REINFORCEMENT LEARNING-BASED TARGET TRACKING FOR UNMANNED AERIAL VEHICLE WITH ACHIEVEMENT REWARDING AND MULTISTAGE TRAINING

NAJM ADDIN MOHAMMED AHMED ABO MOSALI

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

STATUS CONFIRMATION FOR DOCTORAL THESIS

REINFORCEMENT LEARNING-BASED TARGET TRACKING FOR UNMANNED AERIAL VEHICLE WITH ACHIEVEMENT REWARDING AND MULTISTAGE TRANING

ACADEMIC SESSION: 2021/2022

I, **NAJM ADDIN MOHAMMED AHMED ABO MOSALI**, agree to allow this Doctoral Thesis to be kept at the Library under the following terms:

1. This Doctoral Thesis is the property of the Universiti Tun Hussein Onn Malaysia.
2. The library has the right to make copies for educational purposes only.
3. The library is allowed to make copies of this report for educational exchange between higher educational institutions.
4. The library is allowed to make available full text access of the digital copy via the internet by Universiti Tun Hussein Onn Malaysia in downloadable format provided that the Doctoral Thesis is not subject to an embargo. Should an embargo be in place, the digital copy will only be made available as set out above once the embargo has expired.
5. ** Please Mark (√)

[ ]	CONFIDENTIAL	(Contains information of high security or of great importance to Malaysia as STIPULATED under the OFFICIAL SECRET ACT 1972) *Title and Abstract only*
[ ]	RESTRICTED	(Contains restricted information as determined by the Organization/institution where research was conducted) *Title, Abstract and Intro*duction *only*
[ ]		EMBARGO _________ until _________ (date) (date)
[/]	FREE ACCESS	

Approved by,

NAJM ADDIN MOHAMMED AHMED ABO MOSALI

Ts. Dr. SYARIFUL SYAFIQ BIN SHAMSUDIN

Permanent Address:
G3-08, BLOCK G PERWIRA ,
PT RAJA 86400 BATU PAHAT JOHOR

Date : 20/7/2022

Date: 20/7/2022

NOTE:

** If this Doctoral Thesis is classified as CONFIDENTIAL or RESTRICTED, please attach the letter from the relevant authority/organization stating reasons and duration for such classifications.

This thesis has been examined on date 18th April 2022 and is sufficient in fulfilling the scope and quality for the purpose of awarding the Degree of Doctor of Philosophy in Mechanical Engineering.

Chairperson:

ASSOC. PROF. Ts. DR. ZAMRI BIN NORANAI
Faculty of Mechanical and Manufacturing Engineering
Universiti Tun Hussein Onn Malaysia

Examiner:

ASSOC PROF. DR. ARI LEGOWO
Higher College of Technology
United Arab Emirates

PROF. DR. SHAHRUDDIN BIN MAHZAN @ MOHD ZIN
Faculty of Mechanical and Manufacturing Engineering
Universiti Tun Hussein Onn Malaysia

.

REINFORCEMENT LEARNING-BASED TARGET TRACKING FOR UNMANNED AERIAL VEHICLE WITH ACHIEVEMENT REWARDING AND MULTISTAGE TRANING

NAJM ADDIN MOHAMMED AHMED ABO MOSALI

A thesis submitted in
fulfillment of the requirement for the award of the
Doctor of Philosophy in Mechanical Engineering

Faculty of Mechanical and Manufacturing Engineering
Universiti Tun Hussein Onn Malaysia

AUGUST 2022

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged

Student : ....
NAJM ADDIN MOHAMMED AHMED ABO MOSALI

Date : 20/7/2022

Supervisor : ....
Ts. Dr. SYARIFUL SYAFIQ BIN SHAMSDIN

DECLARATION

This thesis is dedicated to: The sake of Allah, my Creator. My great deal, Mohammed S.A.W (May Allah bless and grant him), who taught us the purpose of life; My great parents, who lead me and support; My soul partner in this life my beloved wife Dr Eman Saleh Alerqi, my lovely daughter Lujin and to my princes Yazan; My beloved brothers and sisters and all family; My friends who encourage and support me; I dedicate this research.

ACKNOWLEDGEMENT

Before turning the page off, I would like to express many thanks to those who were truly standing by my side during this Ph.D. journey. I would like to thank all my parents, family members, relatives who have and still support me with their prayers, sponsoring, and unlimited encouragement. I could not find even a word in which I can thank you enough, but you always have the most part of my prayers.

I also would like to thank my true supervisor **Ts. Dr. SYARIFUL SYAFIQ BIN SHAMSUDIN,** whom I am grateful for his support, guidance, and supervision from the beginning until the end of this journey. On top of that, it was my honour to be one of the students who have you as a supervisor of their works. Through him, I thank Universiti Tun Hussein Onn Malaysia for offering such a program that contributes to addressing one of the challenges faced by our world.

To my friends, colleagues, seniors, and teachers, I am grateful for your help and assistance. Thank you again, I could not have pulled this off without you.

Above all, I start and end with thanking ALLAH for giving me the knowledge and blessing to understand and achieve this work. I ask HIM to accept my greatest and sincerest thanks, prayers, and works.

ABSTRACT

Target tracking using an unmanned aerial vehicle (UAV) is a challenging robotic problem. It requires handling a high level of nonlinearity and dynamics device. The aim is to enable accurate target tracking by UAV with responding to the dynamic generated by the target such as sudden trajectory change using reinforcement learning which is proved to learn dynamic effectively. In this thesis, the Twin Delayed Deep Deterministic Policy Gradient Algorithm (TD3), as one recent and composite architecture of reinforcement learning (RL), has been explored as a tracking agent for the problem of UAV-based target tracking. This involved several improvements on the original TD3. First, the proportional-differential controller was used to boost the exploration of the TD3 in training. Second, a novel reward formulation for the UAV-based target tracking was proposed to enable a careful combination of the various dynamic variables in the reward functions. This was accomplished by incorporating two exponential functions to limit the effect of velocity and acceleration to prevent the deformation in the policy function approximation. Third, the concept of multistage training based on the dynamic variables was proposed as an opposing concept to one-stage combinatory training. Fourth, an enhancement of the rewarding function by including piecewise decomposition was used to enable more stable learning behaviour of the policy and move out from the linear reward to the achievement formula. Fifth, a novel agent selection algorithm was developed to enable the selection of the best agent and avoid under-fitting and over-fitting. For the purpose of evaluating the performance of the control system, flight testing was conducted based on three types of target trajectories, namely fixed, square, and blinking. The evaluation was performed in both simulation and real-world experiments. The results showed that the multistage training achieved the best-accomplished performance with both exponential and achievement rewarding for a fixed trained agent with a fixed and square moving target and for a combinatorial agent with both exponential and achievement rewarding for a fixed trained agent in the case of a blinking target. With respect to the traditional proportional differential (PD) controller, the maximum error reduction rate is 86%. The developed achievement rewarding and the multistage training opens the door to various applications of RL in target tracking.

ABSTRAK

Penjejakan sasaran menggunakan pesawat tanpa pemandu (UAV) merupakan masalah robotik yang mencabar yang memerlukan pengendalian alat yang dinamik dan sangat tidak linear. ini. Dalam tesis ini, Algoritma Twin Delayed Deep Deterministic Policy Gradient (TD3), binaan komposit pembelajaran diperkukuh (RL) dan terkini, telah digunakan sebagai agen pengesan untuk menangani masalah penjejakan sasaran berasaskan UAV. Ini melibatkan beberapa penambahbaikan pada algoritma TD3 asal. Pertama, kaedah pengawal pembezaan berkadar digunakan untuk menggalakkan penerokaan TD3 dalam latihan. Kedua, formula ganjaran baharu untuk penjejakan sasaran berasaskan UAV dicadangkan untuk membolehkan gabungan pelbagai pemboleh ubah dinamik yang teliti dalam fungsi ganjaran. Ini dicapai dengan menggabungkan dua fungsi eksponen untuk mengehadkan kesan halaju dan pecutan bagi mengelakkan ubah bentuk dalam anggaran fungsi dasar. Ketiga, konsep latihan berbilang peringkat terhadap pemboleh ubah dinamik dicadangkan berbanding dengan latihan gabungan satu peringkat. Keempat, peningkatan fungsi ganjaran dengan memasukkan penguraian bagian demi bagian telah digunakan untuk membolehkan tingkah laku pembelajaran dasar yang lebih stabil dan beralih daripada ganjaran linear kepada formula pencapaian. Kelima, algoritma pemilihan agen baharu telah dibangunkan untuk membolehkan pemilihan agen terbaik dan mengelak pemadanan yang kurang tepat dan berlebihan. Untuk tujuan penilaian prestasi sistem kawalan, ujian penerbangan telah dijalankan berdasarkan tiga jenis trajektori sasaran, iaitu pegun, segi empat dan berkelip. Penilaian telah dilakukan dalam persekitaran simulasi dan eksperimen. Keputusan menunjukkan bahawa prestasi pencapaian terbaik dicapai melalui latihan berbilang peringkat dengan ganjaran eksponen dan pencapaian untuk agen pelatih pegun dengan sasaran pegun dan bergerak secara persegi dan untuk agen gabungan dengan ganjaran eksponen dan pencapaian untuk agen pelatih pegun dalam kes sasaran berkelip. Kadar pengurangan ralat maksimum adalah sebanyak 86% bagi pengawal pembezaan berkadar (PD) tradisional. Ganjaran pencapaian yang dibangunkan dan latihan berbilang peringkat dapat membuka peluang kepada pelbagai aplikasi RL dalam pengesanan sasaran.

