City brain: practice of large-scale artificial intelligence in the real world

Jianfeng Zhang¹, Xian-Sheng Hua¹✉, Jianqiang Huang¹, Xu Shen¹, Jingyu Chen¹, Qin Zhou¹, Zhihang Fu¹, Yiru Zhao¹,²

¹DAMO Academy, Alibaba Group, 969 West Wenyi Road, Hangzhou, Zhejiang Province, People’s Republic of China
²Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai, People’s Republic of China
✉E-mail: xiansheng.hxs@alibaba-inc.com

Abstract: A city is an aggregate of a huge amount of heterogeneous data. However, extracting meaningful values from that data remains a challenge. City Brain is an end-to-end system whose goal is to glean irreplaceable values from big city data, specifically from videos, with the assistance of rapidly evolving artificial intelligence technologies and fast-growing computing capacity. From cognition to optimisation, to decision-making, from search to prediction and ultimately, to intervention, City Brain improves the way to manage the city, as well as the way to live in it. In this study, the authors introduce current practices of the City Brain platform in a few cities in China, including what they can do to achieve the goal and make it a reality. Then they focus on the system overview and key technical details of each component of the City Brain system, from cognition to intervention. Lastly, they present a few deployment cases of City Brain in various cities in China.

1 Introduction

1.1 About City Brain

As early as 2016, Smart City was presented as a national strategy in China: We should profoundly understand the role of the Internet in nation management and society governance, taking the implementation of e-government and building new smart cities as the key points. We will build a nationally integrated big data center by data integration and promote technology convergence, business integration, and data convergence to achieve collaborative management and services across geographies, systems, departments, and services. Today, the first batch of ‘Digital Twin Cities’ using artificial intelligence (AI) technologies have realised the Internet mode of data sharing, data co-creation, and data automatic control with the help of Alibaba City Brain.

The City Brain is the ‘commanding heights’ of technologies in Alibaba Group. Based on the elastic calculation and large-scale data processing platform of Alibaba Cloud, integrated with the top capabilities of interdisciplinary fields such as machine vision, large-scale topological network computing, and traffic flow analysis, the City Brain is capable of massive multi-source data collection, real-time processing, and intelligent computing. There are three metrics for a real ‘City Brain’: (1) it can deal with ultra-large-scale and multi-source data that humans cannot understand in real time (global cognition); (2) It can understand the complex hidden rules that humans have not discovered (machine learning); (3) It can formulate a global optimal strategy that surpasses local suboptimal decision made by human (global coordination).

The City Brain has become a powerful assistant for city managers in cognising, transforming, and operating cities. It transcends human capabilities with four kinds of ‘super powers’: (1) machine vision cognitive capability to enhance perception of urban data; (2) the full-scale data platform construction capacity to enhance the ‘data density’ and ‘particle management’ level; (3) real-time computing capability under large-scale dynamic topology networks; (4) the City Brain open platform capability to empower the digital city industry.

The City Brain is deployed according to five major application scenarios: urban traffic checkup, urban police monitoring, urban traffic micro-control, urban special vehicles, and urban strategic planning. (1) Urban traffic checkup can completely quantify the urban ‘vital signs’ via the fusion and integration of full-scale, full-network, and cross-domain data, avoiding one-sided solutions for urban problems due to the single source data; (2) By taking advantage of machine learning and computer vision, automatic police monitoring can liberate police officers from laborious legwork, and let the data to run errands, instead of police officers; (3) Urban traffic micro control-and-feedback loop. It opens the feedback control system between ‘brains’, ‘eyes’, and ‘hands and feet’. Based on multi-source data, the global intelligent algorithm provides a fine-grained control of city-scale traffic signals to improve mobility in the city; (4) Route optimisation for emergency vehicles. City Brain identifies the quickest route for emergency vehicles to arrive at the scene within the shortest time frame; (5) Urban layout planning and verification, which analyses the effect of a proposed urban construction blueprint on the cloud with the simulation data model.

1.2 History

In April 2016, the concept of ‘city brain’ was formally proposed. City Brain is a new infrastructure built on massive data, which utilises AI to solve urban governance and development issues that cannot be solved by the human brain. It is a program that offers a comprehensive suite of acquisition, integration, and analysis of big and heterogeneous data generated by a diversity of sources in urban spaces through video and image recognition, data mining and machine learning technology. With this, city council and urban planners will be able to make better decisions for the community.

In November 2017, Alibaba Cloud ET City Brain was selected as one of the first four AI innovation platforms by the Ministry of Science and Technology, which became a major contribution of Chinese technology to the world's urban area.

On January 29, 2018, the Malaysia Digital Economy Corp (MDEC) and the Dewan Bandaraya Kuala Lumpur (DBKL) jointly announced the introduction of Alibaba Cloud ET City Brain. The AI will be fully applied to Malaysia’s traffic management, urban planning, and environmental protection. It is the first time that the City Brain went out to serve worldwide customers.

It has been three years from the birth of City Brain to the present. The City Brain has been launched in Hangzhou, Shanghai, Chongqing, Suzhou, Haikou, Beijing, Chengdu, Quzhou, Jiaxing, Kuala Lumpur, Macao, and many other cities.
to respond to those situations more effectively. For example, if we have cognitive results, e.g. automatic accident alerting [2], traffic light decisions or optimise the ways we run the city based on the data is generated every day in a city. These videos play critical roles in city management, public safety, traffic control, and environment protection etc. However, video data is unstructured. How to effectively store, analyse, and further take advantage of these videos has been a worldwide problem.

In order to address the above problem, our team builds the large-scale visual computing platform to meet the requirement for real-time, comprehensive, large-scale smart video analysis, which makes joint perception, prediction, alarm and prevention in smart city management possible for the first time.

The overall architecture of the platform is illustrated in Fig. 2, which composes of three core systems, namely ‘the Access and Transmitting system’, ‘the Computing system’, and ‘the Searching system’. The access and transmitting stage perform data accessing, data pre-processing, data resource scheduling, data transmitting, and video streaming.

Based on the stream-processing framework (Flink [4]), the computing system has the following key capabilities: batch computing, stream computing, model parallelisation, model scheduling, graphical calculation, and atlas calculation. These key techniques are able to support the top-level applications such as online/offline video analysis, trajectory tracking, feature quantitation etc.

The searching system consists of the large-scale search engine, online feature extraction service and search strategy engine. The search engine performs real-time index compression. Online feature extraction is responsible for extracting features of the city objects from video frames. The search strategy engine links the former two modules and provides an image search service to target customers.

