ARQII Publication
http://arqiipubl.com/ams
APPLICATIONS OF
MODELLING AND SIMULATION
eISSN 2600-8084
VOL 8, 2024, 201-212

Building Detection based on Faster RCNN with Distributional Soft Actor-Critic with Three Refinements

A. Amala Arul Reji* and S. Muruganantham

Department of Computer Application, S. T. Hindu College, Nagercoil, Affiliated to Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India

*Corresponding author: amalasweet.reji05@gmail.com

Submitted 13 January 2024, Revised 02 May 2024, Accepted 13 May 2024, Available online 28 June 2024.

Abstract: This research presents a comprehensive framework for building detection in high-resolution images, integrating advanced techniques from computer vision and reinforcement learning. The methodology employs the Faster Region-based Convolutional Neural Network (RCNN) architecture for efficient feature extraction and region proposal generation, enhancing the accuracy of building detection. A novel Fine Region Proposal Network (FRPN) adapts region proposals based on image characteristics, dynamically adjusting candidate regions for improved efficiency. The study introduces three refinements to the Distributional Soft Actor Critic (DSAC-T) algorithm, addressing stability and sensitivity concerns. These enhancements involve fine-tuning the critic gradient, incorporating twin value distribution learning, and introducing a variance-based mechanism for return clipping the target. Rigorous assessments on demanding datasets, such as the Massachusetts and WHU building dataset, provide compelling evidence of the efficacy of the proposed framework. The proposed approach demonstrates superior performance in building detection, achieving an average precision of 69.48% and an average recall of 84.29% on the Massachusetts dataset, and an average precision of 65.82% and an average recall of 81.52% on the WHU dataset. Thus, the research contributes to the field by providing a robust solution for building detection, leveraging state-of-the-art techniques for improved performance in diverse urban and suburban environments.

Keywords: Deep learning; Distributional soft actor critic; Fine region proposal network; Recurrent convolutional neural networks; Variance-based mechanism.

1. INTRODUCTION

Urbanization and rural development have dramatically reshaped landscapes, with buildings emerging as dominant features in urban areas. Recognizing these landmarks has become increasingly important across various applications due to technological advancements that provide high-resolution spatial imagery from satellites and aerial sources [1]. These intricate images offer profound technical insights and textural details of land covers, enhancing the capacity to discern urban elements and extract distinct building attributes. To address this complexity, researchers have introduced diverse strategies, broadly categorized as physical rule-based techniques, image segmentation methodologies, and, more recently, machine and sophisticated deep learning methodologies [2].

Models for building detection in high-resolution remote sensing images are commonly categorized into two types: those relying on manually designed features and those employing feature learning methodologies [3]. The former involves models based on template matching, SVMs, or classifiers employing spectral or textural building traits, while the latter utilizes deep learning algorithms. Convolutional Neural Networks (CNNs) have played a pivotal role in advancing object detection techniques [4]. Erhan *et al*. introduced the Region-based Convolutional Neural Network (RCNN) to streamline this process by sequentially focusing on individual regions, thereby reducing computational overhead [5]. Building upon this, the Faster RCNN framework leverages a dedicated network to predict region proposals, contributing to improved speed and accuracy by eliminating the need for selective search algorithms [6].

In [7], a region-based CNN with deep learning for obstacle detection was employed, integrating Double Deep Q-Network (DDQN) for dynamic decision-making based on identified obstacles [8]. Deep learning has become a central and influential technique in the realm of remote sensing, finding extensive applications across various remote sensing tasks, particularly in image processing. Recent research has underscored the substantial potential and diverse applications of deep learning within the domain of remote sensing, underscoring its significance and ongoing advancements in this dynamic field. The detection of buildings and objects in remotely sensed images has attracted significant attention, leading to the development of various strategies. A novel object detection strategy utilizing Double Deep Q-Networks (Double DQN), demonstrating superior precision and recall compared to traditional methods [9]. However, limitations such as computational resource requirements

and scalability remain unaddressed. Zhou proposed a sequential approach named "rolis," combining Deep Q-network (DQN) and an object identification neural network (INN) for object detection and segmentation in high-resolution images [10]. Despite its potential advantages, including adaptability to data drift, extensive training requirements pose challenges, particularly with limited computational resources. In [11], they focused on training models to localize objects using deep reinforcement learning, treating object detection as an iterative process. Two action settings were explored within the Markov Decision Process (MDP), aiming for efficient object localization. However, challenges such as shape constraints and recovery from initial poor region choices were noted.

Furthermore, [12] introduced a reinforcement learning (RL) model utilizing Twin Delayed Deep Deterministic Policy Gradient (TD3) for single object tracing in computer vision, demonstrating enhanced convergence and reduced estimation bias. Also, deep RL empowered by deep neural networks has exhibited proficiency across various domains, including gaming and object detection [13]. However, a persistent issue in RL methods is the tendency to overestimate state-action values (Q-values), resulting in suboptimal policies [14]. This overestimation, initially identified in Q-learning [15], extends to most value-based RL algorithms [16].

The challenge of estimating Q-values accurately in RL is pervasive due to inherent estimation biases, variances, and unavoidable function approximation errors. Approaches like Double Q-learning (DQL) and its deep variant, Double DQN, alleviate overestimation but are limited to discrete action spaces [17]. Extensions to continuous control faced challenges due to similarities between online and target Q-assessment, leading to persistent overestimations [18]. Despite attempts to address overestimation, including Clipped DQL introduced by [18] and incorporated into Soft Actor-Critic (SAC) and Twin Delayed Deep Deterministic policy gradient (TD3) [19], the overestimation in RL remains a persistent challenge. While various approaches attempt to mitigate it, they often come with trade-offs, including computational complexity and biases in estimation [20].

In addressing these challenges, the proposed approach introduces several novel contributions and innovations. Firstly, by integrating the Faster RCNN framework with a modified ResNet-50 network and a Fine Region Proposal Network (FRPN), the model efficiently extracts features and generates region proposals tailored specifically for building detection tasks. This enhancement overcomes the limitations of traditional RPNs by dynamically adapting region proposals based on image characteristics, thereby enhancing overall efficiency and effectiveness.

Secondly, the research introduces three refinements to enhance the stability and reduce the sensitivity of the Distributional Soft Actor-Critic with Three Refinements (DSAC-T) algorithm. These refinements include adjusting the critic gradient, implementing twin value distribution learning, and employing variance-based target return clipping. These enhancements address issues related to randomness, improve critic updates, and automate the determination of clipping boundaries, reducing the need for manual hyperparameter tuning and enhancing the model's stability.

Furthermore, the proposed framework undergoes rigorous evaluation on well-known and demanding public datasets such as the Massachusetts and WHU building dataset. This evaluation not only showcases the efficacy of the methodology in building detection tasks but also provides valuable insights into its performance and generalizability across diverse urban and suburban environments.

The contributions and innovations of this research can be outlined as follows:

(a) Integration of Faster RCNN for Building Detection: The research contributes by integrating the Faster RCNN framework, known for its efficiency in object detection, into the proposed methodology for building detection in high-resolution images. This integration enhances the system's ability to autonomously learn region proposals, thereby improving efficiency and accuracy in real-time scenarios.

(b) Development of Fine Region Proposal Network (FRPN): The introduction of the Fine Region Proposal Network (FRPN) represents a novel contribution to the field of object detection. FRPN dynamically adjusts region proposals based on image characteristics, optimizing the efficiency and effectiveness of building detection. This innovative approach addresses the limitations of traditional region proposal networks and improves overall system performance.

