WILBERT PUMACAY

M.Sc. Student in Computer Science
wilbert.pumacay@ucsp.edu.pe
wpumacay.github.io

EDUCATION

San Pablo Catholic University (Perú) , M.Sc. *2017 - present*
Advisor: PhD José Eduardo Ochoa Luna
Department of Computer Science

National University of Engineering, UNI (Perú), B.Eng. *2008 - 2012*
Mechatronics Engineering
Department of Mechanical Engineering

WORK EXPERIENCE

National University of Engineering (UNI), Research Assistant *2012-2013*

- Worked implementing C/C++ Hardware Abstraction Layers for microcontrollers and DSPs, exposing their peripherals through easy to use APIs for various projects.
- Developed required software libraries (in C/C++, MATLAB and LabView) to interact with actuators (motor-drivers) and sensors (IMUs, LIDAR), used in a differential drive robotics platform built from scratch (as part of a funded research project).
- Implemented Localization (particle filter based localization using LIDAR), Planning (A* based path-finding), and Control (sliding-mode and fuzzy) algorithms for the aforementioned robotics platform (as a personal side project).

Bamtang Games, Software Engineer *2013 - 2016*

- Worked as a Gameplay, UI and Engine programmer for the web division of the studio, writing the required game engines for new games and required gameplay features (mostly in Javascript).
- Developed tools for artists and designers to integrate their assets into the engine and speed up their workflow (mostly in C++ and Python).
- Worked as a Gameplay and UI programmer for the console division of the studio, helping with gameplay features for PS4|XBOX|PC games, and tweaking engine features for each platform.

SKILLS

C/C++: Good working knowledge. Can architect and maintain Projects/APIs (some projects are listed below)

- OpenGL-based Rendering engine[1]
- Engine-agnostic simulations[2] (under development)

Python: Good working knowledge. Can create packages, add features to codebases and maintain them (some projects are listed below)

- DeepRL projects from Udacity DeepRL nanodegree[3]
- Robotics projects from Udacity RoboND nanodegree (navigation[4] | kinematics[5] | perception[6] | deep-learning[7]).

[1] https://github.com/wpumacay/tiny_renderer
[2] https://github.com/wpumacay/loco
[3] https://github.com/wpumacay/deeprlnd-projects
[4] https://github.com/wpumacay/RoboND-Rover-Project/blob/master/README.md
[5] https://github.com/wpumacay/RoboND-Kinematics-Project/blob/master/REPORT.md
[6] https://github.com/wpumacay/RoboND-Perception-Project/blob/master/REPORT.md
[7] https://github.com/wpumacay/RoboND-DeepLearning-Project/blob/master/REPORT.md

Javascript: Good working knowledge. Can create web-demos and tools (some projects are listed below) (games can't be listed due to privacy policy of the studio's clients)

- WebGL-based visualizer[8], written in Typescript.
- Denavit-Hartenberg playground[9], and FK Kuka demo[10] (might take a while to load)

Pytorch|Tensorflow: Working knowledge. Can create models, data-utils, debug models, present results and debug|fix existing projects (some projects are listed below)

- DeepRL projects from Udacity DeepRL nanodegree

ROS: Working knowledge. Can design experiments, create tools and use comfortably the piping provided by ROS. (some projects are listed below)

- Robotics projects from Udacity RoboND nanodegree (kinematics | perception)
- Helped in Udacity's forum to debug fellow students' projects.

Embedded systems: Working knowledge (a bit rusty). Can create HAL libraries for microcontrollers and DSPs. Comfortable developing libraries from scratch using the datasheet of a given module|driver|sensor. (some old projects are listed below)

- HAL libraries[11] for Texas Instrument's TM4C123GH6PM ARM-MCU
- HAL libraries[12] for Texas Instrument's MSP430F5529 MCU

Other: I've also worked with the following technologies|libraries:

- OpenCL and CUDA
- OpenCV
- Qt|PyQt|PySide
- Soldering and electronics prototyping skills.
- CAD design (Autodesk Inventor)

PRESENTATIONS

- Poster presentation at ***Khipu.ai***. Presented current progress of my Msc. thesis: *"Loco: A DeepRL framework for Robot Locomotion in Complex Environments".*
- A tutorial on DeepRL at "I Simposio Peruano de Deep Learning". Slides here.

MISCELLANEOUS

- Finished Udacity Deep Reinforcement Learning Nanodegree.
- Finished Udacity Robotics Nanodegree (term-1).
- A pull-request[13] (not merged) for a visualizer[14] for Deepmind's *dm_control* package.
- A pull-request (not merged) for extra tools[15] for Udacity's Quadrotor Simulator (made in Unity).

[8] https://github.com/wpumacay/leoJS
[9] https://wpumacay.github.io/leoJS/playground.html
[10] https://wpumacay.github.io/leoJS/index.html
[11] https://github.com/wilsanph/tivaCppLib/tree/master/tivaCppLIte/tivaCpp
[12] https://github.com/wilsanph/gmsp430cpp
[13] https://github.com/deepmind/dm_control/pull/45
[14] https://github.com/wpumacay/dm_control/blob/master/dm_control/glviz/README.md
[15] https://github.com/wpumacay/RoboND-QuadRotor-Unity-Simulator/blob/master/EXTRAS.md