The visual computing platform can be deployed on the cloud. It could be shared and reused through the cloud resource pool, fully exploiting the efficiency of multi-core and ensuring elastic expansion. Besides, by means of the peak staggered multiplexing, the platform achieves flexible and efficient resource utilisation.

The distributed deployment of cloud host could provide intelligent analysis capability on demand, thus improving the efficiency of intelligent analysis. With the large-scale visual computing platform, we provide the capabilities of AI, large-scale data processing and cloud computing to the upper-level application layer, allowing customers to focus on business innovation.

3.2 Key technical details
3.2.1 Distributed heterogeneous scheduling engine: The large-scale video computing resource scheduling system manages the cloud video computing resources and dynamically adjusts the resource allocation to best utilise the computing ability [4, 5]. Its core functions include single-node heterogeneous computing scheduling, distributed heterogeneous computing resource scheduling, and distributed task dynamic allocation.

Single-node heterogeneous computing scheduling: this part evaluates the model's requirements for computing resources, and allocates appropriate heterogeneous computing resources (central processing unit, graphics processing unit etc.) and model operating parameters to the model on a single node according to the actual configurations of the machines. In this way, we can improve the resource utilisation rate as well as the number of video streams that can be processed on a single node.

Distributed Heterogeneous Computing Resource Scheduling: this part analyses and evaluates the computing resources for all tasks running on the streaming computing platform and allocates

Fig. 1 100 feet view of the City Brain

Fig. 2 Architecture of the large-scale visual computing platform

2 Overview of the City Brain

In this project, the challenges we are facing are all about three keywords: cost, value, and difference. Whether the cost for such a big computation, storage, and network intensive task is manageable, whether the technology is ready to get the values from those data, and whether the values are sufficiently significant.

What has been challenged even more is where are the differences compared with ‘video surveillance’ and ‘edge computing’.

These questions can be well answered by taking a closer look at the City Brain (Fig. 1). First, we have a bunch of data from the city, including the video data. The first step is to acquire the data and understand the data. We call this step 'Cognition', which includes recognising what is on the road and what is happening on the road, such as the cars, the people, the cyclist, the traffic status, the accidents etc [1].

Then, the second step ‘Decision and Optimisation’, we make decisions or optimise the ways we run the city based on the cognitive results, e.g. automatic accident alerting [2], traffic light optimisation. Thereafter, in the ‘Search and Mining’ step, we put everything the cameras have seen into a database and build an index, thus we can apply search on this data. For example, we find a suspicious car or discover patterns in the data, such as finding the root cause of traffic congestion somewhere in the city [3].

Next, based on current and historical data, we can predict what is going to happen next, either in a short period of time, such as the traffic congestion possibility after 20 min for an intersection, or next day's accident possibility of a road section, given the weather condition and event information of the city.

Last, based on predicted results, resources can be pre-allocated to respond to those situations more effectively. For example, if we know the possibility of accident will increase three times given the bad weather tomorrow as well as a few events that will gather a large number of people, we can adjust the traffic lights and send traffic advice to prevent those bad things from happening. We call this 'Prediction' and ‘Intervention’.

In the remaining part of this paper, we will present more details about the aforementioned parts, as well as the specifically designed large-scale visual computing platform.

3 Large-scale visual computing platform

3.1 System overview

With the rapid development of urbanisation, large amount of video data is generated every day in a city. These videos play critical roles in city management, public safety, traffic control, and environment protection etc. However, video data is unstructured. How to effectively store, analyse, and further take advantage of these videos has been a worldwide problem.

In order to address the above problem, our team builds the large-scale visual computing platform to meet the requirement for real-time, comprehensive, large-scale smart video analysis, which makes joint perception, prediction, alarm and prevention in smart city management possible for the first time.

The overall architecture of the platform is illustrated in Fig. 2, which composes of three core systems, namely ‘the Access and Transmitting system’, ‘the Computing system’, and ‘the Searching system’. The access and transmitting stage perform data accessing, data pre-processing, data resource scheduling, data transmitting, and video streaming.

Based on the stream-processing framework (Flink [4]), the computing system has the following key capabilities: batch computing, stream computing, model parallelisation, model scheduling, graphical calculation, and atlas calculation. These key techniques are able to support the top-level applications such as online/offline video analysis, trajectory tracking, feature quantitation etc.

The searching system consists of the large-scale search engine, online feature extraction service and search strategy engine. The search engine performs real-time index compression. Online feature extraction is responsible for extracting features of the city objects from video frames. The search strategy engine links the former two modules and provides an image search service to target customers.

The visual computing platform can be deployed on the cloud. It could be shared and reused through the cloud resource pool, fully exploiting the efficiency of multi-core and ensuring elastic expansion. Besides, by means of the peak staggered multiplexing, the platform achieves flexible and efficient resource utilisation.

The distributed deployment of cloud host could provide intelligent analysis capability on demand, thus improving the efficiency of intelligent analysis. With the large-scale visual computing platform, we provide the capabilities of AI, large-scale data processing and cloud computing to the upper-level application layer, allowing customers to focus on business innovation.

3.2 Key technical details
3.2.1 Distributed heterogeneous scheduling engine: The large-scale video computing resource scheduling system manages the cloud video computing resources and dynamically adjusts the resource allocation to best utilise the computing ability [4, 5]. Its core functions include single-node heterogeneous computing scheduling, distributed heterogeneous computing resource scheduling, and distributed task dynamic allocation.

Single-node heterogeneous computing scheduling: this part evaluates the model's requirements for computing resources, and allocates appropriate heterogeneous computing resources (central processing unit, graphics processing unit etc.) and model operating parameters to the model on a single node according to the actual configurations of the machines. In this way, we can improve the resource utilisation rate as well as the number of video streams that can be processed on a single node.

Distributed Heterogeneous Computing Resource Scheduling: this part analyses and evaluates the computing resources for all tasks running on the streaming computing platform and allocates

Fig. 1 100 feet view of the City Brain

Fig. 2 Architecture of the large-scale visual computing platform
for creating an index. However, creating and maintaining an index takes extra time and physical space, which increases the maintenance cost of the data.

To enable query based on graph data, the large-scale visual computing platform adopts the state-of-the-art index and search techniques. By taking the relationships among the graph nodes into consideration, we can make globally optimised predictions and interventions on the real-time city events.