(c) Introduction of DSAC-T for Reinforcement Learning: The research introduces the Distributional Soft Actor-Critic with Three Refinements (DSAC-T) algorithm, which enhances the stability and accuracy of reinforcement learning in building detection tasks. By addressing issues such as overestimation biases and sensitivity concerns, DSAC-T improves decision-making and ensures reliable performance in challenging environments.

(d) Refinements to Faster RCNN Architecture: The proposed methodology includes novel refinements to the Faster RCNN architecture, such as bounding box regression layers and ROI Max Pooling layers. These enhancements optimize feature extraction and region proposal generation, further improving the efficiency and accuracy of building detection.

(e) Evaluation on Challenging Datasets: The research contributes by rigorously evaluating the proposed framework on well-known public datasets, including the Massachusetts and WHU building datasets. This evaluation provides valuable insights into the system's performance and its ability to generalize across diverse urban and suburban environments, demonstrating its efficacy and potential for real-world applications.

(f) Potential Applications and Impact: Overall, the research offers a novel and effective solution for building detection in challenging environments, with potential applications in urban planning, environmental monitoring, disaster response, and other domains requiring accurate and efficient object detection in high-resolution imagery. The contributions of this research advance the state-of-the-art in remote sensing and deep learning methodologies, paving the way for further advancements in the field.

The paper's organization is as follows: Section 2 delineates the fundamentals of RL and introduces the DSAC framework. Section 3 provides a comprehensive description of the proposed Faster RCNN with DSAC-T algorithm. Section 4 illustrates the experimental results, showcasing the effectiveness of DSAC-T. Lastly, Section 5 offers concluding remarks for the paper.

2. PRELIMINARIES

In reinforcement learning, the environment is often described by a Markov decision process, characterized with a tuple (c, n, p, F). Assuming constant action and space, the stochastic function of reward $F(e_t|c_t, n_t)$: $C \times N \rightarrow \mathrm{P}(e_t)$ maps the state-action (c_t, n_t) for distributing potential rewards, while the unidentified transition probability is $p(c_{t+1}|c_y, n_t)$: $c \times n \rightarrow P(c_{t+1})$. This stochastic reward operation maps a specific state-action pair to a range of potential rewards, alongside the associated transition probability determines the probability distribution over the succeeding state c_{t+1} based on the current state-action pair.

The agent, located in a state $c_t \in c$, selects an action $n_t \in A$ and, in response, receives the succeeding state $c_{t+1} \in S$ and a scalar reward $e_t \sim F(c_t, n_t)$ from the environment at each time step t, until it reaches a terminal state. At each time step t, the agent, situated in a state $c_t \in c$, chooses an action $n_t \in A$ and, consequently, receives the subsequent state $c_{t+1} \in S$ and a scalar reward $e_t \sim F(c_t, n_t)$ from the environment. This process continues until the agent reaches a terminal state. This environment is influenced by a stochastic policy denoted as $\pi(n_t|c_t)$: $c \rightarrow P(n_t)$, which maps states to probability distributions over actions, thereby defining the agent's behavior. The policy π induces distributions over state-action pairs $\rho_\pi(c, n)$ and states $\rho_\pi(c)$.

2.1 Maximum Entropy RL

In conventional RL, the goal formulates a policy which increases the expected cumulative future return, denoted as $\mathrm{E}_{(c,n,e,c') \sim \mathcal{B}, n \sim \pi_{\phi'}, Z(c',n') \sim \mathrm{Z}_{\theta'}(.|c,n)}$, where $\gamma \in [0,1]$ represents the discount factor. This proposed approach extends the traditional RL objective by incorporating a more comprehensive entropy-augmented criterion [20]. This augmented objective introduces a policy entropy term e, enhancing the reward formulation.

$$\mathcal{J}_\pi = \underset{(c_{i\geq t}, n_{i\geq t}) \sim \rho_\pi, e_{i\geq t} \sim \mathcal{R}(.|c_i, n_i)}{\mathrm{E}} \left[\sum_{i=t}^{\infty} \gamma^{i-t} \left[e_i + \alpha \mathrm{e}(\pi(.|c_i)) \right] \right] \tag{1}$$

Here,

$$\mathrm{e}(\pi(.|c_i)) = -\int_{n \in N} \pi(n|c) log\pi(n|c) dn = \underset{n \sim \pi(.|c_i)}{\mathrm{E}} [-log\pi(n|c)] \tag{2}$$

This objective aims to enhance the exploration effectiveness of the policy by exploiting not only the estimated future return but also the policy entropy. In conventional RL, the goal revolves around formulating a policy aimed at maximizing the expected cumulative future return $\mathcal{J}_\pi$, denoted as the expectation over state-action trajectories sampled from the environment distribution B, governed by the policy π and parameterized by $'\phi'$, Z, and θ', where γ represents the discount factor. This augmented objective incorporates a policy entropy term E, enhancing the reward formulation. The objective function $\mathcal{J}_\pi$ is formulated as the expectation over state-action trajectories weighted by the policy entropy and the accumulated reward over time steps, where the entropy term aims to promote exploration effectiveness by introducing randomness into the policy. The attribute α, stated to as temperature parameter, controls the significance of the entropy term relative to the reward. As α tends toward 0, the approach of maximum entropy RL progressively aligns with traditional RL methods.

The $\mathbb{R}_t$ as the accumulated return from time step t, augmented with entropy given by $\mathbb{R}_t = \sum_{i=t}^{\infty} \gamma^{i-t} [e_i - \alpha log\pi(n_i|c_i)]$. The soft Q-value π is formulated as follows:

$$\mathbb{Q}^\pi(n_t, c_t) = \underset{e \sim R(.|n_t, c_t)}{\mathrm{E}} [e] + \gamma \underset{(c_{i>t}, n_{i>t}) \sim \rho_\pi, e_{i>t} \sim \mathcal{R}(.|c_i, n_i)}{\mathrm{E}} [\mathbb{R}_{t+1}] \tag{3}$$

In evaluation phase of soft policy, the soft Q-value is acquired by iteratively applying the soft Bellman operator $\mathcal{B}^\pi$ under policy π. This operator is defined by:

$$\mathcal{B}^\pi \mathbb{Q}^\pi(c, n) = \mathrm{E}_{e \sim \mathcal{R}(.|c,n)} [e] + \gamma \mathrm{E}_{c' \sim p, n' \sim \pi} [\mathbb{Q}^\pi(c', n') - \alpha log\pi(n'|c')] \tag{4}$$

The soft policy improvement denoted as π_{latest}, which outperforms the existing policy π_{past} , ensuring $\mathrm{J}_{\pi_{latest}} \geq \mathrm{J}_{\pi_{past}}$. This update entails exploiting the entropy-augmented outlined in Equation (1) with respect to soft Q-value.

$$\pi_{latest} = \underset{\pi}{\arg max} J_\pi = \underset{\pi}{\arg max} \underset{c \sim \rho_\pi, n \sim \pi}{\mathrm{E}} [\mathbb{Q}^{\pi_{past}}(c, n) - \propto log\pi(n|c)] \tag{5}$$

In the Equations, $\mathbb{R}_t$ represents the accumulated return from time step *t*, augmented with entropy, where e_i denotes the entropy term at time step *i*, and α is the temperature parameter controlling the significance of the entropy term. The soft Q-value $\mathbb{Q}^\pi(n_t, c_t)$ is formulated as the expectation over rewards sampled from the distribution of rewards *R*, given the current state-action pair (n_t, c_t), augmented with entropy. This includes the immediate reward *e* and the discounted future rewards $\mathbb{R}_{t+1}$ obtained by sampling subsequent state-action pairs $(c_{i>t}, n_{i>t})$ under the policy π, parameterized by $\rho_\pi, e_{i>t}$. The soft Bellman operator $\mathcal{B}^\pi$ defines the iterative application of the soft Q-value under policy π, incorporating the expected immediate

reward and the expected future rewards, adjusted by the entropy term $\alpha log\pi(n'|c')$. The soft policy improvement, denoted as π_{latest}, aims to surpass the performance of the previous policy π_{past}, ensuring that the objective function $J_{\pi_{latest}}$ exceeds or equals the objective function $J_{\pi_{past}}$.

