國立高雄科技大學
National Kaohsiung University of Science and Technology

Foxconn Industrial Internet

DEPARTMENT OF ELECTRONIC ENGINEERING
COLLEGE OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Decode barcode image using YOLOv8 & REAL-ESRGAN

陳文俊 Tran Van Tuan

Tran Van Tuan (陳文俊)
National Kaohsiung University of Science and Technology
Department of Electronic Engineering – Jiangong Campus
Address: No. 415, Jiangong Rd., Kaohsiung City 80778, Taiwan (R.O.C.)
Student ID: F111169109
E-mail: f111169109@nkust.edu.tw, trantuan22052k@gmail.com
Line ID: 0862924814
Github: https://github.com/chwenjun225

Single Image Super-Resolution (SR)

Most approaches (such as ESRGAN) assume an ideal Bicubic down-sampling kernel, which is different from real degradations.

- ESRGAN methods can’t restore the real-life image.
- Can’t remove unknown blurs, complicated noises, and common compression artifacts

-> Real-ESRGAN is the extension of ESRGAN to practical restoration application

Challenges

- Unknow and complex degradations:
  - Usually, **paired training data** with similar degradations to real scenarios is required to train the networks.

Capture paired data with specific cameras followed by alignments

e.g., RealSR[1]

Directly learn degradation distributions and then synthesize paired training data

e.g., Cycle-in-Cycle GAN[2]

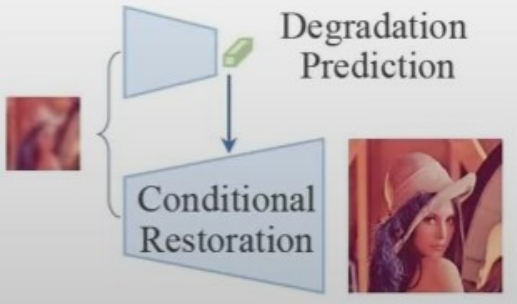
Synthesize paired data with classical operators and generalize trained models to real degradations

As close to real data as possible

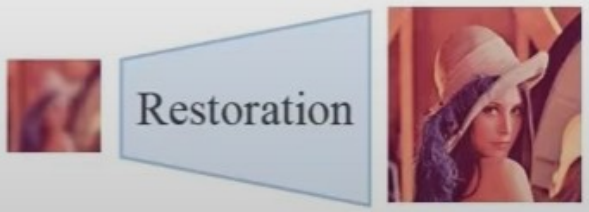
- Deal with diverse degraded images in one unified network.

Typical blind SR methods: two-branch network

e.g., DAN[3]



