# Lab2
# Super Resolution

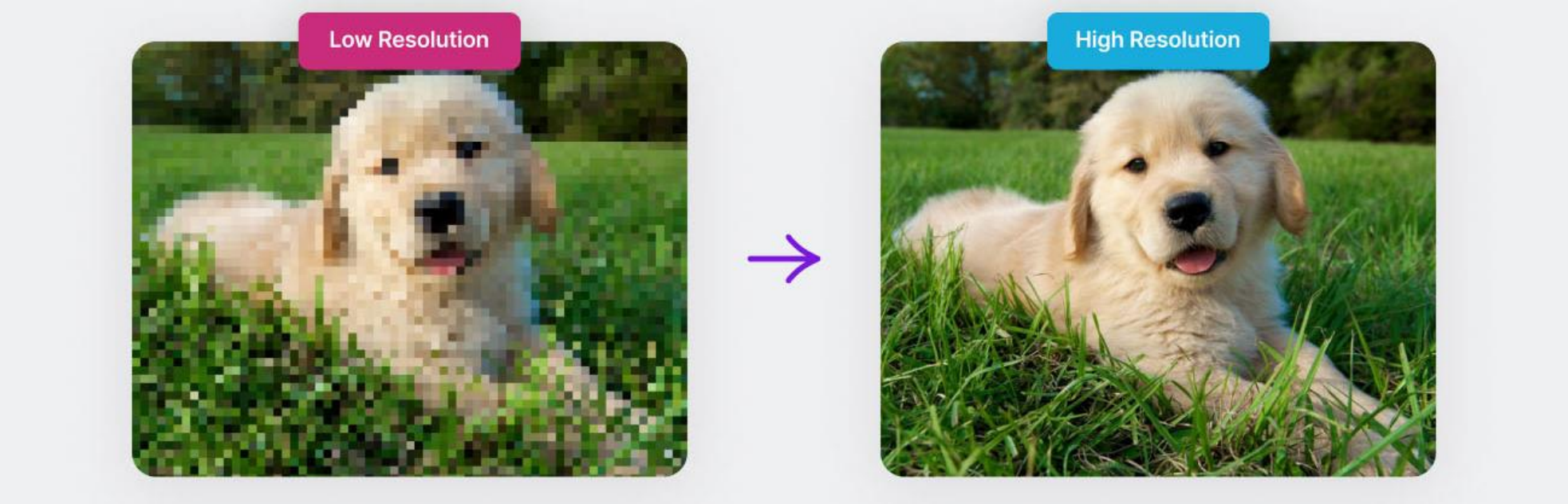
# Pytorch tutorial

- Official tutorial
  - https://pytorch.org/tutorials/
- 莫凡
  - https://mofanpy.com/tutorials/machine-learning/torch/
- AssemblyAI - PyTorch Crash Course
  - https://www.youtube.com/watch?v=OIenNRt2bjg

You can only use pytorch in this Lab!!

# Super Resolution

- Super Resolution is a technique in computer vision aimed at increasing the resolution of an image, i.e., enhancing the quality of a low-resolution (LR) image to generate a high-resolution (HR) image.
- Applications: NVIDIA DLSS/AMD FSR, Medical Imaging, …



# Dataset

- Super Resolution
- image size = (High resolution)256*256;
  (Low resolution) 64*64
- Training: 288  Validation: 72  Testing: 40

# Task1 of this lab

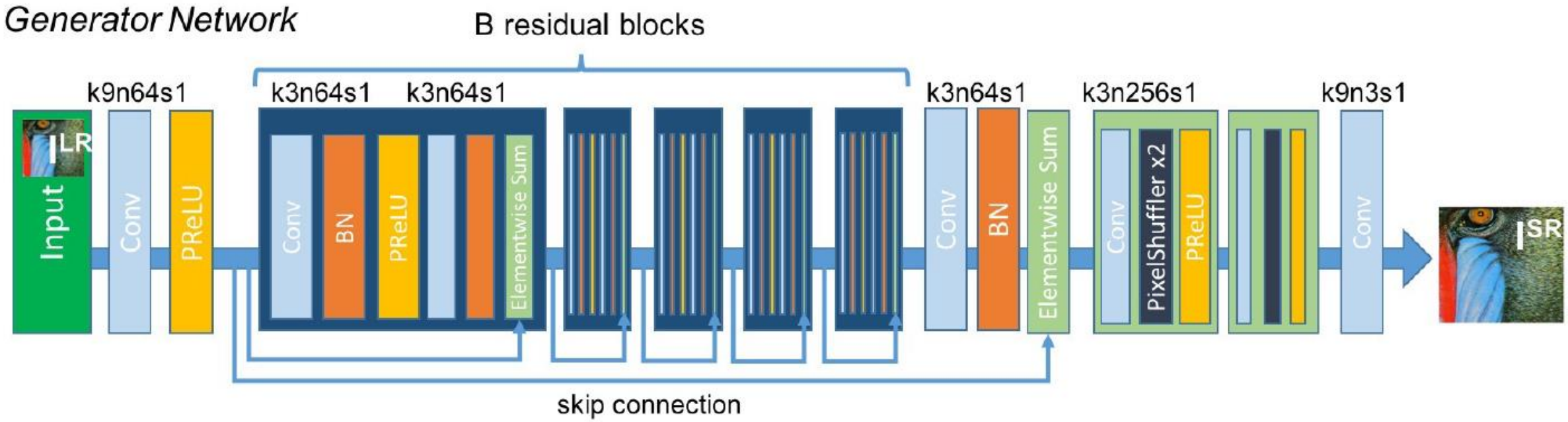
- In “Lab2_SRResNet.ipynb”
  - Build SRResNet by yourself