2.2 Distributional Soft Actor Critic

For addressing continuous state and action spaces, earlier research introduced a conventional model of DSAC approach stated as DSAC-v1. This approach employs neural networks to serve as approximators for both value function and policy function [21]. The DSAC employs parameterized distributions represented by $Z_\theta(.|c,n)$ for the value function and $\pi_\phi(.|c)$ for the stochastic policy, where θ and ϕ denote the respective parameters.

The efficiency of DSAC's learning is upgraded through parallel or distributed learning, similar to off-policy RL algorithms. The utilization of an asynchronous buffer actor-learner architecture helps achieve a high learning throughput. By distributing buffers, actors, and learners across multiple workers, improvements in sampling, storage capacity, exploration, and information utilization are achieved, fostering asynchronous communication among these components. Coordinating asynchronously through shared memory, actors and learners exchange parameters, transmitting the skills generated by each agent to a buffer separately at each time step. The sampled experiences are continuously stored and dispatched to the random learner by the buffer. Utilizing local functions, the learner estimates update gradients based on the sample data to iteratively update both the policy function and the shared value.

Policy Evaluation: The soft return distribution loss function is computed as,

$$\mathbb{Z}_{latest} = \arg\min_{\mathbb{Z}} \mathop{\mathrm{E}}_{(c,n)\sim\rho_\pi}\left[d\left(\mathcal{L}_{\mathcal{D}}^{\pi}\mathbb{Z}_{past}(.|c,n), \mathbb{Z}(.|c,n)\right)\right] \tag{6}$$

This loss function aims to minimize the discrepancy by training the return distribution of state-action value. This is accomplished through the use of the Kullback-Leibler (KL)-divergence metric defined in Equation (7).

$$\mathcal{J}_{\mathrm{Z}}(\theta) = \mathop{\mathrm{E}}_{(c,n)\sim\mathcal{B}}\left[\mathcal{D}_{\mathcal{KL}}\left(\mathcal{L}_{\mathcal{D}}^{\pi_{\phi'}}\mathbb{Z}_{\theta'}(.|c,n), \mathbb{Z}_\theta(.|c,n)\right)\right] \tag{7}$$

The replay buffer containing previously sampled experiences is symbolized as $\mathcal{B}$. To compute the process and stabilize the learning procedure, the target return distribution parameters θ' and the policy function ϕ', as defined in Equation (8), are utilized.

$$\mathcal{J}_{\mathbb{Z}}(\theta) = -\mathop{\mathrm{E}}_{(c,n,e,c')\sim\mathcal{B}, n'\sim\pi_{\phi'}, Z(c',n')\sim\mathbb{Z}_{\theta'}(.|c,n)}\left[logP\left(\mathcal{L}_{\mathcal{D}}^{\pi_{\phi'}}\mathbb{Z}_{\theta'}(.|c,n), \mathbb{Z}_\theta(.|c,n)\right)\right] \tag{8}$$

The parameter θ can be optimized using the gradient, as indicated in Equation (9).

$$\nabla_\theta\mathcal{J}_{\mathbb{Z}}(\theta) = -\mathop{\mathrm{E}}_{(c,n,e,c')\sim\mathcal{B}, n'\sim\pi_{\phi'}, Z(c',n')\sim\mathbb{Z}_{\theta'}}\left[\nabla_\theta log\mathrm{P}\left(\mathcal{L}_{\mathcal{D}}^{\pi_{\phi'}}Z(c,n)|\mathbb{Z}_\theta(.|c,n)\right)\right] \tag{9}$$

In Equation (6), $\mathbb{Z}_{latest}$ is determined as the argument that minimizes the expected discrepancy between the distribution of the previous return $\mathbb{Z}_{past}$ and the distribution of the current return *Z*. This discrepancy is computed using the loss function $d\left(\mathcal{L}_{\mathcal{D}}^{\pi}\mathbb{Z}_{past}(.|c,n), \mathbb{Z}(.|c,n)\right)$, which aims to train the return distribution of state-action values. The Kullback-Leibler (KL)-divergence metric $\mathcal{D}_{\mathcal{KL}}$, defined in Equation (7), is utilized to quantify the discrepancy and optimize the loss function $\mathcal{J}_{\mathbb{Z}}(\theta)$. The replay buffer *B* contains previously sampled experiences, and the optimization process involves stabilizing the learning procedure using target return distribution parameters θ' and the policy function ϕ', as specified in Equation (8). Here, $\mathcal{J}_{\mathbb{Z}}(\theta)$ represents the expected KL-divergence between the target return distribution $\mathcal{L}_{\mathcal{D}}^{\pi_{\phi'}}Z(c,n)$ and the current return distribution $\mathbb{Z}_\theta(.|c,n)$. Finally, Equation (9) outlines the optimization of parameter θ using the gradient of the loss function $\mathcal{J}_{\mathbb{Z}}(\theta)$.

Consider the Gaussian model represented as $\mathbb{Z}_\theta$, as expressed in Equation (10).

$$\mathbb{Z}_\theta(.|c,n) = \mathcal{N}(Q_\theta(c,n), \sigma_\theta(c,n)^2) \tag{10}$$

The outputs of the value network are represented as $Q_\theta(c,n)$ and $\sigma_\theta(c,n)^{\square}$, and the gradient of the Gaussian variant is subsequently updated.

$$\nabla_\theta\mathcal{J}_{\mathbb{Z}}(\theta) = \mathop{\mathrm{E}}_{(c,n,e,c')\sim\mathcal{B}, n'\sim\pi_{\phi'}Z(c',n')\sim\mathbb{Z}_{\theta'}}\left[\nabla_\theta\frac{\left(\mathcal{L}_{\mathcal{D}}^{\pi_{\phi'}}Z(c,n) - \left(\mathbb{Q}_\theta(c,n)\right)^2\right.}{2\sigma_\theta(c,n)^2} + \frac{\nabla_\theta\sigma_\theta(c,n)}{\sigma_\theta(c,n)}\right] \tag{11}$$

To comprehend the structure of $\nabla_\theta \mathcal{J}_\mathbb{Z}(\theta)$, the $log\mathrm{P}\left(\mathcal{T}_\mathcal{D}^{\pi_{\phi'}}\mathbb{Z}_{\theta'}(.|c,n), \mathbb{Z}_\theta(.|c,n)\right)$ is denoted as $\Psi_\mathbb{Z}(\theta)$, hence Equation (11) is expressed as Equation (12).

$$\nabla_\theta \mathcal{J}_\mathbb{Z}(\theta) = \underset{(c,n,e,c')\sim B, n'\sim\pi_{\phi'} Z(c',n')\sim\mathbb{Z}_{\theta'}}{E}\left[-\frac{\partial\Psi_\mathbb{Z}(\theta)}{\partial\mathbb{Q}_\theta(c,n)}\nabla_\theta\mathbb{Q}_\theta(c,n) - \frac{\partial\Psi_\mathbb{Z}(\theta)}{\partial\sigma_\theta(c,n)}\nabla_\theta\sigma_\theta(c,n)\right] \tag{12}$$

