RESIDUAL CONTRASTIVE LEARNING: UNSUPERVISED REPRESENTATION LEARNING ON NOISY IMAGES

Anonymous authors
Paper under double-blind review

ABSTRACT

In the era of deep learning, supervised residual learning (ResL) has led to many breakthroughs in *low-level vision* such as image restoration and enhancement tasks. However, the question of how to formalize and take advantage of unsupervised ResL remains open. In this paper we consider visual signals with additive noise and propose to build a connection between ResL and self-supervised learning (SSL) via contrastive learning (CL). We present *residual contrastive learning* (RCL), an unsupervised representation learning framework for downstream *low-level vision* tasks with noisy inputs. While supervised image reconstruction tasks aim to minimize the residual terms directly, RCL formulates an instance-wise discrimination pretext task by using the residuals as the discriminative feature. Empirical results on low-level vision tasks with noisy inputs show that RCL is able to learn transferable representations, whilst retaining significantly reduced annotation costs over fully supervised alternatives. This further validates that CL is robust against the task misalignment between the pretext task and the downstream task in SSL.

1 INTRODUCTION

In statistics and optimization, a *residual* denotes the difference between an observed value and an estimated value of interest. In the domain of deep learning, the residual commonly takes the form of $r(x) = y - x$, where x is the input, y is the output, and $r(\cdot)$ is the residual function (He et al., 2016a;b). Residual learning (ResL) has fueled substantial recent advances in low-level image restoration and enhancement tasks, *e.g.* denoising (Zhang et al., 2017), demosiacing (Kokkinos & Lefkimmiatis, 2018), and super resolution (SR) (Lim et al., 2017; Li et al., 2018). While previous studies focus on learning residuals in a supervised fashion, the analogous formulation of unsupervised ResL remains an open question.

Low-level computer vision tasks commonly have to deal with degraded signals. Specifically, the task input signals will often contain additive noise. In the case of large-scale supervised training, representations are learned for a specific task (Zhang et al., 2017; Kokkinos & Lefkimmiatis, 2018). However, self-supervised learning (SSL)[1] on noisy images (Lehtinen et al., 2018; Batson & Royer, 2019; Ehret et al., 2019) requires a distinct formulation and may be considered an emerging research area in low-level computer vision. A key concept of SSL is to define a *pretext* task with self-supervision and, crucially, specific pretext tasks are typically designed based on the nature of the downstream problems to be solved. However, in this paper our experimental investigation provides evidence that representations learned by such task-dependent pretext tasks lack transferability for alternative downstream tasks. In this work, we therefore aim to answer an under-explored question: *how to learn informative and universally useful representations for low-level vision tasks from noisy images?*

Motivation Fueled by the breakthroughs in representation learning, a promising solution is contrastive learning (CL) (Chen et al., 2020; He et al., 2020; Misra & Maaten, 2020; Tian et al., 2020; Chuang et al., 2020), which can learn transferable representations for high-level vision tasks. However, *how to formulate an instance-wise discrimination pretext task on noisy inputs* remains a challenging question for CL. Under standard supervised learning (SL), let (x, y) define a noisy and

[1]We use the terms "self-supervised learning" and "self-supervised representation learning" interchangeably in this work.

noise-free image pair, the loss can then be formulated as $\|y - f_\theta(x)\|$, where $f_\theta(\cdot)$ is the model of interest with parameter θ. In many cases, y is unavailable due to annotation costs, *e.g.* obtaining ground truth data for low-level vision tasks typically requires complex and often constraining procedures. By removing the requirement of a noise-free image y, instead minimizing $\|x - f_\theta(x)\|$ can be seen to easily provide a trivial solution: $f_\theta(\cdot)$ is an identity mapping. Thus, various efforts in SSL have been made to minimize $\|\tilde{x} - f_\theta(x)\|$, where $\tilde{x}$ constitutes another noisy variant of x (Lehtinen et al., 2018; Batson & Royer, 2019; Ehret et al., 2019). In this work we propose a distinct alternative approach to this strategy. We observe that, without the norm operator, $x - f_\theta(x)$ can be regarded as a residual term. Inspired by supervised ResL, we aim build a connection between ResL and CL in this work.

Contributions We bridge a methodological gap between SSL on visual signals with additive noise and unsupervised ResL. We present *residual contrastive learning* (RCL), a robust SSL framework on noisy images. Instead of applying a contrastive loss to encoded images (*i.e.* feature vectors) directly, we approach this question by examining the residuals. We conjecture that the residuals can be effectively used as a discrimination feature for CL. The key challenge that arises is that existing contrastive loss terms are commonly designed for semantic feature vectors. To solve this problem, we propose the *residual contrastive loss* based on InfoNCE (Oord et al., 2018). We utilize a measure over distributions to compute the similarity between two residual tensors with the same shape (*c.f.* cosine similarity for two feature vectors). The statistical distance we choose only measures the divergence between two probability distributions, instead of element-wise correspondence between two residual tensors. We systematically evaluate RCL with different data modalities and downstream tasks. Our empirical results show that RCL robustly outperforms SSL baselines for the task of representation learning on noisy images. The representations learned by RCL also express strong generalization ability in multiple downstream tasks, such as denoising, demosaicing, and SR. Finally, we report that a learning paradigm of pre-training on unlabeled data by RCL, followed by fine-tuning on small amount of labeled data with SL, can significantly reduce data annotation costs.

Our contributions can be summarised as:

- We provide the first formulation of an instance-wise discrimination pretext task based on residuals. In order to leverage this task, we demonstrate that contrastive loss functions (*e.g.* InfoNCE) can be reformulated using appropriate distance metrics to discriminate between statistical distributions.
- We propose RCL, a novel unsupervised representation learning framework that can learn transferable representations from only noisy inputs. To the best of our knowledge, this constitutes the first study of CL on noisy images for low-level image reconstruction tasks.
- The empirical results show that RCL can learn robust representations from noisy images without paired ground truth, at significantly reduced annotation cost.

The remainder of the paper is organized as follows. In Sec. 2, we briefly introduce the preliminary knowledge of unsupervised contrastive learning and signal-dependent noise. In Sec. 3, we introduce the proposed *residual contrastive learning*. In Sec. 4, we validate the effectiveness of RCL with extensive experiments. Finally, in Sec. 5, the conclusions are drawn.

2 PRELIMINARY

2.1 UNSUPERVISED CONTRASTIVE LEARNING

Contrastive learning (CL) was originally developed in order to enable neural networks to identify what makes two objects similar or different (Baldi & Pineda, 1991). A widely adopted contrastive loss InfoNCE (Oord et al., 2018), is formulated as,

$$\mathcal{L}_{\text{NCE}} = -\log \frac{\exp(\text{sim}(z_q, z_0)/\tau)}{\sum_{i=0}^{N} \exp(\text{sim}(z_q, z_i)/\tau)} \tag{1}$$

where z denotes the feature vector extracted from an image patch of interest, τ is a temperature parameter, and $\text{sim}(\cdot, \cdot)$ is the cosine similarity function. Firstly image patches are encoded into feature vectors via an encoder and then $\text{sim}(\cdot, \cdot)$ can be used to measure the similarity between

these representations. Here, (z_q, z_0) is a positive pair such that two patches are taken from the same image; and $(z_q, z_{i>0})$ is a negative pair, where two patches are taken from different images.