TABLE OF CONTENTS

DECLARATION **iv**
ACKNOWLEDGEMENT **v**
TABLE OF CONTENTS **viii**
LIST OF TABLES **x**
LIST OF FIGURES **xi**
LIST OF ALGORITHMS **xiv**
LIST OF SYMBOLS AND ABBREVIATIONS **xv**
LIST OF APPENDICES
xvii

CHAPTER 1 : INTRODUCTION **1**
1.1 Background of the study 1
1.2 Motivation 3
1.3 Research Problem 3
1.4 Objectives 5
1.5 Scope of study 5
1.6 Contributions 6
1.7 Outline of the thesis 6

CHAPTER 2 : LITERATURE REVIEW **8**
2.1 Introduction 8
2.2 Background 9
2.2.1 History of Quadrotor 9
2.2.2 DOF Airframe Dynamics 12
2.2.3 UAVs application 15
2.2.4 RL based UAVs applications 17
2.2.5 Gazebo Simulation 18
2.2.6 Reinforcement Learning 21
2.2.7 Q-Learning 25
2.2.8 Deep Reinforcement Learning 26
2.2.9 Proximal Policy Optimization 28
2.2.10 Actor-critic methods 29
2.2.11 Twin Delayed Deep Deterministic policy gradient algorithm (TD3) 30
2.2.12 UAV based Reinforcement Learning 31
2.3 Related Works 34
2.3.1 Estimation algorithms for landing or tracking 34
2.3.2 Proportional Integral Differential Controller (PID) 34
2.4 Summary 44

CHAPTER 3 : METHODOLOGY **46**
3.1 Introduction 46
3.2 Methodology 46
3.2.1 Problem Formulation 49
3.2.2 The general framework 50
3.2.3 Observation and State 51
3.2.4 Action 52
3.2.5 Rewarding Model 52
3.2.6 Episode Completion Logic 54
3.3 Experimental Design 55
3.4 Evaluation Metric 57
3.5 Agent Selection Algorithms 58
3.5.1 Policy Surfaces 58
3.5.2 Videos snapshot 58
3.6 Summary 59

CHAPTER 4 : SIMULATION WORK **60**
4.1 Introduction 60
4.2 Simulation Algorithm 60
4.3 Agent Selection Algorithms 63
4.4 Experimental results 64
4.4.1 Experimental evaluation and results 64
4.5 Summary 81

CHAPTER 5 : REAL-WORD EXPERIMENTAL EVALUATION **83**
5.1 Overview 83
5.2 Experimental Setup 83
5.3 Experimental Results and Analysis 84
5.3.1 Fixed Target 84
5.3.2 Moving Target 88
5.3.3 Blinking Target 91
5.4 Summary 94

CHAPTER 6 : CONCLUSION AND FUTURE WORKS **95**
6.1 Introduction 95
6.2 Conclusion 95
6.3 Limitations and Future Works 96

REFERENCES **97**
APPENDIX A **103**
APPENDIX B **110**
VITA **112**

LIST OF TABLES

2.1	Taxonomy of deep reinforcement learning for UAV tasks	25
2.2	Summary of the literature of autonomous landing for PID based landing	35
2.3	Overview of RL-based approaches for UAV tracking application	42
3.1	The parameters of the training for TD3	55
3.2	TD3 parameters	56
3.3	Labelling coding for the models used in the evaluation	57
4.1	Simulation parameters.	61
4.2	Code for naming the method from the achieved objectives perspective	64
4.3	Summary of the errors in Y and Z for F and FM training types and the different types of the agents for fixed target testing	68
4.4	Summary of the errors in Y and Z for F and FM training types and the different types of agents for square target testing	72
4.5	Summary of the errors in Y and Z for F and FM training types and the different types of agents for blinking target testing	76

LIST OF FIGURES

2.1	History of quadrotors. (a) Brèguet–Richet Gyroplane No. 1; (b) Oehmichen No. 2; (c) Bothezat helicopter; (d) Convertawings Model A; (e) Curtiss-Wright VZ-7.	11
2.2	Reference frame and the body frame with their relative geometric relation	14
3.1	Research Methodology developed to Handle the problem of UAV tracking of a moving target	48
3.2	A conceptual diagram of target tracking using the UAV	49
3.3	The conceptual diagram of the developed RL-based tracking	50
4.1	The conceptual diagram of the UAV tracking to the target performed our simulation algorithm	62
4.2	Different agents and the selected ones using Tarrival and HldTimePrc	64
4.3	Boxplot of error on Y- axis for various agent types trained on fixed target and tested on fixed trajectory scenario	66
4.4	Boxplot of error on Z-axis for various agent types trained on fixed target and tested on fixed trajectory scenario	67
4.5	Boxplot of error on Y-axis for various agent types trained on fixed and moving target and tested on fixed trajectory scenario	67
4.6	Boxplot of error on Z-axis for various agent types trained on fixed and moving target and tested on fixed trajectory scenario	68
4.7	Time response of the scenario of one example fixed target scenario for the best agent	69
4.8	Boxplot of error on Y-axis for various agent types trained on fixed target and tested on the square trajectory scenario	70
4. 9	Boxplot of error on Z-axis for various agent types trained on fixed target and tested on the square trajectory scenario	71

4.10	Boxplot of error on Y-axis for various agent types trained on fixed followed by moving target and tested on the square trajectory	71
4.11	Boxplot of error on Z-axis for various agent types trained on fixed followed by moving target and tested on the square trajectory scenario	72
4.12	Time response of the scenario of one example square target scenario for the best agent	73
4.13	Boxplot of error on Y-axis for various agent types trained on fixed target and tested on Blinking Target scenario	74
4.14	Boxplot of error on Z-axis for various agent types trained on fixed target and tested on Blinking Target	75
4.15	Boxplot of error on Y-axis for various agent types trained on fixed followed by moving target and tested on Blinking Target scenario	75
4.16	Boxplot of error on Z-axis for various agent types trained on fixed followed by moving target and tested on Blinking Target scenario	76
4.17	Time response for the scenario of one example blinking target scenario for the best agent	77
4.18	Policy surface of the y control signal based on the relative distance the UAV and the target for agent 523	78
4.19	Policy surface of the z control signal based on the relative distance the UAV and the target for agent 523	78
4.20	The evolution of policy surface of control signals based on the relative distance of the UAV and the target from agent 0 until agent 1836	79
4.21	Snapshot of three trajectories generated from simulation testing of blinking target for (a) agent of type ML with number 1030 (b) ML with number 1060 (c) PID	80
5.1	Boxplot of an error on Y and Z axis for PID and MLEA based on fixed agent tracking	85
5.2	Time response of Y_RL, Z_RL, Y_PID, Z_PID for fixed agent tracking experiment 1	86
5.3	Time response of RL and PID in both Y and Z axes for fixed agent tracking experiment 2	86
5.4	Time response of RL and PID in both Y and Z axes for fixed agent tracking experiment 3	87

5.5	Time response of RL and PID in both Y and Z axes for fixed agent tracking experiment 4	87
5.6	Boxplot of an error on Y and Z axes for PID and MLEA based on moving agent tracking	88
5.7	Time response of RL and PID in both Y and Z axes for moving agent tracking experiment 1	89
5.8	Time response of RL and PID in both Y and Z axes for moving agent tracking experiment 2	89
5.9	Time response of RL and PID in both Y and Z axes for moving agent tracking experiment 3	90
5.10	Time response of RL and PID in both Y and Z axes for moving agent tracking experiment 4	90
5.11	Boxplot of an error on Y and Z axis for PID and MLEA based on blinking agent tracking	91
5.12	Time response of RL and PID in both Y and Z axes for blinking agent tracking experiment 1	92
5.13	Time response of RL and PID in both Y and Z axes for blinking agent tracking experiment 2	92
5.14	Time response of RL and PID in both Y and Z axis for blinking agent tracking experiment 3	93
5.15	Time response of RL and PID in both Y and Z axes for blinking agent tracking experiment 4	93

LIST OF ALGORITHMS

2.1 Pseudocode of RL based on Q-learning (Azar et al., 2021) 103

2.2 Deep Q-learning with Experience Replay 103

2.3 Pseudocode for DQN-Based Training Method 104

2.4 Pseudocode for Deep Deterministic Policy Gradient (DDPG). 104

2.5 Pseudocode for a single Advantage Actor-Critic Agent. 105

2.6 TD3 (Fujimoto *et al.*, 2018) 105

3.7 Pseudocode of training for TD3 based UAV tracking 106

3.8 Pseudocode Multi-stage Shaping Function 106

3.9 Episode Completion Status 107

4.10 Simulation Algorithm Using TD3 108

4.11 Pseudocode of agent selection algorithm 109

LIST OF SYMBOLS AND ABBREVIATIONS

A3C	-	Asynchronous Advantage Actor-Critic
ACKTR	-	Actor Critic Kronecker-factored trust region
AI	-	Artificial Intelligence
CMDP	-	Constrained Markov Decision Process
CRLB	-	Cramér–Rao Lower Bound
D3QN	-	Duelling Double Deep Q-Network
DCNNP	-	Deep Convolutional Neural Network Policy
DDPG	-	Deep Deterministic Policy Gradient
DDQN	-	Duelling Deep Q-Networks
DNN	-	Deep Neural Networks
DPG	-	Deterministic Policy Gradient
DQN	-	Deep Q-Networks
DRL	-	Deep Reinforcement Learning
EKF	-	Extended Kalman Filtering
GUI	-	Graphical User Interface
IoU	-	Intersection-over-Union
MARL	-	Multiagent Reinforcement Learning
MDP	-	Markov Decision Process
ML	-	Machine Learning
MN-DDPG	-	Mixture Noise DDPG
MPC	-	Model Predictive Control
PD	-	Proportional Differential
PID	-	Proportional-Integral-Derivative

PPO	-	Proximal Policy Optimization
RL	-	Reinforcement Learning
SLAM	-	Simultaneous localization and mapping
TD	-	Temporal Difference
TD3	-	Twin Delayed Deep Deterministic
TF-DQN	-	Target Following DQN
TRPO	-	Trust Region Policy Optimization
UAV	-	Unmanned Aerial Vehicle

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	ALGORITHM	102
B	LIST OF PUBLICATIONS	110

CHAPTER 1

INTRODUCTION

1.1 Background of the study

UAV applications are increasing day after day, and aerial vehicles are part of many recent technological applications. Some applications can be seen in shipping (Grippa, Behrens, Wall, & Bettstetter, 2019), surveillance (Abdallah, Ali, Mišić, & Mišić, 2019; Mishra, Garg, Narang, & Mishra, 2020b), battlefield (You, 2020), rescuing applications (Joshi, Pal, Zafar, Bharadwaj, & Biswas, 2019; Lygouras *et al.*, 2019), and inspection (Kocer, Tjahjowidodo, Pratama, & Seet, 2019; Y. Zhang, Yuan, Li, & Chen, 2017). The aerial vehicle is categorised into three parts: teleoperated (Aleotti *et al.*, 2017; Bareiss, Bourne, & Leang, 2017), semi-autonomous (Khadka, Fick, Afshar, Tavakoli, & Baqersad, 2020; D. Zhang & Khurshid, 2019), and full autonomous (Uryasheva, Kulbeda, Rodichenko, & Tsetserukou, 2019). Enabling aerial vehicles applications require essential autonomous features regarding the degree of autonomy in the system.

