Mining Spatio-Temporal Attributes of Anomalies through Large Ego-Vehicle Dataset

Tiffany Ma

May 2022

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Srinivasa Narasimhan
Christoph Mertz
Stephen Smith

Submitted in partial fulfillment of the requirements for the degree of Master of Science.

Abstract

In recent years, an increasing amount of urban visual big data is collected through a diverse range of sources, such as taxi vehicle records, video from surveillance cameras, or images captured by mobile devices. The large collection of urban data contains rich implicit information that can help numerous downstream tasks, such as monitoring for construction management companies, planning for government units, etc. However, it is challenging to efficiently extract the desired information from a large-scale dataset. In this work, we focus on developing methods for extracting the spatial attribute and the temporal attribute from urban visual data. Specifically, we introduce a method of organizing large-scale urban visual data into a spatial-temporal structure by mining attributes inherent in the data. We demonstrate the effectiveness of our method by using videos captured by the front-facing camera of buses to detect and analyze work zones within the captured videos. First, the raw set of bus data is preprocessed into a spatial-temporal data structure. Next, we exploit the rich spatial and temporal attributes of bus data in the application of work zone detection and analysis. The goal of this work is to demonstrate the effectiveness of using spatial and temporal attributes to break down large-scale urban visual data and extract insights from large-scale unlabeled data.

Acknowledgments

First and foremost, I would like to thank my advisors, Professor Srinivasa Narasimhan and Dr. Christoph Mertz. Their guidance and support throughout this journey are invaluable for my study, research, and personal growth. I would also like to thank Professor Stephen Smith for being on my thesis committee.

I have received many support from talented individuals in Srinivasa's and Christoph's research group. Specifically, I would like to thank Tom Bu, Anurag Ghosh, Robert Tamburo, and Khiem Vuong for such kind guidance and immense inspiration.

I could not have thanked my family and friends more for their immense care and support throughout this journey. The care, love, and encouragement of my parents always motivated me. I would like to give special thanks to my friends Ashley Wu, Ting Wu, Erin Zhang, and Yi-yu Zhang, for their company. I would like to give special thanks to Ching-Yi Lin, who inspired me and taught me so much about research.

Contents

Chapter 1

Introduction

In recent years, an increasing amount of urban visual data is collected through a diverse range of sources, such as vehicle recordings from taxis, videos from surveillance cameras, or images captured by mobile devices. The large collection of urban data contains rich implicit information that can help numerous downstream tasks. For example, construction management companies can use videos near the construction site to monitor its progress. Government units can improve the infrastructure of the city by analyzing the traffic status of commuters. Autonomous vehicle teams can also use large-scale vehicle data to design algorithms that better adapt to realistic road environments. Although large-scale urban visual data contain rich information for many downstream tasks, analyzing and excavating the value of these big data is a significant challenge [18]. Sources such as surveillance cameras and vehicle recorders continuously collect data over long periods of time. This accumulates up to terabytes or petabytes of unlabeled raw data. The amount of raw data makes it difficult to extract points of interest from the large pool of data. Another challenge is the lack of structure in the raw data forms. Each task needs to preprocess the raw data into structures that highlight the desired properties.

Spatial and temporal properties are commonly observed in urban visual data. For example, given a video recording of a taxi that drove through some spatial region, we can infer the boundaries of the traffic region by analyzing the density of cars in each frame of the video. In another example, given a surveillance camera that is facing a parking lot, by analyzing changes at different hours, we can identify the hours at which a parking lot is busiest. Analyzing spatial and temporal attributes enables us to extract interesting events from large-scale urban visual data.

In this work, we demonstrate the effectiveness of using spatial and temporal attributes to extract interesting events. Specifically, we extract instances of work zones from the bus recordings by exploiting the spatial and temporal properties of the bus. One of the commonly overlooked sources of spatial-temporal data is bus data. For safety and liability, nowadays transit buses have cameras installed to observe the environment around the buses, together with some other sensors such as GPS. These sensors provide rich urban visual data in areas where public transport is widely available. Naturally, buses routinely traverse the same spatial region for a long period of time. Such data can be distilled to construct a dataset of rich spatial-temporal information. To mitigate the challenge of scale and lack of structure, we propose a method to map bus data to a spatial-temporal data structure.

Using this structured data, we want to detect and analyze work zones. Work zones are a

common source of traffic disruption, causing great inconvenience for commuters. Gaining a better understanding of work zones can benefit various downstream applications. Construction groups can use this knowledge to improve the planning of future construction sites. Government units can also use the learned patterns of work zones to make decisions about modifying traffic flows near the work zones. Prediction teams in Autonomous Vehicle companies can apply spatial and temporal relations of work zones to better react to roadside anomalies. The completion time of work zones can range from hours to months. For instance, a highway infrastructure change may take up to months and span long regions of the highway; whereas a local road repainting work typically lasts a few hours and span only a few meters. To fully understand a work zone from start to finish, we should study patterns about a work zone in the temporal dimension. Most of the current works that study work zones are mainly in the domain of planning, safety, and transportation. Only a limited number of them use vision inputs. Part of this is due to the variability of the work zones and the lack of a concrete definition for these sites. In this work, we aggregate the definitions of road construction used in the transportation, vision, and safety community to find a common ground for understanding these sites through a visual modality.

In summary, we are interested in studying methods for mining spatial and temporal attributes in large-scale urban visual data. In this work, we work with a specific source of urban visual data: recordings collected from front-facing cameras of the bus (bus data). The raw set of bus data needs to be further preprocessed into a spatial-temporal structure. Next, we exploit the rich spatial and temporal attributes of bus data in the application of work zone detection and analysis.

The main contributions of this thesis are explicitly stated as follows:

- We propose a structure for organizing bus data based on their spatial and temporal relations, which facilitates more accessible analysis for data with similar structure.
- We manually collect and annotate a dataset for work zone related objects.
- We demonstrate different types of spatial-temporal attributes that are present in work zones detected from bus data and show the potential of such findings.

The remainder of this thesis is organized as follows:

- In Chapter 2, we discuss the values, challenges, and applications of urban big data and how to exploit the spatial and temporal properties of urban big data.
- In Chapter 3, we explain our approach in mapping raw bus data to a spatial-temporal data structure.
- In Chapter 4, we dive into more detail definitions of work zones and approaches to understanding work zones through visual input.
- In Chapter 5, we share the results of analyzing the identified sections of work zones from the bus data and discuss how this can be generalized to a wider range of applications.

Chapter 2

Background

2.1 Urban Big Data

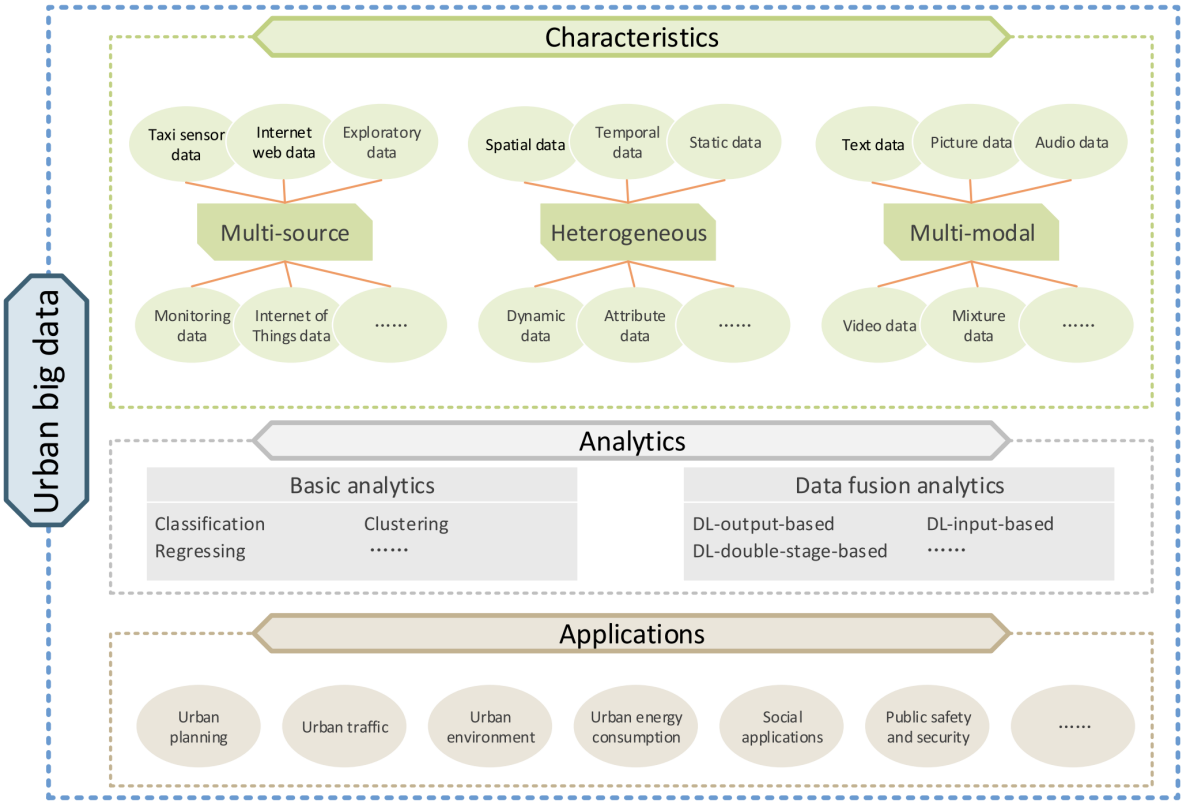


Figure 2.1: This diagram [18] breaks down the definition of urban big data by its characteristics, methods of analysis, and its downstream applications.

Urban big data refers to a combination of structured or unstructured data collected from various urban environments. Zhang et al. [35] summarized the five characteristics of big data as 5Vs, which refers to volume, velocity, variety, veracity, and value. Volume refers to the amount of

data available. Velocity refers to the speed with which data is generated and collected. The latter three metrics may vary from task to task. Variety refers to the diversity of data types. Veracity refers to the quality of the data collected. Last but not least, value refers to the value that this dataset provides. These five features show the key characteristics that big data should have. It also highlights common challenges in analyzing the value of these big data. For example, a large volume of data may contain valuable data, but it is also computationally expensive to parse and analyze this data. As [22] notes, urban big data is very complex, and we only extract a small part of its knowledge.

