The ParallelEye Dataset: Constructing Large-Scale Artificial Scenes for Traffic Vision Research

Xuan Li, Kunfeng Wang, Member, IEEE, Yonglin Tian, Lan Yan, and Fei-Yue Wang, Fellow, IEEE

Abstract—Video image datasets are playing an essential role in design and evaluation of traffic vision algorithms. Nevertheless, a longstanding inconvenience concerning image datasets is that manually collecting and annotating large-scale diversified datasets from real scenes is time-consuming and prone to error. For that virtual datasets have begun to function as a proxy of real datasets. In this paper, we propose to construct large-scale artificial scenes for traffic vision research and generate a new virtual dataset called “ParallelEye”. First of all, the street map data is used to build 3D scene model of Zhongguancun Area, Beijing. Then, the computer graphics, virtual reality, and rule modeling technologies are utilized to synthesize large-scale, realistic virtual urban traffic scenes, in which the fidelity and geography match the real world well. Furthermore, the Unity3D platform is used to render the artificial scenes and generate accurate ground-truth labels, e.g., semantic/instance segmentation, object bounding box, object tracking, optical flow, and depth. The environmental conditions in artificial scenes can be controlled completely. As a result, we present a viable implementation pipeline for constructing large-scale artificial scenes for traffic vision research. The experimental results demonstrate that this pipeline is able to generate photorealistic virtual datasets with low modeling time and high accuracy labeling.

I. INTRODUCTION

The publicly available video image datasets have received much attention in recent years, due to its indispensability in design and evaluation of computer vision algorithms [1]. In general, a computer vision algorithm needs a large amount of labeled images for training and evaluation. The datasets can be divided into two types: unlabeled datasets used for unsupervised learning and labeled datasets used for supervised learning. However, manually annotating the images is time-consuming and labor-intensive, and participants often lack professional knowledge, making some annotation tasks difficult to execute. Experts are always sparse and should be properly identified. As we known, the human annotators are subjective, and their annotations should be re-examined if two or more annotators have disagreements about the label of one entity. By contrast, the computer is objective in processing data and particularly good at batch processing, so why not let the computer annotate the images automatically?

At present, most publicly available datasets are obtained from real scenes. As the computer vision field enters the big data era, researchers begin to look for better ways to annotate large-scale datasets [2]. At the same time, the development of virtual datasets has a long history, starting at least from Bainbridge’s work [3]. Bainbridge used Second Life and World of Warcraft as two distinct examples of virtual worlds to predict the scientific research potential of virtual worlds, and introduced the virtual worlds into a lot of research fields that scientists are now exploring, including sociology, computer science, and anthropology. In fact, synthetic data has been used for decades to benchmark the performance of computer vision algorithms. The use of synthetic data has been particularly significant in object detection [4], [5] and optical flow estimation [6]-[8], but most virtual data are not photorealistic or akin to the real-world data, and lack sufficient diversity [9]. The fidelity of some virtual data is close to the real-world [10]. However, the synthesized virtual worlds are seldom equivalent to the real world in geographic position, and seldom annotate the virtual images automatically. Richter et al. [11] used a commercial game engine to extract virtual images, with no access to the source code or the content. The SYNTHIA dataset [12] provided a realistic virtual city as well as synthetic images with automatically generated pixel-level annotations, but in that dataset there lacks other annotation information such as object bounding box and object tracking. Gaidon et al. [13] proposed a virtual dataset called “Virtual KITTI” as a proxy for tracking algorithm evaluation. While this dataset was cloned from “KITTI”, it cannot extend easily to arbitrary traffic networks. Due to the above limitations, new virtual datasets that match the real world and provide detailed ground truth annotations are still desirable.

Manually annotating pixel-level semantics for images is time-consuming and not accurate enough. For example, annotating high-quality semantics with 10-20 categories in one image usually takes 30-60 minutes [14]. This is known as the “curse of dataset annotation” [15]. The more detailed the

This work was partly supported by National Natural Science Foundation of China under Grant 61533019, Grant 71232006, and Grant 91520301.

Xuan Li is with the School of Automation, Beijing Institute of Technology, Beijing 100081, China, and also with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lixuan0125@126.com).

Kunfeng Wang (Corresponding author) is with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: kunfeng.wang@ia.ac.cn).

