

## Báo cáo Project II

### Bài toán tăng độ phân giải cho ảnh

Giáo viên hướng dẫn: TS Đào Thành Chung  
Sinh viên: Nguyễn Trần Khang MSSV: 20183559

Hà Nội, tháng 6 năm 2021

# Mục lục

- I. Tóm tắt nội dung
- II. Giới thiệu
  - 1. MISR
  - 2. SISR
  - 3. Một số phương pháp SISR tổng quát
    - a. Phương pháp nội suy
    - b. Phương pháp đặc trưng cạnh
    - c. Phương pháp học sâu
- III. Chỉ số đánh giá
  - 1. PSNR
  - 2. SSIM
- IV. Phương pháp
  - 1. SRResnet
  - 2. SRGAN
- V. Cài đặt
- VI. Đánh giá
  - 1. Đánh giá SRResnet
    - a. Đánh giá số Residual Blocks
  - 2. Đánh giá SRGAN
    - a. Đánh giá số Residual Blocks
    - b. Đánh giá kỹ thuật BatchNorm
    - c. Đánh giá hàm mục tiêu mới
  - 3. Đánh giá tổng quan
  - 4. Một số kết quả
- VII. Kết luận
- VIII. Tài liệu tham khảo

# I. Tóm tắt nội dung

Ngày nay việc tiếp cận các dữ liệu dạng text và dạng hình ảnh là rất đa dạng và phong phú. Dữ liệu dạng ảnh nói riêng có thể bị thay đổi trong quá trình truyền tin, đa số các biến dạng gây nên việc giảm chất lượng hình ảnh.

Bài toán đặt ra xử lý vấn đề giảm chất lượng hình ảnh. Trong lĩnh vực này, có các lớp bài toán tăng độ phân giải cho hình ảnh (Image Super Resolution). Bài toán (ISR) có thể chia thành hai hướng. Một là Multiple Image Super Resolution(MISR), thứ hai là Single Image Super Resolution(SISR).

Trong đề tài lần này, em xin trình bày nghiên cứu về bài toán SISR. Các hướng tiếp cận bài toán theo hướng sử dụng thuật toán học sâu (Deep Learning), qua đó đánh giá kết quả và so sánh giữa các phương pháp.

## II. Giới thiệu

### 1. MISR

Bài toán MISR sử dụng thông tin nhiều bức ảnh khác nhau, có chất lượng thấp để tái tạo bức ảnh có chất lượng cao hơn. Những kỹ thuật này đã từng được thực hiện trong các máy ảnh kỹ thuật cao hoặc qua các chương trình xử lý ảnh như Photoshop. Tuy nhiên, nhược điểm của phương pháp này là cần rất nhiều ảnh độ phân giải thấp và chúng phải được căn chỉnh và đối tượng trong ảnh phải đứng yên trong thời gian chụp. Hơn nữa, tốc độ xử lý cũng không được cao.

### 2. SISR

Bài toán SISR là bài toán xuất phát từ một bức ảnh có chất lượng thấp và mục tiêu là sử dụng thông tin đó để tái tạo lại thành bức ảnh chất lượng cao hơn.

Trong đề tài lần này, em đi theo hướng SISR để xử lý bài toán đề ra. Các bài toán SISR lại chia ra thành hai hướng chính: Tiếp cận tổng quát và tiếp cận cho lĩnh vực cụ thể.

Trong tiếp cận theo lĩnh vực cụ thể, ảnh được xử lý thuộc một lĩnh vực cụ thể. Ví dụ như ảnh X-quang, ảnh chân dung, ảnh biển số xe, ... Các nghiên cứu tập trung vào các đặc tính vốn có của lĩnh vực để cố gắng tái tạo chất lượng ảnh.

Trong khi đó ở cách tiếp cận tổng quát, ảnh được xử lý có thể thuộc bất kì một lĩnh vực nào. Các phương pháp chủ yếu dựa trên đặc tính của bức ảnh, ví dụ như: cạnh, phân phối các màu sắc, khối, ...

### 3. Một số phương pháp SISR tổng quát

Các phương pháp tiếp cận dựa trên đặc trưng, đặc tính kể trên của ảnh, có thể kể ra như:

a. Phương pháp nội suy (Interpolation method):

Ý tưởng của phương pháp xuất phát từ giả thuyết: Các pixel trong cùng một vùng thì thường có chung giá trị pixel. Ảnh sinh ra sẽ có độ mịn, tuy nhiên ảnh sẽ không chính xác ở các vị trí như cạnh hoặc khối, do khi đó giả thuyết về pixel hàng xóm không còn đúng nữa.

b. Phương pháp đặc tính cạnh (Edge based methods):

Ý tưởng phương pháp là cố gắng xử lý các vùng pixel cạnh và khối. Ảnh sinh sẽ sắc nét hơn so với phương pháp nội suy. Tuy nhiên đây là phương pháp bù cho phương pháp nội suy. Phương pháp này sẽ có hạn chế khi gặp vùng pixel có độ mịn.

c. Phương pháp học sâu:

Gần đây, với sự phát triển của Deep Learning và cụ thể là Convolutional Neural Network (CNN), bài toán SISR đã được cải tiến và cho nhiều kết quả tiềm năng. Trong đề tài này em chọn hướng tiếp cận học sâu cho bài toán.

