

# End-to-End Unsupervised Document Image Blind Denoising

Mehrdad J Gangeh<sup>1\*</sup>   Marcin Plata<sup>2\*</sup>   Hamid R Motahari Nezhad<sup>1</sup>   Nigel P Duffy<sup>1</sup>  
<sup>1</sup>Ernst & Young (EY) LLP USA   <sup>2</sup>EY GDS (CS) Poland Sp. z o.o.

{Mehrdad.J.Gangeh, Hamid.Motahari, Nigel.P.Duffy}@ey.com   Marcin.Plata@gds.ey.com

## Abstract

*Removing noise from scanned pages is a vital step before their submission to optical character recognition (OCR) system. Most available image denoising methods are supervised where the pairs of noisy/clean pages are required. However, this assumption is rarely met in real settings. Besides, there is no single model that can remove various noise types from documents. Here, we propose a unified end-to-end unsupervised deep learning model, for the first time, that can effectively remove multiple types of noise, including salt & pepper noise, blurred and/or faded text, as well as watermarks from documents at various levels of intensity. We demonstrate that the proposed model significantly improves the quality of scanned images and the OCR of the pages on several test datasets.*

## 1. Introduction

Millions of electronic documents, such as contracts and invoices, are reviewed in the normal course of business in the enterprise. A large percentage of them are scanned documents containing various types of noise, including salt & pepper (S&P) noise, blurred or faded text, watermarks, etc. Noise in documents highly degrades the performance of the optical character recognition (OCR) and their subsequent digitization and analysis. The first step toward automating document analysis is to improve their quality using image processing techniques, such as image denoising and restoration. Most of the literature, places attention on removing noise from pictures [25] (*e.g.*, natural scenes) not text documents. However, these techniques may not be directly applicable due to very different nature of text documents.

In the image restoration problem, a degradation function and noise may both affect the quality of images [8]. Examples are deblurring, defading, and inpainting. If in a special case, there is no degradation function, the problem would be a pure image denoising problem (*e.g.*, S&P noise removal).

The current state-of-the-art (SOTA) solutions for im-

age restoration problem are discriminative models based on convolutional neural networks (CNNs), auto-encoders and their variants such as REDNet (residual encoder decoder network) [19, 7], DnCNN (denoising convolutional neural networks) [29], and RDN (residual dense network) [30, 31]. These solutions can generally be formulated as follows:

$$\arg \min_{\theta} \sum_j \sum_i \mathcal{L} \left( f(\mathbf{x}_i^j; \theta) = \hat{\mathbf{y}}_i^j, \mathbf{y}_i^j \right) \quad (1)$$

where  $\mathbf{x}_i^j$  refers to a noisy patch extracted from the noisy image  $\mathbf{x}_i$ ,  $\mathbf{y}_i^j$  and  $\hat{\mathbf{y}}_i^j$  are the target and predicted clean patches, respectively,  $f$  and  $\theta$  refer to the CNN and its parameters. The main shortcoming of this approach, however, is the requirement for the availability of clean target images/documents, which is hard to address in real-world documents. In the literature, the noisy/clean pairs are usually prepared by adding some synthetic noise to clean images/documents. However, the synthetic noise does not completely model noise on real images/documents, and therefore, the performance of the network trained on these synthetic data is sub-optimal and highly degraded on real noisy images/documents [12, 28].

To address the lack of noisy/clean pairs, noise-to-noise (N2N) [15], noise-to-void (N2V) [13], and noise-to-self (N2S) [3, 14] training strategies have been proposed. However, these solutions are based on the assumption that the noise is additive zero-mean, and/or independent between pixels [15, 13, 14]. This only covers a specific kind of denoising problem and, therefore, not directly applicable to general image restoration problems, including defading and deblurring. Furthermore, in N2N approach, at least two noisy instances of the same document page are needed, which are not readily available in real settings.

There are several challenges in the design of an end-to-end solution for document image clean-up: 1) Noisy/clean pairs are not available, and therefore, the standard SOTA solutions based on discriminative models can not be employed. 2) There are various artifacts at different intensity levels (intra-class variation) in documents. 3) We prefer to have a single model based on one architecture, and one training strategy, *i.e.*, a *unified* solution to address all

\*equal contribution

noise/degradation problems (blind denoising/restoration) as opposed to individual models trained separately for each noise type. Training multiple models raises the problem of routing a document containing a specific artifact to the right model for image clean-up.

This paper addresses these challenges by introducing an end-to-end unsupervised image blind denoising algorithm that presents a single unified model to remove various noise types, without the requirement of paired noisy/clean pages. The main contributions are listed as follows:

1. We propose a novel unified architecture by integrating deep mixture of experts with a cycle-consistent GAN as the base network. We formulate a novel loss function for the proposed model.
2. To the best of our knowledge, the designed unified model is the first that removes various artifacts, including noise (such as S&P noise), and degradation (*e.g.*, faded and blurred text, or watermark) at various intensity levels (image blind denoising).
3. We trained the model on actual noisy documents (not documents with synthetically added noise) without the requirement of noisy/clean images, evaluated it across several public and in-house document datasets, and demonstrated its excellent performance on real documents containing various artifacts.

## 2. Related work

### 2.1. Discriminative methods

Deep learning techniques for image denoising were pioneered by discriminative models and CNN/autoencoder architectures. Dong *et al.* [6] proposed one of the earliest CNN-based models for image denoising in the application of compression artifact reduction with several stacked convolutional layers. Ever since, various architectures and modifications of CNNs have been proposed to improve image denoising, including skip symmetric connections in a model named residual encoder-decoder (RED-Net) [19, 7], denoising CNN (DnCNN) employing batch normalization and residual learning [29], residual dense networks (RDNs) [30, 31], wavelet CNNs [16], feature attention [1], and dual residual networks (DRNs) [17]. For example, using DRNs [17], the authors redesigned one network for each specific denoising task, including additive Gaussian noise removal, deblurring, dehazing, and raindrop removal. The main issue with discriminative approaches is that they need noisy/clean pairs.