Process all degraded images in one network

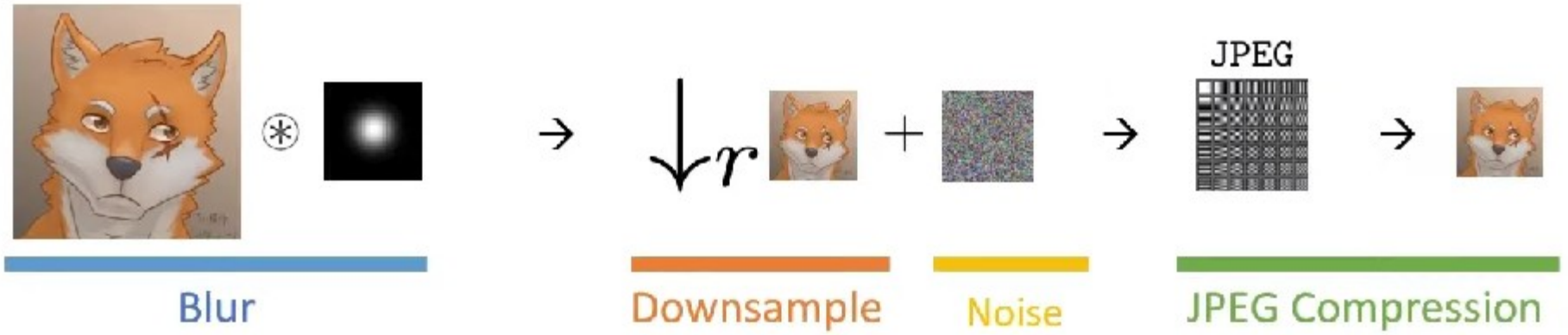


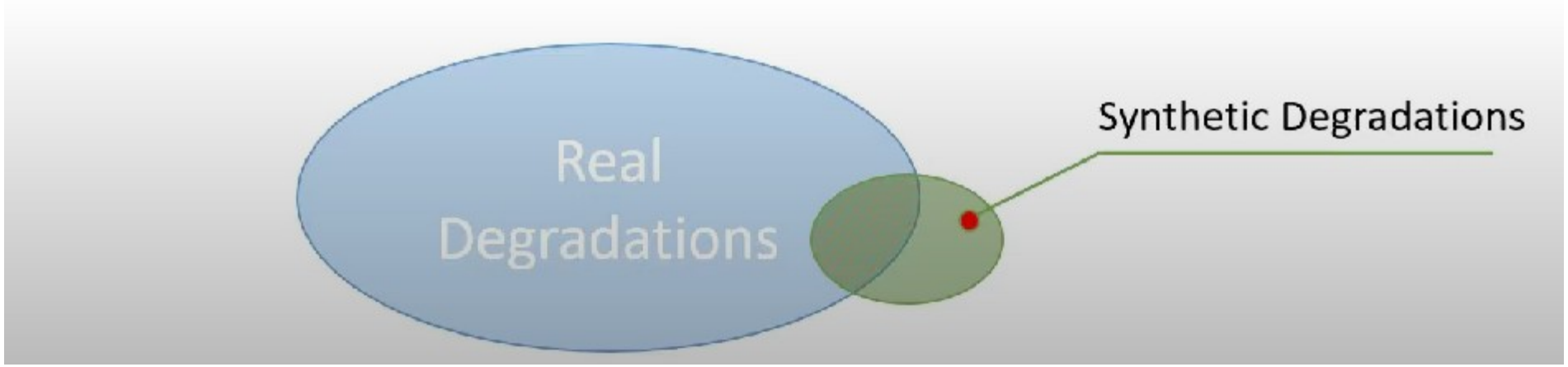
[1] Cai J, Zeng H, et al. Toward real-world single-image super-resolution. A new benchmark and a new model. ICCV 2019.
[2] Yuan Y, Liu S, Zhang J, et al. Unsupervised image super-resolution using cycle-incycle GAN. CVPRW 2018.
[3] Luo J, Huang Y, Li S, et al. Unfolding the alternating optimization for blind super-resolution. NeurIPS 2020.
[4] Xie L, Wang X, Dong C, et al. Finding Discriminative Filters for Specific Degradations in Blind Super-Resolution. NeurIPS 2021.

Finding Discriminative Filters for Specific Degradations in Blind Super-Resolution
Published on NeurIPS 2021

Classical Degradation Model

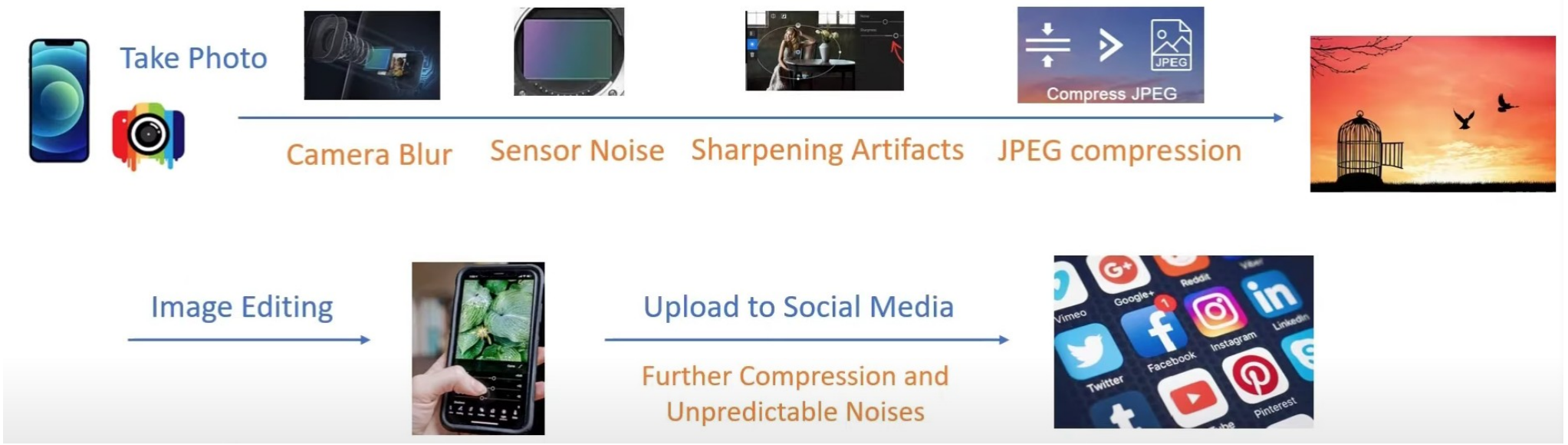
$$\boldsymbol{x} = [(\boldsymbol{y} \circledast \boldsymbol{k}_\sigma)\downarrow_r + \boldsymbol{n}_\delta]_{\text{JPEG}_q}$$





