieved 30.62dB annd 30.76dB in PSNR for BSD test test with Gaussian noise($\sigma = 25$). In Table 4, we can observe that the proposed self-supervised technique, *i.e* learning the joint relation between down-sampled using GP and updating network weights using $\mathcal{L}_{GP}$ improves the performance of both U-Net and Den-T by ~ 0.42dB.

Impact of dual branches in Den-T: we conduct experiments with and without FTB and CTB branches to understand the contributions of individual branches. From Table 5, we can observe that using both branches together help us get a better performance.

Additionally, we compare the performance of Den-T with existing state-of-the-art transformer based denoising networks like SwinIR [27], and Uformer [46]. In Table 5, we can observe that Den-T outperforms Swin-IR [27], and Uformer [46].

5.5. Limitations

Training time of SST-GP with $\mathcal{L}_{GP}$ is 1.5 times slower when compared to training time of Den-T with $L2-norm$, since $\mathcal{L}_{GP}$ involves matrix multiplication for computing $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ (refer Eq. 9). Table 6 shows that Den-T w/ GP requires higher memory during training, this is due to two reasons: (i) matrix multiplication for computing $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$

[0] https://github.com/AbdoKamel/simple-camera-pipeline

Figure 4. comparions on noisy images, first row: Gaussian noise $\sigma = 25$, second row: Poisson noise $\lambda = 30$.

Table 3. PSNR/SSIM comparisons on real-world noise dataset SIDD [1] (Benchmark and validation). Higher number represents better performance.

Methods	N2C [39]	N2N [25]	BM3D [11]	N2V [21]	Laine19-mu [23](Poisson)	DBSN [48]	Huang *et al.* [20]	SS-GMM [29]	SST-GP (ours)	Oracle (ours)
Network	U-Net	U-Net	–	U-Net	U-Net	DBSN	RRGs	–	Den-T w/ GP	Den-T w/ GP
Benchmark	50.60/0.991	50.62/0.991	48.60/0.986	48.01/0.983	50.28/0.989	49.56/0.987	50.76/0.991	48.22/0.984	**50.87/0.992**	51.00/0.994
Vaidation	51.19/0.991	51.21/0.991	48.92/0.986	48.55/0.984	50.89/0.990	50.13/0.988	51.39/0.991	49.84/0.987	**51.57/0.992**	51.68/0.994

Figure 5. Comparisons on real-world noisy images from the SIDD Benchmark in RAW formats. For display purpose we use the code provided by the authors of SIDD[1]to convert images from raw format to srgb.

Table 4. PSNR/SSIM comparisons for ablation study of $\mathcal{L}_{GP}$ using BSD test set.

Dataset	Method	Fully-supervised				Self-supervised			
		U-Net	U-Net w/ GP	Den-T	Den-T w/ GP	U-Net	U-Net w/ GP	Den-T	Den-T w/ GP
	Loss	L2	L2+$\mathcal{L}_{GP}$	L2	L2+$\mathcal{L}_{GP}$	$\mathcal{L}_M$	$\mathcal{L}_M + \mathcal{L}_{GP}$	$\mathcal{L}_M$	$\mathcal{L}_M + \mathcal{L}_{GP}$
BSD	Gaussian $\sigma = 25$	30.96/0.878	31.22/0.881	31.09/0.887	31.44/0.900	30.62/0.869	30.94/0.877	30.76/0.878	31.18/0.884
	Poisson $\sigma = 30$	30.35/0.868	30.84/0.887	30.61/0.903	31.04/0.910	30.11/0.859	30.67/0.880	30.41/0.886	30.84/0.897
Parameters (Miliion)		31	31	24	24	31	31	24	24
GMacs(Million)		55.8	61.6	16.0	20.5	55.8	61.6	16.0	20.5

in GP, and (ii) In FTB we are upsampling features to higher resolutions.

6. Conclusion

In this work, we proposed a new method: Self-Supervised Transformer with Gaussian Process (SST-GP) for image denoising. We proposed a new self-supervised technique where given a noisy image, we generate multiple cyclically shifted noisy down-sampled images and model a joint distribution between them using GP. We also introduced a denoising transformer (Den-T) which is a dual-branch network architecture to extract both coarse and fine details to perform denoising.

Table 5. PSNR/SSIM comparisons for ablation study of Den-T using Kodak testset.

Dataset	Method	SwinIR [27]	Uformer [46]	Den-T w/o FTB w / L2 +$\mathcal{L}_{GP}$	Den-T w/o CTB w / L2 + $\mathcal{L}_{GP}$	Den-T w / L2 +$\mathcal{L}_{GP}$
Kodak	Gaussian $\sigma = 25$	32.89	32.75	32.64	32.69	32.98
	Poisson $\sigma = 30$	32.10	32.07	32.03	32.01	32.16

Table 6. GMacs comparison for image size 256×256.

