

# MM-Path: Multi-modal, Multi-granularity Path Representation Learning—Extended Version

Ronghui Xu<sup>1</sup>, Hanyin Cheng<sup>1</sup>, Chenjuan Guo<sup>1</sup>, Hongfan Gao<sup>1</sup>, Jilin Hu<sup>1</sup>, Sean Bin Yang<sup>2</sup>, Bin Yang<sup>1\*</sup>

<sup>1</sup>East China Normal University, <sup>2</sup>Chongqing University of Posts and Telecommunications  
{rhxu,hycheng,hf.gao}@stu.ecnu.edu.cn, {cjguo,jlhu,byang}@dase.ecnu.edu.cn, sean.byang@gmail.com

## Abstract

Developing effective path representations has become increasingly essential across various fields within intelligent transportation. Although pre-trained path representation learning models have shown improved performance, they predominantly focus on the topological structures from single modality data, i.e., road networks, overlooking the geometric and contextual features associated with path-related images, e.g., remote sensing images. Similar to human understanding, integrating information from multiple modalities can provide a more comprehensive view, enhancing both representation accuracy and generalization. However, variations in information granularity impede the semantic alignment of road network-based paths (road paths) and image-based paths (image paths), while the heterogeneity of multi-modal data poses substantial challenges for effective fusion and utilization. In this paper, we propose a novel Multi-modal, Multi-granularity Path Representation Learning Framework (MM-Path), which can learn a generic path representation by integrating modalities from both road paths and image paths. To enhance the alignment of multi-modal data, we develop a multi-granularity alignment strategy that systematically associates nodes, road sub-paths, and road paths with their corresponding image patches, ensuring the synchronization of both detailed local information and broader global contexts. To address the heterogeneity of multi-modal data effectively, we introduce a graph-based cross-modal residual fusion component designed to comprehensively fuse information across different modalities and granularities. Finally, we conduct extensive experiments on two large-scale real-world datasets under two downstream tasks, validating the effectiveness of the proposed MM-Path. This is an extended version of the paper accepted by KDD 2025. The code is available at: <https://github.com/decisionintelligence/MM-Path>.

## CCS Concepts

• Computing methodologies → Machine learning.

\*Corresponding authors

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-XXXX-X/18/06

<https://doi.org/XXXXXXX.XXXXXXX>

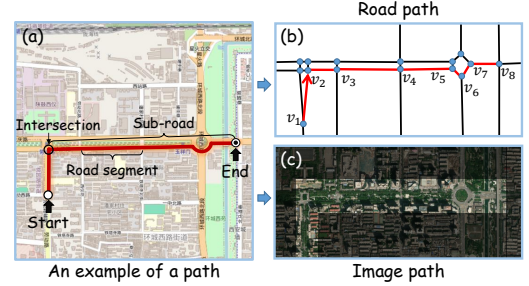


Figure 1: A path in different modalities

## Keywords

Path representation learning, Multi-modal learning, Self-supervised learning

## ACM Reference Format:

Ronghui Xu, Hanyin Cheng, Chenjuan Guo, Hongfan Gao, Jilin Hu, Sean Bin Yang, Bin Yang. 2018. MM-Path: Multi-modal, Multi-granularity Path Representation Learning—Extended Version. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 12 pages. <https://doi.org/XXXXXXX.XXXXXXX>

## 1 Introduction

Understanding paths and developing effective path representations are increasingly essential, offering invaluable insights for diverse fields such as intelligent navigation [18, 19, 34, 35, 43], route recommendation [7, 12, 49], urban planning [11, 17, 42], and urban emergency management [14]. Recent studies focus on developing pre-trained path representation learning models, which have demonstrated outstanding generalization capabilities [5, 21, 45]. These models efficiently produce generic path representations in an unsupervised manner. With simple fine-tuning and little labeled data, they are adaptable to diverse downstream tasks such as travel time estimation and path ranking score estimation. Consequently, they significantly improve computational efficiency by reducing both labeled data and runtime.

Paths have different modalities that provide richer, more diverse information. For example, while paths derived from road networks (road paths for short) elucidate topological relationships among road segments in paths, remote sensing images of paths (image paths for short) provide insights into geometric features and broader environmental contexts (see Figure 1). Integrating these modalities enriches path representations with varied perspectives, thereby improving accuracy and enhancing generalization capabilities. However, current path representation learning models primarily rely on single-modality data from road networks, which fails to capture the deep,

comprehensive context essential for a complete understanding of paths. This calls for developing a multi-modal pre-trained path representation learning model. Nonetheless, constructing such a model faces several challenges:

**Information granularity discrepancies between road paths and image paths significantly hinder cross-modal semantic alignment.** Effective cross-modal alignment, which ensures semantic consistency and complementarity among various modalities, is crucial for constructing multi-modal models [26]. However, the discrepancies in information granularity between road paths and image paths are substantial. As depicted in Figure 1, road paths typically focus on detailed topological structures and delineate road connectivity, while the image paths capture global environmental contexts on a large scale, reflecting the functional attributes of corresponding regions. It is worth noting that images may include extensive regions that show low relevance to the road paths, such as the dark regions in Figure 1 (c). Current image-text multi-modal models [2, 30, 37, 40] typically align individual images with textual sequences. However, such single-granularity and coarse alignment methods introduce noise, which are not suitable for the precise alignment required for paths. Additionally, as shown in Figure 1 (a), roads have different granularities in nature, including intersections, road segments, and sub-roads. Fully understanding paths at different granularities can provide insights from micro to macro levels, mitigating the negative effects caused by the differences in information granularity across modalities. Although some studies [9, 32] have explored multi-granularity in single-modal data, they have not adequately addressed the requirements for multi-granularity analysis in multi-modal contexts. **Thus, it is crucial to refine multi-granularity data processing and develop multi-granularity methods for cross-modal alignment.**

**The inherent heterogeneity of road paths and image paths poses a significant challenge during feature fusion.** The differences in data structure and information granularity between road paths and image paths extend to their learning methods. Road path representation learning typically focuses on connectivity and reachability between roads and intersections, as well as analyzing graph structures [3, 21, 44, 45]. Conversely, image learning methods that are able to learn image paths prioritize object recognition and feature extraction, aiming for a broad understanding of image content [1, 20]. These disparate learning methods lead to road paths and image paths mapped to different embedding spaces, resulting in feature dimensions with similar semantics containing entirely different information. Simple fusion methods like early fusion (i.e., integrating multiple modalities before or during the feature extraction stage) and late fusion (i.e., keeping each modality independently processed until the final fusion stage) may result in information loss and increased bias, and fail to capture subtle correlations between road paths and image paths [26, 47]. Therefore, a multi-modal fusion method that can capture the relationships among entities in different modalities and ensuring effective data fusion, is critically needed.

To address these challenges, we propose a *Multi-modal, Multi-granularity Path Representation Learning Framework*, namely MM-Path, for learning generic path representations.

**To address the first challenge,** we propose a multi-granularity alignment component. This component systematically associates

intersections, road sub-paths, and entire road paths with their corresponding image information to capture details accurately at a finer granularity as well as maintaining global correspondence at a coarser granularity. Specifically, we divide the image of the entire interested region into small fixed-size images, collect the small fixed-size images along each path, and arrange the collected images into an image path (i.e., image sequence). We employ modal-specific tokenizers to generate the initial embeddings for road paths and image paths, respectively. **Subsequently, these initial embeddings are fed into the powerful Transformer architecture to learn complex encoded embeddings for each modality at three granularities. Finally, a multi-granularity alignment loss function is employed to ensure the alignment of road and image encoded embeddings across different granularities.**

**To address the second challenge,** we introduce a graph-based cross-modal residual fusion component, which is designed to effectively fuse cross-modal features while incorporating spatial contextual information. Specifically, we link the encoded embeddings of each modality with the initial embeddings of the other modality to create road and image residual embeddings, respectively, with the purpose of fusing cross-modal features from different stages. We then build a cross-modal adjacency matrix for each path based on spatial correspondences and contextual information. This matrix guides the GCN to iteratively fuse the residual embeddings of each modality separately, thus obtaining road and image fused embeddings. Finally, we apply contrastive loss to ensure the consistency of the fused embeddings across the two modalities. **As the final representation effectively integrates cross-stage features of the two modalities with spatial context information, this component not only achieves deep multi-modal fusion but also enhances the comprehensive utilization of information.**

The contributions of this work are delineated as follows:

- We propose a Multi-modal, Multi-granularity Path Representation Learning Framework that learns generic path representations applicable to various downstream tasks. To the best of our knowledge, MM-Path is the first model that leverages road network data and remote sensing images to learn generic path representations.
- We model and align the multi-modal path information using a fine-to-coarse multi-granularity alignment strategy. This strategy effectively captures both intricate local details and the broader global context of the path.
- We introduce a graph-based cross-modal residual fusion component. This component utilizes a cross-modal GCN to fully integrate information from different modalities while maintaining the consistency of dual modalities.
- We conduct extensive experiments on diverse tasks using two real-world datasets to demonstrate the adaptability and superiority of our model.

## 2 Preliminaries

### 2.1 Basic Conception

**Path.** A path  $p$  is a sequence of continuous junctions, which can be observed from the road network view and the image view.

**Road network.** A road network is denoted as  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V}$  and  $\mathcal{E}$  represent a set of nodes and edges, respectively. Node

$v \in \mathcal{V}$  is a road intersection or a road end. Edge  $e \in \mathcal{E}$  denotes a road segment connecting two nodes.

