

# VINCENT ZHU

✉ vincent9825@gmail.com ◦ 📧 sumail25 ◦ 📝 Blog ◦ in Vincent

## EDUCATION

<b>The University of New South Wales, Sydney</b> Master of Information Technology, Artificial Intelligence	03/2021 – 12/2022
<b>Northeastern University, China</b> Bachelor of Science, Computer Science <i>Courses: Machine Learning, Data Mining, Deep Learning, Computer Vision, Big Data, Algorithms, Data Structure, etc..</i>	09/2016 – 07/2020

## JOB EXPERIENCE

<b>Zhejiang Ruida Machinery Co., Ltd</b> <i>Automation Department</i> Developer Intern	04/2021 – 05/2022
<ul style="list-style-type: none"><li>Developed an ML/DL pipeline based on industrial image features using YOLOv5 for object recognition and detection in assembly line.</li><li>Conducted model optimization experiments to improve model performance by optimizing model structure and training strategies for different scenarios and data.</li><li>Led a team in exploring and researching the application implementation strategies of Deep Learning in defect detection, object detection, and X-ray image enhancement in the industrial field.</li><li>Managed and deployed a rotating object detection project, including research, development, and experiments.</li></ul>	
<b>Northeastern University</b> <i>Machine Learning Lab</i> Research Assistant	07/2018 – 06/2019
<ul style="list-style-type: none"><li>Collaborated on the research of medical imaging data, effectively processed and maintained case data to ensure completeness and accuracy.</li><li>Participated in the development of weakly supervised learning-based medical image detection and segmentation solutions, as well as result visualization, optimizing for small-scale datasets using techniques such as Pseudo Mask.</li><li>Designed and improved data processing workflows to enhance model robustness and accuracy through methods such as image enhancement, data cleaning, and feature extraction.</li><li>Contributed to literature reviews and replication work, gaining deeper understanding in the field of ML/DL/DS.</li></ul>	

## PROJECTS

<b>Image Super Resolution Web App Based on Real-ESRGAN</b> Tech: Python, Pytorch, Flask, html, css	🔗
<i>Improved and optimized the model inference process and deployed it on the web, enabling a convenient image processing solution.</i> <ul style="list-style-type: none"><li>Optimized the model inference CUDA memory allocation to make it more suitable for low-performance computing platforms.</li><li>Improved the preprocessing of input images to support single-channel images.</li><li>Developed a web application using Flask and HTML based on improved model, providing a user-friendly interface for interactive experience.</li><li>Implemented features such as image upload, parameter selection, preview, and download.</li></ul>	
<b>Medical Image Detection and Segmentation</b> Tech: Pytorch, OpenCV, Matplotlib, Pandas, Scikit-learn	🔗
<i>Detect and segment the livers and lungs from medical images.</i> <ul style="list-style-type: none"><li>Preprocessed and augmented medical image datasets (DICOM) for liver and lung detection and segmentation.</li><li>Developed a weakly supervised deep learning pipeline based on U-Net for medical image segmentation.</li><li>Implemented various U-Net and Loss variants to handle diverse data and scenarios.</li><li>Provided tools for analyzing input images, predicting masks, and visualizing ground truth.</li><li>Utilizing ResNet-34 as the backbone, employing Focal Loss, Dice optimization, and other methods. Final mean Intersection over Union (mIoU) achieved a 26% improvement compared to the baseline.</li></ul>	
<b>Rotate Object Detection Based on YOLOv5 and CSL</b> Tech: YOLOv5, Pytorch, OpenCV, Matplotlib	
<i>Locate the coordinate and rotation angle of assembly line product items and deliver the ONNX model.</i> <ul style="list-style-type: none"><li>Utilized CSPResNet-50 as the backbone and developed a pipeline based on YOLOv5 for object detection.</li><li>Used Circular Smooth Label(CSL) that discretizes continuous problem of rotation angle to classification problem to implement rotation of recognition boxes.</li><li>Designed image collection and annotation processes, improved data input/output, and adapted loss functions.</li><li>Applied data augmentation techniques such as Mosaic, Mixup, and random affine transformations.</li><li>Performed Dynamic Anchor Sampling strategy to increase sampling rates and accelerate network convergence.</li></ul>	

## **Research on Cell Detection Based on Computer Vision**

Tech: Pytorch, OpenCV, Matplotlib, Pandas

*Explored the identification, segmentation, tracking, and detection of mitosis for biological cells.*

- Designed and tested pre-processing filters and transformations for image augmentation, achieving an 8% accuracy improvement over the baseline.
- Utilized the segmentation generated by watershed as a pseudo mask to provide sufficient training data.
- Implemented mitosis detection based on an improved VGG-19 model. The final model achieved a prediction accuracy of 97%.
- Visualized cell identification, segmentation, tracking, and mitosis detection.

## **Global Wheat Detection using Faster R-CNN**

Tech: Pytorch, OpenCV, Matplotlib, NumPy, Pandas, Scikit-image

*Detect, locate, and visualize the wheat heads from the images.*

- Analyzed and processed a dataset of 3000+ images, visualized bounding boxes.
- Built a training and prediction pipeline using ResNet-152 and EfficientNet-B0 as backbone networks for wheat spike detection with Faster R-CNN.
- Implemented data augmentation techniques such as pseudo-labeling, Mixup, and affine transformations using Albumentations.
- Implemented nested cross-validation and soft-NMS. Prediction achieved AU-ROC of 0.77, with a 8% improvement from naive baseline model.

## **Handwritten Chinese Optical Character Recognition**

Tech: Python, Tensorflow, OpenCV, NumPy, Kears

*Implemented offline handwritten Chinese character recognition using multiple deep models for training and experimentation.*

- Developed a deep learning OCR pipeline based on CNN using TensorFlow (accuracy: 95%).
- Designed different CNN models and data augmentation techniques to improve model performance.
- Improved the recognition accuracy by 11% by applying Gaussian filter, edge enhancement.

## **SKILLS**

- **Program Language:** Python, C++, C, SQL, R, Scala, Shell, Bash, HTML.
- **Frameworks:** Pytorch, TensorFlow, OpenCV, Flask, Scikit-learn, Scikit-image, NumPy, Pandas, Matplotlib, Spark, hadoop.
- **Platforms and Tools:** Linux, Window, Mac OS, VS Code, RStudio, Conda, Jupyter Notebook, Docker, GitHub, Jira, AWS.