



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



BÁO CÁO ĐỒ ÁN CHUYÊN NGÀNH

CẢI THIỆN BẢN ĐIÁ HÓA TRỰC QUAN  
BẰNG HƯỚNG TIẾP CẬN HỌC SÂU

HỘI ĐỒNG: Khoa học máy tính

GVHD: Nguyễn Đức Dũng

GVPB: GV phản biện

—o0o—

Sinh viên thực hiện:

Lê Minh Nghĩa MSSV: 2010445

Phạm Khai Anh Duy MSSV: 2011015

Nguyễn Trọng Nhân MSSV: 2011744

TP. HỒ CHÍ MINH, 12/2023

### **Lời cam đoan**

Chúng tôi xin cam đoan đây là công trình nghiên cứu của riêng chúng tôi dưới sự hướng dẫn của TS. Nguyễn Đức Dũng. Nội dung nghiên cứu và các kết quả đều là trung thực và chưa từng được công bố trước đây. Các số liệu được sử dụng cho quá trình phân tích, nhận xét được chính chúng tôi thu thập từ nhiều nguồn khác nhau và sẽ được ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, chúng tôi cũng có sử dụng một số nhận xét, đánh giá và số liệu của các tác giả khác, cơ quan tổ chức khác. Tất cả đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kì sự gian lận nào, chúng tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án tốt nghiệp của mình. Trường Đại học Bách Khoa Thành phố Hồ Chí Minh không liên quan đến những vi phạm tác quyền, bản quyền do chúng tôi gây ra trong quá trình thực hiện.

### **Lời ngỏ**

Đồ án chuyên ngành được hoàn thành dưới sự hướng dẫn khoa học của TS. Nguyễn Đức Dũng, Khoa Khoa học và Kỹ Thuật Máy Tính, Trường Đại học Bách Khoa - Đại học Quốc gia Thành phố Hồ Chí Minh. Nhóm thực hiện xin chân thành cảm ơn TS. Nguyễn Đức Dũng, đã giúp đỡ chúng em kiến thức chuyên môn, thảo luận, đưa ra gợi ý và tạo điều kiện thuận lợi cho chúng em hoàn thành đồ án.

Chúng em cũng xin gửi lời cảm ơn đặc biệt đến quý thầy cô phản biện, những người đã đọc và đóng góp ý kiến để chúng em hoàn thiện đồ án chuyên ngành của mình. Nhóm thực hiện cũng xin bày tỏ lòng biết ơn sâu sắc đến quý thầy cô Khoa Khoa học và Kỹ Thuật Máy tính, Trường Đại học Bách Khoa - Đại học Quốc gia Thành phố Hồ Chí Minh, là những người đã truyền thụ kiến thức chuyên môn, đã tạo điều kiện cho chúng em học tập và phát triển trong suốt quãng thời gian vừa qua.

Kính chúc quý thầy cô sức khoẻ, thành công và tiếp tục đào tạo những thế hệ sinh viên mới trong tương lai.

Chúng em xin chân thành cảm ơn.

**Nhóm thực hiện**

Lê Minh Nghĩa  
Phạm Khai Anh Duy  
Nguyễn Trọng Nhân

## Tóm tắt nội dung

Khả năng định vị toàn cầu có vai trò cốt lõi trong những lĩnh vực phụ thuộc nhiều vào việc nhận biết và tương tác phù hợp với môi trường xung quanh như xe tự hành, robot và công nghệ tương tác thực tế ảo AR. Trước đây, đa số những công nghệ này đều sẽ lấy thông tin về môi trường xung quanh nhờ vào những hệ thống định vị toàn cầu như GPS. Tuy nhiên, những hệ thống này cũng sẽ có những giới hạn nhất định về độ chính xác, độ tin cậy tùy thuộc vào khu vực được sử dụng. Vì vậy nên, bài toán bản địa hóa trực quan đã được đưa ra nhằm cung cấp thông tin đáng tin cậy hơn cho nhu cầu ngày càng tăng của quá trình phát triển của công nghệ.

Bản địa hóa trực quan - Visual Localization - là một bài toán hướng đến việc xây dựng một cách biểu diễn phù hợp và hiệu quả cho môi trường đang xét, từ đó có thể định vị một cách chính xác hơn trong môi trường đó. Bài toán sẽ chỉ quan tâm đến việc xử lý hình ảnh, video hoặc kết quả của những bộ cảm biến về môi trường xung quanh.

Hai hướng tiếp cận đã có những cải tiến liên tục trong những năm gần đây chính là hướng nhận dạng địa điểm trực quan - Visual Place Recognition - và hướng ước tính vị trí của máy ảnh - Pose Estimation. Với hướng tiếp cận nhận dạng địa điểm trực quan, mô hình sẽ nhận vào một ảnh truy vấn và chọn ra một hay nhiều ảnh từ tập ảnh được cung cấp sẵn mà được cho là thể hiện chung một cảnh với ảnh đầu vào. Với hướng tiếp cận ước tính vị trí của máy ảnh, từ cặp ảnh truy vấn và ảnh tham khảo đầu vào, mô hình sẽ tính toán vị trí và hướng quay của máy ảnh trong không gian.

Nhận thấy rằng hai hướng tiếp cận của bài toán khi đứng riêng đều đã có thể cung cấp những kết quả khả quan trong phạm vi của mình, và kết quả đầu ra cũng như dữ liệu đầu vào của hai hướng tiếp cận tương thích với nhau, nhóm quyết định sẽ kết hợp hai bài toán này thành một quy trình hoàn chỉnh, nhằm bổ trợ cho những khiếm khuyết của nhau. Cụ thể hơn:

- Với hướng tiếp cận nhận dạng địa điểm trực quan, nhóm sẽ sử dụng mô hình MixVPR, được xây dựng dựa trên mạng MLP-Mixer. Đây sẽ là nửa đầu của quy trình, cung cấp khả năng mở rộng lên những không gian rộng lớn như thành phố.
- Với hướng tiếp cận ước tính vị trí máy ảnh, nhóm sẽ sử dụng hướng tiếp cận 2D-2D được đề xuất trong Map-free Relocalization. Đây sẽ là nửa còn lại của quy trình, cung cấp khả năng đưa ra một dự đoán cụ thể cho quy trình.

# Mục lục

<b>1 GIỚI THIỆU</b>	<b>1</b>
1.1 Động cơ nghiên cứu . . . . .	1
1.2 Mục tiêu đề tài . . . . .	1
1.3 Phạm vi đề tài . . . . .	2
1.4 Cấu trúc đồ án chuyên ngành . . . . .	3
1.5 Kết chương . . . . .	3
<b>2 CÁC CÔNG TRÌNH LIÊN QUAN</b>	<b>5</b>
2.1 Những phương pháp đã được sử dụng . . . . .	5
2.2 Nhận dạng địa điểm trực quan - Visual Place Recognition . . . . .	5
2.2.1 Học biểu diễn - Representational learning . . . . .	5
2.2.2 Học biểu diễn NetVLAD . . . . .	5
2.2.3 Tối ưu hóa đặc trưng - MAC . . . . .	5
2.2.4 Trung bình Hölder - Trung bình GeM . . . . .	5
2.2.5 Truy xuất và tái xếp hạng . . . . .	5
2.2.6 Học biểu diễn bằng Vision Transformer . . . . .	5
2.2.7 Học biểu diễn bằng Feature Mixer . . . . .	5
2.3 Ước tính vị trí của máy ảnh - Pose Estimation . . . . .	5
2.3.1 Hồi quy vị trí tuyệt đối - Absolute Pose Regression . . . . .	6
2.3.2 Hồi quy vị trí tương đối - Relative Pose Regression . . . . .	16
2.3.3 Tái tạo kiến trúc từ chuyển động - Structure From Motion . . . . .	19
2.4 Phân tích và tổng hợp . . . . .	19
2.5 Một số tập dữ liệu phổ biến được sử dụng . . . . .	19
2.5.1 Tập dữ liệu trong không gian nhỏ . . . . .	19
2.5.2 Tập dữ liệu thành thị . . . . .	21
<b>3 PHƯƠNG PHÁP ĐỀ XUẤT</b>	<b>23</b>
3.1 Tổng quan về mô hình . . . . .	23
3.1.1 Cơ chế truy xuất ảnh . . . . .	23
3.1.2 Cơ chế hồi quy vị trí tương đối . . . . .	23
3.2 Tiêu chí đánh giá . . . . .	23
<b>4 ĐO ĐẠC VÀ ĐÁNH GIÁ</b>	<b>24</b>
4.1 Mô hình MixVPR . . . . .	24
4.2 Mô hình Map-free Relocalization . . . . .	24
4.3 Hướng phát triển . . . . .	24
4.4 Kết luận . . . . .	24

<b>5 Kế hoạch tương lai</b>	<b>25</b>
5.1 Thành quả đạt được . . . . .	25
5.2 Kế hoạch luận văn tốt nghiệp . . . . .	25
<b>A Kiến thức nền tảng</b>	<b>30</b>
A.1 Công nghệ đã được sử dụng . . . . .	30
A.2 Những mô hình học sâu nền tảng . . . . .	30
A.2.1 Mạng thần kinh nhân tạo - Artificial Neural Network . . . . .	30
A.2.2 Mạng Nơ-ron tích chập - Convolutional Neural Network . . . . .	32
A.3 Kiến thức về phương pháp nhận dạng địa điểm trực quan bằng MixVPR . . . . .	35
A.3.1 Trích xuất đặc trưng . . . . .	36
A.3.2 MLP-Mixer . . . . .	39
A.4 Kiến thức phương pháp ước tính vị trí của máy ảnh bằng ma trận thiết yếu . . . . .	41
A.4.1 Máy ảnh lỗ kim - Pinhole Camera . . . . .	41
A.4.2 Ma trận thiết yếu - Essential Matrix . . . . .	41
A.4.3 Tìm sự tương ứng giữa đặc trưng ảnh - Feature Matching . . . . .	41
A.4.4 Giải thuật 5 điểm ảnh - 5-Point Solver . . . . .	41
A.4.5 Thuật toán tính độ sâu ảnh qua một ảnh - Monocular Depth Estimation	41
A.5 Sự liên kết giữa mô hình truy xuất ảnh và mô hình hồi quy tương đối . . . . .	41
A.6 Tìm hiểu giới hạn của mô hình hồi quy vị trí tuyệt đối dựa trên mạng Nơ-ron tích chập . . . . .	42
A.7 Tìm hiểu giới hạn của một số tập dữ liệu . . . . .	42

# Danh sách hình vẽ

2.1	Kiến trúc mô hình hồi quy vị trí tuyệt đối đơn ảnh . . . . .	6
2.2	Minh họa mô hình CNN được áp dụng phân phối Bernoulli . . . . .	7
2.3	Minh họa kiến trúc mô hình LSTM PoseNet . . . . .	8
2.4	Minh họa kiến trúc mô hình Hourglass PoseNet . . . . .	8
2.5	Minh họa kiến trúc mô hình AtLoc . . . . .	9
2.6	Minh họa kiến trúc mô hình APANet . . . . .	9
2.7	Minh họa kiến trúc mô hình GPoseNet . . . . .	10
2.8	Minh họa kiến trúc mô hình Multi-Scene Transformer . . . . .	10
2.9	Minh họa kiến trúc mô hình MapNet . . . . .	11
2.10	Minh họa kiến trúc mô hình LSG . . . . .	12
2.11	Minh họa kiến trúc mô hình VlocNet . . . . .	12
2.12	Minh họa kiến trúc mô hình VlocNet++ . . . . .	13
2.13	Minh họa kiến trúc mô hình DGRNet . . . . .	13
2.14	Minh họa kiến trúc mô hình VidLoc . . . . .	15
2.15	Minh họa kiến trúc mô hình NNet . . . . .	16
2.16	Minh họa kiến trúc mô hình RelocNet . . . . .	17
2.17	Minh họa kiến trúc mô hình CamNet . . . . .	17
2.18	Minh họa kiến trúc mô hình EssNet . . . . .	18
2.19	Minh họa kiến trúc mô hình Relative Neural Network . . . . .	18
2.20	Minh họa kiến trúc mô hình AnchorNet . . . . .	19
2.21	Minh họa tập dữ liệu 7-Scenes . . . . .	19
2.22	Minh họa tập dữ liệu Cambridge Landmarks . . . . .	20
2.23	Minh họa tập dữ liệu Niantic Map-free Relocalization . . . . .	20
2.24	Minh họa tập dữ liệu Aachen Day-Night . . . . .	21
2.25	Minh họa tập dữ liệu GSV-Cities . . . . .	22
2.26	Minh họa tập dữ liệu San Francisco Extra Large . . . . .	22
A.1	Synap thần kinh là đơn vị cơ bản của hệ thống thần kinh và là nguồn cảm hứng để ứng dụng Perceptron vào lĩnh vực học sâu [3] . . . . .	31
A.2	Cấu tạo của một mạng nơ-ron nhân tạo cơ bản [26] . . . . .	31
A.3	Biểu đồ mô phỏng quá trình thực hiện cơ chế truyền ngược ở một mạng nơ-ron đơn giản [28] . . . . .	32
A.4	Một mạng nơ-ron tích chập cơ bản [35] . . . . .	33
A.5	Áp dụng một bộ lọc kích thước $3 \times 3$ lên một mảng của ảnh [35] . . . . .	34
A.6	Lớp tổng hợp sử dụng Average Pooling [1] . . . . .	35
A.7	Lớp tổng hợp sử dụng Max Pooling [1] . . . . .	35
A.8	Hình minh họa cho phương pháp nhận diện địa điểm trực quan thông qua những địa điểm đã qua [2] . . . . .	36

A.9	Hình minh họa cho phương pháp nhận diện địa điểm trực quan thông qua việc truy xuất ảnh từ hệ cơ sở dữ liệu [6] . . . . .	36
A.10	Quy trình của mô hình sử dụng SIFT để trích xuất đặc trưng cục bộ [62] . . . . .	37
A.11	Mô hình mã hóa ảnh bằng đặc trưng được truy xuất từ CNN và NetVLAD [6] . .	38
A.12	Cấu trúc của mô hình MLP-Mixer [50] . . . . .	40

**Bảng những từ ngữ chuyên ngành được sử dụng và phiên bản tiếng Anh**

<b>STT</b>	<b>Bản tiếng Việt</b>	<b>Bản tiếng Anh</b>
1	Bản địa hóa trực quan	Visual Localization
2	Ước tính vị trí máy ảnh	Camera Pose Estimation
3	Hồi quy vị trí tuyệt đối	Absolute Pose Regression
4	Hồi quy vị trí tương đối	Relative Pose Regression
5	Máy ảnh lỗ kim	Pinhole Camera
6	Ma trận thiết yếu	Essential Matrix
7	Tìm sự tương ứng giữa đặc trưng ảnh	Feature Matching
8	Giải thuật 5 điểm ảnh	5-Point Solver
9	Thuật toán tính độ sâu ảnh qua một ảnh	Monocular Depth Estimation
10	Truy xuất ảnh	Image Retrieval
11	Bản đồ đám mây điểm 3D	3D point cloud
12	Tái tạo kiến trúc từ chuyển động	Structure from Motion
13	Mạng Nơ-ron tích chập	Convulated Neural Network
14	Trích xuất đặc trưng ảnh	Feature Extraction
15	Đặc trưng ảnh cục bộ	Local Descriptor
16	Đặc trưng ảnh toàn cục	Global Descriptor
17	Bản đồ đặc trưng ảnh	Feature Map
18	Lớp pha trộn đặc trưng ảnh	Feature Mixer
19	Lớp kết nối đầy đủ	Fully Connected Layer
20	Bộ nhớ dài-ngắn hạn	Long Short Term Memory
21	Hồi quy quá trình Gaussian suy luận biến phân ngẫu nhiên	Stochastic Variational Inference Gaussian Process Regressions - SVI GPs
22	Cơ chế tự tập trung	Self-Attention
23	Hàm kích hoạt	Activation Function
24	Chuẩn hóa trên mỗi lớp	Layer Normalization
25	Học có hỗ trợ	Auxiliary Learning
26	Đo lường cảm biến trực quan	Visual Odometry
27	Cục bộ hỗ trợ toàn cục	Local Support Global
28	Tổng hợp trung bình	Average Pooling
29	Entropy chéo	Cross-entropy
30	Lớp kết hợp kết nối đầy đủ	Fully-connected Fusion layer
31	Đơn vị đo quán tính	Inertial Measurement Unit
32	Hệ thống vệ tinh định vị toàn cầu	Global Navigation Satellite System
33	Cải thiện đồ thị vị trí	Pose graph optimization
34	Đồng thuận lân cận	Neighborhood Consensus

**Bảng những từ viết tắt**

<b>STT</b>	<b>Bản viết tắt</b>	<b>Bản đầy đủ</b>
1	MLP	Multi-layer Perceptron
2	APR	Absolute Pose Regression
3	RPR	Relative Pose Regression
4	VPR	Visual Place Recognition
5	LSG	Local Support Global
6	FCFL	Fully-connected Fusion Layer
7	MSE	Mean square error
8	CTC	Cross transformation constraint
9	IMU	Inertial measurement unit
10	GNSS	Global Navigation Satellite System
11	VO	Visual Odometry
12	PGO	Pose graph optimization
13	NC	Neighborhood Consensus

# Chương 1

## GIỚI THIỆU

Giới thiệu đề tài: motivation, nội dung, mục tiêu, etc.

### 1.1 Động cơ nghiên cứu

#### Nhu cầu sử dụng trong thực tế

Đây là phần đặt vấn đề. Nêu ngắn gọn động cơ thực tế mà mình chọn đề tài (mục tiêu ứng dụng trong đời sống, ...).

#### Ứng dụng của bài toán bản địa hóa trực quan

Giả sử Visual localization là một giải pháp để giải quyết vấn đề trên, đây là phần giới thiệu sơ bộ mục tiêu và phương hướng đề ra của giải pháp. Cần thể hiện mục tiêu và phương hướng này giải quyết những mặt gì của vấn đề đặt ra.

Ở đoạn văn hai, liệt kê sơ bộ một số yếu tố đã tạo nên các vấn đề chính mà bài toán này đã và đang gặp phải.

