

Online ResNet-Based Adaptive Control for Nonlinear Target Tracking

Cristian F. Nino[®], *Graduate Student Member, IEEE*, Omkar Sudhir Patil[®], Jordan C. Insinger, Marla R. Eisman, and Warren E. Dixon[®], *Fellow, IEEE*

Abstract—A generalized ResNet architecture for adaptive control of nonlinear systems with black box uncertainties is developed. The approach overcomes limitations in existing methods by incorporating preactivation shortcut connections and a zeroth layer block that accommodates different input-output dimensions. The developed Lyapunov-based adaptation law establishes exponential convergence to a neighborhood of the target state despite unknown dynamics and disturbances. Furthermore, the theoretical results are validated through a comparative experiment.

Index Terms—Neural networks, adaptive control, stability of nonlinear systems.

I. INTRODUCTION

TEURAL networks (NNs) are well established for approximating unstructured uncertainties in continuous functions over compact domains [1], [2], [3], [4]. The evolution of NN-based control has progressed from single-layer architectures with Lyapunov-based adaptation [5], [6], [7] to more complex deep neural network (DNN) implementations, motivated by numerous examples of improved function approximation efficiency [8], [9], [10]. Early DNN approaches develop Lyapunov-based adaptive update laws for the output layer while the inner layers are updated either in an iterative offline manner as in [11] and [12], or using modular adaptive update laws [13]. Recent developments have established frameworks for real-time adaptation of all DNN layers [14], [15] for various DNN architectures, addressing issues in transient performance [16] and leveraging persistence of excitation [17].

Deep residual neural network (ResNet) architectures have emerged as particularly promising candidates for adaptive control applications. The ResNet architecture is popular because

Received 17 March 2025; revised 9 May 2025; accepted 27 May 2025. Date of publication 4 June 2025; date of current version 30 June 2025. This work was supported in part by AFOSR under Grant FA9550-19-1-0169; in part by AFRL under Grant FA8651-24-1-0018; and in part by AFOSR under Grant FA9550-21-1-0157. Recommended by Senior Editor A. Girard. (*Corresponding author: Cristian F. Nino.*)

The authors are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: cristian1928@ufl.edu; patilomkarsudhir@ufl.edu; jordan.insinger@ ufl.edu; marlaeisman@ufl.edu; wdixon@ufl.edu).

Digital Object Identifier 10.1109/LCSYS.2025.3576652

it addresses optimization challenges that arise with increasing network depth, making them potentially valuable for modeling complex system uncertainties.

The key innovation of ResNets is the introduction of skip connections that create direct paths for information to flow through the network during backpropagation as in [18] and [19]. These skip connections help prevent the degradation of gradient information as it passes through multiple layersa phenomenon known as the vanishing gradient problem where the magnitude of gradients becomes too small for effective weight updates in deep networks. Rather than learning completely new representations at each layer, ResNets learn the difference (or "residual") between the input and the desired output of a layer, which simplifies the optimization process. Theoretical analyses have demonstrated that ResNets possess favorable optimization properties, including smoother loss landscapes [20], absence of spurious local optima with every local minimum being a global minimum as in [21] and [22], and stability of gradient descent equilibria [23]. The universal approximation capabilities of ResNets have also been investigated [24], [25], [26], confirming their ability to approximate any continuous function on a compact set to arbitrary accuracy. A critical advancement in ResNet design was the introduction of pre-activation shortcuts by [27], which position the skip connection before activation functions, improving the flow of information through the network and enhancing the network's ability to generalize to unseen data. This architectural modification shares conceptual similarities with DenseNets [28], which strengthen feature propagation through dense connectivity patterns that connect each layer to every other layer, facilitating feature reuse and enhancing information flow throughout the network.

Recently, [29] introduced the first Lyapunov-based ResNet for online learning in control applications. However, their implementation utilized the original ResNet architecture without incorporating the pre-activation shortcut connections that have been demonstrated to improve performance [27]. This limitation potentially restricts the learning capabilities and convergence properties of their approach. Additionally, this approach requires the inputs and outputs of the ResNet architecture to be of the same dimensions, thus limiting the applicability of the development.

This letter presents a generalized ResNet architecture featuring pre-activation shortcut connections and a zeroth layer

2475-1456 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information. block designed for target tracking of nonlinear systems. The developed approach positions the skip connection before activation, with post-activation feeding into the DNN block, leveraging the improved information propagation through the network established by [27]. Additionally, the zeroth layer block can compensate for uncertainties with different input and output size, overcoming the limitations of the approaches in [29]. The key contribution of this letter is the development and analysis of a Lyapunov-based adaptation law for this generalized ResNet architecture, which establishes exponential convergence to a neighborhood of the target state despite unknown dynamics and disturbances. Furthermore, the theoretical results are validated through a comparative experiment.

II. DEEP RESIDUAL NEURAL NETWORK MODEL

Consider a fully connected feedforward ResNet with $b \in \mathbb{Z}_{\geq 0}$ building blocks, input $x \in \mathbb{R}^{L_{in}}$, and output $y \in \mathbb{R}^{L_{out}}$. For each block index $i \in \{0, \ldots, b\}$, let $k_i \in \mathbb{Z}_{>0}$ be the number of hidden layers in the *i*th block, let $\kappa_i \in \mathbb{R}^{L_{i,0}}$ denote the block input (with $\kappa_0 \triangleq x$ and $L_{0,0} \triangleq L_{in}$), and let $\theta_i \in \mathbb{R}^{p_i}$ be the vector of parameters (weights and biases) associated with the *i*th block.

