Monte-Carlo Siamese Policy on Actor for Satellite Image Super Resolution

Litu Rout[1] Saumyaa Shah[2] S Manthira Moorthi[1] Debajyoti Dhar[1]

[1] Signal and Image Processing Group
Space Applications Centre
Indian Space Research Organisation
(lr, smmoorthi, deb)@sac.isro.gov.in

[2] Work done at Space Applications Centre
saumyaashah2498@gmail.com

Abstract

In the past few years supervised and adversarial learning have been widely adopted in various complex computer vision tasks. It seems natural to wonder whether another branch of artificial intelligence, commonly known as Reinforcement Learning (RL) can benefit such complex vision tasks. In this study, we explore the plausible usage of RL in super resolution of remote sensing imagery. Guided by recent advances in super resolution, we propose a theoretical framework that leverages the benefits of supervised and reinforcement learning. We argue that a straightforward implementation of RL is not adequate to address ill-posed super resolution as the action variables are not fully known. To tackle this issue, we propose to parameterize action variables by matrices, and train our policy network using Monte-Carlo sampling. We study the implications of parametric action space in a model-free environment from theoretical and empirical perspective. Furthermore, we analyze the quantitative and qualitative results on both remote sensing and non-remote sensing datasets. Based on our experiments, we report considerable improvement over state-of-the-art methods by encapsulating supervised models in a reinforcement learning framework.

1. Introduction

Despite significant progress in complex environments [39, 40, 31], Deep Reinforcement Learning (DRL) has not received much needed attention from remote sensing community. In this study, we intend to bridge this gap by providing a DRL framework to tackle single image super-resolution in the context of satellite image processing.

Reinforcement Learning (RL) is a sequential decision making process that focuses on maximizing long-term expected return by interacting with the environment iteratively [44]. In the process of maximizing reward, function approximation plays a vital role [45]. In the recent years, several function approximators have been proposed to estimate key ingredients of RL: action value function (Q-function) and state value function (V-function). These functions are estimated differently in two broader categories of reinforcement learning methods: model-based and model-free.

Both model-based and model-free methods have their own merits and demerits. Model-based methods build a representative model of the environment and then sample rewards and transitions based on this approximation to estimate value functions. As per recent studies [32], model-based methods are more efficient in discrete environment due to low sample complexity. On the other hand, model-free methods learn the value functions by rollout samples obtained directly via interaction with the environment [44]. While model-free methods are suitable for continuous and complex Markov Decision Processes (MDPs) in general, these methods suffer from high sample complexity [33]. In this study, we primarily focus on model-free reinforcement learning as we do not have a model of the super-resolution environment.

Model-free control algorithms, such as Monte-Carlo (MC) and Temporal Difference (TD) have shown appealing results in numerous decision making processes [44, 20]. In most of these MDPs, there are two commonly used function approximators: value based and policy based. While value based function approximation is efficient in low dimensional or discrete action space, it does not scale well to continuous action space. In addition, it is less effective in learning stochastic policies. On the contrary, the policy based approximators can learn stochastic policies in high dimensional continuous action space. However, the policy based methods typically converge to local rather than global optimum and have high variance in the estimation.

Several policy based algorithms have been proposed over the years, such as REINFORCE [52], Actor-Critic [50, 21], Proximal Policy Optimization [43, 42], and Supervised Policy Update [48]. A straightforward implementation of these algorithms is not sufficient for single image super-resolution. Most of these algorithms require adequate information about the action space in order to estimate an optimal policy. For instance, the control variables are known in most continuous action space based RL environments, though the values of these variables are estimated based on stochastic policy. Contrary to that, there are several real world control problems in which it is difficult to explicitly model the variables of action space. In such environments, we propose a way to find optimal control policy by allowing the agent to take parametric actions. To put more succinctly, the action variables are represented by matrices and the values within the matrices characterize magnitude of that action. Of particular interest, reinforcement learning based super resolution is one such environment where an actor is expected to transit from a low resolution state to a high resolution state via sequence of actions. In this environment, the sequential parametric actions are taken by shallow neural networks and a reward is received only at the end of an episode provided the terminating state falls within an ϵ-ball of the high resolution state. In this process, the policy network guides the agent by providing probabilistic confidence on the performed actions in each episode.

Many researchers have applied DRL in challenging computer vision problems [10, 11]. Caicedo *et al.* [10] used a set of actions as a part of sequential decision making process and rewarded the agent when the transformed bounding box had optimum overlap with target bounding box. Cao *et al.* [11] exploited global inter-dependency of images to hallucinate missing high frequency details using attention-aware RL agent. Our work differs from these previous approaches in a sense that we do not use explicit set of action variables, such as move left or move right with continuous action values, i.e., the amount of movement in these directions. Instead, we use parametric action variables which allows to perform relevant actions with continuous action values. The underlying hypothesis is that a particular combination of these parameters may lead to a certain action which otherwise would not have been apparent in discrete action space. In other words, a particular combination may allow the agent to take an action that results in edge detection, or another combination may give rise to color feature extraction. Here, the action variable could be edge detection with sharpness of these edges represented by action values. Thus, our primary contribution in this study is to propose a novel reinforcement learning framework to perform the complex task of super resolution. Further, we intend to provide theoretical and empirical evidence of the proposed framework that is shown to outperform state-of-the-art methods in remote sensing.

The rest of the paper is organized as follows. In Section 2, we briefly discuss about prior and concurrent works done to address the problem under investigation. Section 3 contains the preliminary settings of DRL followed by Section 4 which describes the proposed method in detail. We provide theoretical evidence in Section 5 along with empirical experiments in Section 6. At the end, we draw concluding remarks and suggest future line of research in Section 7.