Yonglin Tian is with the Department of Automation, University of Science and Technology of China, Hefei 230027, China, and also with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Lan Yan is with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Fei-Yue Wang is with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Fei-Yue Wang is with The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the Research Center for Computational Experiments and Parallel Systems Technology, National University of Defense Technology, Changsha 410073, China (e-mail: feiyue@gmail.com).
semantics, the more labor-intensive the annotation process. As a result, many datasets do not provide semantic segmentation annotations. For example, ImageNet [16], [17] has 14 million images, in which more than one million images have definite class and the images are annotated with object bounding box for object recognition. However, ImageNet does not have semantic segmentation annotations. Some datasets provide only limited semantic segmentation annotations. For example, NYU-Depth V2 [18] has 1449 densely labelled images, KITTI [1] has 547 images, CamVid [19], [20] has 600 images, Urban LabelMe [21] has 942 images, and Microsoft COCO [22] has three hundred thousand images. These datasets play an important role in the study of semantic segmentation. However, these datasets cannot be used directly in intelligent transportation, especially in automobile navigation, because the number of labeled images is insufficient and the segmented semantics have different categories. Currently, computer vision algorithms that exploit context for pattern recognition would benefit from datasets with many annotated categories embedded in images from complex scenes. Such datasets should contain a wide variety of environmental conditions with annotated object instances co-occurring in the same scenes. However, the real scenes are unrepeatable and the captured images are expensive to annotate, making it difficult to obtain large-scale, diversified datasets with precise annotations.

In order to solve these problems, this paper proposes a pipeline for constructing artificial scenes and generating virtual images. First of all, we use map data to build the 3D scene model of Zhongguancun Area, Beijing. Then, we use the computer graphics, virtual reality, and rule modeling technologies to create a realistic, large-scale virtual urban traffic scene, in which the fidelity and geographic information can match the real world well. Furthermore, we use the Unity3D development platform for rendering the scene and automatically annotating the ground truth labels including pixel-level semantic/instance segmentation, object bounding box, object tracking, optical flow, and depth. The environmental conditions in artificial scenes can be controlled completely. In consequence, we generate a new virtual image dataset, called “ParallelEye” (see Fig. 1). We will build a website and make this dataset publicly available before the publication of this paper. The experimental results demonstrate that our proposed implementation pipeline is able to generate photorealistic virtual images with low modeling time and high fidelity.

The rest of this paper is organized as follows. Section II introduces the significance of parallel vision and virtual dataset. Section III presents our approach to constructing artificial scenes and generating virtual images with ground-truth labels. Section IV reports the experimental results and analyzes the performance. Finally, the concluding remarks are made in section V.

II. PARALLEL VISION AND VIRTUAL DATASET

Parallel vision [23]-[25] is an extension of the ACP (Artificial systems, Computational experiments, and Parallel execution) theory [26]-[30] into the computer vision field. For parallel vision, photo-realistic artificial scenes are used to model and represent complex real scenes, computational experiments are utilized to learn and evaluate a variety of vision models, and parallel execution is conducted to online optimize the vision system and realize perception and understanding of complex scenes. The basic framework and architecture for parallel vision [23] is shown in Fig. 2. Based on the parallel vision theory, this paper constructs a large-scale virtual urban network and synthesizes a large number of realistic images.

The first stage of parallel vision is to construct photorealistic artificial scenes by simulating a variety of environmental conditions occurring in real scenes, and accordingly to synthesize large-scale diversified datasets with precise annotations generated automatically. Generally speaking, the construction of artificial scenes can be regarded as “video game design”, i.e., using the computer animation-like techniques to model the artificial scenes. The main technologies used in this
stage include computer graphics, virtual reality, and micro-simulation. Computer graphics and computer vision, on the whole, can be thought of as a pair of forward and inverse problems. The goal of computer graphics is to synthesize image measurements given the description of world parameters according to physics-based image formation principles (forward inference), while the focus of computer vision is to map the pixel measurements to 3D scene parameters and semantics (inverse inference). Apparently their goals are opposite, but can converge to a common point: parallel vision.

From the parallel vision perspective, we design the ParallelEye dataset. ParallelEye is synthesized by referring to the urban network of Zhongguancun Area, Beijing. Using OpenStreetMap (OSM), an urban network with length 3km and width 2km is extracted. Artificial scenes are constructed on this urban network. Unity3D is used to control the environmental conditions in the scene. There are 15 object classes in ParallelEye, reflecting the common elements of traffic scenes, including sky, buildings, cars, roads, sidewalks, vegetation, fence, traffic signs, traffic lights, lamp poles, billboards, trees, cyclists, pedestrians, and chairs. These object classes can be automatically annotated to generate pixel-level semantics. For traffic vision research, we pay special attention to instance segmentation, with each object of interest segmented automatically. In addition, ParallelEye provides accurate ground truth for object detection and tracking, depth, and optical flow.

