

- TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN, ĐHQG-HCM  
KHOA CÔNG NGHỆ THÔNG TIN



# IMAGE INPAINTING

MÔN HỌC: XỬ LÝ ẢNH VÀ VIDEO SỐ

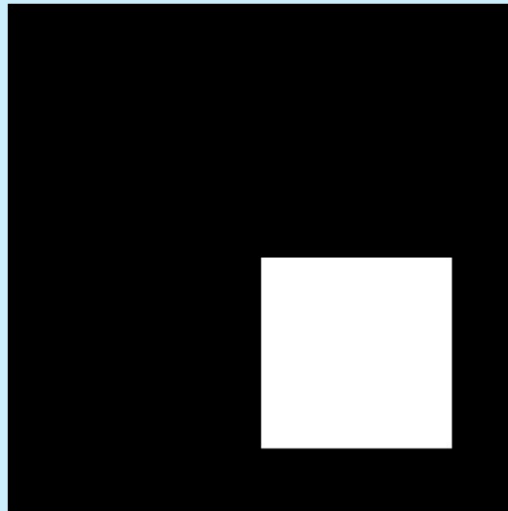
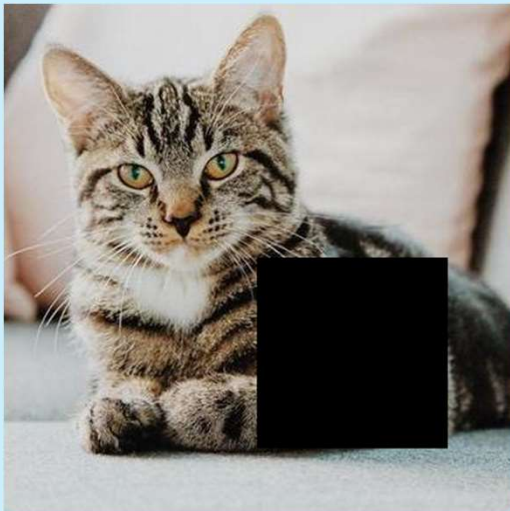
Nhóm Ngọt

21120161	Tiêu Ân Tuấn
21120180	Nguyễn Bích Khuê
21120291	Nguyễn Đức Nam

# Nhắc lại bài toán

# PHÁT BIỂU BÀI TOÁN

INPUT



(Optional)

