

Zheyuan Brian Zhang

540 Thompson Street, Ann Arbor, MI 48104
(413) 800-9529 · zheyuan@umich.edu
cozheyuanzhangde.github.io

RESEARCH INTERESTS

Artificial Intelligence, Robotics, Machine Learning, Deep Learning, Computational Cognitive Modeling, Computer Vision, Natural Language Processing, Cognitive Psychology, Neuroscience

EDUCATION

- **University of Massachusetts Amherst** Amherst, MA, United States
Computer Science (B.S.), GPA: 3.848/4.000 September 2018 - May 2022
Computer Science Courses: Introduction to Problem Solving with Computers, Programming with Data Structures, Programming Methodology, Computer Systems Principles, Reasoning Under Uncertainty, Introduction to Computation, Introduction to Algorithms, Web Programming, Social Issues in Computing (Writing), Introduction to Computer Vision, Artificial Intelligence, Introduction to Robotics: Perception, Mechanics, Dynamics, and Control, Game Programming (graduate-level), Machine Learning (graduate-level), Practice and Applications of Data Management, Introduction to Computer Graphics, Probabilistic Graphical Models (PhD-level)
- **University of Michigan Ann Arbor** Ann Arbor, MI, United States
Robotics (M.S.) August 2022 - April 2024

RESEARCH EXPERIENCE

- **Lab of Autonomous Robotics and Systems, Fudan University** Shanghai, China
Research Intern, Supervisors: Dr.Ruijiao Li, Prof.Huiliang Shang June 2021 - August 2021
 - **Robotic Grasping:** Built software on an Aubo-i5 robot arm with Robotiq gripper (USB), Chishine3D RGB-D camera and a UR5 robot arm with custom gripper (Modbus), Intel RealSense D435 RGB-D camera to grasp various fruits like apples, oranges, and bananas placing on the table using the open-source framework, ROS Melodic on Ubuntu 18.04. Implemented object localization and classification using You Only Look Once: Unified, Real-Time Object Detection (YOLOv4). Experienced hand-eye calibration and implemented grasping procedure including motion planning using MoveIt.
 - **Machine Vision:** Developed low-cost solutions like edge-based geometric shape detection and template matching to perform object localization, classification and grasp pose estimation on 2D RGB images.

PRE-PRINTS

- **Zheyuan Zhang.** A Computational Cognitive Model of Human Memory Based on Invertible Neural Networks. *OSF Preprints*. DOI: 10.31219/osf.io/58adh

PUBLICATIONS

- **Zheyuan Zhang, Huiliang Shang.** Low-cost Solution for Vision-based Robotic Grasping. *International Conference on Networking Systems of AI, IEEE INSAI, 2021*. DOI: 10.1109/INSAI54028.2021.00022.

HONORS AND AWARDS

- **Chancellor's Award (\$10,000 per year)** | University of Massachusetts Amherst
- **8 × Dean's List Honors (all semesters)** | University of Massachusetts Amherst
- **Second Prize - Excellent Paper** | IEEE INSAI2021
- **2nd Place - Win/Lose/Tie: 202/11/9** | Adversarial Search Game Robot Competition Best Record Leader Board
- **ROS Summer School 2021 Certificate** | ROS Education Foundation China
- **Instructor Endorsed Answerer** | Piazza

PROJECTS

- **Deep Reinforcement Learning (In Development)** January 2022 - January 2022
 - Designed a game (environment) including reward setups using Pygame for an agent to explore and make actions.
 - Implemented deep Q-learning.
 - Implemented a fully connected neural network and a convolutional neural network using PyTorch.

