MUHAMMAD AMMAR UL HASSAN

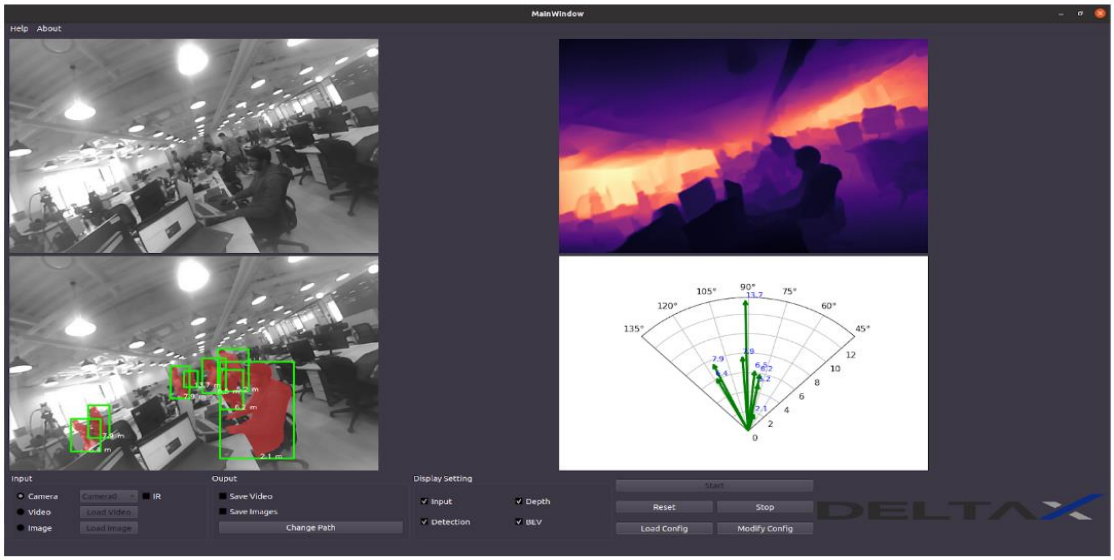
Phone: (+82) 10-2637-3994
Email: ammar.instantsoft@gmail.com
Location: 69 Jayang-ro 19-gil, Gwangjin-gu, Seoul, South Korea
DOB: 1991/01/01, Nationality: Pakistan
GitHub: https://github.com/ammar-deep
Personal Website: https://ammar-deep.github.io

PORTFOLIO

This portfolio highlights expertise in AI, computer vision, and deep learning across industrial and academic projects. It includes real-time systems for tracking, autonomous driving, and anomaly detection, as well as research in generative models, font synthesis, and skeletonization.

INDUSTRIAL PROJECTS AND POC'S

1. REAL-TIME WORKER DISTANCE ESTIMATION AND TRACKING WITH IR AND GPS INTEGRATION



Highlights

- Depth estimation based on IR images using monocular depth estimation model.
- Detected object tracking using YOLO & DeepSORT algorithms.
- Integrate GPS and monocular depth estimation model to optimize BEV (Bird's eye View) coordinates and display final result on UI using PyQt.

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, Scikit-Learn
- **Methodology:** Detection, Segmentation, Monocular depth Estimation, PyQt, GPS