## III. Chỉ số đánh giá

Trong SISR, có hai chỉ số đánh giá hiệu quả của các thuật toán. Đó là chỉ số tín hiệu cực đại trên nhiễu (PSNR-Peak signal to noise ration) và chỉ số tương đồng cấu trúc (SSIM-Structural Similarity Index).

## 1. PSNR

PSNR dùng để tính tỉ lệ giữa giá trị năng lượng tối đa của một tính hiệu và năng lượng nhiễu ảnh hưởng đến độ chính xác thông tin.

$$PSNR = 10 \log_{10} \left( \frac{MAX}{MSE} \right) (dB)$$

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I(i, j) - K(i, j))^2$$

Với ảnh gốc I, ảnh sau khi học K, có kích thước m x n. MAX với B là số bits biểu diễn giá trị của một pixel. PSNR càng cao thì chất lượng ảnh càng gần với ảnh gốc.

## 2. SSIM

Giả thuyết rằng mắt người nhận biết sự khác nhau giữa các bức ảnh thông qua sự thay đổi cấu trúc. SSIM mô phỏng lại hoạt động thị giác, thông thường sự thay đổi tương đối sẽ nhạy cảm hơn so với sự thay đổi tuyệt đối ở bức ảnh.

SSIM tính toán dựa trên 3 thành phần khác nhau. Ánh sáng (I), tương phản (c), cấu trúc (s).

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

Với  $\mu_x$ ,  $\mu_y$  là trung bình của x, y tương ứng,  $\sigma_x^2$ ,  $\sigma_y^2$  là phương sai x, y tương ứng.  $\sigma_{xy}$  là hiệp phương sai của x và y. Các hằng số  $c_1$ ,  $c_2$ ,  $c_3$  là các hằng số dương. Kết hợp lại ta thu được công thức tổng quát cho độ đo.

$$SSIM(x, y) = l(x, y)^\alpha * c(x, y)^\beta * s(x, y)^\gamma$$

Để đơn giản, với  $\alpha = \beta = \gamma = 1$  và  $c_3 = c_2 / 2$ . Khi đó ta có công thức:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Độ đo SSIM mô tả sự tương đối trong khi PSNR hay MSE mô tả sự thay đổi tuyệt đối.

### 3. Ảnh YCbCr

Ảnh RGB thông thường, mỗi pixel sẽ được biểu diễn với 3 giá trị R, G, B tương ứng thang màu đỏ, xanh lá, xanh dương. Ảnh YCbCr có ba giá trị biểu diễn cho 1 pixel, trong đó Y tương ứng với giá trị độ sáng và CbCr là thang màu sắc xanh, đỏ. Theo các nghiên cứu về nhãn quang, mắt người chúng ta nhạy cảm với độ sáng hơn là với màu sắc. Do đó ảnh Ycbcr có ứng dụng trong truyền ảnh, khi người ta chỉ cần bảo toàn kênh màu Y có thể chịu mất mát trên CbCr. Việc chuyển đổi từ RGB sang YCbCr rất đơn giản, hoàn toàn là phép nhân hai ma trận.

Trong bài toán này, các độ đo sẽ được tính toán trên kênh màu Y.

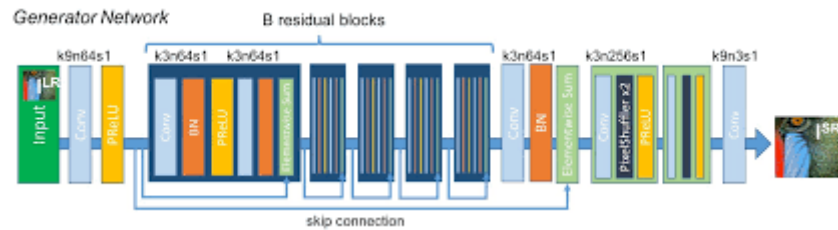
## IV. Phương pháp

Trong bài toán này, em sử dụng 2 mô hình trong theo hướng học sâu để tiến hành phân tích đánh giá. Đó là 2 mô hình SRResnet và SRGAN.

### 1. SRResnet

Mô hình Resnet là bước ngoặt trong các bài toán về xử lý thông tin dạng ảnh, vượt qua hiệu năng của con người trong lĩnh vực nhận diện, phân loại ảnh.