From the perspective of representation learning, recent CL frameworks (Oord et al., 2018; He et al., 2020; Chen et al., 2020; Tian et al., 2020; Misra & Maaten, 2020) design pretext tasks by utilizing the invariance of the semantic information encoded between different views of the same instance, *i.e.* different crops of the same instance should contain similar semantic information. Mathematically, minimizing Eq. (1) is equivalent to maximizing the mutual information of two views of the same image (Oord et al., 2018). Thus, the model is expected to learn, in a self-supervised fashion, invariant semantic features that are shared by pairs of views, to the benefit of downstream tasks.

2.2 Signal-Dependent Noise Model

In the domains of computer vision and image processing, given an observed image $\boldsymbol{x}$ and its underlying noise-free image $\boldsymbol{y}$ serving as ground truth, we have

$$\boldsymbol{x} = \boldsymbol{y} + \boldsymbol{n}, \boldsymbol{n} \sim P(\boldsymbol{n}) \tag{2}$$

where $\boldsymbol{n}$ constitutes additive noise following a probability distribution $P(\boldsymbol{n})$, that is introduced during the image acquisition process due to hardware limitations. The real $P(\boldsymbol{n})$ is typically unknown in practical problems and has been modelled with varying levels of complexity (Foi et al., 2008; Wei et al., 2020). A common simplifying assumption involves statistically modelling $P(\boldsymbol{n})$ as a homoscedastic Gaussian distribution. This simplification makes a strong assumption that the noise is independent of the underlying signal. However, many previous studies (Healey & Kondepudy, 1994; Gow et al., 2007; Liu et al., 2008; Foi et al., 2008; Hasinoff et al., 2010; Makitalo & Foi, 2012) have rigorously shown that image noise is in fact signal-dependent; $\boldsymbol{n} \sim P(\boldsymbol{n}|\boldsymbol{y})$, *i.e.* that the probability distribution of $\boldsymbol{n}$ is conditioned on $\boldsymbol{y}$. Statistical signal-dependent noise modeling has thus also been previously explored and a common modelling choice is the zero-mean heteroscedastic Gaussian model (Mohsen et al., 1975; Liu et al., 2014), known as the noise level function (NLF) in the domain of computational photography,

$$n_i \sim \mathcal{N}(0, \lambda_{\text{shot}} x_i + \lambda_{\text{read}}), \tag{3}$$

where n_i is the noise at pixel y_i of the clean image $\boldsymbol{y}$, and λ_{shot} and λ_{read} are parameters controlling the variance of the Gaussian model.

3 Residual Contrastive Learning

In this section, we present *residual contrastive learning* (RCL) in details. Firstly, we formulate the unsupervised representation learning problem for low-level vision tasks with noisy inputs in Sec. 3.1. Secondly, we introduce an empirical observation, which is also the main assumption of this study, in Sec. 3.2. Thirdly, we propose to use the residuals as the discriminative features for CL and present *residual contrastive loss* in Sec. 3.3. Fourthly, we describe the regularization term for low-level vision tasks. Finally, in Sec. 3.5, we discuss an exception of the general problem formulation and propose an adapted solution.

3.1 Problem Formulation

Following the notation in Sec. 2, we denote $\boldsymbol{x}$ as a noisy image signal with clean image signal $\boldsymbol{y}$ and additive noise $\boldsymbol{n}$. The tuple $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{n})$ can be formed by considering terms from Eq. (2).[2] The noise element $\boldsymbol{n}$ follows an unknown distribution with variance depending on the underlying but unknown $\boldsymbol{y}$. It is common to assume $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\sigma}^2(\boldsymbol{y}))$ *i.e.* $\boldsymbol{n}$ is a zero-mean Gaussian distribution with variance dependent on $\boldsymbol{y}$ (Yue et al., 2019).

For an image reconstruction task under SL, a training set $\mathcal{S} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N_{\mathcal{S}}}$ is given, with $N_{\mathcal{S}}$ training examples. Let f_θ denote a model of interest which takes $\boldsymbol{x}$ as input. The optimization goal is then to minimize $\|f_\theta(\boldsymbol{x}) - \boldsymbol{y}\|_p$, for optimal model weights θ where $\|\cdot\|_p$ denotes the p-norm.

In contrast to SL; unsupervised representation learning, the problem setting of interest in this work, involves an unlabelled training set. We alternatively consider $\mathcal{S} = \{\boldsymbol{x}_i\}_{i=1}^{N_{\mathcal{S}}}$ and the goal is to

[2]For simplicity, we assume $\boldsymbol{x}$ and $\boldsymbol{y}$ take a common image format, *e.g.* RGB or RAW.

learn representations (*i.e.* optimize model weights θ) for downstream tasks with access to the noisy image signal only. In Sec. 4, we evidence the efficacy of our contributions by considering several prominent image reconstruction tasks (denoising, demosaicing, and super resolution), through the lens of unsupervised representation learning.

3.2 ASSUMPTION

Commonly adopted camera models that account for physical noise (Foi et al., 2008; Makitalo & Foi, 2012) state that the acquisition of image signals are subject to various factors such as the radiant power of the scene, the exposure time, the sensor gain, and the analog-to-digital conversion (ADC). Thus, signal-dependent noise may be introduced at different stages due to hardware limitations. Consideration of this model leads to a first assumption:

Assumption 1. *On average, the noise distributions associated with two image crops, extracted from the same image, have delectably smaller divergence than noise distributions pertaining to crops extracted from different images.*

Asm. 1 constitutes a natural extension of the assumption that $\boldsymbol{n} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^2(\boldsymbol{y}))$. In fact, for natural images, the noise distributions of two crops originating from the same instance may also be correlated due to potential self-similarities (Batson & Royer, 2019), that is, similar structures appearing at different locations and scales in the same image.

