

PHẦN 1: GIỚI THIỆU (SLIDE 1 - 6)

Slide 1: Xin chào

Kính thưa Hội đồng bảo vệ, quý Thầy Cô và các bạn. Em tên là Trần Đình Khánh Đăng. Hôm nay, em xin phép trình bày khoá luận tốt nghiệp với đề tài: “**Tăng cường khả năng chuyển kiểu chữ đa ngôn ngữ trong bài toán One-Shot bằng mô hình khuếch tán**”.

Slide 2 & 3: Mục lục

Bài báo cáo sẽ đi qua 4 phần chính: Từ việc đặt vấn đề, đi sâu vào phương pháp đề xuất, chứng minh bằng thực nghiệm và cuối cùng là kết luận.

Kính thưa hội đồng, chúng ta có thể thấy phông chữ hiện diện ở khắp mọi nơi, từ bao bì sản phẩm đến các biển hiệu quảng cáo. Nhu cầu về các bộ font chữ độc đáo, thẩm mỹ chưa bao giờ hạ nhiệt trong đời sống hiện đại.

Slide 4: Thách thức thiết kế truyền thống

Tuy nhiên, quy trình thiết kế font truyền thống đang gặp phải 3 rào cản rất lớn:

- Về Chi phí:** Tốn kém nhân lực và thời gian. Hiệu suất thấp do lặp lại thủ công.
- Về Quy mô:** Hệ Latin chỉ có 52 ký tự, nhưng Hán tự lên tới 50.000 chữ. Vẽ tay là bất khả thi.
- Về Đa ngữ:** Các ngôn ngữ ít tài nguyên (Low-resource) hoặc có dấu phức tạp như Tiếng Việt thường xuyên bị thiếu font đồng bộ.

Slide 5: Giải pháp One-shot

Để giải quyết, em sử dụng hướng tiếp cận **One-shot Font Generation**. Cơ chế của nó là: Máy chỉ cần nhìn **1 mẫu tham chiếu duy nhất** để trích xuất phong cách, sau đó **nhân bản** phong cách đó lên bất kỳ ký tự nào khác. Đây là lời giải toàn diện cho bài toán về tốc độ và quy mô.

Cơ chế One-shot hoạt động cụ thể như hình minh họa này:

- (Chỉ vào chữ Content):** Máy sẽ nhận đầu vào là **Nội dung** (ví dụ khung xương của một chữ Hán).
- (Chỉ vào chữ M - Reference):** Đồng thời, máy chỉ cần nhìn **1 mẫu tham chiếu duy nhất** (ví dụ chữ 'M' phong cách bong bóng này) để học phong cách.
- (Chỉ vào mũi tên và Kết quả):** Sau đó, nó tự động 'bóc' phong cách đó và 'phù' lên nội dung gốc.

- ➔ Kết quả là ta có một chữ Hán mang phong cách bong bóng hoàn toàn mới. Đây chính là **lời giải toàn diện** cho bài toán về tốc độ và quy mô mà em vừa nêu.

Slide 6: Mục tiêu & Đóng góp

Tuy nhiên, đa số mô hình hiện tại chỉ làm tốt đơn ngữ.

- **Mục tiêu của khoá luận:** Xây dựng giải pháp Cross-Lingual (Xuyên ngôn ngữ).
- **Về Phạm vi:** Em lựa chọn kiểm chứng trên cặp **Latin - Hán tự**. Lý do em chọn cặp này không chỉ vì sự phổ biến dữ liệu, mà quan trọng hơn, đây là cặp **đại diện tiêu biểu** và là **chuẩn so sánh** chung của các nghiên cứu cùng đề tài để đo lường khả năng chuyển đổi cấu trúc.
- **Đóng góp chính:**
 1. Xây dựng pipeline dựa trên **Diffusion Model**.
 2. Đề xuất mô-đun **CL-SCR** với cơ chế luồng đôi để xử lý vấn đề khác biệt cấu trúc.

Slide 7: Khoảng cách hình thái học (Morphological Gap)

Tại sao các nghiên cứu tiên tiến (SOTA) đều chọn cặp Latin - Hán tự làm chuẩn so sánh? Xin mời thầy cô nhìn vào trực quan này để thấy sự đối lập.

