



ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

oo

# TĂNG CƯỜNG KHẢ NĂNG CHUYỂN KIẾU CHỮ ĐA NGÔN NGỮ TRONG BÀI TOÁN ONE-SHOT BẰNG MÔ HÌNH KHUẾCH TÁN

Sinh viên thực hiện:

**Trần Đình Khánh Đăng**

Giảng viên hướng dẫn:

**TS. Dương Việt Hằng**

Lớp khoá học:

KHMT2022.1

Khoa:

Khoa học máy tính

# Mục lục

**1. Giới thiệu**

**2. Phương pháp đề xuất**

**3. Thực nghiệm và kết quả**

**4. Kết luận**

# Mục lục

1. Giới thiệu

2. Phương pháp đề xuất

3. Thực nghiệm và kết quả

4. Kết luận

# Thách thức của thiết kế truyền thống

Quy trình thiết kế font truyền thống gấp 3 rào cản lớn:

## 1. Chi phí:

- Quy trình vẽ tay tốn kém nhân lực và thời gian.
- Hiệu suất thấp do tính chất lặp lại thủ công.

## 2. Quy mô:

- Latin: ~52 ký tự.
- **CJK (Hán/Nôm):** Hàng vạn ký tự (> 50.000 ký tự).

→ **Bất khả thi nếu làm tay hoàn toàn.**

## 3. Rào cản Đa ngôn ngữ:

- Các ngôn ngữ **Low-resource** hoặc có **dấu phức tạp** (như Tiếng Việt) thường xuyên bị thiếu font đồng bộ.

→ Gây khó khăn lớn cho việc **Bản địa hóa thương hiệu**.

# Giải pháp: One-shot Font Generation

Cơ chế **One-shot**: Tách biệt phong cách từ **1 mẫu ảnh** → Chuyển giao (Transfer) sang **bất kỳ ký tự nào**.



Nội dung (Content)

1 Mẫu Style (Reference)

Kết quả (Generated)

→ Giải pháp tối ưu cho bài toán Chi phí, Quy mô và Đa ngôn ngữ.

# Mục tiêu & Đóng góp

Mục tiêu: Xây dựng giải pháp **Cross-Lingual (Đa ngôn ngữ)** tổng quát.

→ **Phạm vi (Scope):** Tập trung vào cặp **Latin - Hán tự**. (Lý do: Đây là cặp có cấu trúc khác biệt lớn nhất → Bài toán khó nhất).

## Đóng góp chính:

1. Xây dựng pipeline dựa trên **Diffusion Model**.
2. Đề xuất mô-đun **CL-SCR** để xử lý khác biệt cấu trúc.

# Khoảng cách hình thái học (Morphological Gap)

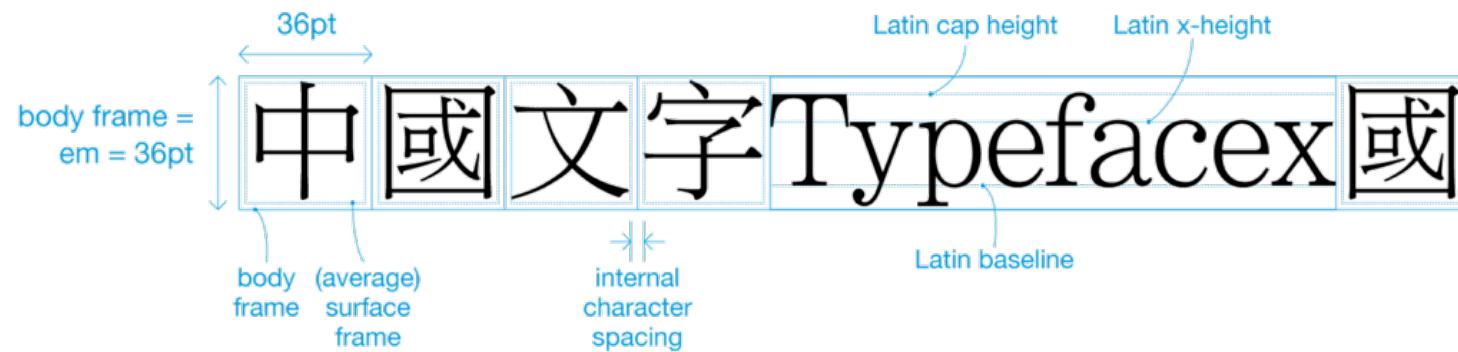
Tại sao Latin - Hán tự là thách thức lớn nhất?