Đưa ra định nghĩa sơ lược của bản địa hóa trực quan và ứng dụng đã có

### 1.2 Mục tiêu đề tài

#### Định nghĩa chính thức của bản địa hóa trực quan

#### Xác định những vấn đề hiện hữu trong bài toán bản địa hóa trực quan

- **Vấn đề lưu trữ**
  - 3D-point cloud tồn quá nhiều bộ nhớ để duy trì, đặc biệt là với những thành phố lớn
  - Có thể compress được nhưng mà sẽ đánh đổi bằng độ chính xác đặc trưng của phương pháp 2D-3D
- **Vấn đề tài nguyên**
  - Những mô hình học sâu thường có xu hướng nặng về tài nguyên tính toán: các mô hình VLAD (do nó cần nhiều descriptor 32768). Được nhắc đến trong Cos-Place

- Những mô hình SfM nặng về việc lưu trữ data + do cần phải tạo bản đồ 3D để có thể bắt đầu tính toán
- **Vấn đề về khả năng mở rộng của mô hình**
  - Mô hình deep learning chỉ chạy tốt trên những dataset nhỏ như 7Scenes và Cambridge Landmark, không phải những tập dữ liệu rộng và thưa thớt như Aachen và Mapillary Street-level Sequences
- **Vấn đề duy trì tập dữ liệu**
  - Ảnh vệ tinh do khó thu thập được nên đa số các ảnh vệ tinh sẽ bị lỗi thời không phản ánh đúng được tình hình hiện tại của môi trường. Ngoài ra, do ảnh vệ tinh chứa thông tin của một vùng rộng lớn nên khi có sự cập nhật của một khu vực nhỏ sẽ không được phản ánh hiệu quả trong ảnh vệ tinh.
  - Do những tập dữ liệu thành thị bao phủ một khu vực rộng lớn và dày đặc các địa danh(tòa nhà, công viên, trường học,...) nên đa số các tập dữ liệu sẽ không thể nào chứa hết được những góc nhìn có thể có. Dẫn đến việc ảnh lấy được từ dữ liệu sẽ không có nhiều điểm tương đồng với ảnh truy vấn.
- **Vấn đề về khả năng khai thác hóa**
  - Phần lớn các mô hình chỉ hoạt động trong một loại địa hình khu vực như một thành phố cố định (CosPlace, MixVPR,...). Trong khi AnyLoc tuyên bố có thể hoạt động trên đa địa hình.
  - Một số mô hình xây dựng cách biểu diễn của khu vực theo cách bao hàm bên trong mô hình học sâu (hồi quy vị trí tuyệt đối), hoặc trải qua quá trình tiền xử lý như xây dựng bản đồ đám mây điểm 3D cho nên khi cập nhật tập dữ liệu, sẽ cần phải xử lý lại. Vậy nên, sẽ cần trải qua quá trình huấn luyện hoặc xử lý lại khi cập nhật tập dữ liệu.
  - Một số mô hình học sâu cho ảnh như CNN không thể nắm bắt được những đặc trưng hình học khi được huấn luyện trên những hàm loss đơn giản(for more information, please refer to the paper to learn or not to learn, EssNet)

## **Đề xuất các mô hình học sâu phục vụ quá trình giải quyết bài toán**

Nhóm đề xuất một kiến trúc mô hình gồm 2 phần: truy xuất ảnh và hồi quy vị trí máy ảnh.

- Truy xuất ảnh
- Hồi quy vị trí tương đối

### **1.3 Phạm vi đề tài**

- **Mục tiêu chung**
  - Hướng đến việc thiết kế và áp dụng một kiến trúc mô hình học sâu để giải quyết bài toán bản địa hóa trực quan.
- **Kết quả mong đợi**
  - Đôi với truy xuất ảnh, nhóm mong có thể giúp cải thiện độ chính xác của những mô hình truy xuất ảnh đã được đề xuất trước đây.
  - Đôi với mô hình hồi quy vị trí máy ảnh, nhóm mong có thể giúp mô hình được sử dụng trên các tập dữ liệu lớn hơn.
- **Thời gian**

- Đồ án chuyên ngành kéo dài trong 15 tuần
- Đồ án tốt nghiệp kéo dài trong 15 tuần
- **Lĩnh vực hướng đến**
  - Đóng góp cho những nghiên cứu về bài toán bản địa hóa trực quan trong chuyên ngành thị giác máy tính.

## 1.4 Cấu trúc đồ án chuyên ngành

Đồ án chuyên ngành sẽ bao gồm năm (6) chương, bao gồm cả chương này. Nội dung ủa mỗi chương như sau:

- **Chương 1: Giới thiệu**  
Trình bày sơ lược về động cơ nghiên cứu, mục tiêu và phạm vi đề tài giải quyết.
- **Chương 2: Công trình liên quan**  
Chương này đề cập tới các kiến thức lý thuyết của các công trình nghiên cứu có liên quan.
- **Chương 3: Kiến thức nền tảng**  
Chương này đề cập tới các kiến thức lý thuyết nền tảng, các nguyên tắc cần được đảm bảo khi hiện thực và tiến hành so sánh, lựa chọn công nghệ sử dụng.
- **Chương 4: Phương pháp đề xuất**  
Chương này đề cập đến phương pháp giải quyết bài toán mà nhóm đề xuất bao gồm tổng quan về cơ chế cũng như lý thuyết cách hoạt động.
- **Chương 5: Kết quả khảo sát**  
Chương này đề cập kết quả khảo sát của nhóm bao gồm kết quả đánh giá hiệu quả của các phương pháp truy xuất ảnh và hồi quy vị trí máy ảnh.
- **Chương 6: Thảo luận**  
Chương này đề cập đến các câu hỏi cũng như các vấn đề nảy sinh trong quá trình nghiên cứu, thu thập kiến thức và hiện thực mô hình.

Cấu trúc của toàn đề tài sẽ được trình bày trong giai đoạn Đồ án tốt nghiệp.

## 1.5 Kết chương

Việc xác định chính xác vị trí có một phạm vi ứng dụng rộng rãi, là công nghệ chủ chốt trong rất nhiều lĩnh vực khác nhau. Tuy nhiên, việc chỉ dựa vào hệ thống không đảm bảo được sự ổn định như GPS sẽ hạn chế khả năng phát triển trong tương lai của những lĩnh vực ấy. Vậy nên bài toán định vị hóa trực quan đã được đưa ra để có một hệ thống đưa ra định vị chính xác và ổn định, dựa vào thông tin hình ảnh môi trường xung quanh.

Trong những phương pháp được đưa ra, nhóm tập trung vào việc kết hợp 2 hướng xử lý là truy xuất ảnh và hồi quy tương đối vị trí. Mỗi phương án sẽ có một vấn đề riêng. Đối với việc truy xuất ảnh, kết quả cho ra được sẽ không có độ chính xác cao, do vị trí của ảnh truy xuất được sẽ được lấy làm kết quả. Đối với hướng hồi quy tương đối vị trí, đa số những phương pháp trước đây đều tập trung vào việc hồi quy trong những không gian nhỏ, có lượng dữ liệu dày đặc, không phù hợp với tập dữ liệu đô thị, mục tiêu của bài nghiên cứu của nhóm. Qua việc ứng dụng cả 2 cách giải quyết trong một mô hình, nhóm hy vọng 2 mô hình có thể bổ trợ, giải quyết điểm yếu của nhau.

Để có thể lựa chọn được những giải pháp phù hợp, nhóm đã tiến hành khảo sát những kiến thức nền tảng và những công trình nghiên cứu đã được xuất bản trước đây. Những nội dung này sẽ được thể hiện trong **Chương 2: Các công trình liên quan**.

## Chương 2

# CÁC CÔNG TRÌNH LIÊN QUAN

### 2.1 Những phương pháp đã được sử dụng

### 2.2 Nhận dạng địa điểm trực quan - Visual Place Recognition

Trước đây, việc định vị trực quan trên quy mô lớn (large-scale visual localization) được coi là một vấn đề truy xuất hình ảnh[58]. Vị trí cho hình ảnh truy vấn được xác định bởi hình ảnh tương tự nhất được lấy từ cơ sở dữ liệu. Tuy nhiên, để đáp ứng nhu cầu xác định vị trí của ảnh với độ chính xác trên 6 bậc tự do (6 Degrees of Freedom), việc sử dụng mô hình 3D để ước tính tư thế của máy ảnh ngày càng được các nhà nghiên cứu đề xuất và sử dụng. Sử dụng phương pháp này, bài toán định vị trực quan có thể được chia thành hai bước: truy xuất hình ảnh và định vị tư thế máy ảnh từ hình ảnh truy xuất được.

#### 2.2.1 Học biểu diễn - Representational learning

#### 2.2.2 Học biểu diễn NetVLAD

#### 2.2.3 Tối ưu hóa đặc trưng - MAC

#### 2.2.4 Trung bình Hölder - Trung bình GeM

#### 2.2.5 Truy xuất và tái xếp hạng

#### 2.2.6 Học biểu diễn bằng Vision Transformer

#### 2.2.7 Học biểu diễn bằng Feature Mixer

### 2.3 Ước tính vị trí của máy ảnh - Pose Estimation

Ước tính vị trí máy ảnh (Camera Pose Estimation) là một bài toán thuộc chuyên ngành thị giác máy tính nhằm xác định vị trí (position) và góc quay (orientation) chính xác nhất có thể của máy ảnh thông qua dữ liệu hình ảnh được chụp từ chính máy ảnh. Đây là một bước cực kỳ quan trọng trong việc giải quyết bài toán bản địa hóa trực quan, thường được áp dụng sau khi bước nhận dạng địa điểm trực quan đã trích xuất

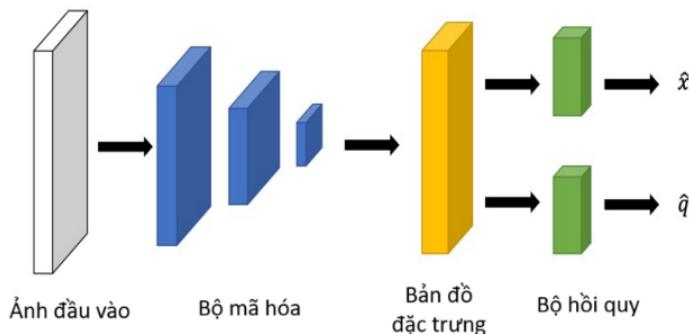
được ảnh từ kho dữ liệu. Hiện nay, có tương đối nhiều hướng tiếp cận đối với bài toán này. Một trong những phương pháp phổ biến nhất là huấn luyện một mô hình học sâu để xác định 6 chiều tự do (Degree of Freedom) từ số ít ảnh (Absolute Pose Regression và Relative Pose Regression) hoặc xây dựng một mô hình 3D từ tập dữ liệu có sẵn rồi tiến hành chuyển ảnh đầu vào sang các điểm 3D để dễ dàng so sánh và xác định vị trí (Structure From Motion).

### 2.3.1 Hồi quy vị trí tuyệt đối - Absolute Pose Regression

Hồi quy vị trí tuyệt đối (Absolute Pose Regression) hướng đến việc dự đoán vị trí và góc quay chính xác nhất của ảnh bằng một mô hình mạng nơ-ron tích chập thông qua việc cải thiện trọng số của mô hình. Tùy thuộc vào đầu vào của mô hình mà hồi quy vị trí tuyệt đối được chia thành ba hướng chính: hồi quy vị trí tuyệt đối với một ảnh, chuỗi ảnh hoặc đoạn phim.

#### **Hồi quy vị trí tuyệt đối đơn ảnh - Absolute pose regression through single monocular image**

Với phương pháp hồi quy vị trí tuyệt đối thông qua một ảnh (Absolute pose regression through single monocular image), quy trình chung thường bao gồm: đầu vào - mạng - đầu ra. Đầu vào sẽ là một ảnh RGB với đầu ra là vị trí 6 độ tự do của máy ảnh. Thông thường, kiến trúc của mạng lưới tính toán sẽ bao gồm các thành phần như sau: bộ mã hóa, bộ định vị, bộ hồi quy.



Hình 2.1: Kiến trúc mô hình hồi quy vị trí tuyệt đối đơn ảnh

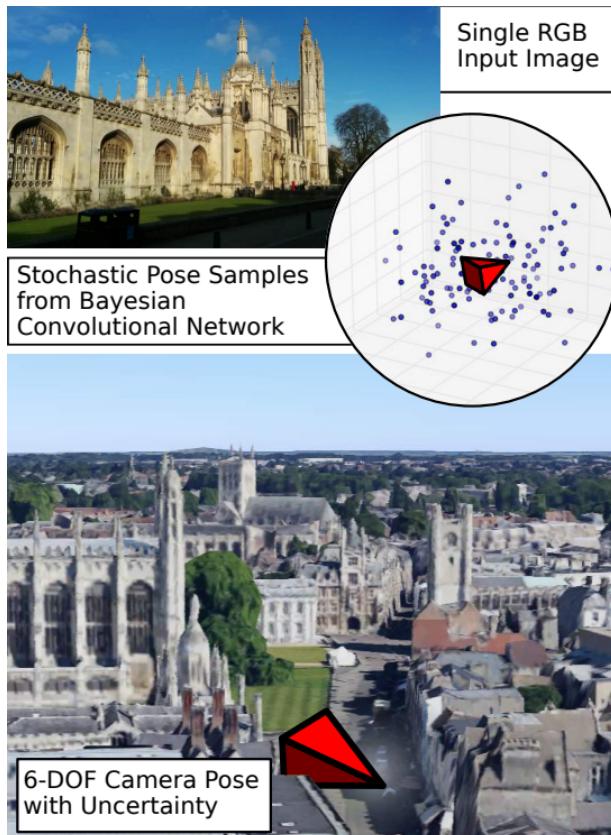
#### **Phương pháp sử dụng hàm mất mát Euclidean cố định:**

PoseNet [25] là công trình đầu tiên huấn luyện mô hình mạng nơ-ron tích chập để hồi quy vị trí máy ảnh từ một ảnh RGB, hoàn toàn không phụ thuộc vào bất kỳ cơ chế bên ngoài nào khác. Vào thời điểm ra mắt, PoseNet đã cho thấy sự vững chắc của mô hình vượt trội so với phương pháp tái tạo kiến trúc từ chuyển động dựa trên cơ chế "biến đổi tính năng bất biến tỷ lệ" (Scale-invariant Feature Transform Structure from Motion): độ hiệu quả của kiến trúc về sau giảm mạnh nếu độ lớn của tập dữ liệu huấn luyện giảm đến một mức nhất định. Hàm mất mát Euclidean của PoseNet được định nghĩa như sau:

$$loss(I) = \|\hat{x} - x\|_2 + \beta \left\| \hat{q} - \frac{q}{\|q\|} \right\|_2$$

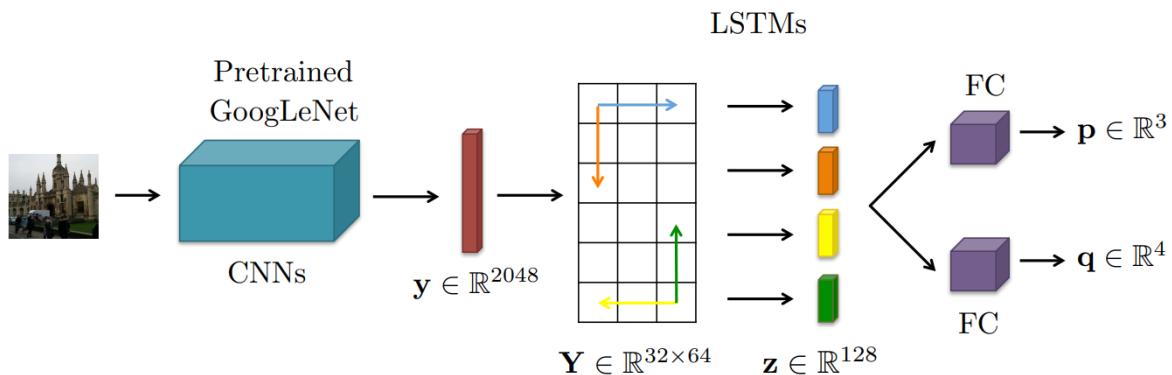
Kế thừa từ PoseNet, đã có nhiều công trình và bài báo tìm cách cải thiện phương pháp định vị hoặc thay thế hàm mất mát nhằm nâng cao hiệu suất chung của toàn kiến trúc. Với các công trình có mong muốn cải thiện hàm mất mát của mô hình, một

chiến thuật chung là kết hợp hàm mất mát Euclidean và phương pháp giảm độ dốc Stochastic. Về mặt cải thiện hiệu quả định vị cũng như tìm hiểu về độ thiêu chính xác của mô hình, một nhóm tác giả đề xuất thêm xác suất Bernoulli vào mô hình nơ-ron tích chập [23] nhằm xác định độ thiêu chính xác của mô hình. Ý tưởng chính của phương pháp này là xác định và tận dụng độ thiêu chính xác để dự đoán sai số trong định vị, phương pháp này đã cải thiện độ chính xác cho PoseNet cho cả những cảnh ngoài trời và bên trong nhà.



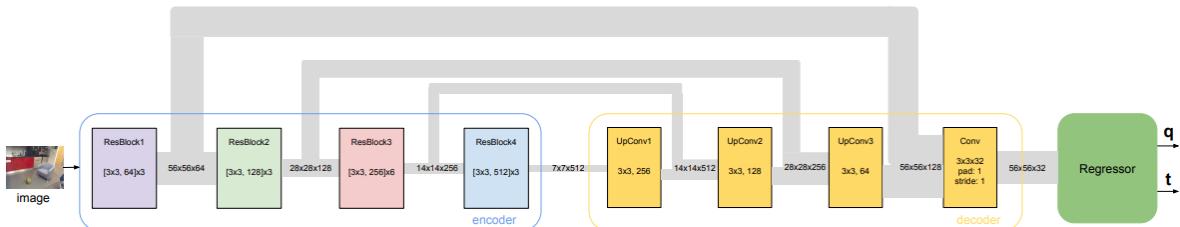
Hình 2.2: Minh họa mô hình CNN được áp dụng phân phôi Bernoulli

Từ những thông tin được nêu ra trong bài báo [25], ta biết được rằng mô hình PoseNet có một lớp kết nối đầy đủ với 2048 chiều, tạo điều kiện cho việc áp dụng một lớp bộ nhớ dài ngắn hạn để giảm chiều đặc trưng giúp cải thiện độ chính xác định vị [54, 55]. Nhóm tác giả Watch và cộng sự [54] đề xuất tận dụng các lớp bộ nhớ dài ngắn hạn lên đầu ra của PoseNet để giảm chiều và chọn ra những đặc trưng hữu ích nhất cho bài toán định vị vị trí. Các thí nghiệm đo lường cho thấy phương pháp này vượt trội hơn PoseNet khoảng 30% về sai số vị trí và 55% về sai lệch góc quay.