3.2.3 Model quantisation and acceleration: To efficiently execute deep models on the proposed large-scale visual computing platform, we introduce network quantisation techniques to reduce the computational load [7].

Our work is devoted to quantising full-precision networks into low-bit networks. Existing methods formulate the low-bit quantisation of networks as an approximation or optimisation problem. Approximation-based methods confront the gradient mismatch problem, while optimisation-based methods are only suitable for quantising weights and can introduce high computational cost during the training stage. In our large-scale visual computing platform, we provide a simple and uniform way for weights and activations quantisation by formulating it as a differentiable non-linear function. As shown in Fig. 3, the quantisation function is formed as a linear combination of several Sigmoid functions with learnable biases and scales. In this way, the proposed quantisation function can be learned in a lossless and end-to-end manner and works for any weights and activations in neural networks, thereby avoiding the gradient mismatch problem. It can further be trained via continuous relaxation of the steepness of the Sigmoid functions (shown in Fig. 4).

4. Cognition

4.1 System overview

City management involves a lot of data resources. Video data, with its intuitive, mass, and real-time characteristics, is an important part of the data resources of the city. The traditional way of city patrolling mainly relies on laborious manual monitoring. In contrast, through the processing and analysis of massive video, the cognition system can not only obtain the running status of the urban public area in real time, but also detect abnormal events in specific areas in time. According to the architecture shown in Fig. 5, the system consists of three main stages: visual data access stage, multimedia processing stage, and visual algorithm application stage.

In the visual data access stage, video resources from different manufacturers are accessed through standard video protocols. The system has the ability to access large-scale video data based on the cloud platform, which meets the demand of comprehensive city cognition. The accessed data includes online video streams, offline video files, and static images, which will further be preprocessed and transcoded at the multimedia processing stage.

In the multimedia processing stage, visual data is transmitted to the system through the local area network of a city. The large-scale video and images are decoded, transcoded, or preprocessed in this stage. Furthermore, this stage also collects parameters of video sources including camera position and alarm configurations to comprehensively manage multimedia information.

In the visual algorithm application stage, the all-time all-area cognition system integrates fundamental tasks such as image recognition, object detection, object tracking, scene recognition, and anomaly detection. These tasks are formed into independent modules to support top-level algorithm applications. Specifically, traffic accident perception integrates image recognition, object detection, and object tracking tasks. The road congestion perception involves object detection and object tracking tasks. The sudden violence event perception is based on scene recognition and anomaly detection tasks. And the object detection and anomaly detection tasks are utilised to achieve the alarming of persons and vehicles in restricted area. Based on the aforementioned rich top-level visual algorithm applications, the system is further applied to a variety of public scenes in the city, such as transportation, subway, campus, and community.
4.2 Key technical details

The all-time all-area city cognition system pursues a precise understanding of a variety of scenarios. It recognises what is on the road and what is happening on the road before making decisions or alarms. In this section, we will introduce our object detection and anomaly detection methods deployed in this system.

4.2.1 Object detection and tracking: Object detection is one of the core tasks in cognition problems. In the cognition system, detecting objects on the road, such as vehicles and pedestrians, is the primary step for perception applications. Therefore, the high accuracy of the detection algorithm is a prerequisite for subsequent applications. We have devoted great efforts in object detection research.

For vehicle detection, we proposed a scheme, which is illustrated in Fig. 6, based on multi-task deep convolutional neural networks (CNN), region-of-interest (RoI) voting, and multi-level localisation, denoted by RV-CNN [1]. In the design of CNN architecture, we enriched the supervised information with subcategory, region overlap, bounding-box regression, and category of each training RoI as a multi-task learning framework. This design allows the CNN model to share visual knowledge among different vehicle attributes simultaneously, and thus, detection robustness can be effectively improved. We introduced the subcategory classification task to enforce the CNN model to learn a good representation for vehicles under different occlusions, truncations, and viewpoints. In addition, we utilised the CNN model to predict the offset direction of each RoI boundary toward the corresponding ground truth. Then, each RoI could vote those suitable adjacent bounding boxes, which are consistent with this additional information. For clarity, suppose a predicted box has coordinates $b_i = (x_i, y_i, x'_i, y'_i)$ and score $s_i$. Then we formulate the voting scheme as

$$s' = s + \lambda \sum_{b \in (i, r, d)} R_k(b, b_i)$$

(1)

where

$$R_k(b, b_i) = \begin{cases} s_i & \text{if } x_i < x'_i \text{ and } D'_i = \text{go to left}, \\ -s_i & \text{if } x_i < x'_i \text{ and } D'_i = \text{go to right}, \\ -s_i & \text{if } x_i > x'_i \text{ and } D'_i = \text{go to left}, \\ s_i & \text{if } x_i > x'_i \text{ and } D'_i = \text{go to right}. \end{cases}$$

(2)

Other $R_k(b, b_i)$ functions follow the same rule as $R_k(b, b_i)$. After the scores of all predicted boxes are computed again. The voting results are combined with the score of each RoI itself to find a more accurate location from a large number of candidates.

For pedestrian detection, we introduced a preview block [8] which previews the objectness probability for the potential regression region of each prior box, using the stronger features with larger receptive fields and more contextual information for better predictions. The proposed previewer blocks preselect regions with high confidences containing objects by involving enough contextual information. The detector then classifies and relocates the prior boxes in these regions. In addition, we introduced a new metric intersection of ground-truth (IoG) ratio to formulate the containment relations between the predictor region and ground-truth bounding boxes.

$$\text{IoG}_{ij} = \frac{\text{area}(P_{ij} \cap \text{GT}_b)}{\text{area}(	ext{GT}_b)}$$

$$\text{status}_{ij} = \begin{cases} 1, & \text{if } \text{IoG}_{ij} = 1 \text{ and } \text{IoG}_{ij} < 1, \forall j = 1, \ldots, I - 1 \\ 0, & \text{otherwise} \\ -1, & \text{if } \text{IoG}_{ij} < 0.8 \end{cases}$$

where $N$ is the number of ground-truth objects. An object is completely contained by the previewer region when $\text{IoG} = 1.0$, and we assign a positive label to this region. A previewer region will get a negative label if $\text{IoG} < 0.8$. Furthermore, the label of a larger region which contains an object is set to be ignored (neither positive nor negative during training) when that object is already contained in a smaller previewer region. With the previewer blocks, plenty of small-scale false positives were eliminated during the inference process and we've got an effective performance on pedestrian detection.

