

Luận văn tốt nghiệp đại học

Đại học Bách Khoa Thành phố Hồ Chí Minh

Khoa Khoa Học và Kỹ Thuật Máy Tính

{phantanphuc2512, buikhanhngoc142}@gmail.com

Nhận dạng biểu thức toán học

Hội đồng : Khoa học máy tính

Giảng viên hướng dẫn : TS. Lê Thành Sách

Giảng viên phản biện : TS. Nguyễn Đức Dũng

Nhóm sinh viên thực hiện : Phan Trần Phúc - 51303058
Bùi Khánh Ngọc - 51302567

Tp. Hồ Chí Minh, Tháng 01/2018

Overview

① Giới thiệu

② Công trình liên quan

③ Mô hình đề xuất

④ Hiện thực và đánh giá

⑤ Tổng kết

Overview

① Giới thiệu

② Công trình liên quan

③ Mô hình đề xuất

④ Hiện thực và đánh giá

⑤ Tổng kết

Overview

- ① Giới thiệu
- ② Công trình liên quan
- ③ Mô hình đề xuất
- ④ Hiện thực và đánh giá
- ⑤ Tổng kết

Overview

- ① Giới thiệu
- ② Công trình liên quan
- ③ Mô hình đề xuất
- ④ Hiện thực và đánh giá
- ⑤ Tổng kết

Overview

- ① Giới thiệu
- ② Công trình liên quan
- ③ Mô hình đề xuất
- ④ Hiện thực và đánh giá
- ⑤ Tổng kết

Mục lục

① Giới thiệu

- Đặt vấn đề
- Phạm vi đề tài
- Lý do chọn đề tài

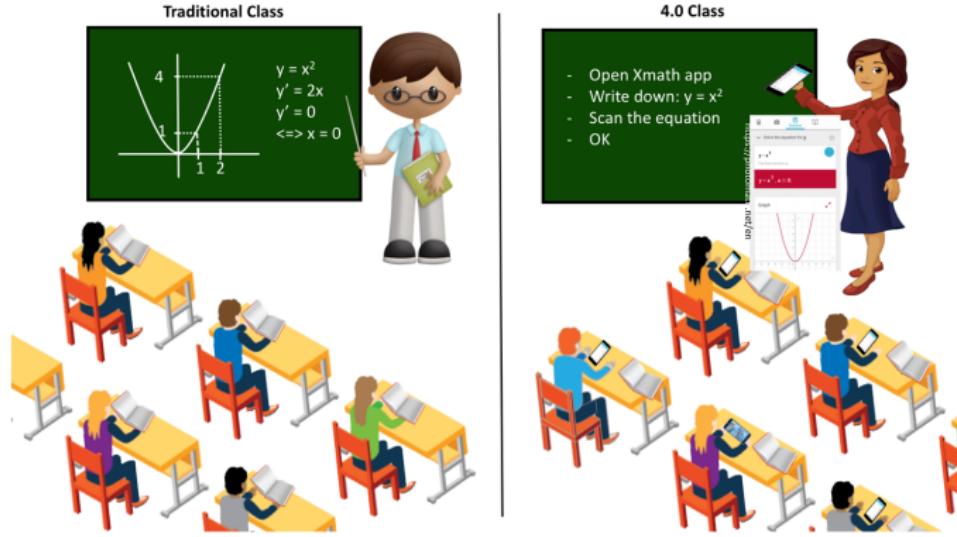
② Công trình liên quan

③ Mô hình đề xuất

④ Hiện thực và đánh giá

⑤ Tổng kết

Đặt vấn đề



Hình: Ứng dụng của công nghệ trong giáo dục.

Phạm vi đề tài

- Nhận dạng biểu thức toán học dạng offline¹.
- Mỗi ảnh dùng để nhận dạng chỉ chứa đúng 1 biểu thức và biểu thức này có đường cơ sở² không bị lệch khỏi đường nằm ngang của ảnh.
- Chuyển biểu thức từ dạng hình ảnh sang dạng Latex.

¹Nhận dạng từ hình ảnh.

²Thuật ngữ tiếng Anh: baseline

Phạm vi đề tài

- Nhận dạng biểu thức toán học dạng offline¹.
- Mỗi ảnh dùng để nhận dạng chỉ chứa đúng 1 biểu thức và biểu thức này có đường cơ sở² không bị lệch khỏi đường nằm ngang của ảnh.
- Chuyển biểu thức từ dạng hình ảnh sang dạng Latex.

¹Nhận dạng từ hình ảnh.

²Thuật ngữ tiếng Anh: baseline

Phạm vi đề tài

- Nhận dạng biểu thức toán học dạng offline¹.
- Mỗi ảnh dùng để nhận dạng chỉ chứa đúng 1 biểu thức và biểu thức này có đường cơ sở² không bị lệch khỏi đường nằm ngang của ảnh.
- Chuyển biểu thức từ dạng hình ảnh sang dạng Latex.

¹Nhận dạng từ hình ảnh.

²Thuật ngữ tiếng Anh: baseline

Lý do chọn đề tài

- Thách thức khi thực hiện đề tài:
 - Làm sao để nhận diện được từng ký hiệu?
 - Làm cách nào nhận dạng cả một biểu thức?
 - Có thể khắc phục được những khuyết điểm trong công trình nhận dạng biểu thức toán học của nhóm sinh viên đi trước.
 - Có thể tạo ra được sản phẩm hoàn thiện như PhotoMath³
 - ...
- Động lực tiến hành khi áp dụng kiến thức đã học để tạo ra một sản phẩm hữu ích cho xã hội và tự mình có thể sử dụng.

³Một ứng dụng nhận dạng biểu thức toán học

Mục lục

① Giới thiệu

② Công trình liên quan

- Watch, Attend and Parse
- Context-aware Recognition
- QAK

③ Mô hình đề xuất

④ Hiện thực và đánh giá

⑤ Tổng kết

Watch, Attend and Parser⁴

Phương pháp

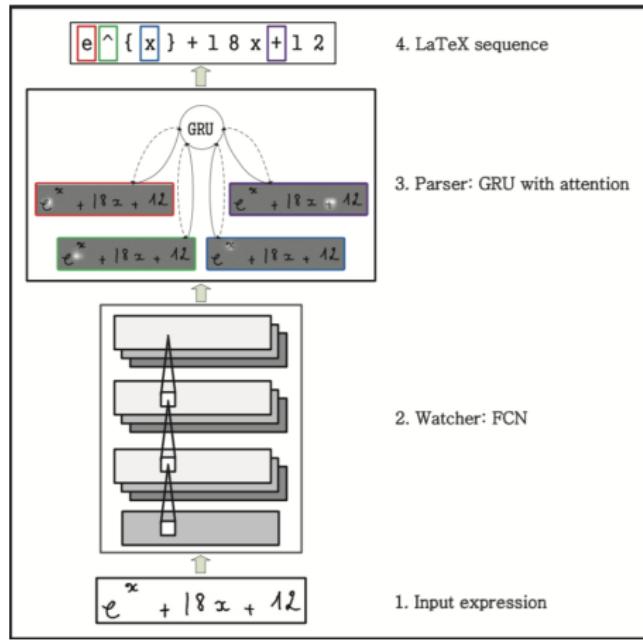
- **Watcher** là mạng nơron tích chập đầy đủ- FCN^a mã hoá ảnh đầu vào (bộ 9 ảnh bao gồm 1 ảnh gốc và ảnh 8 hướng) tạo ra đặc trưng ứng với từng pixel của ảnh gốc.
- **Parser** là kiến trúc mạng GRU^b nhận các vectơ đặc trưng được sinh ra từ **Watcher**, kết hợp cơ chế **Attention** do nhóm tác giả đề xuất để sinh ra chuỗi Latex.

^aFully convolutional network

^bKyunghyun Cho et al. "On the properties of neural machine translation: Encoder-decoder approaches". In: *arXiv preprint arXiv:1409.1259* (2014).

⁴Jianshu Zhang et al. "Watch, Attend and Parse: An End-to-end Neural Network Based Approach to Handwritten Mathematical Expression Recognition". In: *Pattern Recognition* (2017).

Watch, Attend and Parse



Hình: Hình minh họa các bước thực hiện của phương pháp Watch, Attend and Parser.



Context-aware Recognition⁵

Phương pháp

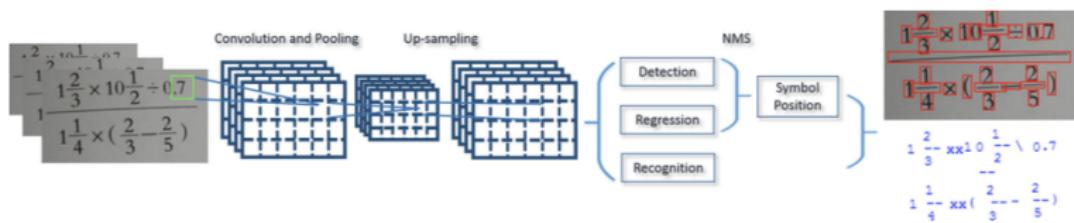
Ảnh đầu vào sẽ qua một số lớp tích chập và pooling để tạo ra một feature map. Feature map này là input cho 3 nhiệm vụ bên dưới. Cụ thể, giả sử một điểm i được cho đặt tại toạ độ (w_i, h_i) của feature map đầu vào.

- **Nhiệm vụ phát hiện** (Detection task) sẽ cho ra một con số s thể hiện độ tin cậy một ký hiệu được đặt tại i .
- **Nhiệm vụ hồi quy** (Regression task) cho ra một vec-tor 4 chiều x_1, y_1, x_2, y_2 thể hiện thông tin về bounding box của ký hiệu được đặt tại i .
- **Nhiệm vụ nhận dạng** (Recognition task) gán nhãn cho ký hiệu đặt tại i cùng với xác suất của nhãn đó.

⁵Wenhao He et al. "Context-aware mathematical expression recognition: An end-to-end framework and a benchmark". In: *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE. 2016, pp. 3246–3251.



Context-aware Recognition



Hình: Mô hình học được đề xuất⁶.

