Postprint: A Street View Change Detection Method Based on Crowdsensing

Authors: Zhong Weizhao, Chen Huihui

Date: 2022-05-10T11:22:58+00:00

Abstract

Mobile crowdsensing data contains image and spatio-temporal contextual information that can be utilized for detecting changes in street view images; however, such data is typically low-quality and non-standard. To accurately detect changes in street scenes, this paper primarily addresses the data quality issues arising from viewpoint variations during capture. First, to tackle the large parallax problem, an image registration method is employed to preliminarily align images and extract registration feature points. Then, based on the distribution of these registration feature points, Regions of Interest (ROIs) are extracted from the images. Third, to address false positives in difference images, a filtering method based on area and multiple feature points is proposed to eliminate false detection regions. Finally, edge detection and superpixel segmentation algorithms are combined to extract complete changed objects. Compared with the MDFNet method, experimental results demonstrate that when street view changes occur, the proposed method achieves an F1-Measure value of 55.8%, representing a 6% improvement, and an error rate of 10.8%, representing a 24% reduction; when no street view changes occur, the error rate of the proposed method is 2.8%, representing a 28% decrease.

Full Text

Preamble

Vol. 39 No. 9
Application Research of Computers
ChinaXiv Cooperative Journal

Change Detection Method for Street View Images Based on Crowdsensing

Zhong Weizhao[1], Chen Huihui[2]†

[1]School of Electromechanical Engineering & Automation
[2]School of Electronics & Information Engineering
FoShan University, FoShan, Guangdong 528225, China

Abstract: The images and spatiotemporal contextual information contained in mobile crowdsensing data can be utilized to detect changes in street view images. However, crowdsensing data are typically low-quality and non-standard. To accurately detect street view changes, this paper primarily addresses the data quality issues caused by differences in shooting perspectives. First, to tackle large parallax problems, an image registration method is employed to initially align image pairs and extract registration feature points. Second, based on the distribution of these registration feature points, regions of interest are extracted from the images. Third, to eliminate false detections in difference images, a screening method based on area and multi-feature points is proposed to remove erroneous detection regions. Finally, complete changed objects are extracted by combining edge detection and superpixel segmentation algorithms. Compared with the MDFNet method, experimental results show that when street view changes occur, the proposed method achieves an F1-Measure of 55.8% (a 6% improvement) and an error rate of 10.8% (a 24% reduction). When no changes occur, the proposed method' s error rate is only 2.8%, representing a decrease of approximately 28%.

Keywords: mobile crowdsensing; image registration; change detection; street view; electronic map

0 Introduction

With the rapid development of urban and rural construction in China, road infrastructure is constantly evolving, and people' s demands for travel assistance software are becoming increasingly diverse. Various electronic map applications have brought convenience to daily travel, and current mobility has become largely dependent on electronic maps. Building upon this foundation, services such as Google Street View and Baidu Street View leverage virtual reality technology to map panoramic street photographs onto electronic maps, enabling users to virtually traverse the globe from their homes.

Street view imagery not only provides convenience and safety for travel, such as through road defect warnings [1,2], but also offers a platform for urbanization development [3]. Professional street view vehicles are the primary tools for street view image collection and updating, yet the workload for regular updates is substantial. Mobile crowdsensing utilizes mobile smart devices (e.g., smartphones) carried by individuals to sense and collect information, offering low-cost data acquisition with extensive coverage that can be applied to street view image updates [4], road condition monitoring [5,6], and environmental quality detection [7].

Street view change detection based on mobile crowdsensing offers excellent spatiotemporal advantages, including low-cost image collection, rapid updates, and wide coverage [8,9], enabling short-cycle updates of street view maps. However, crowdsensing data suffers from drawbacks such as low quality, redundancy, large variations, and non-standardization [8].

Street view changes are typically detected by comparing differences between photographs taken at different times. While crowdsensing-based collection is more flexible than street view vehicles, photographs are affected by factors such as angle differences and lighting variations, resulting in relatively low dataset quality that makes it difficult to judge street view image changes based on image differences alone. First, because collectors' positions are not fixed, differences between two street view images may be caused by perspective variations, leading to detection failures. Second, image differences manifest at the pixel level, whereas street view changes occur as objects appearing or disappearing, requiring expansion from pixel-level difference information to obtain specific changed objects.

To address these two challenges, this paper' s contributions include: (a) proposing an end-to-end algorithm framework for detecting changed objects in street view image pairs with large parallax, which solves parallax issues through image registration and introduces an algorithm to eliminate false detection information based on multi-feature point spatial distribution; (b) extracting specific changed objects by combining edge detection and superpixel segmentation algorithms according to the continuity of object boundaries; and (c) collecting and constructing a campus street view image dataset using a campus as the test scene, demonstrating the method' s effectiveness through comparison with recent related work.