The dimension of $\mathbb{Q}_\theta(c,n)$ is adjusted by diminishing $\frac{\partial\Psi_\mathbb{Z}(\theta)}{\partial\mathbb{Q}_\theta(c,n)}$, which is directly lessened by increasing $\sigma_\theta(c,n)$. Consequently, the overestimation of Q-value is mitigated. Furthermore, the gradients $\nabla_\theta\mathcal{J}_\mathbb{Z}(\theta)$ tend to explode as $\sigma_\theta(c,n)\to 0$ or vanish as $\sigma_\theta(c,n)\to\infty$, and this can be addressed by maintaining the range of $\sigma_\theta(c,n)$ within reasonable bounds and initially minimizing its value as per Equation (13).

$$\overline{\mathcal{L}_z} = clip\left(\mathcal{L}_\mathcal{D}^{\pi_{\phi'}}Z(c,n), \mathbb{Q}_\theta(c,n) - b, \mathbb{Q}_\theta(c,n) + b\right) \tag{13}$$

Here, b represents the clipping boundary and $\mathcal{L}_z = \mathcal{L}_\mathcal{D}^{\pi_{\phi'}}Z(c,n)$. After clipping, the critic gradient (12) becomes,

$$\nabla_\theta\mathcal{J}_\mathbb{Z}(\theta) \approx \mathrm{E}\left[-\frac{(\mathcal{L}_z - \mathbb{Q}_\theta(c,n))}{\sigma_\theta(c,n)^2}\nabla_\theta\mathbb{Q}_\theta(c,n) - \frac{(\overline{\mathcal{L}_z} - \mathbb{Q}_\theta(c,n))^2 - \sigma_\theta(c,n)^2}{\sigma_\theta(c,n)^3}\nabla_\theta\sigma_\theta(c,n)\right] \tag{14}$$

Policy Enhancement: The enhancement of the actor involves the maximization of a parameterized version of Equation (15 & 16):

$$J_\pi(\phi) = \underset{c\sim\mathcal{B}, n\sim\pi_\phi}{\mathrm{E}}\left[\mathbb{Q}_\theta(c,n) - \alpha\, log\left(\pi_\phi(c|n)\right)\right] \tag{15}$$

$$J_\pi(\phi) = \underset{c\sim\mathcal{B}, n\sim\pi_\phi}{\mathrm{E}}\left[\underset{Z(c,n)\sim\mathbb{Z}_\theta(.|c,n)}{\mathrm{E}}[Z(c,n)] - \alpha\log\left(\pi_\phi(c|n)\right)\right] \tag{16}$$

When $'n'$ is unbounded, the logarithm of the action distribution, $\log\left(\pi_\phi(c|n)\right)$, is assessed through the action distribution parameters derived from the Gaussian distribution, which encompass the mean and variance. Additionally, leveraging re-parameterization and log derivatives serves to maximize $J_\pi(\phi)$ while simultaneously reducing the variance in gradient estimation.

The temperature parameter, α, plays a pivotal role in maintaining a balance between the exploitation and exploration. The temperature is adjusted according to the following equation,

$$J(\alpha) = \mathrm{E}_{(c,n)\sim\mathcal{B}}[\alpha(-log\pi_\phi(c|n) - \bar{\mathrm{e}})] \tag{17}$$

Here, the expected entropy is represented as $\bar{\mathrm{e}}$, and a lower frequency of policy updates results in higher-quality policy updates.

3. PROPOSED METHODOLOGY

3.1 Building Detection by Faster RCNN

In detecting building using Deep Neural Networks (DNN), an effective model should adeptly distinguish intricate factors related to building representation in images. To achieve near real-time efficiency with satisfactory accuracy, the suggested approach employed Faster RCNN network [22]. The images are processed through a modified ResNet-50 network for feature extraction.

Specifically, the efficiency of the Faster RCNN is enhanced while training by bounding box regression layer. This consists of a fully-connected layer with RCNN box regression, generating 4 box offsets for every class. Another significant addition is the ROI Max Pooling layer, situated after the modified ResNet-50, which receives feature maps from Faster RCNN. This layer selects corresponding sections from the feature map for each input Region of Interest and scales it to a fixed size of 14×14. The use of fixed-sized feature maps enhances efficiency by promoting reusability in other object proposals.

The last layer is FRPN, assumes a vital role in generating region proposals for Faster RCNN. Its primary task is to extract region boxes that are highly likely to contain building-related features. These region boxes, referred to as anchors, are re-sized selecting optimal regions for CNN for extracting features. Ultimately, image regions are characterized into 2 classes: non-buildings or buildings. Consequently, the system's end output comprises a sequence of images where only the identified buildings are outlined by bounding boxes. Figure 1 provides an overview of the entire process involved in the proposed building detection method.

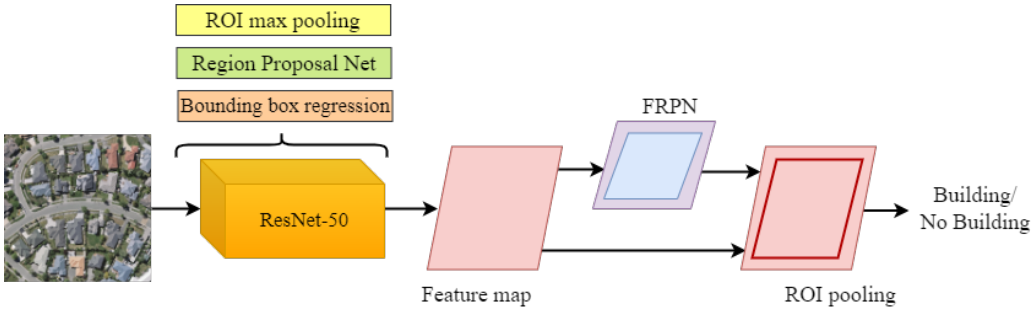

Figure 1. Proposed faster RCNN framework.

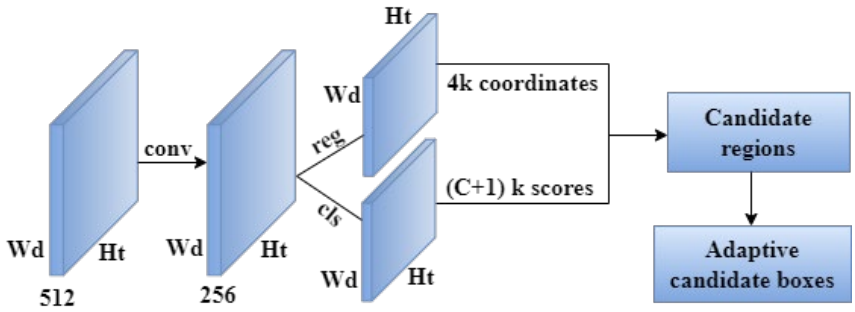

Figure 2. Framework of FRPN.

3.1.1 Fine Region Proposal Network (FRPN)

The advancement in Faster RCNN is propelled by the efficiency of the RPN, which extracts high-quality candidate regions at minimal computational cost. Traditionally, the standard RPN receives an image as input then it generates a fixed set of rectangular object boxes having varying scales and aspect ratios. These boxes undergo binary classification scoring to extract negative ones, and high-ranked boxes are selected as candidate regions. However, using a fixed set of candidate regions for all images might not be practical due to varying object counts across images. To tackle this challenge, the proposed model introduces FRPN for facilitating adaptive region proposals for diverse images. Like RPN, FRPN utilizes a VGG-16 backbone grid for extracting high-level semantic features. FRPN incorporates an n×n convolutional layer (con-rpn) alongside 2 sibling 1×1 output layers—dedicated to fine box classification and box regression. The complete design of FRPN is depicted in Figure 2.

