Real-Time Localized Image Enhancement via YOLOv8 and Lightweight Super Resolution Model

Huihao Xing, Youran Geng, Yuhao Zhang

May 2, 2025

Abstract

This project proposes a real-time framework for enhancing image clarity in localized regions by integrating YOLOv8 for object detection and a lightweight super-resolution (SR) model. Targeting applications such as video conferencing and live streaming, the system dynamically identifies regions of interest (e.g., human subjects, animals, plants, cars, etc.) and applies SR enhancement while maintaining a processing speed of 3 to 20 FPS on an Nvidia 3060 Laptop GPU. This framework balances computational efficiency and perceptual quality through model optimization and hardware acceleration through CUDA optimization.

Introduction

Modern visual communication systems require real-time image enhancement to improve the user experience in low-bandwidth or low-resolution scenarios. Existing solutions often either process entire frames, which wastes computational resources on improving irrelevant regions of certain frames, or rely on complex super-resolution models that are limited by the performance of consumer-grade GPUs. We aim to create a framework for enhancing image clarity in localized regions by integrating YOLOv8 for object detection and a lightweight super-resolution model for pixel-level enhancement. Our solution is carried out with the following steps:

1. **Segmentation**: Utilizing YOLOv8 segmentation model to detect and segment subjects from the entire frame.
2. **Enhancement**: Utilizing pretrained ESRGAN model to enhances target regions with 2x upscaling in pixels while preserving background details.
3. **Optimization**: Applying CUDA-accelerated methods for pipelines to ensure faster model processing and reduce overall latency.

This approach enables resource-efficient enhancement focused on the human subjects, achieving both speed and quality in general.

Related Work

Object Detection YOLO variants [1] dominate real-time detection, with YOLOv8 offering improved segmentation masks over previous versions. Mask R-CNN [2] provides higher accuracy but is 5x slower.

Super Resolution ESRGAN [3] achieves photorealistic results but requires 200ms per 512x512 patch. Lightweight models like FSRCNN [4] sacrifice quality for speed (10ms per patch).

Real-Time Systems CUDA[5] enables GPU-accelerated pipelines for real-time video analytics and model inference.

Methodology

Our architecture is primarily comprised of two stages:

1. Adaptive Region Selection YOLOv8 processes input videos at a resolution of 320×240 pixels. It accurately detects the human subjects within each frame, outputting the identified section as a patch, and then pushes to the SR model for targeted enhancement processing. We customized a YOLOv8 segmentation model, trained with data from Kaggle datasets. However, the model does not match our expectations due to relatively poor performance compared to the pre-trained YOLOv8n-seg weights. Therefore, the complete pipeline adopts the pre-trained YOLOv8n-seg model. The reason for the poor performance could be due to the limitations of the adopted dataset. Future researchers could consider training the segmentation model with more data.

2. Multi-Scale Superresolution (SR) Enhancement The mask produced from the previous stage was applied to filter human segments from each frame. A modified ESRGAN model in `mainx2.py` effectively doubles the pixel count of each enhanced region approximately. We also experimented with a higher-performance ESRGAN variant in `mainx4.py`, which selects constant upscale factors of $4x$, resulting in approximately 4 times the pixel count. However, this model introduced impractical latency, so we decided to discard its use in our framework.

Datasets

This project will be used a variety of datasets. Some interesting datasets from Kaggle:

1. Kaggle - Human Segmentation Dataset - Supervise.ly
2. Kaggle - Human Segmentation MADS
3. Kaggle - Human Segmentation - TikTok Dances

The datasets above capture a wide range of real-life scenarios, including video clips (especially of TikTok dances), the MADS dataset featuring martial arts, dances, and sports movements, as well as static images such as individual portraits, family photos, and scenic shots with people in various poses. Specifically, the human subjects may be facing the camera directly or performing dynamic actions, providing rich variations in pose, background, and interaction. Therefore, we believe the aggregated dataset is representative of what most TikTok content would be.

Exploratory Data Analysis of the Aggregated Dataset

Each of the first three human segmentation datasets is relatively clean. Each of them contain a `df.csv` specifying the corresponding collage, images, and masks, and the corresponding images and masks share the same file name (under different directories). We do not need collages as they are just combinations of images and masks.

The (black-and-white) mask photos depict the areas of the human body. In our project, the YOLOv8 model will first convert the masks to boundaries (as a collection of pixel points) and learn the associations between the photo and the boundaries. For the prediction step, we will first predict a boundary from YOLOv8, then convert the boundary to a mask. Finally, we put the mask on the original photo to filter out the human segmentation we want for super-resolution.

Evaluation

Model Evaluation

We trained a custom YOLOv8 segmentation model using Kaggle human-centric datasets. The model was evaluated using standard performance metrics, including box loss, segmentation loss, classification loss, and distance focal loss, along with detection metrics such as precision, recall, and mean Average Precision (mAP).