1 Related Work

Numerous studies have investigated street view change detection methods [10–24], which can be categorized into 2D image change detection and 3D scene modeling detection based on change type, methodology, and application.

2D image change detection represents the primary approach for street view change detection [10–19]. Traditional methods create appearance models of street views from a set of images captured at different times and compare them with newly captured query images to detect changes. The main focus of such research is handling irrelevant appearance changes, such as illumination differences [10]. To more accurately detect real changes from irrelevant appearance variations, Zhao et al. [11] proposed a siamese encoder-decoder network structure for semantic-level change detection. Chen et al. [12] added a dynamic-aware temporal attention module to the encoder-decoder architecture, combined with horizontal and vertical concurrent attention modules to refine detection results.

Other studies have proposed extracting and fusing features from different network layers to detect changes from global to local regions [13].

Extensive research has also addressed perspective differences. Sakurada et al. [14] proposed a change detection method combining convolutional neural network features with superpixel segmentation. Other neural network-based detection methods include: ChangeNet [15], which fuses multi-layer features through combined convolutional and deconvolutional networks to obtain change information at different scales; a street view image change detection method combining semantic segmentation models and graph cuts [16]; MDFNet (Multiple Difference Features Network) [17], which fuses multiple difference features to obtain multi-scale change information; and Mask-CDNet (Mask based Pixel Change Detection Network) [18], which uses optical flow estimation to register image pairs and obtain mask images, then combines original image structural information to derive change regions.

Beyond perspective differences, Guo et al. addressed image variations caused by scaling and rotation by proposing the fully convolutional siamese metric network CosimNet, incorporating threshold-contrast loss to penalize noisy changes [19].

Numerous solutions also exist in the 3D domain [20–22]. These methods collect multiple street view images to construct Multi-View Stereo (MVS) models, detecting changes by comparing differences between new and original models and locally updating the MVS. The accuracy of such comparisons depends on the quality of the available MVS reconstruction. Scene 3D models are typically created using 3D sensors rather than ordinary cameras. Some studies connect multi-sensor fused Simultaneous Localization and Mapping (SLAM) with rapid dense 3D reconstruction pipelines, feeding coarsely registered image pairs to Deconvolutional Networks (DN) for pixel-wise change detection [20]. Other research reprojects images onto another view through 3D models to discover changes between image pairs, using multiple image combinations to resolve ambiguities during detection and estimate the 3D location of changes [21]. These methods are designed for maintaining/updating existing 3D city models.

Another research type uses large-scale multi-view street view images to create spatiotemporal models through structure-from-motion methods. Some studies utilize decades of urban photographs to enable specific types of temporal inference, such as estimating building construction times [23,24]. Conceptually, these methods use Internet image datasets to detect scene changes, such as variations in advertisements and building wall paintings. Assuming sufficient multi-view images of a scene are available, these approaches can reconstruct 3D scene models using SFM (Structure From Motion).

Urban street view image updating based on crowdsensing primarily identifies street view changes through differences in smartphone-captured photographs, also detecting changes by comparing street view images taken by participants at different times. The proposed method belongs to the 2D image change detection category. Unlike traditional detection methods and existing neural network

approaches, this method detects changes in parallax street view image pairs and extracts specific changed objects through multiple image processing modules without requiring extensive data training or semantic segmentation.

2.1 Algorithm Framework

Street view change detection primarily involves difference detection between two photographs with similar shooting contexts (including location and angle), comprising four steps: image preprocessing, change region extraction, change region filtering, and change region expansion, as illustrated in the framework diagram (Fig. 1).

(a) Image Preprocessing. This includes resizing image pairs to the same dimensions, using histogram specification [25] to align their gray-level histograms, and employing image registration algorithms to align street view image pairs. This paper uses the APAP (As-Projective-As-Possible) algorithm [26] based on SIFT (Scale-Invariant Feature Transform) feature points as an example. The smallest rectangular region covering all registration feature points is extracted as the Region of Interest (ROI) for further processing to reduce unnecessary computation.

(b) Change Region Extraction in ROI. This involves obtaining a binary difference map through image subtraction, using a multi-seed region growing method based on thresholds to extract sky regions and address sky region variability [27], applying hole-filling to address pixel loss caused by registration, and using morphological operations to segment change content into multiple sub-regions.

(c) Change Region Filtering. This includes using area-based filtering to remove change sub-regions with excessively small pixel areas and using multi-feature point filtering to remove sub-regions with numerous distributed feature points.

(d) Change Region Expansion and Complete Object Extraction. Finally, change regions are expanded to extract complete changed objects. This involves using edge detection to extract object contours from change regions, applying edge expansion through 4-neighborhood extension to obtain complete object outlines, and combining superpixel segmentation [28] to extract complete changed objects.