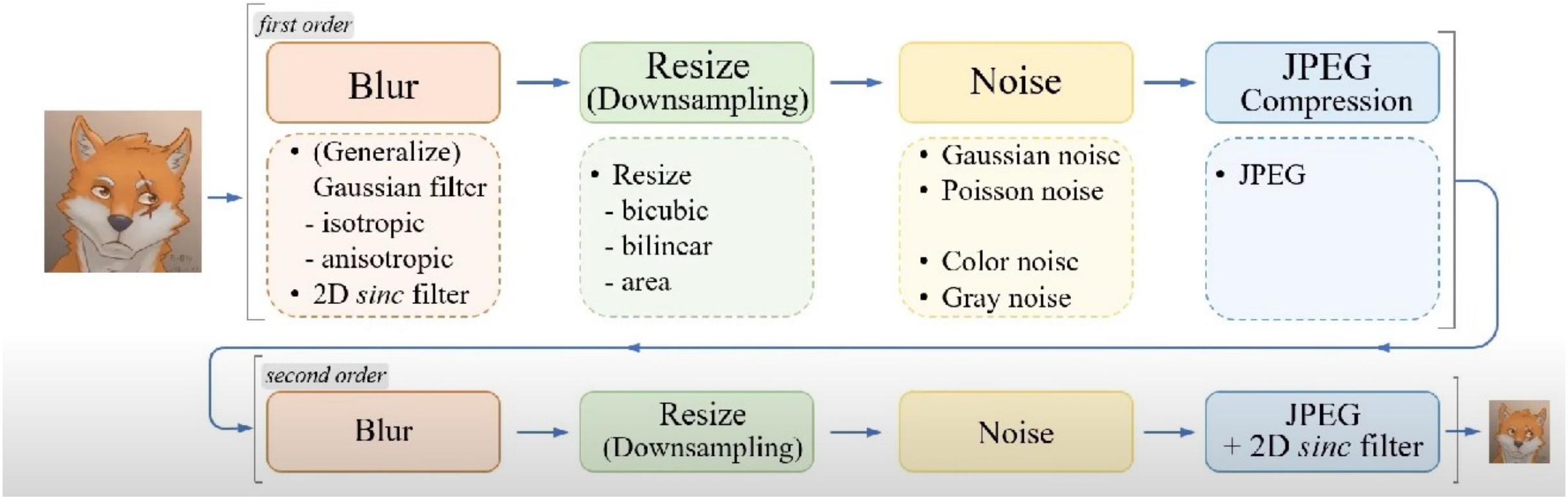
Complicated Combinations of Degradation Processes

The real complex degradations usually come from complicated combinations of different degradation processes, such as imaging system of cameras, image editing, and internet transmission.