OUTPUT



**Ẩn số của bài toán:** Giá trị pixel cần khôi phục lấy từ đâu ra?



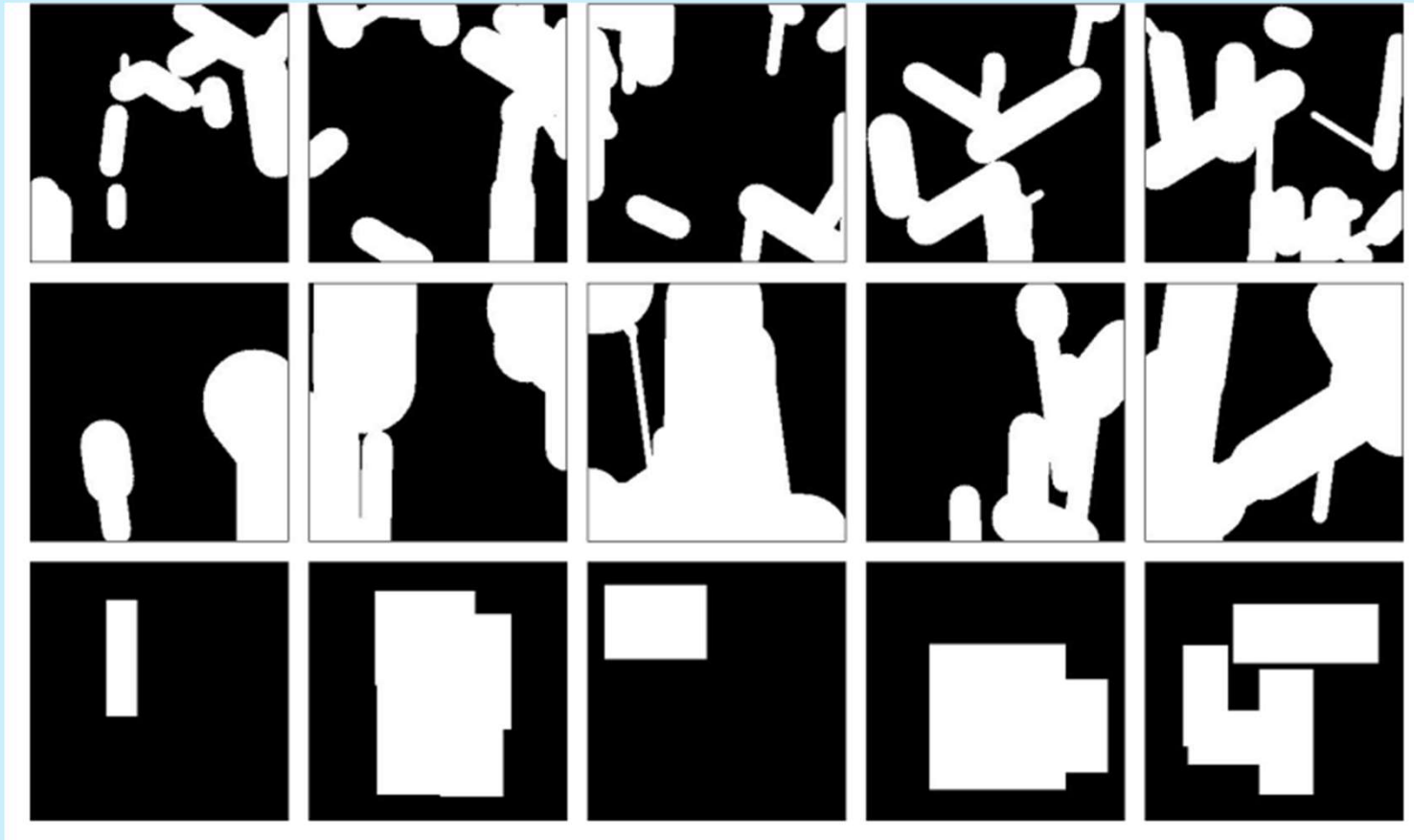


# Dữ liệu

- Mask generation
- Places dataset



# Random Mask Generation



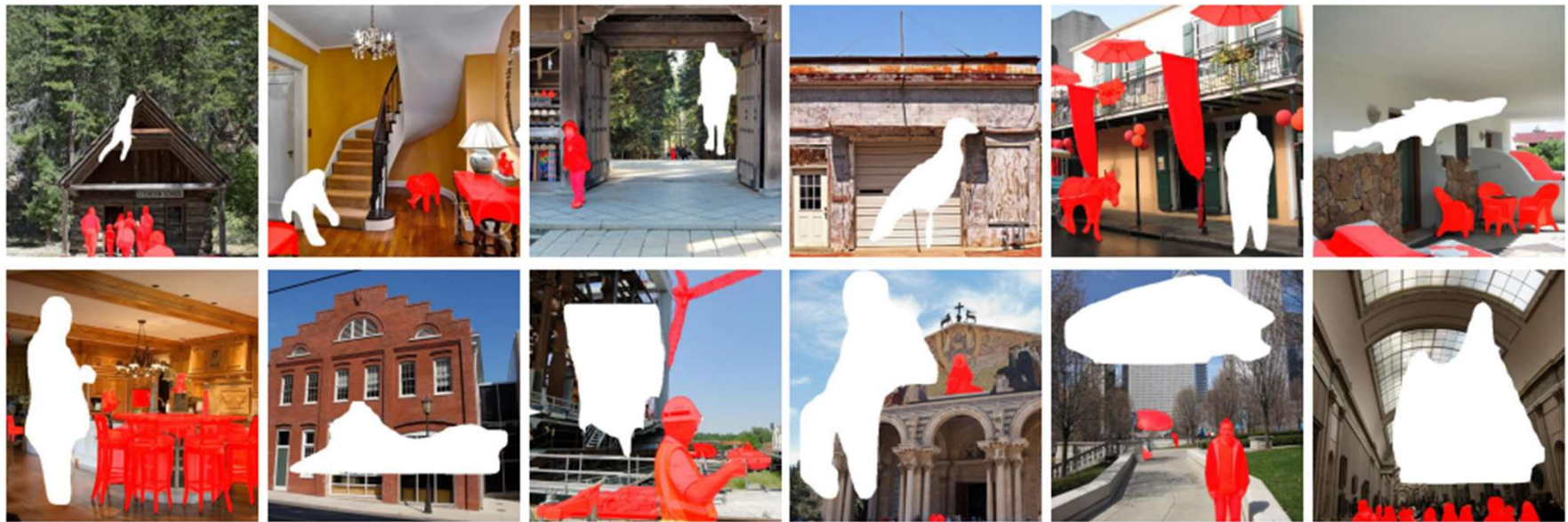
```

1 from np.random import uniform
2
3 def gen_large_mask(img_h, img_w, n):
4     """ img_h:    int, an image height
5         img_w:    int, an image width
6         marg:     int, a margin for a box starting coordinate
7         p_irr:    float, 0 <= p_irr <= 1, a probability of a polygonal chain mask
8
9         min_n_irr: int, min number of segments
10        max_n_irr: int, max number of segments
11        max_l_irr: max length of a segment in polygonal chain
12        max_w_irr: max width of a segment in polygonal chain
13
14        min_n_box: int, min bound for the number of box primitives
15        min_n_box: int, max bound for the number of box primitives
16        min_s_box: int, min length of a box side
17        max_s_box: int, max length of a box side"""
18
19    mask = ones(img_h, img_w)
20
21    if np.random.uniform(0,1) < p_irr: # generate polygonal chain
22        n = uniform(minn_irr, maxn_irr) # sample number of segments
23
24        for _ in range(n):
25            y = uniform(0, img_h) # sample a starting point
26            x = uniform(0, img_w)
27