3.3 RESIDUAL CONTRASTIVE LOSS

In this section, we formally introduce the proposed *residual contrastive loss*. First, we define the residual tensor for $\boldsymbol{x}$ as

$$\hat{\boldsymbol{n}}(\boldsymbol{x}) = r_\theta(\boldsymbol{x}) = \boldsymbol{x} - f_\theta(\boldsymbol{x}). \tag{4}$$

In contrast to common supervised ResL settings (He et al., 2016a;b), we apply a contrastive loss on the residuals to learn f_θ. We formulate a *residual-based* instance-wise discrimination pretext task, *i.e.* we use the residual tensors (*c.f.* the feature vectors in Eq. (1)) as the input for CL. The framework is illustrated in Fig. 1. We note, however, that the $\mathrm{sim}(\cdot,\cdot)$ function in Eq. (1) implicitly imposes two constraints: 1. the input z is required to take the form of a normalized vector; and 2. an element-wise correspondence between two feature vectors is required in the feature space. To realize a residual contrastive loss suitable for dense prediction tasks associated with low-level vision, we relax these constraints by replacing the cosine similarity $\mathrm{sim}(\cdot,\cdot)$ with a negative distance function. The original contrastive loss (Eq. (1)) can then be reformulated as

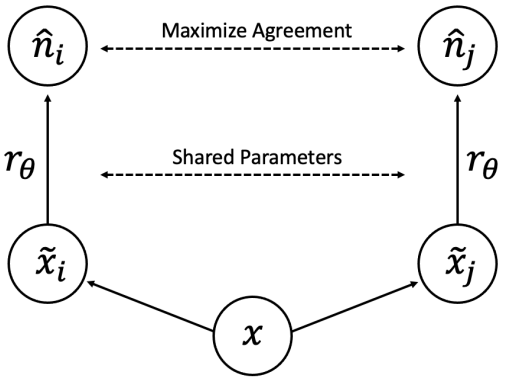

Figure 1: RCL of low-level visual representations for noisy inputs. We use the framework of SimCLR (Chen et al., 2020) to illustrate the main concept. $\boldsymbol{x}$ is a noisy image. $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$ are two random crops from the same $\boldsymbol{x}$ (a positive pair). $r_\theta(\cdot)$ is a residual function, defined in Eq. (4). $\hat{\boldsymbol{n}}_i$ and $\hat{\boldsymbol{n}}_j$ are two corresponding residual tensors.

$$\mathcal{L}_{\text{contrast}} = -\log \frac{\exp(-\mathrm{d}(\hat{\boldsymbol{n}}(\boldsymbol{x}_q), \hat{\boldsymbol{n}}(\boldsymbol{x}_0))/\tau)}{\sum_{i=0}^{N} \exp(-\mathrm{d}(\hat{\boldsymbol{n}}(\boldsymbol{x}_q), \hat{\boldsymbol{n}}(\boldsymbol{x}_i))/\tau)} \tag{5}$$

where τ is a temperature parameter (Chen et al., 2020) and $\mathrm{d}(\cdot,\cdot)$ is a non-negative statistical metric measuring the divergence between two probability distributions, such that larger metric values indicate larger divergence. Similar to Eq. (1), we define a positive pair $(\hat{\boldsymbol{n}}(\boldsymbol{x}_q), \hat{\boldsymbol{n}}(\boldsymbol{x}_0))$ as two image patches $(\boldsymbol{x}_q, \boldsymbol{x}_0)$ cropped from the same instance and a negative pair $(\hat{\boldsymbol{n}}(\boldsymbol{x}_q), \hat{\boldsymbol{n}}(\boldsymbol{x}_{i>1}))$ as two image patches $(\boldsymbol{x}_q, \boldsymbol{x}_{i>1})$ cropped from two different instances.

We note that, unlike cosine similarity, $\mathrm{d}(\cdot,\cdot)$ should not assume a pair-wise relationship between two samples, as the noise distribution is independent of the pixel location. Valid distance measure

Algorithm 1 Batch-wise training of *residual contrastive loss*.

1: Sample a batch of $N+1$ images. ▷ Sample $N+1$ positive pairs
2: Sample two positive patches for each image.
3: Generate $\hat{\boldsymbol{n}}(\boldsymbol{x})$ for each of $2N+2$ patches. ▷ Eq. 4
4: **for** $j = 1, 2, \cdots, N+1$ **do** ▷ Compute *recontrastive loss*
5: Take the j^{th} pair as the positive pair $(\hat{\boldsymbol{n}}(\boldsymbol{x}_q), \hat{\boldsymbol{n}}(\boldsymbol{x}_0))$.
6: Take the second patch of each of the other N pairs as $\hat{\boldsymbol{n}}(\boldsymbol{x}_{i>1})$.
7: Compute $\mathcal{L}_{\text{contrast}}$ for the j^{th} positive pair. ▷ Eq. 5
8: Sum up $\mathcal{L}_{\text{contrast}}$ for a batch $N+1$ images as the batch-wise *residual contrastive loss*.

$\mathrm{d}(\cdot,\cdot)$ should also possess desirable properties such as ease of computation and differentiability, towards enabling efficient end-to-end training. Common information theoretic measures (*e.g. Kullback–Leibler divergence*) that require density estimation do not meet the above requirements.

We therefore investigate a number of alternative feasible options for $\mathrm{d}(\hat{\boldsymbol{n}}(\boldsymbol{x}_p), \hat{\boldsymbol{n}}(\boldsymbol{x}_q))$, where $(\hat{\boldsymbol{n}}(\boldsymbol{x}_p), \hat{\boldsymbol{n}}(\boldsymbol{x}_q))$ are two residual tensors. The smaller the value of $\mathrm{d}(\cdot,\cdot)$, the smaller the distributional divergence between two residual tensors is.

Bhattacharyya Distance The *Bhattacharyya distance* (BD) has a closed-form expression in the case that the two distributions of interest are Gaussian. The sample distribution of residual tensors can be approximated by a Gaussian distribution when the crop size is large enough. Formally, the BD between two Gaussian distributions $\hat{\boldsymbol{n}}(\boldsymbol{x}_p) \sim \mathcal{N}(\mu_p, \sigma_p^2)$ and $\hat{\boldsymbol{n}}(\boldsymbol{x}_q) \sim \mathcal{N}(\mu_q, \sigma_q^2)$ can be estimated as

$$\mathrm{BD}(\hat{\boldsymbol{n}}(\boldsymbol{x}_p), \hat{\boldsymbol{n}}(\boldsymbol{x}_q)) = \frac{1}{4}\ln\left(\frac{1}{4}\left(\frac{\sigma_p^2}{\sigma_q^2} + \frac{\sigma_q^2}{\sigma_p^2} + 2\right)\right) + \frac{1}{4}\left(\frac{(\mu_p - \mu_q)^2}{\sigma_p^2 + \sigma_q^2}\right), \tag{6}$$

where μ and σ^2 are the sample mean and the sample variance.

Earth Mover's Distance The *earth mover's distance* (EMD), also known as the 1^{st} *Wasserstein distance*, is

$$\mathrm{EMD}(\hat{\boldsymbol{n}}(\boldsymbol{x})_p, \hat{\boldsymbol{n}}(\boldsymbol{x})_q) = \inf_{\gamma \in \Pi(P_p, P_q)} \mathbb{E}_{(\hat{\boldsymbol{n}}(\boldsymbol{x})_p, \hat{\boldsymbol{n}}(\boldsymbol{x})_q) \sim \gamma}[\|\hat{\boldsymbol{n}}(\boldsymbol{x})_p - \hat{\boldsymbol{n}}(\boldsymbol{x})_q\|], \tag{7}$$

where $\hat{\boldsymbol{n}}(\boldsymbol{x})_p \sim P_p$, $\hat{\boldsymbol{n}}(\boldsymbol{x})_q \sim P_q$, and $\Pi(\cdot,\cdot)$ denotes the joint distribution. We highlight that the EMD does not make assumptions over the residual distribution (*c.f.* BD).