Hình 2.3: Minh họa kiến trúc mô hình LSTM PoseNet

Để cải thiện độ chính xác định vị, một kiến trúc đồng hồ cát được đề xuất với việc thêm một phần chức năng mã hóa thông tin hữu ích từ kiến trúc vật thể và một phần chức năng thu chi tiết vật thể mịn. Hourglass PoseNet [32] có kiến trúc gồm ba thành phần chính là bộ mã hóa, bộ giải mã và bộ hồi quy. Mô hình này sử dụng một mô hình ResNet34 đã được tinh chỉnh làm bộ mã hóa - giải mã. SVS PoseNet [34] dùng mô hình VGG16 kết hợp thêm hai lớp kết nối đằng sau để có thể dự đoán riêng vị trí và góc quay. BranchNet [57] sử dụng mô hình mạng hai nhánh học đồng thời biểu diễn góc quay và độ dời để giảm thiểu độ thưa của các vị trí được lấy mẫu một cách hiệu quả. Dù hướng tiếp cận có sự khác biệt, các công trình trên đều có cùng hàm mất mát với PoseNet.

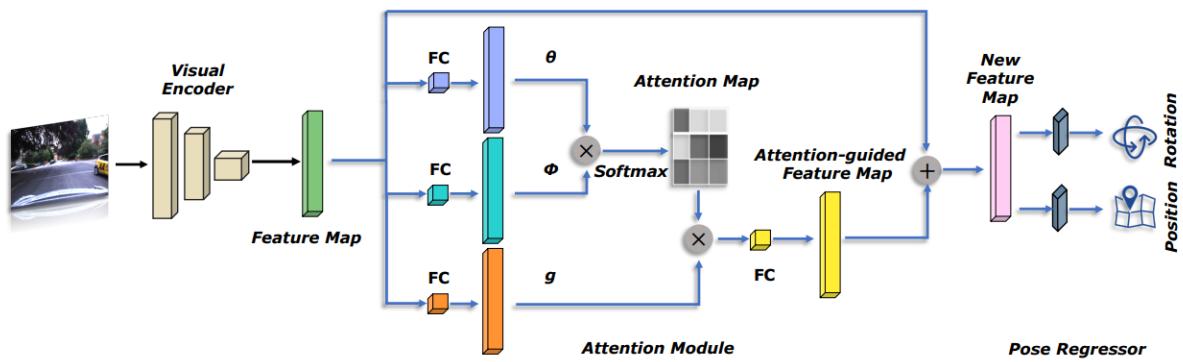


Hình 2.4: Minh họa kiến trúc mô hình Hourglass PoseNet

#### Phương pháp sử dụng hàm mất mát có trọng số học được:

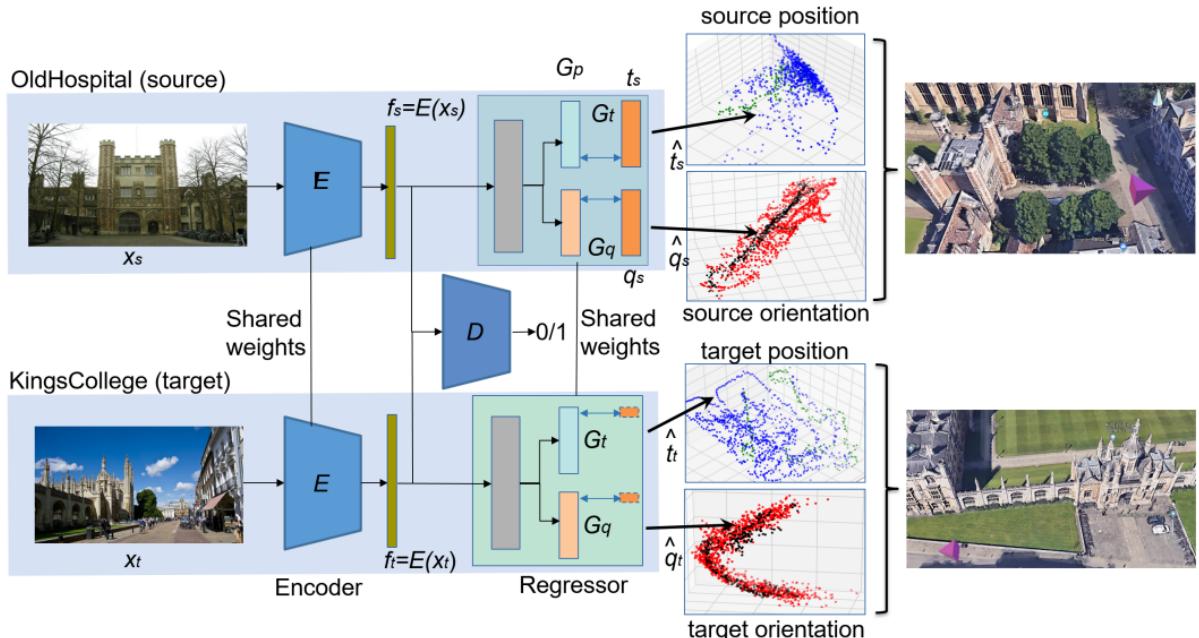
Để học được thông tin về vị trí và góc quay từ dữ liệu ảnh, hàm mất mát cố định Euclidean áp dụng các siêu tham số cân bằng để giúp việc học thông tin vị trí và góc quay một cách độc lập, tuy nhiên để học trọng số thì rất tốn kém. Geometric PoseNet [24] đề xuất sử dụng hàm mất mát vị trí có trọng số học được để cân bằng hiệu suất và cải thiện độ ổn định. Khi so sánh với PoseNet, phương pháp này giữ lại độ mở rộng và độ chắc chắn mà không cần chỉnh sửa các siêu tham số cố định cân bằng trong hàm mất mát.

AtLoc [55] thêm vào mô hình một mô-đun tập trung (Attention Module) trước khi xác định các tọa độ hồi quy để ép buộc mạng phải tập trung vào phần chính - phần mang nhiều thông tin hữu ích nhất của hình ảnh đầu vào. Ngoài ra, AtLoc sử dụng ResNet34 được huấn luyện sẵn trên tập dữ liệu ImageNet làm bộ mã hóa, sau đó hồi quy lớp kết nối đằng sau 2048 chiều của PoseNet.



Hình 2.5: Minh họa kiến trúc mô hình AtLoc

AdPR [12] thêm một mạng phân biệt và học đối lập. Điều này không chỉ hồi quy vị trí mà còn tinh chỉnh vị trí. Khi trích xuất đặc trưng, AdPR áp dụng mạng ResNet-18, vì nó có thể đạt được hiệu suất tốt nhất so với VGG16 và AlexNet. APANet [14] cũng sử dụng một mạng đối lập để tạo ra hình ảnh liên quan đến hình ảnh đầu vào để ước lượng tốt hơn vị trí của máy ảnh. Một mô-đun Dropout được thêm trước bộ mã hóa trích xuất đặc trưng để xuất ra nhiều khả năng không chắc chắn, điều này có thể cải thiện độ chắc chắn của mô hình dưới điều kiện thử thách như thay đổi vị trí, thời tiết,... . Sau khi trích xuất, mô-đun tập trung tự động được thêm để điều chỉnh lại trọng số bản đồ đặc trưng.

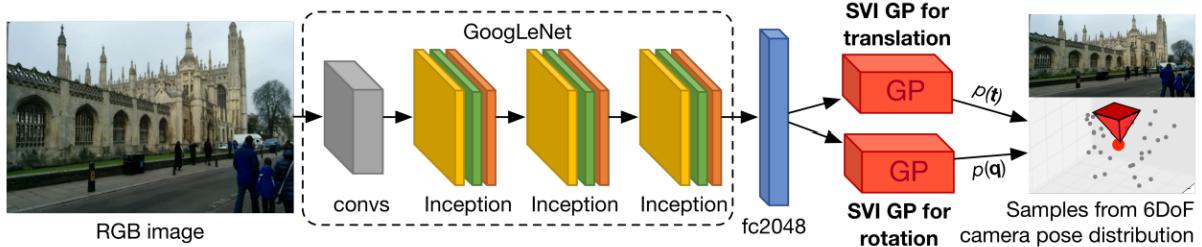


Hình 2.6: Minh họa kiến trúc mô hình APANet

### **Phương pháp sử dụng hàm mất mát khác:**

Không dùng đến cả hàm mất mát cố định hoặc những hàm mất mát có trọng số học được, một số nhóm nghiên cứu đề xuất nên cân nhắc sử dụng các mô-đun khác để cải thiện hiệu suất định vị. GeoPoseNet [24] đề xuất sử dụng hàm mất mát tái chiêu: đặc tả sai sót tái chiêu của cảnh. Hàm mất mát tái chiêu chuyển mất mát chung học được thành khác biệt tọa độ ảnh, do đó có thể thay đổi trọng số giữa vị trí và góc quay, tùy thuộc vào các cảnh khác nhau trong quá trình huấn luyện mô hình. GPoseNet

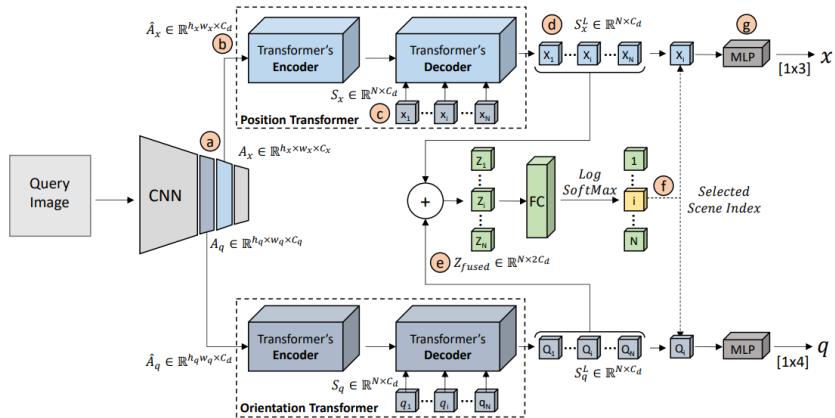
[13] xây dựng mô hình mới bằng cách thêm vào hai bộ "Hồi quy quá trình Gaussian suy luận biến phân ngẫu nhiên" (Stochastic Variational Inference Gaussian Process Regressions - SVI GPs) sau lớp kết nối dày đủ để học phân phối xác suất của vị trí - hướng quay đầu ra và giảm tần suất sử dụng siêu tham số. Hàm mất mát của GPoseNet kết hợp hàm mất mát SVI GPs sử dụng ranh giới điều kiện dưới của hai xác suất tích lũy log  $L_s vi$  và hàm mất mát CNN với siêu tham số  $\beta_{g_i}$  và  $\beta_{n_q}$  của PoseNet.



Hình 2.7: Minh họa kiến trúc mô hình GPoseNet

Các nghiên cứu về việc áp dụng hàm mất mát có xu hướng tiên tới việc tự động hóa, không sử dụng siêu tham số và mang nhiều thông tin hơn để giảm việc sử dụng các tham số cố định. Một hàm mất mát cố định tính tổng độ dời và hướng quay sử dụng một yếu tố cân bằng để cân bằng các mục có trọng số khác nhau, điều này đòi hỏi một khoảng thời gian dài để tối ưu hóa mất mát trên dữ liệu huấn luyện. Sau đó, một hàm mất mát có trọng số học được đã được đề xuất bằng cách thêm không đột phân biệt đồng nhất để tự động cân bằng mất mát dịch và hướng, tránh sử dụng siêu tham số và vượt qua hiệu suất của phương pháp mất mát cố định. Ngoài phương pháp hàm mất mát cố định và hàm mất mát với trọng số học được, các phương pháp khác đề xuất sử dụng mất mát lỗi tái chiếu và mất mát GPoseNet để thêm các định dạng thông tin khác như phân phối xác suất của vị trí - hướng quay đầu ra để cải thiện hàm mất mát.

Một nhóm nghiên cứu [43, 44] đề xuất áp dụng mô hình Transformer vào tác vụ hồi quy vị trí tuyệt đối. Mô hình nhận vào một ảnh đơn và sử dụng một CNN làm bộ trích xuất đặc trưng, sau đó các bản đồ đặc trưng được truyền song song qua hai nhánh: mỗi nhánh là một mô hình Transformer đảm nhiệm một tác vụ riêng lần lượt là hồi quy vị trí và hồi quy hướng quay. Mô hình sử dụng hàm mất mát tương tự với PoseNet.



Hình 2.8: Minh họa kiến trúc mô hình Multi-Scene Transformer

## Hồi quy vị trí tuyệt đối đa ảnh - Absolute pose regression through auxiliary image sequence

Một phương pháp khác được áp dụng để hồi quy vị trí tuyệt đối là áp dụng học có hỗ trợ với một chuỗi ảnh. Học có hỗ trợ được định nghĩa là phương pháp cải thiện hiệu suất của một tác vụ chính thông qua việc học cùng lúc các tác vụ hỗ trợ. Phương pháp học này giúp mô hình phát triển các biểu diễn dữ liệu tốt hơn. Bằng cách tận dụng các tác vụ hỗ trợ có liên quan khác trong quá trình học, hiệu suất của tác vụ chính có thể được cải thiện. Ở đây, học có hỗ trợ ám chỉ việc kết hợp APR với các tác vụ phụ có liên quan như đo lường cảm biến trực quan. Hàm mất mát của các phương pháp học có hỗ trợ thường bao gồm hàm mất mát của APR kết hợp với hàm mất mát của các phương pháp phụ trợ, thậm chí có thể kết hợp cả hàm mất mát của APR và RPR. Khác với các phương pháp hồi quy vị trí tuyệt đối đơn ảnh, phương pháp học có hỗ trợ học từ các cặp ảnh với bản chất là học cách xác định vị trí tuyệt đối bằng cách đánh giá trước hết vị trí tương đối với các ràng buộc phụ thuộc.

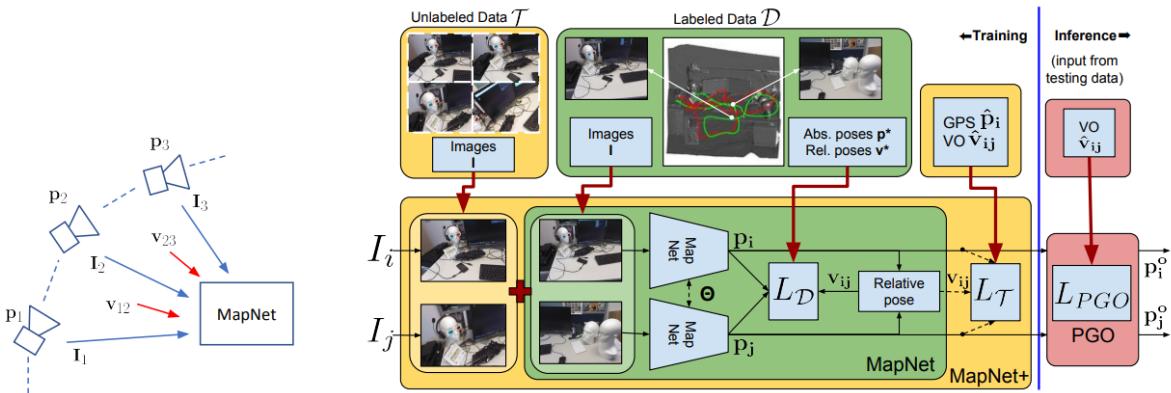
MapNet [11] đề xuất thêm một thuật ngữ mất mát lấy từ các cặp ảnh làm một ràng buộc hình học, điều này đã có thể cải thiện mạnh mẽ hiệu suất khả năng định vị. Về hàm mất mát, MapNet giảm thiểu tối đa cả mất mát vị trí tuyệt đối cho mỗi hình ảnh và mất mát vị trí tương đối giữa các cặp hình ảnh:

$$l(I_{total}) = l(I_i) + \alpha \sum_{i \neq j} loss(I_{ij})$$

Trong đó,  $loss(I_{ij})$  ám chỉ vị trí máy ảnh tương đối  $p_i$  và  $p_j$  giữa các cặp hình ảnh  $I_i$  và  $I_j$ , được tính bởi hàm mất mát với trọng số có thể học được.

Thêm vào đó, MapNet chuyển một giá trị quaternion thành logarit của giá trị đó - biểu diễn phép quay ba độ tự do (3DoF) với ba chiều chưa bị tham số hóa quá mức.  $logq$  được biểu diễn như dưới đây, với  $u$  và  $v$  đại diện cho phần thực và ảo của một quaternion đơn vị:

$$logq = \begin{cases} \frac{v}{\|v\|} \cos^{-1} u, & \|v\| \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

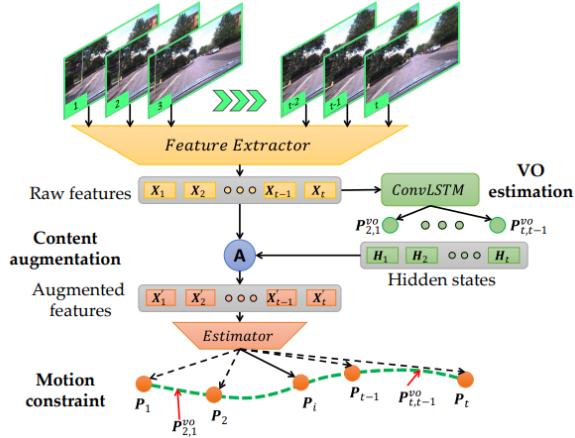


Hình 2.9: Minh họa kiến trúc mô hình MapNet

Nhóm tác giả Xue và những cộng sự [59] cũng có một hướng tiếp cận tương tự khi hồi quy vị trí máy ảnh thông qua những ràng buộc về không gian - thời gian, trong đó đặc trưng cục bộ cải thiện định vị toàn cục - gọi là "Cục bộ hỗ trợ toàn cục" (Local Support Global - LSG). Thêm vào đó, LSG đề xuất sử dụng một đánh giá đã được “tăng cường nội dung” để ước lượng lỗi vị trí và tinh chỉnh dựa trên chuyển động, để tối ưu hóa dự đoán vị trí thông qua các ràng buộc chuyển động. LSG sử dụng một hàm mất mát vị trí toàn cầu  $L_g$  lấy từ hồi quy tuyệt đối, hàm mất mát hồi quy đo lường

cảm biến trực quan  $L_{vo}$ , các ràng buộc hình học và hàm mất mát liên kết chuyển động  $L_{joint}$  để tối ưu hóa hồi quy vị trí như sau:

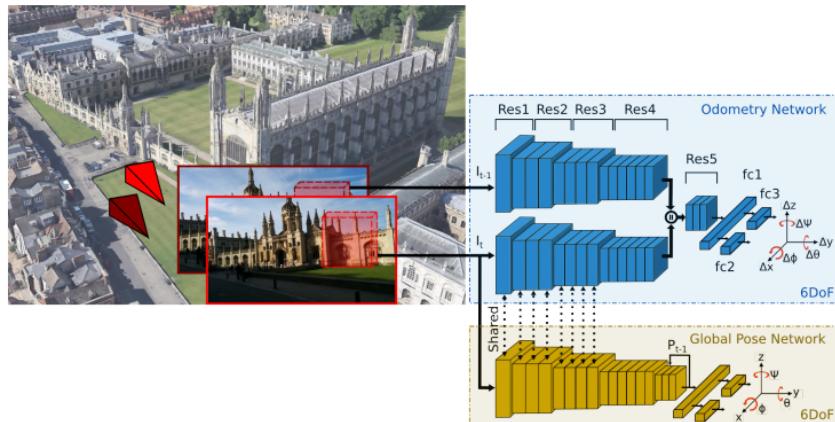
$$l_{total} = l_g + l_{vo} + l_{joint}$$