**Road paths.** We define the sequence of nodes on a road network for a path  $p$  as a road path  $\mathcal{R}(p) = \langle v_1, v_2, \dots, v_{|\mathcal{R}(p)|} \rangle$ , where each element represents a node, and  $|\mathcal{R}(p)|$  represents the length of the road path  $\mathcal{R}(p)$ . It is noted that there must be an edge  $e \in \mathcal{E}$  connecting any adjacent nodes in the road path.

**Image paths.** Given an interested region, we partition the region into fixed-size segments to generate a set of images,  $\mathcal{M}_{rs}$ , consisting of disjoint, fixed-size remote sensing images. Each image within this set is denoted as  $m \in \mathbb{R}^{r \times r \times c}$ , where  $c$  represents the number of channels, and  $(r, r)$  denotes the resolution. Subsequently, given road path  $\mathcal{R}(p)$ , the image path (i.e., image sequence of the path)  $\mathcal{M}(p)$  is formed by selecting a series of image  $m_i$  that correspond to specific latitudes and longitudes along the nodes in the road path.

For example, as shown in the upper part of Figure 2, consider the road path  $\mathcal{R}(p) = \langle v_1, \dots, v_8 \rangle$ , where nodes  $v_1, v_2$ , and  $v_3$  are located in image  $m_1$ , nodes  $v_4, v_5$  and  $v_6$  in image  $m_2$ , and nodes  $v_7$  and  $v_8$  in image  $m_3$ . This results in the image path  $\mathcal{M}(p) = \langle m_1, m_2, m_3 \rangle$ .

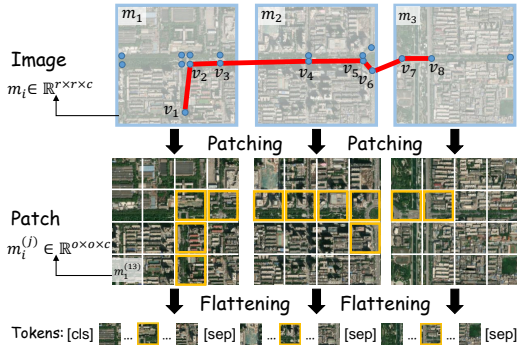


Figure 2: An example of image path processing

**Road Sub-paths.** Given a road path  $\mathcal{R}(p)$  and an image path  $\mathcal{M}(p)$ , the nodes of  $\mathcal{R}(p)$  located in the same image belong to a road sub-path.

Taken Figure 2 as an example, the road path  $\mathcal{R}(p) = \langle v_1, \dots, v_8 \rangle$  has three road sub-paths:  $s_1 = \langle v_1, v_2, v_3 \rangle$ ,  $s_2 = \langle v_4, v_5, v_6 \rangle$  and  $s_3 = \langle v_7, v_8 \rangle$ .

## 2.2 Problem Statement

Given the road path  $\mathcal{R}(p)$  and image path  $\mathcal{M}(p)$  of a path  $p$ , the goal is to learn an embedding function  $f$  that returns a generic representation of path  $p$ . This function can be formalized as follows:

$$\mathbf{x} = f(\mathcal{R}(p), \mathcal{M}(p)), \quad (1)$$

where  $\mathbf{x} \in \mathbb{R}^d$  represents the generic embedding of path  $p$ , and  $d$  denotes the dimension of the embedding  $\mathbf{x}$ .

These learned path embeddings are supposed to be generic, which should support a variety of downstream tasks, e.g., path travel time estimation and path ranking score estimation.

## 3 Methodology

Figure 3 illustrates the framework of MM-Path. This section introduces the framework of MM-Path, describes its two main components, and details the final training objective.

### 3.1 Overall Framework

Different from existing methods that are limited to data from a single modality, MM-Path leverages data from both road networks and images for pre-training, providing a more comprehensive perspective. MM-Path comprises two main components: the multi-granularity alignment component and the graph-based cross-modal residual fusion component.

The multi-granularity alignment component is designed to concentrate on path-related information while capturing fine-grained details and coarse-grained global context. Initially, we convert the image of interested region into fixed-sized image sequences to obtain image paths. Subsequently, We establish road path and image path encoding branches to process the two modalities. In each branch, a modal-specific tokenizer generates initial embeddings for each modality at three granularities: node/patch, road sub-path/image, and road path/image path. These initial embeddings are then processed by road and image transformers to produce road and image encoded embeddings, respectively, which are also generated at the same three granularities. A multi-granularity loss function is utilized to synchronize the semantic information of road encoded embeddings and image encoded embeddings, and to capture their interrelations at different granularities, from fine to coarse.

The graph-based cross-modal residual fusion component is designed to effectively fuse cross-modal heterogeneous data. A cross-modal residual connection merges the encoded embedding from each branch with the initial embedding from the other branch, generating road and image residual embeddings. This connection considers cross-modal features at different stages, promoting deep cross-modal feature fusion. Subsequently, we construct a cross-modal adjacency matrix for each path based on spatial correspondences and contextual information. This matrix, embedded within a GCN, guides the fusion of the two modalities for each branch. Consequently, a fused embedding is obtained for each branch. We introduce a contrastive loss to ensure the consistency between the road fused embedding and image fused embedding. Finally, we concatenate these two fused embeddings to obtain a generic path representation.

### 3.2 Multi-granularity Alignment

We model the road paths and image paths using Transformer architecture, respectively. We then construct a multi-granularity loss function to ensure alignment between these two modalities.

**3.2.1 Input Representations.** Due to information granularity discrepancies between road paths and image paths, direct alignment is often disturbed by irrelevant information. To solve this problem, we use a sequence of fixed-size images, instead of a single image commonly used in traditional image-text multi-modal methods [2, 30, 37, 40], to model image paths. This procedure preserves the scale and shape features of the images by avoiding distortions caused by inconsistencies in image sizes. Furthermore, as these fixed-size images can be utilized across different paths, the storage of images is

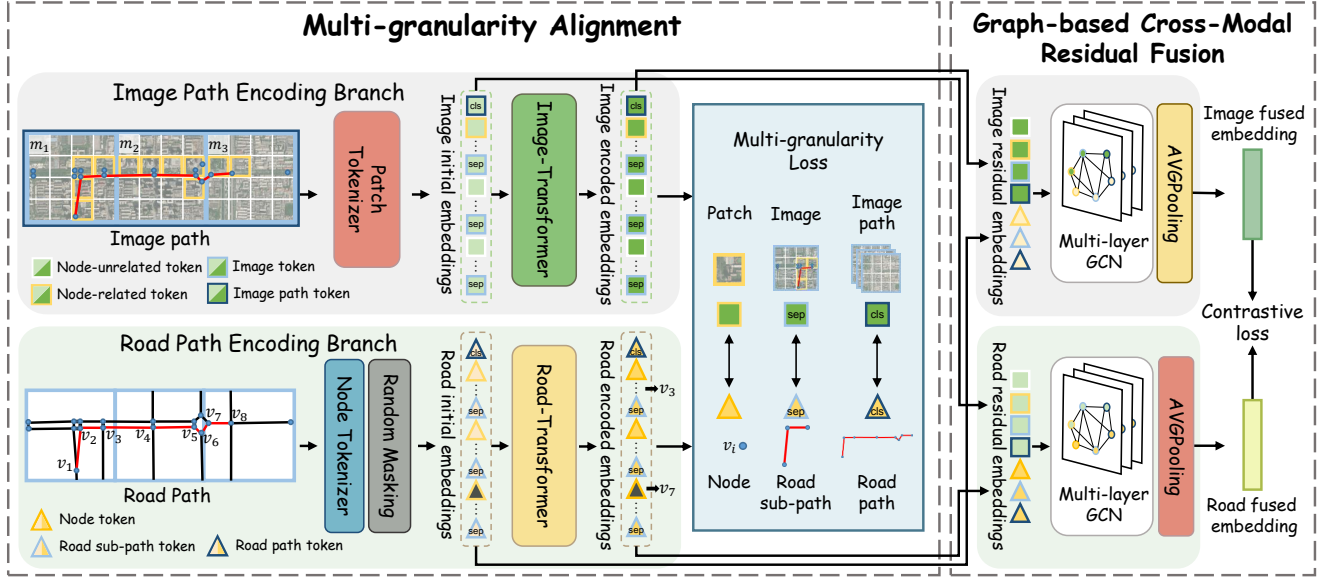


Figure 3: Overall framework of MM-Path

reduced. Then, we utilize specialized tokenizers to separately encode the data of each modality into a unified format.

The patch tokenizer segments each image within an image path into a series of patches to extract fine-grained semantic information. Specifically, as shown in Figure 2, an image  $m_i \in \mathbb{R}^{r \times r \times c}$  is reshaped into a sequence of  $r^2/o^2$  (e.g., 16) patches, where  $c$  represents the number of channels,  $(r, r)$  denotes the resolution of fixed-size image, and  $(o, o)$  defines the resolution per patch. After patching, we concatenate the patch sequences from all images within a image path to form a unified patch sequence. Then, we place a special [cls] token at the beginning of the patch sequence. As the [cls] token captures the global information of the entire sequence [1], it can be regarded as a representation of the entire image path. Special [sep] tokens are placed at the end of each image to delineate local information of each image. For example, the token sequence of image path  $\mathcal{M}(p) = \langle m_1, m_2, m_3 \rangle$  is  $[\text{cls}, m_1^{(1)}, \dots, m_1^{(16)}, \text{sep}, \dots, \text{sep}, m_3^{(1)}, \dots, m_3^{(16)}, \text{sep}]$ , where  $m_i^{(j)} \in \mathbb{R}^{o \times o \times c}$  denotes the  $j$ -th patch of the  $i$ -th image.