From Figure 2.1, we see that [18] lists three broad characteristics that urban big data generally holds. First, we observe that urban big data is collected from a wide range of sources. Meng et al. [21] used real-time GPS readings of taxis, road networks from Internet Web data to infer the volume of urban traffic. Second, we see that urban big data exist in many domains, including spatial, temporal, static, and dynamic data [18]. Third, we see that urban big data is multimodal and may include visual, textual, or numeric data. Yi et al. [33] used three datasets to predict air quality, namely, air quality data, weather forecast data, and meteorological data. Weather is represented as textual input (sunny, cloudy, overcast, foggy, etc.), and wind speed is represented as numerical input. With appropriate analysis, urban big data can be used in a wide variety of applications, including urban planning, urban traffic, urban environment, social applications, and public safety and security.

Recently, there has been a rapid increase in the amount of visual data available in urban locations. However, there is still a lack of a structural approach to organizing and understanding such a vast amount of data. In our case, we are specifically interested in visual urban data. These data can be collected from cameras mounted on stationary traffic light poles, front cameras of moving vehicles, and even surveillance cameras.

2.2 Spatial-Temporal Data

According to [5], spatial-temporal data comprises of spatial and temporal representations. Spatial-temporal data contains three distinct types of attributes, which are non-spatial-temporal, spatial, and temporal attributes. Spatial attributes are those related to the location, shape, and physical aspects of the object. Temporal attributes include timestamps and the duration of in-range data. Non-spatial-temporal attributes typically refer to other additional numeric evaluations of aspects that do not fall under spatial or temporal domains [5]. For example, [5] gave the example of air pollution measures. Air pollution levels and name of location are one example of non-spatial-temporal attributes. [13] used spatial-temporal data from 1,904 residential cars to generate a heat map of vehicle mobility in the city during COVID-19. Based on the heat map, they enforced flexible lockdown strategies to reduce population flow within the city. These examples demonstrate that with adequate analysis, spatial-temporal data can provide rich insights for many event. One common example that uses spatial-temporal analysis is the field of traffic and transportation. Traffic data represent spatial-temporal trajectories that are used to discover periodic patterns. Rao et al. [25] observe that one challenge comes from the influence of nearby objects. Examples of such influence are spatial-temporal events, such as accidents, that may affect traffic patterns in irregular ways.

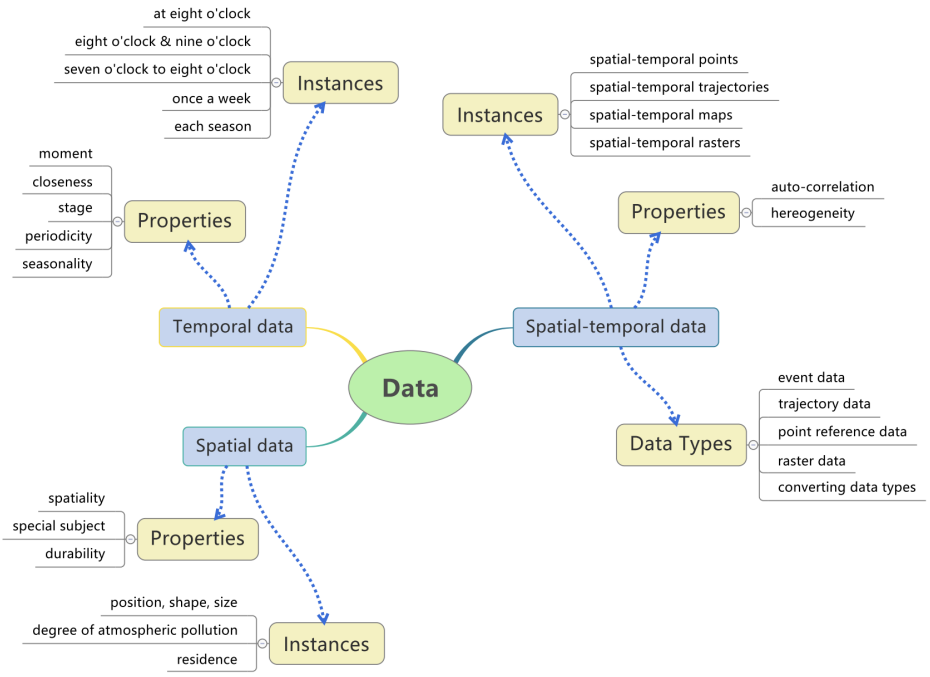


Figure 2.2: Based on the spatial and temporal dimensions, a given data can be divided into data with temporal attribute, data with spatial attribute, and data with both temporal and spatial attributes (spatial-temporal data). [18]

Unlike typical datasets, spatial-temporal data do not follow the assumption that data points are independently and identically distributed. Neighboring spatial-temporal objects share similar characteristics that are often related. Spatial-temporal data can exploit relationships that are usually omitted in normal data distributions. Although spatial-temporal data have the potential of extracting rich insights and relationships, these patterns are challenging to mine for the following reasons. First, spatial-temporal relationships are typically high in complexity. Co-located objects in the spatial and temporal domains may influence each other, making detection of relationships difficult. Second, another difficulty is that these relationships are typically implicitly defined. Non-spatial-temporal data have explicit relationships represented through arithmetic relations, such as ordering, instance of, subclass of, and member of. Spatial relationships are built on the basis of qualities or features such as distance, volume, size, and time. These attributes are expressed in a continuous spectrum. These meanings or representations of these attributes can vary depending on interpretation and context, making it difficult to identify these relationships [5].

Spatial-temporal data mining (STDM) [11] aims to tackle the above challenge. STDM discovers useful patterns from the dynamic interplay between space and time. STDM contains numerous tasks, such as prediction, clustering, hotspot detection, pattern discovery, outlier analysis, visualization, and visual analytics. These tasks are important in different applications, such as understanding the behavior of objects, scenes, and events. STDM pattern mining works on discovering hidden information (occurrences in space and time, such as movement patterns from

trajectories of spatial-temporal objects). Discovering spatial-temporal associations of trajectories is challenging due to long temporal duration, different moving directions, and lack of spatial accuracy.

Many traffic and transportation datasets contain correlated spatial and temporal attributes. Public transportation data fits the exact definition. We know that public transports traverse on a fixed trajectory. Therefore, for each image captured at a location, it is spatially related to nearby images. In addition, buses travel through the same region for a long period of time, adding rich temporal attributes to the collection of images. As [11] observes, modeling trajectory data in a spatial-temporal structure can be challenging. In the latter sections, we describe the design choices made to mitigate these challenges for the purpose of our application.

2.3 Work Zones

Work zones are a common source of traffic disruption, causing great inconvenience to commuters. Gaining a better understanding of work zones can benefit various downstream applications. According to [3], a work zone is an area where road work is carried out and can involve lane closures, detours, and moving equipment. Highway work zones are established according to the type of road and the work to be done on the road. The work zone can be long- or short-term and can exist anytime of the year, but most commonly in the summer. Work zones are expected to follow a set of regularizations to ensure the safety of the workers and nearby vehicles. There are official guidelines for how to set up a work zone [2]. For example, temporary traffic control signs should be placed at some distance before the actual work zone site. Channelizing devices such as cones, vertical panels, and tubular markers should also be placed around actual construction sites.

Barricade	Used to block travel. Consists of horizontal strips often with orange and white stripes (color may vary based on country, city, etc)	
Barrier	Used to block, separate, or channel traffic.	

Temporary Traffic Control (TTC) Sign	Temporarily placed during work period. Usually orange or yellow in the US	
TTC Message Board	Digital sign to provide info	
Arrowboard	Digital sign that uses arrows to direct traffic	
Work Vehicle	Vehicles with specific functions in road work zones, for example, heavy machinery, vehicles with ttc message boards, bucket trucks, etc.	
Guide Sign	Signs used to direct traffic, for example, detour signs	
Tubular Marker	Tube shaped markers used to divide traffic, mark road edges, divert traffic, restrict turns, etc.	
Vertical panel	Rectangular shaped markers used to divide traffic, mark road edges, divert traffic, restrict turns, etc	

Cone	Triangular shaped markers used to divide traffic, mark road edges, divert traffic, restrict turns, etc.	
Fence	Used as a barrier to restrict access to the work area. Temporary meshed fence. Usually metal or plastic	
Worker	Any workers in the work zone. Usually wearing a brightly colored vest and hard hat. Includes flaggers	
Drum	Barrel shaped traffic control device to channel traffic through a work zone or as a warning of nearby road work	

Table 2.1: A table listing work zone related objects.

2.3.1 Building Work Zone Understanding

A work zone generally follows the structure in Figure 2.3 and uses objects in Table 2.1 as delineators. These structures are extremely helpful in understanding construction zones. [15] aims at classifying whether a given scene is a construction site or not using related indicators such as vehicle speed, speed limit on road segment and presence of traffic signs. Each of these indicators is treated as a variable in the Bayesian model and aggregated to produce a probability score of how likely the given scene is a work zone. The goal of this work is to design an online detection pipeline that can handle uncertainty with efficient performance. However, this method does not take into account the spatial information of the indicators. For example, the location of indicator objects in the work zone scene and their relative spatial positioning could provide information on the structure of the work zone.

[28] dives deeper into the understanding aspect of work zones by proposing a geometric

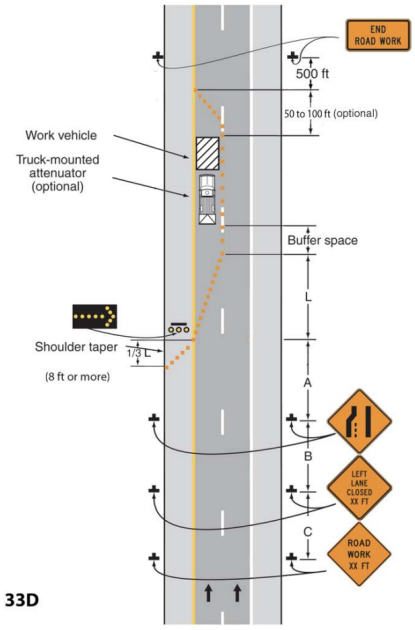


Figure 2.3: An illustration of a short-term stationary setup. The distances A, B, C should be proportional to the work zone speed limit. TTC Signs are placed some distance prior to the actual work zone. Delinators are placed surrounding the boundaries of the work zone. [2]

definition of the boundaries of the work zone. It does so by first considering all detected objects of interest as key points and then lifting the key points onto a bird's eye view plane of the scene. The work zone is mapped onto a contour whose boundaries are defined by the key points on that bird's-eye view plane. The authors of [28] also noted that the use of the RGB input and the LiDAR input produces a more accurate contour. In our work, we will focus on using RGB cameras as our source of visual input, since RGB cameras are more widely available.