  (You can’t call the model directly with a command)
  - Achieve at least 21 dB PSNR on testing data

  (put the screenshot in your report)

# SRResNet

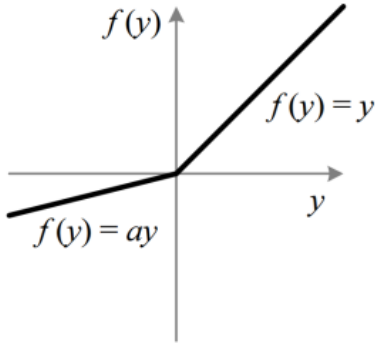
- SRResNet is inspired by the ResNet architecture and uses residual blocks for super-resolution tasks. It contains multiple residual blocks similar to those in ResNet, along with upsampling layers to enhance the resolution of input images.
- We use B = 16 in this lab.



# SRResNet

- PReLU(Parametric Rectified Linear Unit) is an activation function that extends the popular ReLU (Rectified Linear Unit) by making the negative slope a learned parameter rather than a constant. In PReLU, for negative inputs, the function is defined as:

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ a \cdot x & \text{if } x < 0 \end{cases}$$




- Where a is a learnable parameter, allowing the model to adapt the slope of the negative part during training, providing more flexibility in learning complex patterns.

# Task1 of this lab

- Finish the blank part.

MODEL

```
# using gpu if available
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")

#build your model here


#call model
model =
```

```
criterion = nn.MSELoss()
optimizer =

# num_epochs : You can try 10~15 at first.
num_epochs =
psnr_best = 0.0
for epoch in range(num_epochs):

    # Training phase
    running_loss = 0.0
    for data in train_loader:
        low_img, high_img = data['low_img'], data['high_img']
        # Please finish the "Training phase" code here.



    # Validation phase
    model.eval()  # Set the model to evaluation mode
    val_loss = 0
    psnr_total = 0
    with torch.no_grad():
        for lr_imgs, hr_imgs in val_loader:
            lr_imgs, hr_imgs = data['low_img'].to(device), data['high_img'].to(device)

            # Forward pass for validation
            # Please finish the "Validation phase" code here.


            # Calculate PSNR for validation images
            psnr_total += calculate_psnr(outputs, hr_imgs)
```

# Task2 of this lab

- In Task2
  - Do your best to improve the quality of the photos
  - Calling different models with pretrained weight is allowed
  - Basically any methods you learn are allowed
  - Achieve at least 23 dB PSNR on testing data (put the screenshot in your report)

# Report

- Write a report
  - Required
    - Screenshot of task-1 (PSNR on testing data >= 21 dB)
    - Screenshot of task-2 (PSNR on testing data >= 23 dB)
    - In task-2
      - What model do you choose？
      - The advantage of chosen model.
    - Explain what is PixelShuffle.
    - Explain what is PSNR and discuss why it is not the only metric used for evaluating super-resolution. And give some other metrics that provide different perspectives on image quality.
    - Anything you do to improve the quality of the output photos.
    - You can discuss any challenges you faced.

# Score

- PSNR on testing data in Task 1 >= 21 dB (30%)
- PSNR on testing data in Task 2 >= 23 dB (30%)
- Report (30%)
- Performance rank for Task-1 (10%)
  - Ranked based on PSNR on testing data in Task 1

- Please do not plagiarize (0 points will be calculated if caught)

# Reminder

- Submit Deadline : 2 week  (2024/10/7 11:59 PM)
- Upload 3 files to new e3
  - Lab02_SRResNet.ipynb
  - model.pth (of Task1)
  - StudentID_report.pdf (example: 311555555_report.pdf)

# Supplement: SRResNet

- **paper**
  - **https://arxiv.org/abs/1609.04802**

# HAVE FUN !!!