2. Related Work

Recent advances in deep learning has created a surge in single image super resolution. Starting with the pioneering work of Dong *et al.* [14], deep learning based super resolution has been actively explored and often outperforms state-of-the-art methods on various benchmark datasets [37, 4, 56, 6, 24, 46, 1, 7]. Thereafter, Lai *et al.* [28] proposed a deep Laplacian pyramid network for fast and accurate super resolution. It progressively upsampled the coarse resolution band to decompose the difficult task into relatively simple sub-problems. Among other supervised learning framework, Anwar *et al.* [2] proposed Densely Residual Laplacian Network (DRLN) that achieved state-of-the-art results on almost all benchmarks. Further, Ledig *et al.* [30] introduced an adversarial framework to push the reconstructed images towards natural manifold of realistic data. Wang *et al.* [49] improved upon this idea and designed a generative model which achieved higher perceptual quality. A detailed discussion on recent developments in super resolution can be found in [51, 3].

Remote sensing image super resolution is becoming increasingly popular. Particularly intriguing is the complex spatial distribution of remote sensing imagery which makes super resolution a relatively hard problem. Beyond academic interests, multi-band images are especially useful in wide variety of domains including agriculture [27], surveillance [47] and land cover classification [13]. In addition, the fundamental ideas developed in computer vision community are becoming prevalent in certain applications based on remotely sensed multi-band imagery [5, 15, 36, 19, 25].

After decades of research devoted in supervised and adversarial learning, it seems natural to wonder whether another branch of artificial intelligence, namely Reinforcement Learning (RL) would benefit the super resolution community. There has been little study of DRL in single image super resolution [53, 54]. Yu *et al.* [53] have taken a step along this interesting direction of research by dynamically selecting a toolchain for progressive restoration. Further, Yu *et al.* [54] devised a framework by combining deep learning with REINFORCE to restore non-remote sensing images under noisy environment. In this study, we further explore the plausible usage of DRL in the context of satellite image super resolution. We provide theoretical results supported by experimental evidence to corroborate our hypothesis.

3. Preliminaries and Notations

A Markov Decision Process (MDP) is defined as a tuple (S, A, R, P, γ), where S is the continuous or discrete state space, A is the continuous or discrete action space, R is the immediate reward function, P is the transition probability, and $\gamma \in (0, 1)$ is the discount factor. The goal of an agent is to find an optimal policy π^* that maximizes its expected reward,

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{J}(\pi), \tag{1}$$

where Π is the set of policies and $\mathcal{J}(\pi)$ is the policy evaluation metric defined by,

$$\mathcal{J}(\pi) = \mathbb{E}_{\tau \in \pi}\left[\sum_{t=1}^{T+1} \gamma^{t-1} r_t\right]. \tag{2}$$

Here, T represents the time step of terminal state in each episode and τ is the trajectory, $(s_0, a_0, r_1, s_1, a_1, r_2, \ldots, s_T, a_T, r_{T+1})$ sampled from policy π. As we are focusing on model-free approaches, the state and action value functions (V and Q) are approximated based on sampled trajectories unlike model-based approaches where full width backup is taken into consideration as the transition dynamics is accessible. In common policy optimization strategy, the policy π is parameterized by θ where the objective is to find optimal set of parameters θ^* that maximizes expected reward,

$$\theta^* = \arg\max_{\theta} \mathcal{J}(\theta), \tag{3}$$

$$\mathcal{J}(\theta) = \sum_{s \in S} d^{\pi_\theta}(s) \sum_{a \in A} \pi_\theta(s, a) R_{s,a}, \tag{4}$$

where $d^{\pi_\theta}(s)$ is a stationary distribution of Markov chain for π_θ and $R_{s,a}$ is the reward function for state s and action a. The policy parameters are updated by $\theta \leftarrow \theta + \Delta\theta$, where $\Delta\theta$ is computed by the famous likelihood trick [52],

$$\Delta\theta = \nabla_\theta \mathcal{J}(\theta) = \mathbb{E}\left[R_{s,a} \nabla_\theta \log \pi_\theta(s, a)\right]. \tag{5}$$

There are several variants of equation (5) that emphasize on faster convergence to optimal solution and robust policy estimation. To study and analyze the proposed idea at a fundamental level, we choose a simple and effective policy gradient strategy, namely MC-REINFORCE [52] as given in equation (5). However, the proposed approach is not limited to MC-REINFORCE, and would certainly benefit from recent advances in policy optimization.

4. Methodology

The idea of parameterizing action space/variables is inspired by the notion of building a model of the environment in model-based RL. In a model-based RL, the transition dynamics (P) and reward function (R) are parameterized assuming that the state space (s) and action space (a) are known. The proposed approach is slightly different from this model-based approach in a sense that we parameterize the action space (a) and learn the policy in a model-free way using MC sampling.

4.1. Representation Learning

Here, we discuss about efficient representation of each state in our MDP as it plays a vital role in solving MDPs [44]. Instead of naively representing each state, we use Convolutional Neural Network (CNN) as feature extractor due to its tremendous success in learning latent representation. The feature extractor network, $\Phi(s)$ parameterized by θ_f operates on each state, $s \in \mathbb{R}^{H \times W \times C}$,

$$\tilde{s} = \Phi(s; \theta_f), \ \tilde{s} \in \mathbb{R}^{H \times W \times \tilde{C}}, \tag{6}$$

where H, W, C, and $\tilde{C}$ represent height, width, input channels, and number of feature maps, respectively. The output of neural network, $\Phi(s; \theta_f)$ is computed by,

$$\Phi(s; \theta_f) := FE_n\left(FE_{n-1}\left(\ldots\left(FE_0\left(s\right)\right)\right)\right), \tag{7}$$

where FE represents Feature Extraction block consisting of one convolution and one LeakyReLU unit. Here, n represents number of FE blocks.