2. AI-Powered Driving Visualization System for Enhanced Customer Experience

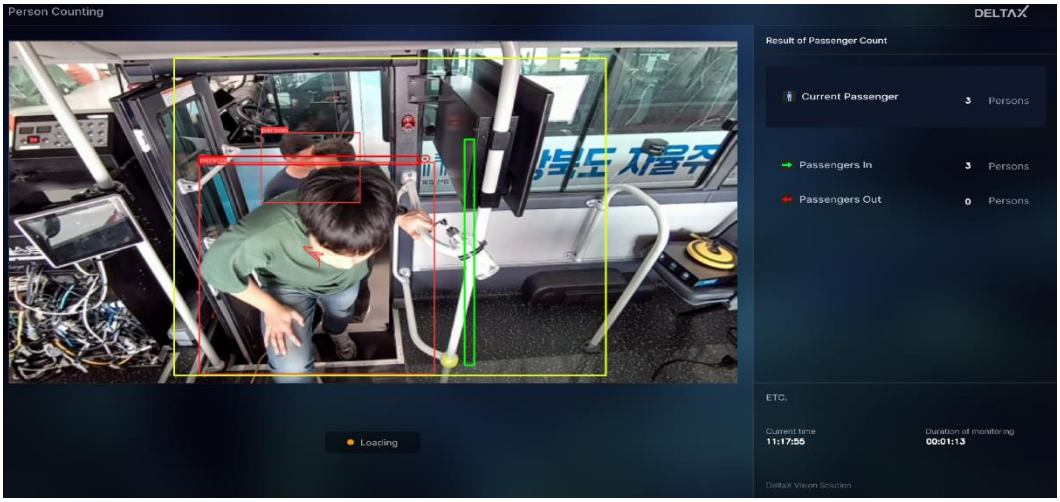
Highlights

- Classification of road hazards (on-road/off-road, unpainted speed bumps, curb, potholes) through segmentation model.
    - Light weight Segmentation model designed to run on TI board
- Curb distance estimation (Warning system)
    - Approach 1: Monocular depth estimation based warning
    - Approach 2: ROI based warning
- Implementation of BEV and JellyView using 4 cameras.
- Height, tire contact, and approach angle prediction through feature matching.

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, Scikit-Learn
- **Methodology:** Segmentation, Light weight monocular Depth Estimation, Bird's Eye View, Feature Matching, Jelly View, Embedding(TI)

3. SMART BUS MONITORING: CAMERA-BASED PASSENGER ENTRY AND EXIT TRACKING



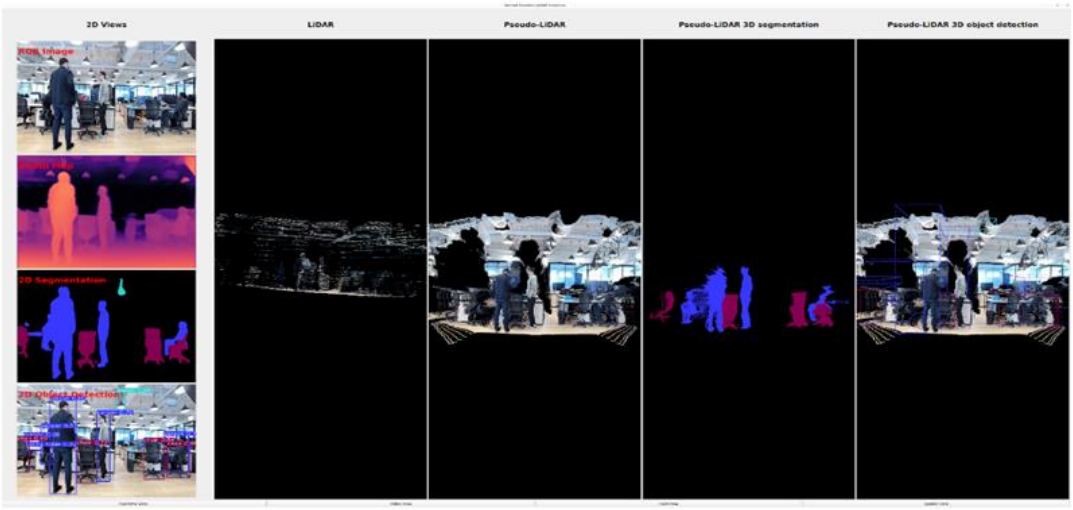
Highlights

- Person detection and tracking in ROI using YOLO & DeepSORT algorithms'
- Dynamic ROI for efficiently tracking passenger movements within a bus
- Counting number of person boarding in and boarding out

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV
- **Methodology:** detection, tracking, dynamic ROI, UI integration

4. EXPLORING PSEUDO LIDAR: COST-EFFECTIVE 3D SCENE UNDERSTANDING FROM CAMERA-BASED DEPTH ESTIMATION



Highlights

- Monocular depth estimation based 3D point cloud (Psueod-LiDAR)
    - Light weight depth estimation model
- 2D-3D object detection and segmentation
- Dense vs Sparse point cloud comparison (Pseudo LiDAR vs LiDAR)

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, Open3D, OpenGL
- **Methodology:** Mono depth estimation, object detection, semantic segmentation, LiDAR camera calibration, real-time LiDAR vs Pseudo LiDAR using PyQT