2.2 ROI Image Extraction

ROI image extraction aims to segment the image region used for street view change recognition and eliminate other regions, as areas without feature point

distribution are either non-overlapping or unregistered, and operating on such regions would generate substantial unnecessary and erroneous computations.

Let the original street view image be denoted as I_t and the newly submitted image as I. After registering street view images I and I_t, we obtain the transformed image I' mapped from I_t. Since the APAP algorithm [26] is a unidirectional registration algorithm where the target image I does not undergo deformation during registration, the smallest rectangular region in image I covering all registration feature points is used to extract the ROIs of both I' and I, denoted as R_t and R, respectively. R_t and R are sub-images of I' and I. Registration feature points refer to the SIFT feature points used in the APAP algorithm to estimate the homography matrix H.

2.3 Street View Change Region Extraction

The street view change region extraction process consists of four components: image differencing, sky elimination, edge repair, and region connectivity.

After registration, the pixel information at corresponding positions in images R_t and R can be considered aligned. The resulting difference image represents the change image. To simplify computation, color images R_t and R are converted to grayscale using Eq. (1), and the difference between the two grayscale images is computed to obtain the difference image D. Otsu' s method [29] is used to find the optimal threshold for converting image D into a binary image D', where pixel values of 1 represent differing regions between the two images and values of 0 represent identical regions.

$$gy = 0.299 \times r + 0.578 \times g + 0.114 \times b \tag{1}$$

where r, g, and b represent the red, green, and blue channel values, respectively, and gy represents the grayscale value.

Typically, two photographs are captured at different times, and varying weather conditions and clouds may be falsely detected as difference regions. Therefore, sky region elimination is crucial for street view difference detection. Seed region growing based on threshold segmentation is a common method for sky region extraction, but using a single seed point often leads to incomplete sky region segmentation. This paper employs a multi-seed region growing method to extract sky regions [21].

The binary mask images of sky regions extracted from R_t and R are denoted as M_t and M, respectively, where pixel values of 1 represent sky regions and values of 0 represent non-sky regions. After applying hole-filling to M_t and M, the sky portion of the difference map D' is updated using Eq. (2):

$$D'_{x,y} \leftarrow D'_{x,y} \wedge \neg M_{x,y} \tag{2}$$

where $M_{x,y}$ represents the pixel value of image M at position (x, y).

Edge regions of registered images may exhibit pixel blocks with value 1 due to boundary pixel loss after large deformations. Such black pixel blocks may be extracted as change regions after image subtraction, necessitating pixel loss region filling before extracting change content from D'. The filling method is shown in Eq. (3):

$$D'_{x,y} \leftarrow 0 \quad \text{if} \quad D_{x,y} = 0 \wedge (D_{x-1,y} = 1 \vee D_{x+1,y} = 1 \vee D_{x,y-1} = 1 \vee D_{x,y+1} = 1) \tag{3}$$

where $D_{x,y}$ represents the pixel value of D at position (x, y).

In the binary difference image D', change object contours may be unclear, with hollow regions inside and adjacent objects potentially connected, resulting in both black and white point noise. After applying opening and closing operations to eliminate black and white noise, multiple noise-free connected regions are obtained. Finally, connected regions are derived through 4-neighborhood connectivity of binary difference image pixels. These connected regions constitute the initial change regions, with the number of connected regions representing the number of change regions. This set of change regions is denoted as C_1.

2.4 Change Region Filtering

C_1 contains some falsely detected change regions caused by imprecise image registration that must be eliminated. The main types of false detection regions include: (a) small-area false detection regions caused by minor deformations, and (b) false detection regions resulting from occlusion issues caused by perspective differences. For type (a), an area-based filtering method is used; for type (b), a grid-based screening method using multi-feature points is employed. The flowchart is shown in Fig. 2.

2.4.1 Area-Based Filtering Method

This paper focuses on relatively large change regions in street view images, and small-area regions composed of misaligned difference pixels can be ignored. Change regions with excessively small areas can be filtered out, with the filtering threshold denoted as th (in this paper, $th = 0.03$). After removing small-area candidate regions using Eq. (4), a new change region set C_2 is obtained:

$$C_2 \leftarrow \{c \mid c \in C_1, S(c) > th \times S_R\} \tag{4}$$

where $S(c)$ represents the area of change region c, and S_R represents the area of the ROI image.

2.4.2 Multi-Feature Point Filtering Method

Changed regions contain relatively few matching feature points, making the number of matching feature points a useful indicator for determining whether a region represents a true change. However, objects in change regions lack matching feature points, and relying solely on SIFT feature points is insufficient in terms of quantity. Moreover, due to perspective differences, feature descriptors require scale invariance. Therefore, this paper combines SIFT and ORB (Oriented FAST and Rotated BRIEF) features to create a multi-feature point set, increasing both the quantity and diversity of feature points.