⁶Wenhai He et al. "Context-aware mathematical expression recognition: An end-to-end framework and a benchmark". In: *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE. 2016, pp. 3246–3251.

Watch, Attend and Parser và Context-aware Recognition

Đánh giá

Cả 2 phương pháp đều tận dụng được thông tin cấu trúc 2 chiều của biểu thức toán học qua quá trình nhận dạng bằng cách sử dụng một lớp mạng CNN để tạo ra đặc trưng từ ảnh biểu thức đầu vào, thay vì sử dụng cách phương pháp phổ biến trong phân tách ký tự như phân tích hình chiểu^a, phân tích thành phần liên thông^b. Do đó, hạn chế lỗi do quá trình nhận dạng ký tự gây ra ảnh hưởng đến kết quả sau cùng của bài toán.

^aThuật ngữ tiếng Anh: projection cutting

^bThuật ngữ tiếng Anh: connected component analysis

QAK⁷

Phương pháp

- **Tiền xử lý:** chuyển ảnh đầu vào về ảnh xám, khử nhiễu bằng bộ lọc Guass, thực hiện chuyển về ảnh nhị phân và phân tích thành phần liên thông để tăng cường chất lượng , hỗ trợ bước phân đoạn.
- **Phân đoạn ảnh:** dùng kỹ thuật phân tích hình chiếu và phân tích thành phần liên thông để tách ảnh đầu vào thành những mảnh ảnh chỉ chứa một ký tự.

⁷Anh Nguyen Quoc and Khoa Nguyen Anh. Nhận dạng biểu thức toán học. 2016. ↩ ↪ ↩

QAK

Phương pháp

- **Nhận dạng:** sử dụng kiến trúc mạng được chỉnh sửa thì Lenet- 5^a để học nhận dạng ký tự.
- **Phân tích cú pháp:** tạo ra các dãy ký hiệu có khả năng là kết quả của ảnh đầu vào, áp dụng tập luật văn phạm phi ngữ cảnh do các tác giả đề xuất để chọn ra kết quả có độ tin tưởng cao nhất.

^aYann LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE 86.11* (1998), pp. 2278–2324.

Dánh giá

Hệ thống của nhóm tác giả vẫn còn một số vấn đề sau:

- Phương pháp phân tích hình chiêu gấp vẫn đề với những ký tự dính nhau, overlap nhau.
- Hạn chế trong việc giải quyết những biểu thức mà ký tự có chỉ số trên hoặc chỉ số dưới.

Mục lục

① Giới thiệu

② Công trình liên quan

③ Mô hình đề xuất

- Kiến trúc mạng SSD
- Xây dựng các phiên bản
- Phân tích cấu trúc

④ Hiện thực và đánh giá

⑤ Tổng kết

Mô hình đề xuất

Mô hình đề xuất

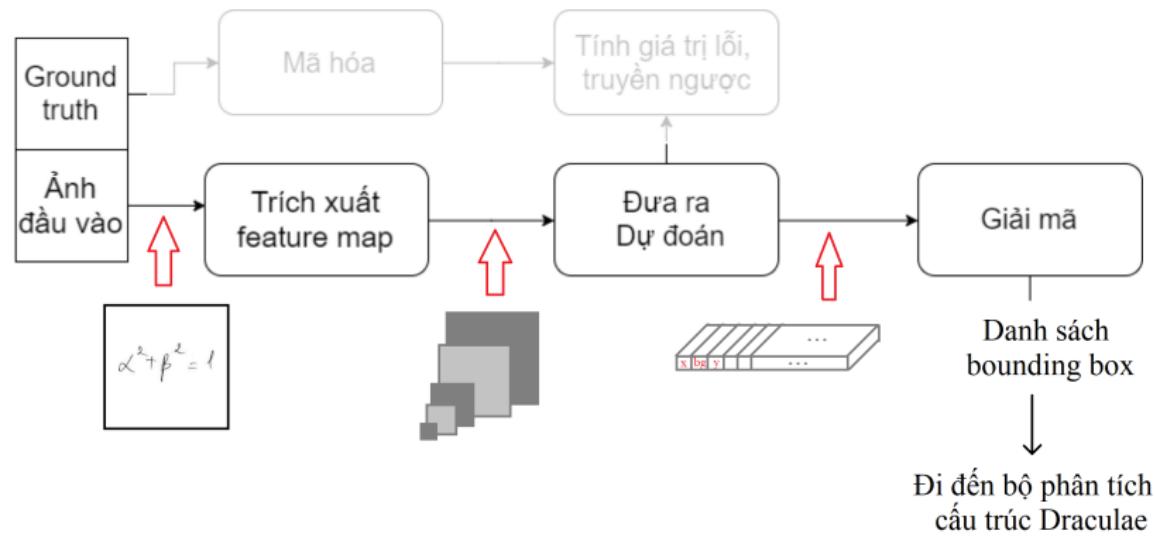
- Mạng SSD⁸ - Single Shot Multibox Detector
- DRACULAE⁹ - Diagram Recognition Application for Computer Understanding of Large Algebraic Expressions

⁸Wei Liu et al. "SSD: Single shot multibox detector". In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

⁹Richard Zanibbi, Dorothea Blostein, and James Cordy. "Recognizing mathematical expressions using tree transformation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.11 (Nov. 2002), pp. 1455 –1467. DOI: [10.1109/TPAMI.2002.1046157](https://doi.org/10.1109/TPAMI.2002.1046157).