### 2.2. Generative adversarial methods

Approaches based on generative adversarial networks (GANs) [9, 2] are recently used to alleviate the requirement for noisy/clean pairs in several different ways. *E.g.*, Chen *et al.* [5] proposed to estimate the noise distribution from

smooth parts of noisy patches using a GAN and to generate noise samples. They composed pairs of noisy/clean images by adding the estimated noise samples to clean patches, which are successively used to train a CNN for denoising. They claim that the estimated noise by the GAN is more realistic than the synthetic noise usually added in discriminative approaches. Cha *et al.* [4] went one step further and estimated noisy patches (instead of noise samples) using a GAN. This alleviated the requirement to have clean images. Having two different noisy instances of the same patch/image, they used N2N training strategy [15] to train a CNN for image denoising. Both previous approaches are merely applicable to situations where the noise is zero-mean, additive, and independent of the clean image, which are strong requirements that make them applicable to only a subset of, but not general image restoration problems.

### 2.3. Image denoising in documents

Discriminative approaches for document image clean-up include a CNN-based approach for deblurring [21], a U-net [20] based approach replacing the skip connections between the encoder and decoder blocks with alternating convolutional and recurrent layers for efficient feature extraction [18], a two-stage CNN-based approach where the first stage is to classify the type of deblurring and the second stage to remove it [11], and conditional GANs (cGANs) [26, 24], which is a supervised image-to-image translation approach [10]. DE-GAN [24], particularly, is recently proposed based on cGANs with a modified loss function with promising results on binarization, deblurring, and watermark removal. However, all these methods, including DE-GAN need noisy/clean pairs generated by adding the corruption to the clean pages/patches and train one model per artifact type. Sharma *et al.* [22] proposed document image cleansing based on cycle-consistent GANs [33]. This approach does not require noisy/clean pairs, however, they trained the model using these pairs [22]. Also, they have trained one model for each noise/artifact type, whereas we have trained one single model for all types of artifacts.

## 3. Methods

### 3.1. Problem statement

Here, we express the generic problem of document image clean-up in two statements:

**Problem statement 1** - In real settings, there are many clean documents, which are unpaired to available noisy pages. The main question is whether we can take advantage of these clean documents in our solution. We can formulate the problem as follows: having two unpaired sets of documents, one set consisting of noisy documents ( $\mathbf{X}$ ) and the other a collection of clean documents ( $\mathbf{Y}$ ), and knowing that these two sets are unpaired, can we transform one set

to have the style of the other? This problem can be formulated as an unsupervised image-to-image translation. Recently, there have been several proposals for solving this problem and one of the most prominent ones is based on cycle-consistent GANs (or cycle-GANs in short) [33]. In fact, in this solution, what is more relevant to our problem is to transform the style of noisy documents to clean documents such that we remove/restore noise/degradation from these documents while preserving their text contents.

**Problem statement 2** - The noisy documents  $\mathbf{X}$  may contain several different noise types. Our primary goal is to design a *single model* that can tackle all these noise types. In our solution, we propose to integrate deep mixture of experts into the cycle-GAN.

### 3.2. Cycle-consistent GANs

Suppose there are two sets of unpaired document images  $\{\mathbf{x}_i\}_{i=1}^N$  and  $\{\mathbf{y}_j\}_{j=1}^M$  taken from two domains of noisy  $\mathcal{X}$  and clean  $\mathcal{Y}$  images, respectively. A cycle-GAN [33] consists of two generators: forward ( $H$ ) and backward ( $F$ ) generators, and two adversarial discriminators  $D_{\mathcal{Y}}$  and  $D_{\mathcal{X}}$ . The generators transform the data from one domain to another, *i.e.*,  $H : \mathcal{X} \rightarrow \mathcal{Y}$  and  $F : \mathcal{Y} \rightarrow \mathcal{X}$ . The adversarial discriminators aim to differentiate between the outputs of generators and the real data, *i.e.*,  $D_{\mathcal{Y}}$  aims to discriminate between  $H(\mathbf{x})$  and  $\mathbf{y}$ , whereas  $D_{\mathcal{X}}$  tries to distinguish between  $F(\mathbf{y})$  and  $\mathbf{x}$ . The objective function in cycle-GAN is based on two losses: the GAN loss that transforms the image style from one domain to another, and the cycle-consistency loss that preserves the contents of the image.

### 3.3. Deep mixture of experts

In order to address the problem statement 2, *i.e.*, designing a single unified model that removes different types of noise from documents, we propose using a mixture of experts model. A naïve approach is to combine individual trained cycle-GANs (for each noise type) with an ensemble learning on top (Figure 1a). This model-level combination of experts results in a very complex model that needs as many cycle-GANs as the number of noise types.

Here, we propose an alternative solution with much less complexity based on deep mixture of experts (deep MoEs) [27]. Deep MoE operates within a single model and treats each channel (of a CNN, for example) as an expert. It extends the standard single layer MoE model to multiple layers within a single CNN. Wang *et al.* [27] derived the equivalence between gated channels in a convolution layer and the classic mixture of experts. A deep MoE consists of three main components: 1) base convolutional network, 2) shallow embedding network, and 3) multi-headed sparse gating network (refer to Figure 1 in [27] for the architecture of deep MoEs). The optimization function for the deep MoEs is based on minimizing three losses: 1) the loss

for the base CNN (*e.g.*, cross-entropy loss), 2) the cross-entropy loss for the shallow embedding network, and 3) the  $\ell_1$  loss for the gated networks.

### 3.4. The proposed architecture: integrated cycle-GAN and deep MoE

In order to design a unified model for image clean-up to remove various noise types from documents, we propose to integrate the deep MoE with cycle-GAN as the base model. Figure 1b depicts the sketch of the proposed architecture.