Previously, most work related to work zone detection has focused on using speed data and lane markers. There is a limited amount of work in the area of detecting vision-based roadside work zones. This is because there is a lack of publicly available work zone delineator datasets. Large ego-vehicle datasets such as BDD100K [34] and Cityscapes [9] contain minimal to no labeled instances of road construction object data. NuScenes [6] contains labels for some work zone delinators, such as barriers and traffic cones. However, training detectors for these objects faces yet another challenge. For example, compared to common instances, such as vehicles or persons, NuScenes [6] contains a lower number of construction-related instances. This presents the challenge of class imbalance in the NuScenes dataset [6].

2.3.2 Challenges in Work Zones Understanding

Previous datasets focused more on common objects, such as vehicles, pedestrians, and traffic signs. As we begin to build better models to capture these common objects, these datasets begin to expand and include annotations for rarer instances. For example, NuScenes [6] included annotations of traffic cones and barriers in their most recent release of the dataset. In the most

recent release of the Argoverse 2.0 dataset [31], the authors added the construction cone and the construction barrel to their annotation set. The growing amount of data related to road construction is crucial to developing an understanding of work zones. However, to obtain a holistic understanding of these work zones, we need to take into account more related and rarer objects related to the work zone.

2.3.3 Hierarchical Scene Understanding of Work Zones

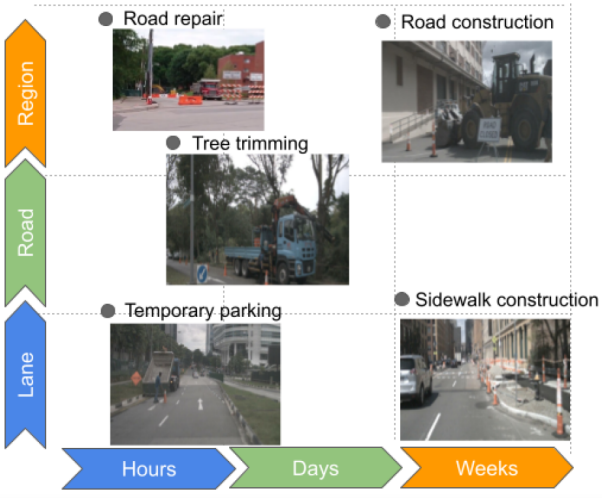


Figure 2.4: Based on the spatial and temporal dimensions, a given data can be divided into data with temporal attribute, data with spatial attribute, and data with both temporal and spatial attributes (spatial-temporal data). [28]

Part of the challenge in understanding work zones comes from the amount of variance on construction sites. These work zones may differ in the delineator used on site, the scale of construction, and the environment in which they are located (urban, suburban, highway). Most scene understanding approaches are divided into two categories: top-down and bottom-up [23]. Top-down approaches look at the scene on a global scale without focusing on specific objects. Bottom-up approaches start with lower-level features, such as objects, and build understanding from the observed object categories and relationships. Understanding work zones can be built on many levels, such as scene level, frame level, and object level. The first two take a top-down approach, while the latter uses a bottom-up methodology. In the following, we will cover different levels of understanding towards construction zones.

As mentioned, road construction zones vary on a variety of scales. Usually, the entire work zone is not fully contained in a single frame. To capture the entire work zone as a whole, we will need to analyze it through a series of neighboring frames that cover the entire work zone. In the work proposed by [28], each work zone is defined by the contoured region on the bird's-eye view plane. Key points detected from a series of neighboring frames are aggregated on the bird's-eye view plane to construct the work zone contour.

At frame level detection, the model loses context from neighboring frames and makes inferences based on a single image. [8] showed efforts to train an image-level classification model from a collection of queried images from the work zone on the road. The authors noticed that the model tends to identify construction zones based on the vibrant orange colors that are commonly seen in construction zones. However, this bias results in many false positives when the same color is visible on non-construction objects. The classification model falsely classified images with similar vibrant, orange color on non-construction objects as a roadside work zone image.

Object level scene understanding is built upon a bottom-up methodology. First, a detection model is trained to detect objects of interest. In this case, the objects of interest are delineators in the work zones. The detection results of each image are extracted and analyzed to gain an understanding of the scene. The spatial and geometric relationships between the detected objects can provide information about the image category.

2.4 Object Detection

In this work, we focus on exploring bottom-up approaches to the scene understanding task. As previously mentioned, bottom-up approaches use detectors to identify key points or key objects, then extract semantics about the scene from the the set of key points. In the following sections, we provide an overview of previous object detection approaches, challenges in object detection, and how it relates to the task of work zone detection.

CNN-based approaches are one of the dominant object detection solutions. Object detection approaches can be broadly split into two categories: two-stage approaches and one-stage approaches [32]. The first line of work, pioneered by R-CNN [10], takes on a coarse-to-fine architecture. First, it generates a class-agnostic region proposals of potential objects and then refines and classifies the proposals into different categories. Faster R-CNN [26] eliminates selective search by introducing the Region Proposal Network (RPN), making it the first end-to-end and near-realtime deep learning detector. With the recent success in attention modules, the vision Transformer architecture [30] has also shown promising results in object detection task. One of the current state of the art is the Swin Transformer [19]. Swin Transformer uses a hierarchical Transformer whose representation is computed with Shifted windows.

As mentioned above, one of the biggest challenges towards training a work zone object detector comes from the lack of dataset in this domain. To the best of our knowledge, there is no dataset that contains a majority of the objects above. There are efforts to expand the annotations toward some of the work zone labels. For example, in the most recent release of Argoverse 2.0 [31], the authors added annotations for construction cones and construction barrels. Similarly, [6] also contains annotations for construction cones and barricades. However, annotations across datasets cannot be easily combined in training. [36] notes that the joining of two datasets may introduce conflicts in their joined annotation. This calls for a stronger need for a specialized dataset that focuses on work zone related objects.

We noticed that most objects appear at work zones at different frequencies and counts. For example, channelizing devices, such as construction cones and vertical panels appear in most construction zones and in large quantities. On the other hand, work vehicles are generally only present at larger scale work zones. This introduces an imbalance between the quantities of each

object, which introduces a class imbalance in the detection model. Class imbalance has been a long-time challenge for detection models. A few ways to combat class imbalance is to apply weighted sampling or weighted loss that is inversely proportional to the quantity of the class label in the training set [14].

Within the domain of work zone objects, there lie many challenges that may be observed. Many of these objects share a similar feature of having a bright orange color and reflective stripes. Since most of these objects share similar salient features, it becomes easy for the detection model to become confused about these work zone related objects.

Another common example is when our objects of interest are subclasses of common objects. For instance, the class "worker" is a specific instance of "person". Some of the instances of a "work vehicle" also fall under the broader category of "vehicle". This makes categorizing these challenging, because "person" and "vehicle" also appear frequently in most ego-vehicle datasets. This requires the model to learn to differentiate between these two instances in the situation where there are fewer instances of our interest.

Chapter 3

Structure of Bus Data

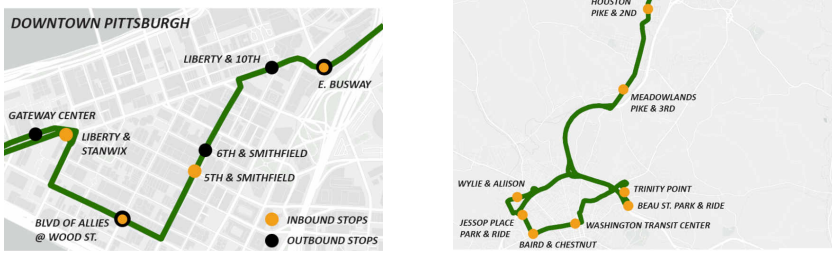
In this work, we are interested in exploiting spatial and temporal attributes within large-scale urban visual data to detect regions of interest. Specifically, we structure unprocessed bus data based on their spatial and temporal attributes to mine patterns about work zones. As mentioned above, visual data from public transit systems is a good source to study work zones for many reasons. The bus traverses on fixed and repeated routes. As such, this allows us to track changes across specific spatial regions of the route over time. This property is especially suitable to mine patterns about work zones, since long-term work zones span a long period of time at a fixed location.

Previous work used bus data for statistical analysis [24], [27], [29]. However, most of these works focus on attributes directly related to the transportation task, such as the number of stops visited, the time elapsed at each stop, the number of pedestrians on board, etc. A limited amount of work places the focus on using visual data collected from outfacing cameras from the bus to perform analysis of its surroundings. In this work, we will use data collected through the BusEdge [32] system. In the following sections, we will give a brief overview of the data collection process and the modalities of the data. Next, we will cover how the bus data is organized into aligned trajectories.

3.1 Data Source and Collection Process

The data used in this work are collected from a bus that traveled between downtown Pittsburgh, PA and Washington County, PA. Two round trips are collected every work day. The route traversed is drawn in Figure 3.1. The inbound trip starts at the intersection of Wylie and Allison Avenues and ends at East Busway in downtown Pittsburgh. The outbound trip starts on the East Busway in downtown Pittsburgh and ends at the Washington Transit Center.

In total, five cameras are installed on the bus. Four waterproof exterior cameras are placed at the top corners of the bus, with two side cameras looking backward and two rear cameras looking forward. The last camera is an interior camera placed behind the windshield of the bus. The technical specifications of the cameras are listed in Table 3.1 and Table 3.2. In addition to the five RGB cameras, the bus is also equipped with a Mobile Mark's LTM501 Series Multiband MIMO antenna. This antenna has five built-in antennas, one of them being a GPS antenna. Data



(a) Downtown Pittsburgh

(b) Washington Country to Downtown Pittsburgh

Figure 3.1: Visualization of the trajectory taken by the bus [1]

Figure 3.2: Pictures of the Transit Bus and its Exterior Cameras [32]

Brand	Safety Vision
Model	43 series IP camera
Highest Resolution	1920 × 1080
Focal Length	2.8mm

Table 3.1: Specification of Interior Camera

Brand	Safety Vision
Model	37 series IP camera
Highest Resolution	1920 × 1080
Focal Length	2.8mm, 4.0mm

Table 3.2: Specifications of Exterior Cameras

are collected continuously through the bus system.