4.2. Actor Network

The Actor Network (AN), $\Omega_{\theta_a}(.)$ parameterized by θ_a performs parametric actions on the latent representation of state space, $\tilde{s}$. Each action is parameterized by a shallow neural network consisting of a single Residual Block (RB). To span the dynamic range of each state, we use a customized RB, as given by equation (8), in contrast to the one proposed in [22].

$$RB(x) = x + \lambda h(x), \tag{8}$$

where $h(x)$ is a sequential neural network consisting of $\{$convolution, ReLU, and convolution$\}$ units. Here, λ represents the scaling factor. The agent performs sequence of actions, $a_n^{RB}(.)$ and the intermediate states are computed by,

$$\tilde{s}_n = a_n^{RB}(\tilde{s}_{n-1}), \ n = 1, 2, \ldots, N, \tag{9}$$

where N represents total number of action variables in our MDP. Here, $\tilde{s}_0$, $\tilde{s}_n$, and $\tilde{s}_N$ represent the latent representation of the initial, intermediate, and arrived state, respectively. Different combinations of these parameters present in each kernel of these action variables lead to different actions necessary to achieve the desired goal. The latent representation of arrived state, $\tilde{s}_N$ is passed through a cascade of Transition Blocks (TB) in order to map latent space into state space, as given in equation (10).

$$\hat{s} = TB_m\left(TB_{m-1}\left(\ldots\left(TB_0\left(\tilde{s}_N\right)\right)\right)\right) \qquad (10)$$

Here, $\hat{s} \in \mathbb{R}^{H \times W \times C}$, and each TB consists of one convolution and one LeakyReLU unit. Since the agent receives reward at the end of each episode, we only convert the final latent representation, $\tilde{s}_N$ to state space, $\hat{s}$ for minimizing time complexity.

4.3. Siamese Policy Network

Here, we provide a justification for parameterizing policy network using Siamese architecture. The standard policy network, $\pi_{\theta_p}(s, a)$ parameterized by θ_p provides a distribution over actions, a given a state, s. Thus, the policy network provides a probabilistic view of how good it is to take an action at a given state. In other words, it imposes a confidence on the agent's actions at a particular state. If the sequence of actions triggers a transition such that the final state falls within an ϵ-ball of the goal state, then the confidence level on agent's actions is enhanced. In such scenarios, the policy gradient approach increases the likelihood of taking these relevant actions.

To estimate the confidence on agent's actions, we propose to use Siamese neural network architecture [9]. The Siamese Policy Network (SPN), $\Psi_{\theta_p}(\hat{s}, s^*)$ measures the discrepancy between arrived state, $\hat{s}$ and goal state, s^*. The SPN is stochastic in nature due to which it does not require the environment to be noisy in order to perform sufficient exploration. The two branches of Siamese neural network take $\hat{s}$ and s^* as inputs, projects them into feature space using shared parameters across both branches, and correlate them in feature space to better estimate their discrepancy,

$$\Psi_{\theta_p}(\hat{s}, s^*) = \Phi_{\theta_p}(\hat{s}) * \Phi_{\theta_p}(s^*) + b, \qquad (11)$$

where $b \in \mathbb{R}$ and $\Phi_{\theta_p}(.)$ represents the CNN in each branch with shared parameters θ_p. The probabilistic confidence is then computed by sigmoidal activation unit,

$$\pi_{\theta_p}(s, a) = \frac{1}{1 + \exp(-\Psi_{\theta_p}(\hat{s}, s^*))}. \qquad (12)$$

The arrived state $(\hat{s})$ is a function of implicit actions, a containing $a_n^{RB}, n = 1, 2, \ldots N$. In the super-resolution environment, the model receives reward only at the final state based on its Euclidean distance from target state.

4.4. Siamese Policy On Actor

The proposed method, which we call Siamese Policy On Actor (SPOA), encapsulates representation learning, AN, and SPN to provide an end to end DRL framework for image super resolution. Motivated by the findings of Goodfellow *et al.* [18], we propose to allow two networks, namely actor and policy to supplement each other in the learning process so as to find a global optimum.

In the current setting, we consider occurrence of each observable state to be equally likely, i.e., $d^{\pi_\theta}(s) \sim \mathbb{U}$, where $\mathbb{U}$ denotes uniform distribution. We define reward function, $R_{s,a}$ as negative mean squared error between $\hat{s}$ and s^*. The agent receives reward at the end of an episode and it is maximum when $\|\hat{s} - s^*\|^2$ falls within an ϵ-ball around s^*. We use $n = 3$ FE blocks in representation learning, $N = 3$ RBs, $m = 3$ TBs in AN, and 3 blocks of {convolution, LeakyReLU} units in SPN.

5. Theoretical Results

In this section, we elaborate on the supplementary training procedure. We independently train AN and SPN in a least expensive way before training SPOA. Thus, we ensure that the arrived state does not reside far away from initial state, which otherwise would make it unattainable.

Lemma I: *Training AN*
Let $\theta_{fa} = \{\theta_f, \theta_a\}$ and $\mathcal{J}(\theta_{fa})$ denote the expected return accumulated by the agent with a given policy π_{θ_p},

$$\mathcal{J}(\theta_{fa}) = \mathbb{E}\left[R_{s,a}\right] = \mathbb{E}\left[-(\hat{s} - s^*)^2\right]. \qquad (13)$$

The parameters are updated by, $\theta_{fa} \leftarrow \theta_{fa} + \Delta\theta_{fa}$ where,

$$\Delta\theta_{fa} = \mathbb{E}\left[-2\left(\hat{s} - s^*\right)\ \nabla_{\theta_{fa}}\left(TB_{[m]}\left(\Omega_{\theta_a}\left(\Phi_{\theta_f}\left(s\right)\right)\right)\right)\right]. \qquad (14)$$

Here, $[m]$ represents a set of $\{0, 1, \ldots, m\}$.