28            a = uniform(0, 360) # sample angle
29            l = uniform(10, max_l_irr) # sample segment length
30            w = uniform(5, max_w_irr) # sample a segment width
31
32            # draw segment starting from (x,y) to (x_,y_) using brush of width w
33            x_ = x + l * sin(a)
34            y_ = y + l * cos(a)
35
36            gen_segment_mask(mask, start=(x, y), end=(x_, y_), brush_width=w)
37            x, y = x_, y_
38    else: # generate Box masks
39        n = uniform(min_n_box, min_n_box) # sample number of rectangles
40
41        for _ in range(n):
42            h = uniform(min_s_box, max_s_box) # sample box shape
43            w = uniform(min_s_box, max_s_box)
44
45            x_0 = uniform(marg, img_w - marg + w) # sample upper-left coordinates of box
46            y_0 = uniform(marg, img_h - marg - h)
47
48            gen_box_mask(mask, size=(img_w, img_h), masked=(x_0, y_0, w, h))
49    return mask

```



# Segmentation Mask Generation





# Bộ dữ liệu Places

**Tập dữ liệu huấn luyện:** 1.8 triệu ảnh từ bộ dữ liệu Places-Standard gồm các hình ảnh về các khung cảnh khác nhau mỗi ảnh có kích thước 512x512 pixel.

**Tập dữ liệu thẩm định:** 2000 cặp ảnh-mặt nạ ảnh được lấy ngẫu nhiên từ tập thẩm định (validation) của bộ dữ liệu Places. Mặt nạ được tạo ra từ segmentation-based mask generation.