Before commencing the training process, it was essential to determine the positive and negative anchor training samples. The proposed approach generated 12 anchors for each position of con-rpn feature maps. These anchors encompassed 4 scales (256×256, 512×512, 128×128, and 96×96 pixels) and 3 aspect ratios (1:2, 1:1, and 2:1). Positive labels were assigned to anchors with an IoU (Intersection over Union) greater than 0.5, corresponding to their category.

Accurately localizing objects poses a significant challenge in object detection, especially in addressing the issue of imbalanced region proposals for each object, restricting optimal object localization. The proposed approach introduces the FRPN, allowing us to balance region proposal numbers across images. Specifically, our fine classification network, an extension of the RPN, handles multiple object categories, determining the precise class of an anchor along with its confidence. The softmax loss computed for all anchor samples as,

$$\mathcal{L}oss_{cls}(p, M) = \frac{1}{C}\left(\sum_{x \in pos}^{C} -M_x^* \log\left(\hat{p}_x^{s^*}\right) + \sum_{x \in Neg} -\log\left(\hat{p}_x^0\right)\right) \tag{18}$$

Here, $\hat{p}_x^s = \frac{\exp(p_x^s)}{\sum_s \exp(p_x^s)}$.

p_x represents the predicted softmax outputs, and p_x^s denotes the corresponding predicted scores for the s_{th} category. $p_x^{s^*}$ represents the predicted scores for the x_{th} sample's ground truth s^* category, with class 0 representing the background. M_x^* serves as an indicator, with M_x^*=1 if the x_{th} candidate region matches a ground truth box and M_x^*=0 otherwise. C denotes matched candidate regions. The fine network incurs minimal cost associated to traditional RPN.

The regression network's architecture is adapted from the RPN in Faster RCNN. Therefore, the training loss for a single image is calculated as the average value across all positive anchors,

$$\mathcal{L}oss_{reg}(b^u, b^v) = \frac{1}{C} \sum_{x \in (a,b,wd,ht)} SmoothL_1\left(b_x^u - b_x^v\right) \tag{19}$$

Here, u denote positive sample index, b_x^v represent offset among x_{th} anchor and ground truth boxes, b_x^u predicted offset. The variables $a, b, wd, and\ ht$ represent center a-coordinates, center b-coordinates, height of boxes, and width of boxes. The introduced approach incorporates a weight term β to maintain a balance among regression network and classification network in FRPN:

$$\mathcal{L}oss_{FRPN}(p, M, b^u, b^v) = \mathcal{L}oss_{cls}(p, M) + \frac{1}{\beta}\mathcal{L}oss_{reg}(b^u, b^v) \tag{20}$$

Here, $\mathcal{L}oss_{FRPN}(p, M, b^u, b^v)$ is the loss used to train the FRPN. For experimentation $\beta = 0.5$, indicating a bias toward better box location.

3.2 DSAC with Three Refinements (DSAC-T)

The incorporation of DSAC-T in this research is driven by its ability to enhance the stability of reinforcement learning, mitigate sensitivity to reward scaling, and address learning instabilities. These refinements contribute to a more robust and effective training framework, optimizing the performance of building detection in high-resolution images alongside Faster RCNN and FRPN. The DSAC algorithm, known for its integration of a distributional value function in both critic and actor updates, has demonstrated practical utility. However, its application in building detection tasks has occasionally resulted in learning instabilities. Additionally, this approach demands specific fine-tuning of hyperparameters, posing challenges for swift task setups. To address these limitations, the proposed methodology introduces three pivotal enhancements to the standard DSAC. These refinements are designed to enhance learning stability and reduce sensitivity to reward scaling. They involve adjusting the critic gradient, incorporating twin value distribution learning, and implementing variance-based target return clipping.

3.2.1 Critic gradient adjusting

DSAC-v1 aims to minimize the variability in the gradient associated with variance by restraining the random target return through clipping. However, it fails to mitigate the heightened randomness in the gradient linked to the mean, induced by the unpredictable target return. To address this challenge, the fundamental approach involves modifying mean-related gradient through substituting erratic target return by more stable surrogate function.

The focus primarily centers on target value employed for non-distributional Q-network:

$$\mathcal{L}_q = e + \gamma(\mathbb{Q}_{\bar{\theta}}(c', n') - \theta log\pi_{\bar{\phi}}(n'|c')) \tag{21}$$

Comparing the target Q-value with the target return introduces additional randomness attributed to the value distribution $\mathbb{Z}$. This heightened randomness, in Equation (17), may lead to instability while learning value distribution.
The equivalence among $\mathcal{L}_z$and $\mathcal{L}_q$is represented as,

$$\mathrm{E}_{\mathrm{Z}(c',n')\sim\mathbb{Z}_{\bar{\theta}}(c',n')}\left[\mathcal{L}_z|_{n'\sim\pi_{\bar{\phi}},\mathrm{Z}(c',n')\sim\mathbb{Z}_{\bar{\theta}}(.|c',n')}\right] = e + \left[\gamma\left(\mathbb{Q}_{\bar{\theta}}(c', n') - \propto log\pi_{\bar{\phi}}(n'|c')\right)\right]|_{n'\sim\pi_{\bar{\phi}}} = \mathcal{L}_q \tag{22}$$

Utilizing this equivalence, replace the initial instance of $\mathcal{L}_z$ in (14) by $\mathcal{L}_q$. Then, the equation becomes,

$$\nabla_\theta \mathcal{J}_{\mathbb{Z}}(\theta) \approx \mathrm{E}\left[-\frac{(\mathcal{L}_q - \mathbb{Q}_\theta(c,n))}{\sigma_\theta(c,n)^2}\nabla_\theta\mathbb{Q}_\theta(c,n) - \frac{(\overline{\mathcal{L}_z} - \mathbb{Q}_\theta(c,n))^2 - \sigma_\theta(c,n)^2}{\sigma_\theta(c,n)^3}\nabla_\theta\sigma_\theta(c,n)\right] \tag{23}$$

The adjusted critic gradient incorporating $\mathcal{L}_q$ instead of $\mathcal{L}_z$ mitigates the heightened randomness linked to mean-related gradients. Specifically, $\mathcal{L}_q - \mathbb{Q}_\theta(c,n)$signifies the Temporal Difference (TD) error accurately. Hence, the first component of Equation (15) is akin to the update gradient seen in conventional RL techniques yet adjusted by a scaling factor of $\sigma_\theta(c,n)^2$. This reformulated Q-value learning approach in Equation (15) bears resemblance to established RL methods such as SAC [19], ensuring a comparable level of learning stability.