Resnet khác biệt so với các mạng CNN thông thường đó là ở các khối Residual Block(RB). Các RB này cải thiện vấn đề vanishing gradient trong học sâu. Cụ thể, đạo hàm cập nhật tại các weight có xu hướng bằng 0, tốc độ di chuyển chậm. Với việc sử dụng các RB, các thông tin đạo hàm được bảo toàn qua kĩ thuật Skip Connection.



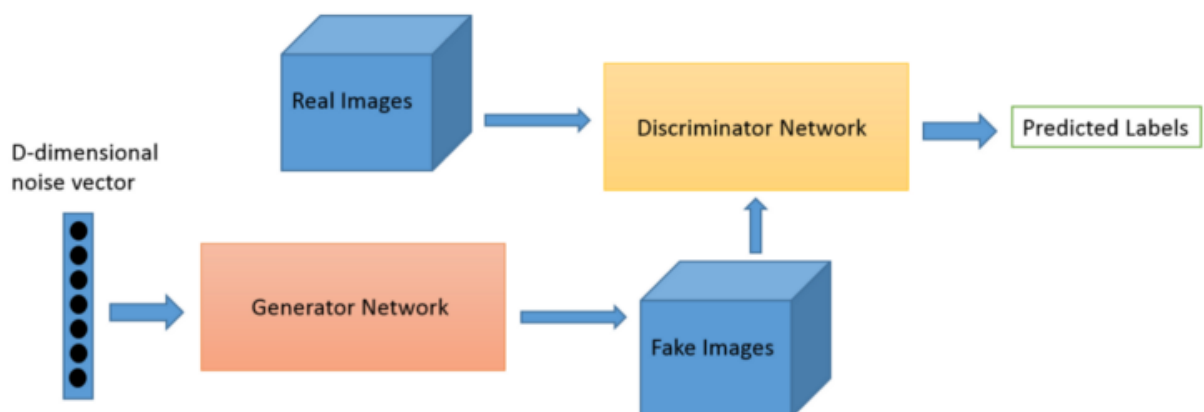
SRResnet là phiên bản áp dụng mạng Resnet vào bài toán SISR, trong đó đầu vào mạng là ảnh có độ phân giải thấp (LR), đầu ra là ảnh có độ phân giải cao (SR). SRResnet bao gồm các lớp Convolution, RB và pixel shuffle. Chi tiết kiến trúc được mô tả qua hình trên.

Hàm mục tiêu cho pha huấn luyện. Quá trình huấn luyện đi tối ưu sự sụt tái tạo lại bức ảnh HR từ ảnh LR.

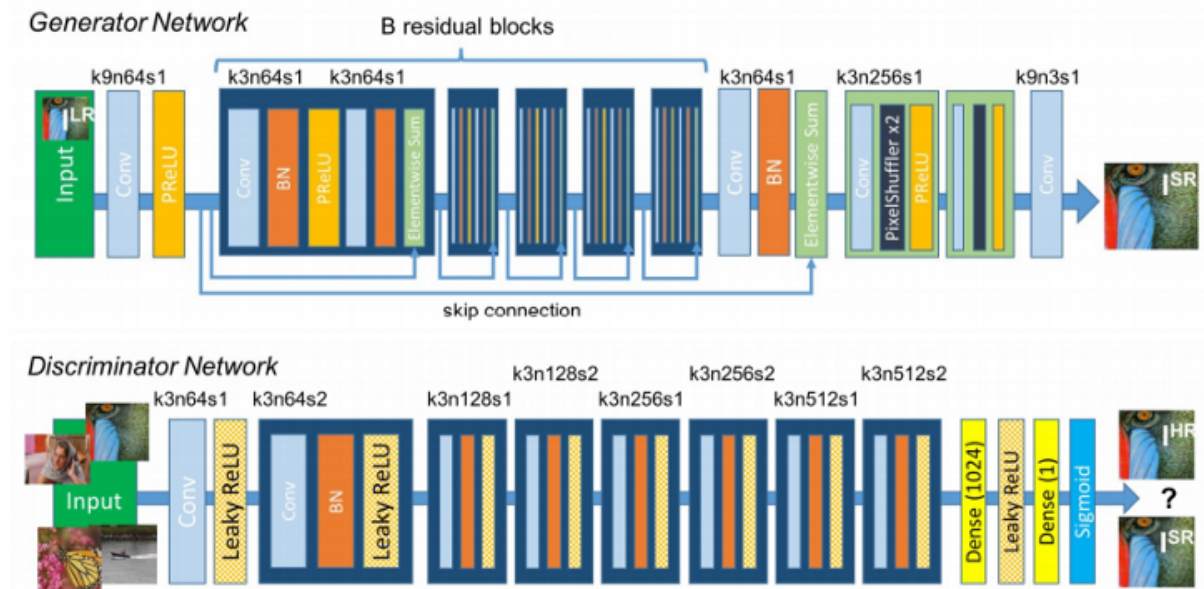
$$L_{Resnet} = L_{construct}(SRResnet(LR), HR)$$

## 2. SRGAN

Generative Adversarial Network(GAN) là mô hình học sâu theo kiểu đối kháng. GAN gồm 2 thành phần chính: Generator(G) và Discriminator (D). Trong đó G đóng vai trò trong việc sinh ra dữ liệu và D có vai trò phát hiện thật giả giữa dữ liệu gốc và dữ liệu sinh ra từ G.