**Tập dữ liệu kiểm tra:** kiểm tra trên 3 loại mặt nạ với kích thước khác nhau (nhỏ - trung bình - lớn) và mặt nạ phân đoạn (segmentation).  
Dùng 3000 cặp ảnh-mặt nạ cho 3 loại đầu và 4000 cho loại mặt nạ phân đoạn.

veterinarians office



elevator door



bedroom



cafeteria



staircase



bar



conference center

shoe shop

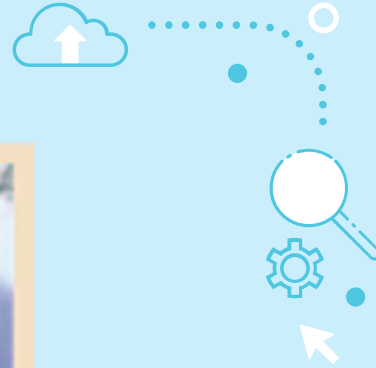
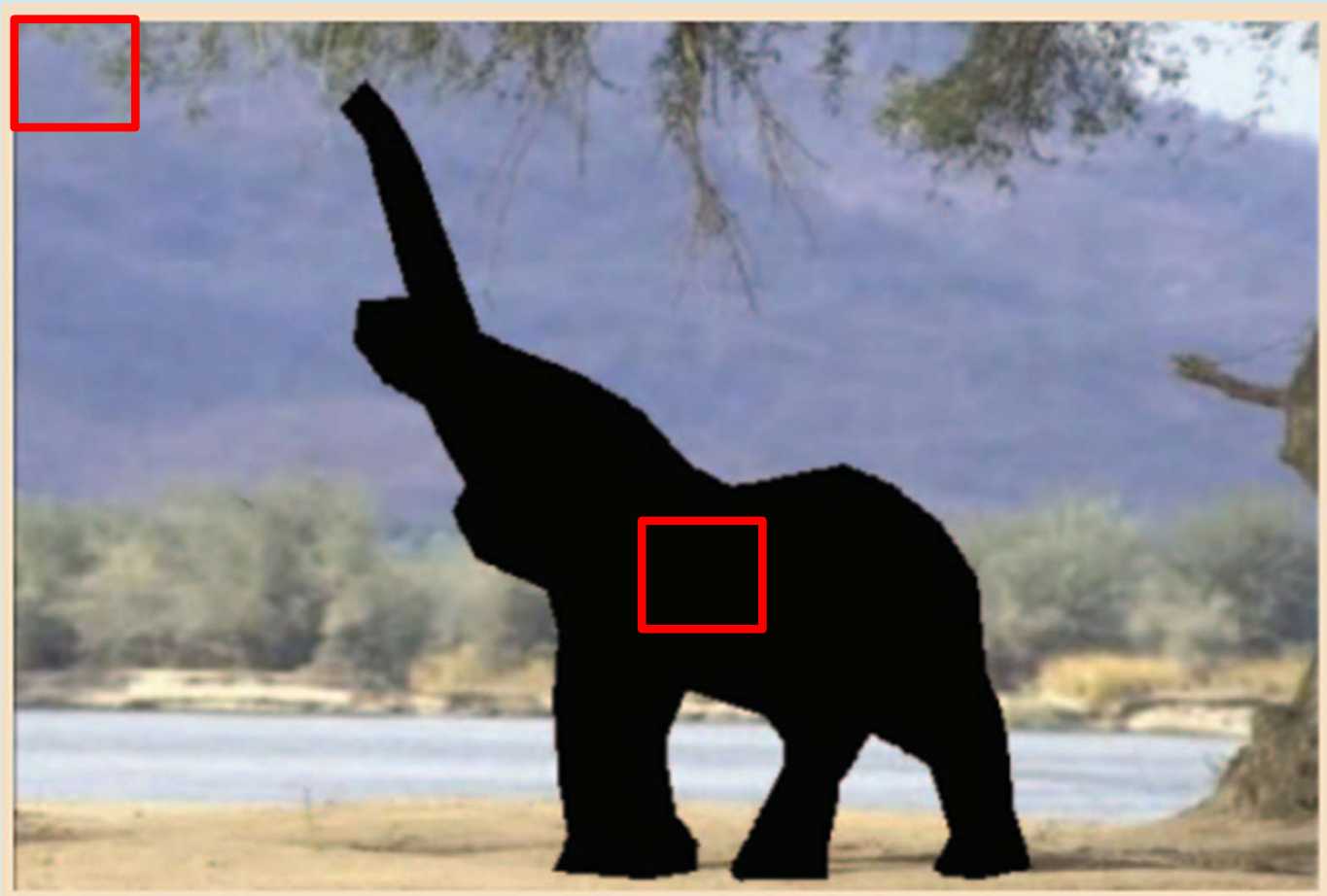
# PHƯƠNG PHÁP LAMA