Well-developed features of autonomous UAV control include, for instance, stability enhancement and waypoint flight, autonomous tracking, and autonomous landing. However, new developments in the design of UAVs and the emergence of new application areas demand robust and adaptive control techniques for different flight conditions, aggressive manoeuvring flight, robust disturbance rejection, obstacle avoidance, fault tolerance, formation flying, and the use of new sensing and perception paradigms, such as computer vision. Even when the vehicle performs tasks autonomously, the efficiency and reliability of the communication link to the ground station or other aerial vehicles are important. This is because the autonomous

UAV may need to send information about itself or its environment to the ground station or other vehicles, or it may need to receive updated mission parameters from the ground station or information from other vehicles. To achieve all the ambitious requirements that autonomous operation brings, systematic and innovative methods for planning, navigation, decision making, control, sensing and communications are needed (Becerra, 2019).

In the non-linear and dynamic type of controls, building a mathematical function of the plant is needed to assure a stable controller. The stability of the controller is analysed based on complicated mathematical methods and techniques. In many real-world applications, the accuracy of the mathematical model of the plant is questioned. Furthermore, engineers perform mathematical approximations to simplify the model development. These approximations are based on some assumptions that limit the generalizability of the controller, which leads to issues in the application and reliability. To avoid such approximations and non-valid assumptions, the concept of free model control is used. However, instead of using it based on repeated trial and error for tuning a simplified controller, it can be used to develop an accurate controller that embeds sufficient gained knowledge from the plant (Fliess, 2009).

Reinforcement learning (RL) is model-free control based on artificial intelligence (AI). It has been proven to be an effective and practical control approach in non-linear and highly dynamic systems, especially when accurate modelling is difficult. Integrating RL with a deep neural network (DNN) to analyse scenes from video and make a decision based on extensive training has found its way as valuable AI products in the automotive industry and driverless cars (Sallab, Abdou, Perot, & Yogamani, 2017) and also in aerial vehicles control (Kersandt, 2018). The reason behind this is the ability to train the RL model based on an extensive number of driving scenarios and then use the learned knowledge for operation. Hence, RL is considered one type of model-free control as it does not need to build a model to control.

1.2 Motivation

Adding the feature of autonomous tracking of a moving target is considered vital for drones. Older variants of drones are controlled manually using a remote control with the human controller nearby (less than one mile). In the near future, it is expected that the drone will be autonomous or at least contain a set of autonomous features that assists human over the manual control process. In this study, the researcher is interested in tracking applications where the UAV is needed to react to a moving target in the environment and keep its field of view. On the other side, using reinforcement learning to enable smart problem solving of UAVs has emerged significantly in the literature. The dynamical aspect of RL is useful for enabling the capability to respond to the dynamical changes in the target and learn them immediately. Such potential creates good motivation for exploiting RL capability in moving target tracking.

1.3 Research Problem

The problem of target tracking using UAVs is an active research topic, and it is still suffering from various challenges. Firstly, the dynamic aspect of the problem is found in the target mobility or manoeuvers. More specifically, the object moves in a non-expected direction, and dynamic variables such as velocity and acceleration make the UAV tracking subject to confusion and loss of the target. Other sources of dynamics exist in the motors' responses, the battery level change, and its effect on the generated forces. Secondly, the non-linearity aspect of the problem is found in the kinematic model of the UAV and the trajectory conducted by the target as well. Third, the requirements of the problem in terms of the needed accuracy of tracking and time of settling are subject to change from one application to another, e.g., in some applications such as surveillance, high accuracy is required with the fastest time while in target tracking for selfie capturing the target is not subject to escape from the view of the camera as the surveillance application. However, in the selfie capturing application, the manoeuver of the UAV is important for visual impact.

Reviewing the previous literature, we found that reinforcement learning gains more interest and proves higher performance than classical and modern control. The reason is its powerfulness in approximating high complex control surface based on only data generated from the plant and enabling dynamics and change handling. This breakthrough of RL-based models was witnessed after proving the power of deep Q learning when integrated with the Q learning approach for the ATARI game (Leibfried, Kushman, & Hofmann, 2016). However, applying RL to UAV-based tracking is not straightforward as it faces challenges in identifying the best RL elements, namely, state, action and reward, and finding the best knowledge embedding structure.

Several problems can be stated in the existing works. First, more recent variants of using DDPG were proposed based on target networks to reduce the accumulation of errors. Furthermore, to address the issue of coupling between value and policy, the work of (Fujimoto, Hoof, & Meger, 2018) has proposed delaying policy updates until the value estimate has converged with a regularization strategy. This work is named Twin Delayed Deep Deterministic policy gradient (TD3), and it has not been applied to the tracking problem. Second, non-of the previous approaches have enabled dynamic aware training for DQN. In the work of (Vankadari, Das, Shinde, & Kumar, 2018), velocity was included in the reward. However, their reward formulation does not provide any incorporation to the acceleration. In addition, the presentation of the rewarding function behaves linearly in terms of the reward presented to the agent based on its achievement which makes the learning slow and not efficient. Third, for all the aforementioned RL based object tracking, we observe that there was no phase for validation before selecting the optimal agent. This leads to sub-optimality because of risk of overfitting. Overall, tackling the problem of target tracking using RL is more effective than alternative approaches.

Overall, three matters are needed to be handled. Firstly, the effective formulation of the reward function to enable the usage of higher dynamic feedback or rewarding. Secondly, moving from a linear way of providing rewarding to agents based on achievement is not stable. Thirdly, the need for a validation phase for the avoidance of sub-optimality due to over-fitting.

1.4 Objectives

The ultimate goal of this study is to develop an RL-based target tracking for a small quadcopter UAV system. The performance and effectiveness of the proposed RL tracking will be evaluated in a series of simulated and real-time tracking scenarios. This goal is accomplished through the following objectives:

1. To develop a novel rewarding function to enable the usage of higher dynamic feedback for better tracking performance.
2. To propose a multi-objective agent selection algorithm that is superior in terms of holding time and tracking error.
3. To evaluate the tracking performance of the proposed RL tracking system against different target tracking scenarios using standard evaluation metrics.

1.5 Scope of study

This research tackles UAV-based target tracking, which is an essential UAV development problem with a wide range of industrial applications. In our study, we assume that the target has the freedom to change its location in the environment within the plane of mobility. Without the loss of generality, we assume that the target is moving in the YZ plane, and the UAV has to maintain the target in the center of the camera. The target is assumed to stay stationary, jump in the location, and move in a certain trajectory with constant speed, such as a circular or rectangular trajectory. Different speeds were used for moving the target. The work will be evaluated based on both simulation and real-world experiments. The latter is done in an indoor environment. To confirm the superiority of certain approaches, statistical-based evaluation where an experiment will be repeated for several rounds, and boxplot visualization is used to express the performance were adopted.

1.6 Contributions

This thesis includes several contributions, we state them as follows

1- To the best of our knowledge, this study is the first to apply TD3 to the problem of UAV-based target tracking. As stated in the problem statement, we adopt this architecture because it solves several issues that exist in DDPG that was applied in target tracking.

2- This thesis proposes a novel reward formulation for UAV-based target tracking that enables a wise combination of the various dynamical variables in the reward functions. The novel rewarding function incorporates two exponential functions to limit the effect of velocity and acceleration in order to prevent the deformation in the policy function approximation.

3- This thesis proposes an enhancement of the rewarding function by including piecewise decomposition in order to enable a more stable learning behavior of the policy and to move out from the linear reward vs. achievement formula.

4- This study proposes a smart agent selection algorithm that takes into consideration two objectives for searching and it selects the best agent out of the entire agents that are generated from the episodes of training.

5- This study enables better explanation fidelity by recreating the dynamic changes in the policy surface of the various models that are developed or implemented as benchmarks.

6- A thorough evaluation is conducted in both simulation and real-world experiments to evaluate the developed models and compare them with the benchmarks using standard evaluation metrics

1.7 Outline of the thesis

The remaining of the study is given as follows.

Chapter 2 presents the related background for carrying. Furthermore, a literature survey of the most recent RL models developed for UAV tracking is also provided.

In Chapter 3, all the stated objectives are defined and accomplished based on the methodology given in this chapter. First, a novel framework for RL-based object tracking using TD3 was built. Second, a novel rewarding function to the framework to enable the usage of higher dynamic feedback or rewarding more effectively. Third, we extend the rewarding to achievement-aware concept to enable more stability in the learning. Lastly, a multi-objective agent selection algorithm was proposed to avoid the over-fitting effect and to assure application preference in the selection.

Chapter 4 presents the simulation setup, algorithm, and agent selection algorithm developed to enable validation before selection. The validation will be based on three experiments for the agent by passing on the agents using an incremental way. Lastly, the agent selection algorithm is provided.

Chapter 5 presents the evaluation of the best achieving model in real-world experiments to confirm the generalization of its superiority. Next, we present the experimental setup. Then, the experimental results and analysis are provided.

Chapter 6 summarises the study, limitations, and future works.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

As presented in Chapter 1, a mathematical function of the plant must be built to ensure a reliable controller in non-linear and dynamic controls for UAV tracking. The controller's stability can be assessed using complex mathematical approaches and techniques. However, one can still question the accuracy of a mathematical model used in many real-world applications. Engineers also use mathematical approximations to make model development easier. Estimates are usually made based on assumptions limiting the controller's generalizability, resulting in difficulties in application and reliability. The concept of free model control is utilised to avoid such approximations and invalid assumptions. Instead of utilising it to tune a simplified controller through repeated trial and error, it can be used to construct an accurate controller that incorporates enough plant knowledge. Reinforcement learning (RL) is a sort of artificial intelligence-based model-free control (AI). It has proven to be a useful and effective control method in nonlinear and highly dynamic systems, especially when proper modelling is difficult. Furthermore, combining RL with a deep-neural network for video interpretation and decision-making based on lengthy training has proven to be useful AI in the automotive industry and driverless cars and aerial vehicle control. This chapter will provide related literature on the most recent RL models developed for UAV tracking.