Method	U-Net	U-Net w/GP	Den-T	Den-T w/ GP
GMacs	9.38	12.75	16.02	20.49

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1700, 2018. 6, 7, 8

[2] Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3155–3164, 2019. 3

[3] Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning*, pages 524–533. PMLR, 2019. 3

[4] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005. 3

[5] Jaeseok Byun, Sungmin Cha, and Taesup Moon. Fbi-denoiser: Fast blind image denoiser for poisson-gaussian noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5768–5777, June 2021. 3

[6] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021. 2, 3

[7] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1256–1272, 2016. 3

[8] Shen Cheng, Yuzhi Wang, Haibin Huang, Donghao Liu, Haoqiang Fan, and Shuaicheng Liu. Nbnet: Noise basis learning for image denoising with subspace projection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4896–4906, June 2021. 3

[9] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 5

[10] Ronald R Coifman and David L Donoho. Translation-invariant de-noising. In *Wavelets and statistics*, pages 125–150. Springer, 1995. 2, 4

[11] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 1, 3, 7, 8

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 3

[14] Rich Franzen. Kodak lossless true color image suite. In *source: http://r0k.us/graphics/kodak*. 6

[15] Shuhang Gu, Yawei Li, Luc Van Gool, and Radu Timofte. Self-guided network for fast image denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2511–2520, 2019. 3

[16] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014. 1, 3

[17] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021. 2, 3

[18] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1712–1722, 2019. 2, 3

[19] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 5

[20] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. *arXiv preprint arXiv:2101.02824*, 2021. 2, 3, 4, 6, 7, 8

[21] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2129–2137, 2019. 2, 3, 7, 8

[22] Alexander Krull, Tomáš Vičar, Mangal Prakash, Manan Lalit, and Florian Jug. Probabilistic noise2void: Unsupervised content-aware denoising. *Frontiers in Computer Science*, 2:5, 2020. 3

[23] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. High-quality self-supervised deep image denoising. *arXiv preprint arXiv:1901.10277*, 2019. 2, 3, 6, 7, 8

[24] Stamatios Lefkimmiatis. Universal denoising networks: a novel cnn architecture for image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3204–3213, 2018. 3

[25] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018. 2, 3, 6, 7, 8

[26] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021. 5

[27] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021. 2, 3, 7, 8

[28] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. *arXiv preprint arXiv:1806.02919*, 2018. 1, 2, 3

[29] Haosen Liu, Xuan Liu, Jiangbo Lu, and Shan Tan. Self-supervised image prior learning with gmm from a single noisy image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2845–2854, 2021. 2, 8

[30] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 3

[31] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 2, 3

[32] Markku Makitalo and Alessandro Foi. Optimal inversion of the anscombe transformation in low-count poisson image denoising. *IEEE transactions on Image Processing*, 20(1):99–109, 2010. 7

[33] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *arXiv preprint arXiv:1603.09056*, 2016. 1, 3

[34] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 6

[35] Nick Moran, Dan Schmidt, Yu Zhong, and Patrick Coady. Noisier2noise: Learning to denoise from unpaired noisy data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12064–12072, 2020. 2, 3

[36] Tobias Plötz and Stefan Roth. Neural nearest neighbors networks. *Advances in Neural Information Processing Systems*, 31:1087–1098, 2018. 1, 2, 3

[37] Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji. Self2self with dropout: Learning self-supervised denoising from single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3, 7

[38] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks?, 2021. 2

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1, 7, 8

[40] Uwe Schmidt and Stefan Roth. Shrinkage fields for effective image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2774–2781, 2014. 3

[41] Shakarim Soltanayev and Se Young Chun. Training and refining deep learning based denoisers without ground truth data. *arXiv preprint arXiv:1803.01314*, 2018. 3

[42] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017. 1, 3

[43] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018. 7

[44] Jeya Maria Jose Valanarasu, Poojan Oza, Ilker Hacihaliloglu, and Vishal M. Patel. Medical transformer: Gated axial-attention for medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*, pages 36–46, Cham, 2021. Springer International Publishing. 2, 3

[45] Jeya Maria Jose Valanarasu, Vishwanath A Sindagi, Ilker Hacihaliloglu, and Vishal M Patel. Kiu-net: Towards accurate segmentation of biomedical images using over-complete representations. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part IV 23*, pages 363–373. Springer, 2020. 5

[46] Zhendong Wang, Xiaodong Cun, Jianmin Bao, and Jianzhuang Liu. Uformer: A general u-shaped transformer for image restoration. *arXiv preprint arXiv:2106.03106*, 2021. 2, 3, 7, 8

[47] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021. 5

[48] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. Unpaired learning of deep image denoising. In *European Conference on Computer Vision*, pages 352–368. Springer, 2020. 2, 3, 7, 8

[49] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021. 5

[50] Yaochen Xie, Zhengyang Wang, and Shuiwang Ji. Noise2Same: Optimizing a self-supervised bound for image denoising. In *Advances in Neural Information Processing Systems*, volume 33, pages 20320–20330, 2020. 3, 7

[51] Jun Xu, Yuan Huang, Ming-Ming Cheng, Li Liu, Fan Zhu, Zhou Xu, and Ling Shao. Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE Transactions on Image Processing*, 29:9316–9329, 2020. 3

[52] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Keypoint localization via transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11802–11812, 2021. 2, 3

[53] Rajeev Yasarla, Vishwanath A. Sindagi, and Vishal M. Patel. Syn2real transfer learning for image deraining using gaussian processes. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5

[54] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 6

[55] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of

deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017. 1, 2, 3

[56] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 3

[57] Dong Zhao, Jia Li, Hongyu Li, and Long Xu. Hybrid local-global transformer for image dehazing. *arXiv preprint arXiv:2109.07100*, 2021. 2

[58] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6881–6890, 2021. 2, 3

[59] Magauiya Zhussip, Shakarim Soltanayev, and Se Young Chun. Training deep learning based image denoisers from undersampled measurements without ground truth and without image prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10255–10264, 2019. 3

[60] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, 2011. 7