Trong bài toán SISR, dựa trên ý tưởng của mạng đối kháng GAN, mô hình SRGAN ra đời. Ở mô hình SRGAN, vai trò của mạng G và D là gần giống với ở GAN. Cụ thể, G đóng vai trò nhận đầu vào là một ảnh LR và sinh ra ảnh SR giống thật nhất có thể; D sẽ cố gắng phát hiện đâu là ảnh SR và đâu là ảnh HR thật.



Hàm mục tiêu của quá trình huấn luyện mô hình SRGAN bao gồm hàm mục tiêu cho G và hàm mục tiêu cho D.

$$L_{gen} = \alpha L_{reconstruct} + \beta L_{vgg} + \gamma L_{adv} + \delta L_{tv}$$

$$L_{disc} = -L_{adv}$$

Trong đó

$$L_{reconstruct} = L1(SR, HR)$$

$$L_{vgg} = L1(VGG(SR), VGG(HR))$$

$$L_{adv} = \log D(HR) + \log(1 - D(SR))$$

$$L_{tv} = L_{tv}(SR)$$

với  $SR = G(LR)$ ,  $VGG$  là mạng CNN đã được tối ưu dùng để đánh giá các biểu diễn ở tầng ẩn. Các hệ số dương  $0 < \alpha, \beta, \gamma, \delta \leq 1$

Mục tiêu của hai mạng G và D là đi tối thiểu hóa hàm mục tiêu. Quá trình huấn luyện thể hiện sự đối kháng ở chỗ: G cố gắng giảm  $L_{adv}$  trong khi D cố gắng tăng  $L_{adv}$ , hay nói cách khác G cố gắng lừa D bằng cách sinh ra ảnh càng giống với ảnh thực càng tốt và D cố gắng phát hiện đâu là ảnh sinh ra đâu là ảnh thực.



## V. Cài đặt

Để thực nghiệm, em sử dụng các ngôn ngữ lập trình Python, thư viện Pytorch để tiến hành cài đặt. Trong đó Pytorch là thư viện mạnh mẽ về học sâu và có cộng đồng lớn mạnh nhất trong lĩnh vực này.

Dữ liệu sử dụng là các bộ VOC2012, BDS100, Set5, Set14. Trong đó tiến hành thí nghiệm điều chỉnh tham số trên bộ VOC2012 rút gọn với 300 ảnh huấn luyện và 100 ảnh đánh giá. Sau khi lựa chọn được tham số phù hợp cho 2 mô hình SRResnet và SRGAN, em tiến hành huấn luyện trên toàn bộ tập VOC2012 với 17125 ảnh và đánh giá trên các bộ BDS100, Set5, Set14.