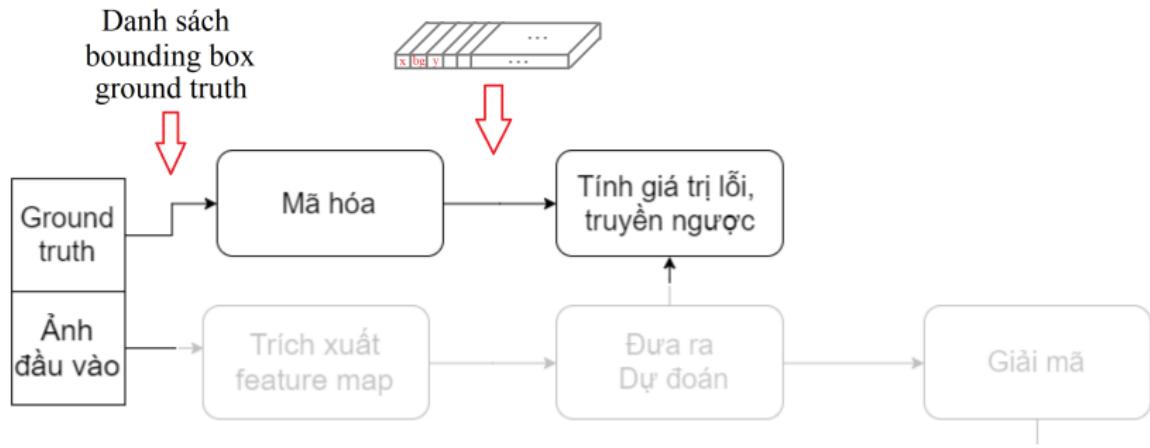
Mô hình đề xuất



Hình: Mô hình mạng SSD nhóm đề xuất

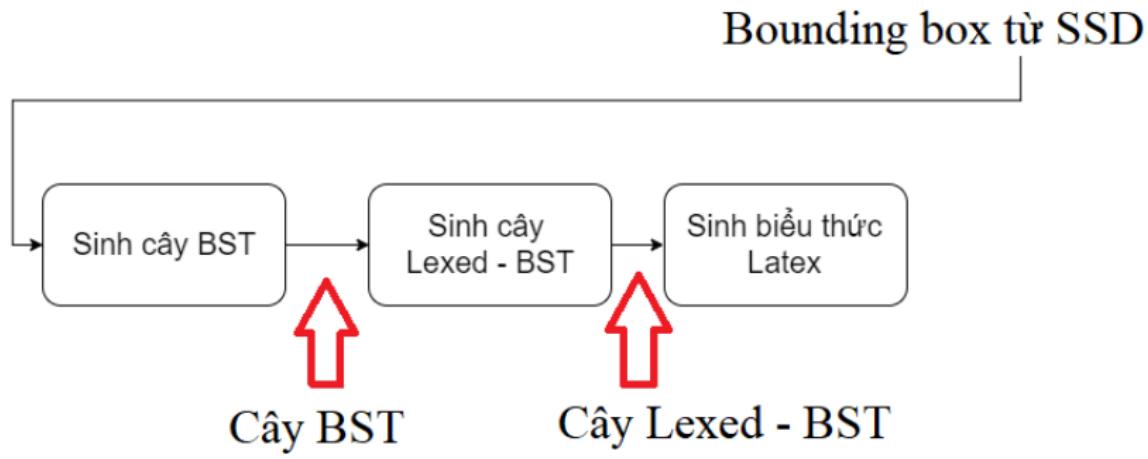


Mô hình đề xuất



Hình: Mô hình mạng SSD nhóm đề xuất

Mô hình đề xuất

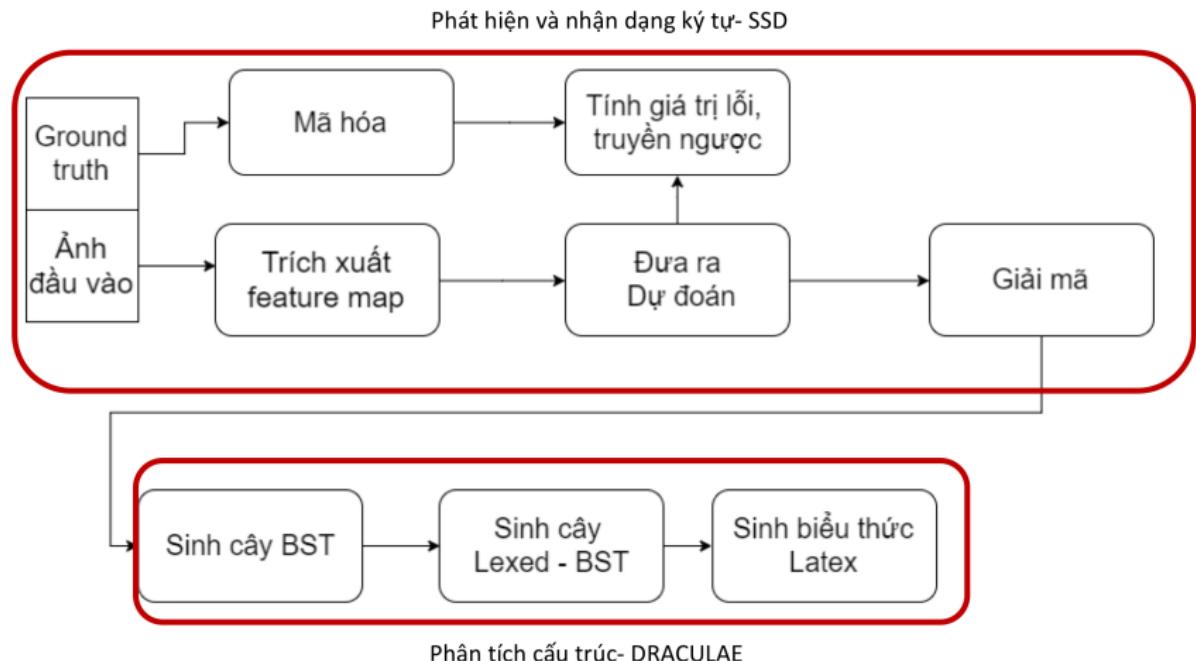


BST: Baseline Structure Tree

Hình: Mô hình bộ phân tích cú pháp Draculae nhóm đề xuất



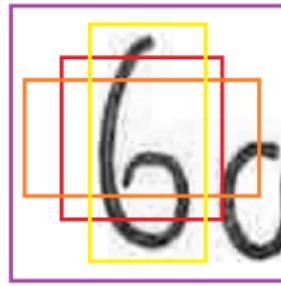
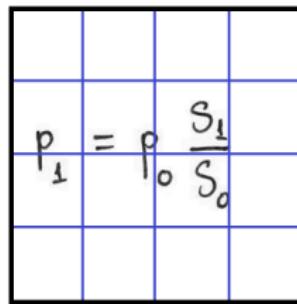
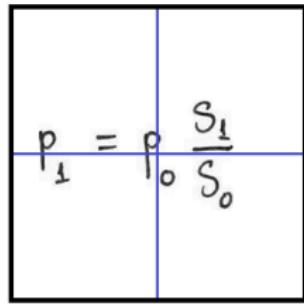
Mô hình đề xuất



Hình: Mô hình nhóm đề xuất



Bộ Mã Hóa (Encoder) - Sinh Default Box

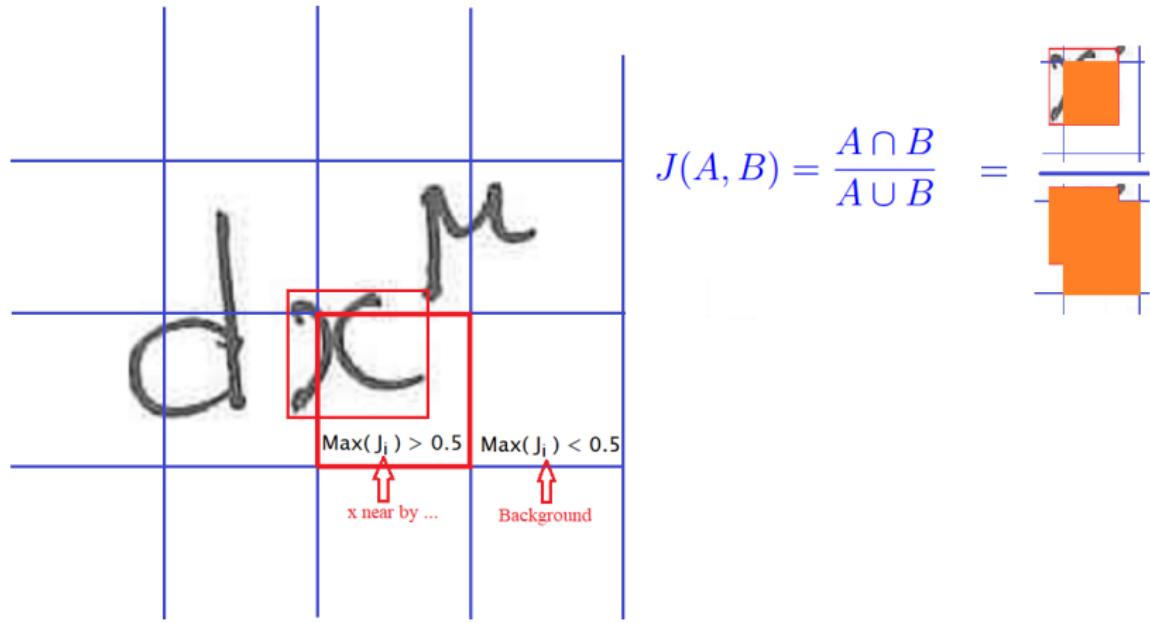


Hình: Các default box với mức kích thước khác nhau

Sinh Default Box

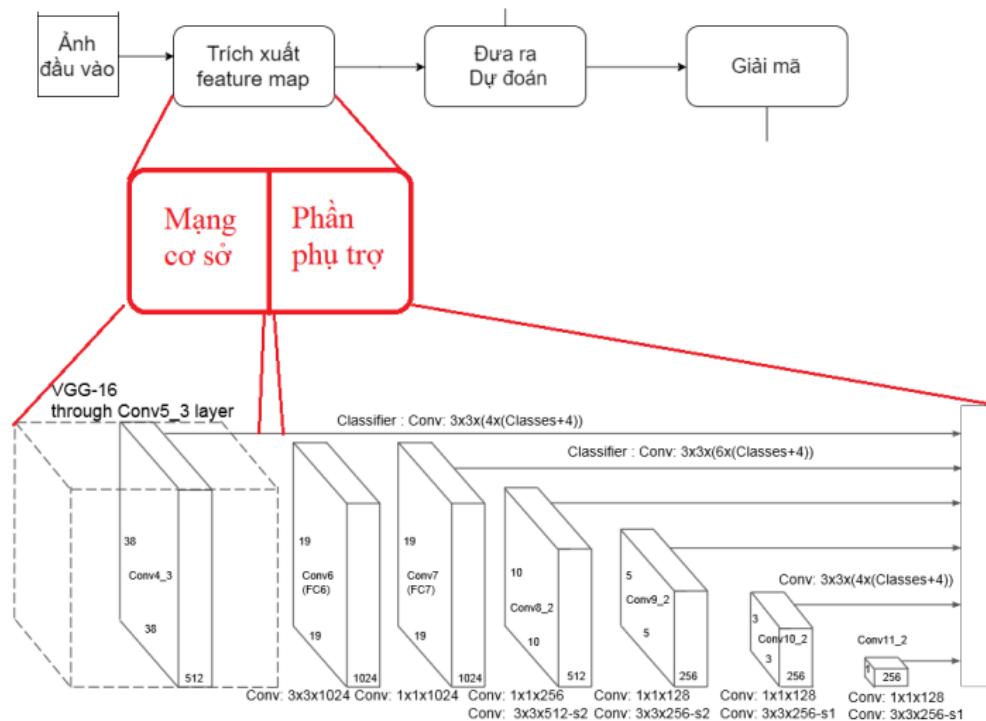
- Các default box được sinh với nhiều mức kích thước khác nhau dựa trên công thức $s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1} (k - 1)$.
- Quá trình sinh default box phụ thuộc vào một số cấu hình: Số mức scale, kích thước của default box trong mỗi mức scale, số lượng default box trong mỗi mức kích thước và số lượng aspect ratio.

Bộ Mã Hóa (Encoder) - Matching



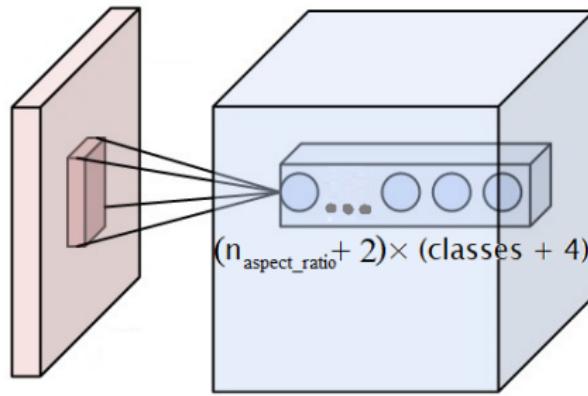
Hình: Quá trình matching

Bộ phận trích xuất các feature map



Hình: Sơ đồ bộ phận trích xuất feature map

Bộ phận đưa ra dự đoán



Hình: Phép tích chập để sinh bounding box

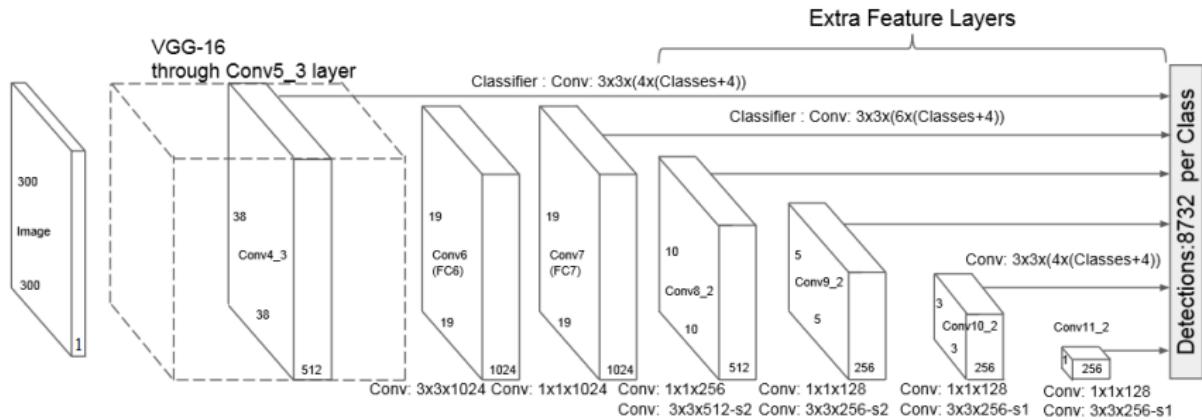
Đưa ra dự đoán

Với mỗi feature map trích ra được, hệ thống sẽ có một lớp tích chập tương ứng để đưa ra dự đoán. Mỗi vùng ảnh được kernel trượt qua sẽ tương ứng với một số dự đoán (ứng với các tỉ lệ bounding box khác nhau).

