Giới thiệu môn học

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA TOÁN – CƠ - TIN HỌC

Học phần:	**Nhập môn trí tuệ nhân tạo**
Mã học phần:	**MAT3508**
Học kỳ:	**Học kỳ 1 - 2025-2026**
Giảng viên:	**Hoàng Anh Đức**
Ngày nộp:	**08/12/2025**

Đề tài

ASL SIGN LANGUAGE RECOGNITION

Thành viên nhóm

Họ và tên	Github	MSV
Đào Ngọc Bảo	DNB-Bao1405	23001500
Mai Thế Cường	Matec76	23001504
Bùi Thanh Lâm	23001532-cloud	23001532
Tống Minh Phong	Mphong2005	23001545
Đỗ Thị Như Quỳnh	quynh2196	23001556

Nội dung chính

Nội dung

Bối cảnh

Theo Tổ chức Y tế Thế giới (WHO), hiện nay trên thế giới có khoảng 466 triệu người bị khiếm thính. Hàng triệu người sử dụng ngôn ngữ ký hiệu làm phương thức giao tiếp chính. Trong đó, ngôn ngữ ký hiệu ASL (American Sign Language) có khoảng 250,000–500,000 người sử dụng.

Đặc điểm của ngôn ngữ ký hiệu ASL

ASL là một ngôn ngữ tự nhiên hoàn chỉnh, không phụ thuộc vào hình thức nói hay viết và được cấu thành từ năm yếu tố cơ bản

- Hình thái bàn tay
- Chuyển động
- Vị trí
- Định hướng
- Yếu tố phi thủ công

Những thách thức kỹ thuật

- Chuỗi thời gian: ngôn ngữ ký hiệu ASL là chuỗi chuyển động dài, cần phân loại chuỗi thời gian thay vì ảnh đơn lẻ.
- Biến thể cá nhân: tốc độ, biên độ, phong cách khác nhau; mô hình phải tổng quát hóa.
- Biến thể môi trường: ánh sáng, góc camera, khoảng cách, trang phục, nền; mô hình cần tách ý nghĩa ký hiệu khỏi yếu tố không liên quan.

Những thách thức kỹ thuật

- Xử lý thời gian thực: <3–4 FPS gây khó chịu, 10–15 FPS chấp nhận được, 20–30 FPS tự nhiên.
- Chuỗi dài và phụ thuộc dài hạn: ký hiệu kéo dài nhiều khung hình, nên mô hình phải nắm được mối quan hệ dài hạn giữa các thời điểm để hiểu đúng ý nghĩa

Các phương pháp hiện đại

Các phương pháp nhận dạng ASL hiện nay gồm

- CNN: trích xuất đặc trưng không gian, nhanh với ảnh, nhưng lớn, chậm suy luận và nhạy môi trường.
- RNN/LSTM: xử lý chuỗi tốt, giảm vanishing gradient, nhưng chậm do xử lý tuần tự.
- Bộ xương (skeleton): dữ liệu nhẹ, bền môi trường, nhưng phụ thuộc chất lượng phát hiện.
- MediaPipe + học sâu: phát hiện tay, mặt, cơ thể nhanh, hiệu quả, chỉ cần RGB.

Phương pháp tiếp cận

Nghiên cứu sử dụng Transformer kết hợp MediaPipe Holistic: trích xuất và chọn lọc các điểm mốc, chuẩn hóa, sau đó sử dụng Transformer nhiều tầng để phân loại ký hiệu ASL. Transformer được chọn vì xử lý song song, nắm bắt phụ thuộc dài hạn, hiệu quả trên GPU và dễ mở rộng, vượt trội so với LSTM.

Đóng góp chính

Nghiên cứu có ba đóng góp chính

- Transformer tùy chỉnh kết hợp MediaPipe: chọn lọc điểm mốc, chuẩn hóa, dùng nhiều tầng mã hóa với positional embedding để giảm nhiễu và tăng khả năng tổng quát hóa.
- Tối ưu cho triển khai thực tế: mô hình nhẹ, chạy được trên laptop, thiết bị di động và thiết bị nhúng với tốc độ suy luận nhanh.
- Đường ống đầu-cuối hoàn chỉnh: từ webcam $\rightarrow$ phát hiện MediaPipe $\rightarrow$ tiền xử lý $\rightarrow$ suy luận $\rightarrow$ hậu xử lý $\rightarrow$ xuất văn bản/giọng nói. Điều này chứng minh tính khả thi của giải pháp, không chỉ dựa trên lý thuyết.

Mục tiêu và phạm vi

- Mục tiêu: đạt độ chính xác cao, hiệu năng thời gian thực, mô hình bền vững trước biến thể, và xây dựng nguyên mẫu tích hợp chuyển văn bản thành giọng nói với giao diện trực quan.
- Phạm vi: gồm các ký hiệu phổ biến, ký hiệu đơn lẻ từ video webcam, huấn luyện trên nhiều người.