The components of base cycle-GAN in proposed architecture include: two generators, *i.e.*, forward  $H$ , and backward  $F$  generators, as well as two discriminators  $D_{\mathcal{Y}}$  and  $D_{\mathcal{X}}$ . The other components, *i.e.*, an embedder  $E$ , a classifier  $C$ , and gating networks  $g_H^* = G_H^{\{1, \dots, L_H\}}$  and  $g_F^* = G_F^{\{1, \dots, L_F\}}$  ( $L_H$  and  $L_F$  are the number of convolutional layers in  $H$  and  $F$  generators, respectively) construct the elements of the deep MoE in the architecture.

#### 3.4.1 The pipeline architecture formulation

Our goal is to learn the generator function, which generates the cleaned/restored image  $\hat{\mathbf{y}} \in \mathcal{Y}$  based on the input noisy image  $\mathbf{x} \in \mathcal{X}$ . Every noisy image is labeled by one type of noise, which is defined by  $c_{\mathbf{x}} \in \mathcal{C}$ . In our experiments, the types of noise/imperfections are S&P noise, blurred, faded text, or watermarked document. During the pipeline training, we sample a tuple composed of a noisy image and its label  $(\mathbf{x}, c_{\mathbf{x}}) \in \mathcal{X} \times \mathcal{C}$  and a clean unpaired image  $\mathbf{y} \in \mathcal{Y}$ .

The first step in the pipeline is to acquire the embedding vector  $\mathbf{e}_{\mathbf{x}} \in \mathbb{R}^d$  by applying the noisy image  $\mathbf{x} \in \mathcal{X}$  to the embedder, *i.e.*  $\mathbf{e}_{\mathbf{x}} = E(\mathbf{x})$ . Next, we calculate the gating network outputs  $g_H^i = G_H^i(\mathbf{e}_{\mathbf{x}}) \in \mathbb{R}_+^{N_H^i}$  for  $i \in \{1, \dots, L_H\}$  and  $g_F^i = G_F^i(\mathbf{e}_{\mathbf{x}}) \in \mathbb{R}_+^{N_F^i}$  for  $i \in \{1, \dots, L_F\}$ , where  $N_H^i$  refers to the number of output channels in the  $i$ -th convolutional layer in generator  $H$ . The embedder network also predicts the label  $\hat{c}_{\mathbf{x}}$  of the input image  $\mathbf{x}$  based on the embedding vector, *i.e.*,  $\hat{c}_{\mathbf{x}} = C(\mathbf{e}_{\mathbf{x}})$ .

The next steps of the pipeline applies the gating networks  $g_H^*$  and  $g_F^*$  to the cycle-GAN generators  $H$  and  $F$ , respectively. This generates the cleaned image  $\hat{\mathbf{y}} = H(\mathbf{x}, g_H^1, \dots, g_H^{L_H})$  and the noisy image  $\hat{\mathbf{x}} = F(\mathbf{y}, g_F^1, \dots, g_F^{L_F})$ . In both generators, we replace a standard convolutional layer by an MoE convolutional layer, *i.e.*, we calculate  $h_o^{t+1} = \sum_{i=1}^{N_H^t} (g_H^t)_i \mathbf{K}_i^t * h_i^t$ , where  $h^t$  is an output tensor of the  $t$ -th convolution layer and  $h_i^t$  refers to a tensor on its  $i$ -th channel;  $*$  is a convolutional operator and  $\mathbf{K}_i^t$  denotes a kernel for the  $t$ -th layer and the  $i$ -th channel.

The discriminators work in the same way as in the original cycle-GAN, *i.e.*  $D_{\mathcal{X}}(\mathbf{x}) \in [0, 1]$  and  $D_{\mathcal{Y}}(\mathbf{y}) \in [0, 1]$ .

<sup>1</sup>Our solution does not require target clean images during training. We use some metadata (noise type) to train the embedder.

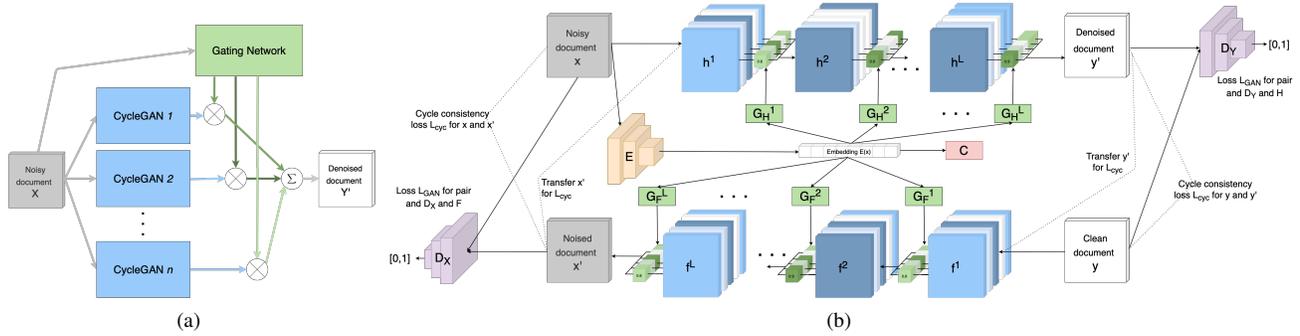


Figure 1: (a) A naïve approach for mixture of experts on cycle-GANs. (b) The architecture of proposed unified model.  $h^i$  and  $f^i$  refer to output tensors of  $i$ -th convolutional layers of  $H$  and  $F$ , respectively. Flat rectangles are fully connected layers, while other components, i.e.,  $D_X$ ,  $D_Y$ ,  $E$ , are CNNs.