For each block $i \in \{0, ..., b\}$, let $L_{i,j} \in \mathbb{Z}_{>0}$ denote the number of neurons in the j^{th} layer for $j \in \{0, ..., k_i + 1\}$. Furthermore, define the augmented dimension $L_{i,j}^a \triangleq L_{i,j} + 1$, for all $(i, j) \in \{0, ..., b\} \times \{0, ..., k_i\}$. Each block function $\Phi_i : \mathbb{R}^{L_{i,0}^a} \times \mathbb{R}^{p_i} \to \mathbb{R}^{L_{i,k_i+1}}$ is a fully connected feedforward DNN, with $L_{i,k_i+1} \triangleq L_{\text{out}}$ for all $(i, j) \in \{0, ..., b\} \times \{0, ..., k_i\}$. For any input $v \in \mathbb{R}^{L_{i,j}^a}$, the DNN is defined recursively by

$$\varphi_{i,j}(v) \triangleq \begin{cases} V_{i,0}^{\top}v, & j = 0, \\ V_{i,j}^{\top}\phi_{i,j}(\varphi_{i,j-1}(v)), & j \in \{1, \dots, k_i\}, \end{cases}$$
(1)

with $\Phi_i(v, \theta_i) = \varphi_{i,k_i}(v)$.

For each $j \in \{0, 1, ..., k_i\}$ the matrix $V_{i,j} \in \mathbb{R}^{L_{i,j}^a \times L_{i,j+1}}$ contains the weights and biases; in particular, if a layer has n(augmented) inputs and the subsequent layer has m nodes, then $V \in \mathbb{R}^{n \times m}$ is constructed so that its $(i, j)^{\text{th}}$ entry represents the weight from the i^{th} node of the input to the j^{th} node of the output, with the last row corresponding to the bias terms. For the DNN architecture described by (1), the vector of DNN weights of the i^{th} block is $\theta_i \triangleq [\operatorname{vec}(V_{i,0})^\top \cdots \operatorname{vec}(V_{i,k_i})^\top]^\top \in$ \mathbb{R}^{p_i} , where $p_i \triangleq \sum_{j=0}^{k_i} L_{i,j}^a L_{i,j+1}$, and $\operatorname{vec}(V_{i,j})$ denotes the vectorization of $V_{i,j}$ performed in column-major order (i.e., the columns are stacked sequentially to form a vector). The activation function $\phi_{i,j} : \mathbb{R}^{L_{i,j}} \to \mathbb{R}^{L_{i,j}^a}$ is given by $\phi_{i,j}(\varphi_{i,j-1})$ $= [\zeta_{i,1}((\varphi_{i,j-1})_1) \zeta_{i,2}((\varphi_{i,j-1})_2) \cdots \zeta_{i,L_{i,j}}((\varphi_{i,j-1})_{L_{i,j}}) 1]^\top \in$ $\mathbb{R}^{L_{i,j}^a}$ where $(\varphi_{i,j-1})_\ell$ denotes the ℓ^{th} component of $\varphi_{i,j-1}$, each $\zeta_{i,j} : \mathbb{R} \to \mathbb{R}$ denotes a smooth activation function, and 1 accounts for the bias term.

A pre-activation design is used so that, before each block (except block 0), the output of the previous block is processed by an external activation function. Specifically, for each block $i \in \{1, ..., b\}$, define the pre-activation mapping $\psi_i : \mathbb{R}^{L_{i,k_i+1}} \to \mathbb{R}^{L_{i,k_i+1}^i}$ by $\psi_i(\kappa_i) = [\varrho_{i,1}((\kappa_i)_1) \ \varrho_{i,2}((\kappa_i)_2)$



Fig. 1. Deep Residual Neural Network Architecture.

 $\cdots \ \varrho_{i,L_{i,k_{i}+1}}((\kappa_{i})_{L_{i,k_{i}+1}}) \ 1]^{\top} \in \mathbb{R}^{L_{i,k_{i}+1}^{a}}$ where $(\kappa_{i})_{\ell}$ denotes the ℓ^{th} component of κ_{i} , each $\varrho_{i,j+1} : \mathbb{R} \to \mathbb{R}$ denotes a smooth activation function, and 1 accounts for the bias term. The output of ψ_{i} serves as the input to block *i* and the residual connection is implemented by adding the current block output to the pre-activated output from the previous block. Hence, the ResNet recursion is defined by

$$\kappa_{i+1} \triangleq \begin{cases} \Phi_0(\kappa_0^a, \theta_0), & i = 0, \\ \kappa_i + \Phi_i(\psi_i(\kappa_i), \theta_i), & i \in \{1, \dots, b\}, \end{cases}$$
(2)

with output $y \in \mathbb{R}^{L_{\text{out}}}$ and overall parameter vector $\Theta \triangleq [\theta_0^\top \cdots \theta_b^\top]^\top \in \mathbb{R}^p$, with $p \triangleq \sum_{i=0}^b p_i$, where $\kappa_0^a \triangleq [\kappa_0^\top 1]^\top \in \mathbb{R}^{L_{0,0}^a}$ denotes the augmented input to block i = 0. Therefore, the complete ResNet is represented as $\Psi : \mathbb{R}^{L_{\text{in}}} \times \mathbb{R}^p \to \mathbb{R}^{L_{\text{out}}}$ expressed as $\Psi(\kappa, \Theta) = \kappa_{b+1}$.

The partial derivative of the ResNet with respect to the parameters is represented as $\frac{\partial}{\partial \Theta} \Psi(\kappa, \Theta) = [\frac{\partial}{\partial \theta_0} \Psi(\kappa, \Theta) \cdots \frac{\partial}{\partial \theta_b} \Psi(\kappa, \Theta)] \in \mathbb{R}^{L_{\text{out}} \times p}$ and $\frac{\partial}{\partial \theta_i} \Psi(\kappa, \Theta) = [\frac{\partial}{\partial \text{vec}(V_{i,0})} \Psi(\kappa, \Theta) \cdots \frac{\partial}{\partial \text{vec}(V_{i,k})} \Psi(\kappa, \Theta)] \in \mathbb{R}^{L_{\text{out}} \times p_i}$, where $\frac{\partial}{\partial \text{vec}(V_{i,j})} \Psi(\kappa, \Theta) \in \mathbb{R}^{L_{\text{out}} \times L_{i,j}^d L_{i,j+1}}$ for all $j \in \{0, \dots, k_i\}$. Using (1), (2), and the property of the vectorization operator yields

$$\frac{\partial \Psi}{\partial \operatorname{vec}(V_{i,j})} = \left(\prod_{m=i+1}^{\infty} \left(I_{L_{\text{out}}} + \left(\prod_{\ell=1}^{k_m} V_{m,\ell}^\top \frac{\partial \phi_{m,\ell}}{\partial \varphi_{m,\ell-1}} \right) V_{m,0}^\top \frac{\partial \psi_m}{\partial \kappa_m} \right) \right)$$
$$\cdot \left(\prod_{\ell=j+1}^{k_i} V_{i,\ell}^\top \frac{\partial \phi_{i,\ell}}{\partial \varphi_{i,\ell-1}} \right) \left(I_{L_{i,j+1}} \otimes \varkappa_{i,j}^\top \right),$$

where $\varkappa_{i,j} \triangleq \kappa_0^a$ if i = 0 and j = 0, $\phi_{0,j}(\varphi_{0,j-1}(\kappa_0^a))$ if i = 0and j > 0, $\psi_i(\kappa_i)$ if i > 0 and j = 0, $\phi_{i,j}(\varphi_{i,j-1}(\psi_i(\kappa_i)))$ if i > 0 and j > 0, where the right-to-left matrix product operator \bigcap is defined as in [14].