By stochastic gradient ascent, the update equation (14) becomes

$$\Delta\theta_{fa} = -\alpha\left(\hat{s} - s^*\right)\nabla_{\theta_{fa}}\left(TB_{[m]}\left(\Omega_{\theta_a}\left(\Phi_{\theta_f}\left(s\right)\right)\right)\right), \qquad (15)$$

where α denotes step size.

Lemma II: *Training SPN*
Let $\mathcal{J}(\theta_p)$ denotes the expected return accumulated by the agent with fixed set of parameters (θ_{fa}), then

$$\mathcal{J}\left(\theta_p\right) = \mathbb{E}_{\theta_p}\left[r\right] = \sum_{s \in S} d^{\pi_\theta}\left(s\right) \sum_{a \in A} \pi_\theta\left(s, a\right) R_{s,a}. \qquad (16)$$

Using stochastic gradient ascent, the parameters are updated using the famous likelihood trick [44] as given by

$$\theta_p \leftarrow \theta_p + \Delta\theta_p,\ \Delta\theta_p = \nabla_{\theta_p}\mathcal{J}(\theta_p) = \beta R_{s,a}\nabla_{\theta_p}\log\pi_\theta\left(s, a\right), \qquad (17)$$

where β denotes step size.

Theorem I: *Training SPOA*
Let $\theta = \{\theta_f, \theta_a, \theta_p\}$ and $\mathcal{J}(\theta)$ denotes the expected return. The parameters of SPOA (θ) are updated by $\theta \leftarrow \theta + \Delta\theta$ where,

$$\Delta\theta = \Delta\theta_p + \Delta\theta_{fa}. \qquad (18)$$

Proof:

$$
\begin{aligned}
\mathcal{J}(\theta) &= \mathbb{E}\left[r\right] = \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} \pi_{\theta_p}\left(s,a\right) R_{s,a} \\
\Delta\theta &= \nabla_\theta \mathcal{J}(\theta) \\
&= \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} \nabla_\theta \left(\pi_{\theta_p}\left(s,a\right) R_{s,a}\right) \\
&= \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} \left(R_{s,a} \nabla_\theta \pi_{\theta_p}\left(s,a\right) + \pi_{\theta_p}\left(s,a\right) \nabla_\theta R_{s,a}\right) \\
&= \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} R_{s,a} \nabla_{\theta_p} \pi_{\theta_p}\left(s,a\right) \\
&\quad + \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} \pi_{\theta_p}\left(s,a\right) \nabla_{\theta_{fa}} R_{s,a} \\
&= \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} R_{s,a} \pi_{\theta_p}\left(s,a\right) \nabla_{\theta_p} \log \pi_{\theta_p}\left(s,a\right) \\
&\quad + \sum_{s\in S} d^{\pi_{\theta_p}}\left(s\right) \sum_{a\in A} \pi_{\theta_p}\left(s,a\right) \nabla_{\theta_{fa}} R_{s,a} \\
&= \mathbb{E}\left[R_{s,a} \nabla_{\theta_p} \log \pi_{\theta_p}\left(s,a\right)\right] + \mathbb{E}\left[\nabla_{\theta_{fa}} R_{s,a}\right]
\end{aligned}
$$

Using stochastic gradient ascent, $\Delta\theta$ becomes,

$$
\begin{aligned}
\Delta\theta &= \beta R_{s,a} \nabla_{\theta_p} \log \pi_{\theta_p}\left(s,a\right) \\
&\quad - \alpha\left(\hat{s} - s^*\right) \nabla_{\theta_{fa}} \left(TB_{[m]}\left(\Psi_{\theta_a}\left(\Phi_{\theta_f}\left(s\right)\right)\right)\right).
\end{aligned} \tag{19}
$$

From ***Lemma I*** and ***Lemma II***, equation (19) becomes,

$$
\Delta\theta = \Delta\theta_p + \Delta\theta_{fa}. \tag{20}
$$

The pseudo code of SPOA is given in **Algorithm 1**. We sample batches of experiences and use replay buffer to update the SPOA parameters for the entire batch.

```
Result: SPOA parameters, $\theta$
initialize $\theta$;
for episode = 1,2,...,E do
    initialize empty replay buffer $\mathbb{D}$;
    while $\mathbb{D}$ not full do
        Sample initial state, $s_0 \sim \mathbb{U}$;
        Sample corresponding goal state, $s^*$;
    end
    for actor=1,2,...,A do
        Take parametric sequential actions on $\mathbb{D}$;
        Compute $\Delta\theta_{fa} =$
          $-\alpha\left(\hat{s} - s^*\right) \nabla_{\theta_{fa}} \left(TB_{[m]}\left(\Omega_{\theta_a}\left(\Phi_{\theta_f}\left(s\right)\right)\right)\right)$;

        Update $\theta_{fa} \leftarrow \theta_{fa} + \Delta\theta_{fa}$;
    end
    for policy=1,2,...,P do
        Given actor parameters $\theta_{fa}$, follow
          parametric policy $\pi_{\theta_p}(s,a)$ on $\mathbb{D}$;
        Compute $\Delta\theta_p = \beta R_{s,a} \nabla_{\theta_p} \log \pi_\theta\left(s,a\right)$;
        Update $\theta_p \leftarrow \theta_p + \Delta\theta_p$;
    end
    for spoa=1,2,...,S do
        Follow policy with implicit actions on $\mathbb{D}$;
        Compute new $\Delta\theta_p$ and $\Delta\theta_{fa}$;
        Compute $\Delta\theta = \Delta\theta_p + \Delta\theta_{fa}$;
        Update $\theta \leftarrow \theta + \Delta\theta$;
    end
end
```