3.2.2 Distributional learning by Twin Value

The subsequent enhancement involves a distributional adaptation of clipped double Q-learning [18], referred to as twin value distribution learning. This method involves the parameterization of two distinct value distributions denoted by θ_1 and θ_2, both trained separately. To generate actor and critic gradients, the distribution with the lower mean value is chosen. We identify the index of the selected value distribution as follows, facilitating subsequent critic updates:

$$\bar{x} \coloneqq \underset{x=1,2}{\arg min}\,\mathbb{Q}_{\bar{\theta}_x}(c', n')|_{n'\sim\pi_{\bar{\phi}}(.|c')} \tag{24}$$

Consequently, $\bar{\theta}_x$ is employed to assess target return as $\mathcal{L}_z$and Q-value as delineated in Equation (16). The formulations of target assessments expressed as,

$$\mathcal{L}_q^{min} = e + \gamma(\mathbb{Q}_{\bar{\theta}_x}(c', n') - \propto log\pi_{\bar{\phi}}(n'|c')) \tag{25}$$

Substitute Equation (20) in (22),

$$\nabla_\theta J_{\mathbb{Z}}(\theta_x) \approx E\left[-\frac{(\mathcal{L}_q^{min} - \mathbb{Q}_{\theta_x}(c,n))}{\sigma_\theta(c,n)^2}\nabla_{\theta_x}\mathbb{Q}_{\theta_x}(c,n) - \frac{(\overline{\mathcal{L}_z^{min}} - \mathbb{Q}_{\theta_x}(c,n))^2 - \sigma_{\theta_x}(c,n)^2}{\sigma_{\theta_x}(c,n)^3}\nabla_{\theta_x}\sigma_{\theta_x}(c,n)\right] \quad (26)$$

Similarly, the actor's objective undergoes a modification involving twin value distributions.

$$J_\pi(\phi) = \underset{c\sim\mathcal{B}, n\sim\pi_\phi}{\mathrm{E}}\left[\min_{x=1,2}\mathbb{Q}_{\theta_x}(c,n) - \alpha\, log\left(\pi_\phi(c|n)\right)\right] \quad (27)$$

3.2.3 Variance-based target return clipping

In accordance with Equation (13), DSAC-v1 employs a fixed threshold for clipping to avoid exploding gradients. The choice of this clipping threshold is crucial, as a smaller value may affect the accuracy of variance learning, while a larger value could result in a significant gradient norm. Inappropriately chosen clipping thresholds can substantially impede learning performance. Particularly, the mean and variance of the value distribution are directly tied to reward magnitudes. This suggests that distinct optimal clipping threshold designs are generally associated with different reward scales. Additionally, reward magnitudes may vary not only across different tasks but also evolve over time as the policy improves during training. To mitigate this sensitivity to reward scaling, the enhanced version automates the determination of the clipping threshold by,

$$b = \xi \underset{(c,n)\sim\mathcal{B}}{\mathrm{E}}\left[\sigma_{\theta_x}(c,n)\right] \quad (28)$$

In this configuration, ξ serves as constant attribute governing clipping range. In general, a recommended choice for ξ is 3, aligning with the three-sigma rule. Despite its simplicity, this refinement proves to be remarkably effective, alleviating the need for intricate fine-tuning of task-specific hyperparameters. Pseudocode 1 provides the proposed DSAC-T learning approach.

Pseudocode 1: Proposed DSAC-T learning approach

Algorithm 1: DSAC-T
Input: θ_1, θ_2, ϕ, α, $\mathcal{L}$, β_π, β_α, β_z
// Set target networks
$\bar{\theta}_1 \leftarrow \theta_1$, $\bar{\theta}_2 \leftarrow \theta_2$, $\bar{\phi} \leftarrow \phi$,
while (every iteration)
  while (every sampling process)
    Compute action $c\sim\pi(c|n)$
    Obtain reward e and current state n'
    Save samples (c, n, e, c') in buffer $\mathcal{B}$
  end while
  while (every modified process)
    Data sample from $\mathcal{B}$
    Modify critic by $\theta \leftarrow \theta - \beta_Z\nabla_\theta J_{\mathbb{Z}}(\theta)$
    Modify actor by $\phi \leftarrow \phi - \beta_\pi\nabla_\phi J_{\mathbb{Z}}(\phi)$
    Modify temperature by Equation (17)
    Modify target by
    $\bar{\theta} \leftarrow \tau\theta + (1-\tau)\bar{\theta}$, $\bar{\phi} \leftarrow \tau\phi + (1-\tau)\bar{\phi}$
  end while
end while

4. EXPERIMENTAL RESULTS

The suggested framework for building detection using high-resolution images is assessed across three commonly used public datasets renowned for evaluation purposes [22]. Herein, the section details the specifics of implementation and outlines the achieved performance metrics.

4.1 Datasets

The WHU building dataset serves as a challenging benchmark for evaluating building detection methods. It includes the Aerial Imagery dataset, Satellite dataset I, and Satellite dataset II. Our evaluation primarily focuses on the Aerial Imagery dataset within the WHU collection to gauge the effectiveness of our proposed framework. This dataset offers comprehensive coverage of diverse land types, encompassing urban, rural, industrial, and residential areas.

Additionally, the Massachusetts building dataset is utilized for performance evaluation. The dataset includes 137 aerial images with a spatial resolution of 1 m, encompassing about 2.25 km^2 in urban and suburban regions near Boston. It consists of 137 training images, 4 validation images, and 10 testing images. The dataset adheres to a labeling scheme akin to ground truth images, discerning buildings from background objects.

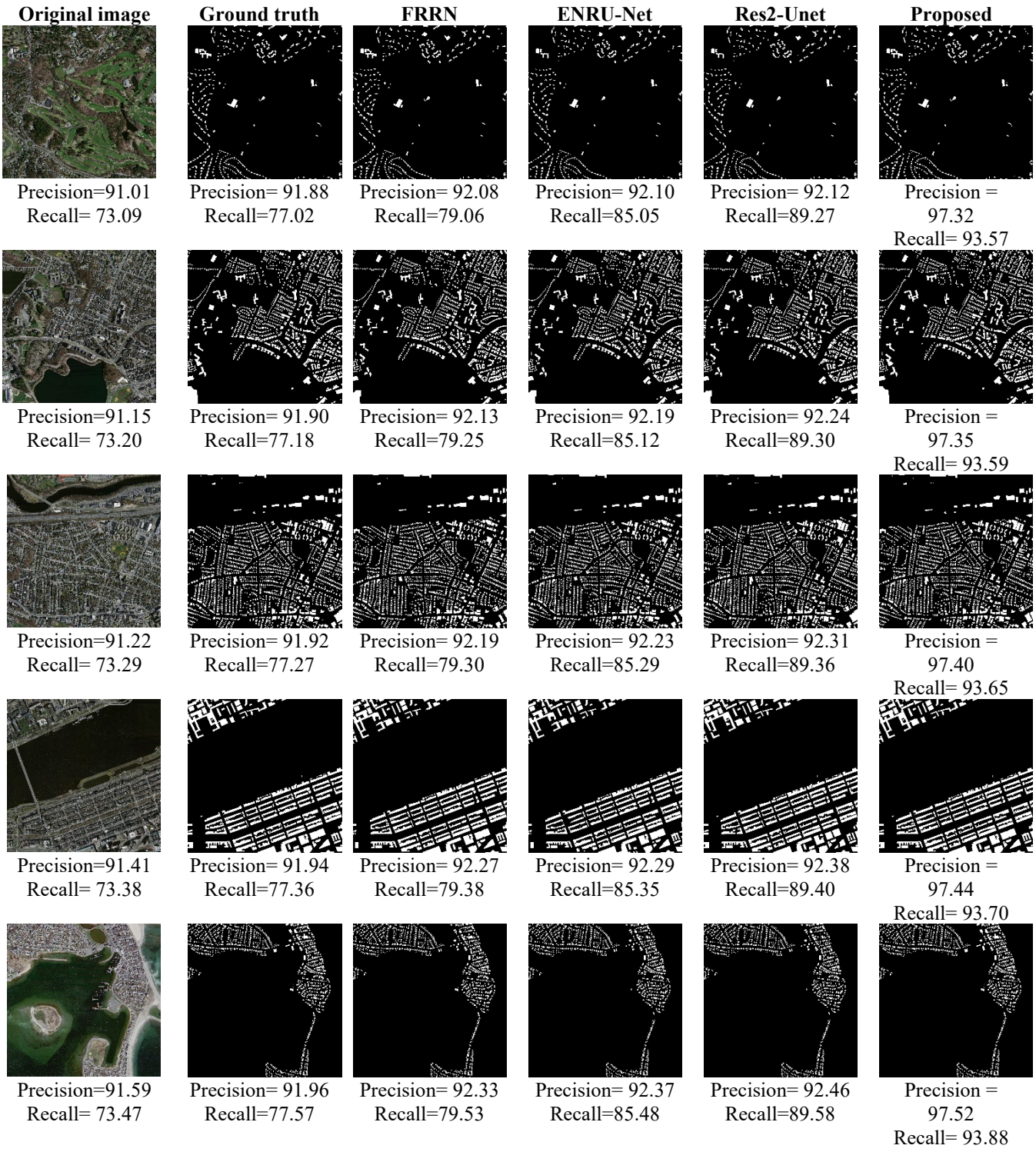

Figure 3. Visual comparisons of Massachusetts building dataset (Aerial images).