$$\begin{split} & \bigcap_{i=1}^{n} \text{ is defined as in [14].} \\ & \text{The Jacobian } \frac{\partial \phi_{i,j}(\varphi_{i,j-1}(\nu))}{\partial \varphi_{i,j-1}(\nu)} : \mathbb{R}^{L_{i,j}} \to \mathbb{R}^{L_{i,j}^{a} \times L_{i,j}} \text{ of the activation function vector at the } j^{\text{th}} \text{ layer is given by } \frac{\partial \phi_{i,j}(\varphi_{i,j-1}(\nu))}{\partial \varphi_{i,j-1}(\nu)} \\ & = \left[\text{diag} \{ \frac{d_{\varsigma_{i,1}((\varphi_{i,j-1})_{1})}}{d(\varphi_{i,j-1})_{1}}, \dots, \frac{d_{\varsigma_{i,L_{i,j}}((\varphi_{i,j-1})_{L_{i,j}})}{d(\varphi_{i,j-1})_{L_{i,j}}} \} \mathbf{0}_{L_{i,j}}^{\mathsf{T}} \right]^{\mathsf{T}}. \text{ Similarly,} \\ & \text{the Jacobian } \frac{\partial \psi_m(\kappa_m)}{\partial \kappa_m} : \mathbb{R}^{L_{\text{out}}} \to \mathbb{R}^{L_{\text{out}}^a \times L_{\text{out}}} \text{ of the pre-activation function vector at block } m \text{ is given by } \frac{\partial \psi_m(\kappa_m)}{\partial \kappa_m} = \left[\text{diag} \{ \frac{d \varrho_{m,1}((\kappa_m)_{1})}{d(\kappa_m)_{1}}, \dots, \frac{d \varrho_{m,L_{\text{out}}}((\kappa_m)_{L_{\text{out}}})}{d(\kappa_m)_{L_{\text{out}}}} \} \mathbf{0}_{L_{\text{out}}}^{\mathsf{T}} \right]^{\mathsf{T}}. \end{split}$$

III. PROBLEM FORMULATION

Consider the second-order nonlinear dynamical system described by the differential equation

$$\ddot{q} = f(q, \dot{q}) + g(q, \dot{q}, t)u + \omega(t), \qquad (3)$$

where $t \in \mathbb{R}_{\geq 0}$ denotes time, $q \in \mathbb{R}^n$ represents the generalized position, $\dot{q} \in \mathbb{R}^n$ the generalized velocity, $\ddot{q} \in \mathbb{R}^n$ the generalized acceleration, $f \in \mathcal{C}^1(\mathbb{R}^n \times \mathbb{R}^n; \mathbb{R}^n)$ represents unknown system dynamics, $g \in \mathcal{C}^1(\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0}; \mathbb{R}^{n \times m})$ denotes the known control effectiveness matrix, $\omega \in \mathcal{C}^1(\mathbb{R}_{\geq 0}; \mathbb{R}^n)$ represents an exogenous disturbance, and $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ denotes the control input signal.

The following assumptions and properties hold. First, the matrix $g(q, \dot{q}, t)$ has full row-rank for all $(q, \dot{q}, t) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0}$. Second, the mapping $t \mapsto g(q, \dot{q}, t)$ is uniformly bounded for all states $(q, \dot{q}) \in \mathbb{R}^n \times \mathbb{R}^n$. Third, there exists a known constant $\overline{\omega} \in \mathbb{R}_{\geq 0}$ such that $\|\omega(t)\| \leq \overline{\omega}$ for all $t \in \mathbb{R}_{\geq 0}$.

By the full row-rank property of g, its right Moore-Penrose pseudoinverse $g^+ \in C^1(\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0}; \mathbb{R}^{m \times n})$ exists, defined as $g^+(q, \dot{q}, t) \triangleq g^\top(q, \dot{q}, t)(g(q, \dot{q}, t)g^\top(q, \dot{q}, t))^{-1}$. Furthermore, the mapping $t \mapsto g^+(q, \dot{q}, t)$ is uniformly bounded for all states $(q, \dot{q}) \in \mathbb{R}^n \times \mathbb{R}^n$.

The reference trajectory is governed by the autonomous second-order system

$$\ddot{q}_d = f_d(q_d, \dot{q}_d),\tag{4}$$

where $q_d \in \mathbb{R}^n$ denotes the reference position, $\dot{q}_d \in \mathbb{R}^n$ denotes the reference velocity, $\ddot{q}_d \in \mathbb{R}^n$ denotes the reference acceleration, and $f_d : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ represents the unknown reference dynamics.

Assumption 1: There exist known constants $\bar{q}_d, \dot{\bar{q}}_d \in \mathbb{R}_{>0}$ such that $q_d, \dot{q}_d \in \mathcal{L}_{\infty}(\mathbb{R}_{\geq 0}; \mathbb{R}^n)$.