## 1. Latin (Hệ chữ cái):

- Cấu trúc tuyến tính (Linear), ít nét.
- **Vấn đề:** Quá ít thông tin để suy diễn.

## 2. Hán tự (Hệ tượng hình):

- Cấu trúc khối vuông (Block), dày đặc.
- **Vấn đề:** Dễ bị biến dạng cấu trúc.



→ Rào cản lớn nhất trong việc bảo toàn cấu trúc khi chuyển đổi phong cách.

# Mục lục

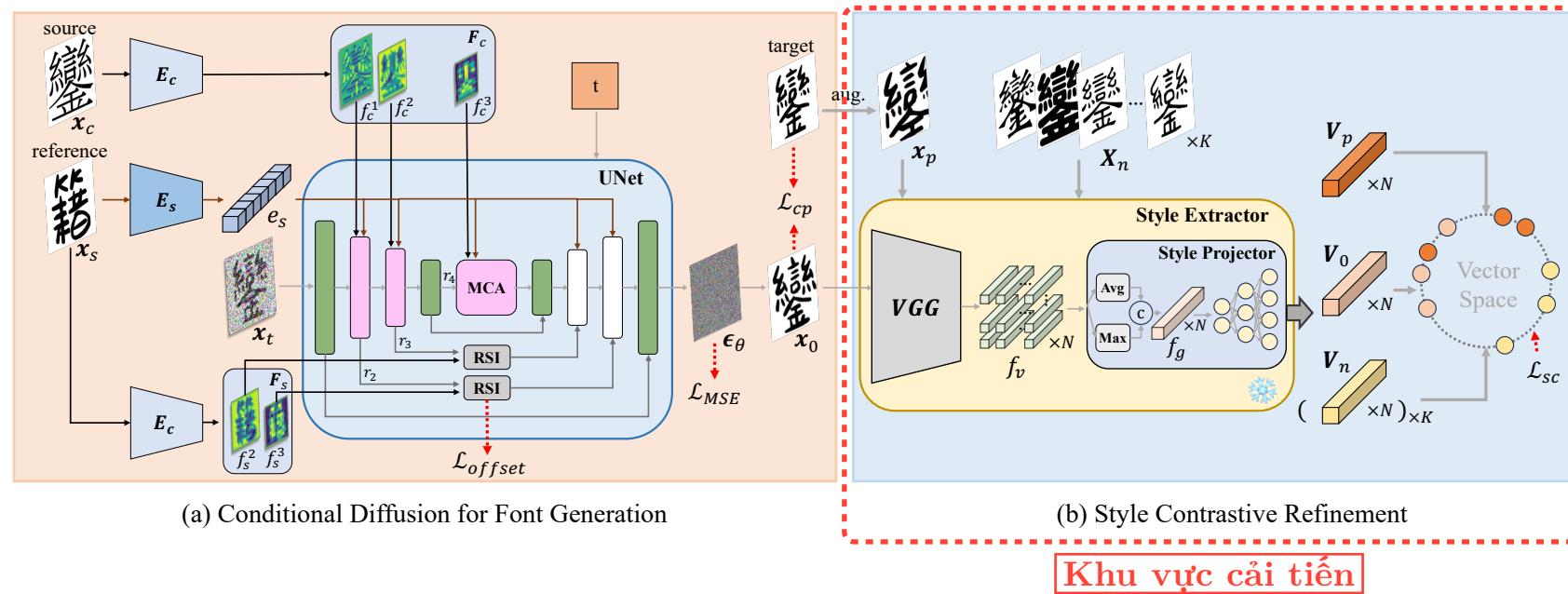
1. Giới thiệu

2. Phương pháp đề xuất

3. Thực nghiệm và kết quả

4. Kết luận

# Kiến trúc đề xuất



## Giai đoạn 1 (Nền tảng FontDiffuser):

- **MCA:** Tổng hợp đặc trưng đa tỷ lệ.
- **RSI:** Xử lý biến dạng hình học.
- → **Mục tiêu:** Đảm bảo tái tạo đúng **cấu trúc chữ**.