2.2 Background

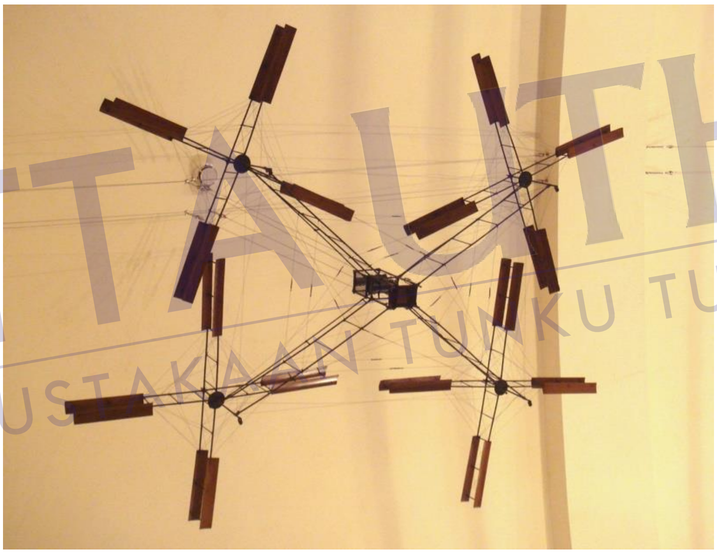
This section provides the background of UAVs in general and quadrotors used in our research. It consists of a historical overview of the quadrotor, their kinematic, dynamics, Gazebo simulation, reinforcement learning and their variants and models.

2.2.1 History of Quadrotor

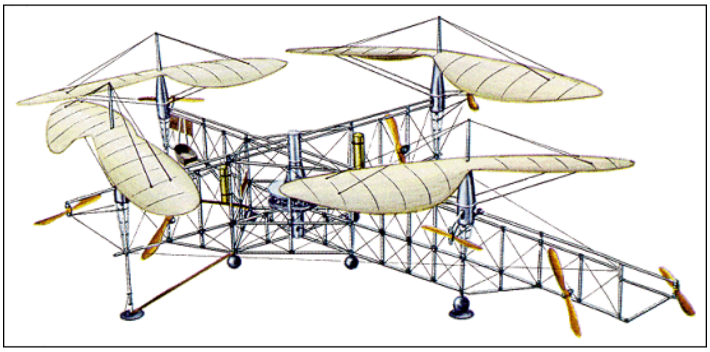
The first trial of the aerial vehicle was done by Abbas ibn Firnas between 800-900 AD. After a long time, the experiment of Abbas was recreated by Orville and Wilbur Wright, who performed the first controlled-powered human flight of a heavier-than-air aircraft in 1903. The Wright fixed-wing design (Search) was used for the majority of the early advancements. However, one of the biggest disadvantages of the fixed-wing design is the necessity for runways to land and take off. This limitation was overcome by developing rotary-wing aerial vehicles in various forms. Early attempts showed the Brèguet brothers introduced the first quadrotor-type design in 1907, named the Brèguet–Richet Gyroplane, as depicted in Figure 2.1 (a) ((Kim, Gadsden, & Wilkerson, 2019). This vehicle had a one-of-a-kind design. Each rotor had four biplane-type blades (two deep), providing a total of 32 independent lifting surfaces, and was coupled to an ICE through a belt and pulley transmission system. The four rotors were set up in two clockwise and counterclockwise revolving pairs, allowing torsion effects on the body frame to be controlled. This design approach is still employed in modern quadrotors. However, the Brèguet–Richet Gyroplane No. 1 did not have any manoeuvering control surfaces, making it unsuitable for use as a monitoring device. Étienne Oehmichen continued to experiment and develop quadrotors. The rotors on the Oehmichen No. 2 were two-bladed and positioned at the end of the frame, as illustrated in Figure 2.1(b). These blades could be twisted, which changed the blade angle of attack and gave the vehicle more control. The vehicle's yaw control was handled by two propellers situated at the vehicle's nose. The Oehmichen No.2 is a hybrid design that combines a quadrotor and a helicopter. The quadrotor structure was further modified in 1922 by George de Bothezat and Ivan Jerome, who added six-bladed rotors and two extra propellers as provided in

Figure 2.1(c). For thrust and yaw control, two tiny propellers were fitted. This vehicle was also subjected to collective pitch control (De)).

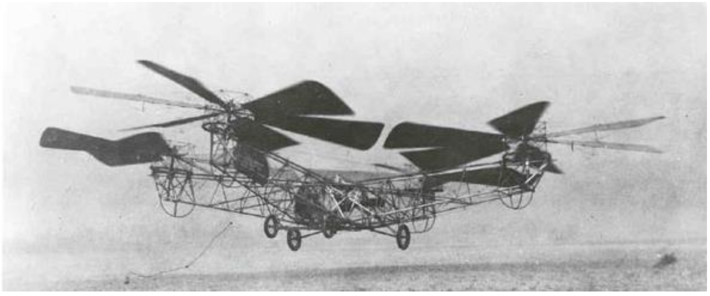
Quadrotors sparked initial attention and study, but the concept fell out of favour during the next two decades due to weight and technological issues. As depicted in Figure 2.1(d), the Convert wings Model A quadrotor was designed and debuted in 1956. The design used two engines to power four rotors for the lift. This design successfully demonstrated forward flight capability and proved it in flight tests. This model, however, was quickly abandoned due to a lack of orders. The Curtiss-Wright VZ-7 was designed for the United States Air Force in 1958 (Figure 2.1(e)). This quadrotor model has four rotors with independent speed controllers for each. The Curtiss-Wright VZ-7 quadrotor design can be seen as a forerunner to modern quadrotor designs.



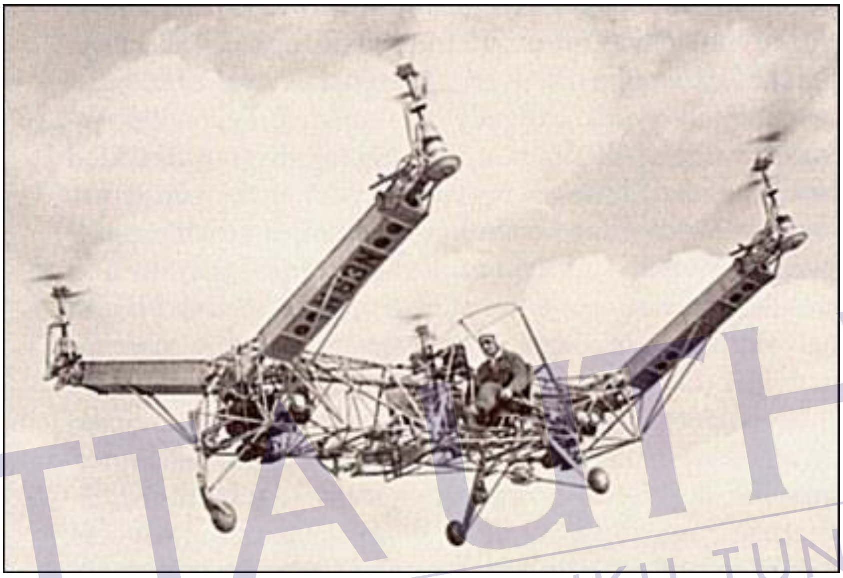
(a)



(b)



(c)



(d)

(e)

Figure 2.1: History of quadrotors. (a) Brèguet–Richet Gyroplane No. 1; (b) Oehmichen No. 2; (c) Bothezat helicopter; (d) Convertawings Model A; (e) Curtiss-Wright VZ-7.

Quadrotor designs are increasingly being used to construct small-scale UAVs. Several institutions and businesses have established research centres to improve quadrotor designs and uses. UAV Market is expected to reach USD58.4 billion by 2026 (marketsandmarkets, 2009). Recently, numerous companies have been offering their platform of Quadrotor with different capabilities, features, and costs.

2.2.2 DOF Airframe Dynamics

The dynamics of an aircraft are modelled using dominant methodologies such as Euler-Lagrange formalism and Newton-Euler formalism. Despite the compact formulation and generalisation demonstrated by Euler-Lagrange formalism, it has been highlighted that the Newton-Euler approach is simple to understand and accept physically. However, when it comes to describing dynamics, two ways are consistent. The Lagrange equation is an expression form of the second Newton Law after applying a speed transform matrix (X. Zhang, Li, Wang, & Lu, 2014).

A) Euler-Lagrange Formalism:

$\mathrm{q} = (x, y, z, \psi, \theta, \phi) \in R^6$ $(x, y, z) = \xi \in R^3$ denotes the position of the mass centre of the quadrotor relative to the inertial frame

$(\psi, \theta, \phi) = \eta \in R^3$ are the three Euler angles (resp., yaw ψ, pitch θ, and roll ϕ), under these conditions:

$$(-\pi \leqslant \psi \leqslant \pi) \ \& \left(-\frac{\pi}{2} \leqslant \theta \leqslant \frac{\pi}{2}\right) \ \& \left(-\frac{\pi}{2} \leqslant \phi \leqslant \frac{\pi}{2}\right)$$

The translation kinetic energy (Ttrans) can be calculated using Equation 2.1.

$$T_{\text{trans}} \triangleq \frac{m}{2} \dot{\xi}^T \dot{\xi} \tag{2.1}$$

$\dot{\xi}$ denotes the velocity

Meanwhile, the rotational kinetic energy (Trot) can be calculated using Equations 2.2 and 2.3.

$$T_{\text{rot}} \triangleq \frac{1}{2} \dot{\eta}^T J \dot{\eta} \tag{2.2}$$

$$J = \mathrm{W}^T I \mathrm{W} \tag{2.3}$$

Where:

m denotes the mass of the UAV

J denotes the moment of inertia

$\dot{\eta}$ denotes the angular rate

I, denotes the inertia

W denotes the transformation matrix to the body frame expressed as Equation 2.4.

$$\mathrm{W} = \begin{bmatrix} -\sin(\theta) & 0 & 1 \\ \cos(\theta)\sin(\psi) & \cos(\psi) & 0 \\ \cos(\theta)\cos(\psi) & -\sin(\psi) & 0 \end{bmatrix} \tag{2.4}$$

The Lagrangian (L) of the rotorcraft is given in the Equation 2.5.

$$L = T - V \tag{2.5}$$

Where:

V denotes the potential energy which is calculated based on $V = mg\mathbf{z}_E$, considering that the gravitational potential is the sole potential, g denotes the gradational force as given in equations (2.6) and (2.7).

$$L = T_{\text{trans}} + T_{\text{rot}} - V \tag{2.6}$$

$$\dot{\xi}^T\dot{\xi} + \frac{1}{2}\dot{\eta}^T J\dot{\eta} - mg\mathrm{z}_E = \frac{m}{2} \tag{2.7}$$

The full rotorcraft dynamics model is derived from the Euler-Lagrange equations under generalised external forces, as given in Equation 2.8.

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \left(F_\xi, \tau\right) \tag{2 8}$$

Where:

$F_\xi = R\hat{F}$ denotes the translational force applied to the quadrotor due to the throttle control input.