Nội dung

Nguồn dữ liệu

Bộ dữ liệu đến từ cuộc thi Kaggle "Google - Isolated Sign Language Recognition (2023)". Gồm 94.477 video của 250 ký hiệu ASL, được chuyển thành tệp parquet chứa tọa độ điểm mốc (thay vì video RGB). Mỗi video tối đa 384 frames, 30 FPS, thu từ hơn 100 người với đặc điểm đa dạng. Dữ liệu là video thực tế, không tổng hợp, và được người dùng ASL bản địa xác thực để đảm bảo độ chính xác.

Lựa chọn và cân bằng dữ liệu

Do 250 lớp của bộ dữ liệu gốc là quá lớn và không đồng đều, nhóm tiến hành lấy mẫu phân tầng, chọn 10 lớp có số lượng mẫu lớn nhất và đảm bảo sự cân bằng giữa các lớp (chênh lệch tối đa chỉ 11 mẫu). Dữ liệu được chia theo tỷ lệ chuẩn:

- Training: 3,264 mẫu (80%)
- Validation: 817 mẫu (20%)
- Random seed: 2176 (đảm bảo kết quả tái lập)

Tổng cộng 4,081 mẫu, phân bố đồng đều ở 10 lớp như: listen, look, shhh, donkey, mouse, duck, hear, uncle, pretend, và bird. Mỗi lớp chiếm khoảng 9.9–10.2% tổng dữ liệu.

Trích xuất và xử lý đặc trưng điểm mốc

MediaPipe Holistic là một đường ống của Google sử dụng các mô hình học sâu:

- BlazeFace: Phát hiện khuôn mặt
- BlazeLand: Phát hiện các điểm mốc khuôn mặt
- BlazePose: Phát hiện tư thế cơ thể
- BlazeLand: Phát hiện các điểm mốc bàn tay

Mỗi điểm mốc là một điểm 3D (x, y, z) với độ tin cậy hiện diện (0-1).

Trích xuất và xử lý đặc trưng điểm mốc

Từ tổng 543 điểm mốc, nhóm chỉ chọn 173 điểm quan trọng cho ASL để giảm nhiễu và giảm chiều dữ liệu từ 1,629 xuống 346 chiều (giảm 79%). Việc chọn lọc giúp mô hình tập trung vào thông tin cốt lõi và giảm overfitting. Các nhóm điểm mốc được giữ lại gồm 173 điểm (27.8% toàn bộ). Trong đó: tay trái (21), tay phải (21), môi (39), mắt (32), lông mày (20), mũi (4), và viền khuôn mặt (36).

Tiền xử lý dữ liệu

Chuẩn hóa z-score: Mô hình cần học "cấu trúc tương đối" của ký hiệu, không phải vị trí tuyệt đối.

- Bước 1: Tính trung bình tham chiếu sử dụng điểm mốc mũi (chỉ số 17)
- Bước 2: Tính thống kê: = tọa độ mũi, = độ lệch chuẩn (tất cả 173 điểm mốc)
- Chuẩn hóa mỗi điểm mốc:

$$x_{\text{chuẩn hoá}} = \frac{x - \mu}{\sigma + \epsilon}, \quad y_{\text{chuẩn hoá}} = \frac{y - \mu}{\sigma + \epsilon}$$

  với = 108 để tránh chia cho không
- Bước 4: Xử lý NaN - Thay thế tất cả NaN bằng 0.

Tiền xử lý dữ liệu

Kết quả

- Chiều đầu ra: 173 điểm mốc $\times$ 2 (chỉ x,y) = 346 đặc trưng
- Hình dạng mỗi khung hình: (346,)
- Tọa độ chuẩn hóa về khoảng [2, 2]
- Bất biến với vị trí tuyệt đối, tỷ lệ, dịch chuyển

Tiền xử lý dữ liệu

Xử lý độ dài chuỗi: Độ dài video trong dữ liệu không đồng đều, nên chọn 384 khung hình làm độ dài tối đa (theo phân vị 99%).

- Video có $<$ 384 khung hình được đệm thêm số 0 ở cuối (post-padding)
- Video có $>$ 384 khung hình được cắt bớt khung hình cuối (<1%)

Đầu vào cuối cùng: có dạng (batch_size, 384, 346) với kiểu float32, mỗi mẫu chiếm khoảng 533 KB.

Tổng quan

Mô hình sử dụng kiến trúc bộ mã hóa Transformer với các thành phần chính:

- Tiền xử lý
- Mã hóa vị trí
- 4 khối Transformer Encoder để trích đặc trưng.
- Global Average Pooling để gộp đặc trưng theo chiều chuỗi.
- Đầu phân loại (MLP)

Mã hóa vị trí (Positional Encoding)

Transformer không có khái niệm về thứ tự như RNN, nên cần thêm mã hóa vị trí để mô hình nhận biết vị trí của từng khung hình

$$\text{PE(position, 2i)} = \sin\left(\frac{\text{position}}{10000^{2i/d_{\text{model}}}}\right) \tag{1}$$

$$\text{PE(position, 2i+1)} = \cos\left(\frac{\text{position}}{10000^{2i/d_{\text{model}}}}\right) \tag{2}$$