3.2 Raw Data Definition

Our data source traverses two endpoints: East busway in downtown Pittsburgh and East Chestnut Street Transit Center. We define a bus run R as a sequence of data points that traverse between the end points. Each run travels in either the inbound or outbound direction. Next, we sort the col-

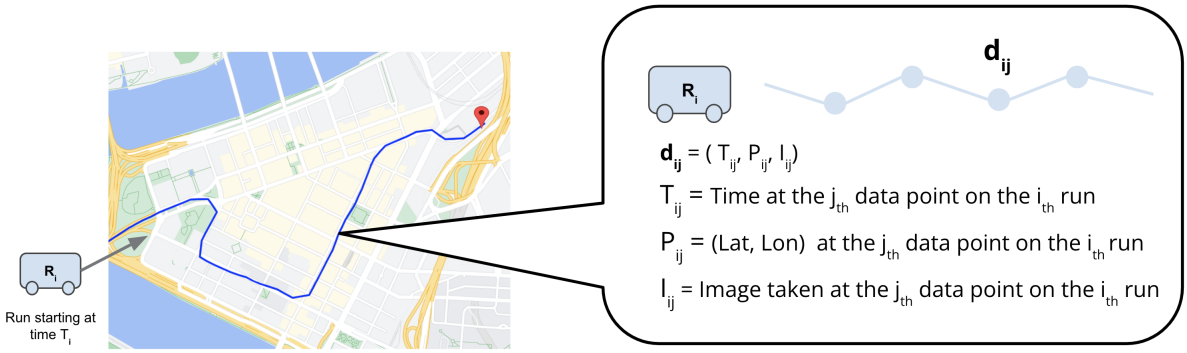


Figure 3.3: Illustration of the data points collected along the i_{th} run of the bus

lection of runs by the starting time of that run and enumerate the runs. Let R_i be the i_{th} run in the collection. Each run is a sequence of raw data instances, denoted as $R_i = [d_{i,0}, ..., d_{i,j}, ..., d_{i,n}]$. Let d_{ij} be the j_{th} raw data instance from the i -th run. $d_{i,j} = (c_{i,j}, t_{i,j}, I_{i,j})$, where $c_{i,j} \in \mathbb{R}^2$ is the latitude and longitude coordinates of the data instance, $t_{i,j} \in \mathbb{N}$ represents the time in the unix timestamp, and $I_{i,j}$ is the RGB image.

Our goal is to detect changes in work zones from the collection of runs. Specifically, we are looking for changes along the spatial or temporal axis. To observe changes in these axes, we should first define an ordering for the collection of runs along the temporal and spatial axes. Based on the above definition, we see that a natural derivation of the temporal axis is the indices of the enumerated runs since the collections of runs are sorted by the starting time of that run. We will derive the spatial axis in the next few sections. In the following sections, we will formulate the bus data as a series of trajectories and cover algorithms available to cluster and subsample these points.

3.3 Data Cleaning

Based on prior knowledge of the bus trajectories, we expect the collected data points to be continuous and the trajectories to be coarsely aligned. However, data collected in the wild do not always adhere to this assumption. There are rare instances where there are missing images or GPS coordinates in the sequence of data points. This could be caused by hardware malfunctions and should be filtered out.

From the bus, images are taken at five frames per second continuously. We perform preprocessing to remove low-quality images. For example, at low illumination scenarios (e.g. before sunrise), the images collected tend to be overly blurry. When the bus is stationary (e.g. waiting for the red light), the collected images tend to be highly similar to its neighbors. We remove these instances by thresholding to some blur and duplicate scores. For each data instance $d_{i,j}$ from run R_i, we filter out images whose $blur\_score$ and dup_score are above some predefined threshold.

$$blur\_score = blur(I_{i,j})$$
$$dup\_score = dup(I_{i,j-1}, I_{i,j}) \text{ with } j > 0$$

Here, $blur(I)$ is a measure of the variance of the Laplacian in the image I. For blurry images, the variance is expected to be low. $dup(I_{j-1}, I_j)$ measures the pixel-wise L_1 distance of the two images. Repeating imags are expected to have a low L_1 distance measure. To ensure the quality of the captured images, in the following experiments, we only used images taken during the day (between 9 am and 5 pm).

(a) Blurred Image (b) Duplicate Image at time t (c) Duplicate Image at time $t+1$

Figure 3.4: (a) Images taken before sunrise tends to result in high-blur images. (b) and (c) is a pair of consecutive frames that are taken when the vehicle is stationary. This will generate a pair of images that are close duplicates of one another.

3.4 Align Images using Spatial Coordinates

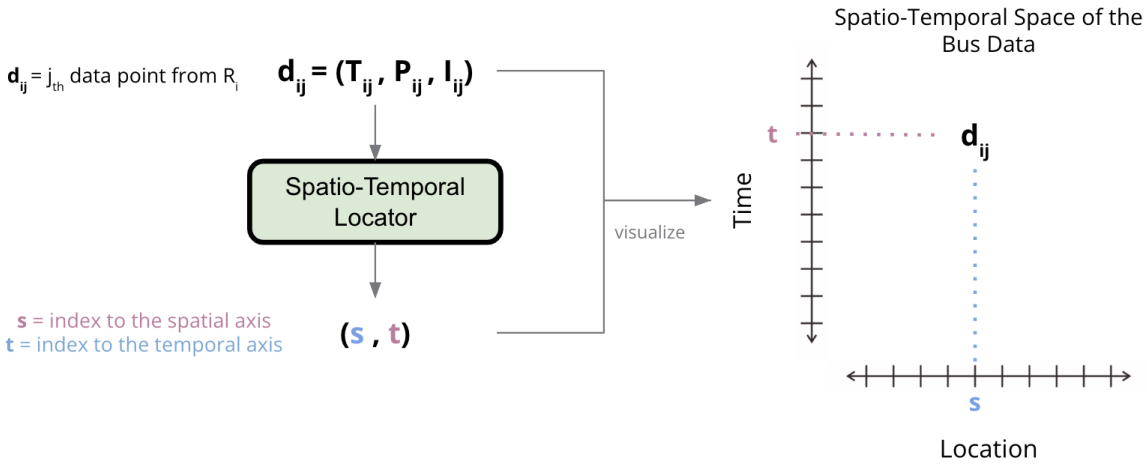


Figure 3.5: Overview of the alignment process: for each data point d_{ij}, we map it to a location on the discretized 2D space formed by a spatial and temporal axis.

By nature, the bus traverses on a repeated path. However, we do not have a set of sample trajectory points that correspond to the repeated path. In addition, there are situations where, on certain days, the bus path deviates from the set routes. For example, the weekday and weekend routes traverse between different points. Another example is that when part of the road is

blocked, there may be a small detour in that section. The direct alignment of the bus data may not be adapted to these changes. Thus, we generate a trajectory baseline by clustering and sampling points from the aggregation of trajectories at different start times and produce a mean representational path. Our goal is to extract a sequence of $K \subseteq \mathbb{R}^2$ coordinates as a representation of the average path that the bus data traverse.

3.4.1 Generating a Baseline Trajectory

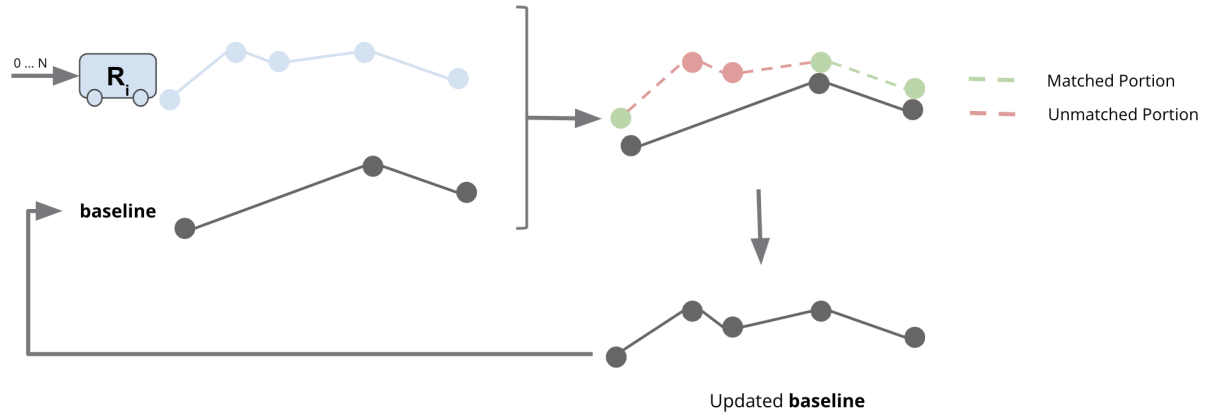


Figure 3.6: The collection of bus runs are iteratively clustered into a dense baseline trajectory.

The work [4] organized a set of algorithms for map construction. They defined map construction as a task that automatically produces or updates street map datasets using vehicle tracking data. This matches our purpose of building a street map from bus tracking data. Map construction algorithms can be organized into three categories: point clustering, incremental track insertion, and intersection linking. Specifically, our data structure best fits the idea of incremental track insertion. Incremental track insertion uses ideas from map matching, where the tracks are clustered and refined based on a rough baseline. This fits our use case, since bus trajectories adhere to the assumption that most of the trajectories are roughly aligned on most of the segments.

We used the implementation of the Cao and Krumm's [7] incremental track insertion algorithm. This incremental track insertion approach proceeds in two stages. In the first stage, a simulation of physical attraction is used to modify the input tracks to group portions of the tracks that are similar together. This results in a cleaner data set in which track clusters are more pronounced and different lanes are more separated. Then, these much cleaner data are used as the input for a fairly simple incremental track insertion algorithm. This algorithm makes local decisions based on distance and direction to insert an edge or vertex and either merge the vertex into an existing edge or add a new edge and vertex. From this algorithm, we now have a set of dense coordinates, K, which contain the sample points of the original runs.