R denotes the rotational matrix $R(\psi, \theta, \phi) \in \mathrm{SO}(3)$

As a result, the group SO(3) can be recognised as the group of these matrices when matrix multiplication is performed. These matrices are called "special orthogonal matrices", which explains the acronym SO(3). It represents the orientation of the rotorcraft relative to a fixed inertial frame.

$\tau \in R^3$ represents the pitch, roll, and yaw moments.

The Euler-Lagrange equation is divided into two sections because the Lagrangian contains no cross-terms in the kinetic energy. Hence, it is written as Equations 2.9 and 2.10.

$$m\dot{\xi} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = F_{\xi} \tag{2.9}$$

$$J\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau \tag{2.10}$$

Where:

$C(\eta, \dot{\eta})$ denotes Coriolis terms and contains the gyroscopic and centrifugal terms in Equation 2.11.

$$\dot{J}\dot{\eta} - \frac{1}{2}\frac{\partial}{\partial \eta}(\dot{\eta}^T J \dot{\eta}) = C(\eta, \dot{\eta})\dot{\eta} \tag{2.11}$$

B) Newton-Euler Formalism:

Typically, it is necessary to define two frames of reference, each with its defined right-handed coordinate system, as shown in Figure 2.2.

$\vec{V} = [u \quad v \quad w]^T$ body linear velocity vector, $\vec{\Omega} = [p \quad q \quad r]^T$ angular rate vector, $E = (X_E, Y_E, Z_E)$ inertial frame, $B = (X_B, Y_B, Z_B)$ body frame

Earth-fixed inertial (also known as navigation) coordinate system. The attitude of the quadrotor, expressed in terms of the Euler angles ϕ (roll), θ (pitch), and ψ (yaw), is evaluated via sequent rotations around each one of the inertial axes. Herein, a reference frame by O_{NED} (North-East- Down) denotes an inertial reference frame and O_B a body-fixed reference frame.

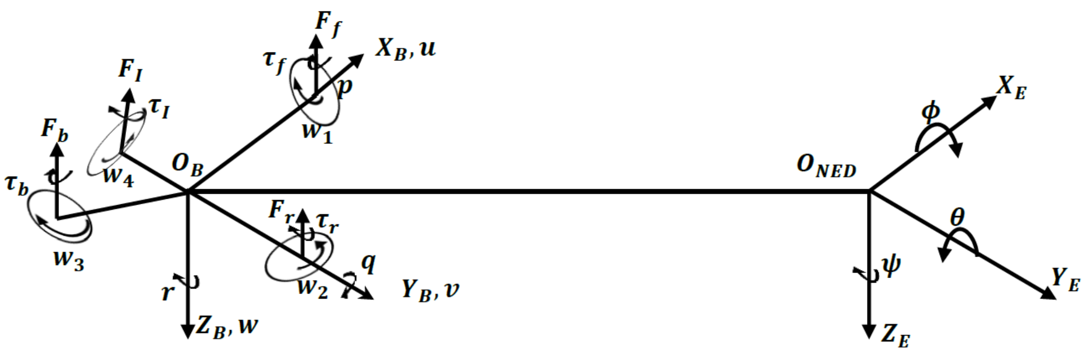


Figure 2.2: Reference frame and the body frame with their relative geometric relation

Generally, a quadrotor is considered a rigid body in a three-dimensional space. The motion equations of a quadrotor subject to external force $F \in R^3$ and torque $\tau \in R^3$are given by the following Newton-Euler equations for the body coordinate frame B which is given in Equation 2.12.

$$\begin{bmatrix} mI_{3\times3} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \omega \times mV \\ \omega \times I\omega \end{bmatrix} = \begin{bmatrix} F \\ \tau \end{bmatrix} \quad (2.12)$$

The rotorcraft orientation in space is presented by a rotation R from B to E as given in Equation 2.13.

$$\begin{gathered} m\ddot{\xi} = RF - mgZ_E \\ I(\eta)\ddot{\eta} + C(\dot{\eta}, \eta) = \tau \end{gathered} \quad (2.13)$$

Where:

$$R = \begin{pmatrix} c_\psi c_\theta & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ s_\theta s_\psi & s_\phi s_\theta s_\psi + c_\theta c_\psi & c_\phi s_\theta s_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}$$

$$I(\eta) = J$$

$$C(\dot{\eta}, \eta) = I\dot{\eta} + W\dot{\eta} \times I\dot{\eta}$$

2.2.3 UAVs application

There are various types of UAVs in today's applications. Four UAVs with different applications were used to explore the potential usage of UAVs in different real-world applications: cinematographic, traffic tracking, deployment on the battlefield, and the auto-landing application for UAVs and ground vehicles collaboration.

(i) Cinematographic using UAV

In cinematography, UAVs are deployed to record videos of the scene to replace human beings with more impressive scenes capturing. In order to enable this functionality, target tracking is an important feature as it guarantees to maintain the target in the camera's view. Commercial UAVs employed in media production are mostly manually controlled, with only a few rudimentary functionalities performed autonomously (Mademlis *et al.*, 2019). In state-of-the-art drones such as the popular DJI Phantom 4 or Skydio R1, such functionalities are obstacle avoidance, landing, physical target following or target orbiting (for low-speed, manually pre-selected targets), as well as automatic central composition framing, i.e., continuously rotating the camera to always keep the pre-selected target properly framed at the centre. These basic functions and any future algorithms for more advanced, automated UAV flight and filming require many enabling technologies to be in place.

(ii) Traffic tracking using UAV

Compared with traditional traffic surveillance systems, detecting and tracking vehicles through the images captured by a UAV is still an active research topic due to various challenges (Mishra, Garg, Narang, & Mishra, 2020a). In the driver behaviour research models, such as car-following and lane-changing models, a car might perform non-expected behaviour. Missing car data and tracking errors could affect the accuracy of the model parameters settings. In addition, the camera of a UAV surveillance platform is changed frequently because the camera in a UAV may rotate, shift and roll during video recording. Sudden shakes might also happen due to wind fluctuations, which can cause negative effects on vehicle tracking. This motivates the researcher to build a robust tracking model for the UAV of a moving target.

(iii) UAV incorporation in the battlefield is becoming essential in the military recent industrial achievements (Johnson, 2020). However, this motivates researchers to solve various challenges in UAVs deployment in a military operation. One important challenge is the capability of a UAV to cooperate with a friend military vehicle which requires tracking functionality. On the other side, the UAV might need to track the enemy vehicles, which adds another challenge due to the uncertainty in the manoeuver of the target and the non-expected surrounding conditions in the environment in the military zones.

(iv) Drone Auto-Landing

One essential autonomous feature is auto-landing on a moving target. In various applications such as ground-aerial vehicle collaboration, aerial vehicles need to identify a certain ground vehicle for tracking and landing. This functionality must be autonomous due to the challenging aspect of teleoperation landing on a moving target. In addition, there is a risk of failure, which might cause damage to the aerial vehicle. Hence, it is essential to include the functionality of autonomous landing in all categories of operation of aerial vehicles, even in the teleoperation category. This has encouraged using this functionality as part of a robotic competition (Beul, Houben, Nieuwenhuisen,

& Behnke, 2017) (Bähnemann *et al.*, 2019). Another example of a flight landing application is the landing on a moving ship deck (Lin, Garratt, & Lambert, 2017) which also requires not only identifying the landing spot but assuring a safe and precise landing while the ship is moving. The functionality of autonomous landing can be needed in the applications when a swarm of UAVs is deployed for inspection, and a charging pod is deployed to assure a longer operation lifetime. This is attained by enabling the autonomous landing of the UAV on the charging pod when the battery is dead. Figure 2.3 provides 1 example of one UAV application for inspection on an electricity tower.

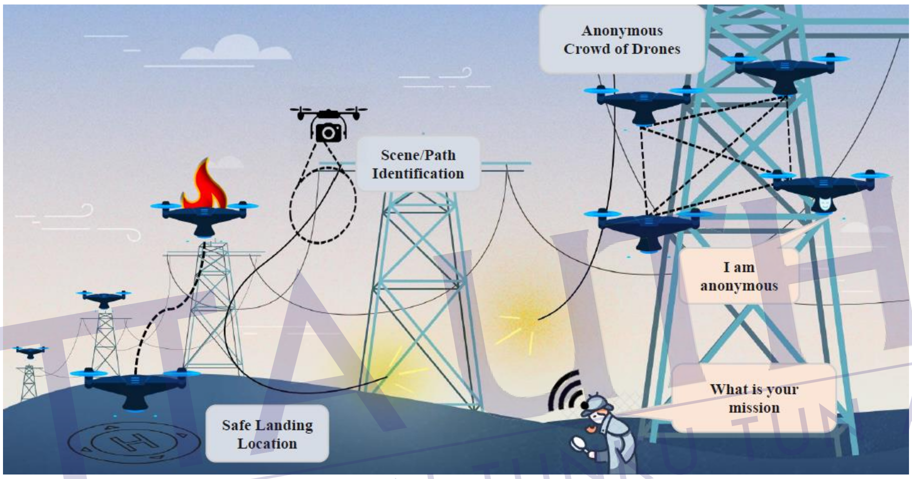


Figure 2.3 One example of UAVs application for inspection on electricity tower and safe landing on a charging pod
(Source: Boukoberine *et al.*, 2019)

2.2.4 RL based UAVs applications

Reinforcement learning-based UAV approaches are widely emerging and prove their effectiveness in different challenges. We take three examples from the recent literature to show the remarkable capability of RL in solving challenging UAV tasks. The first is UAVs racing, the second is chasing UAVs, and the third is fleet decision-making problems.

i. **UAVs Racing:** In many robotic tasks, such as drone racing, the goal is to travel through a set of waypoints as fast as possible. A key challenge for this task is planning the minimum-time trajectory, typically solved by assuming perfect knowledge of the waypoints to pass in advance. The emergence of UAVs racing has increased, enabling UAVs to their limit capability based on RL models (Song, Steinweg, Kaufmann, & Scaramuzza, 2021). In addition, RL assists in learning from human experience (Shin, Kang, & Kim, 2019).