Each patch  $m_i^{(j)}$  is then projected into a patch embedding  $\mathbf{m}_i^{(j)} \in \mathbb{R}^d$ , which can be initialized using pre-trained ResNet50 [20]. The image initial embeddings are computed by summing the patch embeddings with the image position embeddings  $\mathbf{T}_{\text{image}} \in \mathbb{R}^{n_1 \times d}$ , resulting in  $\mathbf{H}^{(0)} = [\mathbf{m}_{\text{cls}}, \mathbf{m}_1^{(1)}, \dots, \mathbf{m}_{|\mathcal{M}(p)|}^{(r^2/o^2)}, \mathbf{m}_{\text{sep}}] + \mathbf{T}_{\text{image}}$ . Here,  $\mathbf{m}_{\text{cls}}$  and  $\mathbf{m}_{\text{sep}}$  are the image initial embeddings of the [cls] and [sep] tokens, respectively.  $n_1$  denotes the length of the patch token sequence, and  $d$  represents the dimension of the embeddings.

The modeling for a road path is similar, starting with a [cls] token at the beginning and placing [sep] tokens at the end of each road sub-path. For instance, a road path  $\mathcal{R}(p)$  comprising three road sub-paths— $s_1 = \langle v_1, v_2, v_3 \rangle$ ,  $s_2 = \langle v_4, v_5, v_6 \rangle$ , and  $s_3 = \langle v_7, v_8 \rangle$ —generates the node token sequence  $[\text{cls}, v_1, v_2, v_3, \text{sep}, v_4, v_5, v_6, \text{sep}, v_7, v_8, \text{sep}]$ . Then, the node tokenizer linearly projects each node  $v_i$  from

the road path  $\mathcal{R}(p)$  to a node embedding  $\mathbf{v}_i \in \mathbb{R}^d$ , initialized using Node2vec [16]. We also integrate standard learnable road position embeddings  $\mathbf{T}_{\text{road}} \in \mathbb{R}^{n_2 \times d}$  with the node embeddings to generate the road initial embeddings  $\mathbf{P}^{(0)} = [\mathbf{v}_{\text{cls}}, \mathbf{v}_1, \dots, \mathbf{v}_{|\mathcal{R}(p)|}, \mathbf{v}_{\text{sep}}] + \mathbf{T}_{\text{road}}$ , where  $\mathbf{v}_{\text{cls}}$  and  $\mathbf{v}_{\text{sep}}$  represent the road initial embeddings of the [cls] and [sep] tokens, respectively.  $n_2$  denotes the length of the road path token sequence.

**3.2.2 Image Path Encoding.** To model the image path, the image initial embeddings  $\mathbf{H}^{(0)}$  are fed into the  $l$  layers Image-Transformer, which can be formulated as,

$$\mathbf{H}^{(j)} = \text{Image-Transformer}(\mathbf{H}^{(j-1)}), \quad (2)$$

where  $j = 1, \dots, l$ , and  $\mathbf{H}^{(l)} \in \mathbb{R}^{n_1 \times d}$  is the final output of the Image-Transformer. For a simplify, we denote the image encoded embeddings  $\mathbf{H}^{(l)}$  as  $\mathbf{H} = [\mathbf{h}_{\text{cls}}, \mathbf{h}_1^{(1)}, \dots, \mathbf{h}_{|\mathcal{M}(p)|}^{(r^2/o^2)}, \mathbf{h}_{\text{sep}}]_{|\mathcal{M}(p)|}$ .  $\mathbf{h}_i^{(j)} \in \mathbb{R}^d$  denotes the encoded embedding of the  $j$ -th patch of the  $i$ -th image.  $\mathbf{h}_{\text{sep}_i}, \mathbf{h}_{\text{cls}} \in \mathbb{R}^d$  represent the encoded embeddings of the  $i$ -th image and the whole image path  $\mathcal{M}(p)$ , respectively.

**3.2.3 Road Path Encoding.** To facilitate alignment between road paths and image paths, we utilize a similar Transformer architecture for road path modeling. The encoder is comprised of  $l$  layers of Transformer blocks, defined as:

$$\mathbf{P}^{(j)} = \text{Road-Transformer}(\mathbf{P}^{(j-1)}), \quad (3)$$

where  $j = 1, \dots, l$ ,  $\mathbf{P}^{(j)} \in \mathbb{R}^{n_2 \times d}$  is the output of the  $j$ -th layer. In a brief, the road encoded embeddings  $\mathbf{P}^{(l)}$  are represented as  $\mathbf{P} = [\mathbf{p}_{\text{cls}}, \mathbf{p}_1, \dots, \mathbf{p}_{|\mathcal{R}(p)|}, \mathbf{p}_{\text{sep}}]_{|\mathcal{R}(p)|}$ . Here,  $\mathbf{p}_i, \mathbf{p}_{\text{sep}_i}, \mathbf{p}_{\text{cls}} \in \mathbb{R}^d$  denote the encoded embeddings of the  $i$ -th node, the sub-path  $s_i$ , and the entire road path  $\mathcal{R}(p)$ , respectively.  $|\mathcal{R}(p)|$  represents the number of nodes in road path  $\mathcal{R}(p)$ , and  $|\mathcal{M}(p)|$  indicates the number of images in image path  $\mathcal{M}(p)$ , which also corresponds to the number of road sub-paths.



To better capture the complex dependencies in a path, similar to masked language modeling task (MLM) [22], we use a masked node modeling task as a self-supervised task. The intuition behind this is that the information density of individual pixels or patches in an image is relatively low compared to the topological structure information relevant to the path. Therefore, image masking tasks are not deemed essential. To this end, we propose to employ node masking tasks. In particular, we randomly mask the nodes in road paths (Ref. as to the gray triangle in Figure 3), and then use a softmax classifier to predict the node tokens corresponding to the masked nodes. The loss function for training is defined as follows:

$$\mathcal{L}_{\text{mask}} = - \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{D}} \log P(v_i | v_i^{\text{mask}}), \quad (4)$$

where  $\mathcal{P}$  represents the training sets of all paths,  $\mathcal{D}$  is randomly masked positions of road path, and  $v_i^{\text{mask}}$  is the node that is masked according to  $\mathcal{D}$ .

**3.2.4 Modalities Aligning.** The encoded embeddings from each branch capture the hidden semantic information within their respective modality, including fine-grained node/patch embeddings, medium-grained road sub-path/image embeddings, and coarse-grained entire road path/image path embeddings. We aim for embeddings with similar semantics across modalities to be proximate within the embedding space. Additionally, we seek detailed alignment between the two modalities while maintaining global correspondence. Accordingly, we design a loss function that operates at three distinct levels of granularity—fine, medium and coarse—corresponding to node/patch, road sub-path/image, and entire road path/image path, respectively.

**Fine granularity.** Since each patch may contain more than one node, the encoded embeddings of a node and the corresponding patch (Ref. as to the dark yellow triangle the dark green rectangle with yellow borders in Figure 3) should maintain directional consistency. To precisely capture the semantic information of fine-grained paths, we minimize the cosine distance between the encoded embeddings of nodes and their corresponding patches. Consequently, we construct the fine-grained loss function as follows:

$$\mathcal{L}_{\text{fine}} = \sum_{p \in \mathcal{P}} \sum_{v_i \in \mathcal{R}(p), L(v_i)=m_j^{(k)}} \left(1 - \frac{\mathbf{p}_i \cdot \mathbf{h}_j^{(k)}}{\|\mathbf{p}_i\| \|\mathbf{h}_j^{(k)}\|}\right), \quad (5)$$

where  $\mathcal{P}$  represents the training set of paths,  $L(v_i)$  is a function that returns the patch corresponding to the node  $v_i$ ,  $\mathbf{p}_i$  and  $\mathbf{h}_j^{(k)}$  are the encoded embeddings of  $v_i$  and  $m_j^{(k)}$ , respectively.

**Medium granularity.** Similarly, to align road sub-paths with images, we construct the following medium-grained loss function:

$$\mathcal{L}_{\text{medium}} = \sum_{p \in \mathcal{P}} \sum_{s_i \in \mathcal{R}(p)} \left(1 - \frac{\mathbf{p}_{\text{sep}_i} \cdot \mathbf{h}_{\text{sep}_i}}{\|\mathbf{p}_{\text{sep}_i}\| \|\mathbf{h}_{\text{sep}_i}\|}\right), \quad (6)$$

where  $\mathbf{p}_{\text{sep}_i}$  and  $\mathbf{h}_{\text{sep}_i}$  (Ref. as to the dark yellow rectangle and the dark green triangle with blue borders in Figure 3) are the encoded embeddings of road sub-path  $s_i$  and the corresponding image  $m_i$ , respectively.

