ANJALI VIJAYARAGHAVAN

Software Engineer

California

(650) 304 5545

vjanjali@gmail.com

Technical Expertise

TensorFlow, CNN, OpenCV, PHP, HTML5, CSS3, JavaScript, cURL, AWS EC2, Bootstrap, Django, Boto3, AWS EC2, AWS RDS, Python, SQLAlchemy, C++, Google Cloud Platform, ElasticSearch, Kibana, Grafana, SQL, MySQL, YAML, StackStorm, NodeJS, ExpressJS, MongoDB, Jade, MEAN Stack, SAP ABAP CRM, SAP WebUI

Certifications

- Six Sigma – Green and Black belts certified (2015)
- Lean Management (2015)
- Open Sap Record of Achievement - SHINE Reference for Native SAP HANA Application Development

Education

M.S. Software Engineering

2017-2019

Charles W. Davidson College Of Engineering, San Jose State University (California, USA)

- Completed M.S in Software Engineering, specializing in Cloud Computing and Virtualization with a CGPA score of 3.454 on 4.

B.Tech. Information Technology

2007-2011

Amrita School Of Engineering, Amrita Vishwa Vidyapeetham (Bangalore, India)

- Completed B.Tech, First Class with Distinction with a CGPA of 8.33 on 10.

Summary

Versatile software developer focussed on bringing forth optimally automated and customer centric solutions.

Work experience

MAY 2018 - OCT 2018

Software Engineer (Internship)

Synopsys Inc (Sunnyvale, USA)

Network IT Intern

- Developed the code to extract, transform and load data to build a dashboard to proactively monitor the network.

IT DevOps Intern

- Worked on developing rules, sensors, actions, workflows on Stack Storm. Rules created to initiate a trigger based on words found in a string. Sensors developed to listen from a server, port to know when to set the trigger. Action to create the schema or structure of the workflow. Workflow to define the tasks to be executed.

JUN 2011 - APR 2014

Software Engineer

Accenture Services Pvt. Ltd. (Bangalore, India)

- Worked with users to gather functional requirements
- Performed technical risk analysis on the requirements and developed the code.
- Debugged CRM-ECC middleware
- Code enhancements done in both CRM and ECC
- Actively managed and reported status of the middleware proactively
- Worked to solve incidents on CRM Service

Projects

California, USA

Machine Learning

- Using OpenCV, built a program for a robotic arm to detect a line. Video capture from the camera was split into frames, each frame is processed to reduce obstacles that would slow the detection process such as - greying out, blurring (to remove image noise) using Gaussian Blur and also creating an area of focus. The line is detected based on its edge and contours have also been taken into account to be more precise.
- Using TensorFlow, built a program to detect an object. After capturing images from the internet, the images were annotated using LabelImg. Single Shot MultiObject Detection algorithm was found to be best with regards to speed, when used in a manufacturing plant and to increase its accuracy MobileNet was combined. Processed images were converted to tfRecords, which was used to train the MobileNet SSD model provided by TensorFlow. Before testing the model, an inference graph is generated to use as a source of truth during the detection phase. OpenCV is used to feed the model with video stream from the camera, then with the trained model and label map (to detect multiple objects) are loaded onto memory. Scores are kept as a confidence check, higher the number most likely to be the object as in the trained model. Boxes drawn on the image, labelled with detected object.

Web Application

- Used PHP to build the server and HTML + CSS to design the user interface. The website was built to be an online e-commerce portal for a fashion design studio. It was hosted using a third party service.

- Used Django as the framework to build a website that was hosted on AWS EC2 and used AWS RDS as the database. The website was built as a custom Sensor as a Service tool allowing users to select sensors to build a sensor management service. Boto3 was used to integrate the EC2 operations with Django.
- Used Python to build a Flask server to host a finance website to display best stock options. Google Cloud Platform was used to host the Flask server.
- Developed an application to determine the safety factor of a locality using the data captured from a public data source. The data stored in MongoDB, was used for visualizing graphs through tools like Tableau and Google Charts. The application was built using Express which is a web framework built for NodeJS and the user interface was enhanced using AngularJS

Bangalore, India

- Developed ABAP script to retrieve confirmations with errors and automate the fix to the same.
- Developed a BW extractor to gather data and transfer to the BW/BI system.