- Mô hình
- Hàm mất mát
- Mật nạ

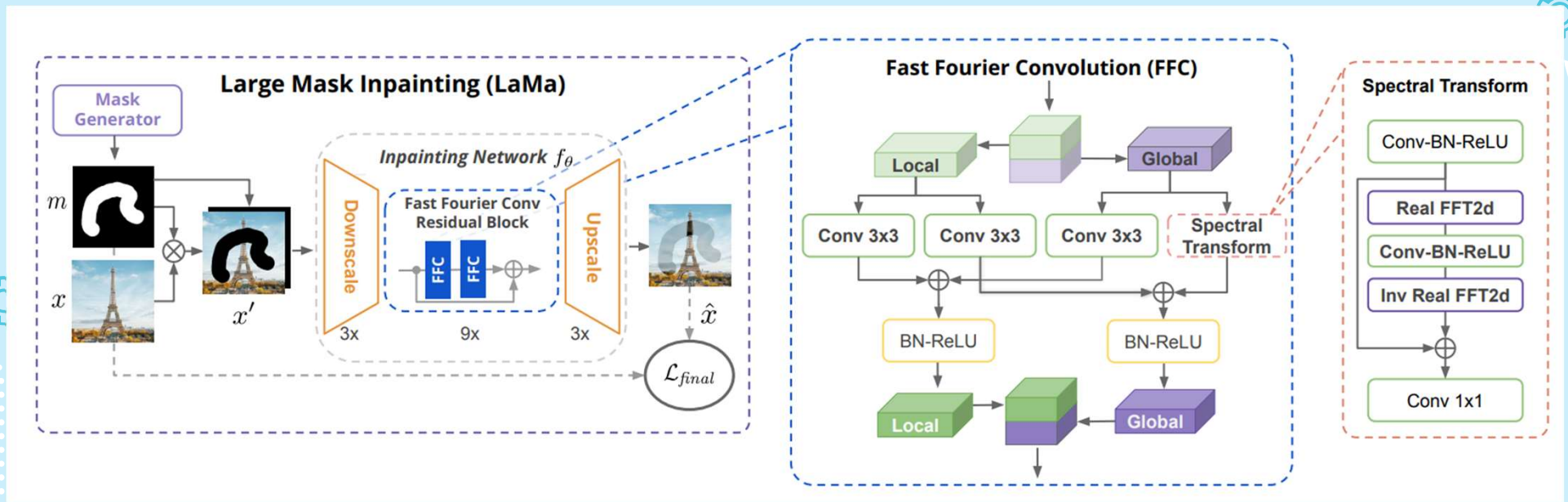


# Đặt vấn đề

3x3 convolution



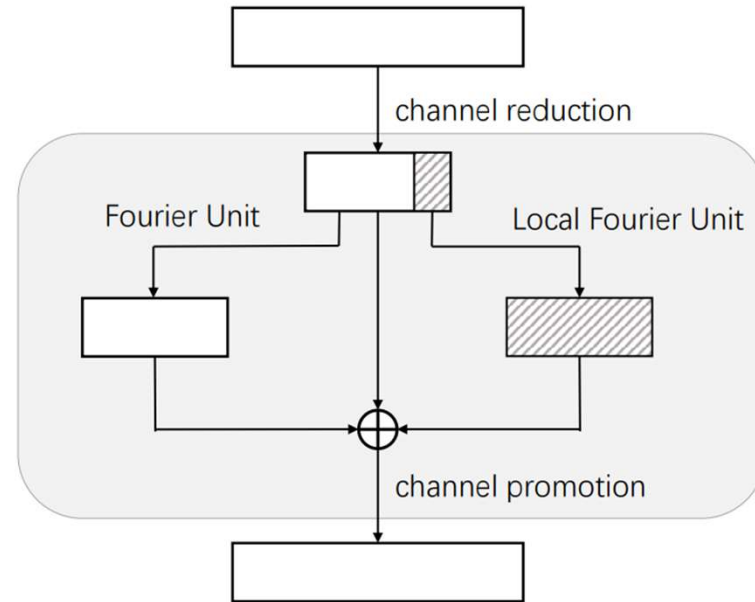
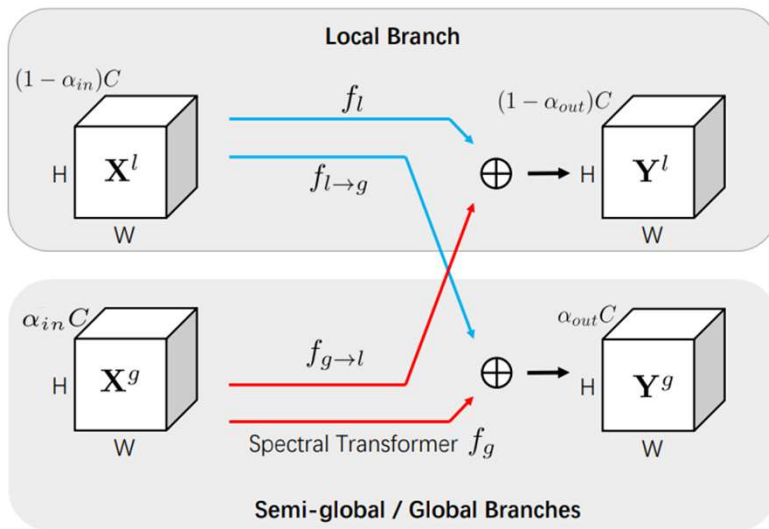