- **Latin:** Đầu tiên là hệ Latin. Xin thầy cô nhìn vào ký tự 'p' và 'e' ở đây. Thầy cô có thể thấy, chúng được sắp xếp chạy dọc theo một đường kẻ xanh gọi là **Baseline**. Cấu trúc của nó rất thoáng, các nét phát triển theo chiều ngang và tách biệt hẳn nhau. Đây là đặc trưng của cấu trúc **Tuyến tính**.
 - **Hán tự:** Ngược lại hoàn toàn là Hán tự. Hãy nhìn vào chữ '**Quốc**' (國) này. Nó bị 'nhốt' chặt trong một ô vuông cố định gọi là **Body frame**. Bên trong là sự chồng chéo tầng tầng lớp lớp: Bộ 'Vi' bao bên ngoài, rồi đến bộ 'Khâu', bộ 'Qua' chen chúc bên trong. Mật độ nét cực kỳ dày đặc và không có khoảng thở như bên Latin. Đây chính là cấu trúc **Khối vuông (Block)** đặc trưng.
- ➔ Sự khác biệt này tạo ra một **Khoảng cách miền dữ liệu** lớn về mặt không gian. Chính sự 'lệch pha' này khiến các phương pháp cũ (dựa trên GAN hay so khớp điểm ảnh) thường xuyên thất bại.

Đó là động lực để em áp dụng **Diffusion** kết hợp với **CL-SCR** để giải quyết bài toán này.

PHẦN 2: PHƯƠNG PHÁP ĐỀ XUẤT (SLIDE 8 - 13)

Slide 8: (Chuyển tiếp)

Sau đây là chi tiết phương pháp đề xuất.

Slide 9: Kiến trúc đề xuất

Về kiến trúc mô hình, em xây dựng giải pháp dựa trên nền tảng **FontDiffuser** nhưng có sự cải tiến trọng yếu ở Giai đoạn 2:

- **Phase 1 (Bên trái):** Ở Giai đoạn 1, em **kết thừa** hai mô-đun nền tảng là **MCA** (để giữ chi tiết nét) và **RSI** (để nắn chỉnh hình dáng). Mục tiêu của giai đoạn này là đảm bảo mô hình có khả năng '**Tái tạo cấu trúc**' (**Reconstruction**), tức là dựng được cái khung xương chữ cho đúng đắn.
- **Phase 2 (Trong khung đỡ):** Tuy nhiên, thay vì dùng mô-đun SCR gốc, em **đề xuất thay thế** bằng mô-đun **CL-SCR** (do em thiết kế). Mục tiêu là để nâng cấp khả năng học phong cách **Xuyên ngôn ngữ** (**Cross-Lingual**).

Slide 10: Động lực & Ý tưởng

Vậy tại sao Phase 1 là chưa đủ?

- Vấn đề cốt lõi là Phase 1 hoạt động dựa trên sự **Khớp nối không gian**. Tức là nó cố tìm vị trí tương đồng giữa chữ mẫu và chữ đích để tô màu. Ví dụ: Khi nhìn thấy chữ Latin mẫu (như chữ 'y') có cái đuôi ở góc dưới, mô hình sẽ cố tìm vị trí tương ứng trên chữ Hán (như chữ 'Nhất') để vẽ cái đuôi đó vào. Nhưng vì chữ 'Nhất' không có chân, mô hình bị '**mất điểm neo**'. Kết quả là nó vẫn cố chấp vẽ một vệt mờ ám vào khoảng không, tạo ra các lỗi '**bóng ma**' (**ghosting**) và làm bẩn cấu trúc chữ.
- Chính vì vậy, em cần Giai đoạn 2 với mô-đun **CL-SCR** để thay đổi hoàn toàn tư duy này. Cơ chế Luồng đôi của em ép mô hình phải tìm ra '**Mẫu số chung**' về phong cách. Nó dạy cho máy hiểu rằng: '*Đừng bắt chước cái đuôi nằm ở đâu, mà hãy học xem cái đuôi đó được vẽ bằng bút gì, độ đậm nhạt ra sao*'. Nhờ học được **bản chất nét bút** (**Style Invariant**) thay vì vị trí, mô hình có thể 'phủ' phong cách lên chữ Hán một cách sạch sẽ mà không gây ra các nét thừa thãi.

Slide 11: Kiến trúc CL-SCR (Chi tiết)

Cụ thể, CL-SCR sử dụng cơ chế giám sát luồng đôi:

- **Luồng Intra:** So sánh ảnh sinh với ảnh cùng ngôn ngữ để giữ bản sắc.
- **Luồng Cross (Quan trọng nhất):** So sánh ảnh sinh với ảnh ngôn ngữ đích để kéo phong cách lại gần nhau bất chấp khác biệt cấu trúc.