Hình 2.10: Minh họa kiến trúc mô hình LSG

VlocNet [52] cũng học đồng thời đo lường cảm biến trực quan như một tác vụ phụ để hồi quy vị trí toàn cục với hai mạng phụ. Mất mát nhất quán hình học được điều chỉnh để giảm thiểu tối đa lỗi vị trí, được định nghĩa như sau:

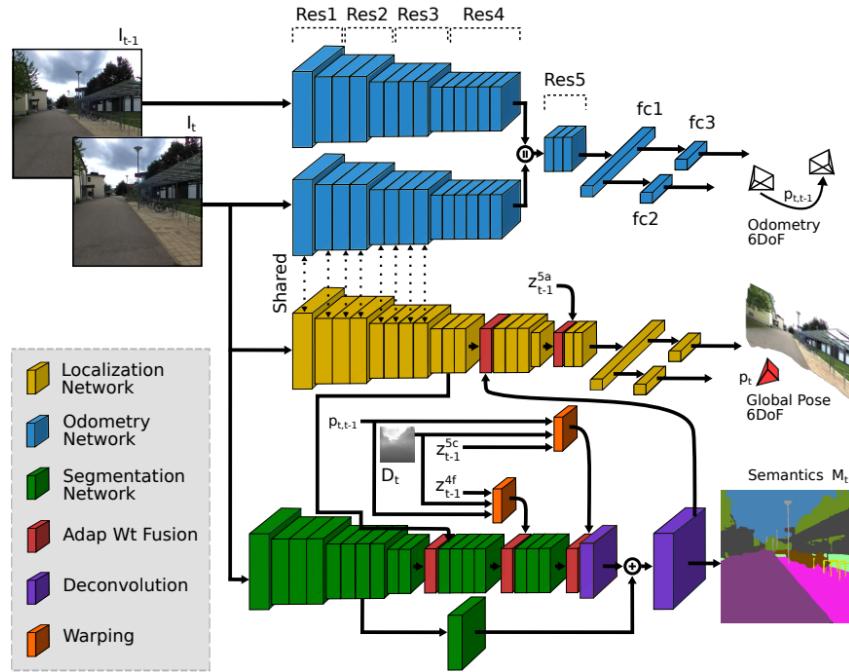
$$l(I_{total}) = (I_{i_x} + I_{ij_x})\exp(-\hat{s}_x) + (I_{i_q} + I_{ij}^q)\exp(-\hat{s}_q) + \hat{s}_q$$



Hình 2.11: Minh họa kiến trúc mô hình VlocNet

VlocNet++ [39] giới thiệu kiến thức ngữ nghĩa vào hồi quy vị trí, kết hợp thông tin hình học-thời gian với các đặc trưng ngữ nghĩa cùng một lúc. Hàm mất mát của VlocNet++ kết hợp hồi quy vị trí toàn cục, mất mát đo lường cảm biến trực quan và mất mát Entropy chéo cho mất mát phân đoạn ngữ nghĩa cùng một lúc, với ba yếu tố  $\hat{s}_{log}$ ,  $\hat{s}_{vo}$  và  $\hat{s}_{seg}$  để cân bằng ba thành phần này:

$$l(I_{total}) = l_{loc}\exp(-\hat{s}_{loc}) + \hat{s}_{loc} + l_{vo}\exp(-\hat{s}_{vo}) + \hat{s}_{vo} + l_{seg}\exp(-\hat{s}_{seg}) + \hat{s}_{seg}$$

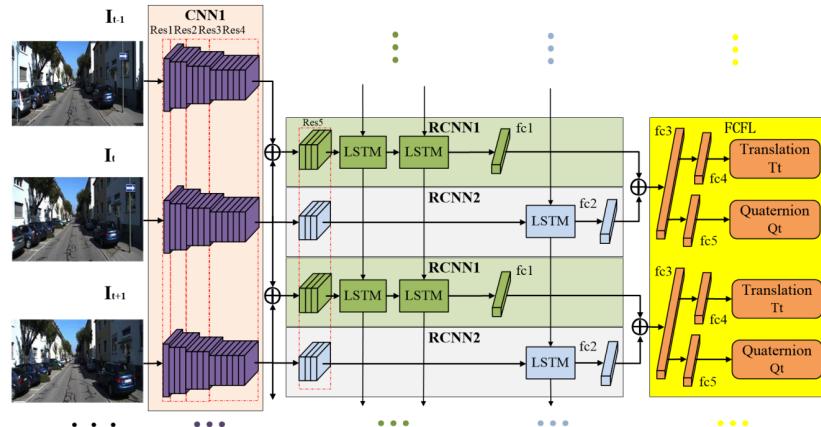


Hình 2.12: Minh họa kiến trúc mô hình VlocNet++

Một bản mở rộng của AtLoc, AtLocPlus [55] cũng kết hợp các ràng buộc thời gian để học cùng lúc mất mát vị trí tuyệt đối và mất mát vị trí tương đối, dẫn đến hiệu suất tốt hơn AtLoc trong việc sử dụng một đầu vào ảnh đơn. AtLocPlus sử dụng hàm mất mát giống với MapNet.

DGRNet [29] đề xuất một kiến trúc với một mạng con hồi quy vị trí tương đối RCNN1, một mạng con hồi quy vị trí toàn cục RCNN2 và lớp kết hợp kết nối đầy đủ dùng để trích xuất đặc trưng từ ảnh. Ràng buộc biến đổi chéo (Cross transformation constraint – CTC) và sai số toàn phương trung bình (Mean squared error – MSE) được áp dụng vào hàm mất mát để cải thiện hiệu suất hồi quy. DGRNet đã sử dụng kết hợp hàm mất mát tương đối, toàn cục, CTC  $\hat{I}_i$  và sự thật nền tảng  $\hat{P}_i$  như sau:

$$w = \underset{w}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^N \left( \sum_{k=0}^6 (I_k^i) + \sum_{j=0}^4 \| P^i j - \hat{P}_j^i \|_2^2 \right)$$



Hình 2.13: Minh họa kiến trúc mô hình DGRNet

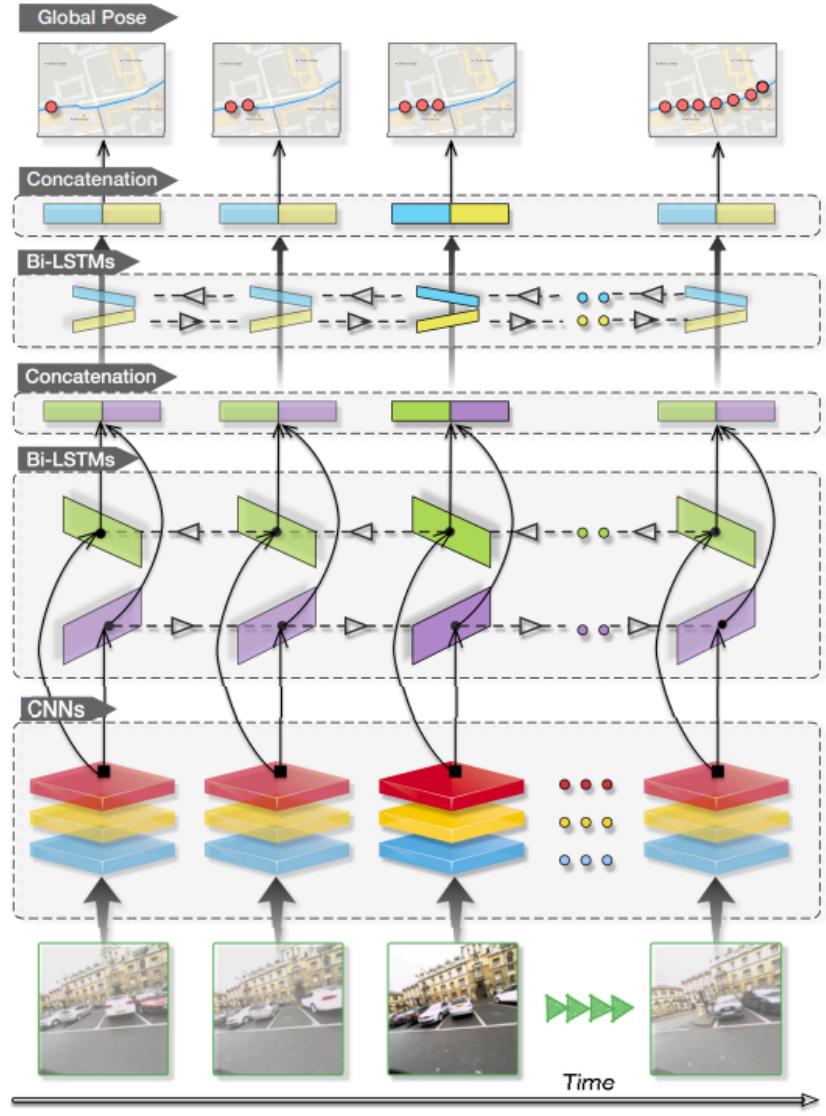
## Hồi quy vị trí tuyệt đối qua đoạn phim - Absolute pose regression through video

Không chỉ đơn ảnh hay đa ảnh, ngay cả đoạn phim cũng có thể được sử dụng để thêm một ràng buộc thời gian có tính trơn tru hơn cho hồi quy vị trí. Các đoạn phim hay các dữ liệu cảm biến khác đều có thể được truy cập dễ dàng bởi các thiết bị di động. Đoạn phim có thể được đồng bộ hóa với các dữ liệu đầu vào khác như đo lường cảm biến trực quan, các cảm biến IMU như đồng hồ tăng tốc và đồng hồ quay và dữ liệu GNSS bằng thông tin thời gian, cụ thể là bằng cách căn chỉnh các mốc thời gian. Với một quy trình tương tự như các phương pháp ARP dựa trên hình ảnh đơn và chuỗi hình ảnh, ARP dựa trên đoạn phim cũng hồi quy độ dời và hướng quay thông qua bộ trích xuất đặc trưng là một mạng nơ-ron tích chập và bộ hồi quy vị trí cục bộ.

VidLoc [16] đề xuất một mô hình hồi quy vị trí máy ảnh dựa trên việc kết hợp CNN – RNN, mục đích là có thể khiến quá trình dự đoán vị trí từ ảnh hay một đoạn phim được trơn tru hơn. Mạng được xây dựng bằng cách sử dụng GoogLeNet Inception [48] mà không dùng đến lớp kết nối đầy đủ để trích xuất đặc trưng ảnh và một mô-đun LSTM hai chiều để mô hình hóa thông tin thời gian với các ô nhớ. Hàm mất mát của VidLoc được tính bằng tổng của lỗi vị trí và lỗi hướng quay từ đầu ra của LSTM như sau:

$$l = \sum_{t=1}^T \alpha_1 \|x_t - \hat{x}_t\| + \alpha_2 \|q_t - \hat{q}_t\|$$

Với  $[x_t, q_t]$  và  $[\hat{x}_t, \hat{q}_t]$  đại diện cho sự thật nền tảng và giá trị dự đoán cho độ dời vị trí và hướng quay.



Hình 2.14: Minh họa kiến trúc mô hình VidLoc

MapNet+ [11] và MapNet+PGO [11] mang cùng một kiến trúc mạng với MapNet trích xuất đặc trưng qua mạng ResNet34 và dùng một lớp tổng hợp trung bình toàn cục. Không chỉ dùng mắt mèo vị trí tuyệt đối, mắt mèo đo lường cảm biến trực quan cũng được tính toán để cải thiện hiệu suất dự đoán vị trí của MapNet. Phương pháp này cũng đồng thời tích hợp dữ liệu GNSS và IMU để giúp cải thiện hồi quy vị trí. Điều này giúp kết hợp dữ liệu đã được gắn nhãn và dữ liệu chưa gắn nhãn từ VO hay cảm biến để phục vụ cho việc học tự giám sát và đã thể hiện hiệu suất tốt dưới những điều kiện khó khăn, thử thách như thay đổi môi trường ngoài, thiếu sáng,... .

$$l = l_{labelleddata} + l_{unlabelleddata}$$

Với mắt mèo từ dữ liệu chưa gắn nhãn có thể được tính toán thông qua việc kết hợp vị trí máy ảnh tương đối  $v_i, j$  và VO  $\hat{v}_i, j$  hay các cảm biến khác như IMU và GNSS.

MapNet+PGO đã có thể cải thiện hiệu suất đồng thời giảm thiểu chi phí tính toán thông qua việc sử dụng cải thiện đồ thị vị trí (PGO) để kết hợp kết quả vị trí MapNet+ và VO.

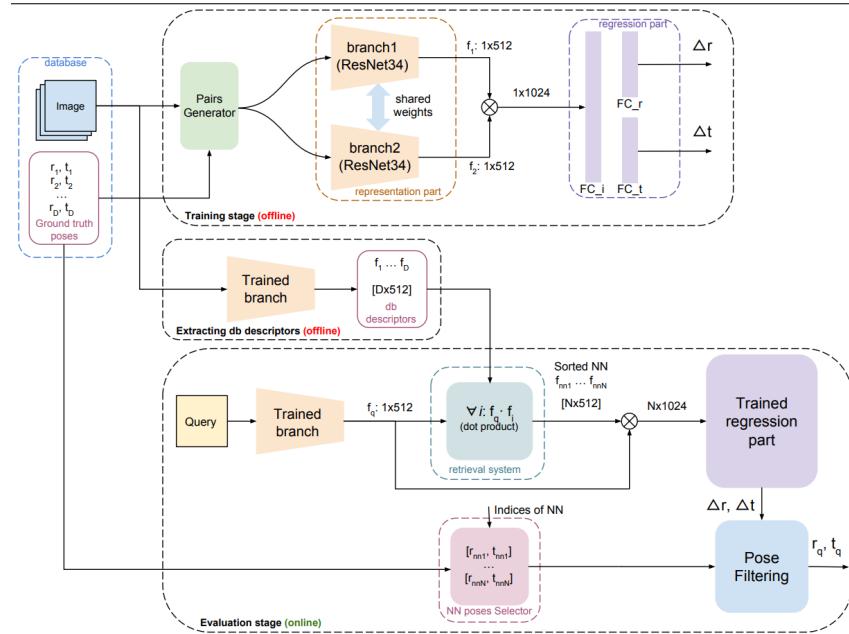
### 2.3.2 Hồi quy vị trí tương đối - Relative Pose Regression

Mô hình hồi quy vị trí tuyệt đối học cách ánh xạ các pixel của ảnh sang vị trí của máy ảnh, thường được quyết định bởi hệ trục tọa độ của chính cảnh vật cụ thể mà máy ảnh đang chụp. Khác với hồi quy vị trí tuyệt đối, các phương pháp mang hướng tiếp cận hồi quy vị trí tương đối (Relative Pose Regression) chỉ tính toán vị trí tương đối của ảnh và thường được huấn luyện trên những tập dữ liệu đa cảnh để tăng cường khả năng mở rộng mô hình đầu cuối.

#### Hồi quy vị trí tương đối thông qua truy xuất rõ ràng - Relative camera pose regression through explicit retrieval

Quy trình hồi quy vị trí tương đối của máy ảnh có thể được hiểu như một quy trình bao gồm truy xuất ảnh có độ tương đồng cao nhất trong kho dữ liệu với ảnh nhận đầu vào và sau đó dự đoán vị trí tương đối giữa chúng để lấy được vị trí tuyệt đối của ảnh nhận đầu vào. Cho một ảnh  $I_a^c$  được chụp từ máy ảnh  $c$ , thông qua các phương pháp truy xuất ảnh từ kho dữ liệu, chúng ta có được ảnh có độ tương đồng cao nhất  $I_b^c$ . Nếu có được vị trí nền tảng  $p_b$  của ảnh  $I_b^c$  và vị trí tương đối hai ảnh là  $p_{a \rightarrow b}$ , vị trí tuyệt đối  $p_a$  của ảnh đầu vào  $I_a^c$  có thể được xác định bằng các phép biến đổi toán học.

NNnet [27] là công trình đầu tiên đề xuất một phương pháp hồi quy vị trí tương đối dựa trên truy xuất ảnh. Đầu vào của phương pháp này là một ảnh và một kho dữ liệu ảnh có bao gồm vị trí nền tảng. Một tập các cặp ảnh được tận dụng để hồi quy vị trí tương đối thông qua một mạng Siamese với hai nhánh ResNet34 đã được hiệu chỉnh và một hàm mất mát cố định. Ảnh có độ tương đồng gần nhất với ảnh nhận đầu vào có thể được tính toán xác định thông qua bộ trích xuất đặc trưng hình thành bởi nhánh mạng CNN, sau đó vị trí tương đối và vị trí nền tảng của ảnh trích xuất sẽ kết hợp để tính toán xác định vị trí tuyệt đối của ảnh đầu vào.