Shared several times on the Internet

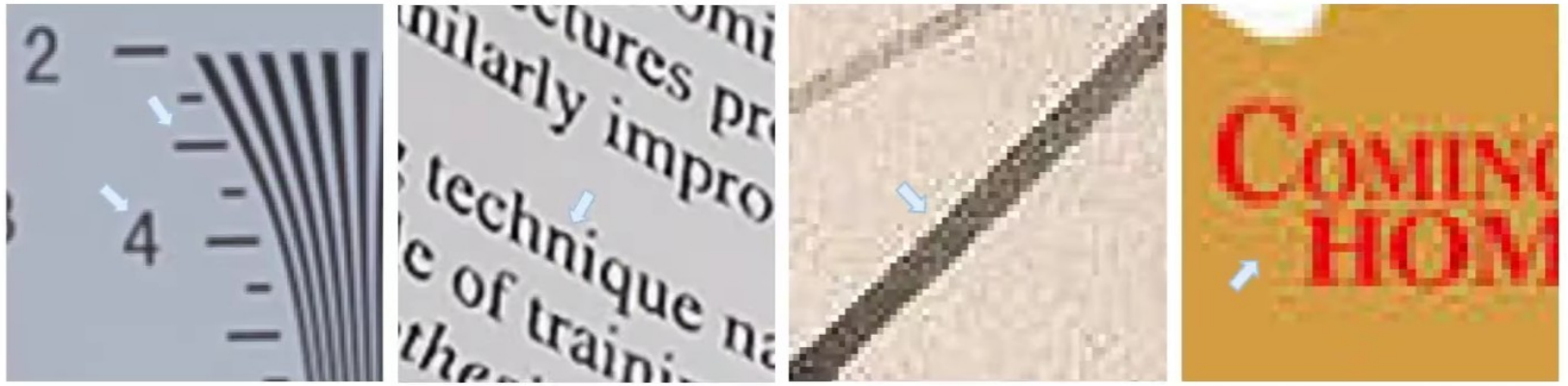
High-Order Degradation Process

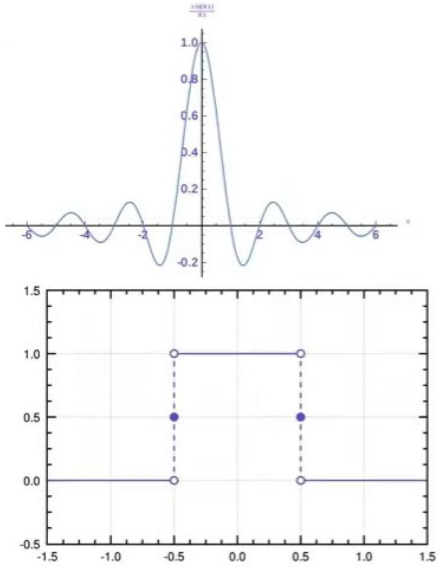


* The “high-order” here is different from that used in mathematical functions. Mainly refers to the implementation time of the same operation.

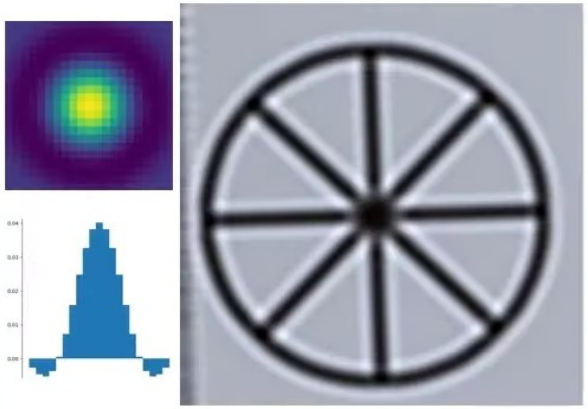
“Sinc” Filter for Ringing and Over-Shoot Artifacts