Xây dựng các phiên bản

Xây dựng các phiên bản

Phiên bản I - Phiên bản nguyên thủy

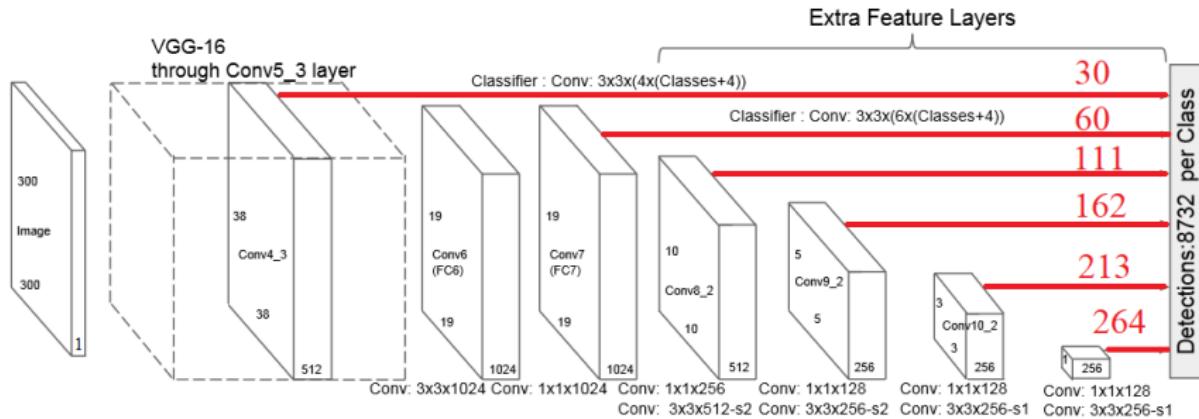


Hình: Mô hình mạng SSD nguyên thủy

Mô hình SSD300

- Mạng cơ sở: VGG
- Số lớp: 106
- Kích thước ảnh đầu vào: 300×300

Phiên bản I - Phiên bản nguyên thủy

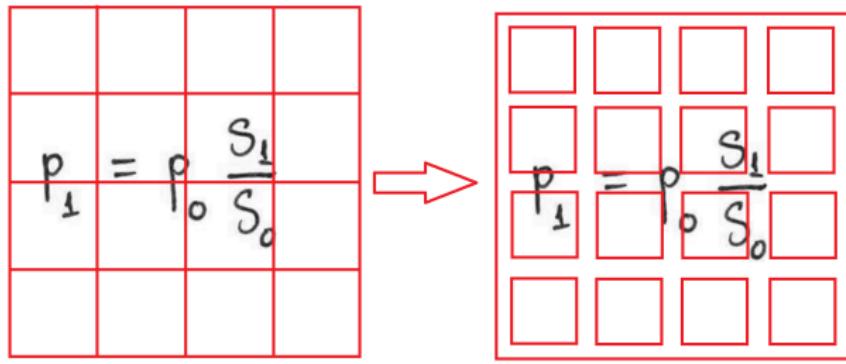


Hình: Mô hình mạng SSD nguyên thủy

Mô hình SSD300

- Do kích thước default box không đủ nhỏ nên hệ thống gấp khó khăn trong việc nhận diện ký tự nhỏ.

Phiên bản II

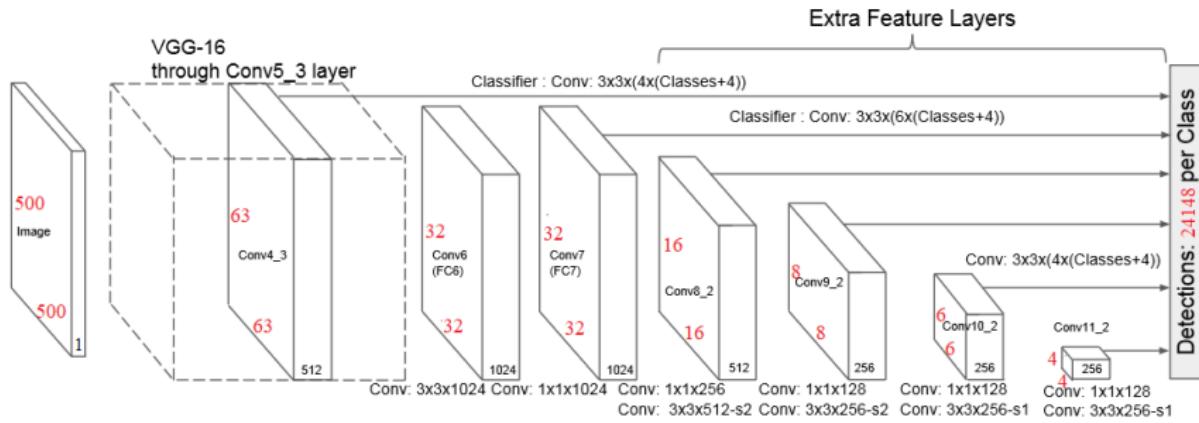


Hình: Thu nhỏ kích thước các default box

Các thay đổi

Giảm kích thước các default box trong mỗi mức kích thước:
 $(30, 60, 111, 162, 213, 264, 315) \rightarrow (9, 24, 54, 84, 114, 144, 174)$.
Điều này khiến lưới default box bị phá hỏng.

Phiên bản III



Hình: Mô hình phiên bản III

Các thay đổi

Tăng kích thước ảnh đầu vào từ 300×300 lên 500×500 kéo theo kích thước các feature map cũng tăng lên.

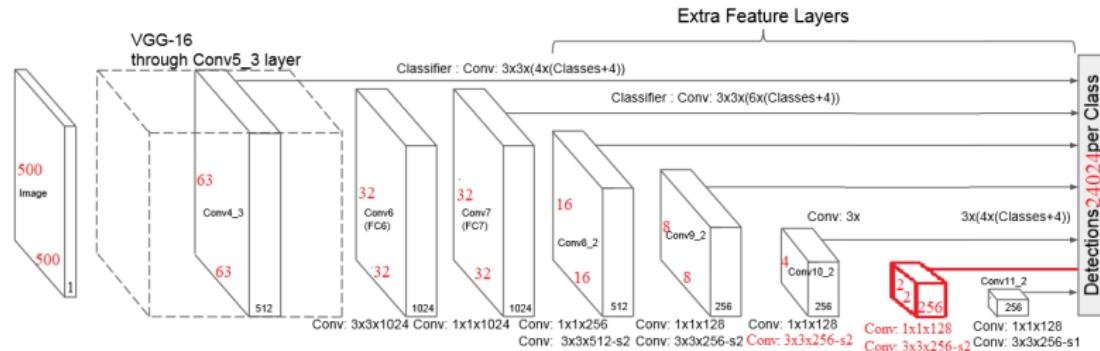
Phiên bản III



Hình: Mô hình phiên bản III

Khi kích thước feature map tăng lên, số lượng default box tương ứng với feature map đó cũng tăng lên góp phần lấp đầy phần lưới bị hỏng.

Phiên bản IV



Hình: Mô hình phiên bản IV

Các thay đổi

- Tăng thêm một khối gồm 02 lớp tích chập nhằm tăng số feature map trích xuất được.

Phân tích cấu trúc

Phân tích cấu trúc

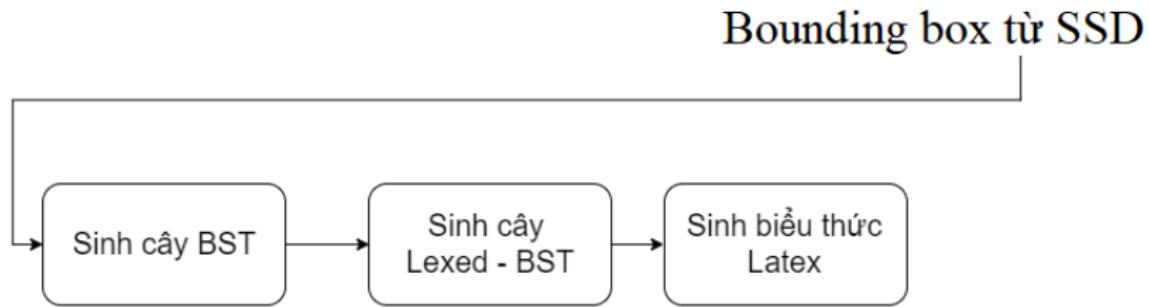
- DRACULAE¹⁰

¹⁰Richard Zanibbi, Dorothea Blostein, and James Cordy. "Recognizing mathematical expressions using tree transformation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.11 (Nov. 2002), pp. 1455 –1467. DOI: [10.1109/TPAMI.2002.1046157](https://doi.org/10.1109/TPAMI.2002.1046157).



Phân tích cấu trúc

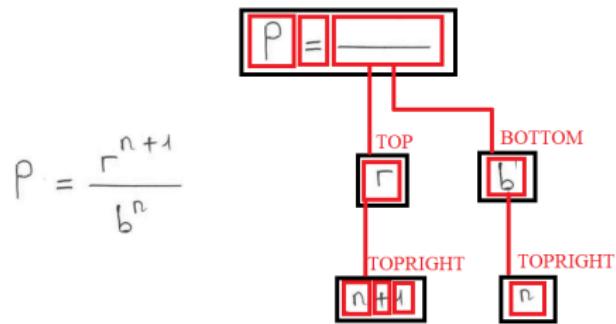
Bộ phận phân tích cấu trúc có nhiệm vụ nhận dữ liệu là các bounding box từ SSD¹¹ và xây dựng một cây Lexed - BST¹² tạo tiền đề để xây dựng biểu thức Latex.