Maximum Mean Discrepancy The *maximum mean discrepancy* (MMD) is a non-parametric method based on the *kernel embedding of distributions* where a probability distribution is represented as an element of a *reproducing kernel Hilbert space* (Gretton et al., 2012). Given a domain Ω, let an arbitrary function $h : \Omega \to \mathbb{R}$ belongs to a class of functions $\mathcal{H}$, MMD is mathematically defined as

$$\mathrm{MMD}(\mathcal{H}, P_p, P_q) = \sup_{h \in \mathcal{H}} (\mathbb{E}_p[h(\hat{\boldsymbol{n}}(\boldsymbol{x}_p))] - \mathbb{E}_q[h(\hat{\boldsymbol{n}}(\boldsymbol{x}_q))]). \tag{8}$$

Similar to EMD, MMD does not make any assumption over the distribution in question and has a robust empirical estimation based on Gaussian kernels (Pan et al., 2010).

For completeness, the batch-wise training details of the residual contrastive loss are illustrated in Algorithm 1.

3.4 Optimization

While Eq. (5) enables self-supervised representation learning, a new question arises: as f_θ could represent an arbitrary function that fits in Eq. (4), the representations learned by RCL may not be meaningful for the downstream tasks of interest. CL works well for high-level visual representations because the pretext tasks and downstream tasks both involve discrimination of visual objects. Similarly, we require to build such a connection between RCL and low-level vision tasks.

As the performance of the low-level image reconstruction tasks is sensitive to pixel-level intensities, a simple solution is to include the term $\|\boldsymbol{x} - f_\theta(\boldsymbol{x})\|$ as a regularizer. Note that minimizing $\|\boldsymbol{x} -$

$f_\theta(\boldsymbol{x})\|$ alone (*i.e.* without RCL) could lead to the trivial solution of an identity mapping. This problem can be mitigated through the introduction of non-linearities to both terms. Inspiring by this, we leverage the basic concept of the *perceptual loss* (Johnson et al., 2016) and can define the *consistency* loss term as

$$\mathcal{L}_{\text{consistency}} = \|\phi(g_e(\boldsymbol{x})) - \phi(g_e(f_\theta(\boldsymbol{x})))\|_2^2, \tag{9}$$

where $\phi(\cdot)$ represents the features extracted from a pre-trained encoder g_e. Note, g_e could be pre-trained in either a supervised fashion (*e.g.* on ImageNet (Deng et al., 2009)) or an unsupervised fashion on the unlabeled noisy inputs (*e.g.* via MoCo (He et al., 2020)). We utilize the assumption that the noisy input image and the reconstructed output image should convey similar semantic information, *i.e.* the noise should not drastically change the semantic content of the image.

The final training objective is then the sum of the two introduced losses:

$$\mathcal{L}_{\text{total}} = \alpha\mathcal{L}_{\text{contrast}} + \mathcal{L}_{\text{consistency}}, \tag{10}$$

where α is a weighting parameter chosen empirically.

3.5 Ill-Defined Residual

In Sec. 3.3 and Sec. 3.4, the difference $\boldsymbol{x} - f_\theta(\boldsymbol{x})$ implicitly assumes that $\boldsymbol{x}$ and $f_\theta(\boldsymbol{x})$ (or $\boldsymbol{y}$) have the same shape, or more specifically, reside in the same color space. A common scenario in low-level vision is that the input and output signals reside in different color spaces. One such example low-level image reconstruction task of fundamental importance is joint demosaicing and denoising (JDD) (Hirakawa & Parks, 2006; Khashabi et al., 2014; Gharbi et al., 2016), which transforms noisy RAW images to clean RGB images. In such cases $\boldsymbol{x}$ now defines an image in RAW format, and f_θ learns a mapping for our considered example task, JDD. Given a RAW input $\boldsymbol{x}$ with shape $H\times W\times 1$[3], the output $f_\theta(\boldsymbol{x})$ will produce an RGB image with shape $H\times W\times 3$. Such a formulation unfortunately results in Eq. (4) becoming ill-defined. One simple solution involves redefinition of the residual term in the RAW domain. We can thus redefine Eq. (4) as $\hat{\boldsymbol{n}}(\boldsymbol{x}) = \boldsymbol{x} - \text{mosaic}(f_\theta(\boldsymbol{x}))$, where mosaic is a standard image mosaic operation that transforms the images in RGB format back to RAW format. In lieu of applying g_e, pre-trained in the RAW domain, we adopt a simple consistency loss formulation for JDD (Ehret et al., 2019):

$$\mathcal{L}_{\text{consistency}} = \|\boldsymbol{x} - \text{mosaic}(\mathcal{T}(f_\theta(\boldsymbol{x})))\| \tag{11}$$

where $\mathcal{T}$ is a linear interpolation operation (Keys, 1981). Note, due to the mosaic operation lacking an inverse, Eq. (11) does not offer a trivial solution.

4 Experiments

In this section, we first describe the experimental setting, implementation details and evaluation metrics in Sec. 4.1. Then, we report and analyze the experimental results on the synthetic RGB and RAW data in Sec. 4.2 and Sec. 4.3, respectively. At last, we evaluate RCL on the real data in Sec. 4.4.

4.1 Experimental Setup

4.1.1 Simulation

To empirically validate the idea of contrastive representation learning with residuals, we firstly design large-scale simulated experiments, based on synthetic signal-dependent noise. Note, we have to leverage synthetic datasets due to the lack of real-world multi-task datasets for low-level vision tasks. To simulate such signal-dependent noise, we generate synthetic heteroscedastic Gaussian noise based on an NLF model (Eq. (3)). We use different (λ_{shot}, λ_{read}) to model different cameras and acquisition settings. The parameters (λ_{shot}, λ_{read}) are randomly sampled to ensure the overall noise variance level σ^2 of each image falls in a reasonable range for the data used in our experiments[4] and we set $\sigma \in [0, 20]$ following (Gharbi et al., 2016). In this way, we consider each image

[3]Following Gharbi et al. (2016), a practical option is to first rearrange $\boldsymbol{x}$ into shape $H/2\times W/2\times 4$ as a pre-processing step, where each channel contains one of the four colors in the CFA array.

[4]See Appendix A.1 for noise simulation.

to have a noise distribution with *unique* parameters. From the perspective on the dataset $\mathcal{S}$, there is therefore an approximate one-to-one mapping between (λ_{shot}, λ_{read}) and each image. We use this challenging simulation model to evaluate the robustness of SSL frameworks.