- **position** $\in [0, 383]$: vị trí khung hình
- **i** $\in [0, 127]$: chỉ số chiều
- $d_{\text{model}} = 256$: chiều vector nhúng

Trực quan:

- Mỗi chiều sử dụng một tần số khác nhau
- Tạo vector mã hóa vị trí độc nhất cho từng khung hình
- Mô hình học được các quan hệ phụ thuộc theo thời gian

Cơ chế tự chú ý đa đầu

Multi-Head Self-Attention cho phép mỗi vị trí "chú ý" đến tất cả các vị trí khác. Cho đầu vào $X \in \mathbb{R}^{T \times d}$, tính:

$$Q = XW^Q, \quad K = XW^K, \quad V = XW^V$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$

- Q (Query): "Cái gì tôi đang tìm kiếm?"
- K (Key): "Tôi là cái gì?"
- V (Value): "Thông tin tôi mang theo"
- $d_k = 64$: chiều đầu
- $\sqrt{d_k}$: chuẩn hóa để ổn định gradient

Cơ chế tự chú ý đa đầu

Đa đầu: Thay vì 1 đầu, sử dụng 4 đầu song song:

$$\text{đầu}_i = \text{Chú ý}(QW_i^Q, KW_i^K, VW_i^V)$$
$$\text{Đa đầu}(Q, K, V) = \text{Ghép nối}(\text{đầu}_1, ..., \text{đầu}_4)W^O$$

Lợi ích:

- Đầu 1 tập trung vào chuyển động tay
- Đầu 2 tập trung vào biểu cảm khuôn mặt
- Đầu 3 tập trung vào quan hệ không gian
- Đầu 4 tập trung vào các mẫu thời gian

Mạng truyền thẳng

Sau cơ chế tự chú ý đa đầu, mỗi vị trí độc lập đi qua mạng truyền thẳng:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2$$

Thông số:

- $W_1 \in \mathbb{R}^{256\times128}$: chiếu từ $256 \to 128$
- $W_2 \in \mathbb{R}^{128\times256}$: chiếu từ $128 \to 256$
- Hàm kích hoạt ReLU: $\text{ReLU}(z) = \max(0, z)$

Chuẩn hóa tầng và kết nối dư

Chuẩn hóa Tầng (Layer Normalization):

$$\text{LayerNorm}(x) = \gamma \frac{x - \text{trung bình}(x)}{\sqrt{\text{phương sai}(x) + \epsilon}} + \beta$$

Kết nối Dư (Residual Connection):

$$x_{\text{đầu ra}} = x_{\text{đầu vào}} + \text{Tầng phụ}(x_{\text{đầu vào}})$$

Lợi ích:

- Chuẩn hóa tầng - LayerNorm: ổn định huấn luyện
- Kết nối dư - Residual Connection: giúp luồng gradient, tránh tiêu biến gradient
- Kết hợp: mô hình 4 tầng vẫn huấn luyện ổn định

Xếp chồng các khối mã hóa

Chúng tôi lựa chọn xếp chồng 4 khối mã hóa giống hệt nhau vì:

- 2 khối: Không đủ khớp (84.3% độ chính xác)
- 4 khối: Tối ưu (87.4% độ chính xác)
- 6 khối: Quá khớp (87.8% độ chính xác, nhưng tốn gấp đôi thời gian huấn luyện)

Gộp Trung bình Toàn cục và Đầu Phân loại

Sau 4 khối mã hóa, đầu ra có kích thước $(\text{batch}, 384, 256)$. Để phân loại, cần 1 vector cố định cho mỗi mẫu:

$$\text{gộp} = \frac{1}{384} \sum_{t=1}^{384} x_t$$

Kết quả: $(\text{batch}, 384, 256) \rightarrow (\text{batch}, 256)$. Các tầng cuối cùng để dự đoán lớp:

$$\text{Input: } (\text{batch}, 256) \rightarrow \text{Dense}(128) + \text{ReLU} \rightarrow (\text{batch}, 128)$$

$$\rightarrow \text{Dropout}(0.4) \rightarrow \text{Dense}(10) + \text{Softmax} \rightarrow (\text{batch}, 10)$$

$$\rightarrow \text{Output: Phân bố xác suất}$$

Cấu hình chi tiết

Thành phần	**Giá trị**
Chiều nhúng (d_{model})	256
Số tầng mã hóa	4
Số đầu chú ý	4
Chiều đầu	64
Chiều truyền thẳng (FFN)	128
Dropout mã hóa	0.2
Dropout MLP	0.4
Số lớp đầu ra	10
Tổng tham số	1,443,978
Kích thước mô hình	5.51 MB

Bảng: Thông số kiến trúc mô hình Transformer nhỏ gọn

Rất nhỏ gọn so với BERT-base (110M), ResNet-50 (25M), VGG-16 (138M).