Figure 3 illustrates the output of segmenting the buildings from the aerial images of Massachusetts building dataset with various state-of-the-arts methods. This image helps to detect the loss of boundary in the buildings for various existing methods like Full-resolution residual networks (FRRN), Efficient Non-local Residual U-shape Network (ENRU-Net) and Res2-Unet. From the visual evaluation it is found that the proposed Faster RCNN with DSAC-T produces less boundary loss and also obtains high feature learning ability. Hence, the small buildings are also segmented in from the original images correctly.

4.2 Experimental Setup

The parameter settings outlined in Table 1 constitute key configurations for the reinforcement learning algorithm employed in the experimental setup. The choice of the Adam optimizer, with actor and critic learning rates set at 1e−4, influences the optimization process for both policy and value networks. A policy update interval of 2 determines how frequently the actor network is updated, impacting the learning trajectory. The discount factor (γ) of 0.99 emphasizes long-term rewards in decision-making. The target smoothing coefficient (τ) at 0.005 ensures a gradual update process for stability. The reward scale

of 1 adjusts the impact of rewards, and the learning rate of α at 3e−4 influences the balance between exploration and exploitation in the maximum-entropy framework. Together, these parameters play a crucial role in defining the algorithm's learning dynamics and effectiveness.

Table 1. Parameter settings.

Parameters	**Value**
Optimizer	Adam
Actor learning rate	1e−4
Target smoothing_coefficient (τ)	0.005
Policy update interval	2
Discount factor(γ)	0.99
Reward scale	1
Learning rate of α	3e−4
Critic learning_rate	1e−4

4.3 Performance Metrics

The proposed approach assesses the performance using several metrics: average precision (AP), average recall (AR), average runtime per image (ms), and the ratio of sampled high resolution (HR) images. To derive AP and AR, individual values are computed across various categories for IoU thresholds ranging from 0.50 to 0.95.

Table 2 presents a comprehensive technical assessment of building detection performance across various reinforcement learning models, including DQN, DDPG, TD3, SAC, DSAC, and the proposed model, on two distinct datasets: the Massachusetts and WHU building datasets. The evaluation metrics employed cover critical aspects of model efficacy and efficiency. Average Precision (AP) and Average Recall (AR) provide insights into the accuracy and recall capabilities of each model, respectively. Run-time, measured in milliseconds, signifies the computational efficiency of the models. High-Resolution Usage (HR) quantifies the percentage of high-resolution images utilized by each model, offering insights into computational resource utilization. Intersection over Union (IoU) measures spatial accuracy by assessing the overlap between predicted building regions and ground truth. The proposed model consistently outperforms benchmark models across AP and AR metrics, showcasing superior accuracy and recall. Furthermore, it demonstrates lower run-time, indicating enhanced computational efficiency. The reduced usage of high-resolution images underscores its resource-efficient nature. Additionally, higher IoU values emphasize the proposed model's spatial precision in aligning predicted and actual building locations. This detailed technical comparison illustrates the efficacy and efficiency of the proposed building detection method within a reinforcement learning framework. Concerning image quality, the proposed approach consistently outperforms other reinforcement learning models on both the Massachusetts and WHU building datasets. Specifically, it achieves higher AP and AR scores compared to benchmark models, indicating superior accuracy and coverage in building detection, even under challenging image conditions. Moreover, the proposed approach demonstrates lower run-time values, indicating enhanced computational efficiency without compromising detection performance. Additionally, it utilizes a lower ratio of sampled HR images, minimizing resource utilization while achieving accurate building detection. Higher IoU values for the proposed approach signify better alignment between predicted and ground truth building regions, enhancing spatial precision in detection results. Overall, these findings underscore the superior quality of building detection achieved by the proposed approach compared to other reinforcement learning models.

Table 2. Results for building detection compared to various reinforcement learning models.

Massachusetts building Dataset					
Model	**AP**	**AR**	**Run-time(ms)**	**HR**	**IoU**
DQN	42.69	62.82	384	56.3	72.93
DDPG	48.52	68.54	358	52.4	73.16
TD3	54.91	74.25	302	49.1	73.49
SAC	57.17	77.83	279	46.7	76.58
DSAC	62.35	81.54	245	43.8	80.17
Proposed	69.48	84.29	218	40.4	84.39
WHU building Dataset					
DQN	47.94	66.28	354	76.5	70.36
DDPG	50.22	70.19	322	68.2	74.68
TD3	53.69	75.53	301	59.4	78.92
SAC	56.75	77.38	294	51.2	80.81
DSAC	60.51	79.45	276	49.7	81.75
Proposed	65.82	81.52	235	44.4	83.33

Table 3. Comparisons with the published statistics (%).

Massachusetts Building Dataset				
Method	Precision	Recall	F1-measure	IoU
SegNet [23]	88.2	82.2	85.1	74.0
FCN [23]	89.5	86.7	88.1	78.8
U-Net [23]	89.9	86.9	88.4	80.3
FRRN [23]	92.8	79.6	85.7	74.9
Deeplab-v3 [24]	-	-	81.34	68.55
ENRU-Net [24]	-	-	84.41	73.02
MSCRF [25]	89.93	80.14	84.75	71.19
Res2-Unet [3]	92.12	89.27	90.67	82.93
Proposed	97.32	93.57	95.63	88.29
WHU Dataset				
SiU-Net [26]	65.3	86.9	74.56	59.4
U-Net [26]	72.5	79.6	75.88	61.1
SR-FCN [27]	79.0	77.0	77.99	64.0
Res2-Unet [3]	81.29	78.64	79.99	64.0
Proposed	92.57	90.81	91.36	86.17

In this comprehensive ablation study, the performance of various building segmentation methods was rigorously assessed on the Massachusetts Building Dataset and the WHU Dataset, employing key metrics including Precision, Recall, F1-measure, and Intersection over Union (IoU), as depicted in Table 3. Among the established techniques, such as SegNet, FCN, U-Net, FRRN, Deeplab-v3, ENRU-Net, MSCRF, and Res2-Unet, each exhibited distinct strengths and weaknesses in addressing building segmentation challenges. Particularly, the Res2-Unet method demonstrated a well-balanced performance on the Massachusetts dataset. However, the proposed approach consistently surpassed these methods across all metrics on both datasets, showcasing superior Precision, Recall, F1-measure, and IoU scores.

Regarding the quality of images, the comparison in Table 3 offers valuable insights into the efficacy of various building segmentation methods. Generally, higher values in Precision, Recall, F1-measure, and IoU metrics indicate better-quality segmentation results. The proposed approach consistently outperformed other methods in all metrics, indicating its ability to accurately identify building structures with high confidence while minimizing false positives and negatives. Additionally, the higher IoU scores for the proposed approach suggest better alignment between predicted and ground truth building regions, highlighting its proficiency in accurately delineating building boundaries.