Algorithm 1: Monte-Carlo Siamese Policy On Actor

6. Experiments

6.1. Datasets and Study Area

Here, we describe the datasets used to analyze the performance of the proposed methodology in two folds. First, we develop the theoretical foundation, and validate the pipeline experimentally on 1000 images of CelebA [34] and 3000 patches (64x64) of Indian Remote Sensing satellite (IRS-1C). While these datasets help us verify the efficacy of proposed theoretical formulation, it is hard to infer the generalization ability from them. For this reason, we extend our analysis to large scale remote sensing images of WorldView-2. With 80-20 split we use 40000 patches of WorldView-2 over Washington having Ground Sampling Distance (GSD) 1.84m.

6.2. Implementation Details

Data Preparation: In this study, the scaling factor between Low Resolution (LR) and High Resolution (HR) images is set to 4x. To prepare training data, we crop 64x64 patches from the training HR images. Following Dong *et al.* [14], the LR training patches are obtained by downsampling the HR patches by a factor of 4 using bicubic kernel. For data augmentation, we randomly choose one of the following techniques: rotation by 90 degree, horizontal flips or vertical flips.

Network Architecture: During development stage, we use SRCNN [14] as the backbone of the actor network to establish the theoretical foundation. To assimilate the performance of SPOA built upon deeper architectures, we explore various state-of-the-art methods. Motivated by recent advances, we use DRLN [2] with a network depth of 4 cascading residual-in-residual blocks in our final AN. For the policy network, we use convolutional layers with kernel size (3,3), followed by LeakyReLU activation with a negative slope of 0.1. All trainable parameters are initialized using Xavier method [17].

Training Details: The interpolated image and the corresponding high resolution image represent the initial and goal state in our MDP, respectively. In each episode, we sample from uniformly distributed initial state space. We set total episodes to 100000 and replay buffer size, $\mathbb{D}$ = 10. For training, we use different step sizes, i.e., $\alpha = 1e-4$ & $\beta =$

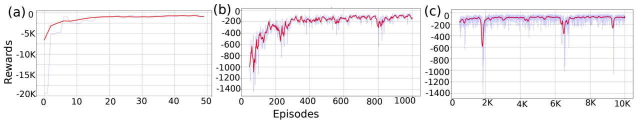

Figure 1. Learning dynamics. The blue curve shows actual reward gathered per episode. The red curve shows windowed average of actual reward per episode. We use a forward window of size 10.

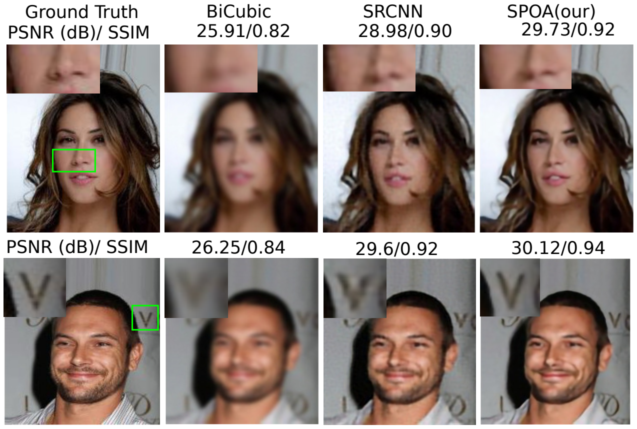

Figure 2. Qualitative analysis on CelebA. Comparison with existing high resolution data. The proposed DRL based approach, SPOA performs favourably against compared approaches.

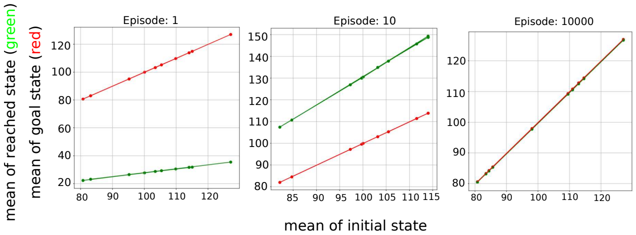

Figure 3. Mean state transition analysis. The implicit actor reaches goal state from almost every initial state.

$1e-7$ and Adam optimizer [26]. A least expensive solution is chosen to update AN, SPN, and SPOA, i.e, $A = 1$, $P = 1$, and $S = 1$. We use a common system configuration with 2x Tesla K40 in all our experiments. The SPOA learning algorithm has been implemented with PyTorch [41].

Evaluation metrics: The resultant super resolved images are evaluated using four commonly used image quality metrics: PSNR [23], SSIM [23], SRE [29], and SAM [55]. However, according to Ledig *et al.* [30], these distortion measures fundamentally go against human perception of image quality. For this reason, to assimilate perceptual quality, we use no reference measures like NIQE [38] and Ma's score [35]. Using these measures, we compute the Perceptual Index (PI) of an image as specified in the PIRM-SR challenge [7].

6.3. Analysis on CelebA

Figure 1 shows gradual development in accumulating rewards over multiple episodes. The agent initially performs random actions, which are sampled from the proposed stochastic SPN, in the form of exploration. This is evident from Figure 1(a), where the accumulated reward is not so surprisingly very low in the earlier episodes. The agent however discovers relevant actions as the interaction with the environment progresses and simultaneously, SPOA increases the likelihood of these particular actions. The agent, therefore, observes a steady growth in gathering rewards, as shown in Figure 1(b). Once the agent figures out relevant actions, it repeatedly performs those actions in order to accumulate maximum rewards. Thereby, it reaches in the proximity of goal state from most of the initial states in almost every episodes, as shown in Figure 1(c).