3.4.2 Downsampling the Baseline Trajectory

One challenge in working with spatial-temporal data comes from the large amount of data. Unlike other independently and identically distributed datasets, time series data lies on a continuous

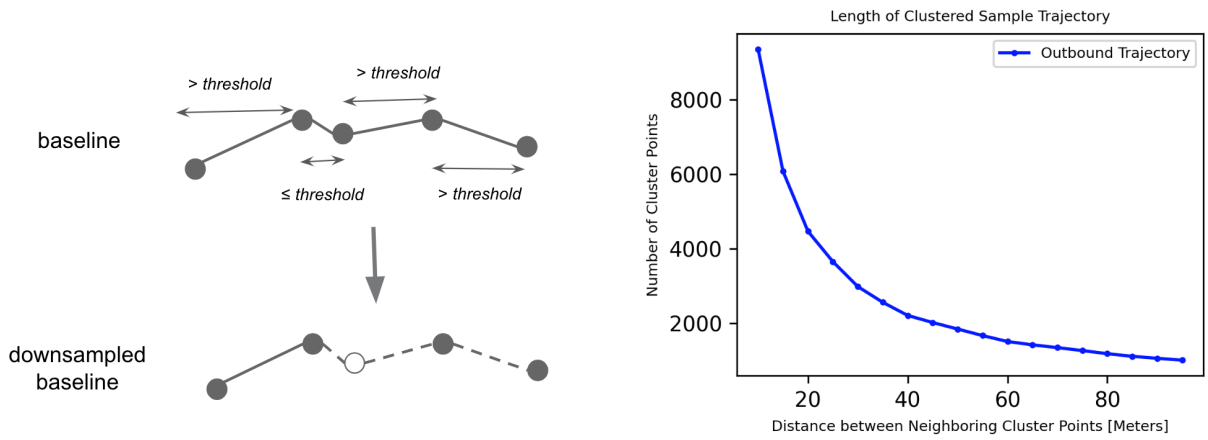


Figure 3.7: (Left) The dense baseline trajectory is downsampled based on a distance threshold to neighboring points. (Right) The number of points on the baseline trajectory decreases exponentially as we increase the distance threshold.

scale. One practical approach to working with these continuous data is to downsample the data to a coarse representation. Ego vehicle datasets, such as NuScenes [6], do so by extracting key frames among the continuous stream of data. However, in our case, we are interested in the relationship between neighboring data on both the spatial and temporal axes. Thus, it is important to find a sampling distance threshold that can extract a manageable set of coordinates while retaining interesting spatial and temporal attributes.

Given the densely clustered baseline trajectory, we want to downsample the points so that the sample clusters can retain spatial information. In the context of bus data, this means that neighboring clusters should share similar visual features. For example, if two neighboring points are too far apart, then we would lose a lot of the spatial imagery feature since they are aggregated under a single cluster point.

3.4.3 Assigning Data Points to the Baseline Trajectory

Now that we have a sequence of baseline coordinates, we want to align the collection of runs onto the baseline coordinates. Let K be the sequence of baseline coordinates composed of the baseline coordinates. Recall that a run is defined as $R_i = [d_{i,0}, ..., d_{i,j}, ..., d_{i,n}]$ and $d_{i,j} = (c_{i,j}, t_{i,j}, I_{i,j})$. For each data instance $d_{i,j}$, we assign it to the cluster $k = \text{argmin}_{k \in K}\, dist(c_{i,j}, k)$. As noted above, we can order the coordinates in K so that they follow the order of the bus trajectory.

From the above sections, we define a structure to organize the vast collection of bus data. First, we split the data into a series of runs R_i. Each run consists of an ordered sequence of data instances. Each data instance stores an image $I_{i,j}$, the time $t_{i,j}$ at which the image is taken, and the geographic location $c_{i,j}$ at which the image is taken. Each location $c_{i,j}$ is further mapped to one of the coordinates $k \in K$, an averaged and downsampled representation of the bus trajectory. Thus, we define the temporal axis as the start time of the run and the spatial axis as the indices of the ordered baseline coordinates.

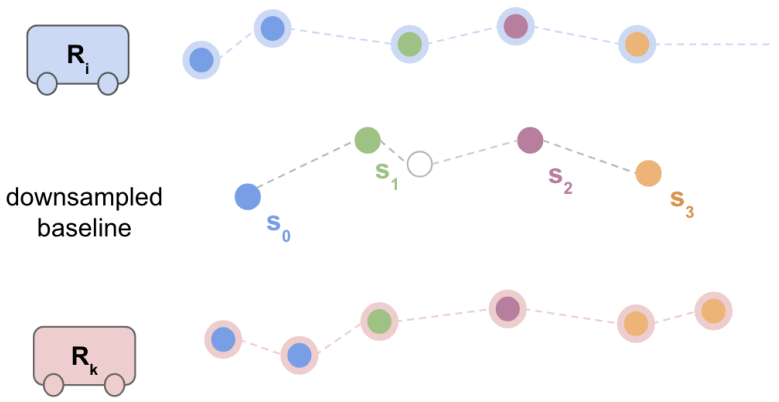


Figure 3.8: Data points on each run is assigned to its nearest point on the downsampled baseline trajectory.

Chapter 4

Work Zone Detection

Work zones are a common source of traffic disruption, causing great inconvenience for commuters. Gaining a better understanding of work zones can benefit various downstream applications. Our goal is to detect work zone among an unlabeled set of large-scale bus data. To do so, we exploit the spatial and temporal properties of the bus data to find patterns for the work zones along these axes. Anomalies can occur across the spatial axis (changes across time) or on a fixed starting time (across regions in space). As such, we are interested in answering the following questions:

- Is there a work zone at the current point (spatial location and time)?
- Is there an observable change when observing across time or space?

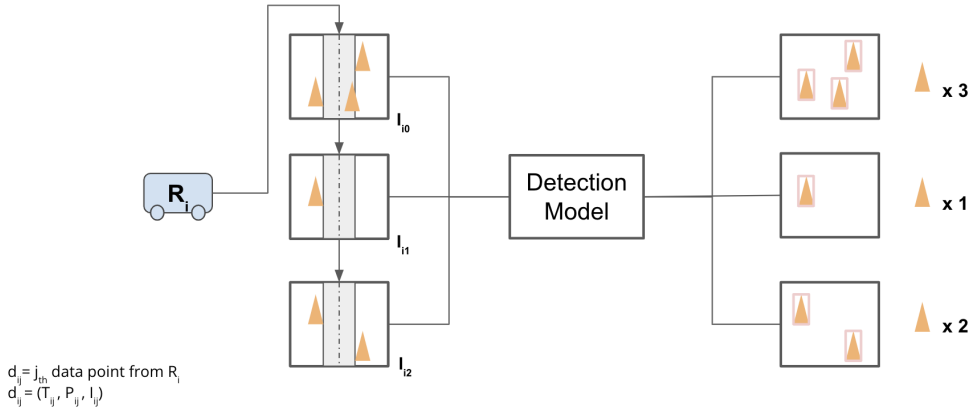


Figure 4.1: For each data point on a run, we assign an anomaly score that indicates the likelihood of that scene being a work zone.

Previously, we discussed several works that cover approaches to define a work zone. [15] uses a probabilistic model to model a binary classifier for work zones. Each factor related to the work zone, such as the speed of the vehicle and the existence of temporary traffic control objects, is treated as a weighted input to the model. Although this approach considers many related factors, it does not capture the spatial and temporal attributes of these related objects. In [28], each work zone is defined by the contoured region on the bird's-eye view plane. Key

points detected from a series of neighboring frames are aggregated on the bird's-eye view plane to construct the work zone contour. This work captures the spatial information of a work zone, but does not capture the temporal changes of a work zone. In our work, we aim to bridge this gap by detecting work zones from data points mapped onto a 2D spatial-temporal space. To do so, we use a bottom-up approach to compute an anomaly score for each data point, which measures the likelihood of a work zone at that data point. Figure 4.1 illustrates how we use the output of the detection models to assign anomaly scores. For each data point, we also consider its spatial and temporal neighbors to capture more spatial and temporal attributes. In this section, we focus on our approach of assigning each data point a corresponding anomaly score.

4.1 Work Zone Related Datasets

4.1.1 Manually Labeled Dataset

In an effort to make progress in research on work zone detection, we collected and annotated a set of images from work zones. We scrapped 4,400 work zone images from the Internet and selected 200 images that contain work zones from the bus. Currently, we have 4,600 work zone images. These images are manually annotated with objects listed in Table 2.1.

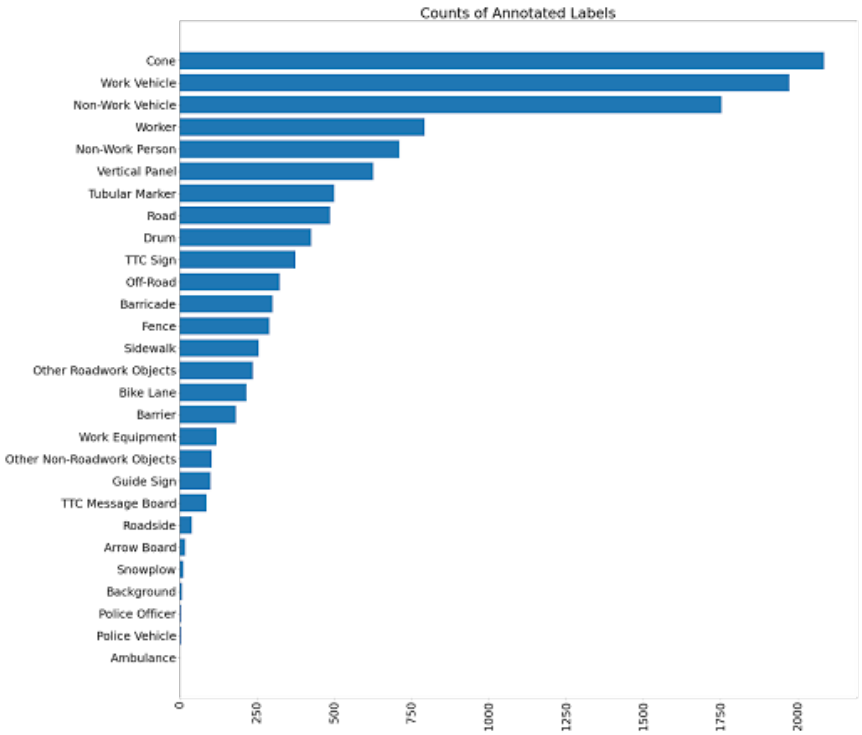


Figure 4.2: Visualization of current annotation progress of work zone related objects.