Besides, we use the renown kernelised correlation filters [9] for multiple objects tracking based on object detection results. Object tracking effectively maps the corresponding detected objects between different frames. Combined with object detection, object tracking module first illustrates the trajectories of vehicles and pedestrians over a period of time, and then identifies target behaviours.

4.2.2 Event detection: Anomalous events detection in real-world video scenes is a challenging problem due to the complexity of ‘anomaly’ as well as the cluttered backgrounds, objects and motions in the scenes. Most existing methods use hand-crafted features in local spatial regions to identify anomalies. We proposed a Spatio-Temporal AutoEncoder (ST AutoEncoder or STAE) [2], which utilises deep neural networks to learn video representation automatically and extracts features from both spatial and temporal dimensions by performing three-dimensional (3D) convolutions. Fig. 7 shows the details of the framework: an encoder followed by two branches of decoder for reconstructing past frames and predicting future frames, respectively.

In addition to the reconstruction loss used in existing typical autoencoders, we introduced a weight-decreasing prediction loss for generating future frames, which enhances the motion feature learning in videos. Specifically, the reconstruction branch and the prediction branch share the same hidden feature layer but perform different tasks: reconstructing the past sequence and predicting the future sequence, respectively. The prediction task guides the model to capture the trajectory of moving objects and enforce the encoder to better extract the temporal features. The prediction loss is formulated by:
Based on the acquisition, integration, and analysis of big and heterogeneous data generated by a diversity of sources in urban spaces, the City Brain can optimise the flow of vehicles and traffic management. In the data perceptron stage, data from various sources in urban spaces and departments are collected and analysed. First is the video data, including general video streams and bayonet camera streams. Traffic accidents (collision, jam etc.) and traffic parameters (road traffic flow, traffic light status, traffic volume and speed in particular lanes etc.) are generated from these video streams. For map data, high-definition map with road network topology, origin-destination data, floating car data, and reported temperature data. Meteorological data mainly contains the weather and status. This is an open access article published by the IET under the Creative Commons Attribution -NonCommercial License.

5 Decision and optimisation
5.1 System overview

Based on the acquisition, integration, and analysis of big and heterogeneous data generated by a diversity of sources in urban spaces, the City Brain can optimise the flow of vehicles and traffic signals, and upgrade the city governance and decision-making on traffic command and road construction. The whole decision and optimisation system are depicted in Fig. 8, which consists of three main stages: the data perceptron stage, the data fusion stage, and the decision and optimisation stage. In the data perceptron stage, data from various sources in urban spaces and departments are collected and analysed. First is the video data, including general video streams and bayonet camera streams. Traffic accidents (collision, jam etc.) and traffic parameters (road traffic flow, traffic light status, traffic volume and speed in particular lanes etc.) are generated from these video streams. For map data, high-definition map with road network topology, origin-destination data, floating car data, and reported incidents from the public are collected. For structured traffic data, SCATS data, induction coil data, and bayonet car-passing data are collected. Meteorological data mainly contains the weather and temperature data. Road administration data consists of information about road infrastructure, road marking, and road construction status.

In the data fusion stage, the first layer contains multi-modality data fusion module and data quality management module. For multi-modality data fusion, AI is adopted to merge all structured summaries of data from the perceptron stage into a single-center data platform. Besides, the data quality management module filters out invalid data, reduces replicated data, and completes missing data based on the synthesis of information from different sources. The second layer is about unifying traffic evaluations, traffic parameters, and traffic representations. Unified traffic evaluations consist of flow speed, delay, line length etc. Unified traffic parameters include lane parameters, intersection parameters, road parameters, and area parameters. Unified traffic representations are map representations, video representations, and structured traffic representations. In the decision and optimisation stage, based on the unified summaries of structured traffic data, intelligence algorithms are adopted for traffic signal optimisation, traffic organisation optimisation, traffic guidance, traffic command and dispatch. For traffic signal optimisation, traffic light timing schedule is dynamically adjusted to improve mobility of an intersection, road or area. For traffic organisation optimisation, the system tries to optimise the spatial distribution and function configuration of the city road network. For traffic guidance, the quickest outgo routes are planned for the public in order to avoid traffic incidents or traffic jams. Specifically, when faced with emergencies, by integrating and analysing real-time data, the system can optimise urban traffic flow such as by identifying the quickest route for emergency vehicles to arrive at the scene within the shortest time frame. For traffic command and dispatch, the system automatically performs traffic accidents reporting, monitoring, and disposition. More importantly, all the traffic patrolmen are dynamically dispatched for each accident, which improves the efficiency of traffic management.

Based on the aforementioned descriptions, we can see that this system can be applied to many scenarios for city management, such as city traffic monitoring, traffic flow guidance, city road construction planning etc.

5.2 Key technical details

5.2.1 Real-time road traffic prediction with spatial–temporal correlations: The spatiotemporal relationship is an essential aspect of road traffic prediction. The fundamental observation is that the traffic condition at a link is affected by the immediate past traffic conditions of some number of its neighbouring links. A time lag function defines how traffic flows are related in the temporal dimension. In parallel, the spatial structure defines which neighbouring links have an effect on the traffic characteristics of other links, as a function of road type, speed, etc.

We have a new method which provides a complete description of the most important spatiotemporal interactions in a road network while maintaining the estimatability of the model [10]. It improves upon existing methods proposed in the area and provides high accuracy on both urban and expressway roads. We adopt a multivariate spatial–temporal autoregressive (MSTAR) model to account for transient behaviour on the traffic network. The standard Vector-ARMA(p,q), or VARMA(p,q), model is

\[ L_{\text{pred}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T} \sum_{t=1}^{T} (T - t) \| X_t + t - f_{\text{pred}}(X)^T \|_2^2 \]  

where \( X_t \) is the input hyper-cuboid, \( f_{\text{pred}}(X) \) is the output of the prediction branch, \( X_{t+r} \) is the ground truth of the future \( T \) frames and the superscript \( r \) in \( X' \) is the \( r \)th frame of the video clip \( X \). The \( r \)th frame has a weight of \( T - r \), which decreases as \( r \) increases.