Further, we compare SPOA with BiCubic and SRCNN [14] in both training and testing datasets. It is worth mentioning that both SRCNN and SPOA share similar architecture to assert direct comparison between these two learning algorithms. Nevertheless, one can implement more sophisticated architectures in the proposed framework to gain optimal benefits. In Figure 2, we compare the reconstructed images of BiCubic, SRCNN and SPOA both qualitatively and quantitatively using PSNR (dB) and SSIM. As per the analysis, the reconstructed image by SPOA has higher structural similarity with existing high resolution data and also, it contains relatively less noise in each pixel. Further, this shows the efficacy of hierarchical composition of local constituent functions, such as implicit actor networks to learn a compact continuous mapping.

6.4. Analysis on IRS-1C

Here, we analyze the performance of SPOA on IRS-1C imagery. As shown in Figure 3, the distance between initial state and goal state gradually decreases as the training progresses. Finally, the implicit actor reaches at the corresponding goal state starting from almost every initial state. In addition, Figure 4 shows consistent improvement of SPOA over SRCNN both qualitatively and quantitatively.

6.5. Comparison with State-of-the-art

To gain further insight about generalization ability of SPOA, we start our discussion by comparing the proposed framework with state-of-the-art methods. It is to be noted that we use SRCNN in SPOA for the sole purpose of building overall pipeline. However, our final framework is built upon DRLN [2]. To discern the usefulness of the proposed method, we study its performance on large scale remote sensing imagery of WorldView-2. In this regard, we analyze the super-resolved images in terms of both perception and distortion metrics [8]. As given in Table 1, DRLN achieves state-of-the-art result among prior approaches on WorldView-2. While the proposed SPOA(DRLN) performs

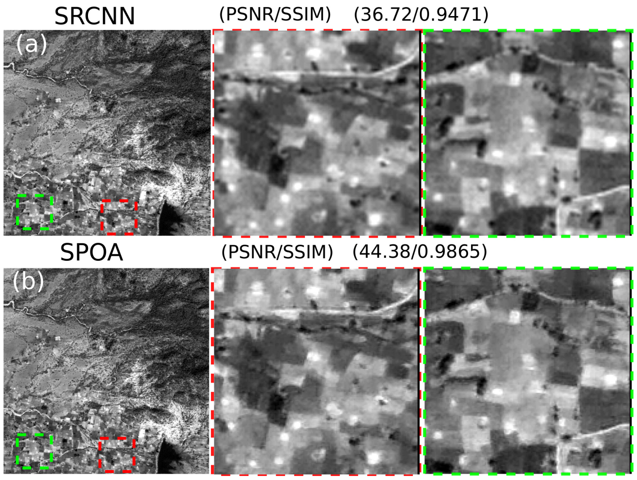

Figure 4. Qualitative analysis on IRS-1C. SPOA performs reasonably well on IRS-1C imagery.

Metrics	PSNR	SSIM	SRE	SAM	NIQE	Ma's	PI
BiCubic	57.51	0.9939	46.48	17.25	5.50	3.77	5.86
SRCNN [14]	59.15	0.9964	48.10	14.14	5.73	4.88	5.42
LapSRN [28]	59.31	0.9964	48.08	13.98	5.08	5.96	4.56
DRLN [2]	59.32	0.9964	48.10	13.97	4.21	6.03	4.08
SPOA(DRLN)	58.89	0.9960	47.94	14.69	3.65	6.60	3.52
SPOA(DRLN)+SA	59.33	0.9966	48.20	13.81	5.02	5.54	4.74
SPOA(DRLN)+SA+VGG	59.22	0.9963	48.23	14.13	4.30	6.20	4.05
SPOA(DRLN)+VGG	58.98	0.9961	47.94	14.60	4.16	6.56	3.80
GT	-	-	-	-	2.05	7.01	2.52

Table 1. Comparison with state-of-the-art methods.

sub-optimally in terms of distortion metrics, it achieves higher perceptual quality in terms of NIQE, Ma's score, and PI. Consistent with the theoretical justification of Blau *et al.* [8], we observe perception-distortion tradeoff similar to adversarial networks [49]. Moreover, the perception metric values of SPOA(DRLN) are relatively closer to Ground Truth (GT) as compared to state-of-the-art methods. Since SPOA derives its foundation from reinforcement learning paradigm, which is quite different from adversarial learning, it will certainly be interesting to study the theoretical basis of such similarity in perception-distortion tradeoff [8].

6.5.1 Ablation Study

In addition, we explore several variants of SPOA to gain intuition about its ability to achieve better distortion quality. We start our discussion by incorporating Self-Attention (SA) units [57] in SPOA. Attending to relevant parts of an image is an interesting line of research. Recent study shows significant improvement in image quality due to attention mechanisms [12, 57, 16]. For this reason, we augment SPOA by adding self-attention units that work in tandem with existing Laplacian channel attention units. As given in Table 1, SPOA(DRLN)+SA outperforms DRLN in terms of distortion metrics. Furthermore, we study whether addition of VGG loss [30] results in better perceptual quality. Though introduction of VGG loss does not boost performance beyond SPOA, it certainly improves the perceptual quality of SPOA(DRLN)+SA.

6.5.2 Analysis on WorldView-2

To this end, we studied quantitatively how reinforcement learning driven SPOA benefits super resolution. Here, we discuss further by correlating the perception-distortion metrics with qualitative measures. As can be inferred from the Natural Color Composite (NCC) and individual bands in Figure 5, SPOA outperforms state-of-the-art methods in terms of perceptual quality. While compared methods lack continuity in linear features, SPOA seems to preserve continuity reasonably well. Even though SSIM values are comparable, evidently the quality of super resolved images is not at par with each other. This is consistent with the observation of Blau *et al.* [8]. It can be observed from Figure 5 that the sharpness and continuity of features are prominent in SPOA. In addition, SPOA produces more high frequency details which tend to improve the naturalness of super resolved images while other methods [28, 2] either fail to capture these details or introduce unwanted artifacts. From the individual bands of various signatures in Figure 5, one can obtain a better visual assessment of image quality.