III. APPROACH

Our pipeline for generating the ParallelEye dataset is shown in Fig. 3. Firstly, the OSM data released by OpenStreetMap is used to achieve the correspondence in geographic location between the virtual and real world. Secondly, CityEngine is used to write CGA (Computer Generated Architecture) rules and design a realistic artificial scene, including roads, buildings, cars, trees, sidewalks, etc. Thirdly, the artificial scene is imported into Unity3D and gets rendered by using the script and the shader. In the dataset, accurate ground truth annotations are generated automatically, and environmental conditions can be controlled completely and flexibly.

A. Correspondence of Artificial and Real Scenes

In order to increase the fidelity, we choose to import geographic data from OpenStreetMap. Although Google Maps occupies an important position in geographic information, it is not an open-source software. By contrast, OpenStreetMap is an open-source, online map editing program with the goal of creating a world where content is freely accessible to everyone. In OpenStreetMap, the ways denote a directional node sequence. Each node of the network can connect 2-2000 paths, and then arrive at another node. The road information includes direction, lane number, lane width, street name, and speed limit. Each path can form three combinations: non-closed paths, closed paths, and regions. The non-closed paths correspond to the roads, rivers, and railways in the real world. The closed paths correspond to subway, bus routes, residential roads, and so on. The regions correspond to buildings, parks, lakes, and so on. Based on the properties of OSM data, it is easy to relate the real world to the geographic information of the artificial scene. Fig. 4 shows the real Automation Building of CASIA (Institute of Automation, Chinese Academy of Sciences) and its virtual proxy generated by CGA rules. They are similar in appearance.

B. Generation of Ground-Truth Annotations

As stated above, ground-truth annotations are essential for vision algorithm design and evaluation. Traditionally, the images were annotated by hand. The manual annotation is time-consuming and prone to error. Taking semantic/instance segmentation as an example, it usually takes 30-60 minutes to annotate an image with 10-20 object categories. Besides, manual annotation is more or less subjective, so that different annotators can make different semantic labels for the same image, especially near the object boundaries. Instead of manual annotation, this paper uses Unity3D to automatically generate accurate ground-truth labels. Fig. 5 shows some examples of ground-truth annotations, including depth, optical flow, object tracking, object detection, instance segmentation, and semantic segmentation.

Generating ground truth with Unity3D is accurate and efficient. Semantic segmentation ground truth can be directly
generated by using unlit shaders on the materials of the objects, with each category outputting a unique color. Instance segmentation ground truth is generated using the same method, but assigns a unique color tag to each object of interest. The modified shaders output a color which is not affected by the lighting and shading conditions. Depth ground truth is generated using built-in depth buffer information to get depth data for screen coordinates. The depth ranges from 0 to 1 with a nonlinear distribution, with 1 representing “infinitely distant”. Optical flow ground truth is generated by calculating the instantaneous velocity of moving objects on the imaging plane and using the pixel changes in the image sequence to find the correspondence between the previous frame and the current frame. For any $\Delta t \to 0$, let $\omega = (u, v)$, the optical flow constraint equation is given by

$$ E(x + \Delta x, y + \Delta y, t + \Delta t) = E(x, y, t) + \frac{\partial E}{\partial x} \Delta x + \frac{\partial E}{\partial y} \Delta y + \frac{\partial E}{\partial t} \Delta t + \varepsilon. \quad (1) $$

For any $\Delta t \to 0$, let $\omega = (u, v)$, the optical flow constraint equation is given by

$$ -\frac{\partial E}{\partial t} = \frac{\partial E}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial E}{\partial y} \frac{\partial y}{\partial t} = \nabla E \cdot \omega, \quad (2) $$

where $\omega$ is the optical flow of $E(x, y, t)$.

We generate multi-object tracking ground truth based on four rules: 1) when the object appears within the field of view of the camera, the three-dimensional bounding box of the object is converted to a two-dimensional bounding box; 2) when the object appears or disappears from the image boundary, we perform special handling for the bounding box; 3) we do not draw bounding boxes for objects that have less than 15 pixels in width or less than 10 pixels in height; 4) when occlusion occurs and the occlusion rate is higher than a threshold, we do not draw bounding boxes for the occluded object.

C. Diversity of Artificial Scenes

In order to increase the diversity and fidelity of artificial scenes, we control the parameters in the script, the material, and the simulated environmental conditions. Specifically, the controllable parameters include: 1) number, type, trajectory, speed, and direction of the vehicles; 2) position and configuration of the camera; 3) weather (sunny, cloudy, rainy, foggy, etc) and illumination (daytime, dawn, dusk, etc).