### 3.4.2 The loss function formulation

First, we define two additional functions to formulate cycle-GAN generators with deep MoE layers:

$$H^{\text{MoE}}(\mathbf{x}, \mathbf{x}) = H(\mathbf{x}, G_H^1(E(\mathbf{x})), \dots, G_H^{L_H}(E(\mathbf{x}))), \quad (2)$$

and

$$F^{\text{MoE}}(\mathbf{y}, \mathbf{x}) = F(\mathbf{y}, G_F^1(E(\mathbf{x})), \dots, G_F^{L_F}(E(\mathbf{x}))). \quad (3)$$

Note that only the noisy image  $\mathbf{x}$  is provided to the embedder and gating networks. The same argument appears twice in Eq. (2) in order to keep the same function structure as in Eq. (3). Next, we formulate a novel loss function  $\mathcal{L}_{\text{GAN}}$  for  $H^{\text{MoE}}$ ,  $D_Y$ ,  $\mathcal{X}$ ,  $\mathcal{Y}$  as follows:

$$\begin{aligned} \mathcal{L}_{\text{GAN}}(H^{\text{MoE}}, D_Y, \mathcal{X}, \mathcal{Y}) &= \mathbb{E}_{\mathbf{y} \sim \mathcal{Y}}[\log D_Y(\mathbf{y})] \\ &+ \mathbb{E}_{\mathbf{x} \sim \mathcal{X}}[1 - \log D_Y(H^{\text{MoE}}(\mathbf{x}, \mathbf{x}))], \end{aligned} \quad (4)$$

Similarly, we can define the  $\mathcal{L}_{\text{GAN}}$  for  $F^{\text{MoE}}$  as follows:

$$\begin{aligned} \mathcal{L}_{\text{GAN}}(F^{\text{MoE}}, D_X, \mathcal{Y}, \mathcal{X}) &= \mathbb{E}_{\mathbf{x} \sim \mathcal{X}}[\log D_X(\mathbf{x})] \\ &+ \mathbb{E}_{\mathbf{y} \sim \mathcal{Y}, \mathbf{x} \sim \mathcal{X}}[1 - \log D_X(F^{\text{MoE}}(\mathbf{y}, \mathbf{x}))]. \end{aligned} \quad (5)$$

On the other hand, the cycle-consistency loss  $\mathcal{L}_{\text{cyc}}$  for  $H^{\text{MoE}}$  and  $F^{\text{MoE}}$  is defined as provided below:

$$\begin{aligned} \mathcal{L}_{\text{cyc}}(H^{\text{MoE}}, F^{\text{MoE}}, \mathcal{X}, \mathcal{Y}) &= \\ &\mathbb{E}_{\mathbf{x} \sim \mathcal{X}}[\|F^{\text{MoE}}(H^{\text{MoE}}(\mathbf{x}, \mathbf{x}), \mathbf{x}) - \mathbf{x}\|_1] \quad (6) \\ &+ \mathbb{E}_{\mathbf{y} \sim \mathcal{Y}, \mathbf{x} \sim \mathcal{X}}[\|H^{\text{MoE}}(F^{\text{MoE}}(\mathbf{y}, \mathbf{x}), \mathbf{x}) - \mathbf{y}\|_1]. \end{aligned}$$

The final objective function for cycle-GAN network is formulated based on a combination of losses given in Eqs. (4), (5), and (6):

$$\begin{aligned} \mathcal{L}_{\text{cycle-GAN}}(H^{\text{MoE}}, F^{\text{MoE}}, D_X, D_Y, \mathcal{X}, \mathcal{Y}) &= \\ &\mathcal{L}_{\text{GAN}}(H^{\text{MoE}}, D_Y, \mathcal{X}, \mathcal{Y}) + \mathcal{L}_{\text{GAN}}(F^{\text{MoE}}, D_X, \mathcal{Y}, \mathcal{X}) \\ &+ \lambda_{\text{cyc}} \mathcal{L}_{\text{cyc}}(H^{\text{MoE}}, F^{\text{MoE}}, \mathcal{X}, \mathcal{Y}). \end{aligned} \quad (7)$$

In addition, we formulate the loss function required for training deep MoE Layers as follows:

$$\begin{aligned} \mathcal{L}_{\text{MoE}}(E, G_H^*, G_F^*, C, \mathcal{X}, \mathcal{C}) &= \\ &\mathbb{E}_{\mathbf{x}, c_{\mathbf{x}} \sim \mathcal{X}, \mathcal{C}}[\text{CrossEntropy}(C(E(\mathbf{x}), c_{\mathbf{x}}))] \\ &+ \lambda_{G_H} \mathbb{E}_{\mathbf{x} \sim \mathcal{X}} \left[ \sum_{l=1}^{L_H} \|G_H^l(E(\mathbf{x}))\|_1 \right] \quad (8) \\ &+ \lambda_{G_F} \mathbb{E}_{\mathbf{x} \sim \mathcal{X}} \left[ \sum_{l=1}^{L_F} \|G_F^l(E(\mathbf{x}))\|_1 \right]. \end{aligned}$$

The total and final objective function for training the integrated cycle-GAN and deep MoE Layers is derived by combining the losses given in Eqs. (7) and (8):

$$\begin{aligned} \mathcal{L}_{\text{cycle-GAN}}(H^{\text{MoE}}, F^{\text{MoE}}, D_X, D_Y, \mathcal{X}, \mathcal{Y}) \\ &+ \lambda_{\text{MoE}} \mathcal{L}_{\text{MoE}}(E, G_H^*, G_F^*, C, \mathcal{X}, \mathcal{C}). \end{aligned} \quad (9)$$