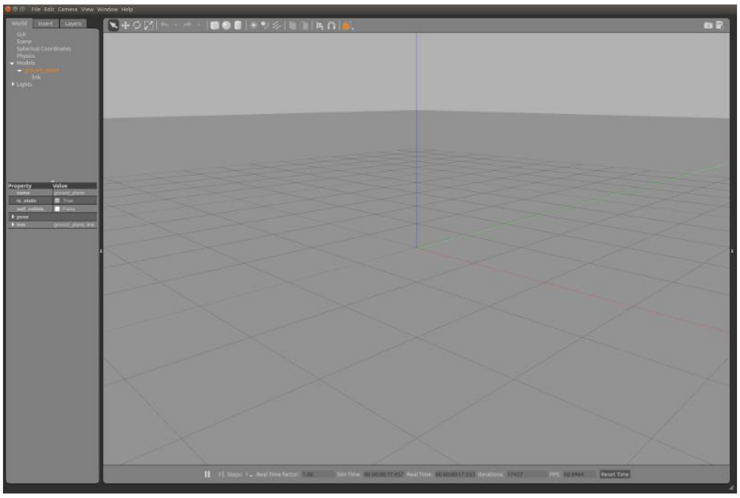
ii. **UAVs Chasing UAVs:** The use of multiple UAVs and UAV swarms is attracting more interest from the research community, leading to the exploration of topics such as UAV cooperation and multi-drone autonomous navigation. Some researchers have developed deep reinforcement learning to predict the actions to apply to the follower UAV to keep track of the target UAV (Akhloufi, Arola, & Bonnet, 2019).

iii. **UAV Fleet Decision-Making Problem:** In the UAV fleets' trajectories planning in cooperative patrolling and tracking missions, the UAV fleet is patrolling in an area split into several grids. d, the UAV or target location is simplified as the coordinate of the grid. Many UAVs are appointed to patrol in the requested area, in which one or more stationary gas stations can refuel the UAV. All UAVs are able to monitor their fuel and need to plan their refuelling automatically. In the patrol process, there might be some immobile or mobile targets presented in the patrol area. Because the horizons of UAVs are limited and are usually smaller than the patrol area, the UAV fleet may not be able to locate the target when it appears. However, the fleet is supposed to find the target collaboratively as quickly as possible. In addition, the fleet is expected to follow the target, which means keeping the target inside at least one UAV's horizon so that the administrators can identify the target and give further instructions. If the fleet loses the target, it should be able to search in the possible area where the target may hide. Such a problem has been addressed by researchers using reinforcement learning (T. Wang, Qin, Chen, Snoussi, & Choi, 2019).

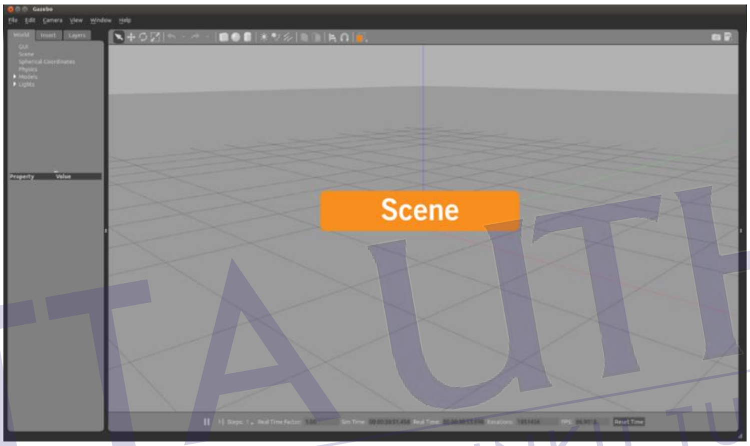
2.2.5 Gazebo Simulation

Every roboticist's toolbox should include robot simulation. A well-designed simulator enables rapid algorithms, robot design, regression testing, and AI system training using realistic scenarios (Gazebo, 2020). allows to precisely and quickly model robot populations in indoor and outdoor contexts (Mengacci *et al.*, 2021). A powerful physics engine, high-quality images, and user-friendly programmatic and graphical interfaces. The Gazebo is a three-dimensional dynamic simulator that can correctly and effectively model robot populations in indoor and outdoor situations. Similar to game engines, Gazebo provides a far higher level of fidelity physics modelling, a sensor suite, and user and programming interfaces. (Rivera, De Simone, & Guida, 2019)

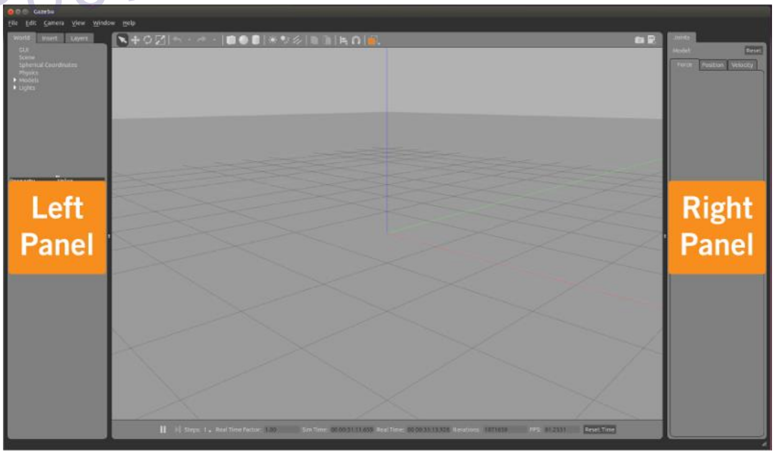
Testing robotics algorithms, building robots, and performing regression testing with realistic scenarios are all common uses of Gazebo. The following are some of the most important aspects of the Gazebo: Several physics engines, a large library of robot models and surroundings, a wide range of sensors, and simple programmatic and graphical interfaces are all available. The Gazebo interface is divided into several pieces. First, the scene is the Simulator's main component. The users interact with the world as the simulated items are animated. Second, the panel can be displayed, hidden, or resized by dragging the bar separating the right and left side panels from the Scene. When Gazebo is running, the left panel appears by default. In the panel, there is the world, insert, and layers. The World tab shows the current models in the scene and allows to see and change model parameters such as posture. The user may also change the camera view angle by extending the “GUI” option and altering the camera posture. The insert allows the user to add additional objects (models) to the simulation using the Insert tab. The user may need to expand the folder by clicking the arrow to see the model list. To add a model, click (and release) on the model to insert, then click again on the Scene. The Layers tab organizes and displays any visualization groups present in the simulation. One or more models can be found in a layer. When the user turns a layer on or off, the models in that layer will be visible or hidden. Figure 2.4 shows the main graphical components in the Gazebo interface: a) GUI; b) Scene; d) Panel.



(a)



(b)



(c)

Figure 2. 4: The main graphical components in Gazebo interface: (a) GUI; (b) Scene; (c) Panel

To identify files and establish communications between the server and clients, Gazebo uses a variety of environment variables. This eliminates the requirement for any variables to be set. Examples of the world files are GAZEBO_MODEL_PATH,GAZEBO_RESOURCE_PATH,GAZEBO_MASTER_URI, GAZEBO_PLUGIN_PATH, and GAZEBO_MODEL_DATABASE_URI. A model file uses the same SDF format as world files but should only contain a single <model> ... </model>. To locate files and build up connections between the server and clients, Gazebo uses a variety of environment variables. Default values have been compiled that will work in the majority of circumstances. This signifies that no variables need to be set. Gazebo's workhorse is the server that parses a command-line-supplied world description file and simulates the environment using physics and a sensor engine. The graphical client establishes a connection with a functioning gz server and renders the elements. Plugins make interacting with Gazebo straightforward, and plugins can be provided in an SDF file or loaded from the command line.

2.2.6 Reinforcement Learning

Machine learning (ML) is a technique in which computer software learns from its previous experiences to enhance its performance on a given job. ML algorithms are frequently categorized into three categories: supervised learning, unsupervised learning, and reinforcement learning (RL). Unsupervised learning includes techniques such as density estimation or clustering applied to unlabeled data, whereas supervised learning algorithms are based on inductive inference, where the model is often trained using labelled data to accomplish classification or regression. In the third type, RL, the expert does not tell RL agents how to act; rather, a reward function R evaluates an agent's performance. The agent picks an action for each condition encountered and receives an occasional reward from its surroundings based on the utility of its decision. The agent's purpose is to maximize the total rewards obtained throughout its existence. The rationality of action selection can be judged by calculating the expected cumulative return value after the action is executed in the current state. In other words, RL is a learning method that uses the ‘trial and error’ method to interact with the environment, which the Markov decision-making process can characterize. As a result, the mappings between state and action generated by the reinforcement learning method consider the action's long-term impact. Rewards of the environment can evaluate the agent, so it is feasible to build an RL-based tracking for moving targets (Hafiz, Parah, & Bhat, 2021).

The main concepts and definitions regarding RL are presented in the subsequent subsections. The agent can gradually enhance its long-term reward by utilizing knowledge gained about the predicted utility (i.e., the discounted sum of projected future rewards) of various state-action combinations. Managing the trade-off between exploration and exploitation is one of the most difficult aspects of reinforcement learning.

A reinforcement learning controller involves sets of elements, namely, state, action, transition function, and reward function. It also contains the learning rate and the discounting factors as parameters. Each of the elements is presented as follows:

a) State:

It refers to a representation of the needed information from the environment to operate the control. The state is application dependent, indicating that different applications imply different state representations which means the number of states depends on the application itself. Another aspect of the state is its finite nature. This means that the number of states is limited. Mathematically, this is given in Equation 2.14.

$$X = \{1,2, \dots i \dots n_X\} \tag{2.14}$$

Where:

n_X denotes the number of states

i denotes an index of the state

b) Action:

It refers to a representation of the needed actions that affect the environment to maintain the control goal or target. Considering that RL is a model free type of control, we have to add all candidate actions to the actions set. Afterwards, selecting an action is based on the maximal satisfaction of the goal of the control based on the learning embedded in the Q-matrix. $A = \{1,2 \dots j, \dots n_A\}$ the actions

Where:

n_A denotes the number of actions

j denotes an index of the action

c) Transition Function:

This function reflects the dynamic of the problem represented by the evolution of the states from one state to another based on the provided actions. Mathematically, it is represented as,

$T: X \times A \rightarrow X$ Transition Function

It means that any association of state and action will move the system from its current state to another state.

d) Reward Function:

This reflects the essential core of the RL's success in providing the needed functionality to reach the goal. The role of the rewarding function to score actions given provided state according to how much added values to the goal of control is accomplished when by the resulted states. This is represented mathematically by the equation,

$R: X \times A \rightarrow R$ Reward Function

This means that any association of state and action will be linked to a certain reward.

e) Discounting Factor:

It quantifies the future consideration of the selected action or how much future reward will contribute to the decision of selecting the current action. It takes a value between 0 and 1.