## Giai đoạn 2 (Trong khung đỏ):

- Thay thế mô-đun SCR gốc bằng kiến trúc **CL-SCR** đề xuất.
- → Nâng cấp khả năng học **Cross-Lingual**.

# Động lực & Ý tưởng

## Vấn đề của Giai đoạn 1:

- Giai đoạn 1 chỉ tập trung tối ưu hoá **điểm ảnh** (Pixel-wise).
- → Học tốt cấu trúc nhưng yếu về **biểu diễn phong cách trừu tượng**. Khi chuyển sang hệ chữ khác, mô hình bị “mất phương hướng” vì không còn điểm ảnh tương đồng để so sánh.

## Trực giác cho Giai đoạn 2 (Cross-Lingual Bridge):

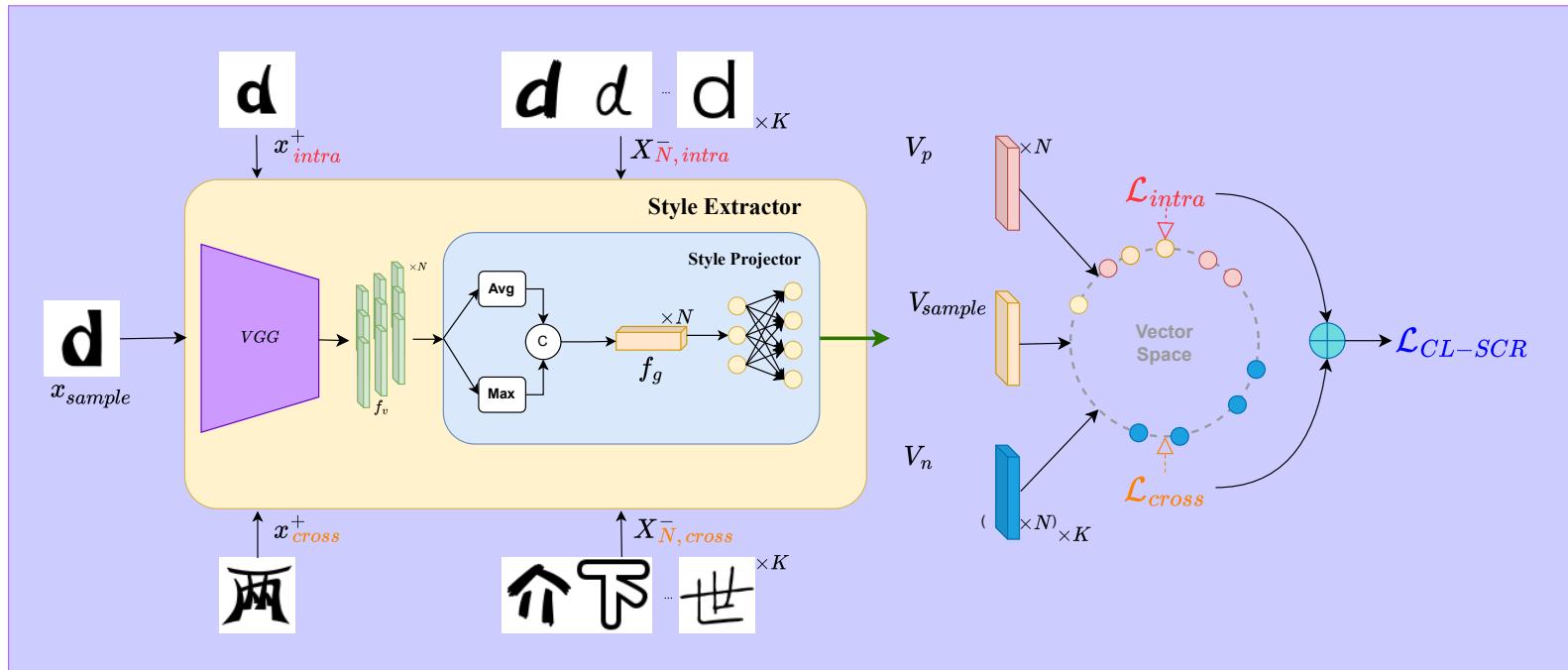
- Cần một cơ chế **tách biệt phong cách** khỏi nội dung.
- Tận dụng các **nét tương đồng** (stroke-level) giữa Latin và Hán tự thông qua cơ chế **Học tương phản (Contrastive Learning)**.
- → Dùng CL-SCR để “ép” mô hình tìm ra mẫu số chung về phong cách giữa hai ngôn ngữ.