7. Concluding Remarks and Future Scope

In this study, we explored the plausible usage of reinforcement learning to address complex supervised learning problems. We designed a DRL based Monte-Carlo policy gradient approach to solve model-free MDPs where adequate information about action variables is not discernible. Guided by our theoretical justification, we introduced a Siamese policy network with implicit action space. Further, we demonstrated the efficacy of the proposed method in a super resolution environment where action variables are not apparent. Using both remote sensing and non-remote sensing imagery, we studied the perception-distortion tradeoff. To satisfy the requirement on multitude of tasks, we introduced two methods: one that achieved state-of-the-art results in distortion and another, in visual perceptual quality.

A few noteworthy extensions of this paper are as follows:

1. *Extension* of SPOA to wide variety of problems currently solved using supervised learning.

2. Instead of building upon MC-REINFORCE, one can explore the broad *spectrum of reinforcement learning* algorithms in this framework.

3. Further, one can study how well SPOA figures out matrix representation of actions by *hiding* known action variables in RL benchmarks.

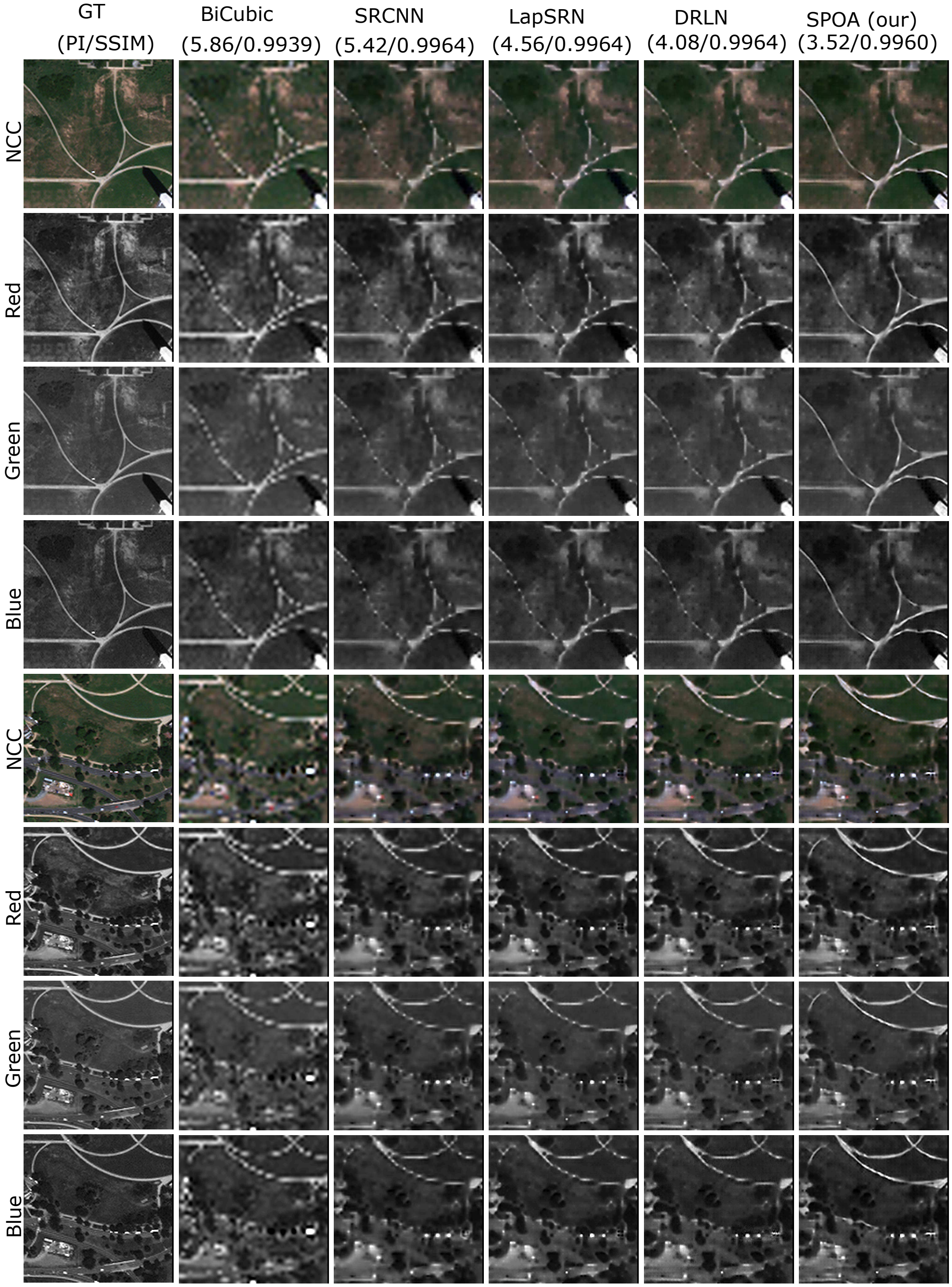

Figure 5. Qualitative analysis on WorldView-2. SPOA outperforms compared methods in perceptual quality, and also generates more natural textures while mitigating unpleasant artifacts, e.g., discontinuity of linear features.