¹¹Wei Liu et al. “SSD: Single shot multibox detector”. In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

¹²Richard Zanibbi, Dorothea Blostein, and James Cordy. “Recognizing mathematical expressions using tree transformation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.11 (Nov. 2002), pp. 1455 –1467. DOI:

Sinh cây BST - Baseline Structure Tree



Hình: Một cây BST

Cây BST

- Mỗi nút là một đường cơ sở.
- Các liên kết giữa các nút thể hiện vị trí tương đối giữa các đường cơ sở.

Phân tích cấu trúc - Tìm ký tự chủ đạo

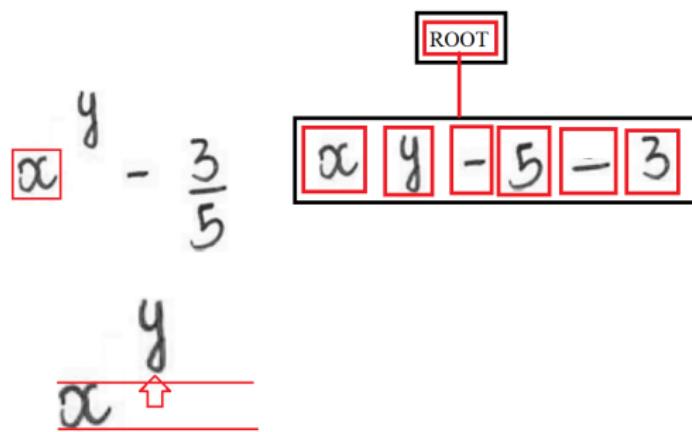
$$\boxed{x!^y} \times y!^x \quad \boxed{\frac{6\sqrt{3}}{4}} \quad \boxed{\sum_{k=P_0}^P} a_k x^k$$

Hình: Một số ký tự chủ đạo

Ký tự chủ đạo

Ký tự chủ đạo là ký tự bắt đầu của một đường cơ sở, là ký tự nằm bên trái nhất mà không bị thống trị.

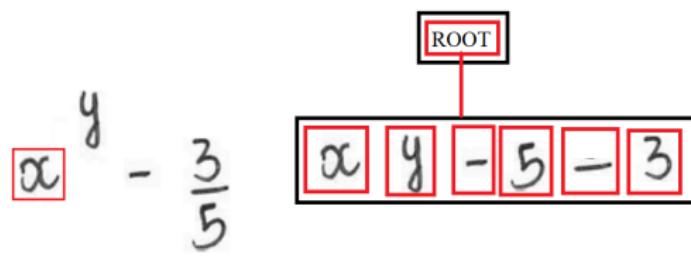
Xác định ký tự trên đường cơ sở



Hình: Xác định ký tự trên đường cơ sở



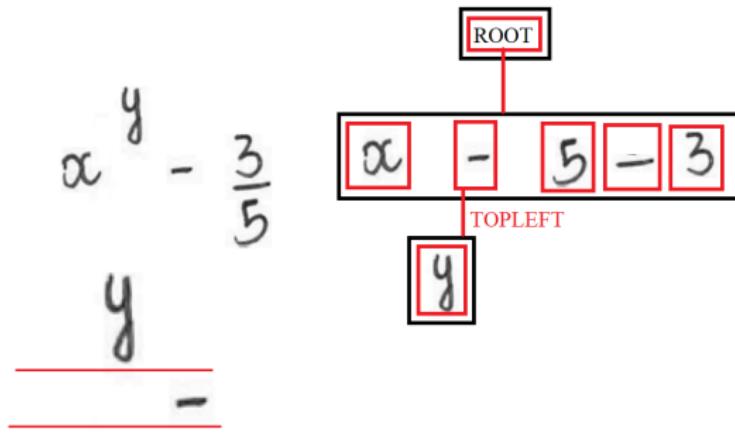
Xác định ký tự trên đường cơ sở



x -

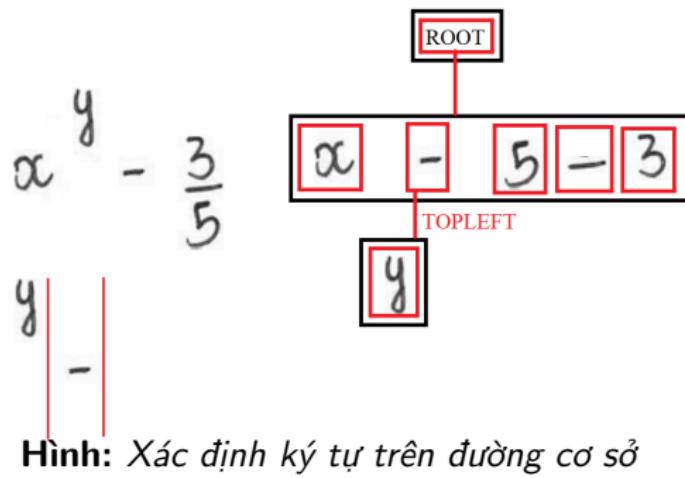
Hình: Xác định ký tự trên đường cơ sở

Xác định ký tự trên đường cơ sở

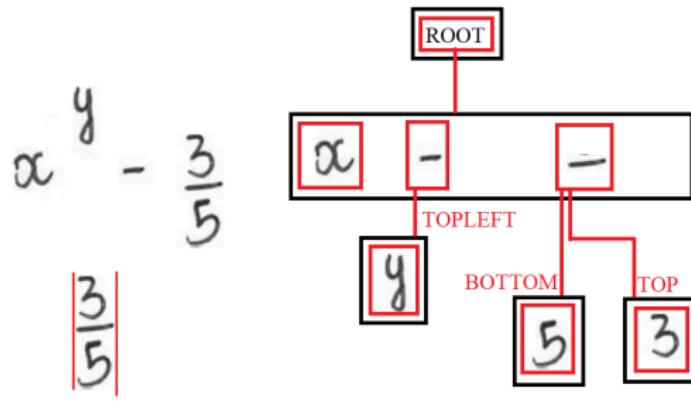


Hình: Xác định ký tự trên đường cơ sở

Xác định ký tự trên đường cơ sở



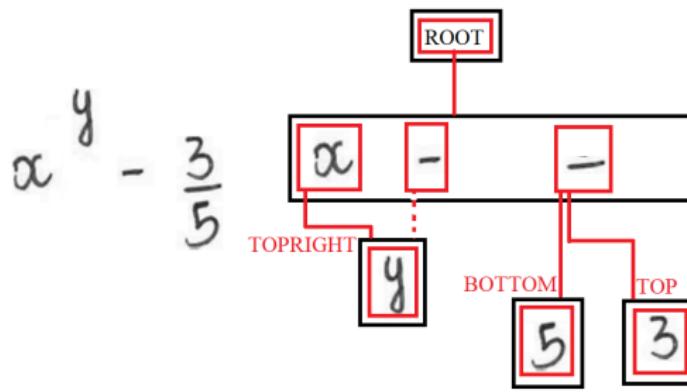
Xác định ký tự trên đường cơ sở



Hình: Xác định ký tự trên đường cơ sở



Tái phân vùng

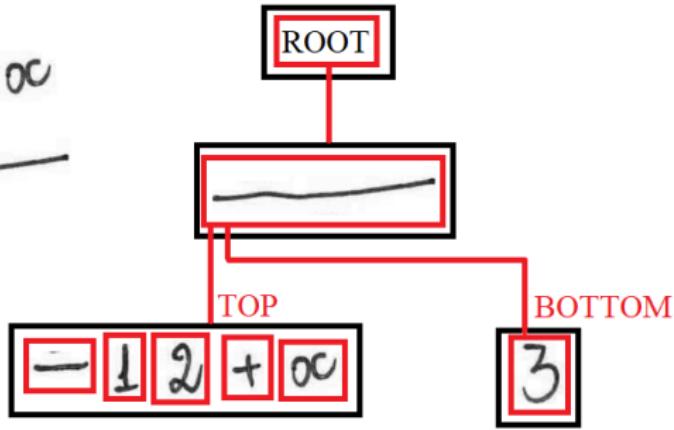


Hình: Tái phân vùng

Xử lý nút con

$$\frac{1}{2} + 0c$$

3

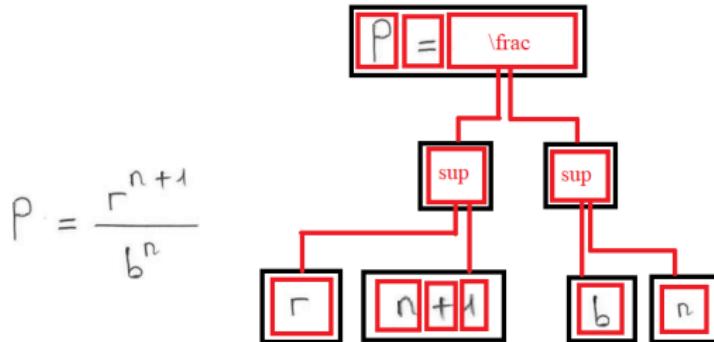


Hình: Xử lý nút con

Xử lý nút con

Sau lần chạy đầu tiên thì các nút con mới chỉ là danh sách các ký tự, vì vậy cần xử lý từng nút con để thu được cây BST hoàn chỉnh.

Sinh cây Lexed BST

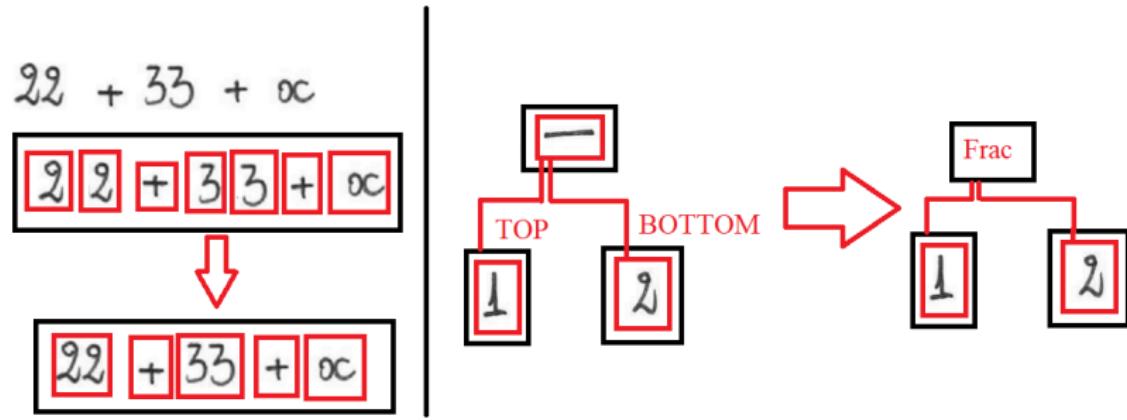


Hình: Một cây Lexed BST

Cây Lexed - BST

- Mỗi nút là một đường cơ sở
- Liên kết giữa các nút biểu thị quan hệ về ngữ nghĩa toán học giữa các đường cơ sở.