5. MULTI-CAMERA ROAD ENVIRONMENT PERCEPTION AND 3D RECONSTRUCTION FOR AUTONOMOUS DRIVING

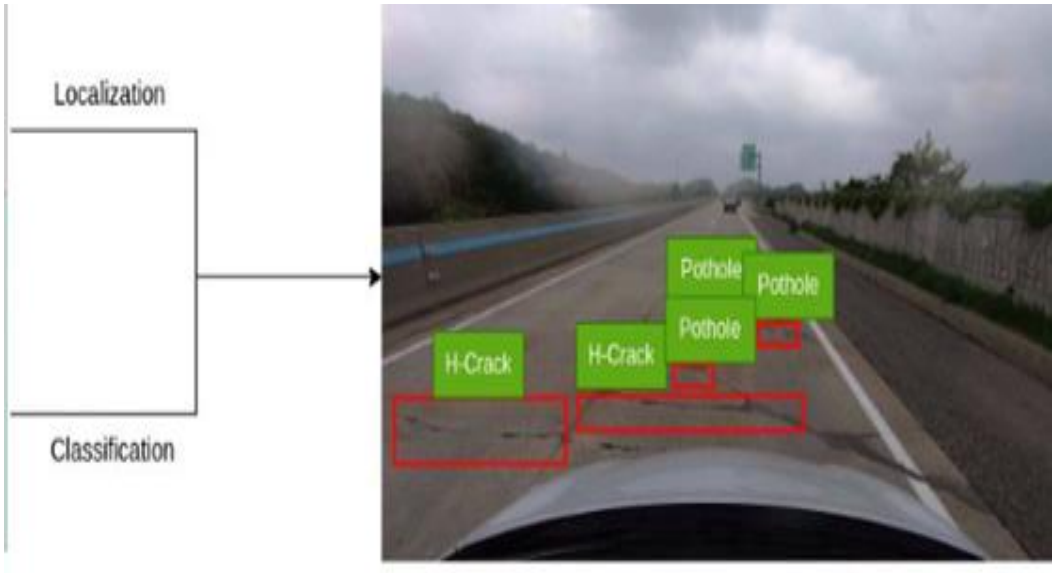
Highlights

- Image based multi-camera 3D object detection and tracking
- Multi-camera Static-Dynamic object segmentation
- Speed and direction estimation of detected vehicles
- Multi camera 3D reconstruction

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, Open3D/OpenGL
- **Methodology:** 3D object detection, instance/panoptic segmentation, point cloud stitching and registration

6. Efficient Detection of Road Surface Anomalies with ONNX and Hailo AI Acceleration



Highlights

- Pothole & crack detection
- obb bounding box based model training

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, onnxruntime
- **Methodology:** 2D object detection, onnxruntime inference, HAILO porting

6. Efficient Gaze Estimation System: From GPU to Optimized CPU with ONNX Integration

Highlights

- Converted a GPU-based gaze estimation pipeline (L2CS-Net) into an optimized CPU-based solution, achieving real-time performance.
- Transformed the entire pipeline into ONNX format, enabling cross-platform compatibility and easier deployment.
- Replaced the default face detector (RetinaFace) with a lightweight, ONNX-compatible detector for streamlined integration.

- Integrated the ONNX-based face detection and L2CS-Net gaze estimation models into a single, PyTorch-free Python module.
- Optimized Python code line by line, achieving 8-12 FPS for single-face gaze estimation on CPU-only hardware.

Tools Used

- **Framework and Libraries:** PyTorch, OpenCV, onnxruntime
- **Methodology:** 2D object detection, onnxruntime inference, HAILO porting

7. DHL AND ENDICIA POSTAGE LABEL PRINTING EXTENSIONS FOR OPENCART

Highlights

- DHL Label Printing Extension serves OpenCart based online stores.
- You can generate DHL label directly from Order page of your store.
- Module settings allow you to change many configurations.
- You can set output format to image or PDF for your labels.
- You can also print labels directly from browser using Zebra Printer.
- You can generate unlimited number of labels using this extension.