4.1.2 DATASETS

In order to simulate the large-scale unlabeled training data with controllable signal-dependent noise, we consider three large-scale public datasets, namely, the MIT Demosaicing dataset (Gharbi et al., 2016) (MIT), the Stanford Taskonomy dataset (Zamir et al., 2018) (Stanford), and the PASCAL VOC dataset (Everingham et al., 2010) (VOC). We generate noisy RGB images by adding synthetic noise following Eq. (3). For JDD, following previous work (Heide et al., 2014; Gharbi et al., 2016; Ehret et al., 2019), each RGB image is first mosaiced to form a Bayer pattern (*e.g.* `RGGB` in this work) RAW image and then synthetic noise is added. The datasets are split into a training set and a test set.[5]

4.1.3 PROXY EVALUATION

CL aims to learn representations for the downstream tasks, *i.e.* pre-training f_θ instead of directly solving the problem. In this work, we aim to test the generalization ability of the learned representations (Zhang et al., 2016). Following previous studies on CL (He et al., 2020; Chen et al., 2020), we adopt a *proxy evaluation* approach. Concretely, we fine-tune the learned representations on downstream tasks with a small amount of annotated data, under SL. We report the performance of the downstream tasks as the *proxy performance* of SSL. In this way, we can systematically evaluate the generalization and transferablility of representations learned under different SSL frameworks. Following the *linear classification protocol* (He et al., 2020), once weights of a network f_θ have been pre-trained using an unlabeled training set, they are then fixed (except the last layer). The original last layer is replaced with a randomly initiated task-dependent layer for the downstream task. The new last layer is then fine-tuned with the labeled training set and evaluated on the test set. Note, under proxy evaluation, the representations of the intermediate layers are fixed. The reported numerical results are used to indirectly reflect the quality of fixed representations, thus this is a *proxy* evaluation. This evaluation protocol is different from the commonly seen protocols in low-level vision tasks, where the end-to-end solutions can be compared directly, without fine-tuning, and the reconstructed images can be visualized for qualitative comparison.

4.1.4 EVALUATION METRICS

We consider two common image reconstruction metrics for the proxy evaluation, *peak signal-to-noise ratio* (PSNR) and *structure similarity index measure* (SSIM). PSNR is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{H \times W \times C} \sum_i^H \sum_j^W \sum_k^C (x-y)^2}, \tag{12}$$

where H, W, C are the height, width, and number of channels for paired images x and y. SSIM is defined as

$$\text{SSIM} = l(x-y)^\alpha c(x-y)^\beta s(x-y)^\gamma, \tag{13}$$

where

$$\begin{aligned} &\alpha > 0, \beta > 0, \ \gamma > 0, \\ &l(x-y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}, \\ &c(x-y) = \frac{\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}, \\ &s(x-y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}. \end{aligned}$$

Here, μ_x and μ_y are the mean of x and y, σ_x and σ_y are the standard deviation of x and y, σ_{xy} is the covariance between x and y, and c_1, c_2, and c_3 are constants. We use the default implementation of

[5] See Appendix A.2 for the dataset descriptions.

`scikit-image` package[6]. We repeat experiments over five trials and report mean results. We use the performance of supervised pre-training as an *Oracle*.

4.1.5 IMPLEMENTATION

Theoretically, f_θ may constitute any model capable of performing dense prediction tasks. In this section, we will show that the representations learned by f_θ can be successfully applied to various downstream image reconstruction tasks: denoising, demosaicing and super resolution. Following (Zamir et al., 2018), we utilize a generic network backbone; U-Net (Ronneberger et al., 2015) to instantiate f_θ. We additionally use a ResNet50 (He et al., 2016a), pre-trained on ImageNet (Deng et al., 2009) for the fixed feature extractor g_e. To instantiate Eq. (5), we follow (Chen et al., 2020) in defining temperature τ values and use a batch size of 64. We use a weighting parameter $\alpha = 10^{-3}$ in the unsupervised pre-training phase and an L_1 loss for the supervised fine-tuning in the evaluation phase. We use an Adam (Kingma & Ba, 2015) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$, and a fixed learning rate 10^{-3}. The minimal crop size is 128×128. All models are implemented in PyTorch on a NVIDIA Tesla V100 GPU.

4.2 EXPERIMENTS ON RGB DATA

4.2.1 BASELINES

We consider two strong SSL baselines, which are designed for representation learning on noisy images, namely *noise2noise* (N2N) (Lehtinen et al., 2018) and *noise2self* (N2S) (Batson & Royer, 2019). For N2N, we generate paired noisy RGB images with the same random parameters (λ_{shot}, λ_{read}). Note that N2N and N2S both utilize a formulation of $\|\tilde{\boldsymbol{x}} - f_\theta(\boldsymbol{x})\|$, where $\tilde{\boldsymbol{x}}$ is a noisy observation of $\boldsymbol{x}$. We also include three state-of-the-art unsupervised CL framework for high-level vision tasks to validate our hypothesis that there exists a task misalignment between semantic understanding tasks and image restoration tasks. The first one is MoCo (V2) (He et al., 2020). As MoCo can only pre-train a feature extractor such as ResNet (*c.f.* U-Net), we use a ResNet-FCN (Long et al., 2015), where the feature extractor is a ResNet50 pre-trained using MoCo and only the deconvolutional component is fine-tuned. The second one is a CL framework for dense prediction tasks, VADeR (O. Pinheiro et al., 2020). Unlike MoCo, VADeR applies CL in a pixel-level, thus can train an encoder-decoder network directly for dense prediction tasks. The third one is also a pixel-level CL framework, *PixContrast* (Xie et al., 2021). Note, in contrast to RCL, VADeR and *PixContrast* are designed for semantic understanding tasks, *i.e.* task misalignment still exists. We use a U-Net backbone for VADeR and *PixContrast*, where the number of output channels of the last layer is set as 3 for RGB images. The pre-training and fine-tuning procedures follow Sec. 4.1.3, same as RCL.

4.2.2 DENOISING

We instantiate denoising as the downstream task and report representation learning results in Table 1. MoCo has the weakest performance for two reasons. Firstly, MoCo is not designed for learning representations for low-level vision tasks. Secondly, ResNet-FCN is not designed for image reconstruction tasks and MoCo cannot pre-train networks (the decoders) for such dense prediction tasks directly (Dong et al., 2021). VADeR and *PixContrast* show slightly improved performance over MoCo, as they can pre-train the decoder. However, all three CL frameworks designed for high-level tasks produce results much worse than SSL methods designed for low-level vision tasks, thus they are omitted in the following discussion. Note, in the proxy evaluation, the pre-training set and testing set do not overlap. RCL shows competitive representation learning performance in comparison with N2N and N2S, which are reported to achieve reasonable performance in blind denoising tasks. Among the three distance metrics explored, EMD shows robust performance in comparison to BD and MMD as RCL with EMD consistently outperforms RCL with BD and MMD. The focus of this work is to present the overall framework of RCL. Though the mathematical analysis on the choice of the best distance metric is beyond the scope of this work, based on the empirical results, EMD may be preferred in practical denoising applications.

[6] `https://scikit-image.org/docs/dev/api/skimage.metrics.html`

Table 1: Proxy evaluation of representation learning using denoising as the downstream task.