# MÔ HÌNH LAMA



**Input:**  $x' = \text{stack}(x \odot m, m)$

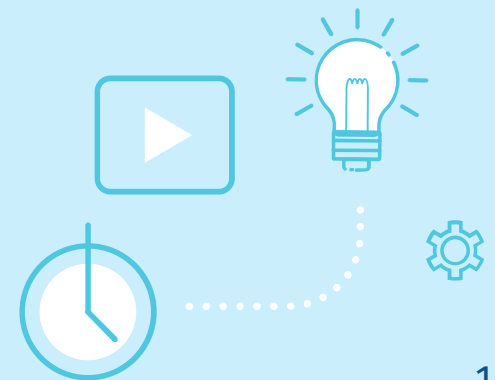
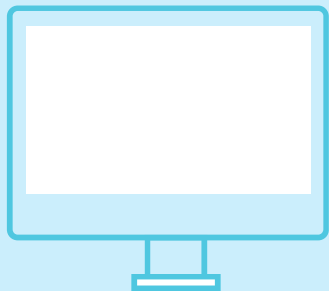
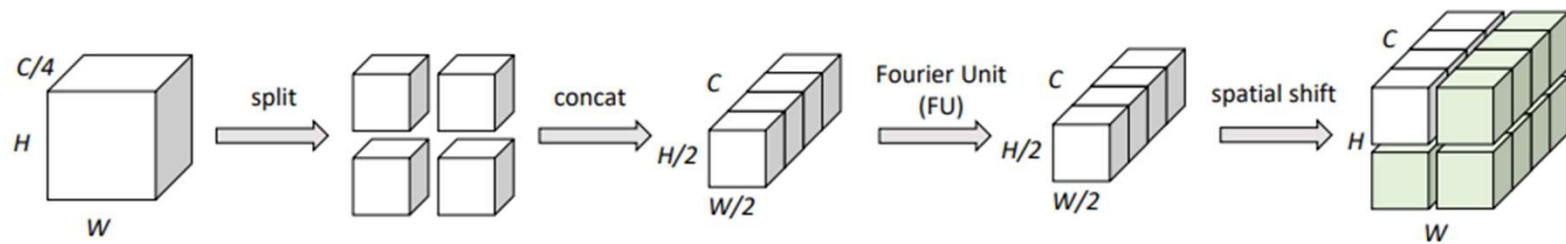
**Output:**  $\hat{x} = f_\theta(x')$