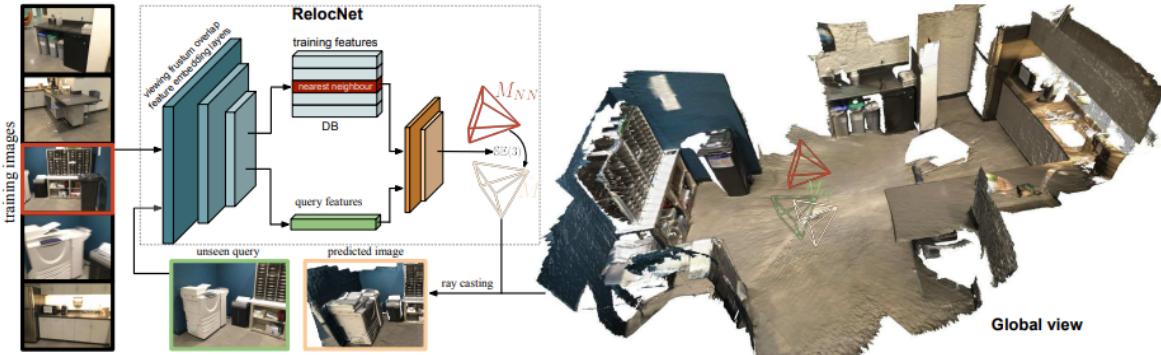


Hình 2.15: Minh họa kiến trúc mô hình NNet

RelocNet [8] cải tiến NNet với việc học liên tục thuộc đo với mục đích học các đặc trưng ảnh toàn cục với một góc nhìn chéo của máy ảnh để cải thiện kết quả, mất mát vị trí tương đối cũng được áp dụng. Mất mát vị trí tương đối học sự khác biệt

vị trí giữa hai ma trận vị trí bằng cách sử dụng một biểu diễn ma trận cho hướng quay và độ dời vị trí. Hàm mất mát huấn luyện được định nghĩa như sau:

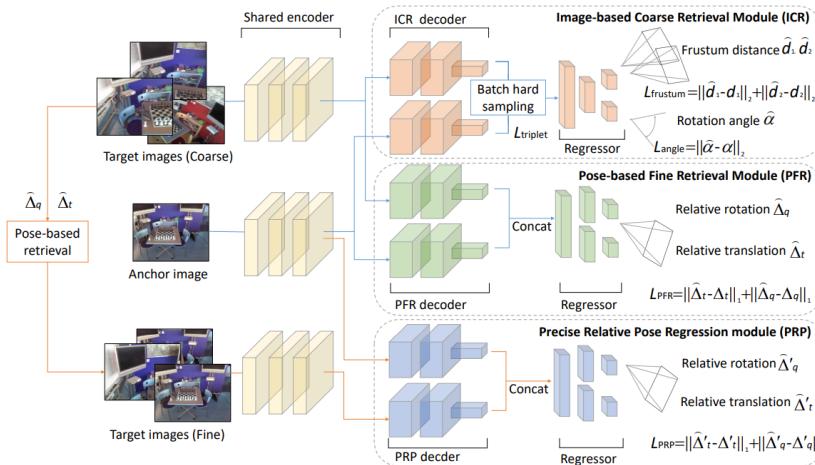
$$l = \alpha l_{SE(3)} + \beta l_{frustum}$$



Hình 2.16: Minh họa kiến trúc mô hình RelocNet

Để giải quyết vấn đề hiệu suất giới hạn của các phương pháp hồi quy vị trí tương đối tiền nhiệm do sử dụng cùng đặc trưng cho cả hai bước truy xuất và hồi quy, CamNet [17] đề xuất một quy trình chia làm ba bước: truy xuất thô, truy xuất mịn và hồi quy vị trí. Kiến trúc mô hình được xây dựng dựa trên kiến trúc Siamese với ba nhánh mỗi bước. Kiến trúc thô-sang-mịn này đã mang lại những cải tiến về hiệu suất hồi quy cũng như khả năng mở rộng. Hàm mất mát của CamNet lấy ý tưởng dựa trên RelocNet, được định nghĩa như sau:

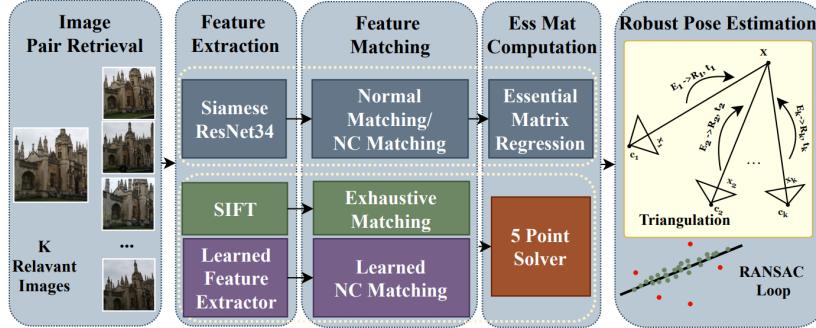
$$l = l_{frustum} + l_{angle} + l_{triplet} + l_{PFR} + l_{PRP}$$



Hình 2.17: Minh họa kiến trúc mô hình CamNet

Qunjie Zhou và những cộng sự [63] sau khi phân tích các phương pháp hồi quy vị trí dựa trên việc truy xuất ảnh đã đề xuất một kiến trúc mới sử dụng ma trận thiết yếu và RANSAC để tính toán vị trí tuyệt đối. Một mạng Siamese ResNet34 với một lớp tìm sự tương ứng cố định (EssNet) và một lớp tìm sự tương ứng đồng thuận lân cận (NC-EssNet) được học để tạo ra một bản đồ điểm tương ứng phục vụ cho mục đích hồi quy về sau của ma trận thiết yếu. Hàm mất mát cải tiến khoảng cách Euclidean giữa ma trận thiết yếu với hai vec-tơ 9 chiều.

$$l_{ess}(E^*, E) = \|e - e^*\|_2$$

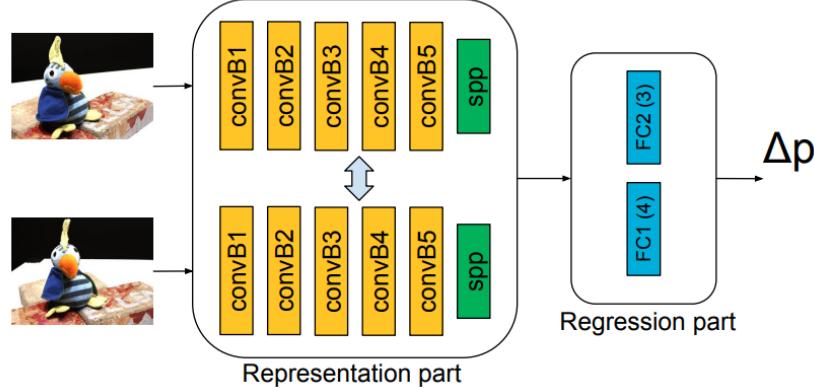


Hình 2.18: Minh họa kiến trúc mô hình EssNet

### Hồi quy vị trí tương đối thông qua mạng CNN ngầm - Relative camera pose regression through implicit CNN

Để tránh việc phải thu thập và xây dựng một kho dữ liệu khổng lồ cũng như tốn kém thời gian thử nghiệm, một số phương pháp tìm cách hồi quy vị trí tương đối của máy ảnh thông qua một mạng Nơ-ron tích chập ngầm.

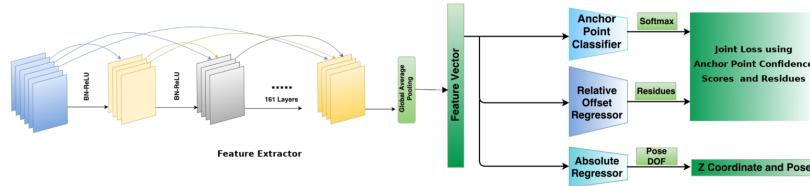
Relative NN [33] đề xuất một phương pháp đầu-cuối để hồi quy vị trí tương đối giữa hai máy ảnh bằng hai ảnh đầu vào. Kiến trúc mô hình là một mạng Nơ-ron hỗn hợp Siamese hai nhánh sử dụng mạng AlexNet đã được huấn luyện từ trước được dùng cho việc hồi quy với hàm mất mát Euclidean cố định.



Hình 2.19: Minh họa kiến trúc mô hình Relative Neural Network

AnchorNet [40] tìm cách khắc phục vấn đề định vị bằng cách định nghĩa các địa danh thành các điểm mốc để học các điểm mốc tương đối của ảnh đầu vào cũng như độ lệch của chúng. Mô hình đa nhiệm bao gồm việc phân loại hình ảnh truy vấn đầu vào theo các điểm mốc cụ thể và tìm sự chênh lệch so với điểm mốc đã phân loại, điều này dẫn đến việc hình thành hàm mất mát.  $\hat{C}$ ,  $X$ , và  $Y$  đại diện cho kết quả phân loại và sự chênh lệch với sự thật nền tảng.

$$l = \sum_i [(X_i - \hat{X}_i)^2 + (Y_i - \hat{Y}_i)^2] \hat{C}^i$$



Hình 2.20: Minh họa kiến trúc mô hình AnchorNet

### 2.3.3 Tái tạo kiến trúc từ chuyển động - Structure From Motion

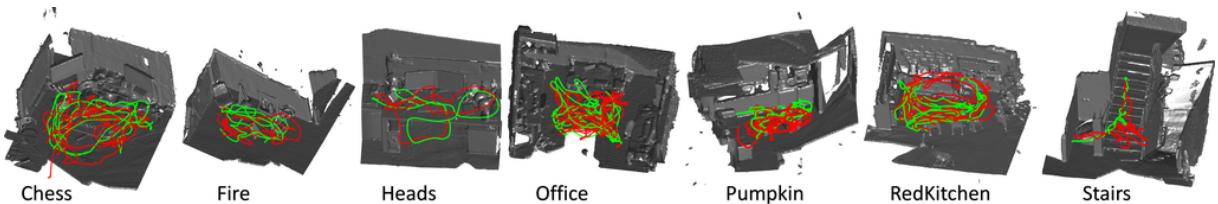
## 2.4 Phân tích và tổng hợp

## 2.5 Một số tập dữ liệu phổ biến được sử dụng

### 2.5.1 Tập dữ liệu trong không gian nhỏ

#### 7Scenes

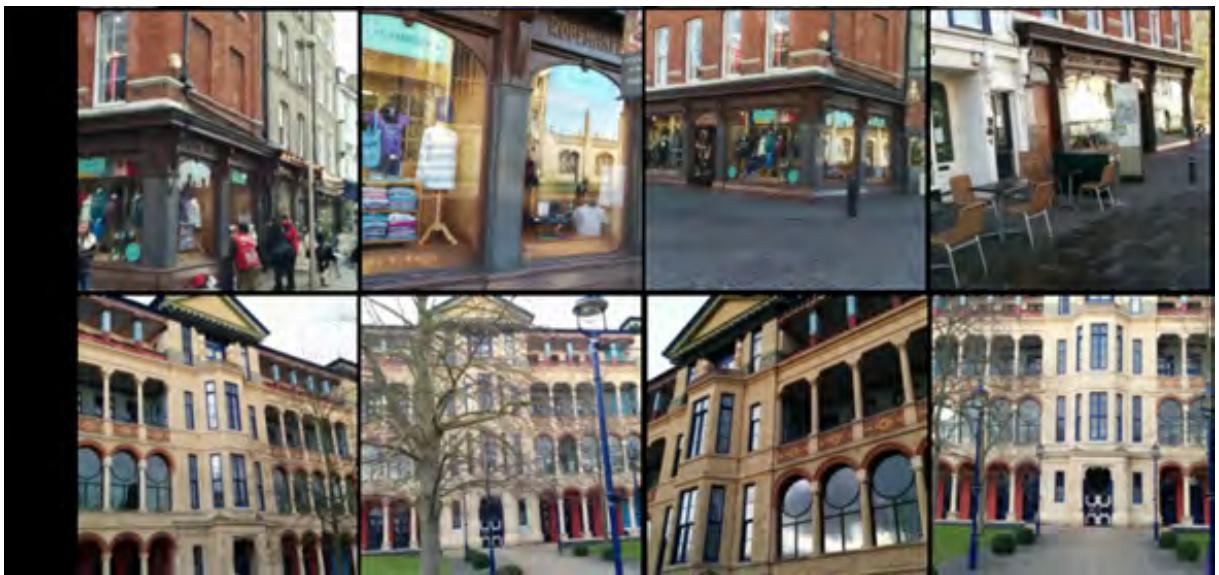
Tập dữ liệu 7-Scenes [45] bao gồm các ảnh RGB-D thuộc bảy khung cảnh khác nhau được chụp từ một máy ảnh cầm tay Kinect RGB-D ở độ phân giải 640x480. Bảy khung cảnh bao gồm: "Chess", "Fire", "Heads", "Office", "Pumpkin", "RedKitchen" và "Stairs". Với mỗi cảnh sẽ có vài chuỗi khung ảnh RGB-D. Mỗi chuỗi bao gồm khoảng từ 1000 đến 5000 khung ảnh. Mỗi khung sẽ gồm: ảnh màu, độ sâu và vị trí.



Hình 2.21: Minh họa tập dữ liệu 7-Scenes

#### Cambridge Landmark

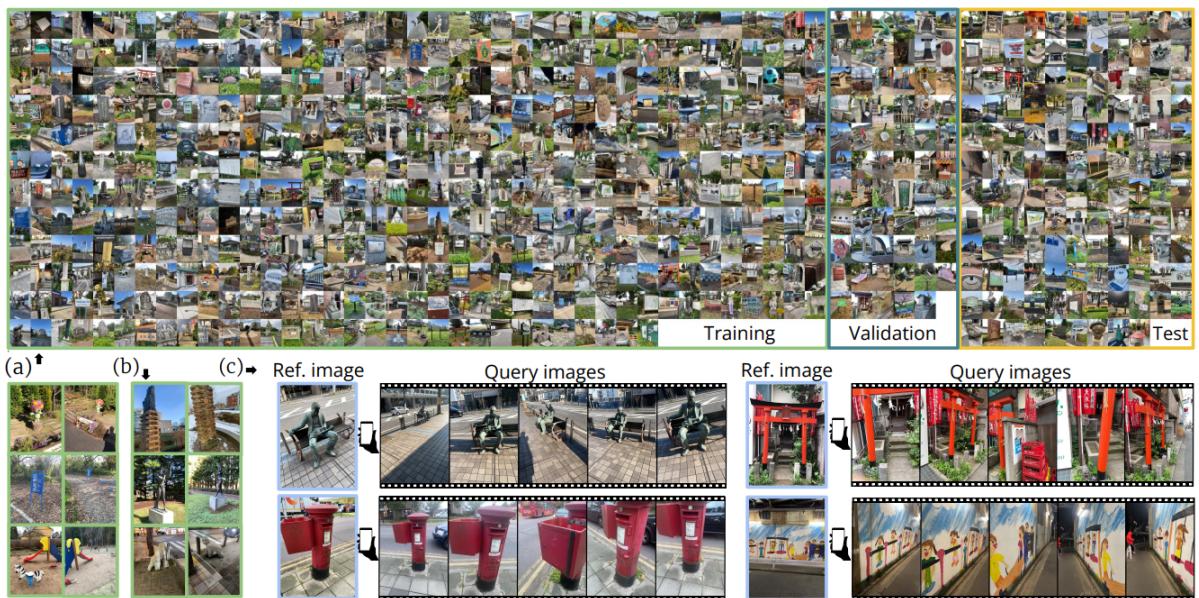
Tập dữ liệu Cambridge Landmarks [25] là một tập dữ liệu định vị thành thị bao gồm năm khung cảnh khác nhau. Các yếu tố dày đặc quan trọng như phương tiện giao thông hay người đi bộ cũng xuất hiện trong tập dữ liệu này, ngoài ra dữ liệu cũng được thu thập ở nhiều thời điểm trong ngày đại diện cho các yếu tố ánh sáng và điều kiện thời tiết khác nhau. Cambridge Landmarks được tạo ra nhờ vào việc áp dụng các kỹ thuật tái tạo kiến trúc từ chuyển động. Một chiếc điện thoại thông minh Google Nexus 5 được một người đi bộ trên phố sử dụng để ghi lại đoạn phim chất lượng cao cho mỗi cảnh. Mỗi đoạn phim sau đó sẽ được lấy mẫu với tần số 2Hz để trích xuất ảnh cho quy trình tái tạo kiến trúc từ chuyển động. Mỗi vị trí máy ảnh sẽ cách nhau khoảng 1m.



Hình 2.22: Minh họa tập dữ liệu Cambridge Landmarks

### Niantic Map-free Relocalization Dataset

Tập dữ liệu Niantic Map-free Relocalization [7] là một tập dữ liệu được thu thập chủ yếu để giúp ích cho phương pháp định vị Map-free [7]. Tập dữ liệu bao gồm 655 cảnh bên ngoài với mỗi cảnh sẽ chứa một "địa điểm đáng chú ý" như một pho tượng, cổng, bảng hiệu,... sao cho địa điểm đó phải được xác định rõ trong một bức ảnh. Các cảnh được chia ra thành 460 cảnh phục vụ cho tác vụ huấn luyện, 65 cảnh phục vụ cho tác vụ kiểm tra quy trình huấn luyện và 130 cảnh phục vụ cho quá trình kiểm thử. Mỗi ảnh trong tập huấn luyện đều được gắn kèm vị trí tuyệt đối. Với tập kiểm thử và kiểm tra quy trình, mỗi cảnh sẽ được kèm theo một ảnh đại diện cũng như vị trí tuyệt đối tại cảnh. Ngoài ra, ma trận tham số nội tại của máy ảnh cũng được gắn kèm theo mỗi ảnh trong tập dữ liệu.



Hình 2.23: Minh họa tập dữ liệu Niantic Map-free Relocalization

## 2.5.2 Tập dữ liệu thành thị

### Aachen Day-Night

Tập dữ liệu Aachen Day-Night [41] bao gồm 14.607 ảnh được chụp với nhiều máy ảnh khác nhau bao phủ cả thành phố Aachen thuộc quốc gia Đức. Các ảnh dữ liệu được chụp ở nhiều thời điểm trong ngày và trong năm, cụ thể là khoảng thời gian trong 2 năm. Hệ quả mang lại là tập dữ liệu bao phủ nhiều điều kiện ngoại cảnh như thời tiết, ánh sáng cũng như sự thay đổi của công trình kiến trúc trong khu vực.



Hình 2.24: Minh họa tập dữ liệu Aachen Day-Night

### Pittsburgh 250k

Tập dữ liệu Pittsburgh 250k [51] là một tập dữ liệu tương đối rộng bao phủ thành phố Pittsburgh của Mỹ. Đây là một tập dữ liệu tương đối phổ biến trong thị giác máy tính, cụ thể là ở tác vụ nhận diện địa điểm trực quan, truy xuất ảnh và bản địa hóa trực quan.

### GSV-Cities

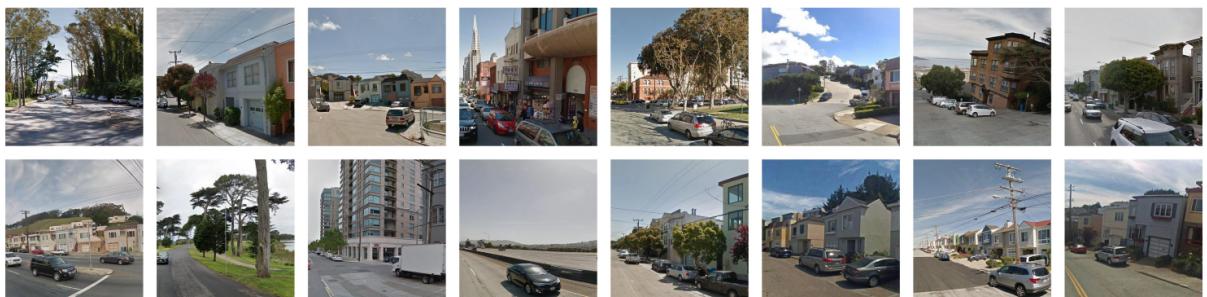
Tập dữ liệu GSV-Cities [4] bao phủ một vùng địa lý cực kỳ rộng lớn với hơn 40 thành phố xuyên lục địa trong khoảng thời gian 14 năm liên tục. GSV-Cities chứa khoảng hơn 530.000 ảnh - khoảng hơn 62.000 vị trí. Mỗi vị trí sẽ có khoảng từ 4 đến 20 ảnh. Đồng thời mỗi vị trí sẽ cách nhau một khoảng ít nhất 100m.