Traditionally, video image datasets are collected by capturing in the real world or retrieving from the Internet. It is impossible to control the environmental conditions and repeat the scene layout under different environments, and thus difficult to isolate the effects of environmental conditions on the performance of computer vision algorithms. By contrast, it is easy to control the environmental conditions in artificial scenes. In this work, we are able to flexibly control the camera’s location, height, and orientation to capture different
Fig. 6. Illustration of the diversity of artificial scenes. Top: Virtual images with illumination at 6:00 am (left) and 12:00 pm (right) in a sunny day. Bottom: Virtual images with weather of fog (left) and rain (right).

IV. EXPERIMENTS

Based on the proposed approach, we construct the artificial scene and configure virtual cameras to capture images from the scene. The virtual cameras can be moving or stationary. For automobile applications, the virtual cameras are installed on moving vehicles. For visual surveillance applications, the virtual cameras are fixed on the roadside or at intersections. The experiments are conducted to verify that the artificial scenes are repeatable and that the camera’s position, height, and orientation can be configured flexibly.

A. Onboard Camera

In this experiment, an onboard camera is configured at a height of 2 meters, mimicking the camera installed on the vehicle roof. There are totally 67 vehicles on the road, including 52 vehicles parking on the roadside (3 buses, 4 trucks, and 45 cars) and another 15 vehicles in motion. We turn the camera orientation from left to right and get five orientations (i.e., -30, -15, 0, 15, and 30 degrees with respect to the lane direction). The distance between two cameras on adjacent lanes is 5 meters. These configurations lead to substantial changes in object appearance. Fig. 7 shows three continuous images captured by the onboard camera.

B. Surveillance Camera

In this experiment, a surveillance camera is installed at an intersection. We rotate the camera and control the rotation speed at 10 degrees per second, and the rotation range is 180 degrees. We also change the camera height, with the lifting speed of 0.1 meters per second and the lifting range of 2-5 meters. Such settings can fully simulate the role of surveillance cameras. Based on this experiment, the artificial scene provides virtual video images for intersection monitoring. Fig. 8 shows images captured by the surveillance camera.
is that it can increase diversity of the ParallelEye dataset. In the experiments, with image resolution of 500*375 pixels for ParallelEye, the pipeline for artificial scene construction and ground truth generation runs at 8-12 fps (frames per second) on a workstation computer. We have collected a total of 31,000 image frames, each of which has been annotated with accurate ground truth. We will build a website and make the dataset publicly available before the publication of this paper.

V. CONCLUDING REMARKS

In this paper, we propose a new virtual image dataset called “ParallelEye”. For that we present a dataset generation pipeline that uses street map, computer graphics, virtual reality, and rule modeling technologies to construct a realistic, large-scale virtual urban traffic scene. The artificial scene matches the real world well in terms of fidelity and geographic information. In the artificial scene, we flexibly configure the camera (including its position, height, and orientation) and the environmental conditions, to collect diversified images. Each image has been annotated automatically with ground truth including semantic/instance segmentation, object bounding box, object tracking, optical flow, and depth.

In the future, we will improve the diversity of ParallelEye by introducing moving pedestrians and cyclists, which are harder to animate. We will increase the scale of ParallelEye. In addition, we will combine ParallelEye and the existing real datasets (e.g., PASCAL VOC, MS COCO, and KITTI) to learn and evaluate traffic vision models, in order to improve the accuracy and robustness of traffic vision models when applied to complex traffic scenes.