# Fast Frouier Convolution (FFC) - based



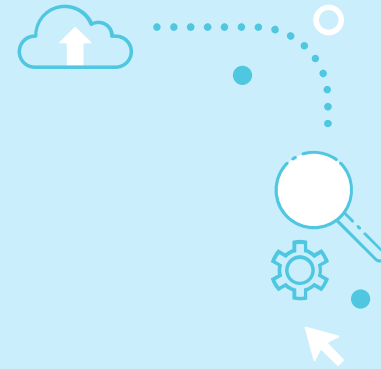
$$\begin{aligned} \mathbf{Y}^l &= \mathbf{Y}^{l \rightarrow l} + \mathbf{Y}^{g \rightarrow l} = f_l(\mathbf{X}^l) + f_{g \rightarrow l}(\mathbf{X}^g), \\ \mathbf{Y}^g &= \mathbf{Y}^{g \rightarrow g} + \mathbf{Y}^{l \rightarrow g} = f_g(\mathbf{X}^g) + f_{l \rightarrow g}(\mathbf{X}^l). \end{aligned}$$

# Local Fourier Unit (LFC)

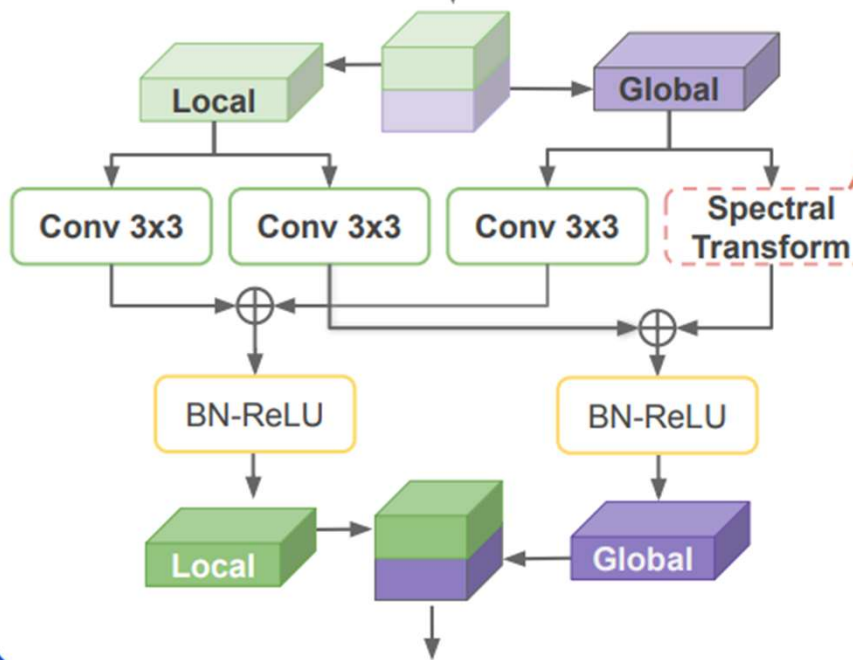




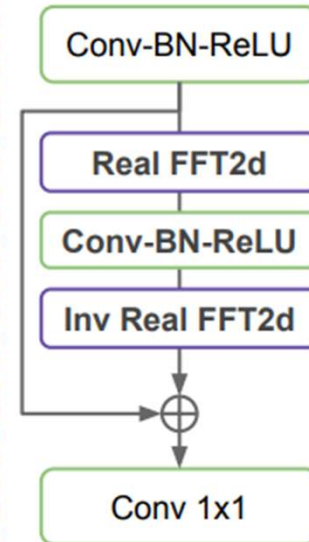
# Fast Fourier Convolution (FFC) in Lama