Hình 2.25: Minh họa tập dữ liệu GSV-Cities

### SF-XL

Tập dữ liệu San Francisco Extra Large (SF-XL) [10] được tạo nên từ 3.43 triệu ảnh 360 độ thu thập từ kho ảnh Google Streetview. Các ảnh này sau đó được cắt ra thành 41.2 triệu ảnh. Mỗi ảnh cắt ra đều được gắn nhãn 6DoF (bao gồm cả GPS). Dữ liệu được thu thập từ năm 2009 đến năm 2021, dẫn đến việc tập dữ liệu bao phủ nhiều điều kiện ngoại cảnh như thời tiết, ánh sáng cũng như sự thay đổi của công trình kiến trúc trong khu vực.



Hình 2.26: Minh họa tập dữ liệu San Francisco Extra Large

# Chương 3

## PHƯƠNG PHÁP ĐỀ XUẤT

### 3.1 Tổng quan về mô hình

#### 3.1.1 Cơ chế truy xuất ảnh

Các bước thực hiện:

- Trích xuất bản đồ đặc trưng từ những lớp bên trong mô hình nền tảng
- Ép ngang các bản đồ đặc trưng thành vector 1 chiều
- Các vector 1 chiều sẽ được đưa vào các lớp pha trộn đặc trưng ảnh
- Các vector 1 chiều sẽ được ghép lại thành một vector 2 chiều
- Vector 2 chiều này sẽ được chạy qua những lớp Perceptron nhiều lớp để thu gọn, tổng hợp lại thành các đặc trưng toàn cục, thông qua chuyển vị ma trận

#### 3.1.2 Cơ chế hồi quy vị trí tương đối

### 3.2 Tiêu chí đánh giá

# **Chương 4**

## **ĐO ĐẠC VÀ ĐÁNH GIÁ**

- 4.1 Mô hình MixVPR**
- 4.2 Mô hình Map-free Relocalization**
- 4.3 Hướng phát triển**
- 4.4 Kết luận**

# **Chương 5**

## **Kế hoạch tương lai**

- 5.1 Thành quả đạt được**
- 5.2 Kế hoạch luận văn tốt nghiệp**

# Tài liệu tham khảo

- [1] Cnn | introduction to pooling layer. <https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/>. Accessed: 2023-11-26.
- [2] Laser slam. [http://wavelab.uwaterloo.ca/indexe424.html?weblizar\\_portfolio=laser-slam/](http://wavelab.uwaterloo.ca/indexe424.html?weblizar_portfolio=laser-slam/). Accessed: 2023-11-26.
- [3] Ajith Abraham. Artificial neural networks. *Handbook of measuring system design*, 2005.
- [4] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, November 2022.
- [5] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Mixvpr: Feature mixing for visual place recognition, 2023.
- [6] Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition, 2016.
- [7] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Áron Monszpart, Victor Adrian Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image, 2022.
- [8] Vassileios Balntas, Shuda Li, and Victor Prisacariu. Relocnet: Continuous metric learning relocalisation using neural nets. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 782–799, Cham, 2018. Springer International Publishing.
- [9] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006. Proceedings, Part I* 9, pages 404–417. Springer, 2006.
- [10] Gabriele Bertone, Carlo Masone, and Barbara Caputo. Rethinking visual geolocation for large-scale applications, 2022.
- [11] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization, 2018.
- [12] Mai Bui, Christoph Baur, Nassir Navab, Slobodan Ilic, and Shadi Albarqouni. Adversarial networks for camera pose regression and refinement, 2019.
- [13] Mingpeng Cai, Chunhua Shen, and Ian D. Reid. A hybrid probabilistic model for camera relocalization. In *British Machine Vision Conference*, 2018.
- [14] Boris Chidlovskii and Assem Sadek. Adversarial transfer of pose estimation regression, 2020.
- [15] Yongju Cho, Muhammad Faisal, Usama Sadiq, Tabasher Arif, Rehan Hafiz, Jeongil Seo, and Mohsen Ali. Learning to detect local features using information change. *IEEE Access*, 9:43898–43908, 2021.

- [16] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization, 2017.
- [17] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera re-localization. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2871–2880, 2019.
- [18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.
- [19] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2023.
- [20] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3304–3311. IEEE, 2010.
- [21] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE transactions on pattern analysis and machine intelligence*, 34(9):1704–1716, 2011.
- [22] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *arXiv preprint arXiv:2308.00688*, 2023.
- [23] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization, 2016.
- [24] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning, 2017.
- [25] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization, 2016.
- [26] Andrej Krenker, Janez Bešter, and Andrej Kos. Introduction to the artificial neural networks. *Artificial Neural Networks: Methodological Advances and Biomedical Applications*. InTech, pages 1–18, 2011.
- [27] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera relocalization by computing pairwise relative poses using convolutional neural network, 2017.
- [28] Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346, 2020.
- [29] Yimin Lin, Zhaoxiang Liu, Jianfeng Huang, Chaopeng Wang, Guoguang Du, Jinqiang Bai, Shigu Lian, and Bill Huang. Deep global-relative networks for end-to-end 6-dof visual localization and odometry, 2019.
- [30] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- [31] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.
- [32] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Image-based localization using hourglass networks, 2017.
- [33] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Relative camera pose estimation using convolutional neural networks, 2017.

- [34] Tayyab Naseer and Wolfram Burgard. Deep regression for monocular camera-based 6-dof global localization in outdoor environments. pages 1525–1530, 09 2017.
- [35] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.
- [36] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.
- [37] Noé Pion, Martin Humenberger, Gabriela Csurka, Yohann Cabon, and Torsten Sattler. Benchmarking image retrieval for visual localization, 2020.
- [38] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1655–1668, 2018.
- [39] Noha Radwan, Abhinav Valada, and Wolfram Burgard. Vlocnet++: Deep multi-task learning for semantic visual localization and odometry. *IEEE Robotics and Automation Letters*, 3(4):4407–4414, October 2018.
- [40] Soham Saha, Girish Varma, and C. V. Jawahar. Improved visual relocalization by discovering anchor points, 2018.
- [41] Torsten Sattler, Tobias Weyand, B. Leibe, and Leif P. Kobbelt. Image retrieval for image-based localization revisited. In *British Machine Vision Conference*, 2012.
- [42] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of cnn-based absolute camera pose regression, 2019.
- [43] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers, 2021.
- [44] Yoli Shavit, Ron Ferens, and Yosi Keller. Coarse-to-fine multi-scene pose regression with transformers. *IEEE transactions on pattern analysis and machine intelligence*, PP, 08 2023.
- [45] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, 2013.
- [46] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.
- [47] Niko Sünderhauf, Sareh Shirazi, Feras Dayoub, Ben Upcroft, and Michael Milford. On the performance of convnet features for place recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 4297–4304. IEEE, 2015.
- [48] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.
- [49] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular object retrieval with integral max-pooling of cnn activations. *arXiv preprint arXiv:1511.05879*, 2015.
- [50] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision, 2021.

- [51] Akihiko Torii, Josef Sivic, Tomá Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 883–890, 2013.
- [52] Abhinav Valada, Noha Radwan, and Wolfram Burgard. Deep auxiliary learning for visual localization and odometry, 2018.
- [53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.
- [54] Florian Walch, Caner Hazirbas, Laura Leal-Taixé, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using lstms for structured feature correlation, 2017.
- [55] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham. Atloc: Attention guided camera localization, 2019.
- [56] Ruotong Wang, Yanqing Shen, Weiliang Zuo, Sanping Zhou, and Nanning Zheng. Transvpr: Transformer-based place recognition with multi-level attention aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13648–13657, 2022.
- [57] Jian Wu, Liwei Ma, and Xiaolin Hu. Delving deeper into convolutional neural networks for camera relocalization. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5644–5651, 2017.
- [58] Meng Xu, Youchen Wang, Bin Xu, Jun Zhang, Jian Ren, Stefan Poslad, and Pengfei Xu. A Critical Analysis of Image-based Camera Pose Estimation Techniques. *arXiv e-prints*, page arXiv:2201.05816, January 2022.
- [59] Fei Xue, Xin Wang, Zike Yan, Qiuyuan Wang, Junqiu Wang, and Hongbin Zha. Local supports global: Deep camera relocalization with sequence enhancement, 2019.
- [60] Peng Yin, Shiqi Zhao, Ivan Cisneros, Abulikemu Abuduweili, Guoquan Huang, Micheal Milford, Changliu Liu, Howie Choset, and Sebastian Scherer. General place recognition survey: Towards the real-world autonomy age, 2022.
- [61] Jun Yu, Chaoyang Zhu, Jian Zhang, Qingming Huang, and Dacheng Tao. Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition. *IEEE transactions on neural networks and learning systems*, 31(2):661–674, 2019.
- [62] Amir Roshan Zamir and Mubarak Shah. Accurate image localization based on google maps street view. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part IV 11*, pages 255–268. Springer, 2010.
- [63] Qunjie Zhou, Torsten Sattler, Marc Pollefeys, and Laura Leal-Taixe. To learn or not to learn: Visual localization from essential matrices, 2020.

# Phụ lục A

## Kiến thức nền tảng

### A.1 Công nghệ đã được sử dụng

**PyTorch**

**Jupyter Notebook và Google Colab**

### A.2 Những mô hình học sâu nền tảng

#### A.2.1 Mạng thần kinh nhân tạo - Artificial Neural Network

Mạng thần kinh nhân tạo là một hướng đi trong lĩnh vực học máy, với nguồn cảm hứng được lấy từ tính kết nối của các nơ-ron thần kinh trong bộ não sinh vật.

Đơn vị nhỏ nhất cấu tạo nên một mạng thần kinh nhân tạo sẽ là một nơ-ron nhân tạo, hay còn được biết đến là một Perceptron. Các Perceptron sẽ được sắp xếp thành từng lớp, với luồng dữ liệu đi từ những nơ-ron ở lớp trước sang lớp sau. Hai nơ-ron được nối với nhau có thể truyền tải dữ liệu cho nhau, mô phỏng lại việc truyền tín hiệu giữa các synap thần kinh.

Mạng thần kinh nhân tạo phụ thuộc vào dữ liệu đã được xử lý sẵn để tự huấn luyện, cải thiện độ chính xác của bản thân. Quá trình tự huấn luyện sẽ giúp mạng phát hiện được những quy luật bên trong dữ liệu mà không cần sự can thiệp của con người. Vì vậy nên, những mô hình này có thể phát hiện ra được những quy luật chính xác hơn và hiệu quả hơn so với những quy luật được định nghĩa thủ công bởi chuyên gia.

##### Perceptron

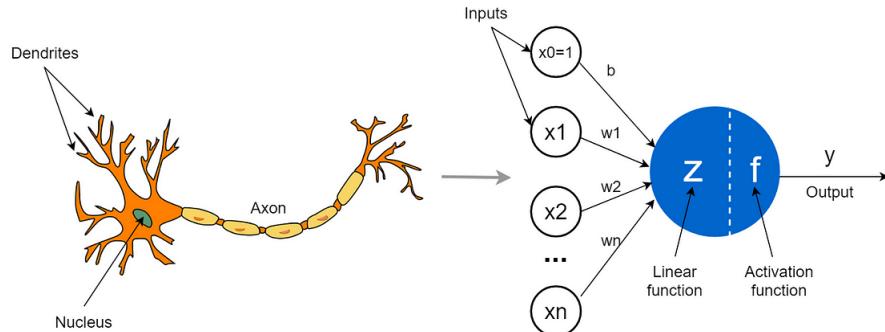
Perceptron là một thuật toán giúp phân loại dữ liệu theo hai lớp. Perceptron sẽ nhận vào một số lượng bất kỳ dữ liệu, áp dụng các trọng số được gán với mỗi dữ liệu đầu vào và một giá trị dời để tạo thành một hàm tuyến tính. Sau đó, kết quả của hàm tuyến tính sẽ được đi qua một lớp kích hoạt, đưa giá trị này về một trong hai phân loại đã được định nghĩa bằng đầu dựa vào một giới hạn đã được định trước (thường là giá trị 0).

Phương thức hoạt động của một Perceptron có thể được thể hiện qua công thức

$$f(X) = \Theta(w \cdot X + b)$$

$$\Theta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

Tuy nhiên, một vấn đề của mô hình Perceptron là mô hình sẽ chỉ xử lý tốt với những dạng dữ liệu có thể phân tách được ở dạng tuyến tính và chỉ hoạt động được với hai lớp phân loại do hàm kích hoạt. Vì vậy nên, để có thể mô hình hóa được những dạng dữ liệu có cấu trúc phức tạp hơn, nhiều mô hình Perceptron sẽ được kết hợp lại với nhau để tạo thành mạng Perceptron nhiều lớp.

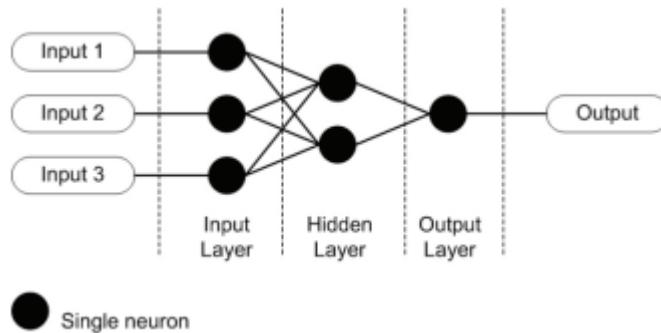


Hình A.1: Synap thần kinh là đơn vị cơ bản của hệ thống thần kinh và là nguồn cảm hứng để ứng dụng Perceptron vào lĩnh vực học sâu [3]

### Mạng Perceptron nhiều lớp

Đây là một cấu trúc cơ bản, một ví dụ điển hình của mạng thần kinh nhân tạo với đơn vị nhỏ nhất là các Perceptron. Mạng này sẽ được cấu tạo từ nhiều lớp khác nhau, mỗi lớp sẽ bao gồm các Perceptron có đầu vào là những Perceptron ở lớp trước và đầu ra sẽ được nối vào Perceptron ở lớp sau.

Tín hiệu, là những số thực, sẽ được truyền từ Perceptron ở lớp trước sang Perceptron của những lớp sau qua các cạnh liên kết. Mỗi cạnh sẽ được gán một trọng số. Trọng số này sẽ quyết định mức độ quan trọng của tín hiệu đi qua cạnh liên kết đó. Trong quá trình xử lý, các tín hiệu qua mỗi cạnh sẽ được nhân với trọng số tương ứng và cộng với lại độ dời ở nút Perceptron. Sau đó, kết quả này sẽ được đi qua một lớp phi tuyến tính để có thể thích ứng với nhiều kiểu dữ liệu khác nhau, không như một Perceptron đơn giản.



Hình A.2: Cấu tạo của một mạng nơ-ron nhân tạo cơ bản [26]

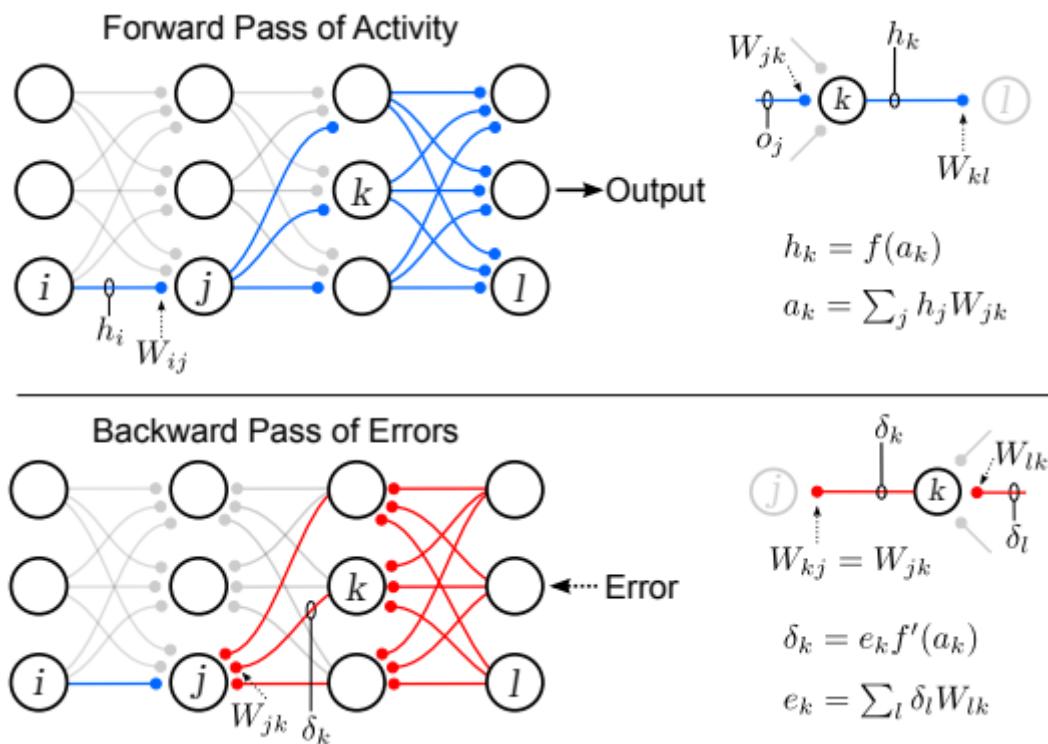
### Huấn luyện mạng nơ-ron

Quá trình học hỏi của một mạng nơ-ron nhân tạo cơ bản sẽ xuất phát từ việc xác định kết quả ban đầu từ dữ liệu đầu vào, tính ra sai số giữa kết quả của mô hình và

kết quả đúng được cung cấp. Sau đó, mô hình sẽ điều chỉnh lại các trọng số bên trong mô hình để giảm thiểu sai số đó. Qua mỗi lần điều chỉnh, kết quả đầu ra của mô hình sẽ càng trở nên giống với kết quả đúng hơn. Khi đạt đến một ngưỡng nhất định, việc huấn luyện sẽ được ngừng lại để ngăn việc kết quả của mô hình quá phụ thuộc vào tập dữ liệu.

Cơ chế điều chỉnh trọng số này được gọi là cơ chế truyền ngược. Cụ thể hơn, cơ chế này sẽ được thực hiện thông qua các bước

1. Tính toán sai số giữa kết quả đầu ra của mô hình và kết quả thực.
2. Đối với mỗi trọng số ở lớp cuối cùng, giá trị đạo hàm dựa trên sai số có thể tính được không quá khó khăn. Một bội âm của mỗi giá trị đạo hàm sẽ được dùng để điều chỉnh trọng số, nhằm hướng đến việc đạt đến sai số tối thiểu.
3. Bước 2 sẽ được lặp lại cho tất cả các lớp, bắt đầu từ lớp đầu tiên của mô hình. Giá trị đạo hàm tại mỗi lớp sẽ được tính dựa trên lớp ngay sau đó bằng quy tắc dây chuyền của Leibniz trong việc tính đạo hàm.