The superiority of the proposed method can be attributed to several key factors. First and foremost, the innovative design of the model leverages a refined architecture, integrating advanced features such as the FRPN and a modified ResNet-50 backbone. This architecture enhances feature extraction, allowing for more precise localization of building structures. Additionally, the proposed method addresses the challenge of imbalanced region proposals through the FRPN, ensuring a more comprehensive and effective selection of candidate regions. The IoU metric, reflecting the overlap between predicted and ground truth regions, benefits significantly from this improved region proposal mechanism.

Furthermore, the proposed approach exhibits exceptional adaptability across diverse datasets, showcasing its robustness in handling different building representations. The model's superior performance on both datasets underscores its generalizability and effectiveness in capturing intricate features related to building structures. In summary, the proposed method surpasses existing techniques due to its innovative architecture, effective region proposal strategy, and robust adaptability, making it a promising solution for accurate and reliable building segmentation in diverse scenarios.

5. CONCLUSION

This research introduces a comprehensive and innovative methodology for building detection that leverages advanced techniques in deep learning and reinforcement learning. The proposed Faster RCNN framework, enhanced by the FRPN, demonstrates superior feature extraction, enabling precise and near real-time building detection. The adaptive region proposals introduced by FRPN contribute to increased computational efficiency and better candidate region selection. Furthermore, the DSAC-T algorithm is refined through three key enhancements: critic gradient adjustment, twin value distribution learning, and variance-based target return clipping. These refinements address learning instabilities and reward scaling sensitivity, making the DSAC algorithm more robust for building detection tasks. Extensive evaluations on benchmark datasets highlight the superiority of the proposed methodology over state-of-the-art techniques. The model consistently outperforms in terms of accuracy, F1-measure, recall, and IoU metrics. Visual comparisons demonstrate enhanced boundary preservation and feature learning capabilities. The research contributes a promising solution for accurate and reliable building segmentation in diverse scenarios. The proposed methodology's adaptability across datasets, computational efficiency, and robust performance position it as a valuable advancement in the field of building detection. Future work may explore additional refinements and applications, further solidifying the proposed approach as a cornerstone in automated building detection systems.

ACKNOWLEDGEMENT AND FUNDING

The authors receive no financial support for the research, authorship, and publication of this article.

DECLARATION OF CONFLICTING INTERESTS

The authors declare no potential conflicts of interest with respect to the research and publication of this article.

REFERENCES

[1] X. Hou, Y. Bai, Y. Li, C. Shang and Q. Shen, High-resolution triplet network with dynamic multiscale feature for change detection on satellite images, *ISPRS Journal of Photogrammetry and Remote Sensing*, 177, 2021, 103-115.

[2] J. Li, X. Huang, L. Tu, T. Zhang and L. Wang, A review of building detection from very high resolution optical remote sensing images. *GIScience & Remote Sensing*, 59(1), 2022, 1199-1225.

[3] F. Chen, N. Wang, B. Yu and L. Wang, Res2-Unet, a new deep architecture for building detection from high spatial resolution images, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 2022, 1494-1501.

[4] T. Ujiie, M. Hiromoto and T. Sato, Approximated prediction strategy for reducing power consumption of convolutional neural network processor, *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, Las Vegas, USA, 2016.

[5] D. Erhan, C. Szegedy, A. Toshev and D. Anguelov, Scalable object detection using deep neural networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, 2155-2162.

[6] S. Ren, K. He, R. Girshick and J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 2016, 1137-1149.

[7] A. Greenwald, K. Hall and R. Serrano, Correlated Q-learning, *Twentieth International Conference on Machine Learning* (*ICML*), Washington, USA, 3, 2003, 242-249.

[8] M. M. Hassan, M. G. R. Alam, M. Z. Uddin, S. Huda, A. Almogren and G. Fortino, Human emotion recognition using deep belief network architecture, *Information Fusion*, 51, 2019, 10-18.

[9] G. Zuo, T. Du and J. Lu, Double DQN method for object detection. *Proceedings of the Chinese Automation Congress (CAC)*, Jinan, China, 2017, 6727-6732.

[10] X. Zhou, *Deep-Q-Network-Facilitated Object Detection*, Project Report, Stanford University, USA, 2021.

[11] M. Samiei and R. Li, Object detection with deep reinforcement learning, *arXiv:2208.04511*, 2022.

[12] S. Zheng and H. Wang, Real-time visual object tracking based on reinforcement learning with twin delayed deep deterministic algorithm, *9th International Conference Intelligence Science and Big Data Engineering. Visual Data Engineering,* Nanjing, China, 2019, 165-177.

[13] J. Duan, S. E. Li, Y. Guan, Q. Sun and B. Cheng, Hierarchical reinforcement learning for self-driving decision-making without reliance on labelled driving data, *IET Intelligent Transport Systems*, 14(5), 2020, 297-305.

[14] C. J. C. H. Watkins, *Learning from Delayed Rewards*, PhD Thesis, King's College, Cambridge, UK, 1989.

[15] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An introduction*, MIT Press, 2018.

[16] H. Van Hasselt, A. Guez and D. Silver, Deep reinforcement learning with double q-learning, *Proceedings of the AAAI Conference on Artificial Intelligence*, Phoenix, USA, 30(1), 2016, 1-7.

[17] H. Hasselt, Double q-learning, *23rd Advances in Neural Information Processing Systems (NeurIPS 2010)*, Vancouver, Canada, 2010, 2613–2621.

[18] S. Fujimoto, H. Hoof and D. Meger, Addressing function approximation error in actor-critic methods, *Proceedings of the International Conference on Machine Learning*, Stockholm, Sweden, 2018, 1587-1596.

[19] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel and S. Levine, Soft actor-critic algorithms and applications, *arXiv:1812.05905*, 2018.

[20] T. Haarnoja, A. Zhou, P. Abbeel and S. Levine, Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, *Proceedings of the International Conference on Machine Learning,* Stockholm, Sweden, 2018, 1861-1870.

[21] J. Duan, Y. Guan, S. E. Li, Y. Ren, Q. Sun and B. Cheng, Distributional soft actor-critic: Off-policy reinforcement learning for addressing value estimation errors, *IEEE Transactions on Neural Networks and Learning Systems*, 33(11), 2021, 6584-6598.

[22] R. Girshick, Fast R-CNN, *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, 2015, 1440-1448.

[23] Y. Liu, L. Gross, Z. Li, X. Li, X. Fan and W. Qi, Automatic building extraction on high-resolution remote sensing imagery using deep convolutional encoder-decoder with spatial pyramid pooling, *IEEE Access*, 7, 2019, 128774-128786.

[24] S. Wang, X. Hou and X. Zhao, Automatic building extraction from high-resolution aerial imagery via fully convolutional encoder-decoder network with non-local block, *IEEE Access*, 8, 2020, 7313-7322.

[25] Q. Zhu, Z. Li, Y. Zhang and Q. Guan, Building extraction from high spatial resolution remote sensing images via multiscale-aware and segmentation-prior conditional random fields, *Remote Sensing*, 12(23), 2020, 3983.

[26] S. Ji, S. Wei and M. Lu, A scale robust convolutional neural network for automatic building extraction from aerial and satellite imagery, *International Journal of Remote Sensing*, 40(9), 2019, 3308-3322.

[27] S. Ji, S. Wei and M. Lu, Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set, *IEEE Transactions on Geoscience and Remote Sensing*, 57(1), 2018, 574-586.