The annotation of this dataset is in ongoing progress. The instance counts in Figure 4.2 give us information about the frequency at which each object is present in the work zones. Top counts,

such as cones, work vehicles, worker, and vertical panels, are observed more frequently in work zone images.

4.1.2 NuScenes

NuScenes [6] is a public large-scale dataset for autonomous driving. It enables researchers to study challenging urban driving situations through densely captured visual inputs. NuScenes contains four class labels that are related to work zones: barriers, traffic cone, construction vehicle, and construction worker.

Barrier	Traffic Cone	Construction Vehicle	Construction Worker
88,545 (12.76%)	87,603 (12.63%)	6,071 (0.88%)	13,582 (1.96%)

Table 4.1: Number of instances and ratio of all annotations in each class label the NuScenes 2D dataset.

For preliminary experiments, we trained a multiclass object detection model using the four class labels related to the work zone: barrier, traffic cone, construction vehicle, and construction worker. In this experiment, we used a Faster R-CNN model [26] with a ResNet-50 backbone. From the training results, we noticed some challenges in the current experiment setup. First, the model performs worse on construction vehicles and construction workers compared to barriers and traffic cones. The imbalance in the number of instances among the four classes makes it difficult to train a multiclass object detector, especially when there are 15 times more instances of barrier than those of construction vehicles. Next, the model struggles to differentiate between normal vehicles (trucks) and construction vehicles (trucks in the construction zone). A similar problem persists between pedestrians and construction workers. This suggests that a hierarchical detection structure may be needed for these subclasses. From the current set of available annotations, the detection model performs consistently when performing single-class traffic cone detection.

For the purpose of this work, we use the number of traffic cones detected as a measure of the probability that the given region contains a work zone. We hope to improve this detector to a multiclass detector as we continue to annotate the work zone dataset. To train our object detection model, we use the traffic cone class [6], and tested our result on the traffic cone class of bus data.

	Number of Images	Traffic Cone
NuScenes [6]	11,867	87,603
Bus Data	158	1,042

Table 4.2: Number of images and number of instances for NuImages and Bus Data datasets.

4.2 Model Architecture

For the purpose of this work, we are interested in detecting work zones and analyzing their spatial-temporal relationship. As such, we are more interested in obtaining accurate bottom-up detection than in efficient real-time performance. The two-stage proposal generation network generally outperforms the one-stage detection model architecture. Thus, for our baseline model, we employ the Faster R-CNN model architecture with ResNet-50 and Feature Pyramid network as backbone. In addition, we also experiment with the current state of the art, which uses a transformer model Swin-T as backbone architecture.

To evaluate the performance of the object detector, we measured the following metrics: AP^{bb}, AP^{bb}_{50}, AP^{bb}_{75}, AP^{bb}_{S}, AP^{bb}_{M}, and AP^{bb}_{L}, which are standard COCO metrics [16]. Average precision (AP) measures the mean precision values set at some recall threshold level (0 to 1 with a step size of 0.1). The numerical subscript of the AP symbols refers to the threshold of the IoU score. AP^{bb}_{50} means that a prediction with IoU > 0.5 is considered a true positive prediction. Usually, the mean average precision (mAP) is averaged across all class labels. In our case, we are working with single-class object detection, so we do not take the mean. The subscript S, M, L refers to the sizes of the detected objects. AP^{bb}_{S} measures the average precision score of objects whose area of the ground truth bounding box is less than 32^2. AP^{bb}_{M} consists of boxes whose area is in the range of 32^2 and 96^2. AP^{bb}_{L} contains boxes of area $> 96^2$.

	Backbone	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}_{S}	AP^{bb}_{M}	AP^{bb}_{L}
Faster R-CNN [26]	ResNet-50 FPN [12, 17]	46.8	83.8	45.5	35.6	57.1	65.2
Faster R-CNN [26]	Swin-T FPN [17, 20]	55.7	90.8	59.0	44.7	65.1	75.0

Table 4.3: Traffic cone detection results on dev set of NuScenes on the Traffic Cones class.

Based on Table 4.3, Faster R-CNN with a Swin-T outperforms the model using a ResNet-50 backbone. Next, we will test the performance of this traffic cone detector on a set of manually annotated bus data using images that contain traffic cone instances. We expect the performance of the traffic cone detector to drop on the bus data, since the training and testing distribution has changed.

Data	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}_{S}	AP^{bb}_{M}	AP^{bb}_{L}
NuScenes [Train]	46.8	83.8	45.5	35.6	57.1	65.2
Bus Data [Evaluation]	35.8	81.4	25.6	32.5	64.7	-

Table 4.4: Traffic cone detection results using Faster R-CNN with Swin-T backbone.

Our goal is to detect work zone events among an unlabeled set of large-scale bus data. To do so, we design a model that assigns an anomaly score to each input data point. The anomaly score indicates the likelihood of a work zone at that data point. Currently, we use predicted traffic cone counts as an anomaly score. Since the goal is to mine and detect patterns of work zones in bus data, we value high recall over high precision. We want to capture as many instances of the work zone as possible to find patterns in the spatial and temporal domains. From the precision recall

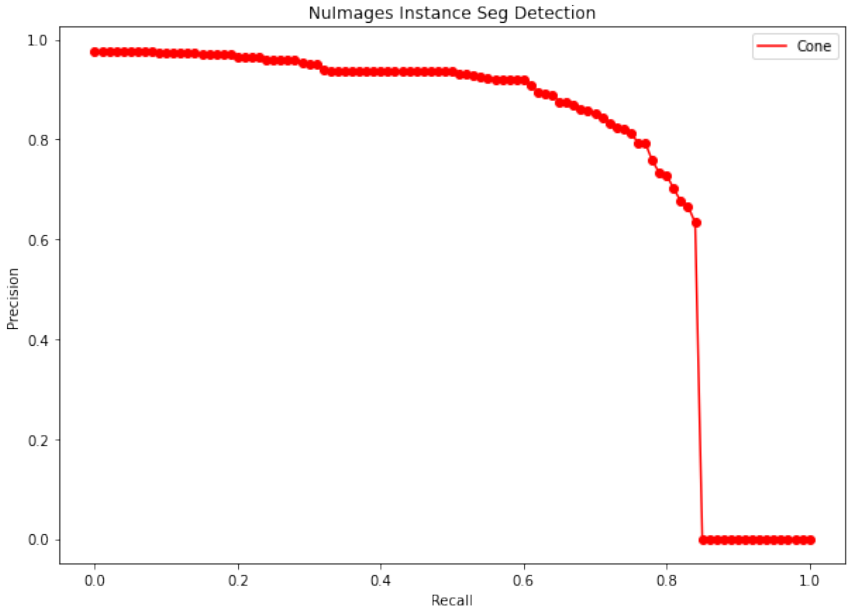


Figure 4.3: Precision-Recall curve of Faster R-CNN Swin-T FPN model on Bus Data.

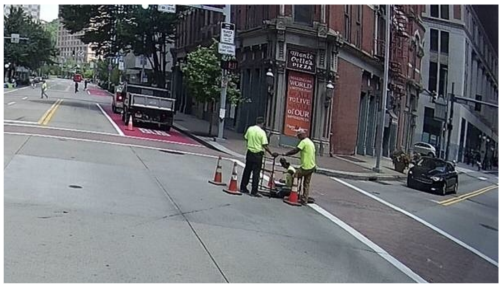
curve, we see that the model is capable of achieving a recall greater than 0.8 while maintaining an accuracy greater than 0.6. Figure 4.3 was generated using a confidence threshold of 0.05. Some of the detected results are shown in Figure 5.2. In the next section, we show that with this performance we can extract interesting patterns about work zones from large amounts of bus data when aggregated across the spatial and temporal dimension.

4.3 Sample Outputs

In the following experiments, we use the predicted traffic cone counts at a data point as an indicator of the likelihood that there is a work zone at that location. This means that if there is a high number of traffic cones, we claim that there is a work zone at that given location. However, traffic cone counts in a single frame may not be the most accurate measurement of the anomaly score. In the following, we provide a few examples where ambiguities arise. We later show that these ambiguities can be avoided by considering the anomaly score across the spatial and temporal axes.

In the top-left image of Figure 4.4, there are many traffic cones in the parking lot, behind each of the parked vehicles. However, it is clear that it is not an active construction site. In the top right image, we see an active work zone in the middle of the road. Our method would then correctly predict that location as a work zone. In the bottom-left image, our method would also correctly mark it as not a work zone since the model detects a low number of traffic cones. The image on the bottom right indicates the start of a medium-to-large scale work zone. Temporary Traffic Control (TTC) sign is placed to warn drivers that there is a work zone ahead. However, since the image contains a low number of traffic cones, our method would falsely classify the location as not a work zone. In the next section, we show that by aggregating neighboring spatial

No Construction, High Score



Yes Construction, High Score

No Construction, Low Score

Yes Construction, Low Score

Figure 4.4: On a single frame, there are ambiguities on whether high predicted traffic cone counts indicate the location is a work zone.

and temporal predictions, we can avoid these ambiguities and mine interesting patterns about work zones from the large amount of bus data.

Chapter 5

Results

Our goal is to detect and analyze instances of work zones from a set of raw bus data. To do so, we introduce a method for mapping data points from bus trajectories into a discretized 2D space defined by a spatial axis and a temporal axis. The temporal axis is defined by the start times for each bus run. The spatial axis is an sequence of baseline coordinates. The collection of bus data is assigned to a point on the baseline trajectory. Next, we use a bottom approach to evaluate the anomaly score at each data point. Finally, we visualize the anomaly scores on a discretized 2D space defined by a spatial and temporal axis. From this 2D map, we discuss the spatial and temporal attributes that we discovered by mining bus data for work zone events.