With the anomaly detection framework, the all-time all-area patrolling and alerting system can detect abnormal events in a variety of scenarios in real time, and then notify the city manager in the form of alarms. Real-time alarm anomaly events through video surveillance can help the government officials to quickly detect and even prevent abnormal emergencies, ensuring the public safety and operation efficiency of a city.
5.2.2 Vehicular traffic prediction with link interactions and multiple data sources: In order to estimate a vehicle arrival time, we invent a system which receives information representing prior travel times of vehicles between pre-determined vehicle stops along a vehicle route [11, 12]. The system comprises a memory device and a processor being connected to the memory device. The system receives information representing prior travel times of vehicles between vehicle stops along a vehicle route. The system receives real-time data representing a current journey. The current journey refers to a movement of a vehicle currently traveling along the route. The system calculates a regular trend representing the current journey based on the received prior travel times information and the received real-time data. The system computes a deviation from the regular trend in the current journey. The system determines a future traffic status in subsequent vehicle stops in the current journey. The system estimates, for the vehicle, each arrival time of each link, that corresponds best, on average, to the relevant neighbouring links during the period. To that requirement, we make use of decomposition of time into intervals, or templates, \( r = 1, \ldots, R \), that permit combining time periods into like sets.

Furthermore, we make use of the data history to induce not only a set of mean values for the speed and volume but in parallel a set of spatial matrices. In other words, each reference period, \( i = 1, \ldots, I \), has associated with it a spatial correlation matrix which corresponds best, on average, to the relevant neighbouring links during the period. The resulting parsimonious transient model is thus defined as

\[
\sum_{l=1}^{P} \sum_{r=1}^{R} \Phi_{lir} S_{lir} Y_{l,r-1} = c_i + \sum_{j=1}^{J} \sum_{r=1}^{R} \Theta_{jr} S_{jr} Y_{j,r-1},
\]

The proposed traffic prediction algorithm is implemented and tested against the actual traffic volume/speed over a medium size road network on real-time basis. The road network consists of 502 links (149 category A, 246 category B, 29 category C, 38 category D, 22 category E, and 18 slip-road). The forecast up to one hour ahead is issued every 5 min using the most recent actual traffic data.

5.2.3 Providing navigational guidance using the states of traffic signals: We invent a method and apparatus by which vehicular traffic prediction can be calculated both accurately and faster than using conventional methods and can be used in the presence of missing real-time data [13]. The missing data is estimated using a calibration model comprising of historical data that can be periodically updated, from select links constituting a relationship vector.

The missing data can be estimated off-line whereafter it can be used to predict traffic for at least a part of the network, the traffic prediction being calculated by using a deviation from historical traffic on the network. The invention further discloses a method for in-vehicle navigation; and a method for traffic prediction for a single lane.

First, as shown in step 101 (Fig. 9), one must perform a division of time and space into, preferably, relatively homogeneous subsets.

An example of dividing time into relatively homogeneous intervals is to consider each day of the week and each hour of the 24-hour day separately. As regards to spatial decomposition, the network in the exemplary embodiment is also divided into links included in the network. In step 102 a relationship vector for every network link to be predicted is defined. The relationship vector for each link contains the other links of the network whose traffic has an impact on that link. Once these steps are performed, the next step 103 of the method exemplarily described herein is to compute off-line average-case estimates of the traffic for each link and for each time period.

This method provides an exemplary technique for determining the traffic state characteristics (e.g. speed, density, flow, etc.) that best characterise the progression of that state into the future.

6 Search and mining
6.1 System overview

In ‘Search and Mining’ system, we aim to put everything the cameras have seen into a database, thus we can apply search on these indexed data. Towards this end, we propose a progressive video search engine to localise objects, such as missing people and hit-and-run vehicles, among the tremendous volume of videos quickly through progressive human–machine interactions. The architecture of the progressive video search engine is shown in Fig. 10. The system consists of three major stages, including stream accessing stage, visual structuring stage, and large-scale visual search stage. Many related technologies are used in this progressive video search engine, among which are video content indexing, retrieval, and attribute recognition algorithms.

In the visual search stage, we build a database to visually index the sensor data of the city, including various cameras, MAC signals, GPS signals, Internet data, etc. Specifically, the visual data from different manufacturers is accessed through standard video protocols. Based on the cloud platform, unified resource schedule, comprehensive analysis as well as reliable storage can be easily realised. The obtained data is then fed into the visual structuring stage to transform into unified standard structured data. In the visual structuring stage, we use deep learning algorithms to analyse the information of pedestrians, non-vehicles, and vehicles based on real-time video content captured from cameras deployed in the city. Specifically, object detection, scene recognition, and attribute recognition algorithms are employed to extract the perceived objects (i.e. pedestrians, non-vehicles, vehicles, and events) and the corresponding attribute features. For example, we consider gender, age, and clothing style for pedestrians and color, type, and moving direction for vehicles. The generated unified standard structured data is used to finally support various applications of the ecosystem through the search engine. In the visual search stage, we build a database to visually index the whole city and a large-scale search engine for city object retrieval. Generally, there are two phases here. In the first phase, the representative features from the pixels are effectively extracted and stored in the database. In the second phase, the queries, i.e. high-dimensional features calculated from a query image, are fed into the database. The accuracy and recall of the search process is
guaranteed with the help of effective indexes combined with high-dimensional global and local features. It is worth noting that challenges may arise in real-world scenarios. For instance, performance loss would certainly appear due to data expansion in both volume and dimension. In order to tackle such challenges, different indexing structures, including M-tree, R-tree, k-d tree etc. should be implemented on top of the database. Furthermore, the proposed search engine performs searching with great efficiency, where a single query among hundreds of billions of images can be executed within one or several hundred milliseconds.

Based on the introduced architecture, the progressive video search engine is widely applied in various scenarios of the city, such as security, transportation, environmental protection, and community service.

6.2 Key technical details

Person ReID is at the core of progressive video search engine. Given a query person, the task aims at matching the same person from multiple non-overlapping cameras. Compared with other image search tasks, person ReID is still very challenging due to the following reasons: (1) dramatic background variations caused by different images from different cameras, (2) significant variations in visual appearance caused by changes in human pose across time and space, and (3) clutter or occlusions. In this section, we will introduce our efforts in image-based person ReID, video-based person ReID, and large-scale similarity search.