This study demonstrated the plausibility of DRL in solving supervised problems as sequential decision making processes. The efficacy of SPOA in this regard broadens the horizon of DRL, suggesting further investigation in this viable research direction might prove beneficial.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017. 2

[2] Saeed Anwar and Nick Barnes. Densely residual laplacian super-resolution. *arXiv preprint arXiv:1906.12021*, 2019. 2, 5, 6, 7

[3] Saeed Anwar, Salman Khan, and Nick Barnes. A deep journey into super-resolution: A survey. *arXiv preprint arXiv:1904.07523*, 2019. 2

[4] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010. 2

[5] Alexei A Bastidas and Hanlin Tang. Channel attention networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[6] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 2

[7] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 2, 6

[8] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018. 6, 7

[9] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. In *Advances in neural information processing systems*, pages 737–744, 1994. 4

[10] Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2488–2496, 2015. 2

[11] Qingxing Cao, Liang Lin, Yukai Shi, Xiaodan Liang, and Guanbin Li. Attention-aware face hallucination via deep reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 690–698, 2017. 2

[12] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016. 7

[13] Yushi Chen, Xing Zhao, and Xiuping Jia. Spectral–spatial classification of hyperspectral data based on deep belief network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2381–2392, 2015. 2

[14] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 2, 5, 6, 7

[15] Xiaoyu Dong, Zhihong Xi, Xu Sun, and Lianru Gao. Transferred multi-perception attention networks for remote sensing image super-resolution. *Remote Sensing*, 11(23):2857, 2019. 2

[16] Hajar Emami, Majid Moradi Aliabadi, Ming Dong, and Ratna Chinnam. Spa-gan: Spatial attention gan for image-to-image translation. *IEEE Transactions on Multimedia*, 2020. 7

[17] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010. 5

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 4

[19] Jun Gu, Xian Sun, Yue Zhang, Kun Fu, and Lei Wang. Deep residual squeeze and excitation network for remote sensing image super-resolution. *Remote Sensing*, 11(15):1817, 2019. 2

[20] Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard L Lewis, and Xiaoshi Wang. Deep learning for real-time atari game play using offline monte-carlo tree search planning. In *Advances in neural information processing systems*, pages 3338–3346, 2014. 1

[21] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018. 1

[22] 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*, pages 770–778, 2016. 3

[23] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369. IEEE, 2010. 6

[24] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 2

[25] Kui Jiang, Zhongyuan Wang, Peng Yi, Guangcheng Wang, Tao Lu, and Junjun Jiang. Edge-enhanced gan for remote sensing image superresolution. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5799–5812, 2019. 2

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[27] FM Lacar, MM Lewis, and IT Grierson. Use of hyperspectral imagery for mapping grape varieties in the barossa valley, south australia. In *IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217)*, volume 6, pages 2875–2877. IEEE, 2001. 2

[28] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Fast and accurate image super-resolution with

deep laplacian pyramid networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2599–2613, 2018. 2, 7

[29] Charis Lanaras, José Bioucas-Dias, Silvano Galliani, Emmanuel Baltsavias, and Konrad Schindler. Super-resolution of sentinel-2 images: Learning a globally applicable deep neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146:305–319, 2018. 6

[30] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2, 6, 7

[31] Dennis Lee, Haoran Tang, Jeffrey O Zhang, Huazhe Xu, Trevor Darrell, and Pieter Abbeel. Modular architecture for starcraft ii with deep reinforcement learning. In *Fourteenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2018. 1

[32] Sergey Levine and Vladlen Koltun. Variational policy search via trajectory optimization. In *Advances in neural information processing systems*, pages 207–215, 2013. 1

[33] Timothy Paul Lillicrap, Jonathan James Hunt, Alexander Pritzel, Nicolas Manfred Otto Heess, Tom Erez, Yuval Tassa, David Silver, and Daniel Pieter Wierstra. Continuous control with deep reinforcement learning, Jan. 26 2017. US Patent App. 15/217,758. 1

[34] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15:2018, 2018. 5

[35] Chao Ma, Chih-Yuan Yang, Xiaokang Yang, and Ming-Hsuan Yang. Learning a no-reference quality metric for single-image super-resolution. *Computer Vision and Image Understanding*, 158:1–16, 2017. 6

[36] Wen Ma, Zongxu Pan, Feng Yuan, and Bin Lei. Super-resolution of remote sensing images via a dense residual generative adversarial network. *Remote Sensing*, 11(21):2578, 2019. 2

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

[38] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2012. 6

[39] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 1

[40] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. 1

[41] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 6

[42] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897, 2015. 1

[43] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1

[44] Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998. 1, 3, 4

[45] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000. 1

[46] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1865–1873, 2016. 2

[47] Burak Uzkent, Aneesh Rangnekar, and Matthew Hoffman. Aerial vehicle tracking by adaptive fusion of hyperspectral likelihood maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 39–48, 2017. 2

[48] Quan Vuong, Yiming Zhang, and Keith W Ross. Supervised policy update for deep reinforcement learning. 2018. 1

[49] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 2, 7

[50] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Remi Munos, Koray Kavukcuoglu, and Nando de Freitas. Sample efficient actor-critic with experience replay. *arXiv preprint arXiv:1611.01224*, 2016. 1

[51] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *arXiv preprint arXiv:1902.06068*, 2019. 2

[52] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 1, 3

[53] Ke Yu, Chao Dong, Liang Lin, and Chen Change Loy. Crafting a toolchain for image restoration by deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2443–2452, 2018. 2

[54] Ke Yu, Xintao Wang, Chao Dong, Xiaoou Tang, and Chen Change Loy. Path-restore: Learning network path selection for image restoration. *arXiv preprint arXiv:1904.10343*, 2019. 2

[55] Roberta H Yuhas, Alexander FH Goetz, and Joe W Boardman. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm. 1992. 6

[56] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 2

[57] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018. 7