**Coarse granularity.** Due to the unique correspondence between road path and the corresponding image path, a clearer distinction is necessary. Therefore, We construct a contrastive loss function

for coarse-grained data. Considering a batch of road path-image path pairs  $\mathcal{B}$ , the objective of this contrastive learning loss is to accurately identify the matched pairs among the  $|\mathcal{B}| \times |\mathcal{B}|$  possible combinations. Within a training batch, there are  $|\mathcal{B}|^2 - |\mathcal{B}|$  negative pairs. The contrastive learning loss function can be formulated as:

$$\mathcal{L}_{\text{coarse}} = - \sum_{p \in \mathcal{P}} \left( \log \left( \frac{\exp(\text{sim}(\mathbf{p}_{\text{cls}}, \mathbf{h}_{\text{cls}})/\sigma)}{\sum_{m^{\text{Neg}} \in \mathcal{B}} \exp(\text{sim}(\mathbf{p}_{\text{cls}}, \mathbf{h}_{\text{cls}}^{\text{Neg}})/\sigma)} \right) + \frac{\exp(\text{sim}(\mathbf{p}_{\text{cls}}, \mathbf{h}_{\text{cls}})/\sigma)}{\sum_{p^{\text{Neg}} \in \mathcal{B}} \exp(\text{sim}(\mathbf{p}_{\text{cls}}^{\text{Neg}}, \mathbf{h}_{\text{cls}})/\sigma)} \right), \quad (7)$$

where  $\mathbf{p}_{\text{cls}}$  and  $\mathbf{h}_{\text{cls}}$  (Ref. as to the dark yellow rectangle and the dark green triangle with dark blue borders in Figure 3) correspond to the encoded embeddings of the entire road path and image path in a road path-image path pairs.  $p^{\text{Neg}}$  and  $m^{\text{Neg}}$  are the negative road path and image path in the batch set  $\mathcal{B}$ , respectively.  $\sigma$  is a learned temperature parameter.  $\text{sim}(\mathbf{p}_{\text{cls}}, \mathbf{h}_{\text{cls}})$  returns the Euclidean distance between  $\mathbf{p}_{\text{cls}}$  and  $\mathbf{h}_{\text{cls}}$ .

Finally, the multi-granularity loss can be formulated as  $\mathcal{L}_{\text{multi}} = \mathcal{L}_{\text{fine}} + \mathcal{L}_{\text{medium}} + \mathcal{L}_{\text{coarse}}$ .

### 3.3 Graph-based Cross-modal Residual Fusion

In this framework, each path is represented through two distinct modalities, providing complementary perspectives. To effectively leverage these modalities, we propose a graph-based cross-modal residual fusion component.

**3.3.1 Cross-modal Residual Connection.** To facilitate comprehensive information exchange between modalities, we introduce cross-modal residual connections that effectively concatenate embeddings across different stages and modalities. These connections enable direct propagation of gradients to earlier layers, thereby enhancing stability and improving training efficiency. Specifically, we concatenate the road initial embeddings  $\mathbf{P}^{(0)}$  with the image encoded embeddings  $\mathbf{H}$ , and the image initial embeddings  $\mathbf{H}^{(0)}$  with the road encoded embeddings  $\mathbf{P}$ . The resulting image residual embeddings and road residual embeddings are defined as  $\mathbf{U} = \mathbf{P}^{(0)} \parallel \mathbf{H}$  and  $\mathbf{Q} = \mathbf{P} \parallel \mathbf{H}^{(0)}$ , respectively. Here,  $\mathbf{U}, \mathbf{Q} \in \mathbb{R}^{(n_1+n_2) \times d}$ , and  $\parallel$  denotes the concatenation operation.

**3.3.2 Graph-based Fusion.** Although traditional attention mechanisms proficiently identify correlations among entities, they often fail to incorporate contextual information concurrently. To address this limitation, we utilize graph neural networks [4], which incorporate contextual information into the learning process by representing it as graph structures. Leveraging this capability, we introduce a graph-based fusion method to enhance the accuracy of information understanding across different modalities.

Initially, we construct a specialized cross-modal directed graph for each path. This graph treats all tokens, including [cls] and [sep] tokens from both modalities, as entities. These entities are connected via three types of relationships: intra-modal context, cross-modal correspondence, and cross-modal context. The intra-modal context focuses on interactions within a single modality, facilitating a deep understanding of its specific information. Cross-modal correspondence aids in comprehending and learning the spatial correspondence between different modalities. Cross-modal context addresses indirect relationships between different modal entities, which enhances

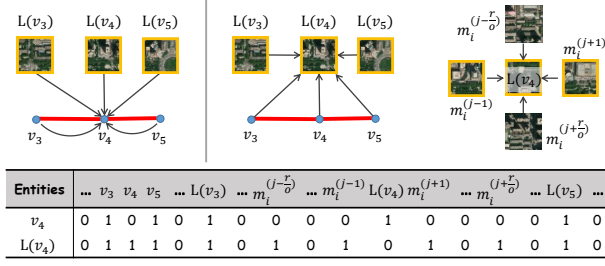


Figure 4: An example of multi-modal graph construction

the model's ability to interpret complex scenes. Collectively, leveraging these relationships significantly boosts the model's capacity to handle multi-modal data effectively.

Figure 4 demonstrates the construction of the graph. Taking node  $v_4$  as an example, node  $v_4$  is connected by directed edges from five entities: adjacent context nodes  $v_3$  and  $v_5$ , its corresponding patch  $L(v_4)$ , and their respective patches  $L(v_3)$  and  $L(v_5)$ . The patch  $L(v_4)$  (i.e.,  $m_i^{(j)}$ ) is connected by directed edges from nine entities, including  $v_3$ ,  $v_4$ ,  $v_5$ ,  $L(v_3)$ ,  $L(v_5)$ , and four geographically adjacent image patches  $m_i^{(j-1)}$ ,  $m_i^{(j+1)}$ ,  $m_i^{(j-\frac{r}{o})}$  and  $m_i^{(j+\frac{r}{o})}$ , where  $\frac{r}{o}$  denotes the number of patches per row of an image. Additionally, the [sep] tokens are connected by their context [sep] tokens, corresponding [sep] tokens from another modality, and cross-modal context tokens. The [cls] token, encapsulating more global information, is connected by all [sep] tokens and corresponding [cls] token from another modality.

Then, we construct an adjacency matrix  $\mathbf{A} \in \mathbb{R}^{(n_1+n_2) \times (n_1+n_2)}$  for each path  $p$  to capture the comprehensive relation within the multi-modal data. Given the effectiveness of GCNs in transferring and fusing information across entities within a graph structure, we employ a GCN to derive the updated embeddings for both branches. The updated embeddings are computed as follows:

$$\hat{\mathbf{U}} = \text{Relu} \left( \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \text{Relu} \left( \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{U} \mathbf{W}_1 \right) \mathbf{W}_2 \right), \quad (8)$$

$$\hat{\mathbf{Q}} = \text{Relu} \left( \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \text{Relu} \left( \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{Q} \mathbf{W}_3 \right) \mathbf{W}_4 \right), \quad (9)$$

where  $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{W}_4 \in \mathbb{R}^{d \times d}$  are weight matrices, and  $\tilde{\mathbf{D}}$  is the degree matrix of  $\tilde{\mathbf{A}}$ . The augmented adjacency matrix  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}'$ , where  $\mathbf{I}'$  is a modified identity matrix with all diagonal elements set to 1, except for those corresponding to patches without relationship to any nodes (Ref. as to the dark green rectangle with a white border in Figure 3). This modification aims to exclude patches that are relatively unrelated to the path, thereby preventing the introduction of noise into the model.

After iterative graph convolution operations, the embeddings of each entity within the graph are updated. We perform average pooling on  $\hat{\mathbf{U}}$  and  $\hat{\mathbf{Q}}$  to aggregate the updated embeddings, respectively. The fused embedding for each branch is then obtained by:

$$\mathbf{y} = \text{AvgPooling}(\hat{\mathbf{U}}), \quad (10)$$

$$\mathbf{z} = \text{AvgPooling}(\hat{\mathbf{Q}}), \quad (11)$$

where  $\mathbf{y}, \mathbf{z} \in \mathbb{R}^d$  denote the image fused embedding and the road fused embedding, respectively.

**3.3.3 Cross-modal Contrastive Loss.** The image fused embedding  $\mathbf{y}$  and the road fused embedding  $\mathbf{z}$  encapsulate features across multiple modalities of the same path, reflecting an inherent similarity. Therefore, we implement a quadruplet loss function to ensure that the difference between  $\mathbf{y}$  and  $\mathbf{z}$  is smaller than the differences with the fused embeddings of other paths. For negative samples, we randomly sample the image fused embedding  $\mathbf{y}_N$  and the road fused embedding  $\mathbf{z}_N$  from the batch. The loss function is defined as:

$$\mathcal{L}_{\text{fuse}} = - \sum_{p \in \mathcal{P}} ([\|\mathbf{y} - \mathbf{z}\|_2^2 - \|\mathbf{y} - \mathbf{z}_N\|_2^2 + \beta]_+ + [\|\mathbf{y} - \mathbf{z}\|_2^2 - \|\mathbf{z} - \mathbf{y}_N\|_2^2 + \beta]_+), \quad (12)$$

where  $\beta$  is a hyperparameter that controls the margin of the distance between pairs of positive and negative samples, and  $[\cdot]_+$  is a shorthand for  $\max(0, \cdot)$ .

### 3.4 Training objective

The final training objective of our model integrates all previously proposed loss functions, formulated as follows:

$$\mathcal{L} = \lambda_{\text{mask}} \mathcal{L}_{\text{mask}} + \lambda_{\text{multi}} \mathcal{L}_{\text{multi}} + \lambda_{\text{fuse}} \mathcal{L}_{\text{fuse}}, \quad (13)$$

where  $\lambda_{\text{mask}}$ ,  $\lambda_{\text{multi}}$  and  $\lambda_{\text{fuse}}$  are the weights assigned to  $\mathcal{L}_{\text{mask}}$ ,  $\mathcal{L}_{\text{multi}}$ , and  $\mathcal{L}_{\text{fuse}}$ .