6.2.1 Image-based person ReID: We first propose a novel deep Siamese architecture [3] based on CNN and multi-level similarity perception. According to the distinct characteristics of diverse feature maps, we effectively apply different similarity constraints to both low-level and high-level feature maps, during the training stage. Fig. 11 shows the overall architecture of the proposed network at the training stage. Our network can efficiently learn discriminative feature representations at different levels, which significantly improves the ReID performance. Besides, the proposed framework has two additional benefits. First, classification constraints can be easily incorporated into the framework, forming a unified multi-task network with similarity constraints. For concrete demonstration, we separately optimise similarity constraints on low-level feature map (e.g. Pool1 layer) and high-level feature map (e.g. FC7 layer). In the meanwhile, softmax loss is also utilised to optimise classification constraints. Second, as similarity comparable information has been encoded in the network's learning parameters via back-propagation, pairwise input is not necessary at test time. That means we can extract features of each gallery image in an off-line manner and combine with the indexing techniques to further improve the retrieval efficiency, which is essential for large-scale real-world applications. Experimental results on two large data sets CUHK03 [14] and Market-1501 [15] demonstrate that our method outperforms the current state-of-the-art approaches by large margins, and we also achieve competitive performance on the small-size data set CUHK01 [16].

Since the human body consists of well-defined parts, i.e. head, torso and legs, a better approach to solve the various appearances caused by pose changes and local differences are part-based models. To merge the global and local features, we propose a set of local operations as a generic family of building blocks for handling various poses, occlusions, and detection localisations. We refine the temporal weights to the sub-feature level for handling various poses, occlusions, and detection localisations within the sequence.

Extensive ablation studies verify the effectiveness of feature disentangling as well as temporal re-weighting. The experimental results on the iLIDS-VID [20], PRID-2011 [21], and MARS [22] data sets demonstrate that our proposed method outperforms existing state-of-the-art approaches.

6.2.2 Video-based person ReID: Video-based person ReID plays an important role in video analysis, expanding image-based methods by learning features of multiple frames. We propose an attribute-driven method [19] for feature disentangling and frame re-weighting. The features of single frames are disentangled into groups of sub-features, each corresponds to specific semantic attributes. The sub-features are re-weighted by the confidence of attribute recognition and then aggregated at the temporal dimension as the final representation. By means of this strategy, the most informative regions of each frame are enhanced and contribute to a more discriminative sequence representation. An example of our proposed method is shown in Fig. 12. The feature of one frame is disentangled into several sub-features corresponding to specific semantic attribute groups. In the displayed image sequences, frame-1 captured clear frontal face so it has a higher weight in Head group. While the bag is invisible in frame-1, the weights of Bag groups are mainly concentrated on frame-2 and frame-3. Frame-2 also has the highest weight in Shoes group. The weights of frame-T are relatively low because of the poor detection bounding box and clutter background. The re-weighted sub-features are aggregated at the temporal dimension and then concatenated as the representation of the input sequence. We re-estimate the temporal weights to the sub-feature level for handling various poses, occlusions, and detection localisations within the sequence.

6.2.3 Large-scale similarity search: Visual structuring stage helps to obtain feature representations (i.e. high-dimensional features) for a large number of pedestrians, non-vehicles, and vehicles in the whole city. Then we need to construct a large-scale retrieval system for efficient similarity search and clustering of dense vectors. To tackle the challenge of ultra-efficient high-dimensional similarity search, we propose a high queries-per-second (QPS) vector search engine, namely CrazySearch. CrazySearch operates in fast register memory and is flexible enough to be fusible with other kernels. Similar with Faiss (https://github.com/facebookresearch/faiss/wiki), we apply coarse quantisation based on product quantisation (PQ), that enables a nearest neighbour implementation that is 8 x faster than prior state-of-the-art methods. Our implementation enables the k-NN search
Furthermore, it slightly improves the search accuracy, as encoding the residual is more precise than encoding the vector itself.

7 Prediction and intervention

7.1 System overview

Based on the cognition of the city data mentioned above, further prediction and intervention are important in many smart city application scenarios. Different from the previous system, we project multi-modal data into 3D models for global and comprehensive prediction and intervention. The system architecture is shown in Fig. 13, which is mainly divided into data access stage, data processing stage, algorithm stage, and application stage.

The data access stage consists of two parts: static offline data and dynamic real-time data. Offline data is mainly used to reconstruct city scenes, such as a square, a building and its surroundings. Offline data mainly includes aerial pictures taken by unmanned aerial vehicles and Internet photos, as well as design drawing data of buildings such as Computer-Aided Design (CAD) and Building Information Modelling (BIM). Real-time data mainly includes surveillance videos and extensive IoT sensor data.

The data processing stage is designed to process and analyse the aforementioned data. Three-dimensional models of city scenes can be obtained from image data and design drawings based on 3D reconstruction and scene modelling techniques. Utilising the computer vision technology mentioned above, intelligent analysis such as detection, tracking, crowd counting, and anomaly detection on objects are performed in surveillance videos. For different application scenarios, IoT sensor devices complement the perception with other information besides visual information, such as temperature, humidity, smoke, and so on.

In order to realise the global perception of a city, the perceived operating status of the city from the 2D videos are mapped to the 3D scene in real time through the coordinate mapping algorithm. Thereafter, crowd counting and forecasting are allowed for specific 3D spaces. Moreover, road planning can be adjusted based on the directional analysis of traffic flow and crowd flow. In addition, emergency plans are obtained in advance by simulation in the constructed virtual scenes. These algorithms can provide service in various application scenarios such as public security, fire protection, subway, and campus.

7.2 Key technical details

The most important problems needed to be issued in this system is reconstructing digital city scenes. As aforementioned, city scenes can be modelled by images or CAD/BIM data, and the former will be introduced in the following section. For the algorithm stage, we will also present a graph-based method to predict traffic and pedestrian flow.

7.2.1 Digital city modelling

Image-based 3D reconstruction is a widely studied problem [25], and the main procedure is shown in Fig. 14. Given a set of images taken around the target scene, the first step is matching features for each image pair. There have been various algorithms to detect and describe image local key-points, which is divided into two categories: hand-crafted methods [26, 27] and neural network methods [28, 29]. After filtering out the error matches by Ransac [30], we can extract the points tracks in the scene. Each track is a set of feature points from different views corresponding to the same physical point. The next step is to figure out the 3D position of each track together with the intrinsic/extrinsic parameters of each view. The optimisation is performed iteratively and the most classical algorithm is bundle adjustment [31], which is extended in the following years [32–34]. Given the sparse point cloud, Multi-View Stereo [33] are utilised to reconstruct a depth map for each view and generate a dense point cloud. Finally, the whole model is produced by mesh reconstruction and texturing.