Hàm Mất mát

Entropy chéo phân loại (Categorical Cross-Entropy)

$$L = -\sum_{i=1}^{C} y_i^{\text{mượt}} \log \hat{y}_i$$

Với làm mượt nhãn (Label Smoothing):

$$y_i^{\text{mượt}} = \begin{cases} 1-\alpha, & \text{nếu } i \text{ là lớp đúng} \\ \frac{\alpha}{C-1}, & \text{các lớp còn lại} \end{cases}$$

Ví dụ $\alpha = 0.1$, $C = 10$:

- Lớp đúng: 0.9 thay vì 1.0
- Các lớp khác: 0.01 thay vì 0.0

Lợi ích:

- Ngăn chặn quá tự tin
- Điều chuẩn (regularization)
- Tổng quát hóa tốt hơn

Bộ tối ưu hóa

Bộ tối ưu hóa Adam:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\nabla_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)\nabla_t^2$$

$$\theta_t = \theta_{t-1} - \alpha \frac{m_t}{\sqrt{v_t} + \epsilon}$$

Mặc định: $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$

Tốc độ học

Giảm tốc độ học khi đạt ngưỡng (ReduceLROnPlateau):

$$\text{LR}_{\text{mới}} = \text{LR}_{\text{cũ}} \times \text{hệ số}; \quad \text{nếu mất mát kiểm định không cải thiện}$$

Cấu hình

- Tốc độ học ban đầu $= 10^{-4}$
- Hệ số $= 0.5$
- Kiên nhẫn (Patience) $= 3$ epoch
- Tốc độ học tối thiểu $= 10^{-6}$

Cấu hình huấn luyện

Bảng: Hyperparameters huấn luyện mô hình

Tham số	**Giá trị**
Batch size	64
Số epoch	100 (dừng sớm ở epoch 68)
Kiên nhẫn dừng sớm	10 epoch
Bộ tối ưu hóa	Adam
Tốc độ học ban đầu	10^{-4}
Tốc độ học tối thiểu	10^{-6}
Hệ số giảm LR	0.5
Kiên nhẫn giảm LR	3
Làm mượt nhãn	0.1
Dropout mã hóa	0.2
Dropout MLP	0.4

Kết quả huấn luyện

Bảng: Kết quả huấn luyện mô hình

Chỉ số	**Tập Huấn luyện**	**Tập Kiểm định**
Độ chính xác cuối cùng	89.7%	87.39%
Mất mát cuối cùng	0.786	0.804
Tổng số epoch	68 / 100	—

Tốc độ học (thực tế)

Epoch	**Tốc độ học**	**Sự kiện**	**Ghi chú**
1–33	10^{-4}	—	Cải thiện nhanh
34–42	5×10^{-5}	Giảm lần 1	Mất mát kiểm định đạt ngưỡng
43–48	2.5×10^{-5}	Giảm lần 2	Tiếp tục đạt ngưỡng
49–53	1.25×10^{-5}	Giảm lần 3	Cải thiện chậm
54–61	6.25×10^{-6}	Giảm lần 4	Tiếp tục học
62–64	3.125×10^{-6}	Giảm lần 5	Gần tốc độ học tối thiểu
65–68	1.56×10^{-6}	Giảm lần 6	Dừng sớm

Tổng: 6 lần giảm tốc độ học trong 68 epoch

Phân tích thời gian huấn

Chỉ số	**Thời gian**	**Ghi chú**
Thời gian trung bình mỗi epoch	$\sim$ 407 giây	$\sim$ 6.8 phút
Epoch nhanh nhất	$\sim$ 380 giây	Các epoch đầu
Epoch chậm nhất	$\sim$ 440 giây	Các epoch giữa
Tổng thời gian huấn luyện	68×6.8 phút	$\sim$ 7.7 giờ

Phân tích bộ nhớ:

- Trọng số mô hình: $1.44\text{M} \times 4$ byte $= 5.8$ MB
- Trạng thái Adam (m, v): $2 \times 1.44\text{M} \times 4 = 11.5$ MB
- Dữ liệu lô $(64 \times 384 \times 346 \times 4) \approx 34$ MB
- Tổng VRAM: $\sim$ 60–80 MB ($\sim$ 0.5% của 16 GB)

Nội dung

Môi trường Thử nghiệm

Quá trình huấn luyện và đánh giá mô hình được thực hiện trên môi trường đám mây Kaggle.

- **Phần cứng sử dụng:**
  - GPU: NVIDIA Tesla P100-PCIE (16GB VRAM).
  - CPU: Intel Xeon (2 core, 2.20GHz).
  - RAM: 13GB (System Memory).
- **Phần mềm và thư viện:**
  - Ngôn ngữ và Thư viện: Python 3.9+, TensorFlow 2.x
  - Môi trường tính toán: CUDA Toolkit 12.8 (Driver Version 570.172.08).
  - Xử lý hình ảnh: MediaPipe $\geq$ 0.10.0.

Các chỉ số đánh giá I

Hiệu suất của mô hình được đánh giá trên tập kiểm tra độc lập thông qua **Ma trận Nhầm lẫn** và các chỉ số sau.