Programming Language

- OpenCart, PHP, HTML, MySQL

Extensions

- DHL Shipping Labels with Postage
-

Demo

- YouTube Tutorial

ACADEMIA PROJECTS AND PUBLICATIONS

1. CONTROLLABLE UNSUPERVISED GENERATIVE MODEL

Highlights

- Designed a controllable unsupervised generative adversarial network architecture
- Disentangled the content and style in an unsupervised fashion
- Applications in image style transfer, attribute manipulation, domain transfer, etc. without label supervision

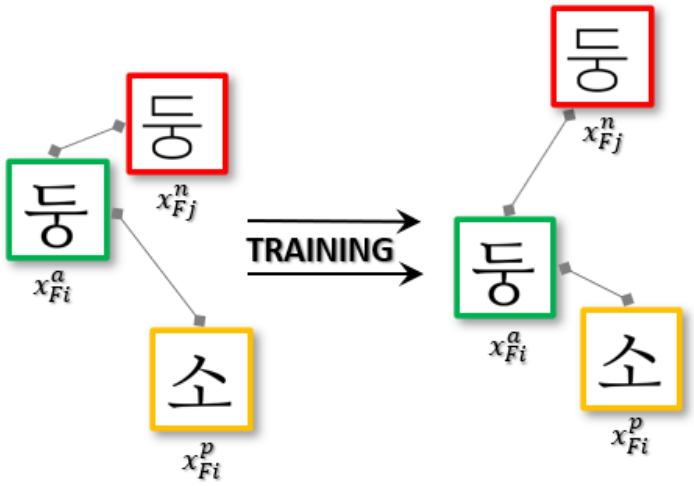
Programming Language

- PyTorch

Publication

- Under review in Pattern Recognition Letters

2. FEW-SHOT FONT GENERATION



Highlights

- Developed Metric learning and Contrastive learning-based network architectures
    - Learning font style latent space for few-shot font generation
- Component-guided Korean and Chinese font generation algorithms
- Applications in text image editing, font library creation, cross-lingual font generation

Programming Language

- PyTorch
- TensorFlow

Publication

- **Hassan, A. U.,** and Choi, J., "Fontnet: Closing the gap to font designer performance in font synthesis," AI for Content Creation (AI4CC), CVPR, 2022.

- **Hassan, A. U.,** and Choi, J., Under Review, AAAI, 2023.

- Park, J., **Hassan, A. U.,** Choi, J., "CCFont: Component-Based Chinese Font Generation Model Using Generative Adversarial Networks (GANs) ," MDPI Applied Science, 2022.

- Park, J., **Hassan, A. U.,** Choi, J., "Few-Shot Korean Font Generation based on Hangul Composability," KTSDE, 2022.

3. FONT FAMILY GENERATION

Highlights

- Font family data collection, preprocessing, and labeling
- Developed a generative model for real-time font family generation
- Applications in Variable font for typeface variations

Programming Language

- PyTorch

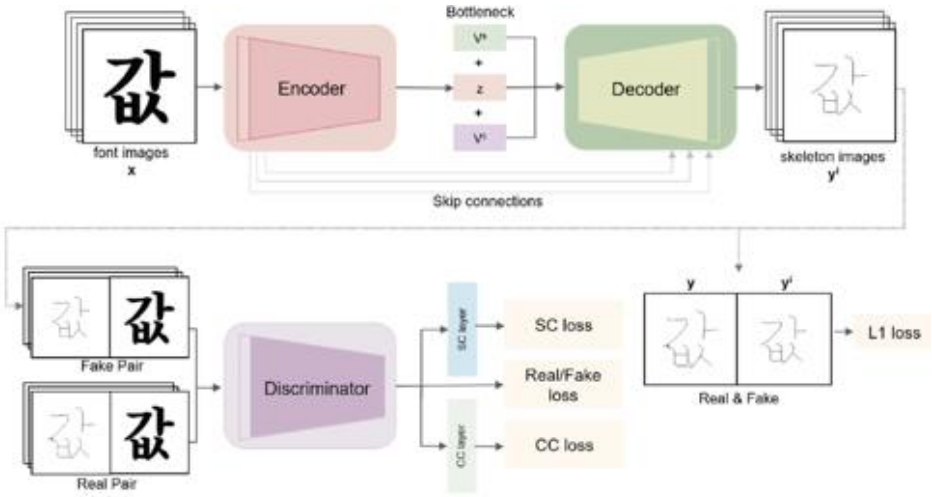
Publication

- **Hassan, A. U**., Ahmed, H., and Choi, J., "Unpaired font family synthesis using conditional generative adversarial networks," Knowledge-Based Systems, 229, 107304. https://doi.org/10.1016/j.knosys.2021.107304. (2021)