- FindUrCourse (www.findurcourse.com) – Node.js Web Application** *October 2020 – January 2021*
 - Designed a website for college students to search, rank, and rate courses. FindUrCourse currently has more than 70 courses and 100 comments.
 - Implemented front-end user interface using HTML/CSS, JavaScript with Bootstrap framework and back-end server using Node.js with Express framework.
 - Created PostgreSQL database for managing data of users and courses. Implemented server APIs which executes SQL commands on Node.js by pg-promise. Constructed 25 API end-points for front-end functionalities.
 - Implemented hash encryption on the server-side to protect users' passwords security.
- ZiZoyaOS - 32-Bit Operating System** *December 2020 – January 2021*
 - Developed a bootloader that reads disk sectors, switches the operating system into 32-bit protected mode, loads the kernel, and initializes the GDT using NASM Assembly.
 - Implemented a kernel using C, which has the following functionalities and components: IDT, ISR, IRQ initialization, ports communication, CPU interrupts handler for capturing, processing and giving feedback, display driver, keyboard driver (allows key combinations).
 - Designed a command-line interface and some functionalities, which include a calculator allowing simple operations.
- Quanin - Automatic Stock Screener** *March 2021*
 - Developed a stock screener based on stock trader's demand using Python and various libraries, including Baostock, Pandas, TA-Lib, Matplotlib.
 - Created a graphical user interface using C Sharp.NET and Winform.
- Clara in Wonderland - Unity3D Open-world Adventure Game** *November 2021*
 - Designed terrains and environment of an open-world scene including items, enemies, obstacles, etc.
 - Designed a game story using triggered narration and user interface including the main menu and in-game menu.
 - Applied various sound effects including background music, hitting, drinking, eating, picking up, etc.
 - Implemented health bar, hunger bar and thirst bar for elements of survival.
 - Implemented an inventory with interactions and items management using object-oriented programming.
 - Implemented game features. For example, filling water into an empty jar and the character can drink it when thirsty.
 - Developed enemies attack logic with animator controller and pathfinding using NavMesh.
- Lazy \LaTeX for Homeworks - \LaTeX Code Generator** *November 2021*
 - Developed a \LaTeX code generator which takes in simple commands and generate \LaTeX code automatically.
 - Implemented commands myname(), courseName(), homeworkName(), question(), answer(), code().
 - Implemented features including parsing from matrix string to \LaTeX matrix.
 - Created a graphical user-interface using C Sharp.NET and Winform.
- Other Projects** *2016 - 2021*
 - Implemented a blockchain using CPP and Cryptography Library.
 - Developed several Windows applications using C Sharp.NET and VB.NET.
 - Experienced compiler design by trying to create an interpreter (not developed) using C, Flex and Bison.

SELECTED COURSE PROJECTS

- Computer Vision** *Spring 2021*
 - Project1: Matrix manipulation, image formation, light, white balance.
 - Project2: Prokudin-Gorskii images alignment (cosine similarity), color image demosaicing (Bayer filter, nearest neighbor, linear interpolation).
 - Project3: Image contrast improvement (contrast stretching, gamma correction, histogram equalization), image denoising (Gaussian filtering, Median filtering), hybrid images.
 - Project4: Image gradient and orientation histogram, corner detection (simple corner detector, Harris corner detector).
 - Project5: Panoramic image stitching (detect corners, feature extraction, computing matches, estimating transformation using RANSAC).
 - Project6: Decision trees, k-nearest neighbors classifier, linear classifier.
- Artificial Intelligence** *Spring 2021*
 - Project 1: Solving a weighted 8-puzzle (searches: breadth-first, uniform-cost, greedy best-first, and A*; Heuristic functions: "number of misplaced tiles", "Manhattan distance" and modified "Manhattan distance").
 - Project 2: Adversarial search (minimax, alpha-beta pruning, evaluation heuristic).
 - Project 3: Constraint satisfaction problems, logic/logical Inference.

- Project 4: Bayesian network probability estimation (simple sampler, rejection sampler, likelihood weighting sampler).
- Project 5: Value iteration for MDP (Markov Decision Processes).
- Project 6: Decision tree classification (learn a decision tree from a data set which contains facts about the 350 cities and towns in Massachusetts, with a class label indicating the local results of the 2020 U.S. presidential election).