## 4. Implementation details

### 4.1. Model architecture

The cycle-GAN network is adopted from the implementation explained in [33]. ResNets of nine blocks are used in the generators. The discriminator networks are  $70 \times 70$  Patch-GANs [10], which classify  $70 \times 70$  overlapping patches as real or fake. The embedder network is a 7-layer CNN with kernel size of  $3 \times 3$ , batch normalization and ReLU activation functions. The classifier  $C$ , which is the last layer of the embedder is a fully connected layer with softmax and cross-entropy loss function for classification of the input patches based on their noise type (4 classes). Finally, the gating networks are fully connected networks with ReLU activation functions, whose inputs are from the penultimate layer of the embedder, i.e., just before the classifier  $C$ . There are 18 gating networks for the two

generators, 9 for each generator corresponding to the number of blocks per generator. It is worthwhile to highlight here that these networks are essential during training of the model. For the inference, *i.e.*, for document image clean-up, a minimal model is required, including the forward generator  $H^{\text{MoE}}$ , the embedder  $E$ , and the gating networks  $g_H^*$  corresponding to the forward generator  $H^{\text{MoE}}$ . This greatly reduces the network complexity during the inference in terms of network latency and memory footprint.

## 4.2. Hyper-parameters and model training

Our experiments showed that the same as the original cycle-GAN [33], using a least-square loss instead of the negative log likelihood or Wasserstein loss for  $\mathcal{L}_{GAN}$  results in a more stable training and a better performance in document image clean-up application, and hence we adopted this loss during training the model. Also, same as the original cycle-GAN, as suggested by Shrivastava *et al.* [23], the discriminators were updated based on a history of 50 previously generated images instead of just the last generated image by the generators. We empirically set  $\lambda_{\text{cyc}} = 10$ ,  $\lambda_{\text{MoE}} = 1$ , and  $\lambda_{G_F} = \lambda_{G_H} = 0.1$ . Adam optimizer with the batch size of 32 was used with a learning rate of  $2 \times 10^{-4}$ . We used 4 Nvidia Tesla V100 GPUs for training the model, which took about 1.5 days to complete. The model was implemented using PyTorch framework.

## 5. Experimental setup and results

### 5.1. Training dataset

There are three main document types: unstructured (such as lease contracts, and scientific papers), semi structured (*e.g.*, invoices), and structured (like tax forms). In order to prepare a *training set*, document pages of various types and noise contents were selected from our in-house documents. The most common noise types on lease contracts are S&P noise, blurred, or faded text, whereas tax forms and invoices are mostly in digital format containing watermarks. The number of pages in each category is shown in Table 1 along with other details about the dataset. Overlapping patches of  $256 \times 256$  pixels with the stride of 128 pixels were extracted from these pages to train the network.

The set of noisy and clean pages for the lease contracts are completely unpaired. As for the tax forms and invoices, extracting patches of size  $256 \times 256$  pixels from the original watermarked pages results in only 10% patches with watermark (since only a small part of the page is watermarked). Therefore, we synthetically added watermarks to the grids of  $4 \times 2$  of clean tax forms and invoices with the variations in fonts, text, size, orientation ( $0, \pm 45$ ), colors (gray, light gray, red, purple, and blue), and transparency (scale of 0.1 to 0.6), randomly selected from a uniform distribution (a sample page is shown in Figure 4j). Although

this approach generates pairs of watermarked/clean pages, we have not used this information in training the proposed model as the patches were randomly selected from the sets of watermarked and clean pages for training the model irrespective of how they were paired.

### 5.2. Test datasets

In order to assess the performance of trained unified model, three datasets are used.

**Dataset I:** 100 clean and high quality pages from scientific papers. This provides more tightly controlled conditions for quantitative assessment as the OCR on the original high-quality pages can be considered as the ground truth. Watermarks were synthetically added to each page.

**Dataset II:** 100 pages selected from Tobacco800 dataset [32]. These pages were originally noisy and were cleaned using our unified image clean-up approach.

**Dataset III:** A dataset of 300 in-house noisy documents containing various noise types, including S&P noise, blurred/faded text, and watermarks.

### 5.3. Evaluation metrics

Since the ultimate purpose of image cleansing is improving the performance of the OCR, we used the improvement in OCR as the metric for the quantitative assessment of the model. We used ABBYY FineReader12 as the OCR engine, and OCRred both the original noisy document pages as well as cleaned ones (outputs of the model). Since the ground truth for the characters are not available, we extracted words from characters on each page and used a *relative metric*: the words found on the cleaned page were considered as the reference and compared with the words on the noisy page. The percentage of mismatches between the two was then computed as a metric to measure the amount of improvement. In case that the original clean pages are available (*e.g.*, in Dataset I), we compared them as a reference with the noisy (watermarked in Dataset I) pages as well as cleaned pages. We then provided the relative metric as a measure of deterioration. Although this is not a perfect metric, it provides a reasonable quantitative assessment in the absence of ground truth for characters. We reported the averaged percentage improvement/deterioration, maximum improvement/deterioration, and the percentage of pages improved/deteriorated more than 5% and 10%.

In addition, since for Dataset I, the original clean and high quality images are available, we have provided the peak signal-to-noise ratio (PSNR) metric as well.

### 5.4. Ablation study

In order to demonstrate the effectiveness of gating networks, we have provided the visualization of their outputs. In Figure 2a, the Pearson correlation coefficients are calculated between vector outputs of forward gating network

Table 1: The details of the dataset used for training the proposed model.

Document Types	Lease Contracts			Tax Forms	Invoices
Noise Types	S&P	Blurred	Faded	Watermarked	Watermarked
No. of noisy/clean pages	663/1125	1513/1125	377/1125	5416/5416	1339/1339
Page Size / Patch Size	Resized to the Closest Multiple of Patch Size / $256 \times 256$				
Total No. of Noisy Patches	290,119	578,316	137,181	2,212,729	542,121
Total No. of Clean Patches	440,159				
Data Augmentation	Overlapping with the Stride of 128 Pixels				

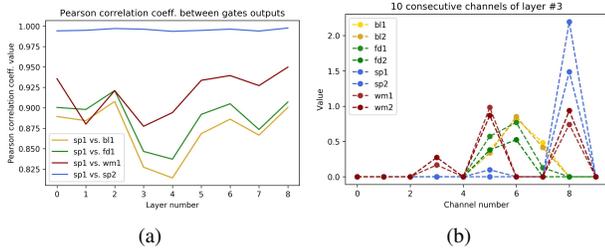


Figure 2: The effectiveness of gating networks on various noise types used in this paper. See text for explanations.