The control objective is to design a ResNet-based adaptive controller such that the state trajectory q is exponentially regulated to a neighborhood of the reference trajectory q_d , despite the presence of unknown dynamics and bounded disturbances. To facilitate the control objective, define the tracking error $e \in \mathbb{R}^n$ as

$$e \triangleq q_d - q. \tag{5}$$

IV. CONTROL SYNTHESIS

To facilitate the control design, the auxiliary tracking error function $r \in \mathbb{R}^n$ is defined as

$$r \triangleq \dot{e} + k_1 e, \tag{6}$$

where $k_1 \in \mathbb{R}_{>0}$ is a constant control gain. Differentiating (6) with respect to time and substituting (3)–(6) yields

$$\dot{r} = h(q, \dot{q}, q_d, \dot{q}_d) - g(q, \dot{q}, t)u - \omega(t) + k_1(r - k_1 e),$$
(7)

where $h : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is defined as $h(q, \dot{q}, q_d, \dot{q}_d) \triangleq f_d(q_d, \dot{q}_d) - f(q, \dot{q}).$

A. Residual Neural Network Function Approximation

The ResNet architecture, characterized by skip connections and hierarchical feature extraction, models incremental changes rather than complete transformations of the underlying nonlinear mapping. This architecture learns the residuals between input and desired output at each layer, thereby enabling effective function approximation for complex nonlinear systems without requiring explicit governing equations. To approximate the unknown dynamics given by h in (7), define the input vector $\kappa : \mathbb{R}_{\geq 0} \to \mathbb{R}^{4n}$ as $\kappa \triangleq [q^\top \dot{q}^\top q_d^\top \dot{q}_d^\top]^\top \in \Omega$, where $\Omega \subset \mathbb{R}^{4n}$ is a compact set over which the universal approximation property holds. The ResNet-based approximation of $h(\kappa)$ is given by $\Psi(\kappa, \widehat{\Theta})$, where $\Psi : \mathbb{R}^{4n} \times \mathbb{R}^p \to \mathbb{R}^n$ denotes the ResNet architecture mapping and $\widehat{\Theta} \in \mathbb{R}^p$ denotes the adaptive parameter estimate.

The approximation objective is to determine optimal estimates $\widehat{\Theta}$ within a predefined search space such that the mapping $\kappa \mapsto \Psi(\kappa, \widehat{\Theta})$ approximates $\kappa \mapsto h(\kappa)$ with minimal error for all $\kappa \in \Omega$. Let $\mathcal{U} \subset \mathbb{R}^p$ denote a user-selected compact, convex parameter search space with a \mathcal{C}^{∞} boundary, satisfying $\mathbf{0}_p \in \operatorname{int}(\mathcal{U})$, and define $\overline{\Theta} \triangleq \max_{\Theta \in \mathcal{U}} \|\Theta\|$. An objective function $\mathcal{J} : \mathcal{U} \to \mathbb{R}_{\geq 0}$ is selected to quantify the quality of the approximation achieved by the parameters $\Theta \in \mathcal{U}$, where $\Theta \in \mathbb{R}^p$ is an arbitrary parameter vector.¹

Assumption 2: The selected objective function $\mathcal{J}: \mho \to \mathbb{R}_{\geq 0}$ is continuous and strictly convex on the compact, convex set \mho .

Under Assumption 2, the existence and uniqueness of a global minimizer for \mathcal{J} within the search space \mho is guaranteed. This unique optimal parameter vector within the search space is denoted by $\Theta^* \in \mho$, defined as²

$$\Theta^* \triangleq \underset{\Theta \in \mathcal{U}}{\arg\min} \mathcal{J}(\Theta).$$
(8)

Remark 1: The universal function approximation property of ResNets was not invoked in the definition of Θ^* . The universal function approximation theorem states that the function space of ResNets is dense in the space of continuous functions $C(\Omega)$ [24], [26]. Consequently, for any prescribed $\overline{\varepsilon} > 0$, there exists a ResNet Ψ and a corresponding parameter vector Θ such that $\sup \|h(\kappa) - \Psi(\kappa, \Theta)\| < \overline{\varepsilon}$. This implies $\int_{\Omega} \|h(\kappa) - \psi(\kappa, \Theta)\|^2 d\mu(\kappa) < \overline{\varepsilon}^2 \mu(\Omega)$. However, determining a search space \Im for arbitrary $\overline{\varepsilon}$ remains an open challenge. Therefore, \Im is arbitrarily selected, at the expense of guarantees on the approximation accuracy.

B. Control Design

Following the previous discussion, the unknown dynamics $h(\kappa)$ in (7) are modeled using a ResNet as

$$h(\kappa) = \Psi(\kappa, \Theta^*) + \varepsilon(\kappa), \qquad (9)$$

where $\varepsilon : \mathbb{R}^{4n} \to \mathbb{R}^n$ is an unknown function representing the optimal reconstruction error associated with Θ^* , which is bounded as

$$\sup_{\kappa \in \Omega} \|\varepsilon(\kappa)\| < \overline{\varepsilon}. \tag{10}$$

¹The function $\mathcal{J}(\Theta)$ typically incorporates a measure of the approximation error, often based on a norm of the difference $h(\kappa) - \Psi(\kappa, \Theta)$ aggregated over the domain Ω . A common example is the mean squared error functional, $\mathcal{J}_{MSE}(\Theta) \triangleq \int_{\Omega} ||h(\kappa) - \Psi(\kappa, \Theta)||^2 d\mu(\kappa)$, where μ is a suitable measure on Ω . However, the specific choice of \mathcal{J} can be adapted based on the application requirements or available data.

²For ResNets with arbitrary nonlinear activation functions, it has been established that every local minimum is a global minimum [22].

Based on (7) and the subsequent stability analysis, the control input is designed as

$$u = g^{+}(q, \dot{q}, t) \left(\left(1 - k_{1}^{2} \right) e + (k_{1} + k_{2})r + \Psi(\kappa, \widehat{\Theta}) \right), (11)$$

where $k_2 \in \mathbb{R}_{>0}$ is a constant control gain. Substituting (9) and (11) into (7) yields

$$\dot{r} = -k_2 r - e + \varepsilon(\kappa) - \omega(t) + \Psi(\kappa, \Theta^*) - \Psi(\kappa, \widehat{\Theta}).$$
(12)

Based on the subsequent stability analysis, the adaptive update law for $\widehat{\Theta}$ is designed as

$$\widehat{\Theta} = \operatorname{proj}_{\mathcal{O}} \left(\Gamma \left(\frac{\partial \Psi(\kappa, \widehat{\Theta})}{\partial \widehat{\Theta}}^\top r - k_3 \widehat{\Theta} \right) \right), \quad (13)$$

where $k_3 \in \mathbb{R}_{>0}$ is a constant control gain, $\Gamma \in \mathbb{R}^{p \times p}$ is a userdefined positive-definite learning rate matrix, and the projection operator ensures $\widehat{\Theta} \in \mathcal{O}$, defined as in [30, Appendix E].