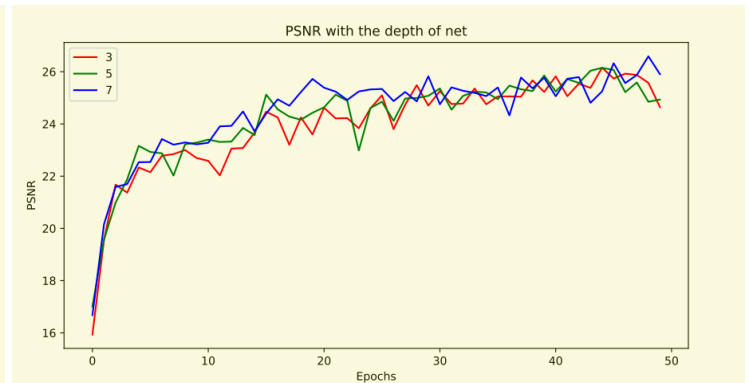
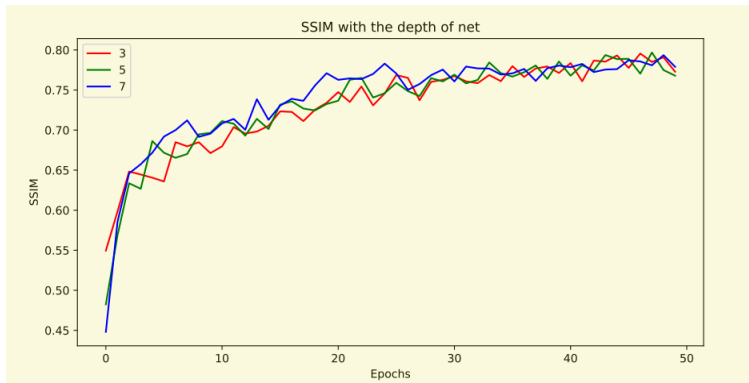
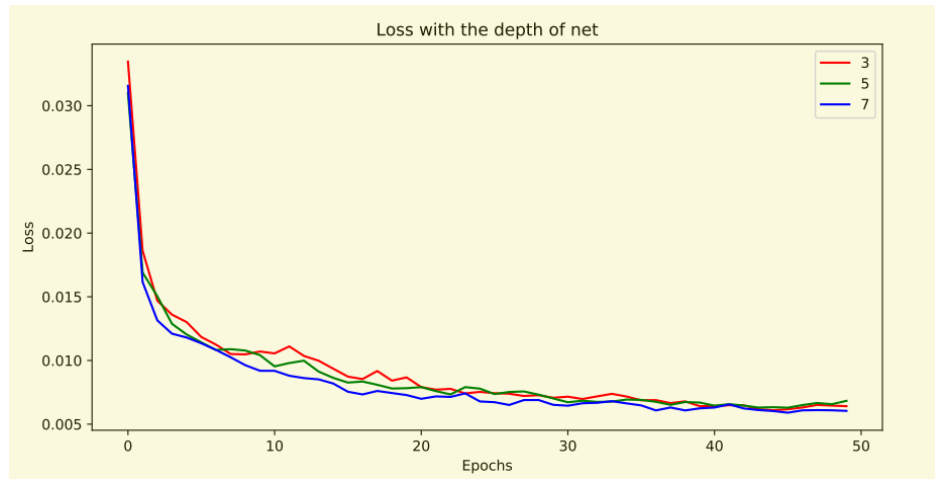
Trong suốt quá trình huấn luyện, dữ liệu được đưa về dạng cặp (LR, HR) với LR có kích thước (22x22) và HR là (88x88). Ảnh LR được tạo ra từ HR bằng phương pháp Bicubic với scale là 4. Cặp ảnh được chuẩn hóa về khoảng [0,1].

## VI. Đánh giá

### 1. SRResnet.

#### a. Đánh giá số RB.

Độ sâu của mô hình SRResnet phụ thuộc vào số lượng RB. Ở trong đánh giá lần này, em thử các giá trị 3, 5, 7.