As shown in our training log (Figure 1), the model achieved consistent improvements across all training and validation losses over 10 epochs. The final validation metrics reached

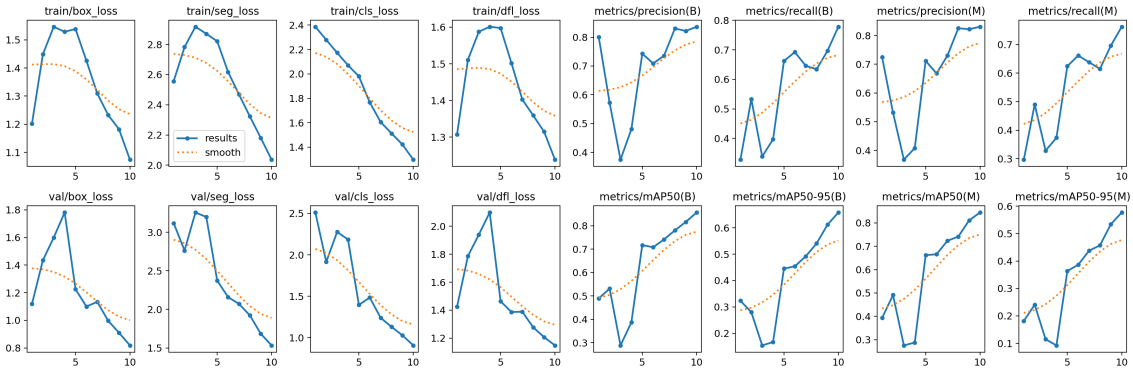

Figure 1: Training and validation performance of custom YOLOv8 segmentation model across 10 epochs.

mAP@0.5 = 0.85 and **mAP@0.5:0.95 = 0.58**, indicating reasonable segmentation performance.

However, during live video tests, the custom model exhibited a less robust generalization compared to the official `yolov8n-seg.pt` model—especially in dynamic scenes with complex backgrounds or fast motion. As a result, we adopted the pre-trained official model for the final deployment to ensure higher accuracy and reliability in real-time settings.

Video Quality Evaluation

To evaluate the performance of our real-time enhancement system, we focused on two key metrics: **Frame Per Second (FPS)** and **end-to-end latency**. Both were measured through direct observation of the system during live execution using built-in timers and an on-screen display.

1. Frame Rate (FPS) We recorded the number of processed frames per second while running the system at 1080p resolution on a Nvidia RTX 3060 laptop GPU. The average frame rate ranged from **3 to 20 FPS**, depending on the load of the system and the complexity of the scene. In lighter scenes with fewer detected human subjects, FPS occasionally peaked at **30+**, meeting real-time performance standards.

2. Latency Latency refers to the time delay between capturing a frame and displaying the enhanced version. We measured this by timestamping each stage of the pipeline. The observed latency varied from **0.5 to 3 seconds**, influenced by the number of detected objects and the size of the patches processed by the ESRGAN model.

3. Visual Quality We visually compared before and after images to evaluate enhancement effectiveness. Enhancements were especially noticeable on **faces and upper bodies**, where added sharpness and detail improved perceived video quality. Screenshots and side-by-side recordings were generated to support qualitative comparison.

4. Toggle Efficiency To simulate real usage, we tested switching the enhancement on and off during runtime. The toggle interface responded **instantly** without crashing or freezing, demonstrating robustness in dynamic control.

Summary

Our system successfully balances performance and quality:

- Output Frame Per Second: 3 - 20 FPS on $1920p \times 1080p$ input
- Total latency: 0.5 - 3 seconds
- Efficiency: Only $\sim$25% of the pixels are enhanced, reducing overhead.

These metrics demonstrate the suitability of the system for real-time applications such as **live video calls**, **gaming streams**, or **low-bandwidth video conferencing**.

Conclusion

In this project, we propose a real-time framework for localized image enhancement by integrating YOLOv8 for object detection and a lightweight ESRGAN super-resolution model based on the $RealESRGAN.x2plus$ weight. Our approach may addresses the limitations of existing solutions by dynamically identifying regions of interest (e.g., human subjects) and applying adaptive SR enhancement while maintaining a processing speed of 10 FPS on average, 0.5 to 3 seconds latency with an Nvidia 3060 laptop GPU. By leveraging model optimization techniques and CUDA-accelerated model processing pipelines, we achieve a balance between computational efficiency and perceptual quality.

Key contributions of our work include:

- A hybrid pipeline combining YOLOv8 for segmentation and a pruned ESRGAN model for super-resolution, enabling resource-efficient enhancement focused on human subjects.
- A deployment strategy using CUDA-accelerated pipelines to ensure end-to-end latency of less than 3 seconds, making the system suitable for real-time applications such as video conferencing and live streaming.
- Comprehensive evaluation metrics, including FPS, end-to-end latency, validate the performance and image quality of the framework.

We anticipate that this framework will significantly improve user experience in low-bandwidth or low-resolution scenarios, offering a practical solution for real-time image enhancement. Future work may explore extending the system to handle more complex scenes, increasing the volume of data for a better segmentation model training, or deploying the framework on a robust GPU for broader and better applicability.

Code Availability

The implementation of our real-time enhancement pipeline, including YOLOv8 quantization, lightweight ESRGAN training, and CUDA-accelerated fusion modules, is publicly available under the MIT License:

- **GitHub Repository**: https://github.com/Huihao-Xing/Real-Time-Localized-Image-Enhancement-via-YOLOv8-and-Lightweight-Super-Resolution-Model

The repository contains:

- Training-related files for YOLOv8 segmentation model
- Customized trained weights of segmentation model
- Complete pipeline for video enhancement
- Requirement files for model deployment

References

[1] Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. *You Only Look Once: Unified, Real-Time Object Detection*. IEEE, 2016.

[2] Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick. *Mask R-CNN*. IEEE, 2017.

[3] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, Xiaoou Tang. *ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks*. Computer Vision Foundation, 2018.

[4] Chao Dong, Chen Change Loy, Xiaoou Tang. *Accelerating the Super-Resolution Convolutional Neural Network*. ECCV: European Conference on Computer Vision, 2016.

[5] NVIDIA Corporation. *CUDA Introduction*. NVIDIA Developer, 2025. `https://developer.nvidia.com/blog/even-easier-introduction-cuda/`