Method	MIT		Stanford		VOC	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
MoCo	12.87	0.0498	16.01	0.1505	14.21	0.1044
VADeR	14.63	0.1088	17.54	0.1601	16.33	0.1573
PixContrast	14.77	0.1101	17.61	0.1610	16.42	0.1585
N2N	28.66	0.8614	34.14	0.8699	30.91	0.8272
N2S	28.16	0.8373	34.04	0.8640	30.71	0.8256
RCL-BD	28.83	0.8871	34.75	0.8618	31.29	0.8274
RCL-EMD	**29.54**	**0.8908**	**35.43**	**0.8783**	**31.39**	**0.8330**
RCL-MMD	28.68	0.8864	34.87	0.8687	31.53	0.8316
Oracle	31.26	0.9187	38.25	0.9422	33.65	0.9038

4.2.3 SUPER RESOLUTION

Although the problem formulation in Sec. 3.1 is defined for noisy images, we also explore the generalization ability of the learned representations by investigating downstream tasks with *clean* input images. Super resolution (SR) commonly constitutes one such example task. As each image in the Stanford dataset has two resolutions (512×512 and 1024×1024), we simply use the higher resolution image as the upsampled ground truth. Following the proxy evaluation protocols introduced previously, results are presented in Table 2. We observe that RCL outperforms N2N and N2S by a large margin. We conclude that the representations learned by RCL from noisy images can also be utilized for clean images. By comparing Table 1 with Table 2, we find the performance gap between RCL and N2N (N2S) becomes larger, *i.e.* N2N and N2S tend to learn less meaningful representations for downstream problems when the problem differs from their pretext tasks, in the investigated setting. This phenomenon has also been discussed in (Zhang et al., 2016), where a task-dependent colorization-based SSL shows limited performance in image classification. Again, RCL with EMD consistently outperforms RCL with BD and MMD.

Table 2: Proxy evaluation of representation learning using SR and JDenSR as the downstream task on the Stanford dataset.

Method	SR		JDenSR	
	PSNR	SSIM	PSNR	SSIM
N2N	31.61	0.8730	27.90	0.7860
N2S	31.11	0.8699	27.80	0.7831
SL (Den)	34.18	0.9118	**32.89**	**0.8353**
RCL-BD	38.89	0.9654	32.10	0.8046
RCL-EMD	**39.01**	**0.9658**	32.63	0.8214
RCL-MMD	38.31	0.9634	31.95	0.8068
Oracle	38.93	0.9603	35.98	0.9175

4.2.4 JOINT DENOISING AND SUPER RESOLUTION

As a natural extension to independent denoising and SR tasks, we consider joint denoising and super resolution (JDenSR) as the downstream task, which has two sub-tasks and can further demonstrate the versatility of the learned representations. The results are presented in Table 2. Again, RCL outperforms the baseline SSL frameworks by a large margin. Notably, as EMD shows more robust performance than BD and MMD in three downstream tasks, we will use EMD as the default distance metric in the rest of experiments.

4.2.5 TRANSFERABILITY: SUPERVISED PRE-TRAINING *vs*. UNSUPERVISED PRE-TRAINING

In addition to unsupervised pre-training, we report the performance of supervised pre-training by denoising in the "SL (Den)" row of Table 2. We learn representations by applying SL to the denoising task, defined in Table 1. We then fine-tune to the alternative downstream tasks in a fashion identical to the considered SSL methods. We note that interestingly, RCL is able to outperform "SL (Den)" for the SR task and also RCL-EMD achieves higher performance than the *Oracle* in Table 2. This unintuitive phenomenon that unsupervised pre-training can improve performance over supervised pre-training, has been corroborated in CL studies that consider high-level vision tasks (Wang & Isola, 2020). While supervised pre-training tends to learn task-dependent representations, the representations learned by CL are more informative. In Table 2, the "SL (Den)" row, pertaining to

Table 3: Standard SL (left) and RCL pre-training with SL fine-tuning (right), evaluated with denoising on the VOC dataset. # Labels denotes the number of labeled data available for SL.

# Labels	SL		RCL + SL	
	PSNR	SSIM	PSNR	SSIM
0	-	-	22.62	0.7989
10	20.74	0.7299	28.20	0.8834
10^2	27.19	0.8734	30.24	0.9028
10^3	31.31	0.9184	32.09	0.9280
10^4	33.41	0.9437	33.85	0.9514

JDenSR task results, provides strong performance, and a marginal advantage over RCL, which may be explained by the fact that our JDenSR task can be considered closely related to a pure denoising task.

4.2.6 RCL *vs*. SL

To further quantify the performance and labelling-cost trade-off, we perform an ablation study. We train a U-Net in a supervised fashion (SL) for the denoising task using VOC data and compare this with RCL(-EMD) under various magnitudes of available training labels. We re-use the *same* random seeds for both methods. In the first row of Table 3, we report the performance of RCL by directly applying the representations pre-trained on the unlabelled training set, on the test set. In the remaining Table 3 rows, it can be observed that the performance gain obtained by pre-training with RCL grows larger as SL suffers more from label scarcity.

4.2.7 Label-Efficient Learning

Our ablation affords us some initial evidence towards answering the questions: *can RCL help SL?* and, if so, *when can RCL help?* We fine-tuned the U-Net, pre-trained by RCL(-EMD) on the entire training set, with additional paired RGB training images, as above. Pre-training with RCL consistently improves the performance of standard SL. In cases where labelled data are rare, expensive to collect or curate, such pre-training may be able to offer significant improvement (*e.g.* +7.46dB with only ten labels). We can also observe that improvement margins diminish as the number of labels available significantly grow (*e.g.* +0.44dB with $10,000$ labels).

4.2.8 Analysis on Residual Contrastive Loss

It is important to validate that RCL indeed learns from the residuals in the proposed formulation. To illustrate the learning outcome directly, we extract the residual tensors by using a U-Net trained on the MIT dataset with RCL-EMD. Given an anchor image, we calculate the pair-wise difference for EMD between a negative pair and EMD between a positive pair. Given the same network, we record the differences before the training starts (*i.e.* the weights are randomly initialized) and after the loss converges. The density plot of the differences is shown in Fig. 2. RCL contracts the predicted distribution closer to the true underlying distribution, where we use the sampled noise as the residual. We also find that large α in Eq. (10) will degrade the performance. We conjecture that this is because low-level vision tasks are sensitive to pixel-level perturbation. To provide an example: a very small change in the predicted pixel intensity can change the reconstructed pixel color but an analogous change in predicted pixel probability may not meaningfully change a segmentation result. RCL with large α can still learn representations, however the representations might not make sense for the demonstrated downstream tasks, discussed in Sec. 3.4.

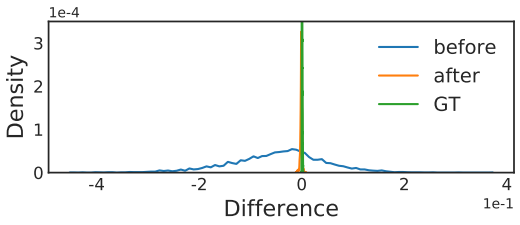

Figure 2: Comparison of residual contrastive loss (illustrated with EMD) before and after training. The density plot depicts the pair-wise differences of EMD between negative pairs minus EMD between positive pairs.