Real Samples

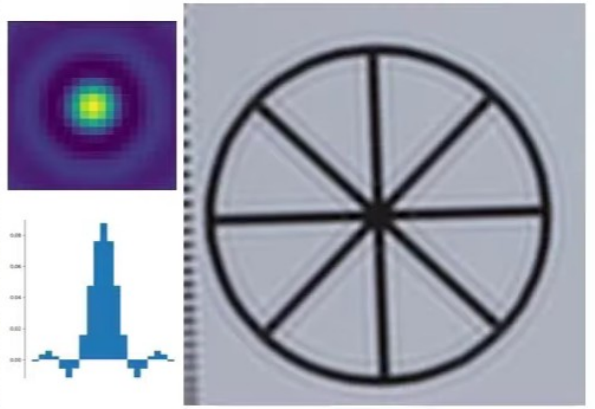




Input



$\omega_c = \pi/5$

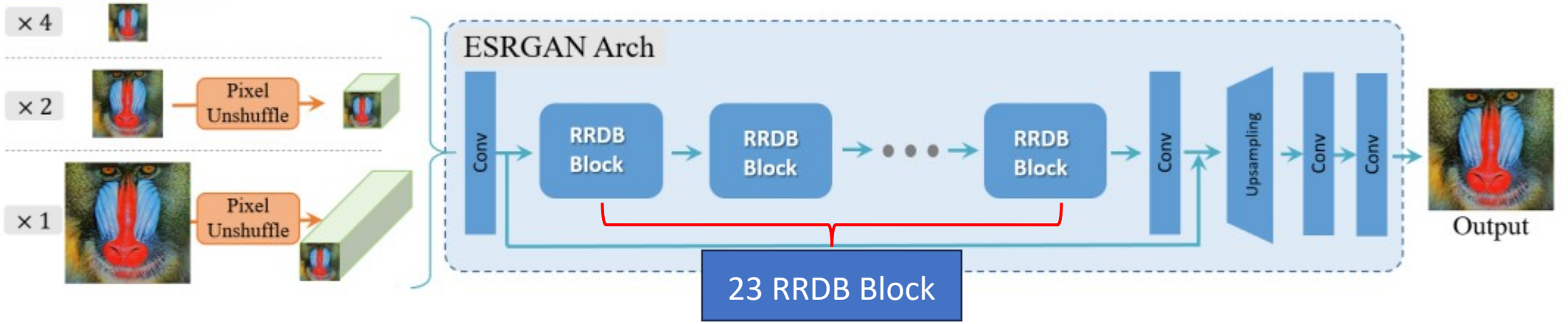


$\omega_c = \pi/3$

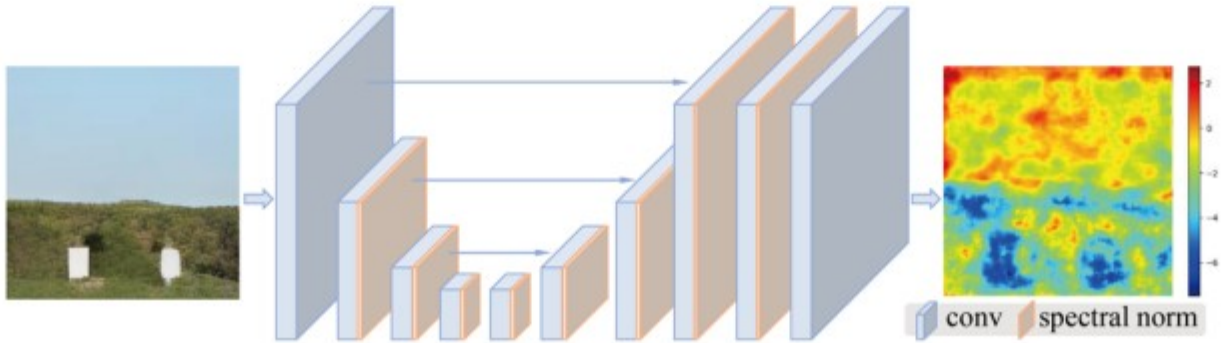
➢ *Network Architecture*

- We employ the same generator architecture as ESRGAN
- U-Net discriminator with spectral normalization is used to increase discriminator capability and stabilize the training dynamics