- **Định nghĩa cơ bản** (Trong đó TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative):
- **Accuracy (Độ chính xác)**: Tỷ lệ tổng số ký hiệu được phân loại đúng trên tổng số mẫu.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **Precision (Độ chuẩn xác)**: Đo lường khả năng tránh Positive giả, tức là tỷ lệ các dự đoán Positive là chính xác.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Các chỉ số đánh giá II

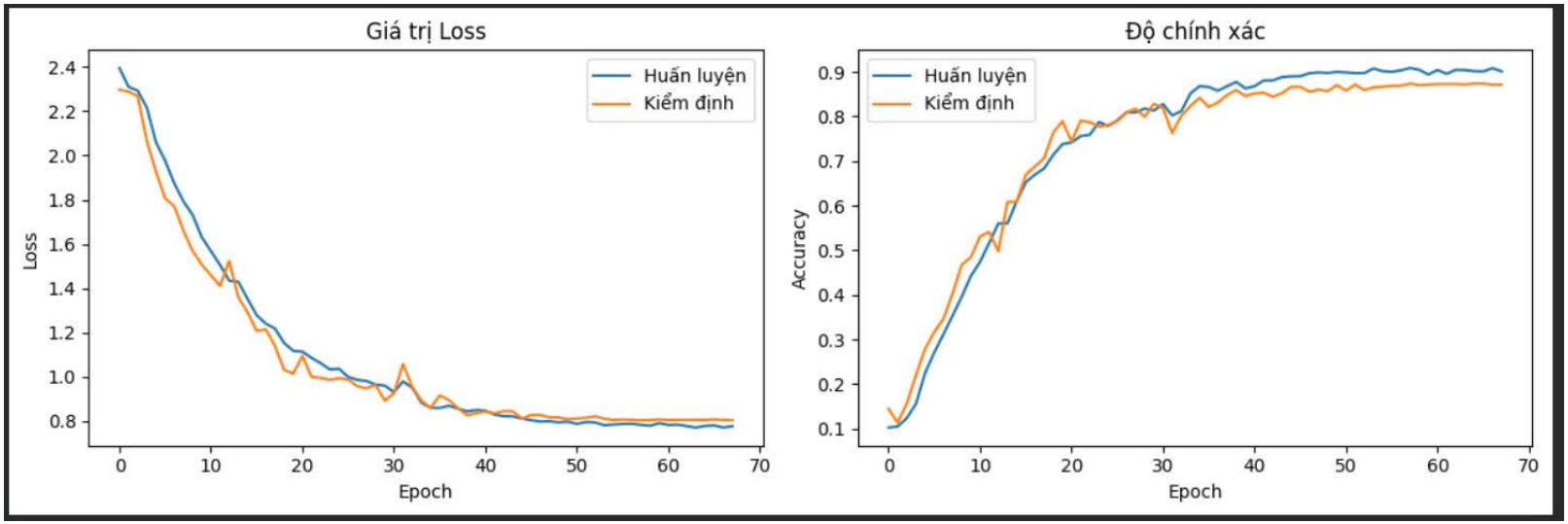
- **Recall (Độ nhạy)**: Đo lường khả năng tìm kiếm Positive thật, tức là tỷ lệ các mẫu Positive thực tế được dự đoán đúng.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1-score**: Là trung bình điều hòa (*Harmonic Mean*) của Precision và Recall. Chỉ số này cung cấp đánh giá cân bằng và là chỉ số chính khi dữ liệu có sự mất cân bằng giữa các lớp.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Biểu đồ đường cong huấn luyện I