## Giải pháp CL-SCR: Áp dụng cơ chế **Contrastive Learning**:

- **Intra-domain:** Giữ bản sắc ngôn ngữ nguồn.
- **Cross-domain:** Tìm điểm chung giữa hai hệ chữ.

# Kiến trúc mô-đun CL-SCR

Cơ chế giám sát luồng đôi (Dual-stream Supervision):



**Hình 3.7:** Kiến trúc mạng CL-SCR với hai luồng giám sát Intra và Cross.

# Công thức hàm Loss (CL-SCR)

Dựa trên nguyên lý **InfoNCE** (Cơ chế Kéo - Đẩy):

## 1. Intra-Lingual ( $L_{\text{intra}}$ )

$$L_{\text{intra}} = -\log \frac{\exp(q \cdot k^+)}{\exp(q \cdot k^+) + \sum \exp(q \cdot k^-)}$$

→ **Mục tiêu:** Đảm bảo tính nhất quán nội bộ.

## 2. Cross-Lingual ( $L_{\text{cross}}$ )

$$L_{\text{cross}} = -\log \frac{\exp(q \cdot k_{\text{cross}}^+)}{\exp(q \cdot k_{\text{cross}}^+) + \sum \exp(q \cdot k_{\text{cross}}^-)}$$

→ **Mục tiêu:** Kéo ảnh sinh về phong cách đích (Target).

## Tổng hợp Loss:

$$L_{\text{CL-SCR}} = \alpha \cdot L_{\text{intra}} + \beta \cdot L_{\text{cross}}$$

(Trong đó  $\beta > \alpha$  để ưu tiên học chuyển đổi đa ngữ)

# Hàm mục tiêu tổng quát

Mô hình tối ưu hoá đồng thời 4 thành phần:

$$L_{\text{total}} = \underbrace{L_{\text{MSE}}}_{\text{Tái tạo ảnh}} + \lambda_{\text{cp}} \underbrace{L_{\text{cp}}}_{\text{Nội dung}} + \lambda_{\text{offset}} \underbrace{L_{\text{offset}}}_{\text{Cấu trúc}} + \lambda_3 \underbrace{L_{\text{CL-SCR}}}_{\text{Phong cách (Đề xuất)}}$$

- $L_{\text{MSE}}$  &  $L_{\text{offset}}$ : Giữ vai trò bảo toàn khung xương (Giai đoạn 1).
- $L_{\text{CL-SCR}}$ : Đóng vai trò then chốt trong việc chuyển giao phong cách (Giai đoạn 2).

# Mục lục

1. Giới thiệu

2. Phương pháp đề xuất

**3. Thực nghiệm và kết quả**

4. Kết luận

# Chiến lược Huấn luyện

**Thiết lập:** GPU Tesla P100 (16GB) ◇ Batch size: 4 ◇ **Inference:** DPM-Solver++ (20 bước).  
(Mô-đun CL-SCR được tiền huấn luyện (pre-train) độc lập trước khi đưa vào Giai đoạn 2).

## Giai đoạn 1: Khởi tạo

- **Mục tiêu:** Học tái tạo cấu trúc chữ (Skeleton).
- **Loss:**  $L_{\text{MSE}} + \lambda_{\text{cp}} L_{\text{cp}} + \lambda_{\text{offset}} L_{\text{offset}}$ .
- **Quy mô:** 400.000 bước (Steps).
- **Learning Rate:**  $1 \times 10^{-4}$ .