• **Robotics: Perception, Mechanics, Dynamics, and Control**

Fall 2021

- Project 1: Degree of freedom, forward kinematics, jacobian, torques.
- Project 2: Euler angles, $SO(3)$, $SE(3)$, animation of frame end-effector.
- Project 3: Robot kinematic simulation.
- Project 4: Jacobian and inverse kinematics.
- Project 5: Dynamic simulation.
- Project 6: Operational space control.
- Project 7: Discrete impact based contact simulation.
- Paper Review: Mini Cheetah: A Platform for Pushing the Limits of Dynamic Quadruped Control.

• **Game Programming**

Fall 2021

- Project 1: Simple 3D game includes implementation of textured terrain, 3D primitive, character controller, jump scripting, terrain escaping prevention, obstacles.
- Project 2: UI menu, keyframe animation, programmatic animation, camera switching.
- Project 3: An action game includes implementation of animation controller and turret deflection shooting strategy (kinematics).
- Project 4: Physics simulation of a plane pendulum with various methods of numerical integration and a cannon game with second Newton's law of motion.
- Project 5: A game where a random, maze-like environment is automatically generated using constraint satisfaction problem (CSP) and post-processing using A* search algorithm.

• **Machine Learning**

Fall 2021

- Project 1: K-nearest neighbors classifier, decision tree classifier, hyper-parameter tuning, MLE (Maximum Likelihood Estimation), MAP (Maximum A Posterior).
- Project 2: Bag of words, Naive Bayes, probabilistic classification, logistic regression, hyper-parameter tuning.
- Project 3: SVM (Support Vector Machine), ensemble methods (random forest and AdaBoost), cross-validation, hyper-parameter tuning.
- Project 4: Linear regression and hyper-parameter tuning, fully connected neural network, stacking classifiers.
- Project 5: Convolutional neural networks, singular value decomposition (optimal k-rank approximation, sub-optimal k-rank approximation), principle component analysis.
- Final Project: Kaggle PetFinder.my - Pawpularity Contest: Predict the popularity of shelter pet photos.

• **Computer Graphics**

Spring 2022

- Project 1: Introduction to WebGL and three.js.
- Project 2: Image processing: adjusting contrast, saturation and sharpness, uniform quantization, random dithering, ordered dithering, image mosaic.
- Project 3: Curve generation: Bezier, BSpline (open and closed), Hermite (open and closed).
- Project 4: Loop subdivision: computing subdivided vertices and faces.
- Project 5: Hierarchical Models: bipedal robot simulation.
- Project 6: Ray Tracing: sphere's intersect function, triangle's intersect function, shading function, raytracing function and SpotLight class.

• **Probabilistic Graphical Models**

Spring 2022

- Project 1: Bayesian network: maximum likelihood learning, conditioning, probability queries answerer, computing test log-likelihood, designing network structure for the heart disease domain (modeling).
- Project 2: Undirected models and message-passing: computation of node potentials, negative energy, log normalizing constant using exhaustive summation, most likely joint labeling, marginal probability distribution, log-space messages, single variable marginals and pairwise marginals using message-passing, prediction accuracy, average log likelihood, numerical optimization.
- Project 3: Learning in CRFs: implementation of objective function using average log conditional likelihood and its gradient functions, optimization for maximizing the objective function using L-BFGS-B algorithm.
- Project 4: Markov chain Monte Carlo: derivation of conditional distribution, Gibbs sampling, image denoising using MCMC, tuning hyper parameters.
- Project 5: Variational Auto-Encoder: derivation of ELBO with reparameterization trick, VAE training, dimension reduction, data generalization (generative model), log-likelihood estimation, anomaly detection.

SKILLS

Programming Languages and Tools/Software: *Python, C, C++, JavaScript, Java, VB/VB.NET, C#/C#.NET, NASM Assembly, SQL, Shell Script, HTML/CSS, MATLAB, L^AT_EX, Linux, Robot Operating System (ROS)*

INVITED PRESENTATIONS

- **Oral presentation at INSAI2021. November 24, 2021.**

ACTIVITIES

- **HackUMass IX**

LANGUAGES

- **Chinese** (Native)
- **English** (Fluent)
- **Korean** (Beginning)