After pre-training, we combine the image fused embedding  $\mathbf{y}$  with the road fused embedding  $\mathbf{z}$  into a generic path embedding  $\mathbf{x} = \mathbf{y} \parallel \mathbf{z}$ , achieving a more robust and generalized representation. The generic path embedding is then fine-tuned using interchangeable linear layer task heads, enabling the model to adapt to a variety of downstream tasks effectively.

## 4 Experiments

### 4.1 Experimental Setups

**4.1.1 Datasets.** We utilize the road networks, GPS datasets, and remote sensing image datasets of two cities: Aalborg, Denmark, and Xi'an, China. The road networks are sourced from OpenStreetMap<sup>1</sup>, while the remote sensing image datasets are acquired from Google Earth Engine[15]. Employing an existing tool [31], we map-match all GPS records to road networks to generate the path datasets and historical trajectory datasets. The details of the datasets are shown in Table 1.

**4.1.2 Implementation Details.** All experiments are conducted using PyTorch [33] on Python 3.8 and executed on an NVIDIA Tesla-A800 GPU. Each fixed-size image is  $500 \times 500$  pixels, with each pixel corresponding to 2 meters on the earth. In other words, an image covers a  $1\text{km} \times 1\text{km}$  region. We segment each image into 16 patches and set the embedding dimension  $d$  to 64. Both the Image-Transformer and Road-Transformer comprise five layers. To enhance the pre-training efficiency, we initialize our Road-Transformer with the pre-trained LightPath [45]. The mask ratio is set at 15%. The weights  $\lambda_{\text{mask}}$ ,  $\lambda_{\text{fuse}}$ , and  $\lambda_{\text{multi}}$  are uniformly set to 1. Training proceeds for up to 60 epochs with a learning rate of 0.02. The linear layer task head includes two fully connected layers, with dimensions of 32 and 1, respectively. Training MM-Path on the Aalborg and

<sup>1</sup><https://www.openstreetmap.org>

**Table 1: Data statistics**

	Aalborg	Xi'an
Number of nodes	7,561	7,051
Number of edges	9,605	9,642
AVG edge length (m)	124.78	86.49
Number of path	47,865	200,000
AVG node number per road path	25.77	55.39
AVG path length (m)	3,252.75	4,743.70
Number of traj.	149,246	797,882
Max travel time of traj. (s)	3,549	8,638
Avg travel time of traj. (s)	199	662
Number of images	950	133
AVG number of nodes per image	7.96	53.01
AVG image number per image path	6.28	6.87

Xi'an datasets takes 78 and 161 minutes, respectively. Since training is conducted offline, the runtime is acceptable.

**4.1.3 Downstream Tasks and Metrics. Path Travel Time Estimation:** We calculate the average travel time (in seconds) for each path based on historical trajectories. The accuracy of travel time estimations is evaluated using three metrics: Mean Absolute Error (MAE), Mean Absolute Relative Error (MARE), and Mean Absolute Percentage Error (MAPE). **Path Ranking Score Estimation (Path Ranking):** Each path is assigned a ranking score ranging from 0 to 1, derived from historical trajectories by following existing studies [44–46]. We evaluate the effectiveness of path ranking using MAE, the Kendall rank correlation coefficient ( $\tau$ ), and Spearman's rank correlation coefficient ( $\rho$ ).

**4.1.4 Baselines.** We compare the proposed model with 5 unsupervised single-modal path pre-trained methods and 5 unsupervised multi-modal methods. The single-modal path pre-trained methods are:

- **Node2vec** [16]: This is an unsupervised model that learn node representation based on a graph.
- **PIM** [3]: This is an unsupervised path representation learning approach based on mutual information maximization.
- **LightPath** [45]: This is a lightweight and scalable path representation learning method.
- **TrajCL** [5]: This is a contrastive learning-based trajectory modeling method.
- **START** [21]: This is a self-supervised trajectory representation learning framework with temporal regularities and travel semantics.

The multi-modal methods are:

- **CLIP** [37]: This is a classic pre-trained multi-modal model. For each path, we use a single rectangular image for the image modality and replace the original text sequence with a node sequence. After pre-training, we concatenate the representations of the two modalities and use them as input to the linear layer task head.

- **USPM** [8] utilizes both images and road network to profile individual streets, but not paths (i.e., sequences of streets). We adapt USPM to support path representation learning.
- **JGRM** [29]: This is a representation learning framework that combines GPS data and road network-based data. In this study, we replace the GPS data with the image path.
- **LightPath+image**: This is a multi-modal variant of LightPath. We concatenate the patch embedding with the node embedding to replace the original node embedding for training.
- **START+image**: This is a multi-modal variant of START, processed similarly to LightPath+image.

For all methods, we standardize the embedding dimensionality ( $d$ ) to 50. All parameters are set according to the specifications in the original papers. All baselines are fine-tuned using a linear layer task head. The output of this task head serves as the prediction result.

For all methods, we initially pre-train using unlabeled training data (e.g., 30K unlabeled Aalborg dataset and 160K unlabeled Xi'an dataset). Subsequently, we use a smaller volume of labeled data (e.g., 10K labeled Aalborg dataset and 40K labeled Xi'an dataset) for task-specific fine-tuning. Validation and evaluation processes are conducted on separate validation dataset (e.g., 5K Aalborg dataset and 20K Xi'an dataset) and test dataset (e.g., 10K Aalborg dataset and 40K Xi'an dataset), respectively.

## 4.2 Experimental Results

**4.2.1 Overall Performance.** Table 2 presents the overall performance on both tasks. We use '↑' (and '↓') to indicate that larger (and smaller) values are better. For each task, we highlight the best and second-best performance in **bold** and underline. "Improvement" and "Improvement\*" quantify the enhancements achieved by MM-Path over the best single-modal and multi-modal baselines, respectively.

Overall, MM-Path outperforms all baselines on these tasks across both datasets, demonstrating its superiority. Specifically, we can make the following observations: The graph representation learning method Node2vec significantly underperforms compared to MM-Path, primarily due to its focus solely on the topological information of nodes while overlooking the sequential information of paths. Single-modal models like PIM, LightPath, and TrajCL show improved performance over Node2vec, indicating the importance of capturing sequential correlations within paths. Among the single-modal models, START achieves the best performance. It adeptly integrate sequential path information with spatio-temporal transition relationships derived from historical trajectory data. However, as a single-modal model, its capabilities are inherently constrained. As a multi-modal model, CLIP exhibits the weakest performance. Designed primarily for general corpora, it focuses on single, coarse-grained image representations, which often introduce noise into path modeling. Consequently, CLIP struggles to effectively capture complex spatial information and correspondences, making it unsuitable for modeling paths. USPM performs poorly because it analyzes individual streets using images and road networks, rather than paths (i.e., street sequences). As a result, it fails to effectively mine the sequential relationships present in the two modalities. The variants LightPath+image and START+image perform comparably to their single-modal models (i.e., LightPath and START), suggesting that

**Table 2: Overall accuracy on travel time estimation and path ranking**

Methods	Aalborg						Xi'an					
	Travel Time Estimation			Path Ranking			Travel Time Estimation			Path Ranking		
	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	$\tau$ ↑	$\rho$ ↑	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	$\tau$ ↑	$\rho$ ↑
Node2vec [16]	76.228	0.281	54.182	0.203	0.119	0.140	227.129	0.269	30.919	0.218	0.079	0.098
PIM [3]	63.812	0.237	47.054	0.144	0.284	0.343	207.266	0.246	27.716	0.207	0.091	0.102
Lightpath [45]	58.818	0.221	40.219	0.124	0.413	0.483	201.400	0.229	26.429	0.178	0.209	0.252
TrajCL [5]	53.822	0.208	34.239	<u>0.113</u>	0.499	0.577	202.757	0.238	26.506	0.181	0.211	0.256
START [21]	<u>51.176</u>	0.191	34.315	0.117	0.475	0.556	<u>199.843</u>	0.215	<u>25.022</u>	0.179	<u>0.229</u>	<u>0.279</u>
CLIP [37]	72.155	0.261	50.284	0.162	0.179	0.185	219.048	0.256	30.962	0.213	0.087	0.099
USPM [8]	66.714	0.249	51.916	0.148	0.308	0.383	205.594	0.244	26.039	0.209	0.105	0.110
JGRM [29]	51.251	0.193	<u>32.380</u>	0.115	<u>0.512</u>	<u>0.592</u>	201.010	0.228	26.400	<u>0.177</u>	0.228	0.262
Lightpath+image	59.698	0.224	40.920	0.131	0.383	0.405	205.556	0.242	27.058	0.182	0.188	0.231
START+image	51.859	<u>0.188</u>	33.401	0.122	0.437	0.521	200.059	<u>0.211</u>	26.046	0.184	0.183	0.226
<b>MM-Path</b>	<b>47.756</b>	<b>0.172</b>	<b>29.808</b>	<b>0.106</b>	<b>0.558</b>	<b>0.643</b>	<b>187.452</b>	<b>0.193</b>	<b>23.644</b>	<b>0.165</b>	<b>0.257</b>	<b>0.294</b>
Improvement	6.682%	9.947%	12.941%	6.194%	11.823%	11.443%	6.201%	10.236%	5.507%	7.303%	12.227%	5.376%
Improvement*	6.819%	8.511%	7.943%	7.826%	8.984%	8.614%	6.312%	8.531%	9.222%	6.780%	12.719%	12.213%

merely concatenating two modalities does not effectively enhance multi-modal fusion. Having adapted JGRM to integrate image paths and road paths, JGRM outperforms other multi-modal baselines. It is specifically designed for multi-modal integration and excels at merging information from various sources. However, JGRM's limitations in handling multi-modal information of varying granularities and its lack of use of cross-modal context information to guide the fusion process make its performance less optimal compared to MM-Path.