Hình A.3: Biểu đồ mô phỏng quá trình thực hiện cơ chế truyền ngược ở một mạng nơ-ron đơn giản [28]

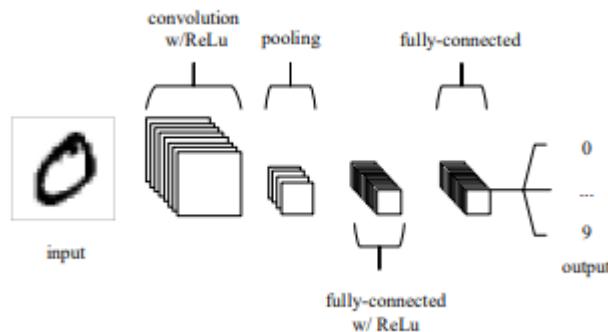
### A.2.2 Mạng Nơ-ron tích chập - Convolutional Neural Network

#### Định nghĩa

Mạng nơ-ron tích chập, hay còn được gọi là CNN hay ConvNet, là một loại mạng nơ-ron nhân tạo được thiết kế để xử lý những dạng dữ liệu có cấu trúc dạng bảng, như một hình ảnh. Trong máy tính, hình ảnh sẽ được lưu dưới dạng một bảng các điểm ảnh, với mỗi điểm ảnh chứa các dữ liệu nhằm thể hiện màu sắc và độ sáng của điểm ảnh đó.

Mạng nơ-ron tích chập sẽ giúp tối ưu hóa quá trình huấn luyện mô hình trên một lượng lớn dữ liệu ở dạng bảng. Cụ thể hơn, để huấn luyện một lớp của mạng nơ-ron nhân tạo truyền thống, là một mạng kết nối dày đủ giữa các lớp, trên một ảnh có kích thước  $100 \times 100$  điểm ảnh, 10000 trọng số sẽ cần được sử dụng và điều chỉnh trong quá trình huấn luyện. Việc sử dụng một số lượng lớn trọng số sẽ gây ra các vấn đề về tài nguyên tính toán, cũng như việc áp dụng cơ chế truyền ngược - làm cho giá trị đạo hàm bị tăng không kiểm soát hoặc biến mất.

Tuy nhiên, nhờ vào việc áp dụng các bộ lọc trong những lớp tích chập, số lượng trọng số cần được xử lý sẽ giảm đi một cách đáng kể. Một ảnh có độ phân giải  $100 \times 100$  vẫn có thể được giải quyết bằng một bộ lọc  $5 \times 5$ , với 25 trọng số. Kết quả đầu ra vẫn sẽ giữ cấu trúc dạng bảng, và những bảng giá trị ở những lớp càng nằm ở phía sau sẽ càng chứa nhiều thông tin từ bức ảnh, mà không tiêu tốn quá nhiều tài nguyên tính toán, cũng như né được những vấn đề liên quan đến cơ chế truyền ngược bằng đạo hàm.



Hình A.4: Một mạng nơ-ron tích chập cơ bản [35]

Mạng nơ-ron tích chập được lấy cảm hứng từ cách mà bộ não con người xử lý hình ảnh thu nhận được từ mắt. Mỗi nơ-ron sẽ hoạt động dưới một lớp tế bào cảm thụ ánh sáng riêng và các nơ-ron sẽ được nối với những nơ-ron ở những lớp khác để có thể bao phủ được toàn bộ vùng võng mạc ở mắt. Tương tự như cách mà mỗi phần của một bức ảnh sẽ được xử lý ở những vùng trên võng mạc mắt, các bộ lọc của mạng nơ-ron tích chập cũng sẽ chạy và xử lý từng mảng nhỏ của bức ảnh. Qua các lớp, những chi tiết được phát hiện sẽ tăng dần về độ phức tạp, từ những chi tiết đơn giản như cạnh, đường cong,... và tăng dần đến những chi tiết phức tạp như khuôn mặt, đồ vật,... Nói cách khác, mạng nơ-ron tích chập đã cho các mô hình học máy khả năng xử lý ảnh của mắt người.

### Lớp tích chập

Lớp tích chập là một lớp trích xuất đặc trưng ở dữ liệu đầu vào và bảo toàn mối quan hệ giữa các điểm ảnh. Lớp tích chập sẽ nhận vào một tập các điểm ảnh có dạng bảng, có cấu tạo là  $H \times W \times C$  với  $H$  là độ cao,  $W$  là độ rộng và  $C$  là số kênh dữ liệu có trong ảnh. Ban đầu,  $H$  và  $W$  sẽ phụ thuộc vào độ phân giải của ảnh và các ảnh thông thường sẽ gồm 3 kênh dữ liệu, đại diện cho 3 màu RGB.

Sau đó, ở bên trong lớp tích chập, một hay nhiều bộ lọc sẽ được trượt trên tất cả các mảng của ảnh. Tại mỗi mảng được một bộ lọc trượt qua, phép toán nhân ma trận Frobenius sẽ được sử dụng để tính ra một giá trị mã hóa cho chi tiết tại mảng đó. Công thức của phép toán nhân ma trận Frobenius được thể hiện như sau.

Với ma trận  $A$  và  $B$  được thể hiện như bên dưới,

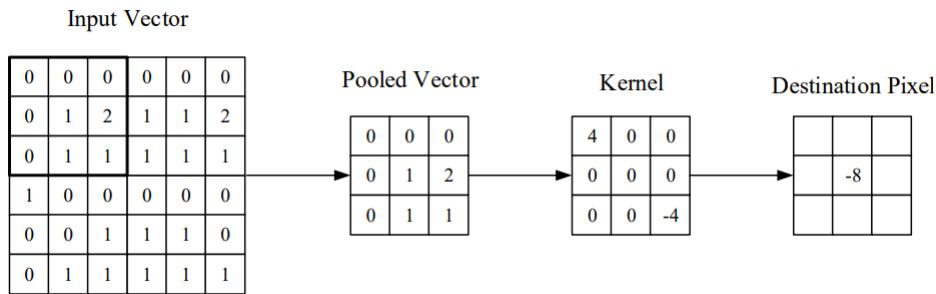
$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1m} \\ B_{21} & B_{22} & \cdots & B_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ B_{n1} & B_{n2} & \cdots & B_{nm} \end{pmatrix}$$

Giá trị của phép nhân ma trận Forbenius giữa  $A$  và  $B$  là

$$\langle \mathbf{A}, \mathbf{B} \rangle_F = \sum_{i,j} \overline{A_{ij}} B_{ij} = \text{Tr}(\overline{\mathbf{A}^T} \mathbf{B}) \equiv \text{Tr}(\mathbf{A}^\dagger \mathbf{B})$$

với ký hiệu gạch đầu là phép tính số phức liên hợp và ký hiệu  $\dagger$  thể hiện phép nghịch đảo ma trận và tính số phức liên hợp trên tất cả các phần tử. Tuy nhiên, do lối tích chập chỉ xử lý dữ liệu các số thực, cho nên công thức có thể được đơn giản hóa thành

$$\begin{aligned} \langle \mathbf{A}, \mathbf{B} \rangle_F = & A_{11}B_{11} + A_{12}B_{12} + \cdots + A_{1m}B_{1m} \\ & + A_{21}B_{21} + A_{22}B_{22} + \cdots + A_{2m}B_{2m} \\ & \vdots \\ & + A_{n1}B_{n1} + A_{n2}B_{n2} + \cdots + A_{nm}B_{nm} \end{aligned}$$



Hình A.5: Áp dụng một bộ lọc kích thước  $3 \times 3$  lên một mảng của ảnh [35]

Sau đó, kết quả sẽ được chạy qua một hàm kích hoạt, thông thường là ReLU để tạo sự phi tuyến tính.

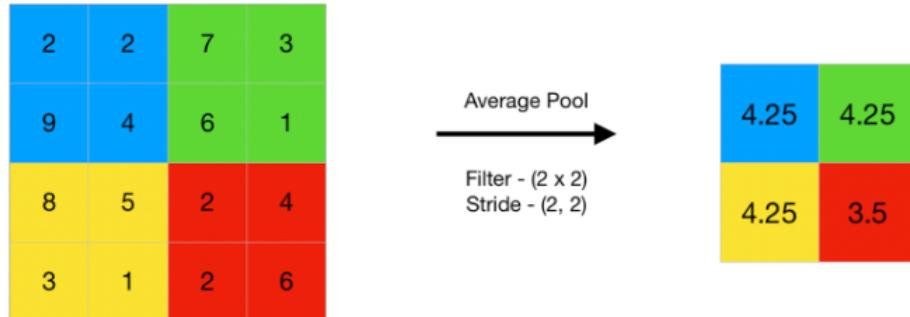
Một số tham số có thể điều chỉnh cho quá trình chạy lớp tích chập chính là kích thước của lớp lọc, bước nhảy, và kích thước phần đệm. Bước nhảy sẽ quy định độ dời về điểm ảnh sau mỗi lần xử lý từng mảng của bộ lọc. Kích thước phần đệm sẽ có tác dụng thay đổi kích thước của dữ liệu đầu vào bằng cách thêm các số 0 dọc theo đường viền của bảng dữ liệu, giúp điều chỉnh ảnh phù hợp với kích thước bộ lọc và bước nhảy. Cuối cùng, kích thước của bộ lọc sẽ quy định độ lớn của mảng được xử lý trong một lần chạy. Trước đây, các bộ lọc có kích thước lớn như  $7 \times 7$  sẽ được sử dụng để làm lớp tích chập đầu tiên, nhằm gom cụm lại các chi tiết trên ảnh hiệu quả hơn. Tuy nhiên, qua quá trình phát triển của mạng nơ-ron tích chập, việc sử dụng nhiều bộ lọc có kích thước  $3 \times 3$  liên tiếp đã được phát hiện là vẫn sẽ bao phủ được một vùng có kích thước lớn với số trọng số cần điều chỉnh ít hơn. Vì vậy nên, những bộ lọc có kích thước  $3 \times 3$  đã trở nên phổ biến trong việc thiết kế mạng nơ-ron tích chập [46].

### Lớp tổng hợp

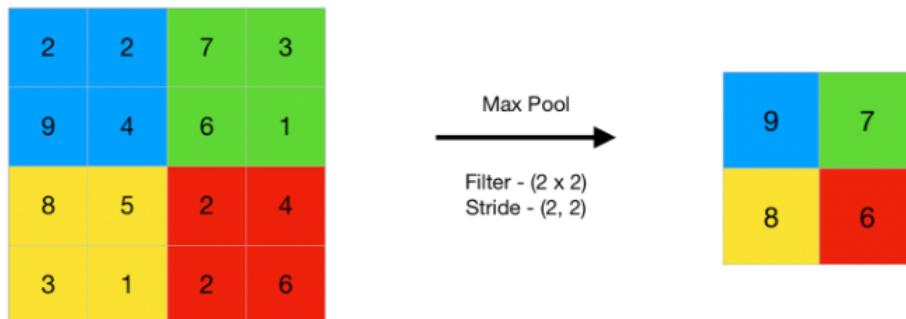
Lớp tổng hợp sẽ có tác dụng giúp làm giảm lượng dữ liệu trên bảng khi hình có kích thước quá lớn, mà vẫn giữ được những thông tin quan trọng. Các giá trị tại một

mảng sẽ được tổng hợp lại thành một giá trị đại diện sau khi qua lớp tổng hợp. Các phương pháp chính được áp dụng cho lớp tổng hợp bao gồm:

- Max pooling: từ mảng đang xét, giá trị lớn nhất sẽ được chọn để đại diện cho cả mảng.
- Average pooling: giá trị trung bình của mảng đang xét sẽ được chọn làm đại diện.



Hình A.6: Lớp tổng hợp sử dụng Average Pooling [1]

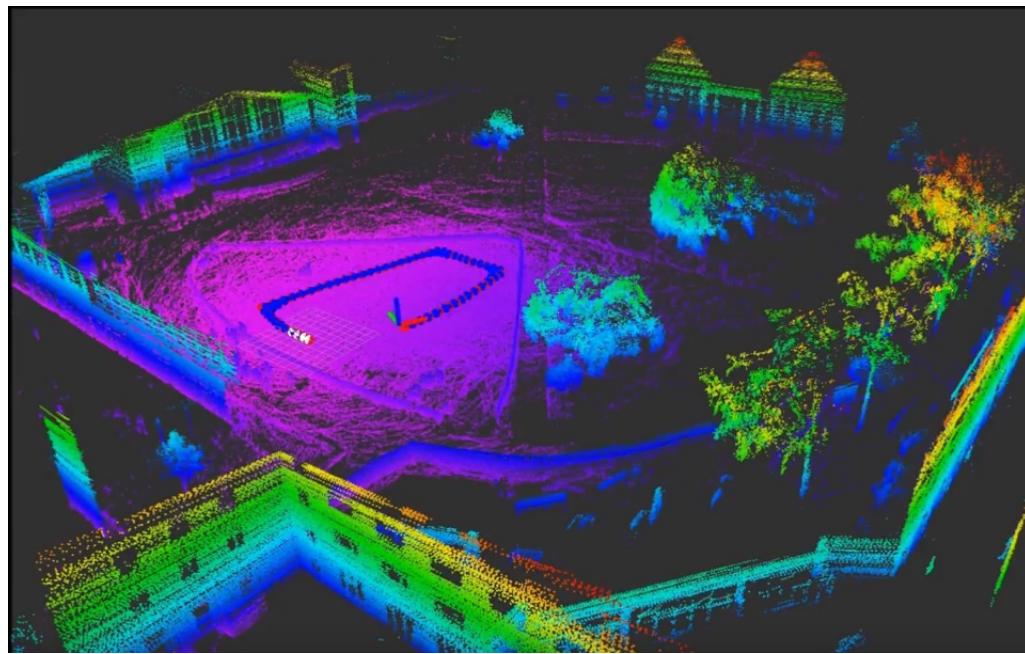


Hình A.7: Lớp tổng hợp sử dụng Max Pooling [1]

### A.3 Kiến thức về phương pháp nhận dạng địa điểm trực quan bằng MixVPR

Trong bài toán về nhận dạng địa điểm trực quan - Visual Place Recognition, đa số các phương pháp có thể được phân vào hai lớp chính, dựa vào vị trí hoặc dựa vào những điểm chung[60].

- Đối với các phương pháp dựa vào vị trí, một địa điểm đã được đi qua rồi sẽ có thể được nhận biết mà không cần phụ thuộc vào góc nhìn và những điều kiện môi trường xung quanh.
- Đối với những phương pháp nhận biết dựa vào điểm chung, thông tin về môi trường xung quanh sẽ được thu lại thông qua ảnh chụp góc nhìn. Từ đó, những phương pháp truy xuất ảnh sẽ được sử dụng nhằm tìm kiếm từ hệ cơ sở dữ liệu một ảnh có độ tương đồng cao.



Hình A.8: Hình minh họa cho phương pháp nhận diện địa điểm trực quan thông qua những địa điểm đã qua [2]



(a) Mobile phone query      (b) Retrieved image of same place

Hình A.9: Hình minh họa cho phương pháp nhận diện địa điểm trực quan thông qua việc truy xuất ảnh từ hệ cơ sở dữ liệu [6]

Mô hình Mix-VPR được chọn là một phương pháp nhận biết dựa vào điểm chung giữa ảnh truy vấn và các ảnh bên trong hệ cơ sở dữ liệu, đại diện cho một khu vực cần được biểu diễn. Mix-VPR so sánh các ảnh với nhau thông qua việc truy xuất đặc trưng của các ảnh và tìm cặp ảnh có đặc trưng mang điểm tương đồng cao nhất.

### A.3.1 Trích xuất đặc trưng

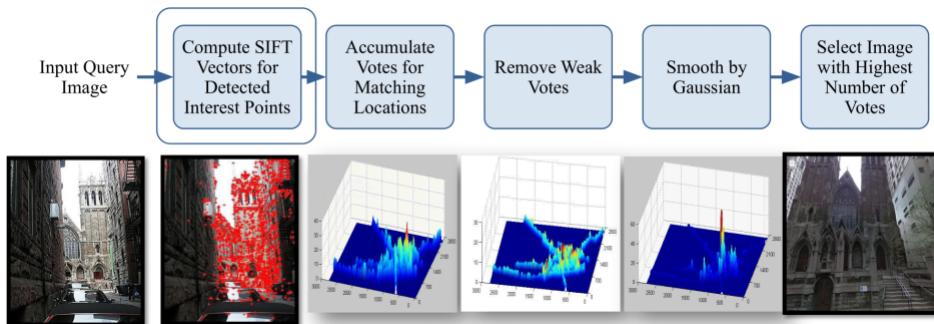
Để có thể tìm được những ảnh lân cận, có sẵn tọa độ GPS, những đặc trưng của ảnh sẽ được tạo ra thông qua một quá trình trích xuất và so sánh với lẫn nhau. Phương pháp này hoạt động dựa trên việc những ảnh có góc nhìn, vị trí gần nhau thì sẽ nhìn cùng một tập hợp các vật thể, dẫn đến việc ảnh sẽ có những chi tiết bị trùng lặp ở các

vật thể đó. Tuy nhiên, việc chỉ so sánh ảnh với nhau ở mức độ từng điểm ảnh sẽ tạo ra một lượng dữ liệu rất lớn và có thể không cần thiết.

Vì vậy nên việc sản sinh ra các đặc trưng cho hình ảnh sẽ giúp mã hóa ảnh một cách tối ưu mà không làm mất những thông tin quan trọng trong hình. Qua thời gian, việc trích xuất đặc trưng trong lĩnh vực nhận diện địa điểm trực quan(Visual Place Recognition) đã có những bước phát triển nhất định, từ việc chỉ trích xuất những đặc trưng cục bộ trong ảnh. Sau đó, một lớp tổng hợp các đặc trưng cục bộ sẽ được sử dụng [37]. Cuối cùng, với sự phát triển của các mô hình học sâu, mạng nơ-ron tích chập đã được huấn luyện để có thể tạo thành những đặc trưng toàn cục của ảnh từ bản đồ đặc trưng đạt được từ những lớp tích chập của mạng.

### Đặc trưng cục bộ

Quá trình trích xuất đặc trưng cục bộ sẽ bao gồm hai bước là xác định những yếu tố trọng tâm trong ảnh và xây dựng những đặc trưng xung quanh những yếu tố đó. Trong trường hợp lý tưởng, kết quả đầu ra sẽ là những đặc trưng cục bộ tương đồng dưới những điều kiện khác nhau như thay đổi độ sáng, tỷ lệ hình ảnh, độ nhiễu ảnh, cũng như những góc quay khác nhau [30].