Generator



Discriminator

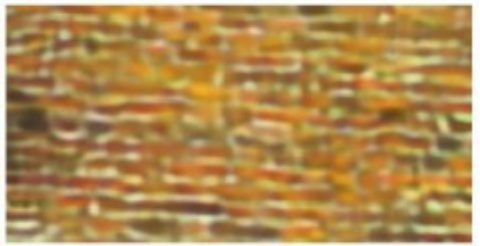


➢ Qualitative Comparisons

Bicubic

ESRGAN

DAN



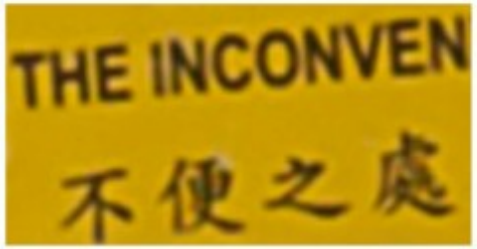
CDC

RealSR

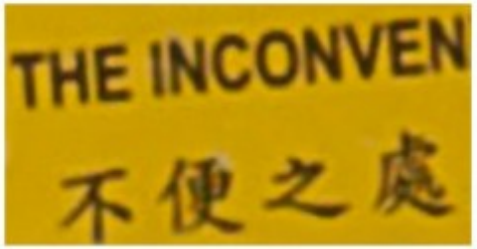
BSRGAN

Real-ESRGAN

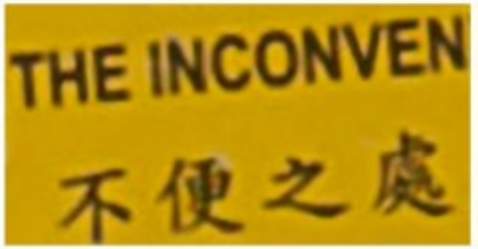
Real-ESRGAN+



Bicubic



ESRGAN



DAN

CDC

RealSR

BSRGAN

Real-ESRGAN

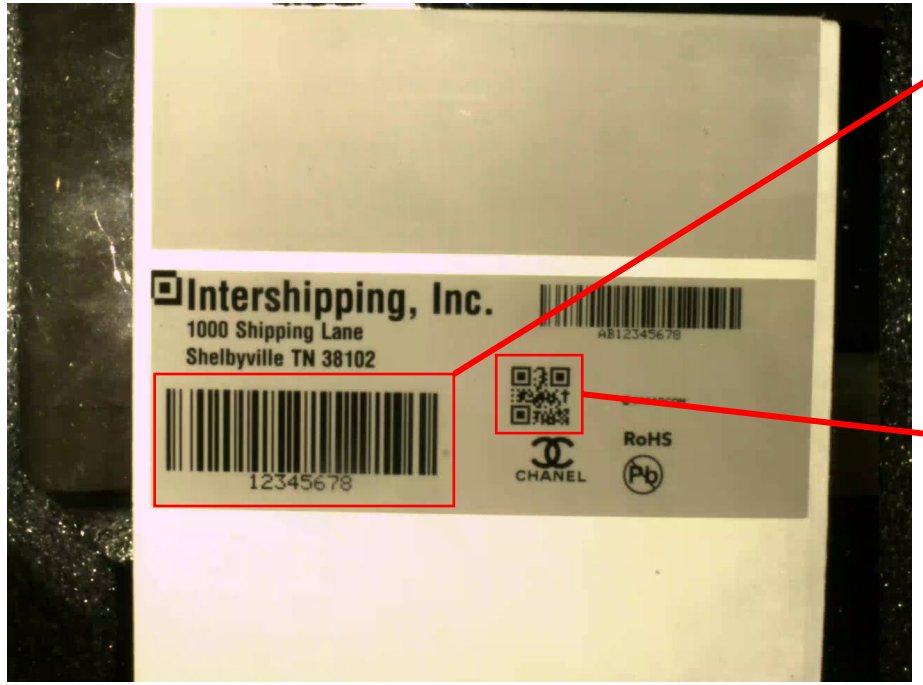
Real-ESRGAN+

➢ Optimize for Anime Images

Deploy on YOLOv8 and REAL-ESRGAN on a practical data

System workflow

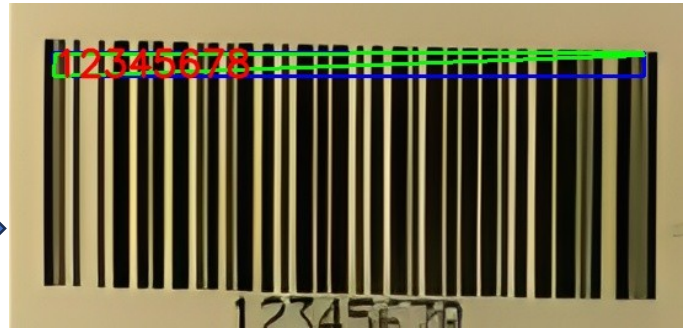
Image resized 416x416 → Yolov8 detected barcodes image → Real-ESRGAN restores the image barcodes → Use Pyzbar libraries to decode restored Image barcodes