$g_H^*$  for an input image containing S&P noise (sp1), and another image containing any other noise type (S&P, blurred, faded, or watermarked). The correlations between two samples containing S&P are close to one for all layers, whereas these correlations are much lower between S&P and other noise types. Figure 2b displays 10 consecutive values of a section of gating network  $g_H^*$  for the third convolutional layer of forward generator<sup>2</sup>. These values are displayed for two samples of every considered noise type. Both these two figures demonstrate that the gating network has similar responses for the same input noise types and different patterns for different noise types. Note that in Figure 2b some values are 0, which is the result of  $\ell_1$  loss on the gating networks.

### 5.5. Qualitative results

Here, we present sample outputs of the trained model on unseen test patches and pages for various noise and document types<sup>3</sup>. Figure 3 depicts samples of cleaned patches (along with the corresponding inputs of noisy patches) for four different artifacts, including S&P noise, blurred, faded text, and watermark. As can be observed from these results, the proposed trained model can efficiently clean patches containing various artifacts at various intensity levels.

Figure 4 displays samples of cleaned pages along with the corresponding inputs of noisy pages. The rightmost images are cleaned pages generated by the proposed approach. The images in the middle are the cleaned pages generated

<sup>2</sup>The responses for all layers are provided in the Supplementary.

<sup>3</sup>More results are provided in the Supplementary.

by models trained on individual noise types (one cycle-GAN per noise type), except for the watermarked page where the middle image is cleaned using REDNet [19, 7]. The proposed model is able to remove all noise types without distorting the contents of the page as effectively as individual models solely trained for one noise type. Also, for the watermarked page, the proposed approach was able to remove watermark as good as REDNet, which is a supervised approach and solely trained for watermark removal.

### 5.6. Quantitative results

For dataset I, the OCR on the original high-quality pages are considered as the ground truth, and the OCR on watermarked pages as well as cleaned pages were compared with this ground truth. The results are provided in Table 2. On average, the proposed unified image clean-up improves the performance of ABBYY OCR by approximately a factor of three. The low percentage of relative change in the OCR quality when comparing the original and cleaned pages demonstrates the effectiveness of the proposed algorithm in removing watermarks, and reducing the OCR errors due to watermarks on the page. We compare our proposed approach with REDNet [19, 7] and DE-GAN [24] as two representative discriminative methods. We trained the REDNet and DE-GAN separately using watermarked tax forms as specified in Table 1, and tested them on Dataset I to compare with our results. What we should consider in this comparison is that both REDNet and DE-GAN have *solely* been trained on watermarked tax forms/clean pairs in a *supervised* manner, whereas the proposed approach has been trained on *all* noise types (including S&P, blurred/faded text, and watermark) and all document types in an *unsupervised* fashion. However, the performance of the proposed method is just slightly inferior to the REDNet and better than DE-GAN. The proposed approach outperforms DE-GAN mainly because DE-GAN, by design, removes color from inputs. Since watermarks on pages can be in color, DE-GAN has more difficulty to remove them. In addition, to demonstrate the effectiveness of the MoE in our proposed approach to train one single model to remove multiple noise types, we compare the proposed approach with a standard cycle-GAN [33, 22] without deep MoE, and also with our

