
Alireza KAZEMIPOUR
Tehran/Iran | (+98)09128607650
kazemipour.alireza@gmail.com
alirezakazemipour.github.io
github | linkedin


## Highlights

- Reimplementation of 14 famous Deep Reinforcement Learning papers.
- 3rd place in The RoboCup Iran Open 2018 International Competitions.
- 2 years of teamwork experience as a researcher in the KN2C robotics lab.
- Interdisciplinary knowledge in Reinforcement Learning, Deep Learning, Computer Vision and Robotics.

## Research Interest

- Machine Learning (Especially: Reinforcement learning and Deep Learning.)
- Computer Vision and Image Processing.
- Robotics.
- Control theory.

## Education

**K.N. Toosi University of Technology** *2015-2020*

B.Sc in Electrical Engineering

- Cumulative GPA: **3.2/4** (16.09/20) -via 143 Credits.

**Razi High school** *2011-2015*

High School Diplomas in Physics and Mathematics.

- Total GPA: **3.89/4** (19.49/20).

## Honors and Awards

- **3rd place** of The RoboCup Iran Open Competitions in Unmanned Aerial Vehicle League. [link] *2018*
- **6th place** of The RoboCup Asia-Pacific Competitions in Unmanned Aerial Vehicle League. [link] *2018*
- Ranked **within top 0.7 percent** in Iran's National University Exam among nearly 252,000 participants. *2015*

## Certificates

- **Reinforcement Learning Specialization** - University of Alberta on Coursera [link] *2021*
- **Neural Networks and Deep Learning** - DeepLearning.AI on Coursera [link] *2020*
- **Divide & Conquer, Sorting & Searching, and Randomized Algorithms** - Stanford on Coursera [link] *2021*

## Experience

**Research Assistant** *2017-2019*

Researcher at KN2C Robotics Lab, Supervisor: Dr. Hamid D. Taghirad

- Member of Computer Vision and A.I group of Micro Aerial Vehicle section.

**Teaching Assistant** *Fall 2019*

Signals and Systems, Instructor: Dr. Maryam mohebbi

**Teaching Assistant** *Winter 2020*

Fundamentals of Computer Vision, Instructor: Dr. Behrooz Nasihatkon

**Co-Instructor** *Spring 2020*

KN2C's course "Fundamentals of Artificial Intelligence", Instructor: Armin sadreddin (KN2C ex-Leader)

## Research Experience

### Rainbow *2020*

Combining Improvements in Deep Reinforcement Learning.

• Implementation of the DeepMind's paper Rainbow on Pong. [Project page]

### Exploration by Random Network Distillation *2020*

Exploration based on the agent's Intrinsic rewards that are proportional to the novelty it faces.

• Implementation of the OpenAI's paper Exploration by RND on Montezuma's Revenge. [Project page]

### Proximal Policy Optimization Algorithms *2020*

Policy gradient methods that alternate between sampling data and optimizing a "surrogate" objective function.

• Implementation of the OpenAI's paper PPO on Breakout, Ant and Mario. [Atari Page] [MuJoCo Page] [Mario Page]

### B.Sc. Final Project: Soft Actor-Critic *2020*

Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor.

• Implementation of the Berkeley's paper SAC on Humanoid. [Project Page]

### Discrete Soft Actor-Critic *2020*

Soft Actor-Critic For Discrete Action Settings.

• Implementation of the ICL's paper Discrete-SAC on Pacman. [Project Page]

### Deep deterministic Policy Gradient and Hindsight Experience Replay *2020*

Using DDPG to control continuously and HER to solve sparse-reward environments problem.

• Implementation of the DeepMind's paper DDPG and OpenAI's paper HER on FetchPickAndPlace. [Project Page]

### Twin Delayed Deep Deterministic Policy Gradient *2020*

Addressing Function Approximation Error in Actor-Critic Methods.

• Implementation of the McGill's paper TD3 on Hopper and Ant. [Project page]

### Tabular Reinforcement Learning *2020*

Implementing fundamental Reinforcement Learning algorithms in tabular format on Taxi gym environment.

• Off-Policy Temporal Difference Learning (Q-Learning). [Project Page]

• State–action–reward–state–action (SARSA). [Project Page]

• Backward View of TD(λ) both by Q-Learning and SARSA. [Q(λ) Project Page] [SARSA(λ) Project Page]

• Combination of Q-learning and Q-planning (Dyna-Q). [Project Page]

### Cycle GAN *2020*

Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks.

• Implementation of the BAIR's paper Cycle GAN on Horse2Zebra datatset. [Project Page]

### Persian Digits Classification *2020*

Designing and preparing instructions of 13th lab of Fundamentals of Computer Vision course

• Using Linear and RBF SVMs, KNN and Random Forest methods to classify images of handwritten Persian digits.

### Auxiliary and Deep Convolutional GANs *2019*

Using DCGAN to generate real-like images and Auxiliary GAN to improve quality.

• Implementation of the indico Research's paper DCGAN and Google Brain's paper AUXGAN on MNIST. [Project Page]

### Deep Dream and Style Transfer *2019*

Using Deep Dream to visualize a model's layer output and Style Transfer to transfer the style of an image to another.

- Implementation of Deep Dream using Inception-V3 and Style Transfer using VGG19. [Project Page]

### UAV Geo-Localization *2019*

Using convolutional neural networks to estimate GPS coordinates.

- Using features of ResNet50 to match current downward view of an UAV with a known provided map.

### Face Detection and Facial Expression Classification *2019*

Final project of Fundamentals of Computer Vision course.

- Using Cascade Detectors with LBP features to detect the face and CNN to classify the expression of it.

### Fast Fourier Transform on FPGA *2019*

Final project of Design of Digital System course.

- Implementation of the 64 point FFT on FPGA using VHDL.

### Line Detection and Following *2018*

Task in KN2C robotics lab.

- Using pure Computer Vision techniques like Contour Approximation Method, Image Filtering and Histograms on the embedded system (Odroid-XU4) of a drone to detect a colorful line in the downward view navigating to follow a specific path autonomously.

### Pole Collision Avoidance *2018*

Task in KN2C robotics lab.

- Using pure Computer Vision techniques **without any external aid of Depth Cameras or Laser Scanners** to navigate a drone autonomously through some obstacles (poles) without any collisions.

## Technical Skills

| | |
|---|---|
| **Programming Languages** | ***Experienced:*** Python \| C/C++ \| Bash \| VHDL ***Familiar:*** Java \| Assembly |
| **Libraries** | PyTorch \| TensorFlow \| Keras \| Scikit-learn \| Numpy \| Gym \| OpenCV \| Matplotlib |
| **Frameworks** | Qt \| ROS |
| **Version Control System** | Git |
| **Operating systems** | Linux (Ubuntu) \| Microsoft Windows |
| **Engineering software** | MATLAB and Simulink \| Keil µVision \| Altium designer \| CodeVision AVR \| ISE - Xilinx |
| **Embedded systems** | Upboard \| Odroid-XU4 \| Raspberry Pi |
| **Microcontrollers** | Atmel AVR Atmega and AtXmega series \| ARM Cortex-M series (STM32) |
| **General software** | Microsoft Office \| Adobe Photoshop |

## Language Proficiency

**Farsi:** Native

**English:**

TOEFL (internet-based) score: 104

- Reading: 24 Listening: 29 Speaking: 25 Writing: 26

GRE (General)

- Analytical writing assessment: 3.5 (38th Percentile)
- Verbal Reasoning: 154 (63rd Percentile)
- Quantitative Reasoning: 160 (72nd Percentile)

**❗References, further information, and proofs are available upon request.**