**4.2.2 Ablation Study.** We design eight variants of MM-Path to verify the necessity of the components of our model: (1) MM-Path-z: This variant leverages the road fused embedding  $z$  as a generic representation of the path. (2) MM-Path-y: This model utilizes the image fused embedding  $y$  as a generic representation of the path. (3) w/o alignment: This version excludes the multi-granularity loss. (4) w/o fusion: This variant substitutes the graph-based residual fusion component with average pooling of the encoded embeddings from both modalities. (5) w/o GCN: This model replaces the GCN in the graph-based cross-modal residual fusion component with a cross-attention mechanism. (6) w/o fine, (7) w/o medium, and (8) w/o coarse: These variants omit the fine-grained, medium-grained, and coarse-grained loss, respectively.

The results are summarized in Tables 3 and 4. We can observe that MM-Path w/o alignment shows poor performance, which is attributed to its reliance solely on multi-modal data fusion without considering multi-granularity alignment. The variants, w/o fine, w/o medium, and w/o coarse, outperform w/o alignment but still worse than the full MM-Path, demonstrating the importance of multiple granularity alignments. MM-Path w/o fusion also exhibits poor performance, while MM-Path w/o GCN performs slightly worse than MM-Path. These results indicate that complex fusion methods with cross-modal context information enhance path understanding. Both MM-Path-y and MM-Path-z demonstrate comparable performance in travel time estimation and path ranking. This indicates that different modal perspectives contribute valuable insights for various downstream tasks. The overall performance of MM-Path surpasses all variants. This result implies that each of the proposed components

significantly enhances the model's effectiveness. This conclusively validates that MM-Path optimally utilizes all designed components.

**Table 3: Effect of variants of MM-Path in Aalborg**

Methods	Aalborg					
	Travel Time Estimation			Path Ranking		
	MAE	MARE	MAPE	MAE	$\tau$	$\rho$
MM-Path-z	49.649	0.185	30.193	0.114	0.528	0.622
MM-Path-y	48.529	0.181	32.722	0.118	0.511	0.603
w/o alignment	52.832	0.201	36.251	0.131	0.300	0.379
w/o fusion	51.237	0.192	30.529	0.115	0.476	0.560
w/o GCN	48.651	0.183	33.371	0.111	0.532	0.619
w/o fine	51.641	0.192	33.277	0.129	0.441	0.523
w/o medium	50.932	0.187	34.250	0.114	0.494	0.583
w/o coarse	50.688	0.189	35.341	0.117	0.505	0.596
<b>MM-Path</b>	<b>47.756</b>	<b>0.172</b>	<b>29.808</b>	<b>0.106</b>	<b>0.558</b>	<b>0.643</b>

**Table 4: Effect of variants of MM-Path in Xi'an**

Methods	Xi'an					
	Travel Time Estimation			Path Ranking		
	MAE	MARE	MAPE	MAE	$\tau$	$\rho$
MM-Path-z	194.301	0.231	24.455	0.183	0.199	0.241
MM-Path-y	196.331	0.233	24.747	0.196	0.178	0.214
w/o alignment	200.335	0.239	26.459	0.195	0.131	0.167
w/o fusion	200.652	0.239	25.433	0.208	0.113	0.130
w/o GCN	189.659	0.221	24.496	0.173	0.234	0.286
w/o fine	199.214	0.235	26.913	0.177	0.226	0.275
w/o medium	192.514	0.229	24.826	0.175	0.227	0.278
w/o coarse	194.256	0.230	25.757	0.176	0.231	0.278
<b>MM-Path</b>	<b>187.452</b>	<b>0.193</b>	<b>23.644</b>	<b>0.165</b>	<b>0.257</b>	<b>0.294</b>

**4.2.3 Effect of Pre-training.** In this section, we evaluate the effect of pre-training. We vary the size of labeled data used for fine-tuning, and compare the performance of the proposed MM-Path (Pre-trained) with its variant that lacks pre-training (No Pre-trained). Figure 5 shows the performance of travel time estimation and path



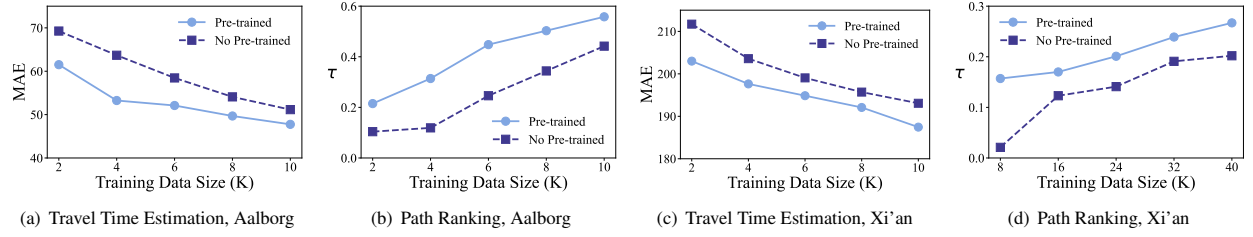


Figure 5: Effect of pre-training

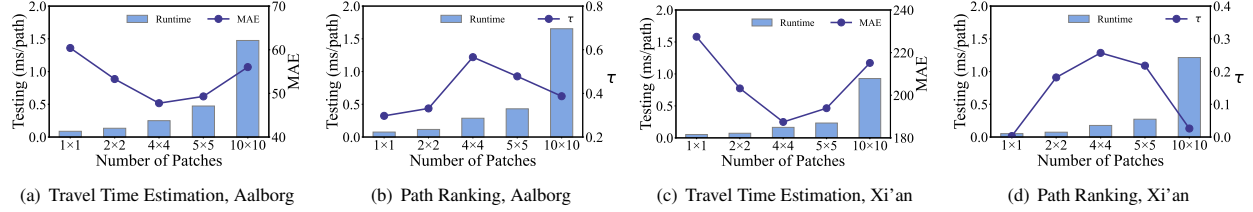


Figure 6: Effect of varying granularity size

Table 5: Accuracy on travel time estimation and path ranking for short paths

Methods	Aalborg						Xi'an					
	Travel Time Estimation			Path Ranking			Travel Time Estimation			Path Ranking		
	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑
Node2vec [16]	67.898	0.306	55.179	0.201	0.140	0.153	179.828	0.269	32.512	0.220	0.081	0.084
PIM [3]	55.871	0.252	44.605	0.150	0.292	0.348	150.356	0.238	26.221	0.206	0.174	0.198
Lightpath [45]	52.708	0.237	40.414	0.127	0.445	0.502	149.816	0.237	26.702	0.180	0.201	0.249
TrajCL [5]	49.427	0.223	34.466	0.116	0.523	0.597	150.657	0.239	26.891	0.183	0.219	0.260
START [21]	<u>47.347</u>	0.216	36.523	0.122	0.513	0.586	<u>147.745</u>	<u>0.227</u>	<u>25.973</u>	0.179	0.219	<u>0.277</u>
CLIP [37]	62.300	0.281	49.164	0.166	0.176	0.217	170.818	0.271	33.201	0.215	0.079	0.095
USPM [8]	57.415	0.274	48.949	0.152	0.314	0.360	149.012	0.234	26.942	0.207	0.133	0.151
JGRM [29]	47.770	<u>0.213</u>	<u>33.715</u>	0.117	0.516	<u>0.612</u>	148.832	0.236	26.341	<u>0.176</u>	<u>0.221</u>	0.262
Lightpath+image	54.819	0.247	42.258	0.134	0.383	0.434	150.657	0.239	26.891	0.184	0.186	0.212
START+image	49.427	0.223	34.462	0.125	0.476	0.548	148.684	0.236	26.503	0.185	0.181	0.231
<b>MM-Path</b>	<b>44.092</b>	<b>0.202</b>	<b>31.058</b>	<b>0.110</b>	<b>0.562</b>	<b>0.662</b>	<b>138.543</b>	<b>0.218</b>	<b>25.112</b>	<b>0.164</b>	<b>0.253</b>	<b>0.294</b>
Improvement	6.874%	6.481%	9.809%	5.172%	7.456%	10.887%	6.228%	3.964%	3.315%	8.379%	15.525%	5.755%
Improvement*	7.699%	5.164%	7.881%	5.982%	8.914%	8.169%	6.821%	7.234%	4.665%	6.818%	14.479%	12.213%