Sinh cây Lexed BST



Hình: Sinh cây Lexed BST

Sinh cây Lexed BST

- Các ký tự sẽ được nhóm lại phù hợp
- Sử dụng tập luật để xây dựng ngữ nghĩa.

Mục lục

- ① Giới thiệu
- ② Công trình liên quan
- ③ Mô hình đề xuất
- ④ Hiện thực và đánh giá
 - Chuẩn bị dữ liệu
 - Đánh giá kết quả
- ⑤ Tổng kết

Xây dựng tập ký tự

Phát triển từ tập ký tự đã có từ đề tài QAK¹³.

| Thông tin mô tả | QAK | Hiện tại |
|-------------------|-------|----------|
| Số người tham gia | 20 | 36 |
| Số lượng nhãn | 88 | 106 |
| Số lượng ký tự | 46197 | 52353 |

Bảng: Thông tin mô tả tập dữ liệu ký tự.

Các nhãn ký hiệu trong tập dữ liệu hiện tại:

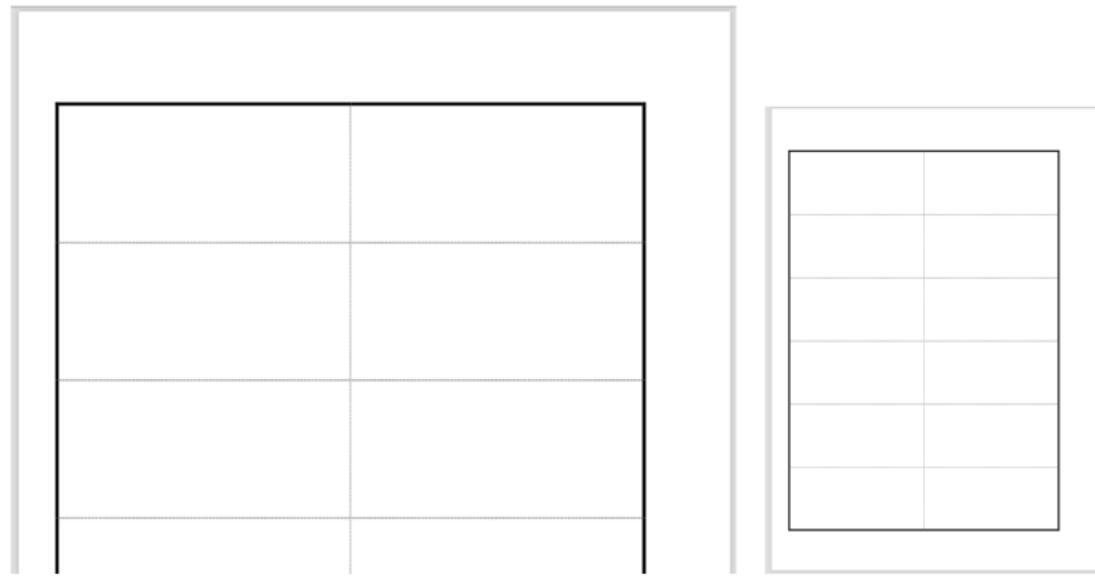
0-9, a-z, A-Z, (,), +, -, *, /, =, \int , α , β , γ , δ , ϵ , θ , λ , μ , π , ρ , σ , ϕ , ω , Δ , Π , Σ , Φ , Ω , sin, cos, tan, log, lim, $\sqrt{ }$, \rightarrow , \geq , \leq , \forall , \exists , \in , !, ., ..., \div , \neq , ∞ .

¹³ Anh Nguyen Quoc and Khoa Nguyen Anh. Nhận dạng biểu thức toán học. 2016.



Xây dựng tập ảnh biểu thức

- ① Chuẩn bị mẫu thu dữ liệu.



Hình: Mẫu thu dữ liệu biểu thứ.



Xây dựng tập ảnh biểu thức

- ② Chuẩn bị biểu thức mẫu làm gợi ý cho người tham gia viết¹⁴.

| | |
|--|--|
| 1. $x = r \cos \theta$ | 1. $1 + 2 + \dots + n = \frac{n(n+1)}{2}$ |
| 2. $\cos \theta = \frac{x}{\sqrt{x^2 + y^2}}$ | 2. $\frac{1}{4} + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^3 + \dots = \frac{1}{3}$ |
| 3. $c^2 = a^2 + b^2 - 2ab \cos C$ | 3. $H \in P$ |
| 4. $a = b \cos C + c \cos B$ | 4. $P^M P^N$ |
| 5. $P = \sqrt{a^2 + b^2 - 2ab \cos A}$ | 5. $G \in X$ |
| 6. $\frac{1}{r^2} = \frac{1}{(R-m)^2} + \frac{1}{(R+m)^2}$ | 6. $M < N$ |
| 7. $\mathbf{x} = x \cos \theta + y \sin \theta$ | 7. $\forall x \in E \exists y \in E \text{ s.t. } xy$ |
| 8. $b = \cos B = \cos A$ | 8. $\exists M, R > 0$ |
| 9. $S = \left(\sum_{i=1}^n \theta_i - (n-2)\pi \right) r^2$ | 9. $\wp(x)$ |
| 10. $y^q + y + 1 = 0$ | 10. $x = \log_a b$ |
| 11. $a^x q^y = a^{x+y}$ | 11. $w_n = aq^{n-n_0}$ |
| 12. $151 + 143 : 97$ | 12. $151 + 143 : 97$ |

Hình: Ví dụ một biểu thức mẫu.

¹⁴ CROHME: Competition on Recognition of Online Handwritten Mathematical Expressions. http://www.isical.ac.in/~scrohme/CROHME_data.html



Xây dựng tập ảnh biểu thức

- ③ Tập hợp biểu mẫu đã cho viết, mang đi scan.

| | |
|--|---|
| Đoàn Thị Ngọc Nhung | |
| $1+2+\dots+n = \frac{n(n+1)}{2}$ | $\forall x \in E \exists y \in E x R y$ |
| $\frac{1}{4} + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^3 + \dots = \frac{1}{3}$ | $R > 0$ |
| $H \in P$ | $y(x)$ |
| $P^\mu P_\mu$ | $x = \log_a b$ |
| Đoàn Thị Ngọc Nhung $\forall x \in E \exists y \in E x R y$ $\frac{1}{4} + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^3 + \dots = \frac{1}{3}$ $R > 0$ $y(x)$ $x = \log_a b$ | |

Hình: Mẫu đã được điền và scan.

Xây dựng tập ảnh biểu thức

- ❸ Sử dụng đoạn mã Matlab¹⁵ để tạo ảnh đầu vào vào quá trình huấn luyện.

$$\frac{1}{4} + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^3 + \dots = \frac{1}{3}$$

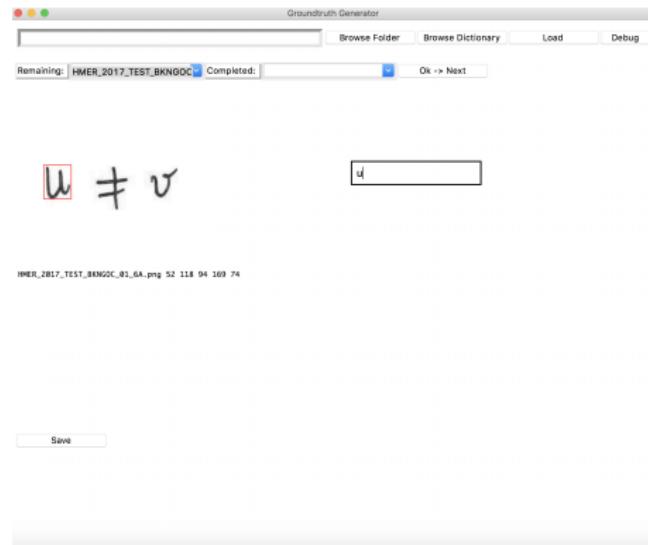
Hình: Ví dụ một ảnh là input cho quá trình huấn luyện.

¹⁵ Anh Nguyen Quoc and Khoa Nguyen Anh. Nhận dạng biểu thức toán học. 2016.



Xây dựng tập ảnh biểu thức

- ⑤ Xây dựng bộ nhãn và vị trí bounding box của từng ký tự trong các ảnh đầu vào.



Hình: Giao diện công cụ hỗ trợ quá trình gán nhãn và xác định bounding box.



Thông tin mô tả

- Số người tham gia: khoảng 88
- Số lượng ảnh: 2256. Trong đó có 1746 ảnh dùng cho training, 138 ảnh dùng cho quá trình validation và 372 ảnh để test.
- Số loại biểu thức: khoảng 552
- Số người gán nhãn: 3

| Thông tin thêm | Tập huấn luyện | Tập kiểm tra |
|---------------------------------------|----------------|----------------|
| Tổng số ký tự | 13697 | 2237 |
| Chiều dài trung bình của biểu thức | ≈ 8 | ≈ 6 |
| Kích thước trung bình của ký tự | 24×29 | 31×41 |

Bảng: Thông tin mô tả hai tập dữ liệu dùng cho huấn luyện và kiểm tra.



Nhắc lại 4 phiên bản thử nghiệm

| Phiên bản | I | II | III | IV |
|--|---------|---------|---------|---------|
| Kích thước ảnh đầu vào | 300x300 | 300x300 | 500x500 | 500x500 |
| Kích thước default boxes ¹⁶ | (20,90) | (8,50) | (8,50) | (8,50) |
| Số lượng default boxes | 8732 | 8732 | 24148 | 24024 |

Bảng: Sự khác nhau giữa 4 phiên bản thử nghiệm.