f) Learning Rate:

It quantifies how much new information override current knowledge. Typically, it takes a value between 0 and 1. A higher value means that the memory of knowledge is reduced, and lower values mean that older knowledge is kept, and we change it gradually as new information is received. A typical value for the learning rate is 0.1. The pseudocode of RL based on Q-learning is given in Algorithm 2.1 **APPENDIX A**. The training of RL is based on the Bellman equation, given in Equation 2.15. the accumulated reward Q which has the state s_t number of rows and the actions a_t number of columns. The state is the numeric representation of what the agent is observing at a particular point of the time in the environment and the action is the input of the agent that provides to the environment that calculated by applying a policy to the current state. In addition, reward r_t is a feedback signal from the environment reflecting how well the agent is performing the goals. Furthermore, the learning rate α is the quantifies how much new information override current knowledge while the discount factor γ is quantifying the future consideration of the selected action or how much future reward will contribute in the decision of selecting current action.

$$Q^{new}(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \overbrace{(\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}}}^{\text{new value (temporal difference target)}} \underbrace{\text{temporal difference}}_{\text{old value}} \quad (2.15)$$

The use of deep reinforcement learning (DRL) in UAV control was first presented to address specific issues in the sector. DRL aids in the task of UAV control by allowing it to work with model-free algorithms when the UAV model is difficult to find, to account for nonlinearities in the system, to actively learn how to achieve the target without being explicitly trained, and to work in environments where the UAV is unfamiliar

Some examples of the applications of deep learning models in the UAV domain are listed in Table 2.1. As shown, path planning, navigation, control, trajectory tracking, exploration and autopilot systems are considered the most common applications. In addition, the models of Duelling Double Deep Q-Network (D3QN), Deep Q-Network (DQN), TRPO, DDPG, PPO, ACKTR, and DCNNP were among the deep learning algorithms used in the UAV applications.

REFERENCES

Abdallah, A., Ali, M. Z., Mišić, J., & Mišić, V. B. (2019). Efficient security scheme for disaster surveillance UAV communication networks. *Information, 10*(2), 43.

Akhloufi, M., Arola, S., & Bonnet, A. (2019). Drones Chasing Drones: Reinforcement Learning and Deep Search Area Proposal,‖ *Drones*, vol. 3, no. 3. In: Jul.

Aleotti, J., Micconi, G., Caselli, S., Benassi, G., Zambelli, N., Bettelli, M., & Zappettini, A. (2017). Detection of nuclear sources by UAV teleoperation using a visuo-haptic augmented reality interface. *Sensors, 17*(10), 2234.

Almeshal, A. M., & Alenezi, M. R. (2018). A vision-based neural network controller for the autonomous landing of a quadrotor on moving targets. *Robotics, 7*(4), 71.

Araar, O., Aouf, N., & Vitanov, I. (2017a). Vision based autonomous landing of multirotor UAV on moving platform. *Journal of Intelligent Robotic Systems, 85*(2), 369-384.

Araar, O., Aouf, N., & Vitanov, I. (2017b). Vision based autonomous landing of multirotor UAV on moving platform. *Journal of Intelligent & Robotic Systems, 85*(2), 369-384.

Azar, A. T., Koubaa, A., Ali Mohamed, N., Ibrahim, H. A., Ibrahim, Z. F., Kazim, M., . . . Hameed, I. A. (2021). Drone deep reinforcement learning: A review. *Electronics, 10*(9), 999.

Bähnemann, R., Pantic, M., Popović, M., Schindler, D., Tranzatto, M., Kamel, M., Nieto, J. (2019). The ETH-MAV team in the MBZ international robotics challenge. *Journal of Field Robotics, 36*(1), 78-103.

Baranov, O. V., Smirnov, N. V., Smirnova, T. E., & Zholobov, Y. (2020). Design of a quadrocopter with PID-controlled fail-safe algorithm. *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl., 11*(2), 23-33.

Bareiss, D., Bourne, J. R., & Leang, K. K. (2017). On-board model-based automatic collision avoidance: application in remotely-piloted unmanned aerial vehicles. *Autonomous Robots, 41*(7), 1539-1554.

Becerra, V. M. (2019). *Autonomous control of unmanned aerial vehicles.* In: Multidisciplinary Digital Publishing Institute.

Beul, M., Houben, S., Nieuwenhuisen, M., & Behnke, S. (2017). *Fast autonomous landing on a moving target at MBZIRC.* Paper presented at the 2017 European Conference on Mobile Robots (ECMR).

Bhagat, S., & Sujit, P. (2020). *UAV Target Tracking in Urban Environments Using Deep Reinforcement Learning.* Paper presented at the 2020 International Conference on Unmanned Aircraft Systems (ICUAS).

Bouhamed, O., Ghazzai, H., Besbes, H., & Massoud, Y. (2020). Autonomous UAV navigation: A DDPG-based deep reinforcement learning approach. Paper presented at the 2020 *IEEE International Symposium on Circuits and Systems (ISCAS).*

Chen, Y.-J., Chang, D.-K., & Zhang, C. (2020). Autonomous tracking using a swarm of UAVs: A constrained multi-agent reinforcement learning approach. *IEEE transactions on vehicular technology, 69*(11), 13702-13717.

Choi, S., Kim, S., & Kim, H. J. (2017a). Inverse reinforcement learning control for trajectory tracking of a multirotor UAV. *International Journal of Control, Automation Systems, 15*(4), 1826-1834.

Choi, S., Kim, S., & Kim, H. J. (2017b). Inverse reinforcement learning control for trajectory tracking of a multirotor UAV. *International Journal of Control, Automation Systems, 15*(4), 1826-1834.

de Souza, J. P. C., Marcato, A. L. M., de Aguiar, E. P., Jucá, M. A., & Teixeira, A. M. (2019). Autonomous landing of UAV based on artificial neural network supervised by fuzzy logic. *Journal of Control, Automation Electrical Systems, 30*(4), 522-531.

De, B. G. Helicopter US1573228A. Retrieved from https://www.google.com/patents/US1573228

Du, W., Guo, T., Chen, J., Li, B., Zhu, G., & Cao, X. (2021). Cooperative pursuit of unauthorized UAVs in urban airspace via Multi-agent reinforcement learning. *Transportation Research Part C: Emerging Technologies, 128*, 103122.

Feng, Y., Zhang, C., Baek, S., Rawashdeh, S., & Mohammadi, A. (2018). Autonomous landing of a UAV on a moving platform using model predictive control. *drones, 2*(4), 34.

Fliess, M. (2009). Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control? *IFAC Proceedings Volumes, 42*(10), 1531-1550.

Fujimoto, S., Hoof, H., & Meger, D. (2018). *Addressing function approximation error in actor-critic methods.* Paper presented at the International Conference on Machine Learning.

Gazbo (2020). *About Gazebo.* Retrieved September 20, 2020, from https://gazebosim.org/about

Grippa, P., Behrens, D. A., Wall, F., & Bettstetter, C. (2019). Drone delivery systems: Job assignment and dimensioning. *Autonomous Robots, 43*(2), 261-274.

Guo, Y., Qin, H., Xu, B., Han, Y., Fan, Q.-Y., & Zhang, P. (2019). Composite learning adaptive sliding mode control for AUV target tracking. *351*, 180-186.

Hafiz, A., Parah, S., & Bhat, R. (2021). Reinforcement learning applied to machine vision: state of the art. *International Journal of Multimedia Information Retrieval*, 1-12.

He, S., Wang, H., & Zhang, S. (2019). Vision based autonomous landing of the quadrotor using fuzzy logic control. *Paper presented at the 2019 Chinese Control And Decision Conference (CCDC).*

Johnson, J. (2020). Artificial intelligence, drone swarming and escalation risks in future warfare. *The RUSI Journal, 165*(2), 26-36.

Joshi, G., Pal, B., Zafar, I., Bharadwaj, S., & Biswas, S. (2019). *Developing Intelligent Fire Alarm System and Need of UAV.* Paper presented at the International Conference on Unmanned Aerial System in Geomatics.

Kersandt, K. (2018). *Deep reinforcement learning as control method for autonomous uavs.* Universitat Politècnica de Catalunya, Khadka, A., Fick, B., Afshar, A., Tavakoli, M., & Baqersad, J. (2020). Non-contact vibration monitoring of rotating wind turbines using a semi-autonomous UAV. *Mechanical Systems and Signal Processing, 138*, 106446.

Kim, J., Gadsden, S. A., & Wilkerson, S. A. (2019). A comprehensive survey of control strategies for autonomous quadrotors. *Canadian Journal of Electrical Computer Engineering, 43*(1), 3-16.

Kocer, B. B., Tjahjowidodo, T., Pratama, M., & Seet, G. G. L. (2019). Inspection-while-flying: An autonomous contact-based nondestructive test using UAV-tools. *Automation in Construction, 106*, 102895.

Koch, W., Mancuso, R., West, R., & Bestavros, A. (2019). Reinforcement learning for UAV attitude control. *ACM Transactions on Cyber-Physical Systems, 3*(2), 1-21.

Lee, S., Shim, T., Kim, S., Park, J., Hong, K., & Bang, H. (2018). Vision-based autonomous landing of a multi-copter unmanned aerial vehicle using reinforcement learning. *Paper presented at the 2018 International Conference on Unmanned Aircraft Systems (ICUAS).*

Leibfried, F., Kushman, N., & Hofmann, K. (2016). A deep learning approach for joint video frame and reward prediction in atari games.

Li, B., Gan, Z., Chen, D., & Sergey Aleksandrovich, D. (2020). UAV Maneuvering Target Tracking in Uncertain Environments Based on Deep Reinforcement Learning and Meta-Learning. *Remote Sensing, 12*(22), 3789.

Li, B., & Wu, Y. (2020). Path planning for UAV ground target tracking via deep reinforcement learning. *IEEE Access, 8*, 29064-29074.

Li, B., Yang, Z.-p., Chen, D.-q., Liang, S.-y., & Ma, H. (2021). Maneuvering target tracking of UAV based on MN-DDPG and transfer learning. *Defence Technology, 17*(2), 457-466.

Lin, S., Garratt, M. A., & Lambert, A. J. (2017). Monocular vision-based real-time target recognition and tracking for autonomously landing an UAV in a cluttered shipboard environment. *Autonomous Robots, 41*(4), 881-901.

Lygouras, E., Santavas, N., Taitzoglou, A., Tarchanidis, K., Mitropoulos, A., & Gasteratos, A. (2019). Unsupervised human detection with an embedded vision system on a fully autonomous uav for search and rescue operations. *Sensors, 19*(16), 3542.