Table 6: Accuracy on travel time estimation and path ranking for long paths

Methods	Aalborg						Xi'an					
	Travel Time Estimation			Path Ranking			Travel Time Estimation			Path Ranking		
	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑	MAE ↓	MARE ↓	MAPE ↓	MAE ↓	τ ↑	ρ ↑
Node2vec [16]	193.021	0.295	25.416	0.231	0.019	-0.008	262.160	0.279	28.294	0.171	0.015	0.108
PIM [3]	135.348	0.207	17.855	0.098	0.210	0.239	246.606	0.246	24.981	0.186	0.179	0.207
Lightpath [45]	114.599	0.175	16.079	0.080	0.371	0.326	241.666	0.237	24.752	<u>0.107</u>	0.259	0.317
TrajCL [5]	99.041	0.151	14.215	0.077	0.346	0.372	243.581	0.238	24.951	0.111	<u>0.266</u>	<u>0.328</u>
START [21]	87.890	<u>0.136</u>	13.220	0.078	0.328	0.364	<u>237.245</u>	<u>0.230</u>	<u>23.973</u>	0.108	0.244	0.301
CLIP [37]	174.421	0.266	23.065	0.126	0.145	0.172	255.160	0.249	28.752	0.208	0.093	0.104
USPM [8]	147.236	0.232	24.435	0.105	0.201	0.218	243.380	0.240	24.549	0.157	0.216	0.277
JGRM [29]	89.125	0.138	<u>13.131</u>	<u>0.076</u>	<u>0.381</u>	<u>0.379</u>	239.386	0.234	24.379	0.109	0.265	0.319
Lightpath+image	111.703	0.171	<u>15.897</u>	0.092	0.326	0.308	243.581	0.238	24.951	0.112	0.236	0.323
START+image	99.041	0.151	14.215	0.083	0.317	0.323	237.347	0.232	24.641	0.113	0.258	0.312
<b>MM-Path</b>	<b>83.125</b>	<b>0.121</b>	<b>11.214</b>	<b>0.067</b>	<b>0.421</b>	<b>0.424</b>	<b>224.227</b>	<b>0.190</b>	<b>21.832</b>	<b>0.103</b>	<b>0.279</b>	<b>0.345</b>
Improvement	5.421%	11.029%	15.174%	12.987%	13.477%	13.978%	3.833%	17.391%	8.931%	3.738%	2.307%	5.182%
Improvement*	6.732%	12.318%	14.599%	11.842%	10.499%	11.873%	5.527%	18.103%	10.447%	5.504%	5.283%	7.836%

ranking. We observe that the performance of both models improves with an increase in labeled data size, and the pre-trained model consistently outperforms the no pre-trained model. This illustrates that the pre-trained model, equipped with extensive cross-modal context information, requires less labeled data and achieves superior performance compared to the model without pre-training. These findings suggest that MM-Path can effectively serve as a pre-training model to enhance supervised learning methods.

**4.2.4 Parameter Sensitivity.** We explore the impact of image granularity size on the model's performance. For uniform segmentation, each  $500 \times 500$  pixel image is segmented into  $1 \times 1$ ,  $2 \times 2$ ,  $4 \times 4$ ,  $5 \times 5$ , and  $10 \times 10$  patches, respectively. The performance and testing runtime for each granularity size are detailed in Figure 6.

We observe that a larger number of patches also incurs longer inference runtime for each path. Moreover, the model's performance improves as the number of patches increases from  $1 \times 1$  to  $4 \times 4$ , suggesting that finer granularity enhances the capture of detailed features. However, performance begins to decline with further increases to  $5 \times 5$  and  $10 \times 10$  patches. This decrease is due to the limited context features extracted by excessively fine granularity, which negatively impacts path understanding.

Then, we explore the model scalability in terms of road path length. We evaluate the performance of all models on paths with varying numbers of nodes. Specifically, paths with fewer than 50 nodes are classified as short paths, while those with more than 50 nodes are classified as long paths. The performance of all models on short and long paths is detailed in Tables 5 and 6, respectively. The experimental results show that MM-Path outperforms other models across both path lengths, demonstrating its superiority.

**4.2.5 Case Study.** We inspect a pair of representative paths in Aalborg to demonstrate the superiority of MM-Path. The road paths and image paths are visualized in Figure 7. The travel time estimation results of MM-Path and the superior baselines are shown in Table 7.

Two paths in Figure 7 exhibit a similar structure on the road network, both having a node degree sequence of  $\langle 3, 2, 3, 3, 3 \rangle$ . Such single-modal data might suggest that these paths have comparable travel times. However, the visual information from their images differs significantly. Specifically, path 1 traverses a roundabout and runs along a trunk road, where paths typically allow for faster travel speeds. In contrast, path 2 is located on an ordinary road near residential buildings, typically associated with slower speeds.

As shown in Table 7, the travel time estimates from TrajCL, START, and START+image suggest a shorter travel time for path 2 and a longer travel time for path 1, which are contrary to the ground truth. This illustrates that a single modality provides limited information, and the simple concatenation of multi-modal data in the START+image model fails to effectively extract image information. In contrast, the results from the multi-modal model JGRM align with the relative magnitudes of the actual travel times, but its estimates exhibit large deviations. Meanwhile, MM-Path demonstrates superior travel time estimation performance compared to other models, indicating its effective fusion and utilization of image information.

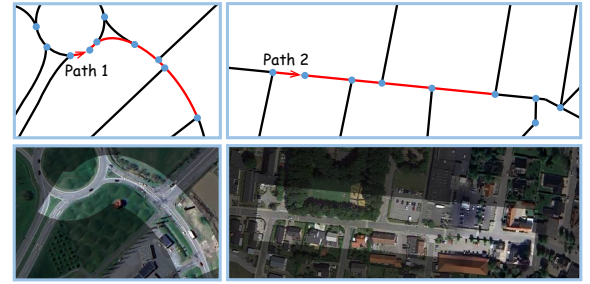


Figure 7: Visualization of two paths

Table 7: Travel time estimation results of different models

Case	Ground truth	Travel Time Estimation				
		MM-Path	TrajCL	START	JGRM	START+image
Path 1	17.0	22.7	30.0	32.9	26.9	29.2
Path 2	36.0	40.1	17.1	24.3	73.8	26.5

## 5 Related work

### 5.1 Path Representation Learning Models

With the advancement of location-based services, the trajectory generation [51, 52] and trajectory analysis [6, 10, 13, 27, 28] have become increasingly prevalent. To better understand static representation of trajectories, recent studies have increasingly focused on developing path representation learning models that do not rely on labeled training data, showing robust generalization across multiple downstream tasks [5, 21, 29, 45]. For instance, Jiang et al. [21] introduce a self-supervised trajectory representation learning framework that includes tasks such as span-masked trajectory recovery and trajectory contrastive learning to leverage temporal patterns and travel semantics effectively. Yang et al. [45] aim to minimize resource consumption and enhance model scalability by developing LightPath, a lightweight and scalable framework designed to conserve resources while maintaining accuracy. Additionally, Ma et al. [29] propose a representation learning framework that integrates GPS and route modeling based on self-supervised technology, further expanding the field's methodologies.

Recent advancements in Large Language Models (LLMs) [38, 39] have facilitated the development of general spatio-temporal prediction models [23–25, 48, 50]. For example, Unist [48] maps spatio-temporal data onto grids and provides generic predictions across various scenarios using elaborated masking strategies and spatio-temporal knowledge-guided prompts. However, since each path exhibits geographical continuity and cannot be discretized into a single region, these models are not well-suited for path modeling.

Despite recent advancements, current path representation learning models overlook the potential contributions of images in path understanding, which can provide valuable insights into the geometric features and contextual environmental information from a global perspective.

## 5.2 Multi-modal Pre-trained Models

Multi-modal pre-trained models aim to enhance target representation by leveraging diverse modalities, including text, images, and audio. Some methodologies converge the information from various modalities into a unified latent space [37, 41]. For example, Wang et al. [41] propose an unsupervised multi-modal method that encodes visual, textual, and geospatial data of urban neighborhoods into a unified vector space using multiple triplet loss functions. Radford et al. [37] develop CLIP, a large-scale multi-modal model that evaluates the congruence of image-text pairs through cosine similarity. Some research opts to map each modality into distinct embedding spaces, while enforcing fusion on the representations [2, 36, 37]. For instance, Pramanick et al. [36] achieve robust video-text representation by embedding cross-modal fusion within the core video and language structures, thereby facilitating various downstream tasks and reducing fine-tuning requirements. Bao et al. [2] present VLMO, a unified vision-language pre-trained model that jointly trains a dual encoder and a fusion encoder within a modular Transformer network.

Nevertheless, these models are predominantly trained on general corpora. The characteristics of road paths and image paths, which include complex correspondences and spatial topological relationships, are distinct from those found in conventional multi-modal datasets. Consequently, existing multi-modal models exhibit limited generalizability to this specialized domain.

## 6 Conclusion and future work

In this paper, we propose a Multi-modal Multi-granularity Path Representation Learning Framework (MM-Path), which is the first work that integrate data from road networks and remote sensing images into generic path representation learning. Initially, we model the road paths and image paths separately, and implement a multi-granularity alignment strategy to ensure the synchronization of both detailed local information and broader global context. Furthermore, we develop a graph-based cross-modal residual fusion component that effectively fuses information from both modalities while preserving the semantic consistency between modalities. MM-Path outperforms all baselines on two real-world datasets across two downstream tasks, demonstrating its superiority.

In the future, we plan to further investigate the capability of multi-modal models for generic path representing learning, with particular focus on few-shot and zero-shot learning scenarios.