Although image-based 3D reconstruction has been successfully applied for modelling various objects, there still remains some problems in the large-scale city scenes. The first problem is the among billions of images with approximately tens of thousands of QPS. Specifically, a single query delay is ~10 milliseconds. Moreover, we adopt an elastic mechanism for expansion, which can flexibly expand the distributed systems cluster to handle the massive volume of data. We apply CrazySearch in the progressive video search scenarios of city brain. The key techniques used in CrazySearch is coarse quantiser.

An exhaustive comparison of the query vector with all vectors is impractical for very large data sets. The coarse quantiser [23] is designed for non-exhaustive search. It retrieves a candidate set first, then searches within the candidate set for nearest neighbours based on PQ [23]. We introduce a modified inverted file structure [24] to rapidly access the most relevant vectors. A coarse quantiser is used to implement this inverted file structure, where vectors corresponding to a cluster (index) are stored in an associated list. The vectors in the list are represented by short vectors generated by the product quantiser, which encodes the residual vector with respect to the cluster center. This approach significantly accelerates the search at the cost of a few additional bits/bytes per descriptor.
Fig. 15 Framework of the proposed DST-GCNN, which contains two stream. The first stream predicts the dynamic traffic conditions and the second predicts the future flow

data scale. Thousands of photos are taken into calculation for reconstructing a campus-sized place. The feature matching and parameter optimising are extremely time-consuming in such data scale, which can be addressed by the feature indexing and calculation acceleration technology we introduced before. Another problem is that the moving city objects (people, vehicles) appearing in videos need to be projected into the 3D model for global and comprehensive prediction and intervention. A direct approach is involving the surveillance images into reconstruction to obtain the intrinsic/extrinsic parameters of each camera. In order to deal with the cluttered video frames, background extraction is performed first. The domain gap between video frames and photos should also be taken into consideration when selecting the feature descriptors.

7.2.2 Flow prediction: Accurate prediction for crowd and traffic flow is the basis of intervention. For example, traffic prediction is important for the adjustment of the traffic light. However, accurate traffic forecast is a challenging problem due to the large-scale problem size, as well as the complex and dynamic nature of spatiotemporal dependency of traffic flow.

Most existing graph-based CNNs attempt to capture the static relations while largely neglecting the dynamics underlying sequential data. We proposed a dynamic spatiotemporal graph-based CNNs (called DST-GCNN) [36] by learning expressive features to represent spatiotemporal structures and predict future traffic from the historical traffic flow. In particular, DST-GCNN is a two-stream network. In the flow prediction stream, we present a novel graph-based spatiotemporal convolutional (STC) layer to extract features from a graph representation of traffic flow. Then several such layers are stacked together to predict future traffic over time. Meanwhile, the proximity relations between nodes in the graph are often time variant as the traffic condition changes over time. To capture the graph dynamics, we use the graph prediction stream to predict the dynamic graph structures, and the predicted structures are fed into the flow prediction stream.

The overview of the proposed framework is shown in Fig. 15. The network consists of two streams, the first stream predicts the dynamic traffic conditions which are encoded in an affinity matrix. The second stream, equipped with the predicted traffic conditions and the proposed STC layers, first predicts future flow from $t + 1$ to $t + T_F$, then predicts the target future flow at $t + T_F$.

Predicting the dynamic graph enables DST-GCNN to adapt to the fast-varying traffic condition. In the future, we plan to apply the proposed framework to other traffic prediction tasks like pedestrian crowd prediction.

8 Practices of the city brain

Powered by Alibaba Cloud’s large-scale computing engine Apsara, City Brain offers a comprehensive suite of acquisition, integration, and analysis of big and heterogeneous data generated by a diversity of sources in urban spaces. The power and functionality of the City Brain enable urban planners and city officials to upgrade their city governance and decision-making to turn the city into an intelligent one. A few current deployment cases of City Brain are listed as follows:

Xiong'an District: On 8, November 2017, Alibaba signed a strategic cooperation agreement with Xiong'an New District to plan and design the future city through the City Brain.

Chongqing: Alibaba creates an Intelligent Chongqing based on the City Brain, driving smart cities, smart manufacturing, and smart services.

Macau: Since 2017, the City Brain has improved the livelihood and visitor experience of Macao through smart services.

Guangzhou: The real-time scheduling of City Brain enabled Baiyun Airport to increase the dispatch usage rate of the parking space by 73%.

Malaysia: The City Brain will be applied to Malaysia’s transportation management, urban planning, environmental protection etc. and in the first phase, it had been used to alleviate congestion in Kuala Lumpur.

Shanghai: The City Brain is widely applied for protecting public safety and providing community service. By optimising traffic light timing strategy, the average travel time dropped by 8% and the roadway congestion index dropped by 15%.

Hangzhou: By building city traffic index and optimising the traffic light timing strategy, the ambulance response time dropped by 50% and the average travel time dropped by 15.3%. Moreover, the accuracy of traffic incident real-time detection reaches 95%. The formalisation of co-operation between Alibaba Group and Sports Bureau of Zhejiang Province provides an opportunity to build the intelligent engine for Hangzhou 2022 Asian Games.

Suchou: Dynamic adjustment of bus departure time increased the number of people taking buses by 17%.

Qichou: With progressive video search, we located 50% more people than before. We are able to locate people with only one photo, even a photo of a person’s back.

Wuchen: The City Brain comprehensively escorts the fourth World Internet Conference.

9 Conclusion

In summary, we introduced the City Brain project, which aims at extracting meaningful and irreplaceable values from an aggregate of a huge amount of heterogeneous data, with a focus on city-scale AI technologies and applications. Current new technologies empower AI and enable us to create city brain. As a platform, the proposed city brain can incubate, hasten, and solidify many more AI technologies and applications in future. From cognition to optimisation, to decision-making, from search to prediction and ultimately, to intervention, City Brain improves the way we manage the city, as well as the way we live in it.

10 Acknowledgments

Jianfeng Zhang and Xian-Sheng Hua have contributed equally.