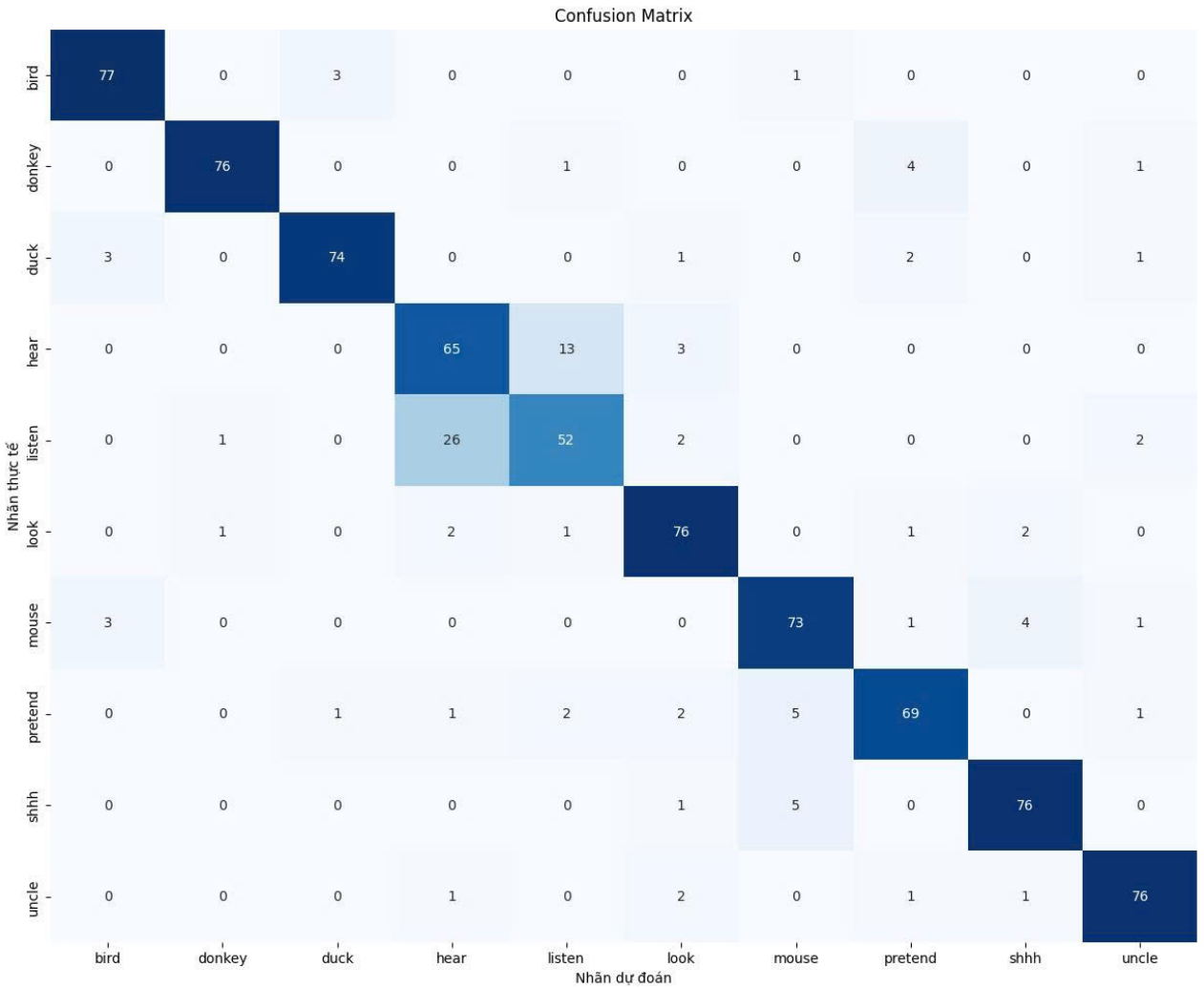
Hình: Diễn biến giá trị Loss và Độ chính xác (Accuracy) qua các epoch

Biểu đồ đường cong huấn luyện II

Quan sát đồ thị Hình 3.1, quá trình huấn luyện có thể được chia thành các giai đoạn cụ thể như sau:

- **Hội tụ nhanh (0 - 20 epoch):** Loss giảm sâu ($> 2.4 \rightarrow < 1.0$), Accuracy tăng vọt ($10\% \rightarrow 70\%$) nhờ Learning Rate phù hợp.
- **Tổng quát hóa tốt:** Đường Validation bám sát Training (Accuracy $\sim 90\%$), khẳng định mô hình không bị Overfitting.
- **Điểm bão hòa (Sau epoch 50):** Các chỉ số đi vào ổn định. Việc dừng tại epoch 70 (chiến lược *Early Stopping*) là tối ưu để tiết kiệm tài nguyên.

Ma trận nhầm lẫn



Hình: Ma trận nhầm lẫn của mô hình Transformer trên tập kiểm tra

Phân tích Lỗi và Nguồn Gốc Nhầm lẫn I

Lỗi tập trung chủ yếu ở các cặp ký hiệu sau:

- **Hình thái tay Tương đồng**: Các ký hiệu chỉ khác nhau ở chi tiết nhỏ trong hình thái ngón tay (ví dụ: *'Hear'* và *'Listen'*), là nguyên nhân chính gây nhầm lẫn. Lỗi này có thể do:
  1. Trong thực tế Keypoints 3D từ MediaPipe không đủ phân giải để phân biệt các chi tiết ngón tay nhỏ .
  2. Cử chỉ bị che khuất một phần trong quá trình ký hiệu.
  3. Chuyển động tay quá nhanh dẫn đến thiếu frame quan trọng

- **Ký hiệu Tĩnh (Static Signs)**: Các ký hiệu không có chuyển động rõ ràng phụ thuộc nhiều hơn vào hình thái tay, khiến việc phân biệt trở nên khó khăn hơn.

So sánh với các Phương pháp Khác I

So sánh đối chứng với các mô hình Deep Learning phổ biến trong xử lý chuỗi thời gian, gồm **CNN-LSTM** và **CNN-GRU**

Mô hình	**Đặc trưng**	**Accuracy (%)**	**Tốc độ (FPS)**
CNN-LSTM (Baseline)	Keypoints 3D	79.5	18
CNN-GRU	Keypoints 3D	82.1	21
Transformer (Đề xuất)	**Keypoints 3D**	**87.4**	**30**

- **Accuracy:** Transformer chính xác cao hơn **7.9%** so với LSTM. Do cơ chế *Self-Attention*, cho phép mô hình nắm bắt các mối quan hệ phụ thuộc dài hạn (*long-term dependencies*) trong toàn bộ chuỗi cử chỉ mà không bị mất thông tin như (RNNs).