Patent

- **Hassan, A. U.,** and Choi, J., "METHOD AND APPARATUS FOR GENERATING FONT FAMILY USING DEEP LEARNNING,", Soongsil University Industry-Academic Cooperation Foundation, Patent, No. 2-2006-027849-9 (Korea), 2022.

4. TEXT IMAGE SKELETONIZATION



Highlights

- Character image skeletonization using an end-to-end generative adversarial network (GAN)
- Developed Skeleton-driven Korean font synthesis model
- Applications in object representation, manipulation, tracking, recognition

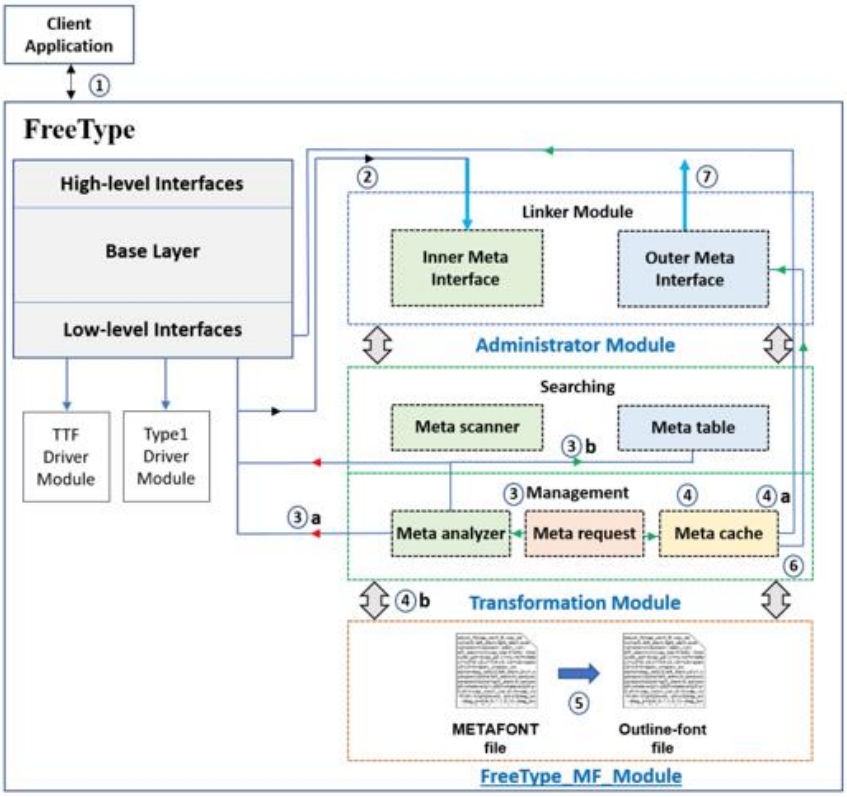
Programming Language

- TensorFlow
- Code SkelGAN, Code SKFont

Publication

- Ko, D. H., **Hassan, A. U**., Suk, J., and Choi, J., "SKFont: Skeleton-driven Korean font generator with conditional deep adversarial networks," International Journal on Document Analysis and Recognition (IJDAR), 1–13. https://doi.org/10.1007/s10032-021-00374-4
- Ko, D. H., **Hassan, A. U**., Majeed, S., and Choi, J., "Skelgan: A font image skeletonization method. Journal of Information Processing Systems," Journal of Information Processing Systems, 17(1), 1–13.

5. METAFONT MODULE FOR FREETYPE RASTERIZER



Highlights

- Rasterized MetaFont in Linux operating system
- Integrated driver module of MetaFont in FreeType rasterizer

Programming Language

- C, Linux

Publication

- **Hassan, A. U**., Jeong. G., and Choi, J., “FreeType MF Module: A module for using METAFONT directly inside the FreeType rasterizer,” TUGboat, Volume 39 (2018), No. 2).