→ **Kết quả:** Học cấu trúc nội dung và phong cách cơ bản.

## Giai đoạn 2: Tinh chỉnh

- **Mục tiêu:** Tách biệt và chuyển giao Style (Cross-Lingual).
- **Loss:** Thêm hàm **CL-SCR** (Contrastive Loss).
- **Quy mô:** 30.000 bước.
- **Learning Rate:** Giảm xuống  $1 \times 10^{-5}$ .
- **Kỹ thuật:** Áp dụng **Data Augmentation** (Random Crop) để chống học vét.

→ **Kết quả:** Phong cách sắc nét, chuẩn xác.

# Dữ liệu & Kịch bản đánh giá

Cơ sở thực nghiệm của khoá luận:

## 1. Bộ dữ liệu:

- **Nguồn:** 818 font song ngữ (FTransGAN).
- **Cấu trúc:** Ghép cặp Latin ( $\sim 52$  ký tự) và Hán tự ( $\sim 800$  ký tự).  
→ Đảm bảo sự nhất quán phong cách (Ground-truth).

## 2. Kịch bản:

### a. SFUC (Font đã biết):

- Sinh ký tự mới từ font đã train.
- **Mục tiêu:** Kiểm tra khả năng “học thuộc”.

### b. UFSC (Font chưa biết - Quan trọng):

- Sinh ký tự từ font **mới hoàn toàn**.
- **Mục tiêu:** Đánh giá khả năng **One-shot Generalization**.

# Các thước đo đánh giá

Để đảm bảo tính khách quan, khoá luận sử dụng hệ thống đo lường đa chiều:

## 1. Định lượng:

- **FID (Quan trọng nhất):** Đo khoảng cách phân bố giữa ảnh sinh và ảnh thật.  
→ **FID càng thấp → Ảnh càng chân thực.**
- **L1 / SSIM:** Đo độ chính xác về điểm ảnh (Pixel) và cấu trúc (Structure).
- **LPIPS:** Đo độ tương đồng theo nhận thức của mắt người.

## 2. Định tính:

- **Kiểm tra trực quan:** So sánh trực quan các chi tiết nét.
- **Khảo sát người dùng:** Khảo sát mù trên người dùng để đánh giá độ hài lòng.