Table 4: Proxy evaluation of representation learning using JDD, JDemSR, and JDDSR as the downstream tasks on Stanford data.

Method	JDD		JDemSR		JDDSR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
SL (JDD)	-	-	**32.55**	**0.8727**	**30.70**	**0.8323**
M2M	26.62	0.7463	27.21	0.8219	26.07	0.7292
RCL	**27.37**	**0.7514**	28.00	0.8160	26.87	0.7331
Oracle	33.60	0.8979	36.63	0.9431	31.56	0.8490

4.2.9 CROP SIZE *vs*. BATCH SIZE

Under the constraint of limited computational resource, there will be a trade-off between crop size and batch size. Intuitively, a larger sample size should improve the estimation of the statistical distance. We do observe a performance drop of 2.06 dB in PSNR when RCL-EMD is trained with a crop size reduced from 128×128 to 64×64 on the Stanford dataset. However, increasing the batch size under crop size 64×64 also leads to a decrease in performance (*e.g.* -0.98 dB when batch size is increased from 64 to 128). This contradicts theoretical findings that imply a larger batch size is always preferred (Chuang et al., 2020). As (Chuang et al., 2020) focus on image classification problems where images are distinct, we conjecture that this phenomenon may be caused by the fact that different residual tensor pairs may have close scalar values by coincidence (*c.f*. a feature vector cosine similarity does not have this concern). A large batch size may also increase the chance of such coincidence, thus is less preferable.

4.3 EXPERIMENTS ON RAW DATA

4.3.1 BASELINES

We evaluate RCL with RAW data following a similar protocol as the RGB data. There is limited related work in the literature. We use *mosaic2mosaic* (M2M) (Ehret et al., 2019), a state-of-the-art SSL JDD framework, as the SSL baseline. In this section, we use RCL to denote RCL-EMD, which is reported to have the most robust performance in Sec. 4.2.

4.3.2 DOWNSTREAM TASKS

Following Sec. 4.2, the representations are evaluated with three downstream tasks on RAW data, which are joint demosaicing and denoising (JDD), joint demosaicing and super resolution (JDemSR), and joint demosaicing, denoising and super resolution (JDDSR). The results are evaluated on Stanford data and are presented in Table 4. While RCL outperforms M2M overall, the performance gap between RCL and the baseline becomes smaller while the gap between RCL and SL becomes larger, compared with Sec. 4.2. We believe this to be partially caused by the nature of the demosaicing task. The redefined residuals between $\boldsymbol{x}$ and $f_\theta(\boldsymbol{x})$ in Sec. 3.5 no longer have a pixel-level correspondence, which may hamper RCL from learning transferable representations.

4.4 LABEL-EFFICIENT LEARNING ON REAL NOISY DATA

4.4.1 PREPARATION

To illustrate the practical value of RCL, we apply it to denoising of real noisy camera data. In this experiment, we aim to show that RCL can be efficiently utilized to reduce the annotation cost with the learning paradigm of unsupervised pre-training followed by supervised fine-tuning. We use the RGB data from the Smartphone Image Denoising Dataset (SIDD) (Abdelhamed et al., 2018) where 80% of the data is considered as the training set and the final 20% is reserved for testing. The data is rearranged into patches of size 128×128. In order to show that the formulation of RCL generalizes well, we let f_θ be a DnCNN (Zhang et al., 2017), a state-of-the-art denoiser. While common approaches predict the reconstructed pixels (*e.g.* U-Net in Sec. 4.2), DnCNN predicts the residuals directly[7]. In contrast to Eq. (4), the residual term now takes the form of $\boldsymbol{x} - (\boldsymbol{x} - f_\theta(\boldsymbol{x}))$,

[7]Under this formulation, representations will have a lower generalization ability for other downstream tasks, *c.f*. UNet in Sec. 4.2

such that Eq. (5) and Eq. (9) still hold. We use a batch size of 64 and an Adam optimizer with a constant learning rate 10^{-3}.

4.4.2 RESULTS

Following the same procedure used to generate the results in Table 3, the goal is to utilize unsupervised pre-training with RCL to improve the performance of SL. As the baseline, we train a DnCNN with the full training set in a supervised fashion. For label-efficient learning, we first train another DnCNN with the full training set in an unsupervised fashion via RCL. Then, the DnCNN is fine-tuned with only 30% of labeled training set. The results are presented in Table 5. We find that, with only 30% of labeled data, label-efficient learning can outperform standard SL using the full training set and efficiently reduce the annotation cost. The results in Table 3 and Table 5 suggest that the representations pre-trained by RCL can slightly improve the performance while using only limited number of labels. We believe that this is due to the fact that, while standard SL could lead to overfitting, the representations learned by RCL generalizes well on the unseen data (the test set).

Table 5: Label-efficient learning on real noisy data, evaluated with denoising on the SIDD dataset. SL (left) denotes SL with trained with the full labeled training set. RCL + SL (right) denotes RCL pre-training with full unlabeled training set with SL fine-tuning with only 30% of labeled data.

SL		RCL + SL	
PSNR	SSIM	PSNR	SSIM
37.21	0.936	37.64	0.944

5 CONCLUSION

We present a principled unsupervised representation learning strategy which can learn transferable representations from images with additive noise for image reconstruction tasks. To the best of our knowledge, we are the first to formulate an instance-wise discrimination pretext task using image residuals and unify CL and ResL, for SSL on large-scale noisy data. The empirical studies validate the robustness and generalization of the representations learned by RCL, and further pose a new generic and label-efficient learning direction for low-level vision tasks. In the future, we will explore the efficacy of RCL for additional downstream tasks.

REFERENCES

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*, pp. 1692–1700, 2018.

Pierre Baldi and Fernando Pineda. Contrastive learning and neural oscillations. *Neural Computation*, 3(4):526–545, 1991.

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

Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11036–11045, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8765–8775, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Nanqing Dong, Michael Kampffmeyer, and Irina Voiculescu. Self-supervised multi-task representation learning for sequential medical images. In *European Conference on Machine Learning*, pp. 779–794. Springer, 2021.

Thibaud Ehret, Axel Davy, Pablo Arias, and Gabriele Facciolo. Joint demosaicking and denoising by fine-tuning of bursts of raw images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8868–8877, 2019.

Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88 (2):303–338, 2010.

Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008.

Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics*, 35(6):1–12, 2016.

Ryan D. Gow, David Renshaw, Keith Findlater, Lindsay Grant, Stuart J. McLeod, John Hart, and Robert L. Nicol. A comprehensive tool for modeling cmos image-sensor-noise performance. *IEEE Transactions on Electron Devices*, 54(6):1321–1329, 2007.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(1):723–773, 2012.

Samuel W Hasinoff, Frédo Durand, and William T Freeman. Noise-optimal capture for high dynamic range photography. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 553–560. IEEE, 2010.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016a.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pp. 630–645. Springer, 2016b.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

G.E. Healey and R. Kondepudy. Radiometric ccd camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):267–276, 1994.

Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pajak, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, et al. Flexisp: A flexible camera image processing framework. *ACM Transactions on Graphics*, 33(6):1–13, 2014.

Keigo Hirakawa and Thomas W. Parks. Joint demosaicing and denoising. *IEEE Transactions on Image Processing*, 15(8):2146–2157, 2006.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pp. 694–711. Springer, 2016.

Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981.

Daniel Khashabi, Sebastian Nowozin, Jeremy Jancsary, and Andrew W Fitzgibbon. Joint demosaicing and denoising via learned nonparametric random fields. *IEEE Transactions on Image Processing*, 23(12):4968–4981, 2014.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Filippos Kokkinos and Stamatios Lefkimmiatis. Deep image demosaicking using a cascade of convolutional residual denoising networks. In *European Conference on Computer Vision*, pp. 303–319, 2018.

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In *International Conference on Machine Learning*, pp. 2965–2974. PMLR, 2018.

Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In *European Conference on Computer Vision*, pp. 517–532, 2018.

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition workshops*, pp. 136–144, 2017.

Ce Liu, Richard Szeliski, Sing Bing Kang, C. Lawrence Zitnick, and William T. Freeman. Automatic estimation and removal of noise from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):299–314, 2008.

Xinhao Liu, Masayuki Tanaka, and Masatoshi Okutomi. Practical signal-dependent noise parameter estimation from a single noisy image. *IEEE Transactions on Image Processing*, 23(10):4361–4371, 2014.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

Markku Makitalo and Alessandro Foi. Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise. *IEEE Transactions on Image Processing*, 22(1):91–103, 2012.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

A.M. Mohsen, M.F. Tompsett, and C.H. Sequin. Noise measurements in charge-coupled devices. *IEEE Transactions on Electron Devices*, 22(5):209–218, 1975.

Pedro O O. Pinheiro, Amjad Almahairi, Ryan Benmalek, Florian Golemo, and Aaron C Courville. Unsupervised learning of dense visual representations. In *Advances in Neural Information Processing Systems*, volume 33, pp. 4489–4500. Curran Associates, Inc., 2020.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010.

Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1586–1595, 2017.

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*, pp. 234–241. Springer, 2015.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6827–6839, 2020.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Kaixuan Wei, Ying Fu, Jiaolong Yang, and Hua Huang. A physics-based noise formation model for extreme low-light raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2758–2767, 2020.

Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16684–16693, 2021.

Zongsheng Yue, Hongwei Yong, Qian Zhao, Lei Zhang, and Deyu Meng. Variational denoising network: Toward blind noise modeling and removal. In *Advances in Neural Information Processing Systems*, pp. 1690–1701, 2019.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3712–3722, 2018.

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.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*, pp. 649–666. Springer, 2016.

A Appendix

A.1 Noise Simulation

Typically, the shot and read noise parameters (λ_{shot}, λ_{read}) for any given camera sensor approximately follow a log-linear relationship in the parameter space (Foi et al., 2008; Brooks et al., 2019). The specific values of these parameters for a given camera and acquisition settings can be automatically obtained through automatic calibration procedures (Foi et al., 2008) and can be found in the metadata pertaining to the raw images for most cameras. In (Brooks et al., 2019), the authors use a number of (λ_{shot}, λ_{read}) samples to define the log-linear distributions of the noise in the Darmstadt Noise Dataset (DND) (Plotz & Roth, 2017). Such distribution is further perturbed along the log-line by adding a small Gaussian perturbation which models the specific variability of DND. For example, the simulation model for RAW data in (Brooks et al., 2019) is

$$\log(\lambda_{\text{shot}}) \sim \mathcal{U}(\log(0.001), \log(0.12)), \tag{14}$$

$$\log(\lambda_{\text{read}}) \sim \mathcal{N}(2.18 \log(\lambda_{\text{shot}}) + 1.2, 0.26). \tag{15}$$

In our case, we need to go one step further as we would like to fully randomize the relation of the noise parameters in the (λ_{shot}, λ_{read}) domain so as to simulate noise from different cameras and different acquisition settings. We randomly sample (λ_{shot}, λ_{read}) in the parameter space accordingly.

Given $\mathcal{U}(\sigma_{\min}, \sigma_{\max})$, where $\sigma^2_{\min}$ and $\sigma^2_{\max}$ are the minimum possible variance and maximum possible variance, the simulation model is

$$a = \sqrt{2(\frac{\sigma_{\min}}{2})^2}, b = \sqrt{2(\frac{\sigma_{\max}}{2})^2}, \tag{16}$$

$$\sqrt{\lambda_{\text{shot}}} \sim \mathcal{U}(a, b), \sqrt{\lambda_{\text{read}}} \sim \mathcal{U}(a, b). \tag{17}$$

The noise simulation model presented above will sample (λ_{shot}, λ_{read}) for Eq. (3). The minimum possible variance $\sigma^2_{\min}$ and the maximum possible variance $\sigma^2_{\max}$ enable us to fully control the strength of the noise applied to the input. Note that our parameterization, in contrast to (Brooks et al., 2019), will not constrain (λ_{shot}, λ_{read}) to follow a pre-defined log-linear relationship, but instead can sample the parameters at arbitrary positions in the domain hence naturally enabling us to simulate noise from different camera sensors.

A.2 SIMULATION DATASETS

The three datasets contain RGB images with varying semantic information and image quality. MIT is a benchmark demosaicing dataset, containing images that are collected from the web with large variety in terms of object categories and scenes. Stanford is a benchmark multi-task learning (MTL) dataset, which consists of various indoor scenes. VOC is semantic segmentation dataset of 20 semantic categories. The images of MIT and VOC have various resolutions, while the images of Stanford have two fixed resolutions 512×512 and 1024×1024. Limited by available resources, we select two subsets of MIT and Stanford, and VOC to validate our ideas. The MIT subset contains 11000 images[8], divided into 10000 images in the training set and 1000 images in the test set. The Stanford subset contains 9464 images[9], and we use 8464 images for training and 1000 images for testing. VOC contains 17125 images[10], including 16125 for training and 1000 for testing. The minimal crop size is 128×128 for the Stanford dataset and only 64×64 for the MIT and VOC datasets, limited by the original image size.

A.3 REAL DATASET

The Smartphone Image Denoising Dataset (SIDD) (Abdelhamed et al., 2018) is a camera image dataset with real additive noise[11]. SIDD contains 160 noisy and clean image pairs, with ten individual scenes repeatedly captured using five different smartphone cameras. We use the SIDD-Medium Dataset, which is a subset of SIDD. Compared with the simulated signal-dependent data, SIDD has less variation in terms of $(\lambda_{\text{shot}}, \lambda_{\text{read}})$.

[8] https://groups.csail.mit.edu/graphics/demosaicnet/dataset.html

[9] https://github.com/alexsax/taskonomy-sample-model-1

[10] http://host.robots.ox.ac.uk/pascal/VOC/

[11] https://www.eecs.yorku.ca/~kamel/sidd/index.php