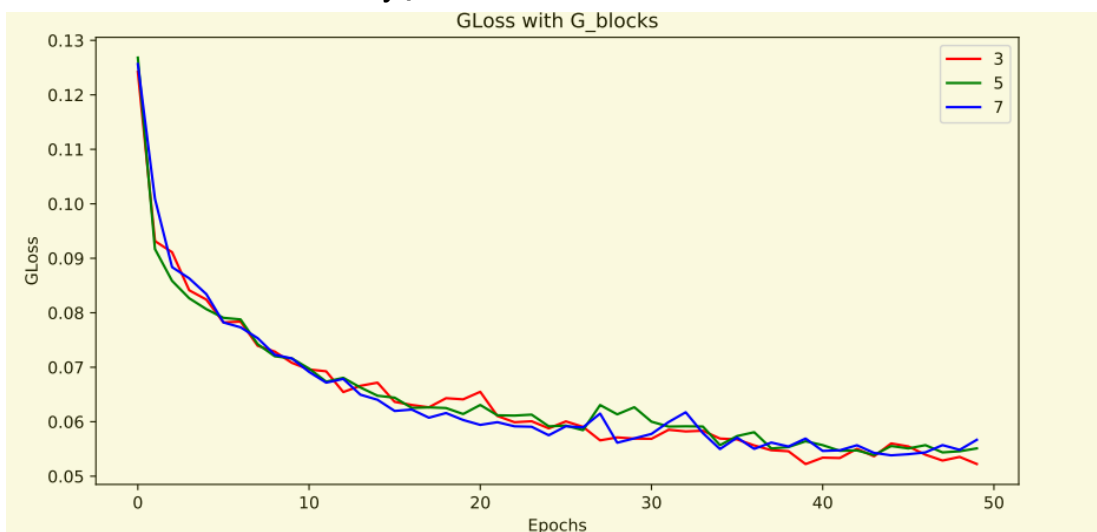


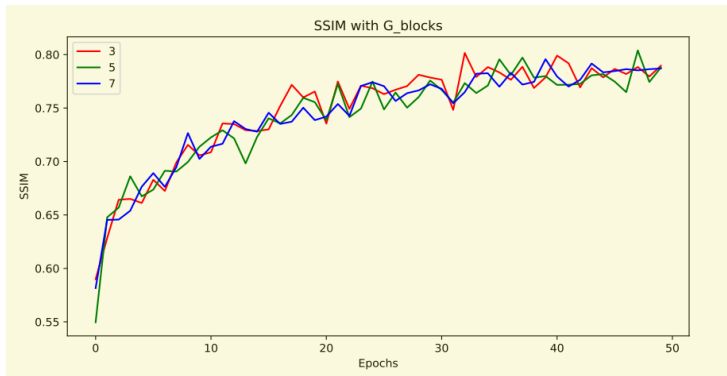
Qua hàm loss ta thấy, Cả 3 trường hợp cho cùng một khoảng giá trị. Tuy vậy ở hai chỉ số là SSIM và PSNR, với RB = 7 thì mô hình đạt được kết quả nhỉnh hơn.

## 2. SRGAN

### a. Đánh giá số RB ở Generator

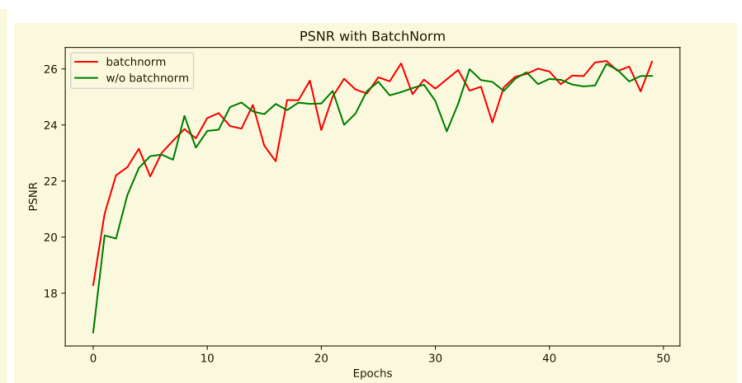
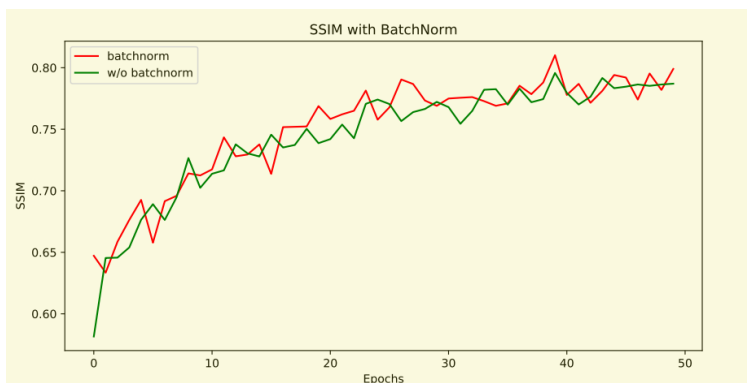
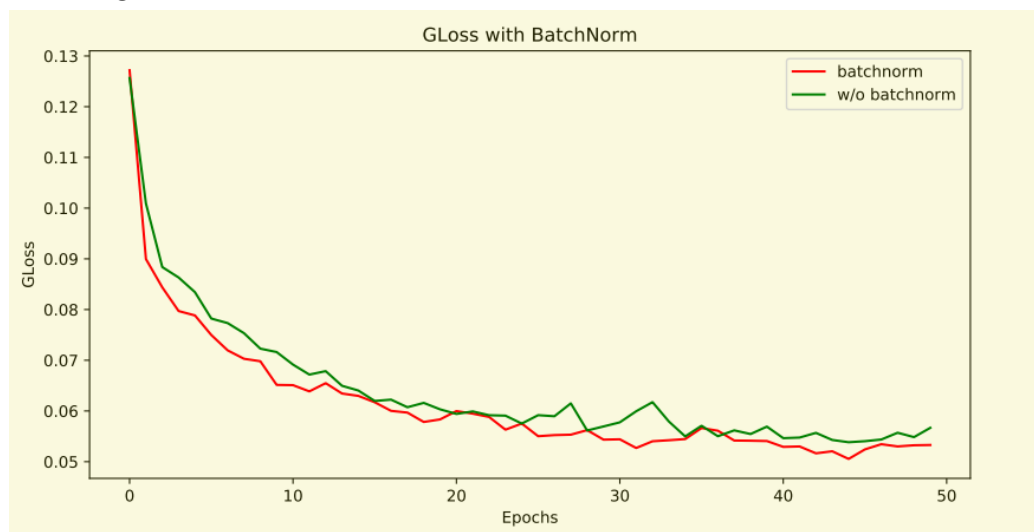
Cũng giống như mô hình SRResnet, ở SRGAN số lượng RB ảnh hưởng đến độ sâu của mô hình huấn luyện.