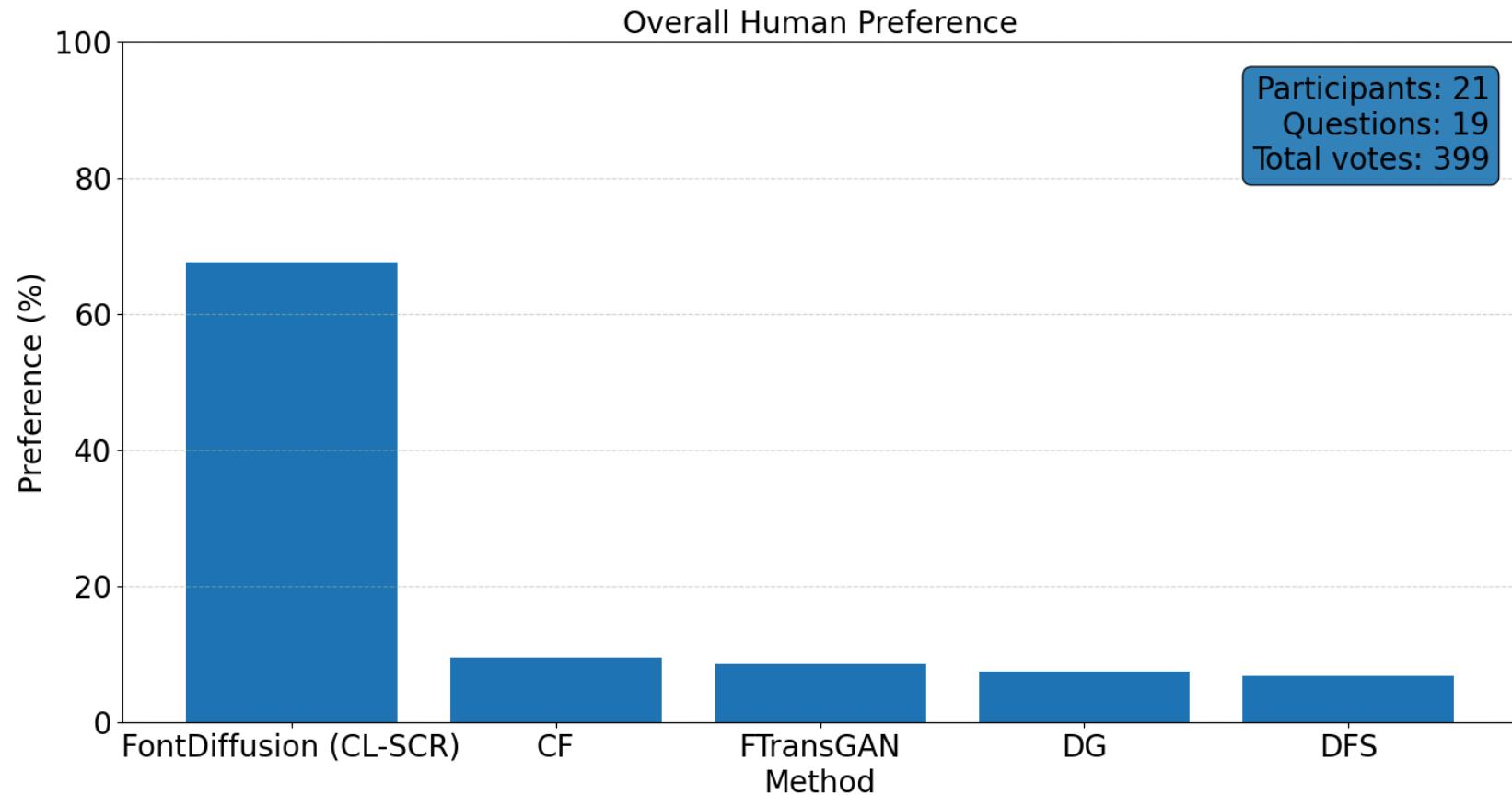
# Kết quả định lượng

	Model	SFUC				UFSC			
		L1 ↓	SSIM ↑	LPIPS ↓	FID ↓	L1 ↓	SSIM ↑	LPIPS ↓	FID ↓
L↑C ↑	DG-Font	0.2773	0.2702	0.4023	106.3833	0.2797	0.2654	0.3649	54.0974
	CF-Font	0.2659	0.2740	0.3979	91.2134	0.2638	0.2716	0.3615	51.3925
	DFS	0.2131	0.3558	0.3812	45.4212	<b>0.2008</b>	0.3048	0.3876	62.7206
	FTransGAN	<b>0.1844</b>	<b>0.3900</b>	0.3548	40.4561	<u>0.2089</u>	<u>0.3109</u>	0.3329	42.1053
	FontDiffuser (Baseline)	0.1976	0.3775	<u>0.2968</u>	<u>14.6871</u>	0.2283	0.2946	<u>0.3184</u>	<u>29.0999</u>
	<b>Ours</b>	<u>0.1939</u>	<u>0.3890</u>	<b>0.2911</b>	<b>11.7691</b>	0.2214	<b>0.3197</b>	<b>0.2954</b>	<b>13.5508</b>
C↑L ↑	DG-Font	0.1462	0.5542	0.2821	74.1655	0.1397	0.5624	0.2751	89.8197
	CF-Font	0.1402	0.5621	0.2790	67.1241	0.1317	0.5756	0.2726	84.3787
	DFS	<b>0.1083</b>	<u>0.6140</u>	0.2585	40.4042	<u>0.1139</u>	<u>0.5819</u>	0.2907	75.2760
	FTransGAN	0.1381	0.5291	0.2851	55.5859	0.1456	0.4949	0.3023	88.4450
	FontDiffuser (Baseline)	<u>0.1223</u>	0.6107	<u>0.2270</u>	<u>21.2234</u>	0.1370	0.5731	<u>0.2476</u>	<u>59.5788</u>
	<b>Ours</b>	<b>0.1083</b>	<b>0.6406</b>	<b>0.2019</b>	<b>14.7298</b>	<b>0.1090</b>	<b>0.6377</b>	<b>0.1985</b>	<b>41.1152</b>