V. STABILITY ANALYSIS

The ResNet mapping $\Psi(\kappa, \Theta)$ described in (9) is inherently nonlinear with respect to its weights. Designing adaptive controllers and performing stability analyses for systems that are nonlinearly parameterizable presents significant theoretical challenges. A method to address the nonlinear structure of the uncertainty, especially for ResNets, is to use a first-order Taylor series approximation. To quantify the approximation, the parameter estimation error $\widetilde{\Theta} \in \mathbb{R}^p$ is defined as

$$\widetilde{\Theta} = \Theta^* - \widehat{\Theta}. \tag{14}$$

Applying first-order Taylor's theorem to the mapping $\Theta \mapsto \Psi(\kappa, \Theta)$ evaluated at $\widehat{\Theta}$ and using (14) yields

$$\Psi(\kappa, \Theta^*) - \Psi(\kappa, \widehat{\Theta}) = \frac{\partial \Psi(\kappa, \widehat{\Theta})}{\partial \widehat{\Theta}} \widetilde{\Theta} + R(\kappa, \widetilde{\Theta}), \quad (15)$$

where $R : \mathbb{R}^{4n} \times \mathbb{R}^{p} \to \mathbb{R}^{n}$ denotes the Lagrange remainder term. Substituting (15) into (12) yields

$$\dot{r} = \frac{\partial \Psi(\kappa, \widehat{\Theta})}{\partial \widehat{\Theta}} \widetilde{\Theta} + R(\kappa, \widetilde{\Theta}) + \varepsilon(\kappa) - k_2 r - e - \omega(t).$$
(16)

Let $z \in \mathbb{R}^{\varphi}$ denote the concatenated state vector $z \triangleq [e^{\top} r^{\top} \widetilde{\Theta}^{\top}]^{\top}$, where $\varphi \triangleq 2n + p$. The evolution of z is governed by the initial value problem

$$\dot{z} = f(z, t), \ z(t_0) = z_0,$$
 (17)

where $t_0 \ge 0$ is the initial time and $z_0 \in \mathbb{R}^{\varphi}$ is the initial state. Using (5), (13), (14), and (16), the vector field $f : \mathbb{R}^{\varphi} \times \mathbb{R}_{\ge 0} \to \mathbb{R}^{\varphi}$ is defined as

$$f(z,t) = \begin{bmatrix} r - k_1 e \\ \left(\frac{\partial \Psi(\kappa,\widehat{\Theta})}{\partial \widehat{\Theta}} \widetilde{\Theta} + R(\kappa, \widetilde{\Theta}) \\ +\varepsilon(\kappa) - k_2 r - e - \omega(t). \right) \\ -\operatorname{proj}_{U} \left(\Gamma\left(\frac{\partial \Psi(\kappa,\widehat{\Theta})}{\partial \widehat{\Theta}}^{\top} r - k_3 \widehat{\Theta} \right) \right) \end{bmatrix}.$$
(18)

Since the universal approximation property of the ResNet holds only on the compact domain Ω , the subsequent stability analysis requires ensuring $\kappa \in \Omega$. This is achieved by establishing a stability result which constrains the solution *z* to a compact domain. Consider the Lyapunov function candidate $V : \mathbb{R}^{\varphi} \to \mathbb{R}_{>0}$ defined as

$$V(z) \triangleq \frac{1}{2} z^{\top} P z, \qquad (19)$$

where $P \triangleq$ blkdiag{ I_{2n}, Γ^{-1} } $\in \mathbb{R}^{\varphi \times \varphi}$. By the Rayleigh quotient, (19) satisfies

$$\frac{1}{2}\lambda_1 \|z\|^2 \le V(z) \le \frac{1}{2}\lambda_{\varphi} \|z\|^2,$$
(20)

where $\lambda_1 \triangleq \lambda_{\min}\{P\} = \min\{1, \lambda_{\min}\{\Gamma^{-1}\}\}$ and $\lambda_{\varphi} \triangleq \lambda_{\max}\{P\} = \max\{1, \lambda_{\max}\{\Gamma^{-1}\}\}$. Based on the subsequent analysis, define $\delta \triangleq \frac{3(\overline{\omega} + \overline{\varepsilon})^2}{4k_2} + \frac{k_3\overline{\Theta}^2}{2}$ and $k_{\min} \triangleq \min\{k_1, \frac{1}{3}k_2, \frac{1}{2}k_3\}$. Furthermore, let $\rho : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ denote a strictly increasing function that is subsequently defined, and define $\overline{\rho}(\cdot) \triangleq \rho(\cdot) - \rho(0)$, where $\overline{\rho}$ is strictly increasing and invertible. The region to which the state trajectory is constrained is defined as

$$\mathcal{D} \triangleq \left\{ \iota \in \mathbb{R}^{\varphi} : \|\iota\| \le \overline{\rho}^{-1} (k_2(k_{\min} - \lambda_V) - \rho(0)) \right\}, \quad (21)$$

where $\lambda_V \in \mathbb{R}_{>0}$ is a user-defined rate of convergence parameter. The compact domain $\Omega \subset \mathbb{R}^{4n}$ over which the universal approximation property must hold is selected as

$$\Omega \triangleq \left\{ \iota \in \mathbb{R}^{4n} : \|\iota\| \le 2(\overline{q}_d + \overline{\dot{q}}_d) + (k_1 + 2)\overline{\rho}^{-1}(k_2(k_{\min} - \lambda_V) - \rho(0)) \right\}.$$
 (22)