137x66

X4

548x264

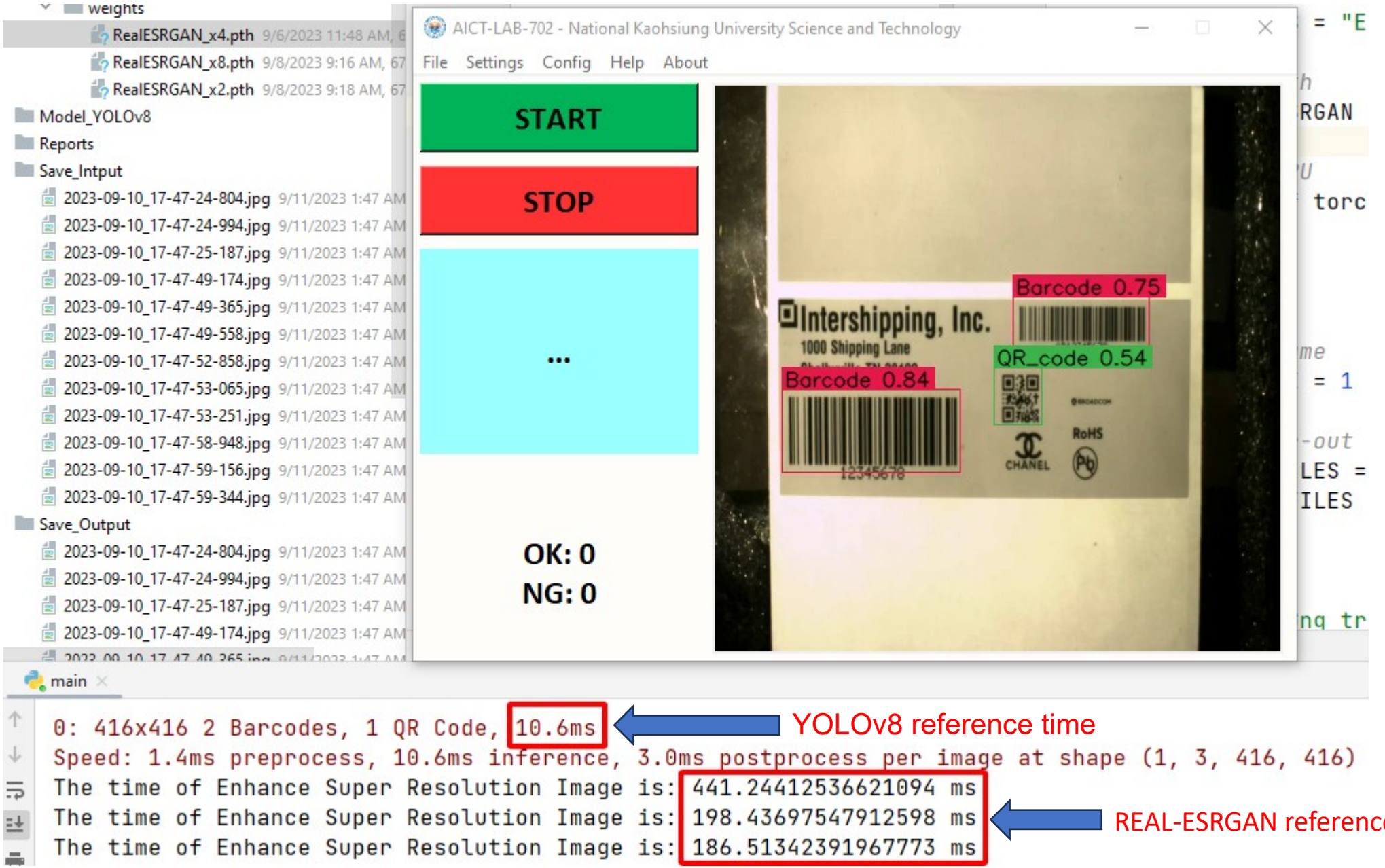


37x46

X4

148x184

System workflow, deploy and test on practical data from Honhai Factory



GitHub

- My project is now available on Github 🔥: Link
- YOLOv8 🚀: link
- Real-ESRGAN 🚀: link
- Fast deployment Real-ESRGAN 🔥: link
- For easy integrated Real-ESRGAN on your project 🎁: link

Thank you for listening