Slide 12: Công thức Loss (Kéo - Đẩy)

Về mặt toán học, em dùng hàm Loss InfoNCE với cơ chế **Kéo và Đẩy**:

- Tử số là lực **KÉO**: Kéo ảnh sinh về phía phong cách chuẩn.
- Mẫu số là lực **ĐÂY**: Đẩy nó ra xa khỏi các phong cách sai. Đặc biệt, em đặt trọng số beta (Cross) lớn hơn alpha (Intra) để ưu tiên việc học đa ngữ.

Slide 13: Hàm mục tiêu tổng quát

Tổng kết lại, hàm Loss toàn cục là sự kết hợp của 4 thành phần: MSE và Offset (để giữ cấu trúc từ Phase 1) và quan trọng nhất là **CL-SCR** (để tinh chỉnh phong cách ở Phase 2).

PHẦN 3: THỰC NGHIỆM & KẾT QUẢ (SLIDE 14 - 20)

Slide 14: (Chuyển tiếp)

Chuyển sang phần thực nghiệm.

Slide 15: Chiến lược Huấn luyện

Để hiện thực hóa kiến trúc trên, em đã tiến hành thực nghiệm trên 1 GPU **Tesla P100 16GB**. Lưu ý rằng mô-đun CL-SCR (phần đóng góp chính) đã được em **tiền huấn luyện** (pre-train) độc lập trước đó để mô hình hội tụ nhanh hơn khi ghép vào hệ thống.

Về quy trình, em áp dụng chiến lược '**Coarse-to-Fine**' (**Từ Thô đến Tinh**), chia làm 2 giai đoạn rõ rệt:

- **Giai đoạn 1 là Khởi tạo (Pre-train)**: Em huấn luyện mô hình qua 400.000 bước với tốc độ học (Learning Rate) khá lớn (10^{-4}). Hàm Loss ở đây là tổng hợp có trọng số (λ) của MSE, Content và Offset. \rightarrow Mục tiêu giai đoạn này đơn giản là để mô hình học cách 'dựng khung xương' (Skeleton) của chữ sao cho đúng nét, chưa cần quan tâm nhiều đến phong cách tinh tế.
- **Giai đoạn 2 là Tinh chỉnh (Fine-tune) - Đây là bước quan trọng nhất**: Lúc này, em giảm Learning Rate xuống 10 lần (10^{-5}) để mô hình học chậm lại và sâu hơn. Em kích hoạt hàm loss **CL-SCR** và đặc biệt là áp dụng kỹ thuật **Data Augmentation** (như cắt ảnh ngẫu nhiên) -> Mục tiêu là để mô hình không học vẹt pixel nữa, mà tập trung nắm bắt các **đặc trưng phong cách** (Style) phức tạp của bài toán đa ngữ.

Cuối cùng, nhờ sử dụng bộ giải **DPM-Solver++**, quá trình sinh ảnh (Inference) chỉ tốn 20 bước, đảm bảo tốc độ thực thi nhanh chóng.

Slide 16: Dữ liệu & Kịch bản

Em sử dụng bộ dữ liệu chuẩn 818 font song ngữ. Quan trọng nhất là kịch bản đánh giá: Em tập trung vào **UFSC (Unseen Font)** - tức là đưa vào font lạ hoàn toàn. Đây là thước đo **quan trọng** cho bài toán One-shot.

Slide 17: Thước đo đánh giá

Hệ thống đánh giá dựa trên:

- **Định lượng:** Tập trung vào chỉ số **FID** (càng thấp càng tốt) để đo độ chân thực.
- **Định tính:** So sánh mắt thường và User Study.
- Đặc biệt, em có áp dụng chiến lược **Data Augmentation** để giúp mô hình bền vững hơn

Slide 18: Kết quả Định lượng

Mời thầy cô nhìn vào bảng kết quả. Ở kịch bản khó nhất (UFSC):

- Baseline (FontDiffuser gốc) có FID khoảng 29.09.
- Phương pháp của em (**Ours**) giảm xuống còn **13.55**. Việc giảm hơn 50% sai số FID chứng tỏ mô hình của em sinh ảnh chân thực hơn rất nhiều.

Slide 19: Kết quả Định tính

Trực quan hơn, ở cột **Ours**, các nét xước của chữ Hán (Reference) được tái hiện cực kỳ sắc sảo trên chữ Latin. Trong khi đó, các phương pháp cũ (GAN) thường bị mờ hoặc mất nét.

Slide 20: Đánh giá người dùng

Khảo sát trên 21 người dùng cũng cho thấy gần **70%** bình chọn cho kết quả của mô hình đề xuất.