Maciel-Pearson, B. G., Marchegiani, L., Akcay, S., Atapour-Abarghouei, A., Garforth, J., & Breckon, T. P. (2019). Online deep reinforcement learning for autonomous UAV navigation and exploration of outdoor environments. *arXiv preprint arXiv:.05684*.

Mademlis, I., Nikolaidis, N., Tefas, A., Pitas, I., Wagner, T., & Messina, A. (2019). Autonomous UAV cinematography: A tutorial and a formalized shot-type taxonomy. *ACM Computing Surveys, 52*(5), 1-33.

marketsandmarkets. (2009). *Nanophotonics*- Advanced Technologies and Global Market (2009 - 2014).

Mengacci, R., Zambella, G., Grioli, G., Caporale, D., Catalano, M. G., & Bicchi, A. (2021). An Open-Source ROS-Gazebo Toolbox for Simulating Robots With Compliant Actuators. *Frontiers in Robotics AI*, 246.

Mishra, B., Garg, D., Narang, P., & Mishra, V. (2020a). Drone-surveillance for search and rescue in natural disaster. *Computer Communications, 156*, 1-10.

Mishra, B., Garg, D., Narang, P., & Mishra, V. (2020b). Drone-surveillance for search and rescue in natural disaster. *Computer Communications*.

Moon, J., Papaioannou, S., Laoudias, C., Kolios, P., & Kim, S. (2021). Deep Reinforcement Learning Multi-UAV Trajectory Control for Target Tracking. *IEEE Internet of Things Journal*.

Olivares-Mendez, M. A., Kannan, S., & Voos, H. (2015). *Vision based fuzzy control autonomous landing with UAVs: From V-REP to real experiments*. Paper presented at the 2015 23rd Mediterranean Conference on Control and Automation (MED).

Peake, A., McCalmon, J., Zhang, Y., Myers, D., Alqahtani, S., & Pauca, P. (2021). *Deep Reinforcement Learning for Adaptive Exploration of Unknown Environments*. arXiv preprint arXiv:.01606.

Pham, H. X., La, H. M., Feil-Seifer, D., & Nguyen, L. V. (2018). *Autonomous uav navigation using reinforcement learning*. arXiv preprint arXiv:.05086.

Pi, C.-H., Hu, K.-C., Cheng, S., & Wu, I.-C. (2020). Low-level autonomous control and tracking of quadrotor using reinforcement learning. *Control Engineering Practice, 95*, 104222.

Pina Navarro, M. (2021). Control mediante aprendizaje por refuerzo del robot Pepper.

Polvara, R., Patacchiola, M., Hanheide, M., & Neumann, G. (2020). Sim-to-Real quadrotor landing via sequential deep Q-Networks and domain randomization. *Robotics, 9*(1), 8.

Polvara, R., Patacchiola, M., Sharma, S., Wan, J., Manning, A., Sutton, R., & Cangelosi, A. (2018). *Toward end-to-end control for UAV autonomous landing via deep reinforcement learning.* Paper presented at the 2018 International conference on unmanned aircraft systems (ICUAS).

Polvara, R., Sharma, S., Wan, J., Manning, A., & Sutton, R. (2018). Vision-based autonomous landing of a quadrotor on the perturbed deck of an unmanned surface vehicle. *drones, 2*(2), 15.

Polvara, R., Sharma, S., Wan, J., Manning, A., & Sutton, R. (2019). Autonomous Vehicular Landings on the Deck of an Unmanned Surface Vehicle using Deep Reinforcement Learning.

Qing, Z., Zhu, M., & Wu, Z. (2018). Adaptive neural network control for a quadrotor landing on a moving vehicle. *Paper presented at the 2018 Chinese Control And Decision Conference (CCDC).*

Rivera, Z. B., De Simone, M. C., & Guida, D. (2019). Unmanned ground vehicle modelling in Gazebo/ROS-based environments. *Machines, 7*(2), 42.

Rodriguez-Ramos, A., Sampedro, C., Bavle, H., De La Puente, P., & Campoy, P. (2019a). A deep reinforcement learning strategy for UAV autonomous landing on a moving platform. *Journal of Intelligent & Robotic Systems, 93*(1-2), 351-366.

Rodriguez-Ramos, A., Sampedro, C., Bavle, H., De La Puente, P., & Campoy, P. (2019b). A deep reinforcement learning strategy for UAV autonomous landing on a moving platform. *Journal of Intelligent Robotic Systems, 93*(1-2), 351-366.

Ryze (2021). *TELLOSPECS.* Retrieved March 15, 2021, from https://www.ryzerobotics.com/tello/specs

Sallab, A. E., Abdou, M., Perot, E., & Yogamani, S. (2017). Deep reinforcement learning framework for autonomous driving. *Electronic Imaging, 2017*(19), 70-76.

Search, G. Introduction to Flight. Retrieved from https://g.co/kgs/5MdA44

Shim, T., & Bang, H. (2018). *Autonomous landing of UAV using vision based approach and PID controller based outer loop.* Paper presented at the 2018 18th International Conference on Control, Automation and Systems (ICCAS).

Shin, S.-Y., Kang, Y.-W., & Kim, Y.-G. (2019). Obstacle avoidance drone by deep reinforcement learning and its racing with human pilot. *Applied Sciences, 9*(24), 5571.

Si, W., She, H., & Wang, Z. (2017). *Fuzzy PID controller for UAV tracking moving target.* Paper presented at the 2017 29th Chinese Control And Decision Conference (CCDC).

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014). Deterministic Policy Gradient Algorithms. *Paper presented at the Proceedings of the 31st International Conference on Machine Learning, Proceedings of Machine Learning Research.* https://proceedings.mlr.press/v32/silver14.html

Song, Y., Steinweg, M., Kaufmann, E., & Scaramuzza, D. (2021). Autonomous Drone Racing with Deep Reinforcement Learning. *arXiv preprint arXiv:.08624*.

Srivastava, R., Lima, R., Das, K., & Maity, A. (2019). *Least square policy iteration for ibvs based dynamic target tracking.* Paper presented at the 2019 International Conference on Unmanned Aircraft Systems (ICUAS).

Sudevan, V., Shukla, A., & Karki, H. (2017). *Vision based autonomous landing of an unmanned aerial vehicle on a stationary target.* Paper presented at the 2017 17th International Conference on Control, Automation and Systems (ICCAS).

Sutton, R. S., & Barto, A. G. (1998). *Introduction to reinforcement learning* (Vol. 135): MIT press Cambridge.

Talha, M., Asghar, F., Rohan, A., Rabah, M., & Kim, S. H. (2019). Fuzzy logic-based robust and autonomous safe landing for UAV quadcopter. *Arabian Journal for Science Engineering, 44*(3), 2627-2639.

Uryasheva, A., Kulbeda, M., Rodichenko, N., & Tsetserukou, D. (2019). DroneGraffiti: autonomous multi-UAV spray painting. In *ACM SIGGRAPH 2019 Studio* (pp. 1-2).

Van Seijen, H., Van Hasselt, H., Whiteson, S., & Wiering, M. (2009). *A theoretical and empirical analysis of Expected Sarsa.* Paper presented at the 2009 ieee symposium on adaptive dynamic programming and reinforcement learning.

Vankadari, M. B., Das, K., Shinde, C., & Kumar, S. (2018). *A reinforcement learning approach for autonomous control and landing of a quadrotor*. Paper presented at the 2018 International Conference on Unmanned Aircraft Systems (ICUAS).

Wang, J., Sadler, C., Montoya, C. F., & Liu, J. C. (2016). *Optimizing ground vehicle tracking using unmanned aerial vehicle and embedded apriltag design.* Paper presented at the 2016 International Conference on Computational Science and Computational Intelligence (CSCI).

Wang, T., Qin, R., Chen, Y., Snoussi, H., & Choi, C. (2019). A reinforcement learning approach for UAV target searching and tracking. *Multimedia Tools Applications*, *78*(4), 4347-4364.

Xie, J., Peng, X., Wang, H., Niu, W., & Zheng, X. (2020). UAV Autonomous Tracking and Landing Based on Deep Reinforcement Learning Strategy. *Sensors, 20*(19), 5630.

Xu, Y., Liu, Z., & Wang, X. (2018). *Monocular Vision based Autonomous Landing of Quadrotor through Deep Reinforcement Learning.* Paper presented at the 2018 37th Chinese Control Conference (CCC).

Xu, Z., Wei, R., Zhao, X., & Wang, S. (2018). Coordinated standoff target tracking guidance method for UAVs. *IEEE Access, 6*, 59853-59859.

Yan, C., Xiang, X., & Wang, C. (2020). Towards real-time path planning through deep reinforcement learning for a UAV in dynamic environments. *Journal of Intelligent Robotic Systems, 98*(2), 297-309.

You, H. (2020). Mission-driven autonomous perception and fusion based on UAV swarm. *Chinese Journal of Aeronautics*.

Zhang, D., & Khurshid, R. P. (2019). Variable-Scaling Rate Control for Collision-Free Teleoperation of an Unmanned Aerial Vehicle. *arXiv preprint arXiv:1911.04466*.

Zhang, Q., Miao, J., Zhang, Z., Yu, F. R., Fu, F., & Wu, T. (2020). *Energy-Efficient Video Streaming in UAV-Enabled Wireless Networks: A Safe-DQN Approach.* Paper presented at the GLOBECOM 2020-2020 IEEE Global Communications Conference.

Zhang, W., Song, K., Rong, X., & Li, Y. (2018). Coarse-to-fine uav target tracking with deep reinforcement learning. *IEEE Transactions on Automation Science Engineering, 16*(4), 1522-1530.

Zhang, X., Li, X., Wang, K., & Lu, Y. (2014). *A survey of modelling and identification of quadrotor robot.* Paper presented at the Abstract and Applied Analysis.

Zhang, Y., Yuan, X., Li, W., & Chen, S. (2017). Automatic power line inspection using UAV images. *Remote Sensing, 9*(8), 824.

Zheng, Z., Jin, Z., Sun, L., & Zhu, M. (2017). Adaptive sliding mode relative motion control for autonomous carrier landing of fixed-wing unmanned aerial vehicles. *IEEE Access, 5*, 5556-5565.

Zhou, B., Wang, W., Liu, Z., & Wang, J. (2019). *Vision-based Navigation of UAV with Continuous Action Space Using Deep Reinforcement Learning.* Paper presented at the 2019 Chinese Control And Decision Conference (CCDC).