# Kết quả định tính

Source	c	d	e	f	g	毛	毫	民	气	水
Reference	衣	牛	十	生	至	Z	D	W	B	J
DG-Font	奄	𢃔	𠂇	𠂊	𠂊	𠂊	𠂊	𠂊	𠂊	𠂊
CF-Font	𠂊	d	e	𠂊	g	毛	毫	民	气	水
DFS	𠂊	𠂊	𠂊	𠂊	𠂊	毛	毫	民	气	水
FTransGAN	𠂊	𠂊	𠂊	𠂊	𠂊	毛	毫	民	气	水
FontDiffuser (Baseline)	𠂊	𠂊	𠂊	𠂊	𠂊	毛	毫	民	气	水
<b>Ours</b>	c	d	e	+	g	毛	毫	民	气	水
Target	c	d	e	f	g	毛	毫	民	气	水

# Đánh giá người dùng



# Hiệu quả của các mô-đun kiến trúc

Mô-đун				SFUC				UFSC			
	M	R	CL	L1 ↓	SSIM ↑	LPIPS ↓	FID ↓	L1 ↓	SSIM ↑	LPIPS ↓	FID ↓
C	✓	✓	✗	0.1977	0.3809	0.2927	<b>10.9069</b>	0.2266	0.3072	0.3009	14.8680
↑	✗	✗	✓	0.2679	0.2415	0.5199	161.0711	0.2966	0.1687	0.5606	180.2861
L	✓	✓	✓	<b>0.1939</b>	<b>0.3890</b>	<b>0.2911</b>	<u>11.7691</u>	<b>0.2214</b>	<b>0.3197</b>	<b>0.2954</b>	<b>13.5508</b>
L	✓	✓	✗	<b>0.1076</b>	<b>0.6449</b>	<b>0.2005</b>	<b>14.3511</b>	<b>0.1070</b>	<b>0.6413</b>	<b>0.1980</b>	<u>42.8665</u>
↑	✗	✗	✓	0.3234	0.2520	0.5469	205.2360	0.3882	0.1849	0.5951	239.9641
C	✓	✓	✓	<u>0.1083</u>	<u>0.6406</u>	<u>0.2019</u>	<u>14.7298</u>	<u>0.1090</u>	<u>0.6377</u>	<u>0.1985</u>	<b>41.1152</b>

# Mục lục

1. Giới thiệu

2. Phương pháp đề xuất

3. Thực nghiệm và kết quả

4. Kết luận

# Tổng kết đóng góp

Khoá luận đã hoàn thành các mục tiêu đề ra ban đầu:

- Xây dựng thành công Pipeline chuyển đổi phong cách **xuyên hệ chữ (Cross-Script)**, đặc biệt là cặp khó Latin - Hán tự.
- Đề xuất mô-đun **CL-SCR** với cơ chế **Contrastive Learning**, giải quyết hiệu quả vấn đề “Domain Gap” giữa các ngôn ngữ.
- Vượt trội SOTA hiện tại (FID giảm ~ 50%), khắc phục triệt để lỗi “**Bóng ma**” (Ghosting) và “**Biến dạng cấu trúc**” thường gặp ở GAN.

# Hạn chế & Hướng phát triển

Định hướng nghiên cứu trong tương lai:

## Hạn chế:

- Thách thức về Tốc độ:** Do bản chất khử nhiễu lặp lại (Iterative Denoising) của Diffusion, tốc độ suy diễn chậm hơn các phương pháp One-step (GAN).
- Đánh đổi:** Chất lượng ảnh cao đổi lấy chi phí tính toán lớn.

## Hướng phát triển:

- Tăng tốc:** Áp dụng **Consistency Distillation** hoặc **Latent Consistency Models (LCM)** để giảm xuống còn 4-8 bước.
- Mở rộng:** Ứng dụng cho **Tiếng Việt (Thư pháp)** và các ngôn ngữ Low-resource khác.
- Ứng dụng:** Sinh font dạng **Vector (SVG)** để tích hợp trực tiếp vào phần mềm thiết kế.