Slide 21: Phân tích hiệu quả

Để trả lời câu hỏi: '*Liệu việc thêm CL-SCR vào có thực sự tốt hơn không, hay chỉ làm hệ thống nặng nề thêm?*', kính mời thầy cô nhìn vào bảng phân tích cắt giảm (Ablation Study) này. Em đã kiểm thử 3 cấu hình: Chỉ có Phase 1 (M+R), Chỉ có CL-SCR, và Mô hình đầy đủ.

Luận điểm 1: Vai trò của Phase 1 (Nhìn vào dòng 2)

Đầu tiên, thầy cô nhìn Dòng 2. Khi em bỏ Phase 1 và chỉ dùng CL-SCR, chỉ số FID tăng vọt lên hơn 160 (rất tệ).

⇒ Điều này khẳng định: CL-SCR không thể hoạt động một mình. Nó bắt buộc phải có mạng nền tảng (Phase 1) để dựng khung xương chữ trước.

Luận điểm 2: Sự đánh đổi ở SFUC (Nhìn vào cột SFUC)

Tiếp theo, so sánh Dòng 1 (Baseline) và Dòng 3 (Ours).

Ở cột SFUC (Font đã biết), kết quả của em có thấp hơn nhẹ so với Baseline (FID 11.76 vs 10.90).

- **Lý do:** Phase 1 (Baseline) chỉ dùng MSE Loss nên có xu hướng '**học thuộc lòng**' (**Overfitting**) các pixel của tập train, do đó điểm số trên tập quen rất cao.
- Khi thêm CL-SCR, mô hình bị ép học từ duy trì tương tự, nên khả năng 'nhớ vẹt' giảm đi một chút. Nhưng đây là sự đánh đổi cần thiết để đạt được mục tiêu quan trọng hơn ở bên phải.

Luận điểm 3: Chiến thắng ở UFSC (Nhìn vào cột UFSC - Quan trọng nhất)

Giá trị thực sự nằm ở cột UFSC (Font lạ) - đây mới là mục tiêu của bài toán One-shot.

- **Chiều Latin sang Hán (\$L \rightarrow C\$):** Mô hình của em giảm FID từ 14.86 xuống **13.55**. Đây là sự cải thiện rõ rệt về khả năng **Tổng quát hóa (Generalization)**.
- **Chiều Hán sang Latin (\$C \rightarrow L\$):** Có một hiện tượng thú vị là chỉ số điểm ảnh (L1/SSIM) của em thấp hơn Baseline, NHƯNG chỉ số **FID** lại tốt hơn (41.11 so với 42.86).
 - Điều này chứng minh: Baseline cố gắng khớp từng điểm ảnh một cách máy móc (dẫn đến ảnh mờ).
 - Trong khi đó, mô hình của em ưu tiên **Độ chân thực phong cách (Perceptual Realism)** hơn, chấp nhận lệch pixel một chút để ảnh có 'hồn' hơn.

Tóm lại, Dòng 3 (Full Model) là cấu hình tối ưu nhất, chấp nhận hy sinh một chút ở tập quen (SFUC) để đạt hiệu suất cao nhất trên tập lạ (UFSC) và độ chân thực thị giác.

PHẦN 4: KẾT LUẬN (SLIDE 21 - 25)

Slide 23: Tổng kết đóng góp

Tổng kết lại, khoá luận đã hoàn thành 3 mục tiêu:

1. Giải quyết thành công bài toán One-shot Cross-Lingual (Latin-Hán).
2. Đóng góp kỹ thuật với mô-đun **CL-SCR**.
3. Hiệu quả thực nghiệm vượt trội SOTA.

Slide 24: Hạn chế & Hướng phát triển

Tuy nhiên, em cũng nhận thấy:

- **Hạn chế:** Tốc độ suy diễn còn chậm do bản chất khử nhiễu lặp của Diffusion (đây là sự đánh đổi để lấy chất lượng).

- **Hướng phát triển:** Em sẽ áp dụng kỹ thuật **Consistency Distillation** để tăng tốc, đồng thời mở rộng sang **Tiếng Việt (Thư pháp)** và sinh font dạng **Vector** để ứng dụng thực tế.

Slide 25: Công bố khoa học

Kết quả nghiên cứu đã được đúc kết thành bài báo khoa học và đang được review tại tạp chí *The Journal of Supercomputing*.

Slide 26: Lời cảm ơn

Em xin chân thành cảm ơn TS. Dương Việt Hằng đã hướng dẫn tận tình. Cảm ơn quý Thầy Cô đã lắng nghe. Em rất mong nhận được góp ý và câu hỏi của quý Thầy Cô.