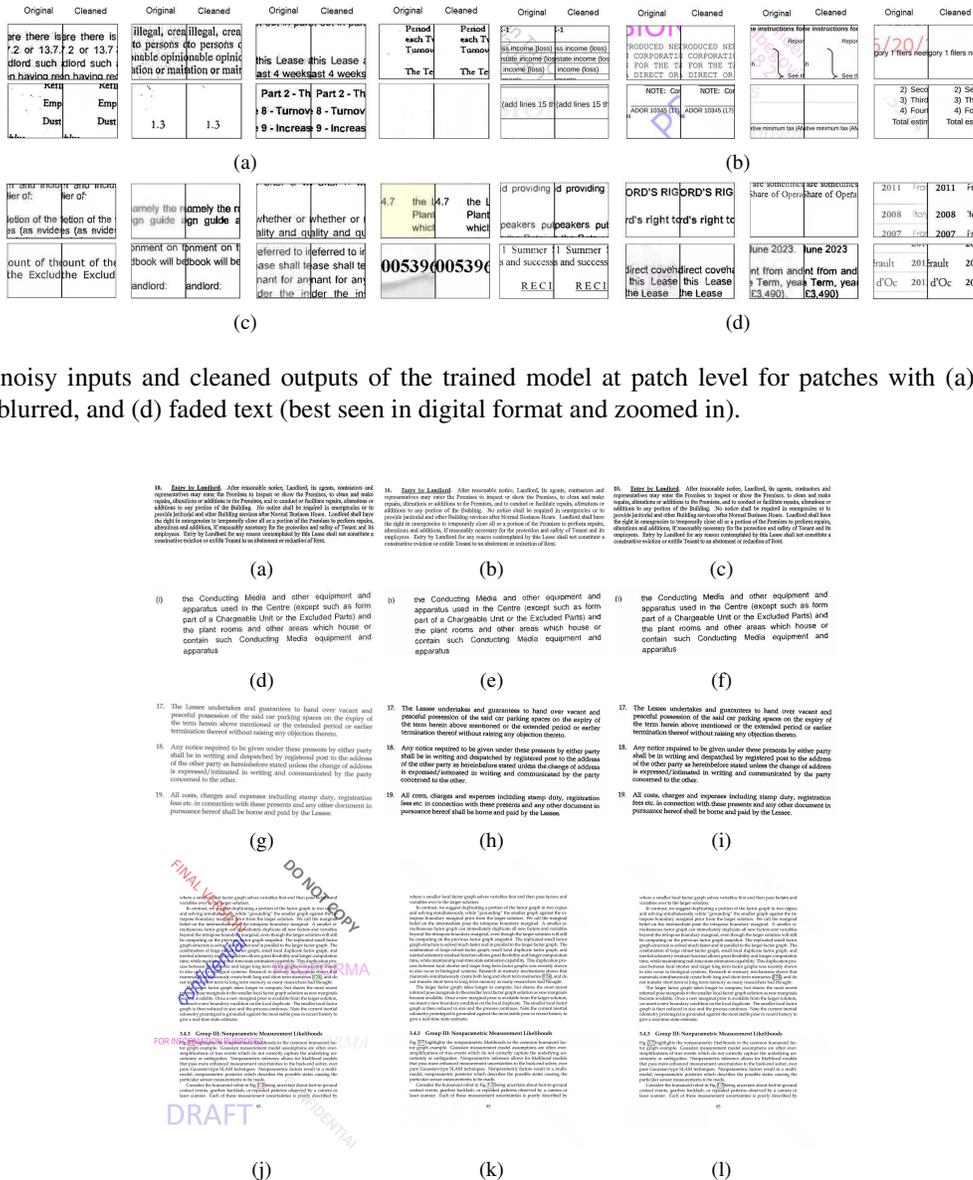


Figure 3: The noisy inputs and cleaned outputs of the trained model at patch level for patches with (a) S&P noise, (b) watermark, (c) blurred, and (d) faded text (best seen in digital format and zoomed in).

Figure 4: The noisy inputs (left images), cleaned outputs using individual model (middle images) or REDNet (for watermarked page), as well as the proposed unified model (right images) at page level for pages with (a) S&P noise, (d) blurred text, (g) faded text, and (j) watermark (best seen in digital format and zoomed in).

proposed approach without classifier  $C$  (both trained the same way as the proposed approach). The results in Table 2 shows that the proposed approach outperforms these two networks as well. This demonstrates that removing the classifier  $C$  from the final stage of the embedder makes training the embedder less efficient, and deteriorates the overall performance of the model. The classifier  $C$  only supports training the embedder network  $E$ , and this embedding has two important functions: (a) allows the forward generator  $H^{\text{MoE}}$  to adapt to various noise types (represented by the

embedding vectors) and efficiently clean them, (b) informs the backward generator  $F^{\text{MoE}}$  the type of noise to be added to its input clean image. In the standard cycle-GAN, information about noise is stored in a denoised image. In case of multiple noise types, this confuses the backward generator and the function of embedder is to avoid this confusion.

Neither original clean pages nor the ground truth OCR are available for Datasets II and III. However, the relative OCR performance of ABBYY on the original noisy pages and the corresponding cleaned pages is used to evaluate the

Table 2: The results of quantitative assessment of the proposed model using the *relative metric* and PSNR on Dataset I. The best results, which in this case are lower numbers for OCR-based metrics and higher numbers for PSNR, as *cleaned* documents are compared with the *original clean* ones, are highlighted.

Measures		Original vs Watermarked (%)	Original vs Cleaned (%)				
			RED-Net [7]	DE-GAN [24]	cycle-GAN [22]	Proposed w/o Classifier	Proposed
Averaged Deterioration (%)		7.71	<b>1.84</b>	3.74	7.85	2.54	2.55
Max. Deterioration (%)		53.09	<b>6.99</b>	19.70	33.33	14.73	9.93
Perc. of Pages	+5% Deterioration	70	<b>3</b>	14	67	12	10
	+10% Deterioration	27	<b>0</b>	5	34	3	<b>0</b>
PSNR		35.65	37.64	37.43	36.52	37.99	<b>38.33</b>

Table 3: The results of quantitative assessment of the proposed model using the *relative metric* on Datasets II and III. The best results, which in this case are higher numbers, as *cleaned* documents are compared with *original noisy* ones, are highlighted.

Measures		Dataset II			Dataset III		
		Cleaned vs Original (%)			Cleaned vs Original (%)		
		cycle-GAN [22]	cycle-GAN (Sequential)	Proposed	cycle-GAN [22]	cycle-GAN (Sequential)	Proposed
Averaged Improvement (%)		5.94	6.03	<b>7.2</b>	6.30	7.10	<b>9.52</b>
Max. Improvement (%)		39.02	52.54	<b>63.06</b>	48.17	57.98	<b>66.94</b>
Perc. of Pages	+5% Improvement	37	42	<b>52</b>	52	58	<b>67</b>
	+10% Improvement	22	24	<b>28</b>	37	42	<b>49</b>

performance of the proposed model. The results are shown in Table 3. Using the proposed approach on average, the OCR is improved by 7.2% and 9.52% on Datasets II and III, respectively. Also, more than 50% of pages on both datasets receive more than 5% improvement in the OCR as a result of cleaning the pages using the proposed unified model. As for comparison with other approaches, there is no *single model* in the literature that can remove all these noise types. Also, discriminative models like REDNet or DE-GAN cannot be used for comparison as they need noisy/clean pairs for training, which are not available here. Nonetheless, We compare the proposed approach with two alternative methods using standard cycle-GAN: 1) same as previous experiment on Dataset I, training one single cycle-GAN (without deep MoE) using a training set consisting of all patches with various noise types. 2) *Standard cycle-GAN with Sequential training*: since it is difficult to train the standard cycle-GAN using all noise types in the training set, we first trained the model on more difficult to learn noise types (*e.g.*, S&P) for several epochs. Then we stopped training and resumed again using patches from noise types that can be removed easier like faded/ blurred text. Although this sequential training strategy improved the performance compared to “*standard*” training, the performance is still far below the performance of individual trained models, as well as the proposed unified model. In addition, “*sequential*” training is heuristic and needs some trial-and-error in order to

find the optimal number of steps, as well as the number of epochs in each step. As can be observed in Table 3, the proposed model significantly outperforms the alternative methods in improving the OCR quality of cleaned pages.