Với các giá trị 3, 5, 7, ta thấy rằng ở  $RB = 7$  mặc dù mô hình sâu hơn tuy nhiên do dữ liệu lựa chọn nhỏ nên kết quả không ổn định, mặc dù với 2 chỉ số đánh giá đạt giá trị tốt nhất.

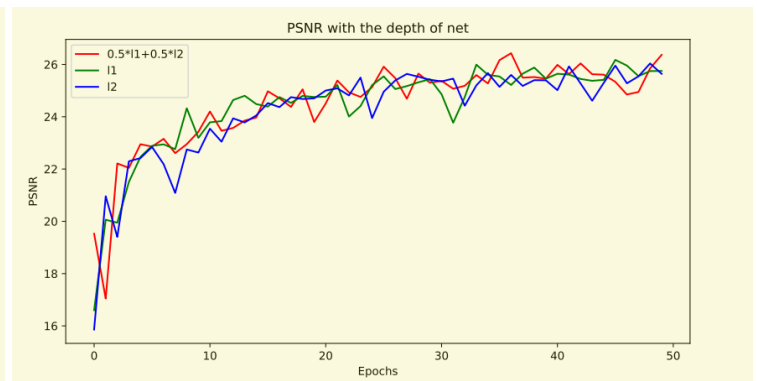
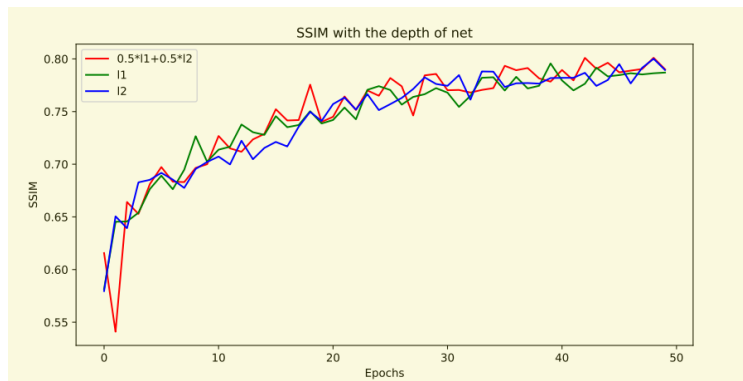
## b. Đánh giá Batchnorm



Giữa các lớp mạng CNN, mô hình SRGAN có các lớp Batchnorm để chuẩn hóa các giá trị trong cùng một batch về dạng chuẩn. Qua đánh giá việc sử dụng Batchnorm cho kết quả tốt hơn khi không sử dụng Batchnorm. Hàm Loss của mô hình sử dụng Batchnorm sẽ giảm mượt hơn, trong khi không sử dụng thì mô hình dễ bị biến thiên.

## c. Đánh giá hàm mục tiêu mới

Ở trong hàm mục tiêu của SRGAN, thành phần  $L_{reconstruct}$  đảm nhận vai trò giữ cho mô hình đưa ra được hình ảnh với nội dung giống như ảnh gốc.  $L_{reconstruct}$  có thể sử dụng  $L1$ ,  $L2$  hoặc kết hợp cả hai. Trong đánh giá này, em đánh giá trên  $L1$ ,  $L2$  và  $L_{combine} = 1/2 L1 + 1/2 L2$ .



Qua kết quả cho thấy, việc sử dụng kết hợp cả hai  $L1$ ,  $L2$  cho kết quả nhỉnh hơn ở hai chỉ số. Ở những epoch đầu,  $L2$  cho kết quả không tốt tuy nhiên mô hình vẫn hội tụ. Ảnh sinh ra từ mô hình  $L2$  sẽ có độ mượt hơn, trong khi ảnh sinh từ mô hình  $L1$  cho ảnh góc cạnh hơn.

### 3. Đánh giá trên bộ dữ liệu khác

Qua bước đánh giá mô hình để chọn tham số phù hợp:

- ☐ Với SRResnet, chọn số RB=7, có dùng Batchnorm
- ☐ Với SRGAN, chọn số RB = 7, có dùng Batchnorm, sử dụng hàm loss kết hợp  $L1$  và  $L2$ .

Đánh giá trên các bộ dữ liệu tiêu chuẩn khác trong bài toán SISR, em thu được kết quả như sau.