Hình A.10: Quy trình của mô hình sử dụng SIFT để trích xuất đặc trưng cục bộ [62]

Trong những phương pháp sử dụng những phương pháp trích xuất truyền thống trước đây như SIFT [31] và SURF [9] những đặc trưng được định nghĩa một cách thủ công sẽ được lấy ra từ ảnh. Những đặc trưng này bao gồm đặc điểm bề mặt, đường nét, những điểm đặc trưng, không gian hình học đặc biệt. Khoảng cách giữa các hình trong không gian đặc trưng sẽ là tiêu chí đánh giá về độ tương đồng. Từ đó, một tập những ảnh có những đặc trưng cục bộ giống với ảnh truy vấn nhất sẽ được tìm thấy từ hệ cơ sở dữ liệu và có thể xác định vị trí chụp ảnh truy vấn từ đó [37]

### Tổng hợp đặc trưng cục bộ

Các đặc trưng cục bộ có thể được tổng hợp một lần nữa để có thể tạo ra một đoạn mã hóa mới cho ảnh. Việc này được thực hiện nhằm thu nhỏ kích thước mã hóa của ảnh, so với khi dùng đặc trưng cục bộ và giúp làm giảm sức ảnh hưởng từ những chi tiết của những vật không cố định bên trong ảnh, như xe hơi, cây cối, người đi bộ,... nhằm hướng đến việc sinh ra những cách mã hóa ảnh không bị thay đổi khi góc chụp, độ sáng hay bị che khuất bởi vật thể [20].

Một số phương pháp tổng hợp cổ điển đã được sử dụng bao gồm hướng tiếp cận bag-of-visual-words [36], VLAD [20], vector Fischer [21].

Với sự phát triển của công nghệ học sâu, các mạng nơ-ron đã được ứng dụng vào cho việc trích xuất các đặc trưng của ảnh và đã có sự cải thiện rõ rệt về hiệu quả [47] so với cách trích xuất truyền thống

## Ứng dụng của mạng nơ-ron

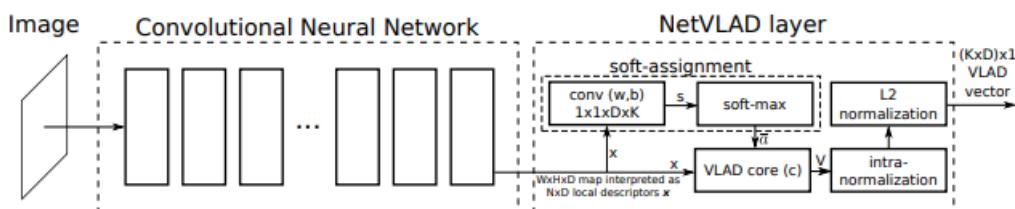
Với sự xuất hiện của mạng nơ-ron, đặc biệt là mạng nơ-ron tích chập, các tác vụ trích xuất và tổng hợp đặc trưng cục bộ của ảnh có thể được thực hiện mà không cần con người định nghĩa sẵn những quy luật nào.

Với việc trích xuất đặc trưng cục bộ, một bản đồ đặc trưng có thể được tạo ra từ việc chạy một ảnh qua một mô hình mạng nơ-ron đã được huấn luyện. Tiếp theo đó, việc tổng hợp các đặc trưng này có thể được thực hiện bởi những lớp tiếp theo của mạng nơ-ron tích chập, hoặc những phương pháp được truyền cảm hứng từ những phương pháp tổng hợp truyền thống. Do việc trích xuất và tổng hợp đặc trưng được thực hiện cùng trong một lần ảnh chạy qua mô hình, nên cơ chế này được gọi là detect-and-describe, khác với những phương pháp truyền thống là detect-then-describe [15]

Một mạng nơ-ron tích chập bình thường có thể được dùng cho mục đích trích xuất. Tuy nhiên, để có hiệu quả cao và tiết kiệm thời gian huấn luyện cho mô hình, các mô hình đã được huấn luyện sẵn sẽ được sử dụng như VGG, Res-Net, EfficientNet,... và bản đồ đặc trưng sẽ được lấy từ một lần truyền qua một mạng cơ sở đã được loại bỏ lớp cuối cùng.

Sau đó, ở bước tổng hợp, một số các lớp phương pháp đã xuất hiện như:

- Một số phương pháp sẽ lấy cảm hứng từ những cách truyền thống, nhưng được điều chỉnh để trở thành một mạng huấn luyện được, như NetVLAD [6], SPE-VLAD [61].
- Một lớp các phương pháp sẽ tập trung vào việc sử dụng một lớp tổng hợp - pooling layer để tổng hợp. Các phương pháp phổ biến bao gồm GeM [38], MAC, R-MAC [49]



Hình A.11: Mô hình mã hóa ảnh bằng đặc trưng được truy xuất từ CNN và NetVLAD [6]

Trong thời gian gần đây, với sự phát triển của mô hình Transformer [53], rất nhiều nghiên cứu đã hướng đến việc áp dụng mô hình này lên lĩnh vực trích xuất đặc trưng mã hóa ảnh cho bài toán truy xuất ảnh như AnyLoc [22], TransVPR [56]. Những hướng đi này đều đạt kết quả khả quan, tuy nhiên khả năng biểu diễn cho hình ảnh vẫn chưa thể vượt qua được NetVLAD [5].

Với những tiến bộ trong mạng thần kinh đãng hướng gần đây, cơ chế tự tập trung đã được chứng minh là không quá quan trọng cho những Vision Transformer [18]. Một minh chứng cho việc này chính là mô hình MLP-Mixer [50], có cấu trúc chỉ gồm những lớp Perceptron. Mô hình này đã có kết quả cạnh tranh trong những tác vụ cơ bản của lĩnh vực thị giác máy tính.

### A.3.2 MLP-Mixer

#### Định nghĩa

Mạng MLP-Mixer, khác hẳn với những cấu trúc phức tạp như xu hướng của lĩnh vực học sâu trong thời gian gần đây, được cấu tạo hoàn toàn từ những lớp Perceptron. Vì vậy nên, các thao tác được thực hiện trong một lần xử lý ảnh chỉ bao gồm tác vụ nhân ma trận, thêm yếu tố phi tuyến tính vào mô hình và thay đổi bộ cục của dữ liệu(nghịch đảo, thay đổi số chiều của ma trận).

Mạng MLP-Mixer sẽ nhận đầu vào là một ảnh. Ảnh này sau đó sẽ được cắt thành các mảnh không trùng lặp với nhau. Những mảnh này sẽ được tham chiếu lên một chiều không gian khác, gọi là không gian  $C$ , được thể hiện bằng một vector 1 chiều. Mỗi vector sẽ được gọi là một token và tập hợp các phần tử nằm ở cùng vị trí trên không gian  $C$  ở mỗi vector sẽ được gọi là một kênh dữ liệu.

Mạng MLP-Mixer hoạt động dựa trên 2 cơ chế chính là pha trộn tokens và pha trộn theo kênh dữ liệu.

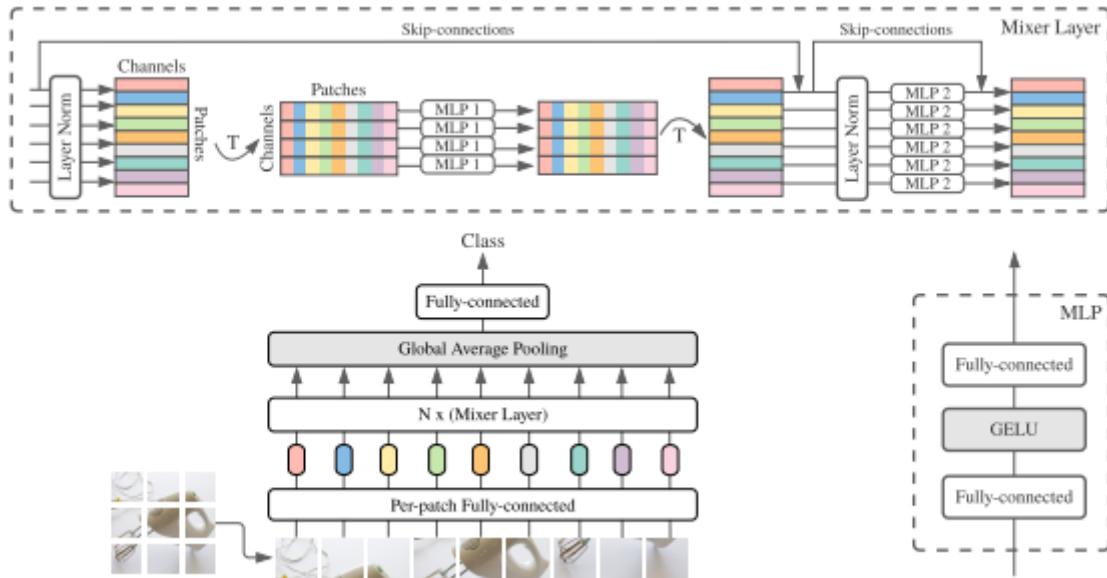
- Pha trộn tokens - Tokens mixing sẽ giúp cho dữ liệu trên cùng một kênh có thể biết và tác động đến nhau. Cơ chế pha trộn tokens sẽ được thực hiện trên từng kênh dữ liệu một.
- Pha trộn theo kênh dữ liệu - Channel mixing giúp cho các dữ liệu nằm trên cùng một token có thể tác động lẫn nhau. Cơ chế pha trộn theo kênh sẽ được thực hiện trên từng token một.

#### Thực hiện

Mạng Mixer sẽ nhận đầu vào là một ảnh đã được phân thành  $S$  mảnh không trùng nhau. Từng mảnh sẽ có dạng dữ liệu là ma trận 2D sẽ được tham chiếu đến một chiều không gian  $R^C$  và chuyển thành một vector 1 chiều có  $C$  phần tử. Ma trận được dùng để tham chiếu sẽ được dùng chung cho quá trình chuyển đổi. Thông số  $S$ , số lượng những mảnh có kích thước  $(P, P)$  của một ảnh có độ phân giải  $(H, W)$  sẽ được tính bằng công thức:

$$S = \frac{H * W}{P^2}$$

Sau đó các vector đại diện cho từng mảnh sẽ được nối lại với nhau, tạo thành một ma trận 2 chiều có dạng  $R^{S \times C}$ . Với các hàng là những token(mã hóa của từng mảnh được cắt từ ảnh) và các cột là những kênh dữ liệu trên không gian  $C$ . Ma trận 2 chiều này sẽ được ký hiệu là  $X$ .



Hình A.12: Cấu trúc của mô hình MLP-Mixer [50]

Ma trận này sau đó sẽ được đưa qua những lớp Mixer, lần lượt pha trộn thông tin giữa các token và giữa các kênh dữ liệu.

- Pha trộn tokens - Token mixing: Mạng MLP dành cho token-mixing sẽ được sử dụng để chiếu  $R^S \rightarrow R^S$ . Đối tượng thực hiện sẽ là các cột của  $X$ , là thông tin của một kênh dữ liệu trên những token.
- Pha trộn kênh dữ liệu - Channel mixing: Mạng MLP dành cho channel-mixing sẽ được sử dụng trên các hàng của  $X$ , gồm giá trị của các kênh trên từng token một. Mạng MLP này sẽ tham chiếu  $R^C \rightarrow R^C$

Mỗi lớp MLP được sử dụng bên trong mô hình sẽ được cấu tạo từ 2 lớp mạng kết nối đầy đủ. Giữa lớp mạng đầu tiên và lớp mạng thứ hai, một lớp hàm kích hoạt sẽ được dùng để tạo sự phi tuyến tính cho quá trình. Ngoài ra, đầu vào của các mạng MLP đều sẽ được chạy qua một lớp chuẩn hóa và mỗi lần chạy qua một lớp MLP đều sẽ được kết nối tắt với giá trị ban đầu. Các công thức bên dưới sẽ tóm tắt được quá trình xử lý của một lớp mạng Mixer

$$U_*, i = X_*, i + W_2 * \sigma(W_1 * \text{LayerNorm}(X)_*, i), \\ Y_j, * = U_j, * + W_4 * \sigma(W_3 * \text{LayerNorm}(U)_j, *)$$

với  $i = [1\dots C]$  và  $j = [1\dots S]$ .  $\sigma$  là hàm không tuyến tính, trong bài báo này sẽ là hàm GELU [19].

Cuối cùng, dữ liệu sẽ được đưa vào một lớp tổng hợp trung bình toàn cục và qua một mạng kết nối đầy đủ trước khi trả về kết quả là một đoạn mã hóa cho ảnh đầu vào.

### Đánh giá với các mô hình học sâu trước đây

Về độ phức tạp khi tính toán, do các hàm pha trộn tokens và pha trộn kênh dữ liệu đều là những mô hình MLP vậy nên độ phức tạp của quá trình tính toán sẽ phát triển tuyến tính với số token và độ phân giải của hình. Cụ thể hơn, do độ rộng của các lớp bên trong mô hình MLP của mạng pha trộn tokens được định nghĩa không phụ thuộc vào số lượng các mảnh đầu vào,  $S$ , nên độ phức tạp tính toán sẽ phát triển tuyến tính với số lượng mảnh, khác với những mô hình Vision Transformer với độ phức tạp bậc

hai. Độ rộng của các mô hình MLP pha trộn theo kênh cũng được chọn không phụ thuộc vào số lượng channel trong một mảnh,  $C$ , vậy nên độ phức tạp cũng sẽ tăng tuyến tính theo độ phân giải của ảnh, giống như một mạng nơ-ron tích chập.

Một mô hình MLP dùng cho pha trộn các tokens sẽ được áp dụng cho mọi kênh dữ liệu và một mô hình MLP dùng để pha trộn các kênh trong một tokens cũng sẽ được dùng chung cho mọi tokens. Việc dùng chung một mạng MLP cho các kênh trong một tokens là một quyết định bình thường, nhằm giữ lại sự bất biến về vị trí trong ảnh, một yếu tố quan trọng trong mạng nơ-ron tích chập. Tuy nhiên, việc sử dụng chung một mạng MLP nhằm pha trộn tokens cho các kênh dữ liệu thì ít phổ biến hơn. Tuy nhiên, việc sử dụng chung các thông số sẽ giúp giảm tốc độ tăng trưởng và tiết kiệm bộ nhớ của mô hình khi số mảnh  $S$  hoặc số chiều của không gian  $C$  tăng lên.

Mỗi lớp Mixer sẽ nhận đầu vào có kích thước không đổi. Kiến trúc này có những nét tương đồng với mạng Transformer hoặc những mô hình RNN trong những lĩnh vực khác. Đây là điểm khác biệt lớn với những mạng nơ-ron tích chập khác, khi mà độ phân giải của đầu vào ở những lớp sâu hơn sẽ thấp, nhưng số kênh dữ liệu tăng lên.

## A.4 Kiến thức phương pháp ước tính vị trí của máy ảnh bằng ma trận thiết yếu

### A.4.1 Máy ảnh lỗ kim - Pinhole Camera

Máy ảnh đo cường độ ánh sáng được chiếu đến nó bằng một con chíp. Bề mặt của chíp sẽ gồm nhiều khu vực và mỗi khu vực sẽ đo cường độ ánh sáng bằng cách đếm số photon đến được bề mặt của chíp. Lens của máy ảnh sẽ điều hướng ánh sáng đến được máy ảnh lên chíp và ảnh sẽ được tạo ra dựa trên lượng ánh sáng được điều đến các khu vực trên máy ảnh. Vậy nên, máy ảnh chỉ đo được lượng ánh sáng tới từ một hướng của một khu vực tới máy ảnh.

### A.4.2 Ma trận thiết yếu - Essential Matrix

### A.4.3 Tìm sự tương ứng giữa đặc trưng ảnh - Feature Matching

### A.4.4 Giải thuật 5 điểm ảnh - 5-Point Solver

### A.4.5 Thuật toán tính độ sâu ảnh qua một ảnh - Monocular Depth Estimation

## A.5 Sự liên kết giữa mô hình truy xuất ảnh và mô hình hồi quy tương đối

Mặc dù có nhiều cách để thực hiện việc hồi quy ra vị trí ảnh:

- Hồi quy dựa trên bản đồ đám mây điểm 3D toàn cục
- Hồi quy dựa trên bản đồ đám mây điểm 3D cục bộ
- Hồi quy dựa trên vị trí của các ảnh được truy xuất

Đa số các mô hình đều sử dụng cùng một cách biểu diễn khu vực, dựa trên những đặc trưng đã được thu gọn trên hình. Những đặc trưng này thường sẽ được huấn luyện cho

những bài toán truy xuất địa danh, nhận diện khu vực. Vậy nên các đặc trưng này sẽ có những giá trị giống nhau khi các ảnh cùng nhìn một tòa nhà, một địa danh, kể cả dưới những góc nhìn khác nhau. Vậy nên, có thể kết quả của một mô hình thực hiện cả 2 tác vụ rời rạc này sẽ không đạt được kết quả tối ưu.[37]

**Lưu ý:** Liệt kê ra những yêu cầu về đặc trưng cho những phương pháp khác nhau ở trên, cũng trong bài [37]

## A.6 Tìm hiểu giới hạn của mô hình hồi quy vị trí tuyệt đối dựa trên mạng Nơ-ron tích chập

Các mô hình hồi quy vị trí tuyệt đối sẽ xây dựng cách biểu diễn của khu vực trong tập dữ liệu một cách bao hàm. Thông qua đó, một số địa điểm bên trong khu vực được biểu diễn bởi tập dữ liệu sẽ được ngầm chọn làm điểm mốc. Điều này dẫn đến việc khi được huấn luyện trên những tập dữ liệu có hạn chế về mặt không gian thì mô hình hồi quy vị trí tuyệt đối sẽ không thể nào tổng quát hóa ra những khu vực không có trong tập dữ liệu ban đầu được mà phải huấn luyện lại trên tập dữ liệu của khu vực đó.[42]

Điều tương tự cũng có thể xảy ra với hồi quy vị trí tương đối, khi mà những ảnh truy xuất được có vị trí không đủ tổng quát thì có thể ảnh hưởng xấu đến kết quả của quá trình hồi quy vị trí tương đối

**Lưu ý:** Cần ghi ra phương pháp, cấu trúc mô hình được sử dụng trong bài. Cấu trúc khá đơn giản nên có thể không tổng quát hóa được. Cần ghi thêm các thí nghiệm của người ta.

## A.7 Tìm hiểu giới hạn của một số tập dữ liệu

Một số tập dữ liệu quá thưa, không phản ánh được chính xác vùng mà tập dữ liệu muốn miêu tả, mà nó dày quá thì mô hình chạy không nổi[10]