5.1 Overview

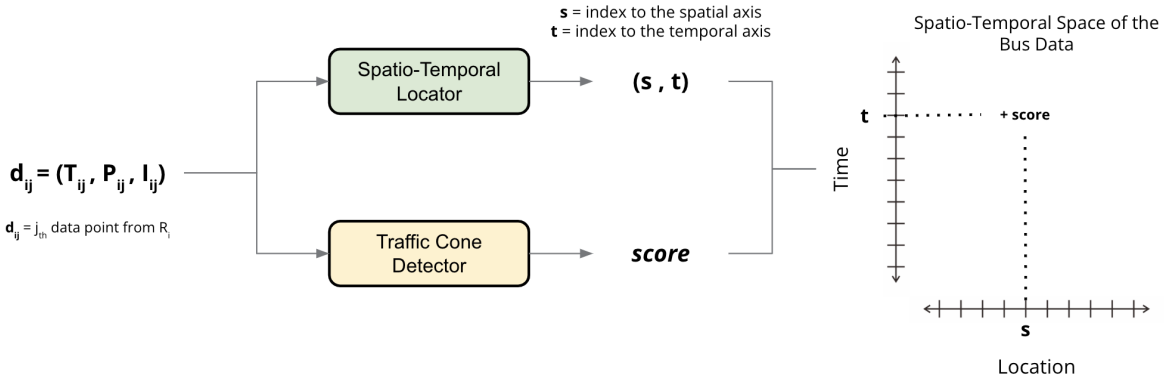


Figure 5.1: Pipeline of mapping each input data point to a value on the spatial-temporal space.

Previously, we discussed how to formulate the temporal axis and the spatial axis from the collection of raw bus data. Based on our definition, the temporal axis corresponds to the ordering of the runs based on the start time. For example, the first run R_0, which started at time T_{00}, would correspond to the index 0 on the temporal axis. The last run R_N would correspond to the index

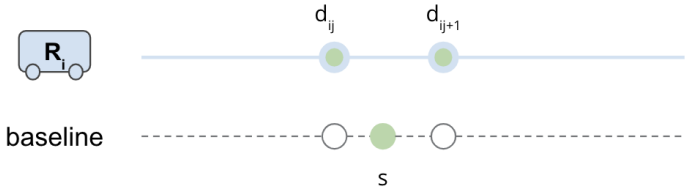
$N - 1$ on the temporal axis. Formulating the spatial axis requires more work. The coordinates of all the runs are clustered and then downsampled to form a baseline trajectory. This baseline trajectory is a representative sequence of coordinates on which the bus travels. Since the bus travels to and from two fixed endpoints, we assign an ordering to the baseline trajectory and use this ordering as the spatial axis. In the spatial dimension, each data point is mapped to the index that corresponds to the coordinate on the baseline trajectory closest to the data point's location. We call this module the Spatial-Temporal Locator.

We also defined a method for assigning an anomaly score to each data point. For each data point, we pass it through a trained traffic cone object detector to obtain a predicted traffic cone count of the scene. We use this count score as an indication of the likelihood of a work zone at that data point. We call this module the Traffic Cone Detector.

Let d_{ij} be a data point that corresponds to the j_{th} data point in i_{th} run R_i. We feed d_{ij} through the two above modules. The Spatial-Temporal Locator returns two indices (s, t), which indicates the location where d_{ij} is assigned to the 2D discretized space. The Traffic Cone Detector returns an integer score, which is the value at that location. The discretized 2D space can be naturally visualized in the form of a heat map. In the following sections, we will analyze the spatial and temporal attributes of work zones by mining for patterns on the heat map.

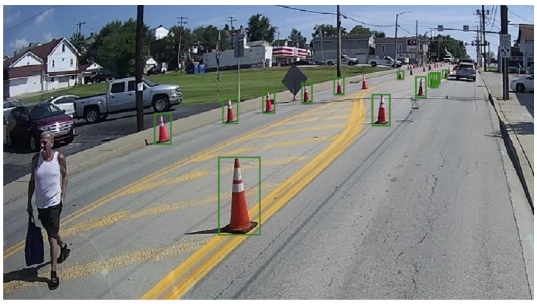
5.2 Visualizations

5.2.1 Spatial Aggregation of Detection Results



(a) Two neighboring frames being aggregated to the same baseline point.

(b) Image from d_{ij}

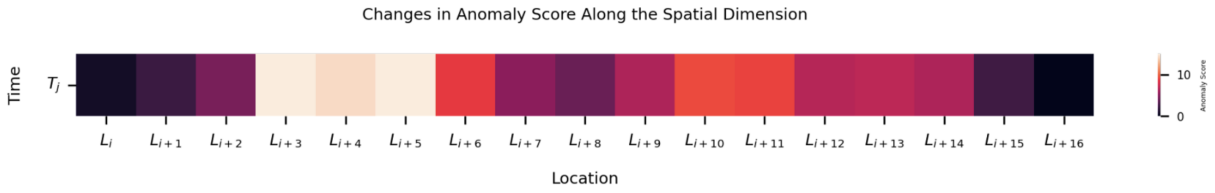


(c) Image from d_{ij+1}

Figure 5.2: (b) and (c) are two neighboring frames along the same run. As we get closer to the objects, we can detect them more accurately.

In Figure 5.2(b), the traffic cones in the second rows were not detected at time step T_{ij}. As the bus approaches closer to the work zone, the second row of traffic cones are detected by the detector more easily. In this scene, the work zone cannot be fully captured in a single shot. As the bus moves along the spatial axis, it can gather more information about the same work zone. By aggregating neighboring frames to a coarser baseline trajectory, we can capture more indicative information about the region in space. This is especially applicable to work zones, since work zones vary in scale and are rarely contained within a single frame.

5.2.2 Spatial Changes



(a) Heat map of spatial changes along run R_i

(b) Image at L_i

(c) Image at L_{i+2}

(d) Image at L_{i+14}

(e) Image from L_{i+15}

Figure 5.3: A heat map indicating changes across the spatial axis and their corresponding images.

Figure 5.3 demonstrates a large scale work zone region where half of the road is blocked. From the heat map, we see an increase in traffic cone counts near the start of the segment, then a decrease at the tail of the region. Looking at the matching images, we see that at the beginning of the work zone, there is a denser placement of traffic cones to notify the incoming drivers. In the middle sections of the work zones, the traffic cones are placed more sparsely. The gradient

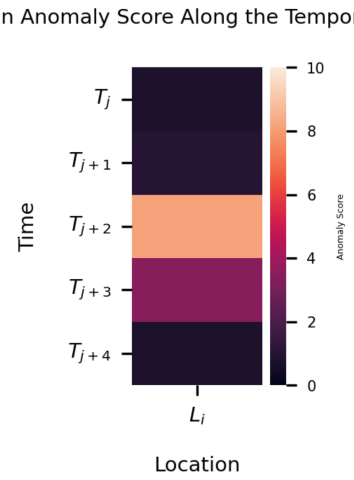
pattern on the heat map gives clear indications of the spatial boundaries of the work zone. Using this pattern, we can efficiently identify the potential boundaries of the work zones at a coarse level (from downsampled trajectories). These data can help autonomous vehicle teams to learn to predict the spatial boundaries of construction zones, which can enhance the robustness of the autonomous vehicle when faced with roadside anomalies.

5.2.3 Temporal Changes

From the heat map in Figure 5.4 (a), we see a similar gradient effect along the temporal axis at a specific location L_i. Looking at the corresponding image, we see that the location L_i is undergoing a long-term construction that spans multiple runs of the bus. At time T_{i+1} (Figure 5.4 (b)), there is a white box near the center of the road and there are no indicators of work zones yet. The bus data clearly captured the progression of this work zone from start to finish. At time T_{i+2} (Figure 5.4 (c)), the same location has become an active construction site with traffic cones and construction vehicles present. At time T_{i+2} (Figure 5.4 (e)), the traffic cones were removed and we observed a change in the white box region in Figure 5.4 (b). From changes along the temporal axis, we were able to pinpoint a specific region of change among the vast amount of large-scale bus data. By mapping the bus data onto a spatial-temporal structured space, we were able to capture the progress of a work zone from start to finish. These data could be invaluable to construction management companies for an automated process of monitoring the progress of work zones.

5.2.4 The Bigger Picture

In the previous two sections, we analyzed the patterns of work zones along the temporal and spatial axis. The patterns extracted along either of the axes can be beneficial for a wide range of downstream tasks. For example, construction companies may be interested in monitoring changes in a work zone over time. Autonomous vehicle teams may be interested in data containing road anomalies for training. The possibilities of these insights are immense, but they are challenging to mine. In total, we used two months of bus data in these experiments, which contains approximately 916K raw data points. Figure 5.5 shows the heat maps in different subsegments along the spatial axis. We notice that most of the heat maps are null, which means that there are no signs of the work zone at that time and location. Without an efficient way of organizing the bus data, it would be computationally expensive to extract spatial and temporal attributes over large-scale data.



(a) Heat map of temporal changes at location L_i

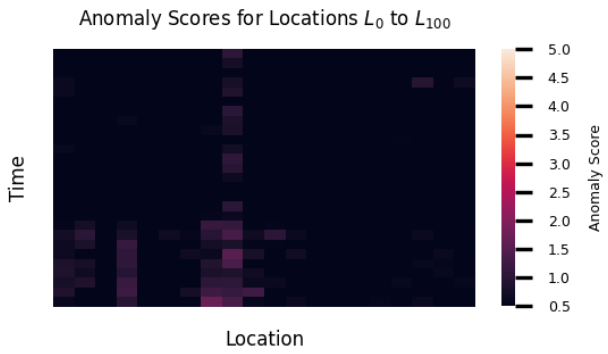
(b) Image at T_{i+1}

(c) Image at T_{i+2}

(d) Image at T_{i+3}

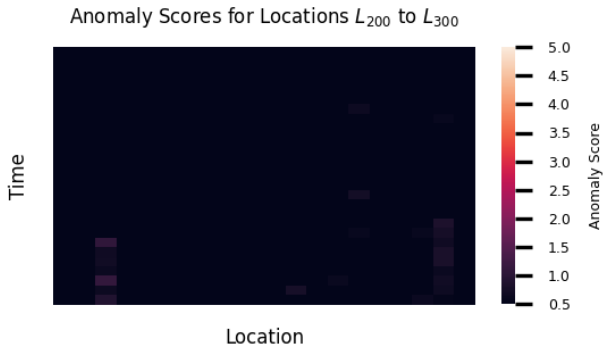
(e) Image from T_{i+4}

Figure 5.4: A heat map indicating changes across the temporal axis and their corresponding images.