For the dynamical system described by (17), the set of initial conditions $S \subset D$ is defined as

$$S \triangleq \left\{ \iota \in \mathbb{R}^{\varphi} : \|\iota\| \le -\sqrt{\frac{\delta}{\lambda_{V}}} + \sqrt{\frac{\lambda_{1}}{\lambda_{\varphi}}} \overline{\rho}^{-1} (k_{2}(k_{\min} - \lambda_{V}) - \rho(0)) \right\},$$
(23)

and the uniform ultimate bound $\mathcal{U} \subset \mathbb{R}^{\varphi}$ is defined as

$$\mathcal{U} \triangleq \left\{ \iota \in \mathbb{R}^{\varphi} : \|\iota\| \le \sqrt{\frac{\lambda_{\varphi}\delta}{\lambda_1 \lambda_V}} \right\}.$$
 (24)

Theorem 1: Consider the dynamical system described by (3) and (4). For any initial conditions of the state vector $z(t_0) \in S$, the controller given by (11) and the adaptation law given by (13) ensure that *z* uniformly exponentially converges to \mathcal{U} in the sense that

$$|z(t)|| \leq \sqrt{\frac{\lambda_{\varphi}}{\lambda_{1}}} ||z(t_{0})||^{2} e^{-\frac{2\lambda_{V}}{\lambda_{\varphi}}(t-t_{0})} + \frac{\lambda_{\varphi}\delta}{\lambda_{1}\lambda_{V}} \left(1 - e^{-\frac{2\lambda_{V}}{\lambda_{\varphi}}(t-t_{0})}\right),$$

for all $t \in \mathbb{R}_{\geq t_0}$, provided that the sufficient gain condition $k_{\min} > \lambda_V + \frac{1}{k_2}\rho(\sqrt{\frac{\lambda_\varphi\delta}{\lambda_1\lambda_V}})$ is satisfied and Assumptions 1 and 2 hold.

Proof: Taking the total derivative of (19) along the trajectories of (17) yields

$$\frac{\mathrm{d}}{\mathrm{d}t}V(z(t)) = \nabla V(z(t))^{\top} \frac{\mathrm{d}}{\mathrm{d}t}z(t),$$

= $e^{\top}(t)\dot{e}(t) + r^{\top}(t)\dot{r}(t) + \widetilde{\Theta}^{\top}(t)\Gamma^{-1}\widetilde{\Theta}(t).$ (25)

Substituting (18), invoking [30, Appendix E.4], and using (14) yields

$$\frac{\mathrm{d}}{\mathrm{d}t}V(z(t)) \leq -k_1 \|e(t)\|^2 - k_2 \|r(t)\|^2 - k_3 \|\widetilde{\Theta}(t)\|^2
+ r^{\top}(t) \big(R\big(\kappa(t), \widetilde{\Theta}(t)\big) + \varepsilon(\kappa(t)) \big)
- r^{\top}(t)\omega(t) + k_3 \widetilde{\Theta}^{\top}(t)\Theta^*.$$
(26)

Using (5), Assumption 1, and the triangle inequality yields $||q(t)|| \le ||e(t)|| + \overline{q}_d$. Similarly, using (6), Assumption 1, and the triangle inequality yields $||\dot{q}(t)|| \le k_1 ||e(t)|| + ||r(t)|| + \overline{\dot{q}}_d$. Therefore, using the definition of κ yields

$$\|\kappa(t)\| \le (k_1 + 2)\|z(t)\| + 2(\bar{q}_d + \bar{\dot{q}}_d).$$
(27)

From [31, Th. 1], there exists a polynomial $\rho_0(||\kappa||) = a_2||\kappa||^2 + a_1||\kappa|| + a_0$ with $a_2, a_1, a_0 \in \mathbb{R}_{\geq 0}$ such that $||R(\kappa, \widetilde{\Theta})|| \le \rho_0(||\kappa||) ||\widetilde{\Theta}||^2$. Thus, using (8), (10), (14), (27), and the definition of $\overline{\Theta}$ yields $||R(\kappa(t), \widetilde{\Theta}(t)) + \varepsilon(\kappa(t))|| \le 2\overline{\Theta}\rho_0((k_1+2)||z(t)|| + 2(\overline{q}_d + \overline{\dot{q}}_d))||\widetilde{\Theta}(t)|| + \overline{\varepsilon}$.

Since ρ_0 is a polynomial with non-negative coefficients, it is strictly increasing. Consequently, there exists a strictly increasing function $\rho : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ such that $\rho(||z||) = 3\overline{\Theta}^2 \rho_0^2((k_1 + 2)||z|| + 2(\overline{q}_d + \overline{\dot{q}}_d))$. Thus, using Young's inequality and the definitions of δ and k_{\min} yields that (26) is upper bounded as

$$\frac{\mathrm{d}}{\mathrm{d}t}V(z(t)) \le -\left(k_{\min} - \frac{\rho(\|z(t)\|)}{k_2}\right) \|z(t)\|^2 + \delta.$$
(28)

To establish the existence of a solution on $\mathbb{R}_{\geq t_0}$, a contradiction argument is employed. Let $[t_0, T_{\max})$ be the maximal interval of existence for solution $t \mapsto z(t)$ to (17) with $z(t_0) \in S$. Suppose, for contradiction, that $T_{\max} < \infty$. Then $\lim_{t \to T_{\max}^-} ||z(t)|| = \infty$.

Define the set $\mathcal{I} \triangleq \{t \in [t_0, T_{\max}) : z(\tau) \in \mathcal{D} \text{ for all } \tau \in [t_0, t]\}$. Since $z(t_0) \in \mathcal{S} \subset \mathcal{D}$ and the solution $t \mapsto z(t)$ is continuous, \mathcal{I} is non-empty and contains a non-trivial interval $[t_0, t_0 + \eta)$ for some $\eta > 0$. Consequently, using (20) yields that (28) is upper bounded as

$$\frac{\mathrm{d}}{\mathrm{d}t}V(z(t)) \le -\frac{2\lambda_V}{\lambda_{\varphi}}V(z(t)) + \delta, \tag{29}$$

for all $t \in \mathcal{I}$. Solving the differential inequality given by (29) over \mathcal{I} yields

$$V(z(t)) \le V(z(t_0)) e^{-\frac{2\lambda_V}{\lambda_{\varphi}}(t-t_0)} + \frac{\lambda_{\varphi}\delta}{2\lambda_V} \left(1 - e^{-\frac{2\lambda_V}{\lambda_{\varphi}}(t-t_0)}\right), \quad (30)$$