So sánh với các Phương pháp Khác I

So sánh đối chứng với các mô hình Deep Learning phổ biến trong xử lý chuỗi thời gian, gồm **CNN-LSTM** và **CNN-GRU**

Mô hình	**Đặc trưng**	**Accuracy (%)**	**Tốc độ (FPS)**
CNN-LSTM (Baseline)	Keypoints 3D	79.5	18
CNN-GRU	Keypoints 3D	82.1	21
Transformer (Đề xuất)	**Keypoints 3D**	**87.4**	**30**

- **Tối ưu hóa thời gian thực (Inference Speed):** Tốc độ xử lý đạt **30 FPS**, cao gấp 1.6 lần so với LSTM (18 FPS). Khác với LSTM phải xử lý tuần tự từng frame ($t \rightarrow t+1$), Transformer cho phép tính toán song song trên GPU, tận dụng triệt để tài nguyên phần cứng

Đánh giá Pipeline End-to-End

- **Môi trường triển khai:** Hệ thống chạy mượt trên thiết bị cá nhân (Local Deployment) với cấu hình NVIDIA Tesla P100.
- **Độ trễ tối ưu:** Độ trễ hệ thống rất thấp ($48 - 62$ ms), nằm dưới ngưỡng nhận thức của con người (≈ 100ms).
- **Thông lượng cao:** Duy trì ổn định $20 - 25$ FPS (720p), đảm bảo phản hồi tức thì và trải nghiệm mượt mà cho chuỗi ký hiệu liên tục.

Đánh giá Độ bền vững trong Môi trường Thực tế I

Bảng: Hiệu năng mô hình dưới các điều kiện môi trường khác nhau trên 100 mẫu ngẫu nhiên

Điều kiện	**Mô tả**	**Độ chính xác**
Ánh sáng chuẩn	Phòng đủ sáng, không ngược sáng	87.0%
Thiếu sáng	Phòng tối, chỉ có ánh sáng màn hình	81.5%
Góc nghiêng	Người dùng quay góc 30° so với camera	76.0%
Khoảng cách xa	Khoảng cách $> 1.5m$ so với webcam	83.2%

Đánh giá Độ bền vững trong Môi trường Thực tế II

Nhận xét:

- **Ổn định khi thiếu sáng:** Hệ thống chỉ giảm nhẹ hiệu suất ($\approx 5.5\%$) nhờ khả năng trích xuất đặc trưng mạnh mẽ của MediaPipe.
- **Hạn chế góc quay:** Hiệu suất giảm đáng kể ở góc nghiêng 30°. Điều này cho thấy sự cần thiết của việc bổ sung dữ liệu đa góc nhìn (multi-view dataset) trong tương lai.

Hạn chế của Hệ thống Hiện tại I

- **Phụ thuộc chất lượng Keypoints:** Độ chính xác bị ảnh hưởng lớn bởi đầu ra của MediaPipe. Đặc biệt khi tay chuyển động nhanh (*Motion Blur*) hoặc bị che khuất (*Self-occlusion*), vector đặc trưng dễ bị nhiễu.

- **Giới hạn quy mô từ vựng:** Mô hình hiện tại tối ưu trên tập từ vựng đóng (50-100 từ). Việc mở rộng lên quy mô thực tế gặp thách thức lớn về sự trùng lặp đặc trưng (*Feature Overlap*).

Hạn chế của Hệ thống Hiện tại II

- **Thiếu ngữ cảnh và NLP:** Hệ thống dừng lại ở mức nhận diện từ đơn lẻ (*Isolated Signs*). Do thiếu module NLP, máy chưa thể ghép từ thành câu hoàn chỉnh hoặc hiểu ngữ pháp không gian.

- **Chưa khai thác yếu tố phi thủ công:** Các biểu cảm khuôn mặt (nhướn mày, phồng má) đóng vai trò quyết định ngữ khí trong ASL nhưng chưa được mô hình hóa triệt để.

Nội dung

Kết luận về Dự án I

Dự án xây dựng thành công hệ thống **Nhận diện ASL Thời gian thực** dựa trên Transformer và MediaPipe với các kết quả chính:

1. **Hệ thống End-to-End hoàn chỉnh:** Quy trình khép kín từ hình ảnh đầu vào $\rightarrow$ trích xuất đặc trưng $\rightarrow$ nhận diện $\rightarrow$ giọng nói (TTS). Độ chính xác đạt **87.4%**.

2. **Hiệu suất vượt trội:** Kiến trúc Transformer vượt qua CNN-LSTM: Độ chính xác tăng **+7.9%**, tốc độ suy luận tăng **67%** (đạt **30 FPS**).