As a final remark, the inference latency of the proposed algorithm was benchmarked on one GPU (Nvidia Tesla V100). On average, it takes 4.49 seconds to clean a page.

## 6. Conclusion

In this paper, we proposed an end-to-end unsupervised multi-document image blind denoising that presents a unified model to remove various artifacts from all document types, without any need for target paired cleaned pages. We formulated the loss function for the proposed model and demonstrated that it can successfully remove artifacts and improve OCR on various document types. The proposed model is of low complexity and inference latency. In future work, we will replace the embedder and the classifier components by an autoencoder-based approach in order to train the embedding vector in an unsupervised manner.

## Disclaimer

The views reflected in this article are the views of the author and do not necessarily reflect the views of the global EY organization or its member firms.

## References

- [1] Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3155–3164, 2019.
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.
- [3] Joshua Batson and Loïc Royer. Noise2self: Blind denoising by self-supervision. In *Proceedings of Machine Learning Research*, pages 2124–2132, 2019.
- [4] Sungmin Cha, Taeon Park, Byeongjoon Kim, Jongduk Baek, and Taesup Moon. GAN2GAN: Generative noise learning for blind denoising with single noisy images, 2021.
- [5] Jingwen Chen, Jiawei Chen, Hongyang Chao, and Ming Yang. Image blind denoising with generative adversarial network based noise modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3155–3164, 2018.
- [6] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *IEEE International Conference on Computer Vision (ICCV)*, pages 576–584, 2015.
- [7] Mehrdad J. Gangeh, Sunil R. Tiyyagura, Sridhar V. Dasaratha, Hamid Motahari, and Nigel P. Duffy. Document enhancement system using auto-encoders. In *Workshop on Document Intelligence at NeurIPS*, 2019.
- [8] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., USA, 2006.
- [9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS'14*, pages 2672–2680, Cambridge, MA, USA, 2014. MIT Press.
- [10] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017.
- [11] Jile Jiao, Jun Sun, and Naoki Satoh. A convolutional neural network based two-stage document deblurring. In *International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 703–707, 2017.
- [12] Yoonsik Kim, Jae Woong Soh, Gu Yong Park, and Nam Ik Cho. Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3479–3489, 2020.
- [13] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void - learning denoising from single noisy images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2124–2132, 2019.
- [14] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. High-quality self-supervised deep image denoising. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 6970–6980. Curran Associates, Inc., 2019.
- [15] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning image restoration without clean data. volume 80 of *Proceedings of Machine Learning Research*, pages 2965–2974, Stockholm, Sweden, 10–15 Jul 2018. PMLR.
- [16] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 886–895, 2018.
- [17] Xing Liu, Masanori Suganuma, Zhun Sun, and Takayuki Okatani. Dual residual networks leveraging the potential of paired operations for image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7000–7009, 2019.
- [18] Bharat Mamidibathula and Prabir K. Biswas. SVDNet: Spatially variant U-Net for blind document deblurring. In *Workshop on Document Intelligence at NeurIPS*, 2019.
- [19] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 2802–2810. Curran Associates, Inc., 2016.
- [20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, Cham, 2015. Springer International Publishing.
- [21] Michal Hradíš, Jan Kotera, Pavel Zemčík, and Filip Šroubek. Convolutional neural networks for direct text deblurring. In X. Xie, M. W. Jones, and G. K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 6.1–6.13. BMVA Press, September 2015.
- [22] Monika Sharma, Abhishek Verma, and Lovekesh Vig. Learning to clean: A GAN perspective. In G. Carneiro and S. You, editors, *Proceedings of Asian Conference on Computer Vision (ACCV) Workshops*, volume 11367 of *Lecture Notes in Computer Science*, pages 174–185, 2018.
- [23] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2242–2251, 2017.
- [24] Mohamed A. Souibgui and Yousri Kessentini. DE-GAN: A conditional generative adversarial network for document enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.
- [25] Chunwei Tian, Lunke Fei, Wenxian Zheng, Yong Xu, Wangmeng Zuo, and Chia-Wen Lin. Deep learning on image denoising: An overview. *Neural Networks*, 131:251–275, 2020.
- [26] D. Vishwanath, Rohit Rahul, Gunjan Sehgal, Swati, Arindam Chowdhury, Monika Sharma, Lovekesh Vig, Gau-

- tam Shroff, and Ashwin Srinivasan. Deep reader: Information extraction from document images via relation extraction and natural language. In G. Carneiro and S. You, editors, *Proceedings of Asian Conference on Computer Vision (ACCV) Workshops*, volume 11367 of *Lecture Notes in Computer Science*, 2018.
- [27] Xin Wang, Fisher Yu, Lisa Dunlap, Yi-An Ma, Ruth Wang, Azalia Mirhoseini, Trevor Darrell, and Joseph E. Gonzalez. Deep mixture of experts via shallow embedding. In Ryan P. Adams and Vibhav Gogate, editors, *Proceedings of Machine Learning Research*, volume 115, pages 552–562, Jul 2020.
- [28] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. Unpaired learning of deep image denoising. In A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, editors, *European Conference on Computer Vision (ECCV)*, pages 352–368, 2020.
- [29] 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.
- [30] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2472–2481, 2018.
- [31] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.
- [32] Guangyu Zhu, Yefeng Zheng, David Doermann, and Stefan Jaeger. Multi-scale structural saliency for signature detection. In *Proc. IEEE Conference Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.
- [33] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.