for all $t \in \mathcal{I}$. Applying (20) to (30) yields

$$\|z(t)\| \le \sqrt{\frac{\lambda_{\varphi}}{\lambda_1}} \|z(t_0)\|^2 e^{-\frac{2\lambda_V}{\lambda_{\varphi}}(t-t_0)} + \frac{\lambda_{\varphi}\delta}{\lambda_1\lambda_V} \left(1 - e^{-\frac{2\lambda_V}{\lambda_{\varphi}}(t-t_0)}\right), \quad (31)$$

for all $t \in \mathcal{I}$. Since $z(t_0) \in \mathcal{S}$, $||z(t_0)|| \leq \sqrt{\frac{\lambda_1}{\lambda_{\varphi}}}\overline{\rho}^{-1}(k_2(k_{\min} - \lambda_V) - \rho(0)) - \sqrt{\frac{\delta}{\lambda_V}}$. Substituting this into (31) and using the fact that $e^{-\frac{2\lambda_V}{\lambda_{\varphi}}(t-t_0)} \leq 1$ for all $t \geq t_0$ yields $||z(t)|| < \overline{\rho}^{-1}(k_2(k_{\min} - \lambda_V) - \rho(0))$ for all $t \in \mathcal{I}$. This implies that $z(t) \in int(\mathcal{D})$ for all $t \in \mathcal{I}$. By continuity of z(t), if $\sup \mathcal{I} < T_{\max}$, then $z(\sup \mathcal{I}) \in \partial \mathcal{D}$, which contradicts the established bound. Therefore, $\sup \mathcal{I} = T_{\max}$ and thus $\mathcal{I} = [t_0, T_{\max})$.

TABLE I

PARAMETERS USED IN THE COMPARATIVE EXPERIMENT OF SNN, DNN, AND RESNET -BASED ADAPTIVE CONTROLLERS

	SNN	DNN	ResNet
Neurons	8	2	2
Layers	1	32	2
Blocks	0	0	4
Parameters	254		226
Outer Activation	tanh		
Inner Activation	N/A	Swish ³	
Shortcut Activation	N/A		Swish
Learning Rate	$\Gamma = 0.05$	$\Gamma = 0.1$	$\Gamma = 0.025$
Search Space Bound	$\overline{\Theta} = 4$	$\overline{\Theta} = 8$	$\overline{\Theta} = 1$
Control Gains	$k_1 = 0.77, k_2 = 0.66, k_3 = 1e^{-6}$		

Because the projection operator defined in defined in (13) is locally Lipschitz [30, Lemma E.1], the right-hand side of (17) is piecewise continuous in *t* and locally Lipschitz in *z* for all $t \ge t_0$ and all $z \in \mathbb{R}^{\varphi}$. Since z(t) remains in the compact set \mathcal{D} for all $t \in [t_0, T_{\max})$, the solution is uniformly bounded, meaning $\sup_{t \in [t_0, T_{\max})} ||z(t)|| < \infty$. Therefore, by [32, Th. 3.3], the solution can be extended beyond T_{\max} , contradicting the maximality of $[t_0, T_{\max})$. Therefore, $T_{\max} = \infty$, and the solution exists for all $t \in \mathbb{R}_{\ge t_0}$ with $z(t) \in \mathcal{D}$ for all $t \in \mathbb{R}_{\ge t_0}$.

Consequently, all trajectories with $z(t_0) \in S$ satisfy (31), for all $t \in \mathbb{R}_{\geq t_0}$. As $t \to \infty$, this bound converges to $\|z(t)\| \leq \sqrt{\frac{\lambda_{\varphi}\delta}{\lambda_1\lambda_V}}$ which, by (24), implies that the trajectory z(t) converges to \mathcal{U} . Furthermore, since $z(t) \in \mathcal{D}$ for all $t \in \mathbb{R}_{\geq t_0}$, it follows from (22) and (27) that $\kappa(t) \in \Omega$ for all $t \in \mathbb{R}_{\geq t_0}$, ensuring that the universal approximation property of the ResNet expressed in (9) holds for all time.

Because λ_V is independent of the initial time t_0 or initial condition $z(t_0)$, the exponential convergence is uniform [33]. Additionally, the boundedness of ||z(t)|| implies that ||e(t)||, ||r(t)||, and $||\widetilde{\Theta}(t)||$ are bounded for all $t \in \mathbb{R}_{\geq t_0}$. Therefore, since q_d , $\dot{q}_d \in \mathcal{L}_{\infty}(\mathbb{R}_{\geq 0}; \mathbb{R}^n)$ by Assumption 1, using (5) and (6) yields that $q, \dot{q} \in \mathcal{L}_{\infty}(\mathbb{R}_{\geq 0}; \mathbb{R}^n)$, implying that $g^+(q, \dot{q}, t)$ is bounded. Following (27) and the fact that $z \in$ $\mathcal{L}_{\infty}(\mathbb{R}_{\geq t_0}; \mathbb{R}^{\varphi})$ yields that $\kappa \in \mathcal{L}_{\infty}(\mathbb{R}_{\geq t_0}; \mathbb{R}^{4n})$. Due to the projection operator, $\widehat{\Theta} \in \mathcal{L}_{\infty}(\mathbb{R}_{\geq t_0}; \mathbb{R}^p)$. Since $(\kappa, \widehat{\Theta}) \in$ $\mathcal{L}_{\infty}(\mathbb{R}_{\geq 0}; \mathbb{R}^{4n} \times \mathbb{R}^p)$, $\Psi(\kappa(t), \widehat{\Theta}(t))$ is bounded. Thus, by (11), u is bounded.

VI. EXPERIMENT

Experimental validation was performed on a Freefly Astro quadrotor equipped with a PX4 flight controller at the University of Florida's Autonomy Park outdoor facility. State estimation utilized the onboard EKF2 fusing GPS, optical flow, and Lidar data. A cascaded control architecture was employed, where the proposed ResNet-based controller, implemented as a 50 Hz ROS2 outer loop, generated acceleration commands sent via MAVROS to the PX4's inner-loop controller operating at 400 Hz. The quadrotor autonomously tracked a 15 m × 5 m figure-eight trajectory at 2.5 m altitude for 360 s.