¹⁶Kích thước default box được quy định bởi bộ 2 thông số: (s_{min} , s_{max})



Đánh giá định tính

$$1 + \boxed{2} + \boxed{3} + 4$$

$$1 \boxed{\text{add}} \quad \boxed{2} + 3 \boxed{\text{add}} \quad \boxed{4}$$

Phiên bản I

Phiên bản II

Phiên bản III

Phiên bản IV

$$\boxed{1} \quad \boxed{\text{add}} \quad \boxed{2} \quad \boxed{\text{add}} \quad \boxed{3} \quad \boxed{\text{add}} \quad \boxed{4}$$

$$1 \quad \boxed{\text{add}} \quad \boxed{2} \quad \boxed{\text{add}} \quad \boxed{3} \quad \boxed{\text{add}} \quad \boxed{4}$$

Hình: Kết quả nhận diện của 4 phiên bản thử nghiệm.



Đánh giá định tính

$$E = \frac{1}{2} k \underline{\underline{oc}}^2 - F_{oc}$$

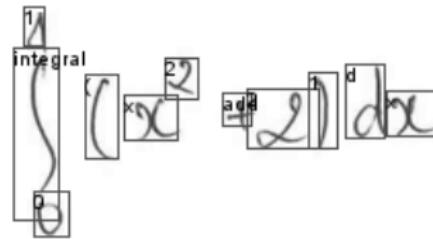
$$E = \frac{1}{2} k \underline{\underline{oc}}^2 - F_{oc}$$

| | Phiên bản I | Phiên bản II |
|---|---|--------------|
| Phiên bản III | | |
| Phiên bản IV | | |
| E = $\frac{1}{2} k \underline{\underline{oc}}^2 - F_{oc}$ | E = $\frac{1}{2} k \underline{\underline{oc}}^2 - F_{oc}$ | |

Hình: Kết quả nhận diện của 4 phiên bản thử nghiệm.

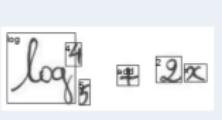
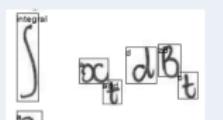
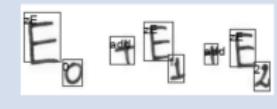
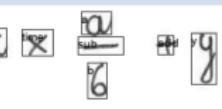
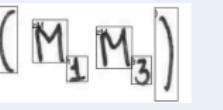
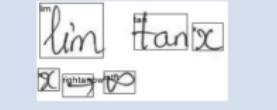
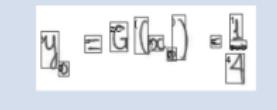
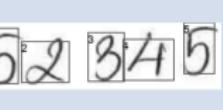


Đánh giá định tính



Hình: Kết quả nhận diện phiên bản IV trên ký tự nhỏ.

Đánh giá định tính

| | | | |
|---|---|---|---|
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Hình: Một số ảnh kết quả sau quá trình nhận dạng của phiên bản IV.



Đánh giá định lượng

- Điều kiện đánh giá

- Cùng tập ảnh test gồm 372 ảnh.
- Chỉ số đánh giá: *mAP*.
- Máy chạy đánh giá: Ubuntu 14.04 LTS, Intel Core i5- 2500M 3.30GHz, 8GB RAM.

- Kết quả

| phiên bản | I | II | III | IV |
|-------------------------------|-------|-------|-------|-------|
| <i>mAP</i> | ≈0.58 | ≈0.51 | ≈0.64 | ≈0.65 |
| Số lượng ký tự nhận dạng được | 1581 | 1054 | 1837 | 1920 |
| Số file ảnh không xử lý được | 4/372 | 8/372 | 0/372 | 0/372 |

Bảng: Kết quả *mAP* của 4 phiên bản thử nghiệm.



Công thức tính chỉ số mAP

Tính precision cho từng nhãn theo công thức¹⁷:

$$\text{precision} = \frac{TP}{TP + FP}, \quad (1)$$

với *TP* được viết tắt từ *TruePositive*¹⁸ và *FP* từ *FalsePositive*¹⁹.

Tính trọng số của từng nhãn trong tập dữ liệu kiểm tra theo công thức:

$$\text{class_weight}_i = \frac{N_i}{\sum_{u=0}^{106} N_u}, \quad (2)$$

với *N_i* là số lượng ký tự thuộc nhãn *i*.

mAP là trung bình có trọng số giữa *precision* và *class_weight*.

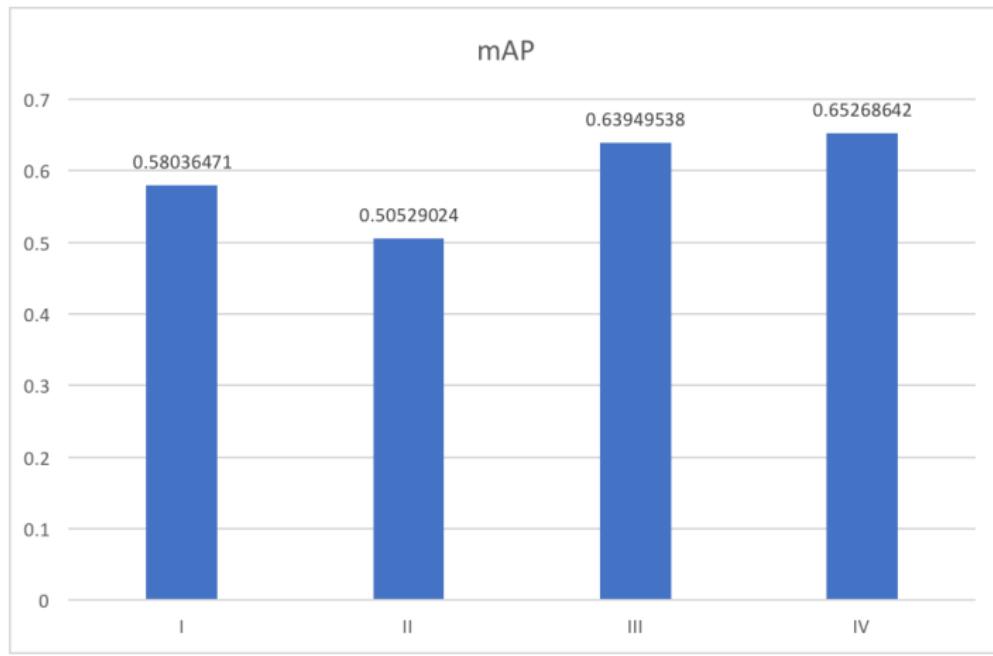
¹⁷WIKIPEDIA. https://en.wikipedia.org/wiki/Precision_and_recall.

¹⁸WIKIPEDIA. https://en.wikipedia.org/wiki/Precision_and_recall.

¹⁹WIKIPEDIA. https://en.wikipedia.org/wiki/Precision_and_recall.



Kết quả



Hình: Trực quan kết quả mAP cho 4 phiên bản thử nghiệm.



Mục lục

① Giới thiệu

② Công trình liên quan

③ Mô hình đề xuất

④ Hiện thực và đánh giá

⑤ Tổng kết

- Ưu điểm
- Nhược điểm
- Những điều đã làm được
- Hướng phát triển

Ưu điểm

- Hệ thống có thể nhận diện được ký tự ở nhiều kích thước khác nhau.
- Nhận diện được nhiều biểu thức mà nếu chỉ dùng kỹ thuật phân vùng thì khó có thể làm được (ví dụ như biểu thức có ký tự dính nhau).
- Nhận diện được nhiều loại ký tự.

Nhược điểm

- Vẫn còn sai sót trong việc gán nhãn ký tự.
- Tốc độ xử lý một ảnh còn chậm.
- Bộ phân tích cú pháp đơn giản.
- Các ký tự overlap nhau dễ bị bỏ sót (dấu căn).

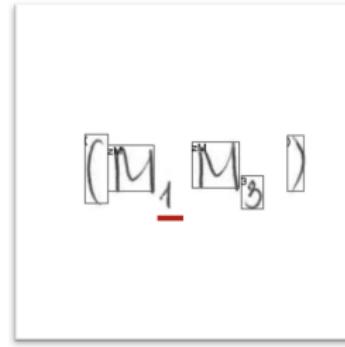
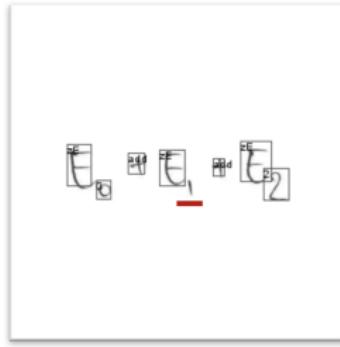
Những điều đã làm được

- Xây dựng tập dữ liệu ký tự và biểu thức.
- Cải tiến mạng SSD gốc để phát hiện và nhận dạng được những đối tượng có kích thước nhỏ, cụ thể là ký tự toán học viết tay.
- Hiện thực bộ parser dựa trên lý thuyết của Zanibbi²⁰.

²⁰Richard Zanibbi, Dorothea Blostein, and James Cordy. "Recognizing mathematical expressions using tree transformation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.11 (Nov. 2002), pp. 1455 –1467. DOI: [10.1109/TPAMI.2002.1046157](https://doi.org/10.1109/TPAMI.2002.1046157).

Phát triển tập dữ liệu biểu thức

- Sử dụng phiên bản IV để kiểm tra.
- Hai ảnh cụ thể trong 372 ảnh test không nhận được chỉ số dưới, cụ thể số 1.