Mô hình \ bộ dữ liệu PSNR	BDS100	Set14	Set5
SRResnet	<b>26.68</b>	<b>30.87</b>	<b>27.46</b>
SRGAN	26.07	29.36	26.49

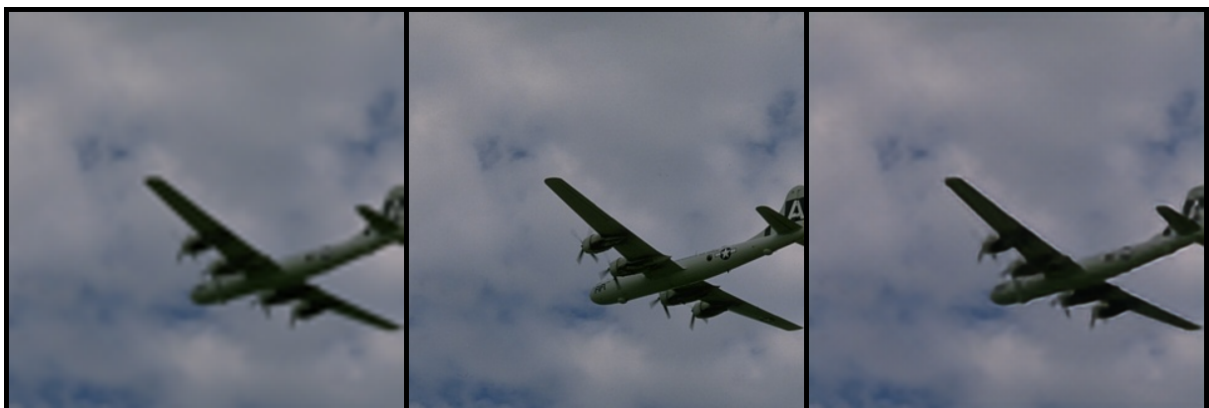
Mô hình\ bộ dữ liệu SSIM	BDS100	Set14	Set5
SRResnet	<b>0.82</b>	<b>0.94</b>	<b>0.86</b>
SRGAN	0.81	0.92	0.85

#### 4. Một số kết quả.

- a. Từ trái qua phải (Ảnh từ phương pháp Bicubic, Ảnh gốc, Ảnh từ SRGAN với SSIM = 0.93 và PSNR = 30.68)



- b. Từ trái qua phải (Ảnh từ phương pháp Bicubic, Ảnh gốc, Ảnh từ SR Resnet với SSIM = 0.98 và PSNR = 37.2)



- c. Ảnh từ bộ dữ liệu Set14

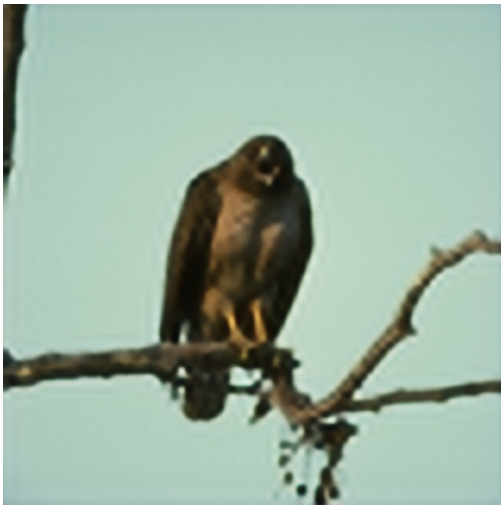


SRGAN: (28.06/ 0.94)



SRResnet: (29.47/ 0.95)

d. Ảnh từ bộ dữ liệu BDS100



SRGAN: (28.39/ 0.95)



SRResnet:(30.19/0.96)

e. Ảnh từ bộ dữ liệu Set5



SRGAN: (31.86/0.94)

SRResnet (33.17/0.94)

Mã nguồn bài báo cáo:

<https://github.com/khangt1k25/Super-Resolution-Image>

## VII. Kết Luận

Trong bài toán lần này, em tiếp cận bài toán tăng độ phân giải cho ảnh theo các mô hình học sâu. Bài toán dựa trên cài đặt và đánh giá mô hình SRResnet và SRGAN. Các kết quả đạt được so với các bài báo gốc là tương đối nhau.

Bài toán SISR có khả năng tăng chất lượng bằng cách tăng kích thước của bức ảnh gốc nhỏ và vẫn giữ được các đặc trưng của bức ảnh. Tuy vậy điểm hạn chế là khi tăng kích thước, vẫn có những bức ảnh chưa thực sự sắc nét như mong muốn. Trong tương lai, em sẽ nghiên cứu, thí nghiệm thêm về các bài báo SIS và các bài toán làm nét ảnh, tăng chất lượng mà vẫn giữ nguyên được chiều của bức ảnh như bài toán tăng nét, chuyển ảnh.

## VIII. Tài liệu tham khảo

- **[Photo-Realistic Single Image Super-Resolution Using a GAN]**  
<https://arxiv.org/pdf/1609.04802.pdf>
- **[PytorchGAN]**  
<https://github.com/eriklindernoren/PyTorch-GAN>
- **[PytorchSISR]**  
<https://github.com/vutrongnghiahust99/SuperResolutionML-DL>
- **[Blog]**  
<https://medium.com/analytics-vidhya/super-resolution-with-srresnet-srgan-2859b87c9c7f>