11 References

[1] Chu, W., Liu, Y., Shen, C., et al.: ‘Multi-task vehicle detection with region-of-interest voting’, IEEE Trans. Image Process., 2018, 27, (11), pp. 432–441

[2] Zhao, Y., Deng, B., Shen, C., et al.: ‘Spatio-temporal autoencoder for video anomaly detection’. Proc. 25th ACM Int. Conf. on Multimedia, 2017, pp. 1933–1941

[3] Shen, C., Jin, Z., Zhao, Y., et al.: ‘Deep Siamese network with multi-level similarity perception for person re-identification’. Proc. 25th ACM Int. Conf. on Multimedia, 2017, pp. 1942–1950

[4] http://https://github.com/apache/flink/tree/blink

[5] Ewans, R., Nickolas, S.: ‘Effective task scheduling for heterogeneous distributed systems using firefly algorithm’, Int. J. Comput. Sci. Eng., 2015, 11, (2), pp. 132–142

[6] Yang, X., Yu, P.S., Han, J.: ‘Graph indexing: a flexible structure-based approach’. Proc. 2004 ACM SIGMOD Int. Conf. on Management of Data, 2004, pp. 335–346

[7] Yang, J., Shen, X., Xing, J., et al.: ‘Quantization networks’. Conference Computer Vision and Pattern Recognition, 2019

[8] Fu, Z., Jin, Z., Qi, G.-J., et al.: ‘Previewer for multiscale object detector’. Proc. 26th ACM Int. Conf. on Multimedia, 2018, pp. 265–273

[9] Henriques, J.F., Cucchiara, R., Martin, P., et al.: ‘High-speed tracking with kernelized correlation filters’. IEEE Trans. Pattern Anal. Mach. Intell., 2015, 37, (3), pp. 583–596

[10] Min, W., Wuyter, L.: ‘Real-time road traffic prediction with spatio-temporal correlations’, 2011, pp. 606–616
[11] Wanli, M., Wynter, L.: ‘Vehicle arrival prediction using multiple data sources including passenger bus arrival prediction’. US Patent 9,177,473, 2017

[12] Wynter, L., Min, W., Morris, B.G.: ‘Method and structure for vehicular traffic prediction with link interactions and missing real-time data’. US Patent 8,755,991, 2014

[13] Wanli, M., Wynter, L.: ‘Method and apparatus for providing navigational guidance using the states of traffic signal’. US Patent 9,599,488, 2017

[14] Li, W., Zhao, R., Xiao, T., et al.: ‘Deepreid: deep filter pairing neural network for person re-identification’. 2014 IEEE Conf. on Computer Vision and Pattern Recognition, 2014, pp. 152–159

[15] Zheng, L., Shen, L., Tian, L., et al.: ‘Scalable person re-identification: a benchmark’. IEEE Int. Conf. on Computer Vision (ICCV), 2015, pp. 1116–1124

[16] Li, W., Zhao, R., Wang, X.: ‘Human reidentification with transferred metric learning’. Asian Conf. on Computer Vision (ACCV), 2012, pp. 31–44

[17] Yang, J., Shen, X., Tian, X., et al.: ‘Local convolutional neural networks for person re-identification’. Proc. 26th ACM Int. Conf. on Multimedia, 2018, pp. 1074–1082

[18] Zheng, Z., Zheng, L., Yang, Y.: ‘Unlabeled samples generated by GAN improve the person re-identification baseline in vitro’. IEEE Int. Conf. on Computer Vision, 2017, pp. 3774–3782

[19] Zhao, Y., Shen, X., Jin, Z., et al.: ‘Attribute-driven feature disentangling and temporal aggregation for video person re-identification’. Conf. Computer Vision and Pattern Recognition, 2019

[20] Wang, T., Gong, S., Zhu, X., et al.: ‘Person re-identification by video ranking’. 13th European Conf. on Computer Vision, ECCV, 2014, pp. 688–703

[21] Hirzer, M., Beleznai, C., Roth, P.M., et al.: ‘Person re-identification by descriptive and discriminative classification’.17th Scandinavian Conf. On Image Analysis, SCIA, 2011, pp. 91–102

[22] Zheng, L., Bie, Z., Sun, Y., et al.: ‘MARS: a video benchmark for large-scale person re-identification’. 14th European Conf. On Computer Vision, ECCV, 2016, pp. 868–884

[23] Jégou, H., Douze, M., Schmid, C.: ‘Product quantization for nearest neighbor search’, IEEE Trans. Pattern Anal. Mach. Intell., 2011, 33, (1), pp. 117–128

[24] Sivic, J., Zisserman, A.: ‘Video Google: a text retrieval approach to object matching in videos’. Int. Conf. on Computer Vision (ICCV) 2003, 2003, pp. 1470–1477

[25] Agarwal, S., Snively, N., Simon, L., et al.: ‘Building Rome in a day’. 2009 IEEE 12th Int. Conf. on Computer Vision, 2009, pp. 72–79

[26] Bay, H., Tuytelaars, T., Gool, L.V.: ‘Surf: speeded up robust features’. European Conf. on Computer Vision, 2006, pp. 404–417

[27] Lowe, D.G.: ‘Distinctive image features from scale-invariant keypoints’, Int. J. Comput. Vis., 2004, 60, (2), pp. 91–110

[28] DeTone, D., Malisiewicz, T., Rabinovich, A.: ‘Superpoint: self-supervised interest point detection and description’. Proc. IEEE Conf. on Computer Vision and Pattern Recognition Workshops, 2018, pp. 224–236

[29] Zhao, Y., Li, Y., Shao, Z., et al.: ‘LSOD: local sparse orthogonal descriptor for image matching’. Proc. 24th ACM Int. Conf. on Multimedia, 2016, pp. 232–236

[30] Fischler, M.A., Bolles, R.C.: ‘Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography’, Commun. ACM, 1981, 24, (6), pp. 381–395

[31] Triggs, B., McLauchlan, P.F., Hartley, R.I., et al.: ‘Bundle adjustment – a modern synthesis’. Int. Workshop on Vision Algorithms, 1999, pp. 298–372

[32] Agarwal, S., Snively, N., Seitz, S.M., et al.: ‘Bundle adjustment in the large’. European Conf. on Computer Vision, 2010, pp. 29–42

[33] Sibley, D., Mei, C., Reid, I.D., et al.: ‘Adaptive relative bundle adjustment’. Robotics: Science and Systems, 2009, vol. 32, p. 33

[34] Wu, C., Agarwal, S., Curless, B., et al.: ‘Multicore bundle adjustment’. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3057–3064

[35] Goesele, M., Snively, N., Curless, B., et al.: ‘Multi-view stereo for community photo collections’. 2007 IEEE 11th Int. Conf. on Computer Vision, 2007, pp. 1–8

[36] Wang, M., Lai, B., Jin, Z., et al.: ‘Dynamic spatiotemporal graph-based CNNs for traffic prediction’, arXiv preprint:1812.02019, 2018