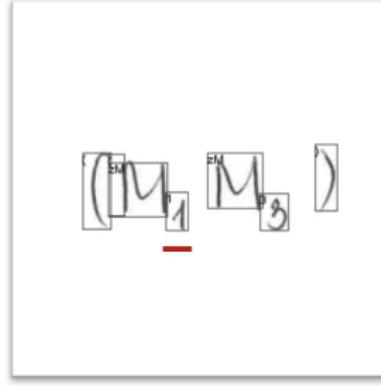
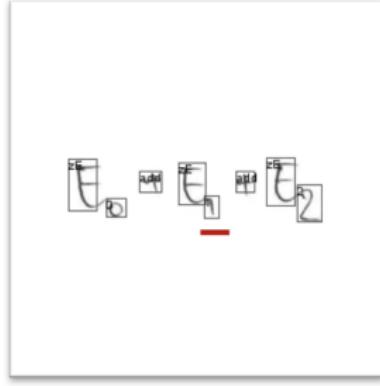
Hình: Trường hợp không nhận được chỉ số dưới.



Phát triển tập dữ liệu biểu thức

Giải pháp

Bổ sung mẫu biểu thức chứa cụ thể E_1 và M_1 vào tập huấn luyện. Cụ thể, tăng cường 96 ảnh vào tập huấn luyện và 24 ảnh để test.



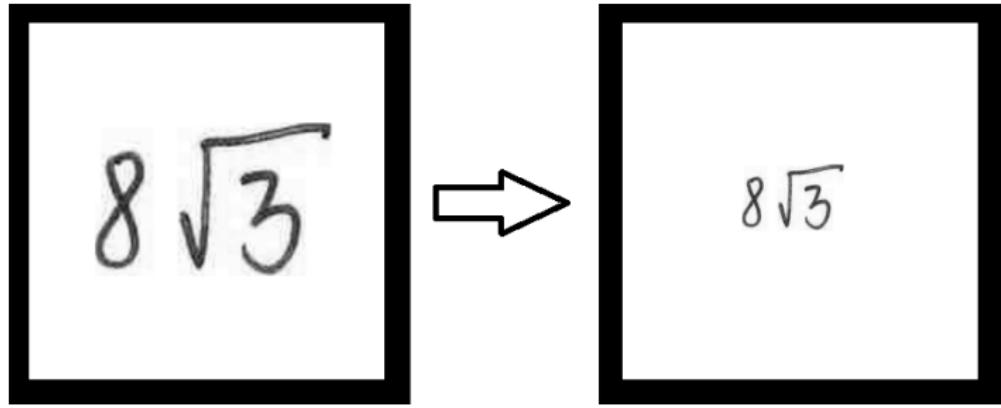
Hình: Kết quả nhận dạng sau khi được huấn luyện với dữ liệu bổ sung.



Khắc phục lỗi nhận dạng sai ở ký tự lớn

Giải pháp

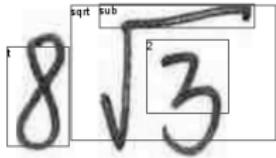
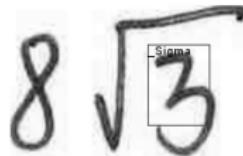
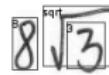
Thu nhỏ nội dung ảnh.



Hình: Ví dụ thu nhỏ ảnh.

Khắc phục lỗi nhận dạng sai ở ký tự lớn

Mô hình I

Mô hình IV
Dùng ảnh gốcMô hình IV
Dùng ảnh thu nhỏ

Hình: Kết quả thu được sau khi thu nhỏ nội dung ảnh.



Hướng phát triển khác

- Những điều cần kiểm tra:

- Ảnh chứa đựng các ký tự nghiêng.
- Ảnh có nhiễu.
- Ảnh với nền khác màu trắng.
- Ảnh chụp thay vì scan.

- Những điều cần hoàn thiện:

- Thêm phần hậu xử lý ở các giai đoạn nhận diện ký tự và sinh cây Lexed - BST.
- Cải thiện khả năng nhận diện các ký tự có aspect ratio đặc biệt.
- Tối ưu hoá bộ phân tích cú pháp.

Demo

Demo

Cảm ơn quý thầy cô và các bạn!

Tham khảo

- Cho, Kyunghyun et al. "On the properties of neural machine translation: Encoder-decoder approaches". In: *arXiv preprint arXiv:1409.1259* (2014).
- He, Wenhao et al. "Context-aware mathematical expression recognition: An end-to-end framework and a benchmark". In: *Pattern Recognition (ICPR), 2016 23rd International Conference on*. IEEE. 2016, pp. 3246–3251.
- Jaccard, Paul. "Distribution de la Flore Alpine dans le Bassin des Dranses et dans quelques régions voisines.". In: *37* (Jan. 1901), pp. 241–72.
- LeCun, Yann et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE 86.11* (1998), pp. 2278–2324.
- Liu, Wei et al. "SSD: Single shot multibox detector". In: *European conference on computer vision*. Springer. 2016, pp. 21–37.
- ## Online Handwritten Mathematical Expressions, CROHME: Competition on Recognition of.
- http://www.isical.ac.in/~scrohme/CROHME_data.html



Phụ lục

Phụ lục

Mẫu thu ký tự

| Name: Pečený říční Název Názvum | | | | | | | | | | | |
|---------------------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| sin | sín |
| cos | cos | cos | cos | cos | cos | cos | cos | cos | cos | cos | cos |
| tan | tan | tan | tan | tan | tan | tan | tan | tan | tan | tan | tan |
| log | log | log | log | log | log | log | log | log | log | log | log |
| lim | lim | lim | lim | lim | lim | lim | lim | lim | lim | lim | lim |
| $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ | $\sqrt{}$ |
| \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow | \rightarrow |
| \geq | \geq | \geq | \geq | \geq | \geq | \geq | \geq | \geq | \geq | \geq | \geq |
| \leq | \leq | \leq | \leq | \leq | \leq | \leq | \leq | \leq | \leq | \leq | \leq |
| \forall | \forall | \forall | \forall | \forall | \forall | \forall | \forall | \forall | \forall | \forall | \forall |
| \exists | \exists | \exists | \exists | \exists | \exists | \exists | \exists | \exists | \exists | \exists | \exists |
| \in | \in | \in | \in | \in | \in | \in | \in | \in | \in | \in | \in |
| $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ | $!$ |
| $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ | $,$ |
| $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ | $+$ |
| \neq | \neq | \neq | \neq | \neq | \neq | \neq | \neq | \neq | \neq | \neq | \neq |
| \dots | \dots | \dots | \dots | \dots | \dots | \dots | \dots | \dots | \dots | \dots | \dots |
| ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |

Hình: Ví dụ một mẫu thu ký tự đã được viết.



Cấu trúc mạng VGG16



Hình: Cấu trúc mạng VGG16

Những lỗi sai của hệ thống nhận dạng

Ảnh đầu vào

$$p = 2 \times l - q$$

$$p = 2 \times l - q$$

$$y = 2x$$

Ảnh sau bước nhận dạng

$$\boxed{p} = \boxed{2} \boxed{\times} \boxed{l} = \boxed{q}$$

$$\boxed{p} = \boxed{2} \boxed{\times} \boxed{l} = \boxed{q}$$

$$\boxed{y} = \boxed{2} \boxed{x}$$

Ảnh sau bước phân tích cấu trúc

$$p = 2 \times k - q$$

$$p = 2 \times 1 - q$$

$$y = 2x$$

Hình: Trường hợp sinh chuỗi Latex sai khi biểu thức chứa các ký tự có chân.



Những lỗi sai của hệ thống nhận dạng

| Ảnh đầu vào | Ảnh sau bước nhận dạng | Ảnh sau bước phân tích cấu trúc |
|------------------------|------------------------|---------------------------------|
| $4\pi^2 aR$ | $4\pi^2 aR$ | $4\pi^2 ak$ |
| S^{4m+1} | S^{4m+1} | 3^{4m+1} |
| $h \rightarrow \infty$ | $h \Rightarrow \infty$ | $h - \infty$ |

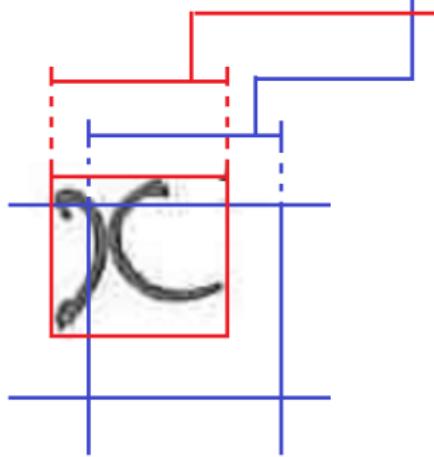
Hình: Trường hợp hệ thống gán nhãn cho những ký tự có nét tương tự nhau.



Mục tiêu huấn luyện - "Hệ quy chiếu" ?

Width:

$$\hat{g}^w = \log \left(\frac{g^w}{d^w} \right)$$

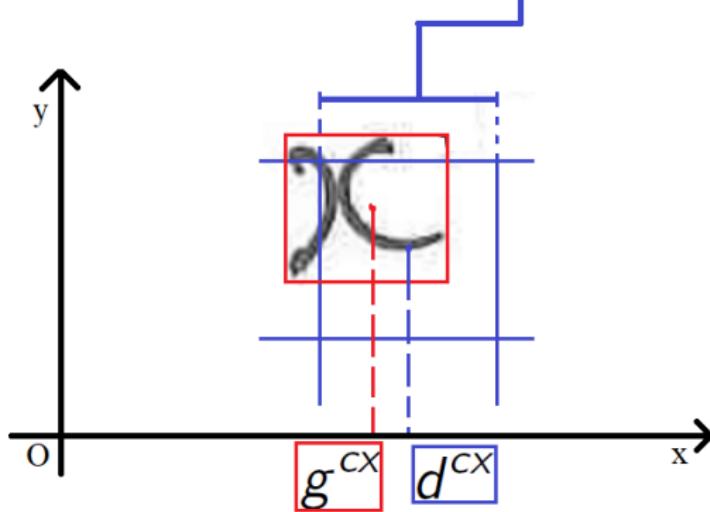


Hình: Độ lệch về kích thước



Mục tiêu huấn luyện - "Hệ quy chiếu" ?

$$\hat{g}^{cx} = (g^{cx} - d^{cx})/d^w$$



Hình: Độ lệch về vị trí trọng tâm



Mục tiêu huấn luyện

Mục tiêu huấn luyện

- Cực tiểu chênh lệch về vị trí.
- Cực đại độ tin cậy của nhãn.