# Công trình khoa học

D. K. D. Tran and V. H. Duong, “CL-SCR: Decoupling Style and Structure for One-Shot Cross-Script Font Generation,” *The Journal of Supercomputing (under review)*, 2026.

## Lời cảm ơn

**Xin cảm ơn Thầy Cô và Hội đồng  
đã theo dõi và lắng nghe!**

Sinh viên thực hiện:

**Trần Đình Khánh Đăng**

Giảng viên hướng dẫn:

**TS. Dương Việt Hằng**

Lớp khoá học:

KHMT2022.1

Khoa:

Khoa học máy tính

# Tối ưu hoá CL-SCR

Cơ sở thực nghiệm để lựa chọn các siêu tham số tốt nhất.

**a. Chế độ Hàm Loss (Loss Modes):** Tại sao phải kết hợp cả Intra và Cross?

Chế độ	FID (UFSC) ↓	
	L → C	C → L
Intra-only	<u>15.72</u>	<u>41.34</u>
Cross-only	16.26	44.78
<b>Both</b>	<b>13.55</b>	<b>41.12</b>

→ **Both** tận dụng sự ổn định của Intra và khả năng chuyển đổi của Cross.

**b. Trọng số Alpha ( $\alpha$ ) & Beta ( $\beta$ ):** Tại sao ưu tiên  $\beta = 0.7$ ?

$\alpha$	$\beta$	FID (UFSC) ↓	
		L → C	C → L
0.7	0.3	<u>14.48</u>	45.23
0.5	0.5	15.18	<u>43.42</u>
<b>0.3</b>	<b>0.7</b>	<b>13.55</b>	<b>41.12</b>

→ Bài toán Cross-Lingual cần ưu tiên học các đặc trưng xuyên ngôn ngữ ( $\beta$  lớn).

# Phân tích độ nhạy

Ảnh hưởng của Số mẫu âm & Guidance Scale

c. **Số lượng mẫu âm ( $K$ ):** Trong hàm loss InfoNCE.

$K$	FID (UFSC) ↓	
	$L \rightarrow C$	$C \rightarrow L$
4	<b>13.55</b>	<b>41.12</b>
8	<u>15.02</u>	43.81
16	16.79	<u>43.50</u>

→  $K=4$  là điểm cân bằng tối ưu cho cả hai chiều.

d. **Trọng số hướng dẫn (Scale -  $s$ ):** Cân bằng giữa đa dạng và chính xác.

Scale ( $s$ )	FID (UFSC) ↓	
	$L \rightarrow C$	$C \rightarrow L$
2.5	<b>13.28</b>	<u>40.05</u>
5.0	<u>13.39</u>	<b>40.00</b>
7.5	13.55	41.12
10.0	13.78	44.74
12.5	14.78	47.15
15.0	17.01	52.76

→  $s$  thấp ( $\in [2.5, 7.5]$ ) cho kết quả tốt nhất.

# Phân tích độ nhạy

Đánh giá hiệu quả của chiến lược Tăng cường dữ liệu (Data Augmentation).

e. **Tăng cường dữ liệu:** So sánh mô hình khi dùng/không dùng kỹ thuật tăng cường dữ liệu.

Cấu hình	FID (UFSC) ↓	
	L → C	C → L
w/o Augmentation	<u>15.77</u>	<u>43.07</u>
w/ Augmentation	<b>13.55</b>	<b>41.12</b>

→ Việc áp dụng Augmentation giúp giảm đáng kể FID, chứng tỏ mô hình học được các đặc trưng phong cách **bền vững** hơn, tránh bị Overfitting.

**Chiến lược: Random Resized Crop**

- **Scale (0.8 – 1.0):** Cắt ngẫu nhiên nhưng giữ lại phần lớn cấu trúc chữ.
- **Ratio (0.8 – 1.2):** Thay đổi tỷ lệ khung hình nhẹ để mô phỏng các biến thể viết tay.

→ Giúp mô-đun **CL-SCR** không bị “học vẹt” (memorize) các vị trí pixel cố định.