REFERENCES

[1] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets Robotics: The KITTI dataset,” International Journal of Robotics Research, vol. 32, no. 11, pp. 1231-1237, 2013.
[2] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, “A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM,” in Proceedings of the International Conference on Robotics and Automation, IEEE, pp. 1524-1531, 2014.
[3] W. S. Bainbridge, “The scientific research potential of virtual worlds,” Science, vol. 317, no. 5837, pp. 472-476, 2007.
[4] J. Marin, D. Vazquez, D. Geronimo, and A. M. Lopez, “Learning appearance in virtual scenarios for pedestrian detection,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 137-144, 2010.
[5] J. Papon and M. Schoeler, “Semantic pose using deep networks trained on synthetic RGB-D,” in Proceedings of the IEEE Conference on International Conference on Computer Vision, IEEE, pp. 774-782, 2015.
[6] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, “Performance of optical flow techniques,” International Journal of Computer Vision, vol. 12, no. 1, pp. 43-77, 1994.
[7] B. McCane, K. Novins, D. Crannitch, and B. Galvin, “On benchmarking optical flow,” Computer Vision and Image Understanding, vol. 84, no. 1, pp. 126-143, 2001.
[8] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, “A database and evaluation methodology for optical flow,” International Journal of Computer Vision, vol. 92, no. 1, pp. 1-31, 2011.
[9] G. Ros, S. Ramos, M. Granados, A. Bakhthiary, D. Vazquez, and A. M. Lopez, “Vision-based offline-online perception paradigm for autonomous driving,” in Proceedings of the IEEE Conference on Applications of Computer Vision, IEEE, pp. 231-238, 2015.
[10] H. Prendergast, K. Gajananan, A. Bayoumy Zaki, A. Fares, R. Molenar, D. Urbano, H. Van Lint, and W. Gomaa, “Tokyo Virtual Living Lab: Designing smart cities based on the 3d internet,” in Proceedings of the IEEE Conference on Applications of Computer Vision, Springer International Publishing, vol. 17, no. 6, pp. 30-38, 2013.
[11] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, “Playing for data: Ground truth from computer games,” in Proceedings of the European Conference on Computer Vision, IEEE Internet Computing, pp. 102C118, 2010.
[12] G. Ros, L. Sellaert, J. Materzynska, D. Vazquez, and A. M. Lopez, “The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 3234-3243, 2016.
[13] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, “Virtual worlds as proxy for multi-object tracking analysis,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 4340-4349, 2016.
[14] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. M. Rehg, “Joint semantic segmentation and 3D reconstruction from monocular video,” in Proceedings of the European Conference on Computer Vision, pp. 703-718, 2014.
[15] J. Xie, M. Kiefel, M.-T. Sun, and A. Geiger, “Semantic instance annotation of street scenes by 3D to 2D label transfer,” 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3688-3697, 2016.
[16] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. F. Li, “Large-scale video classification with convolutional neural networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725-1732, 2014.
[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F.-F. Li, “ImageNet Large Scale Visual Recognition Challenge,” International Journal of Computer Vision, vol. 115, no. 3, pp. 211-252, 2015.
[18] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor segmentation and support inference from RGBD,” in Proceedings of the European Conference on Computer Vision, pp. 746-760, 2012.
[19] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” Pattern Recognition Letters, vol. 30, no. 2, pp. 88-97, 2009.
[20] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, “Segmentation and recognition using structure from motion point clouds,” in Proceedings of the European Conference on Computer Vision, Heidelberg: Springer Berlin Heidelberg, 5302, pp. 44-57, 2008.
[21] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, “LabelMe: A database and web-based tool for image annotation,” International Journal of Computer Vision, vol. 77, no. 1, pp. 157-173, 2008.
[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in Proceedings of the European Conference on Computer Vision, pp. 740-755, 2014.
[23] K. Wang, C. Gou, and F.-Y. Wang, “Parallel vision: An ACP-based approach to intelligent vision computing,” Acta Automatica Sinica, vol. 42, no. 10, pp. 1490-1500, 2016.
[24] K. Wang, C. Gou, N. Zheng, J. M. Rehg, and F.-Y. Wang, “Parallel vision for perception and understanding of complex scenes: Methods, framework, and perspectives,” Artificial Intelligence Review, vol. 48, no. 3, pp. 298-328, 2017.
[25] K. Wang, Y. Lu, Y. Wang, Z. Xiong, and F.-Y. Wang, “Parallel imaging: A new theoretical framework for image generation,” Pattern Recognition and Artificial Intelligence, vol. 30, no. 7, pp. 577-587, 2017.
[26] F.-Y. Wang, “Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications,” IEEE Transactions on Intelligent Transportation Systems, vol. 11, no. 3, pp. 630-638, 2010.
[27] F.-Y. Wang, “Parallel control: A method for data-driven and computational control,” Acta Automatica Sinica, vol. 39, no. 4, pp. 293-302, 2014.
[28] F.-Y. Wang, X. Wang, L. Li, and L. Li, “Steps toward parallel intelligence,” IEEE/CAA Journal of Automatica Sinica, vol. 3, no. 4, pp. 345-348, 2016.
[29] L. Li, Y. Lin, D. Cao, N. Zheng, and F.-Y. Wang, “Parallel learning — A new framework for machine learning,” Acta Automatica Sinica, vol. 43, no. 1, pp. 1-8, 2017.
[30] X. Liu, X. Wang, W. Zhang, J. Wang, and F.-Y. Wang, “Parallel data: From big data to data intelligence,” Pattern Recognition and Artificial Intelligence, vol. 30, no. 8, pp. 673-682, 2017.