The proposed controller was compared against three benchmarks: a proportional-derivative (PD) controller, a shallow NN-based adaptive controller (SNN) and a DNN-based adaptive controller (employing Φ instead of Ψ in (11)). Controller parameters are provided in Table I. Figure 2 illustrates that



Fig. 2. Tracking error comparison over the 360-second experiment.



Fig. 3. Trajectory comparison over the 360-second experiment.

the proposed ResNet controller reduced tracking error by 22.81%, 8.75%, and 4.76% relative to the SNN, PD, and DNN controllers, respectively, while using approximately 11% fewer parameters compared to the SNN and DNN controllers.

VII. CONCLUSION

The presented generalized ResNet architecture addresses adaptive control of nonlinear systems with black box uncertainties. Key innovations include pre-activation shortcut connections that improve signal propagation and a zeroth layer block that allows handling different input-output dimensions. The Lyapunov-based adaptation law guarantees exponential convergence to a neighborhood of the target state.

ACKNOWLEDGMENT

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsoring agency.

REFERENCES

- M. H. Stone, "The generalized Weierstrass approximation theorem," Math. Mag., vol. 21, no. 4, pp. 167–184, 1948.
- [2] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Netw.*, vol. 4, no. 2, pp. 251–257, 1991.
- [3] P. Kidger and T. Lyons, "Universal approximation with deep narrow networks," in *Proc. Conf. Learn. Theory*, 2020, pp. 2306–2327.
- [4] A. Kratsios and L. Papon, "Universal approximation theorems for differentiable geometric deep learning," J. Mach. Learn. Res., vol. 23, pp. 1–73, Jul. 2022.
- [5] F. L. Lewis, "Nonlinear network structures for feedback control," Asian J. Control, vol. 1, no. 4, pp. 205–228, 1999.

- [6] F. L. Lewis, S. Jagannathan, and A. Yesildirak, *Neural Network Control* of Robot Manipulators and Nonlinear Systems. Philadelphia, PA, USA: CRC Press, 1998.
- [7] P. M. Patre, S. Bhasin, Z. D. Wilcox, and W. E. Dixon, "Composite adaptation for neural network-based controllers," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 944–950, Apr. 2010.
- [8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
- [9] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press 2016.
- [10] D. Rolnick and M. Tegmark, "The power of deeper networks for expressing natural functions," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–14.
- [11] G. Joshi and G. Chowdhary, "Deep model reference adaptive control," in Proc. IEEE Conf. Decis. Control, 2019, pp. 4601–4608.
- [12] R. Sun, M. Greene, D. Le, Z. Bell, G. Chowdhary, and W. E. Dixon, "Lyapunov-based real-time and iterative adjustment of deep neural networks," *IEEE Control Syst. Lett.*, vol. 6, pp. 193–198, 2022.
- [13] D. Le, M. Greene, W. Makumi, and W. E. Dixon, "Real-time modular deep neural network-based adaptive control of nonlinear systems," *IEEE Control Syst. Lett.*, vol. 6, pp. 476–481, 2022.
- [14] O. Patil, D. Le, M. Greene, and W. E. Dixon, "Lyapunov-derived control and adaptive update laws for inner and outer layer weights of a deep neural network," *IEEE Control Syst. Lett.*, vol. 6, pp. 1855–1860, 2022.
- [15] M. Ryu and K. Choi, "CNN-based end-to-end adaptive controller with stability guarantees," 2024, arXiv.2403.03499.
- [16] D. Le, O. Patil, C. Nino, and W. E. Dixon, "Accelerated gradient approach for deep neural network-based adaptive control of unknown nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 4, pp. 6299–6313, Apr. 2025.
- [17] O. S. Patil, E. J. Griffis, W. A. Makumi, and W. E. Dixon, "Composite adaptive Lyapunov-based deep neural network (Lb-DNN) controller," 2023, arXiv.2311.13056.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [19] A. Veit, M. Wilber, and S. Belongie, "Residual networks behave like ensembles of relatively shallow networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 550–558.
- [20] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 6391–6401.
- [21] M. Hardt and T. Ma, "Identity matters in deep learning," 2017, arXiv.1611.04231.
- [22] K. Kawaguchi and Y. Bengio, "Depth with nonlinearity creates no bad local minima in ResNets," *Neural Netw.*, vol. 118, pp. 167–174, 2019.
- [23] K. Nar and S. Sastry, "Residual networks: Lyapunov stability and convex decomposition," 2018, arXiv:1803.08203.
- [24] H. Lin and S. Jegelka, "ResNet with one-neuron hidden layers is a universal approximator," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–10.
- [25] P. Tabuada and B. Gharesifard, "Universal approximation power of deep residual neural networks through the lens of control," *IEEE Trans. Autom. Control*, vol. 68, no. 5, pp. 2715–2728, May 2023.
- [26] C. Liu, E. Liang, and M. Chen, "Characterizing ResNet's universal approximation capability," in *Proc. Mach. Learn. Res.*, 2024, pp. 31477–31515.
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," 2016, arXiv.1603.05027.
- [28] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.
- [29] O. S. Patil, D. M. Le, E. Griffis, and W. E. Dixon, "Deep residual neural network (ResNet)-based adaptive control: A Lyapunov-based approach," in *Proc. IEEE Conf. Decis. Control*, 2022, pp. 3487–3492.
- [30] P. K. M. Krstic and I. Kanellakopoulos, Nonlinear and Adaptive Control Design. New York, NY, USA: Wiley, 1995.
- [31] O. S. Patil, B. C. Fallin, C. F. Nino, R. G. Hart, and W. E. Dixon, "Bounds on deep neural network partial derivatives with respect to parameters," 2025, arXiv.2503.21007.
- [32] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.
- [33] A. Loría and E. Panteley, "Uniform exponential stability of linear time-varying systems: Revisited," *Syst. Control Lett.*, vol. 47, no. 1, pp. 13–24, 2002.
- [34] P. Ramachandran, B. Zoph, and Q. V. Le, "Swish: A self-gated activation function," 2017, arXiv.1710.05941.