## References

- [1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2021. BEiT: BERT pre-training of image Transformers. In *ICLR*.
- [2] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. 2022. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *NeurIPS* 35 (2022), 32897–32912.
- [3] Sean Bin Yang, Chenjuan Guo, Jilin Hu, Bin Yang, Jian Tang, and Christian S. Jensen. 2022. Weakly-supervised temporal path representation learning with contrastive curriculum learning. In *ICDE*. 2873–2885.
- [4] Shaosheng Cao, Wei Lu, and Qiongkai Xu. 2015. Grarep: Learning graph representations with global structural information. In *CIKM*. 891–900.
- [5] Yanchuan Chang, Jianzhong Qi, Yuxuan Liang, and Egemen Tanin. 2023. Contrastive trajectory similarity learning with dual-feature attention. In *ICDE*. IEEE, 2933–2945.
- [6] Lu Chen, Yunjun Gao, Ziquan Fang, Xiaoye Miao, Christian S Jensen, and Chenjuan Guo. 2019. Real-time distributed co-movement pattern detection on streaming trajectories. *PVLDB* 12, 10 (2019), 1208–1220.
- [7] Lu Chen, Qilu Zhong, Xiaokui Xiao, Yunjun Gao, Pengfei Jin, and Christian S Jensen. 2018. Price-and-time-aware dynamic ridesharing. In *ICDE*. IEEE, 1061–1072.
- [8] Meng Chen, Zechen Li, Weiming Huang, Yongshun Gong, and Yilong Yin. 2024. Profiling urban streets: A semi-supervised prediction model based on street view imagery and spatial topology. In *KDD*. 319–328.
- [9] Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. 2024. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *ICLR*.
- [10] Yile Chen, Gao Cong, and Cuauhtemoc Anda. 2023. Teri: An effective framework for trajectory recovery with irregular time intervals. *PVLDB* 17, 3 (2023), 414–426.
- [11] Yile Chen, Xiucheng Li, Gao Cong, Zhifeng Bao, Cheng Long, Yiding Liu, Arun Kumar Chandran, and Richard Ellison. 2021. Robust road network representation learning: When traffic patterns meet traveling semantics. In *CIKM*. 211–220.
- [12] Jian Dai, Bin Yang, Chenjuan Guo, and Zhiming Ding. 2015. Personalized route recommendation using big trajectory data. In *ICDE*. IEEE, 543–554.
- [13] Xin Ding, Lu Chen, Yunjun Gao, Christian S Jensen, and Hujun Bao. 2018. UITraMan: A unified platform for big trajectory data management and analytics. *PVLDB* 11, 7 (2018), 787–799.
- [14] Ahmed Elbery, Hossam S Hassanein, Nizar Zorba, and Hesham A Rakha. 2020. Iot-based crowd management framework for departure control and navigation. *IEEE Trans. Veh. Technol.* 70, 1 (2020), 95–106.
- [15] Noel Gorelick, Matt Hancher, Mike Dixon, Simon Ilyushchenko, David Thau, and Rebecca Moore. 2017. Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sens. Environ.* (2017).
- [16] Aditya Grover and Jure Leskovec. 2016. Node2vec: Salable feature learning for networks. In *KDD*. 855–864.
- [17] Chenjuan Guo, Christian S. Jensen, and Bin Yang. 2014. Towards total traffic awareness. *SIGMOD Record* 43, 3 (2014), 18–23.
- [18] Chenjuan Guo, Ronghui Xu, Bin Yang, Ye Yuan, Tung Kieu, Yan Zhao, and Christian S Jensen. 2024. Efficient stochastic routing in path-centric uncertain road networks. *PVLDB* 17, 11 (2024), 2893–2905.
- [19] Chenjuan Guo, Bin Yang, Jilin Hu, and Christian Jensen. 2018. Learning to route with sparse trajectory sets. In *ICDE*. IEEE, 1073–1084.
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
- [21] Jiawei Jiang, Dayan Pan, Houxing Ren, Xiaohan Jiang, Chao Li, and Jingyuan Wang. 2023. Self-supervised trajectory representation learning with temporal regularities and travel semantics. In *ICDE*. IEEE.
- [22] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*. 4171–4186.
- [23] Duc Kieu, Tung Kieu, Peng Han, Bin Yang, Christian S. Jensen, and Bac Le. 2024. TEAM: Topological evolution-aware framework for traffic forecasting. *PVLDB* 18 (2024).
- [24] Zhonghang Li, Lianhao Xia, Jiabin Tang, Yong Xu, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024. UrbanGPT: Spatio-Temporal Large Language Models. *arXiv:2403.00813 [cs.CL]*
- [25] Zhonghang Li, Lianhao Xia, Yong Xu, and Chao Huang. 2024. FlashST: A simple and universal prompt-tuning framework for traffic prediction. *arXiv preprint arXiv:2405.17898* (2024).
- [26] Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. 2024. Foundations & trends in multimodal machine learning: Principles, challenges, and open questions. *ACM Comput. Surv.* 56, 10 (2024), 1–42.
- [27] Ziqiao Liu, Hao Miao, Yan Zhao, Chenxi Liu, Kai Zheng, and Huan Li. 2024. LightTR: A lightweight framework for federated trajectory recovery. *arXiv preprint arXiv:2405.03409* (2024).
- [28] Yandi Lun, Hao Miao, Jiaxing Shen, Renzhi Wang, Xiang Wang, and Senzhang Wang. 2024. Resisting tul attack: balancing data privacy and utility on trajectory via collaborative adversarial learning. *GeoInformatica* 28, 3 (2024), 381–401.
- [29] Zhipeng Ma, Zheyuan Tu, Xinhai Chen, Yan Zhang, Deguo Xia, Guyue Zhou, Yilun Chen, Yu Zheng, and Jiangtao Gong. 2024. More than routing: Joint GPS and route modeling for refine trajectory representation learning. In *WWW*. 3064–3075.
- [30] Sachit Menon and Carl Vondrick. 2022. Visual classification via description from large language models. In *ICLR*.
- [31] Paul Newman and John Krumm. 2009. Hidden markov map matching through noise and sparseness. In *SIGSPATIAL*. 336–343.
- [32] Zhicheng Pan, Yihang Wang, Yingying Zhang, Sean Bin Yang, Yunyao Cheng, Peng Chen, Chenjuan Guo, Qingsong Wen, Xiduo Tian, Yunliang Dou, et al. 2023. Magicscaler: Uncertainty-aware, predictive autoscaling. *PVLDB* 16, 12 (2023), 3808–3821.
- [33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library.
- [34] Simon Aagaard Pedersen, Bin Yang, and Christian S. Jensen. 2020. Anytime stochastic routing with hybrid learning. *PVLDB* 13, 9 (2020), 1555–1567.

- [35] Simon Aagaard Pedersen, Bin Yang, Christian S. Jensen, and Jesper Møller. 2023. Stochastic routing with arrival windows. *ACM Trans. Spatial Algorithms Syst.* 9, 4 (2023), 30:1–30:48.
- [36] Shraman Pramanick, Yale Song, Sayan Nag, Kevin Qinghong Lin, Hardik Shah, Mike Zheng Shou, Rama Chellappa, and Pengchuan Zhang. 2023. EgoVLPv2: Egocentric video-language pre-training with fusion in the backbone. In *ICCV*. 5285–5297.
- [37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*. 8748–8763.
- [38] Yi Tay, Mostafa Dehghani, Vinh Q Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, et al. 2022. UL2: Unifying language learning paradigms. In *ICLR*.
- [39] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239* (2022).
- [40] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. 2023. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *CVPR*. 19175–19186.
- [41] Zhecheng Wang, Haoyuan Li, and Ram Rajagopal. 2020. Urban2vec: Incorporating street view imagery and pois for multi-modal urban neighborhood embedding. In *AAAI*, Vol. 34. 1013–1020.
- [42] Ronghui Xu, Weiming Huang, Jun Zhao, Meng Chen, and Liqiang Nie. 2023. A spatial and adversarial representation learning approach for land use classification with POIs. *ACM Trans. Intell. Syst. Technol.* 14, 6, Article 114 (nov 2023), 25 pages.
- [43] Bin Yang, Jian Dai, Chenjuan Guo, Christian S Jensen, and Jilin Hu. 2018. PACE: a PAth-Centric paradigm for stochastic path finding. *VLDB J.* 27 (2018), 153–178.
- [44] Sean Bin Yang, Chenjuan Guo, and Bin Yang. 2020. Context-aware path ranking in road networks. *IEEE Trans. Knowl. Data Eng.* 34, 7 (2020), 3153–3168.
- [45] Sean Bin Yang, Jilin Hu, Chenjuan Guo, Bin Yang, and Christian S. Jensen. 2023. LightPath: Lightweight and scalable path representation learning. In *KDD*. ACM, 2999–3010.
- [46] Sean Bin Yang and Bin Yang. 2020. Learning to rank paths in spatial networks. In *ICDE*. IEEE, 2006–2009.
- [47] Ziyi Yang, Yuwei Fang, Chenguang Zhu, Reid Pryzant, Dongdong Chen, Yu Shi, Yichong Xu, Yao Qian, Mei Gao, Yi-Ling Chen, et al. 2023. i-code: An integrative and composable multimodal learning framework. In *AAAI*, Vol. 37. 10880–10890.
- [48] Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. 2024. UniST: A prompt-empowered universal model for urban spatio-temporal prediction. In *KDD*.
- [49] Sen Zhang, Senzhang Wang, Xiang Wang, Shigeng Zhang, Hao Miao, and Junxing Zhu. 2022. Multi-task adversarial learning for semi-supervised trajectory-user linking. In *ECML PKDD*. Springer, 418–434.
- [50] Kai Zhao, Chenjuan Guo, Yunyao Cheng, Peng Han, Miao Zhang, and Bin Yang. 2023. Multiple time series forecasting with dynamic graph modeling. *PVLDB* 17, 4 (2023), 753–765.
- [51] Yuanshao Zhu, Yongchao Ye, Shiyao Zhang, Xiangyu Zhao, and James Yu. 2023. Difftraj: Generating gps trajectory with diffusion probabilistic model. *NeurIPS* 36 (2023), 65168–65188.
- [52] Yuanshao Zhu, James Jianqiao Yu, Xiangyu Zhao, Qidong Liu, Yongchao Ye, Wei Chen, Zijian Zhang, Xuetao Wei, and Yuxuan Liang. 2024. Controltraj: Controllable trajectory generation with topology-constrained diffusion model. In *KDD*. 4676–4687.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009