## Fast Fourier Convolution (FFC)



## Spectral Transform



$$\begin{aligned}
 \text{Real FFT2d} &: \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{C}^{H \times \frac{W}{2} \times C}, \\
 \text{ComplexToReal} &: \mathbb{C}^{H \times \frac{W}{2} \times C} \rightarrow \mathbb{R}^{H \times \frac{W}{2} \times 2C}, \\
 \text{ReLU} \circ \text{BN} \circ \text{Conv1x1} &: \mathbb{R}^{H \times \frac{W}{2} \times 2C} \rightarrow \mathbb{R}^{H \times \frac{W}{2} \times 2C}, \\
 \text{RealToComplex} &: \mathbb{R}^{H \times \frac{W}{2} \times 2C} \rightarrow \mathbb{C}^{H \times \frac{W}{2} \times C}, \\
 \text{Inverse Real FFT2d} &: \mathbb{C}^{H \times \frac{W}{2} \times C} \rightarrow \mathbb{R}^{H \times W \times C}.
 \end{aligned}$$

# HÀM MẤT MÁT (LOSS FUNCTION)

## Perceptual Loss

$$\mathcal{L}_{HRFPL}(x, \hat{x}) = \mathcal{M}([\phi_{HRF}(x) - \phi_{HRF}(\hat{x})]^2),$$

## Adversarial

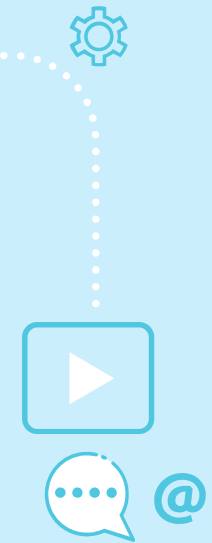
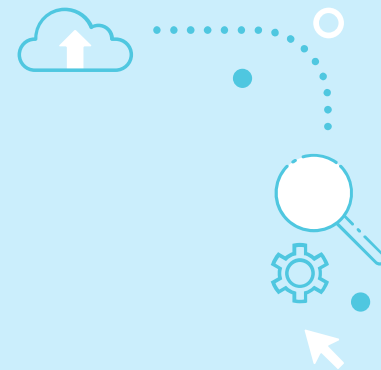
$$\begin{aligned} \mathcal{L}_D = & -\mathbb{E}_x [\log D_\xi(x)] - \mathbb{E}_{x,m} [\log D_\xi(\hat{x}) \odot m] \\ & - \mathbb{E}_{x,m} [\log (1 - D_\xi(\hat{x})) \odot (1 - m)] \end{aligned} \quad (2)$$

$$\mathcal{L}_G = -\mathbb{E}_{x,m} [\log D_\xi(\hat{x})] \quad (3)$$

$$L_{Adv} = \text{sg}_\theta(\mathcal{L}_D) + \text{sg}_\xi(\mathcal{L}_G) \rightarrow \min_{\theta, \xi} \quad (4)$$

# Fast Frouier Convolution (FFC)

$$\mathcal{L}_{final} = \kappa L_{Adv} + \alpha \mathcal{L}_{HRFPL} + \beta \mathcal{L}_{DiscPL} + \gamma R_1$$



# HÀM MẤT MÁT

- Ý nghĩa khoa học và ứng dụng của chủ đề
- Phát biểu bài toán
- Đóng góp

# DEMO

Link demo:

[https://colab.research.google.com/drive/1PDRmMPso0Hg-3YLoPASvbJfkIPQckc1M?usp=sharing#scrollTo=7b\\_\\_kN-n-hZH](https://colab.research.google.com/drive/1PDRmMPso0Hg-3YLoPASvbJfkIPQckc1M?usp=sharing#scrollTo=7b__kN-n-hZH)

Work

Fourier Convolutions (Lama).pdf x Result\_Lama\_Inpainting demo.ip x YouTube

https://colab.research.google.com/drive/1PDRmMPso0Hg-3YLoPASybJfklPQcke1M?usp=sharing#scrollTo=-VZWysTMeGDM


yup TP MXH US GG\_S PL Study KTX SHEET News LLM Trello F8 20ex CRF Best Book to Read i...

+ Code + Text

RAM Disk Colab AI

> Draw a Mask, Press Finish, Wait for Inpainting

Show code



mask

img

img \* mask

0s completed at 3:38 PM

Type here to search

3:38 PM 1/6/2024

20





**Cảm ơn mọi người  
đã lắng nghe**