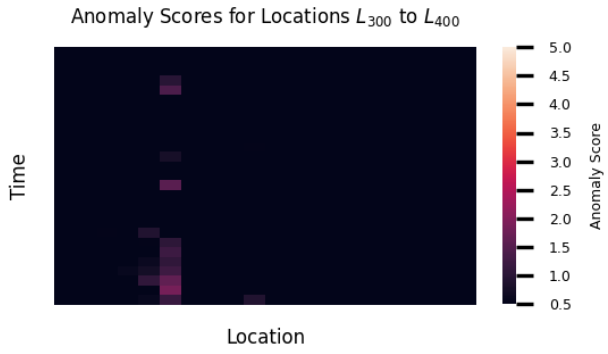


(a) Location L_0 to L_{100}

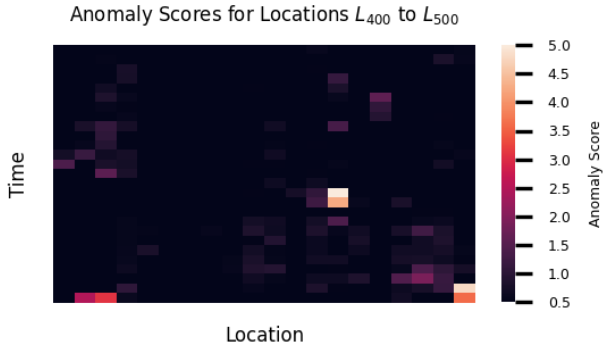
(b) Location L_{100} to L_{200}



(c) Image at Location L_{200} to L_{300}

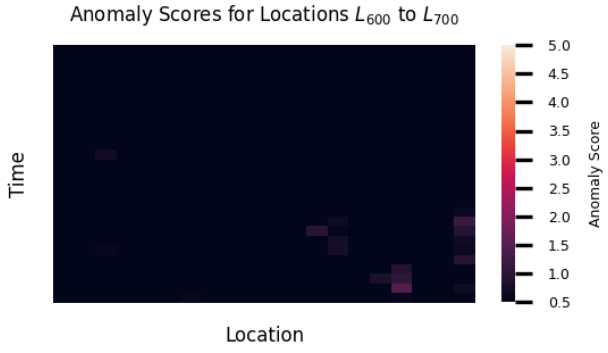


(d) Location L_{300} to L_{400}



(e) Location L_{400} to L_{500}

(f) Location L_{500} to L_{600}



(g) Location L_{600} to L_{700}

(h) Location L_{700} to L_{800}

Figure 5.5: Collection of heat map at different subsegments of the spatial axis.

Chapter 6

Conclusion and Future Work

In this work, we detected and analyzed the spatial and temporal patterns of work zones from a large scale, unprocessed collection of bus data. To efficiently draw inferences from the bus data, we clustered and downsampled the data points based on their spatial property. Next, we imposed an ordering on the sequences of bus data based on the start time of each run. This organization enables us to effectively exploit the spatial and temporal patterns of the events of interest. In our work, we focus on detecting and analyzing patterns of work zones. We noticed that work zones naturally span long periods of time and cover a certain region in space. Through looking at locations or times where there is a high number of traffic cone counts, we were able to mine instances of work zones among 916K raw image data. The scope of this work focuses on exploring ways to organize and extract information about events of interest from large-scale, unprocessed data. From observing the spatial and temporal axes, we identified coarse regions where events of interest are located. This opens up possibilities for a variety of downstream tasks. For example, we can extend the current scope of our method and perform fine analysis on the extracted work zone regions. These could include tasks such as quantifying the scale of a work zone, localizing objects within a work zone, and performing semantic understanding of a work zone.

Like bus data, many data sources naturally contain spatial or temporal properties. For example, taxi vehicle records contain spatial information about the trajectory the taxi drove on. Videos from surveillance cameras contain rich information about changes over time in some fixed location. By defining a spatial or temporal organization for these data, we can more effectively extract information from these urban big data. This can tackle the difficulty of extracting points of interest from a large set of data. In boarder terms, our work highlights the effectiveness of exploiting spatial and temporal patterns to perform more efficient urban visual data mining.

Bibliography

[1] Illustration of freedom transit metro route. URL https://freedom-transit.org/Metro-commuter-bus-service-Mon-Fri-Washington-County-PA.htm. 3.1

[2] 2009 edition, original, dated december 2009 (pdf). URL https://mutcd.fhwa.dot.gov/pdfs/2009/pdf_index.htm. 2.3, 2.3

[3] Work/construction zones. URL https://www.nhtsa.gov/. 2.3

[4] Mahmuda Ahmed, Sophia Karagiorgou, Dieter Pfoser, and Carola Wenk. A comparison and evaluation of map construction algorithms using vehicle tracking data. *GeoInformatica*, 19(3):601–632, 2015. 3.4.1

[5] Gowtham Atluri, Anuj Karpatne, and Vipin Kumar. Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys (CSUR)*, 51(4):1–41, 2018. 2.2

[6] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 2.3.1, 2.3.2, 2.4, 3.4.2, 4.1.2, 4.1.2, **??**

[7] Lili Cao and John Krumm. From gps traces to a routable road map. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, pages 3–12, 2009. 3.4.1

[8] Brian Chen, Robert Tamburo, and Srinivas Narasimhan. Automatic detection of road work and construction with deep learning model and a novel dataset. URL https://www.youtube.com/watch?v=pPjIS1rz5SU. 2.3.3

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2.3.1

[10] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 2.4

[11] Ali Hamdi, Khaled Shaban, Abdelkarim Erradi, Amr Mohamed, Shakila Khan Rumi, and Flora D Salim. Spatiotemporal data mining: a survey on challenges and open problems. *Artificial Intelligence Review*, pages 1–48, 2021. 2.2

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for

image recognition. *arXiv preprint arXiv:1512.03385*, 2015. **??**

[13] Peng Jiang, Xiuju Fu, Yee Fan, Jiri Klemeš, Piao Chen, Stefan Ma, and Wanbing Zhang. Spatial-temporal potential exposure risk analytics and urban sustainability impacts related to covid-19 mitigation: A perspective from car mobility behaviour. *Journal of Cleaner Production*, 279:123673, 08 2020. doi: 10.1016/j.jclepro.2020.123673. 2.2

[14] Harsurinder Kaur, Husanbir Singh Pannu, and Avleen Kaur Malhi. A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Comput. Surv.*, 52(4), aug 2019. ISSN 0360-0300. doi: 10.1145/3343440. URL `https://doi.org/10.1145/3343440`. 2.4

[15] Philipp Kunz and Matthias Schreier. Automated detection of construction sites on motorways. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1378–1385, 2017. doi: 10.1109/IVS.2017.7995903. 2.3.1, 4

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick. Microsoft coco: Common objects in context. In *ECCV*. European Conference on Computer Vision, September 2014. URL `https://www.microsoft.com/en-us/research/publication/microsoft-coco-common-objects-in-context/`. 4.2

[17] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. **??**, **??**

[18] Jia Liu, Tianrui Li, Peng Xie, Shengdong Du, Fei Teng, and Xin Yang. Urban big data fusion based on deep learning: An overview. *Information Fusion*, 53:123–133, 2020. 1, 2.1, 2.2

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2.4

[20] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. URL `https://arxiv.org/abs/2103.14030`. **??**

[21] Chuishi Meng, Xiuwen Yi, Lu Su, Jing Gao, and Yu Zheng. City-wide traffic volume inference with loop detector data and taxi trajectories. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 1–10, 2017. 2.1

[22] Daniel Moreira, Sandra Avila, Mauricio Perez, Daniel Moraes, Vanessa Testoni, Eduardo Valle, Siome Goldenstein, and Anderson Rocha. Multimodal data fusion for sensitive scene localization. *Information Fusion*, 45:307–323, 2019. ISSN 1566-2535. 2.1

[23] Prajakta Ganesh Pawar and V Devendran. Scene understanding: A survey to see the world at a single glance. In *2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT)*, pages 182–186, 2019. doi: 10.1109/ICCT46177.2019.

8969051. 2.3.3

[24] Simone Porru, Francesco Edoardo Misso, Filippo Eros Pani, and Cino Repetto. Smart mobility and public transport: Opportunities and challenges in rural and urban areas. *Journal of traffic and transportation engineering (English edition)*, 7(1):88–97, 2020. 3

[25] Amudapuram Mohan Rao and Kalaga Ramachandra Rao. Measuring urban traffic congestion-a review. *International Journal for Traffic & Transport Engineering*, 2(4), 2012. 2.2

[26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2.4, 4.1.2, **??**, **??**

[27] Prasanta K Sahu, Babak Mehran, Surya P Mahapatra, and Satish Sharma. Spatial data analysis approach for network-wide consolidation of bus stop locations. *Public Transport*, 13(2):375–394, 2021. 3

[28] Weijing Shi and Ragunathan Raj Rajkumar. Work zone detection for autonomous vehicles. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 1585–1591, 2021. doi: 10.1109/ITSC48978.2021.9565073. 2.3.1, 2.4, 2.3.3, 4

[29] Shijie Sun, Naveed Akhtar, Huansheng Song, Chaoyang Zhang, Jianxin Li, and Ajmal Mian. Benchmark data and method for real-time people counting in cluttered scenes using depth sensors. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3599–3612, 2019. 3

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`. 2.4

[31] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021. 2.3.2, 2.4

[32] Canbo Ye. Busedge: Efficient live video analytics for transit buses via edge computing. Master's thesis, Pittsburgh, PA, July 2021. 2.4, 3, 3.2

[33] Xiuwen Yi, Junbo Zhang, Zhaoyuan Wang, Tianrui Li, and Yu Zheng. Deep distributed fusion network for air quality prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, KDD '18, page 965–973, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450355520. doi: 10.1145/3219819.3219822. URL `https://doi.org/10.1145/3219819.3219822`. 2.1

[34] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2.3.1

[35] Lili Zhang, Yuxiang Xie, Luan Xidao, and Xin Zhang. Multi-source heterogeneous data fusion. In *2018 International conference on artificial intelligence and big data (ICAIBD)*, pages 47–51. IEEE, 2018. 2.1

[36] Bowen Zhao, Chen Chen, Wanpeng Xiao, Xi Xiao, Qi Ju, and Shutao Xia. Towards a category-extended object detector without relabeling or conflicts, 12 2020. 2.4