

# Zheyuan Brian Zhang

Ann Arbor, MI  
zheyuan@umich.edu  
cozheyuanzhangde.github.io

## RESEARCH INTERESTS

---

Computer Vision, Multimodal Learning, Multitask Learning, Generative Models, Robotics, Cognitive Science

## 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  
*Related Courses:* Mathematics for Robotics (Fall 2022), Robotic Systems Laboratory (Fall 2022), Natural Language Processing (Fall 2022)

## 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.

## WRITINGS AND 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
- **2<sup>nd</sup> 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

## ONLINE COURSES

---

DeepLearning.AI Deep Learning Specialization taught by Andrew Ng, DeepLearning.AI Generative Adversarial Networks (GANs) Specialization taught by Sharon Zhou, Reinforcement Learning Course (UCL) taught by David Silver

## 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.
- **FindUrCourse ([www.findurcourse.com](http://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 arithmetic 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-friendly 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 with object-oriented programming.
  - Implemented game features. For example, filling water into an empty jar and the character can drink it when thirsty.
  - Developed enemies attacking 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 functionalities including parsing from matrix string to  $\LaTeX$  matrix.
  - Created a graphical user-interface including quick view 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 a interpreter (not developed) using C, Flex and Bison.

## SELECTED COURSE PROJECTS

---

- **Computer Vision** *Spring 2021*
  - Project1: Used NumPy to perform matrix operations. Solved theoretical problems of image formation and light. Implemented white balance algorithm.
  - Project2: Implemented image aligning. Implemented color image mosaicing according to the Bayer pattern and image demosaicing including nearest neighbor and linear interpolation methods.
  - Project3: Implemented image contrast improving algorithms including contrast stretching, gamma correction and histogram equalization methods. Implemented image denoising including Gaussian filtering and medium filtering methods. Implemented hybrid image (rabbit and cat) algorithm.
  - Project4: Wrote a function to compute image gradient magnitude and orientation with histogram visualization. Implemented simple corner detector and Harris corner detector.
  - Project5: Implemented panoramic image stitching algorithm (detecting corners, feature extraction, computing matches and estimating transformation using RANSAC).

- Project6: Implemented machine learning algorithms including decision trees, k-nearest neighbors classifier and linear classifier on MNIST handwritten digit classification.

## • Artificial Intelligence

*Spring 2021*

- Project 1: Implemented breadth-first search, uniform-cost search, greedy best-first search and A\* search with heuristic functions including “number of misplaced tiles”, “Manhattan distance” and modified “Manhattan distance” to solve a weighted 8-puzzle.
- Project 2: Implemented minimax adversarial search with an evaluation heuristic and alpha-beta pruning as a computer player in a variant Connect game.
- Project 3: Solved constraint satisfaction problems (CSPs) and logic inference problems.
- Project 4: Implemented Bayesian network probability estimator with simple sampler, rejection sampler and likelihood weighting sampler.
- Project 5: Implemented value iteration for gridworld Markov Decision Process (MDP).
- Project 6: Implemented 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.

## • Machine Learning

*Fall 2021*

- Project 1: Implemented k-nearest neighbors classifier and decision tree classifier with hyper-parameter tuning on Pima Indians Diabetes Dataset. Derived posterior probability and Maximum Likelihood Estimation (MLE) estimator (N samples independently drawn from a normal distribution with known variance and unknown mean) and Maximum A Posterior (MAP) estimator.
- Project 2: Constructed bag-of-words model and implemented multinomial naive Bayes model on Women’s Clothing E-Commerce Reviews Dataset. Derived Bayes optimal classifier and MLE estimates. Implemented a l2-regularized logistic regression classifier.
- Project 3: Implemented a support vector machine (SVM) classifier with linear kernel, polynomial kernel and RBF kernel for Breast Cancer Wisconsin (Diagnostic) Dataset. Developed ensemble models including random forest classifier and AdaBoost classifier using Scikit-learn. Implemented cross-validation procedure.
- Project 4: Derived mathematical proofs for ridge regression. Developed ordinary least squares (OLS) linear, lasso and ridge regression model with hyper-parameter tuning. Derived partial derivatives used for backpropagation and implemented a fully connected neural network using PyTorch library for classifying MNIST digits. Developed stacking classifiers using Scikit-learn.
- Project 5: Developed convolutional neural networks for classifying MNIST digits using PyTorch, Implemented singular value decomposition (SVD) to perform optimal k-rank approximation and sub-optimal k-rank approximation on image compression. Implemented principle component analysis (PCA) on MNIST dataset.
- Final Project: Attended Kaggle PetFinder.my - Pawpularity Contest which aims to Predict the popularity of shelter pet photos. Developed traditional machine learning models including linear models and ensemble models using Scikit-learn to train and predict on the dataset. Implemented DL-enhanced model including a CNN with a self-created network architecture for images using PyTorch and lasso regression for meta-data. Implemented a 2-head neural network model which includes a CNN for images and a FCNN for meta-data.

## • Probabilistic Graphical Models

*Spring 2022*

- Project 1: Derived and implemented maximum likelihood learning in the directed model with a probability query answerer. Computed log-likelihood with five-fold cross-validation to assess the performance of the model. Designed a network structure for the heart disease domain and repeated the assessment.
- Project 2: Implemented simple exhaustive inference algorithm for the conditional random field (CRF) model. Implemented sum-product message passing inference algorithm. Derived maximum likelihood learning for conditional random field models.
- Project 3: Implemented average log conditional likelihood as the objective function and its gradient functions using sum-product message passing. Implemented the learning algorithm for optimization of maximizing the objective function using L-BFGS-B algorithm. Applied log-sum-exp trick to resolve numerical underflow and overflow.
- Project 4: Derived and implemented Gibbs sampling Markov chain Monte Carlo (MCMC) algorithm to perform image denoising using a grid-structured CRF model. Tuned hyper-parameters by grid search.
- Project 5: Derived evidence lower bound (ELBO) with reparameterization trick. Implemented variational autoencoder (VAE) training algorithm on images of 0/1 picked from the MNIST dataset with the automatic differentiation tool JAX. Applied VAE to dimension reduction and image generalization as a generative model. Estimated the log likelihood of training and test set by Monte Carlo samples. Applied VAE to anomaly detection to non 0/1 images.

## 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<sup>A</sup>T<sub>E</sub>X, Linux, Robot Operating System (ROS)*

## INVITED PRESENTATIONS

---

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

## ACTIVITIES

---

- **HackUMass IX**

## LANGUAGES

---

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