3. **Tính thực tiễn cao:** Hệ thống vận hành ổn định, độ trễ thấp (Real-time) trên phần cứng phổ thông, sẵn sàng ứng dụng hỗ trợ người khiếm thính.

Ý nghĩa khoa học và xã hội I

- **Về Khoa học:** Nghiên cứu khẳng định hiệu quả của hướng tiếp cận *Skeleton-based* kết hợp cơ chế *Self-Attention*. Đây là giải pháp cân bằng tốt giữa độ chính xác cao và chi phí tính toán thấp, khắc phục nhược điểm của các phương pháp dựa trên ảnh RGB truyền thống.

- **Về Xã hội:** Dự án góp phần xóa bỏ rào cản giao tiếp, hỗ trợ người khiếm thính tiếp cận tốt hơn với các dịch vụ công nghệ, giáo dục và y tế, từ đó thúc đẩy sự **bình đẳng** và hòa nhập trong cộng đồng.

Nhận dạng Liên tục (CSLR)

Mục tiêu

Chuyển đổi từ nhận dạng từ đơn lẻ (Isolated) sang cấp độ câu và đoạn văn hoàn chỉnh.

- **Mở rộng bài toán:** Xử lý chuỗi ký hiệu liên tục (*Continuous Sign Language Recognition*) trong giao tiếp tự nhiên.
- **Kiến trúc Encoder-Decoder:** Ứng dụng Transformer cho bài toán dịch thuật (Machine Translation).
- **Giải quyết gióng hàng (Alignment):** Tích hợp hàm mất mát **CTC Loss** để ánh xạ chuỗi video sang văn bản mà không cần gán nhãn từng frame.

Tối ưu yếu tố Phi thủ công

- **Khai thác đa kênh thông tin:** Tích hợp đặc trưng biểu cảm khuôn mặt (lông mày, miệng) và hướng đầu - yếu tố quyết định ngữ pháp trong ASL.

- **Kiến trúc Multi-stream Fusion:** Kết hợp hai luồng dữ liệu song song:
  - **Hand stream:** Cử chỉ tay.
  - **Face stream:** Biểu cảm mặt.

  $\rightarrow$ Sử dụng cơ chế *Attention Fusion* để tổng hợp kết quả.

Triển khai trên Thiết bị biên

Hướng tới phổ cập ứng dụng trên thiết bị di động cá nhân (Smartphone/IoT):

- **Kỹ thuật nén mô hình:** Áp dụng **Quantization (INT8)**, **Pruning** (cắt tỉa) và **Knowledge Distillation** (chưng cất tri thức).
- **Tương thích đa nền tảng:** Chuyển đổi sang định dạng **TensorFlow Lite** hoặc **ONNX**.
- **Hiệu suất mục tiêu:** Chạy mượt trên Android/iOS không cần GPU rời, kích thước Model < 30**MB**.

Mở rộng và Bản địa hóa

Tầm nhìn

Không chỉ dừng lại ở ASL mà hướng tới các ngôn ngữ ký hiệu địa phương.

- **Hỗ trợ VSL (Việt Nam):** Thu thập dữ liệu và áp dụng kỹ thuật *Transfer Learning* từ mô hình ASL sẵn có để giảm thời gian huấn luyện.
- **Dịch song ngữ 2 chiều:** Phát triển tính năng linh hoạt:

$$\text{Ngôn ngữ Ký hiệu} \longleftrightarrow \text{Ngôn ngữ Nói/Văn bản}$$

Tích hợp GenAI LLM

Tận dụng sức mạnh của các Mô hình Ngôn ngữ Lớn (GPT-4, Llama 3, Gemini):

- **Hậu xử lý (Post-processing):** Dùng LLM để sửa lỗi ngữ pháp, sắp xếp lại trật tự từ trong câu dịch thô.
- **Hội thoại tự nhiên:** Tạo ra các phản hồi mạch lạc, giàu cảm xúc và đúng ngữ cảnh, vượt xa việc dịch "word-by-word" khô khan.

Bảo mật và Riêng tư

- **Ưu điểm của Skeleton-based:** Hệ thống chỉ lưu trữ và truyền tải vector tọa độ (x, y, z), **hoàn toàn không** chứa dữ liệu hình ảnh RGB.
- **Ẩn danh tuyệt đối:** Đảm bảo quyền riêng tư (Privacy-Preserving).
- **An toàn khi sử dụng:** Người dùng có thể yên tâm sử dụng camera trong các không gian nhạy cảm/riêng tư mà không lo lộ lọt hình ảnh cá nhân.

THANK YOU FOR YOUR ATTENTION!

"Công nghệ không chỉ là những dòng code khô khan, mà còn có thể là nhịp cầu kết nối giữa im lặng và thanh âm,xóa nhòa ranh giới giữa chúng ta và những người khiếm khuyết, để không ai bị bỏ lại phía sau..."

- Mai Cheng Cheng -

Q & A

Rất mong nhận được câu hỏi và đóng góp ý kiến từ Quý thầy cô.