To ensure the quality of matching feature points, the RANSAC algorithm is used to filter out incorrect feature point matches. All subsequent feature point extractions employ RANSAC by default.

Since the spatial distribution of matching feature points in images is non-uniform —for instance, if numerous matching feature points are clustered, a larger region containing these clustered points would still have many matching feature points —directly using the number of matching feature points can easily lead to misjudgment. Therefore, this paper adopts a grid-based feature point quantity comparison method.

Because multi-feature point matching is inevitably affected by perspective differences, this paper proposes extracting SIFT and ORB features from image pairs (R_t, R) and (I_t, I) separately and combining them into a new multi-feature set, thereby increasing both feature point quantity and matching "field of view."

The change region set C_2 may contain regions where no actual changes occurred. For any region in C_2, sub-images on R_t and R are extracted and denoted as I_1 and I_2. Each sub-image is divided into a 5×5 grid. The similarity index d between images is represented by the inverse of the number of grids covering matching feature points, as shown in Eq. (5):

$$d = \frac{1}{\max(gn_1, gn_2)} \tag{5}$$

where gn_i represents the number of grids covering matching feature points in image I_i. A larger similarity index indicates greater similarity between images.

The similarity index d_k for sub-image k is calculated using Eq. (6):

$$d_k = \frac{1}{\max(gn_1, gn_2)} \tag{6}$$

If the similarity index is less than threshold thr, the region is removed from C_2. The final filtered change region set is denoted as C_3.

2.5 Change Region Expansion

After filtering change regions, region connectivity and expansion methods are required to obtain complete change regions and extract changed objects. Edge pixels of the same object typically exhibit continuity and similarity. Therefore, starting from edges for connected region expansion, edges are first detected to extract regions containing changed objects in the image, followed by superpixel-based segmentation.

(a) Edge Detection and Connectivity. The Sobel operator is used for edge detection on sub-images of change regions in C_3 to obtain a gradient magnitude map. After removing points with excessively small gradient values (threshold = 80), pixels with non-zero gradient values serve as initial edge points and are placed in set G. Each point in G is then traversed in the original image, and its connected points are gradually added to G. Point A is considered a connected point of point B if three conditions are met: (1) point A is adjacent to point B; (2) in the gradient magnitude map, the difference between gradient values at points A and B is less than threshold th_g (in this paper, $th_g = 50$); and (3) within a 21×21 pixel neighborhood centered at A, there are no matching feature points. The region composed of points in G is the connected region.

(b) Superpixel Segmentation for Complete Object Extraction. Based on the superpixel segmentation method [22], complete changed image content is obtained. First, ns seed points are randomly selected, and the superpixel method segments the original image into ns irregularly shaped pixel blocks. For each pixel point g_i in set G, Eq. (7) calculates the distance to seed points within a radius of q pixels (in this paper, $q = 10$), finding the nearest seed point s_j. The pixel blocks containing g_i and s_j are then merged, and the center point of the merged pixel block serves as a new seed point for the next iteration.

$$\text{distance}(p, q) = \|b_p - b_q\| + k \times \|l_p - l_q\| \tag{7}$$

where b_p represents the RGB color vector of pixel p, l_p represents the position vector of pixel p, and k is a weight coefficient.

After n iterations, the pixel block corresponding to each pixel point in set G represents the final annotated street view change region.

3 Experiments

3.1 Dataset

The experiments used a campus as the street view change detection scenario, recruiting five volunteers to collect data according to specified shooting loca-

tions and poses. Shooting pose includes azimuth and pitch angles. By utilizing smartphone sensors, the data collection app ensured that deviations in azimuth and pitch angles during shooting did not exceed 3 degrees, and GPS positioning distance was within 1 meter. The image collection time span was no more than 3 months. A total of 100 campus street view image groups were collected, covering 15 scenes, with all experimental images uniformly compressed to 640×480 pixels. The 100 street view image groups were divided into two sets: the first set (G1) contained 60 image pairs with street view changes, while the second set (G2) contained 40 image pairs without changes. To evaluate and compare method effectiveness, the dataset was annotated, with 0 (black regions) representing non-change areas and change regions retaining their original color information. Data primarily captured changes in road and building facade objects, such as decorations, obstacles, and appearance/disappearance of vehicles.

3.2 Experimental Parameters

Through extensive experimentation, the parameter set yielding the best results was selected for this study. Some parameters were directly provided in the text and are not reiterated here. In street view change region extraction, the opening operation used a 5×5 convolution kernel, while the closing operation used a 15×15 kernel. In change region filtering, SIFT feature matching parameters were consistent with those in reference [26]; the distance threshold for ORB feature matching was set to 0.9; the ratio threshold for APAP algorithm matching was set to 0.8 (i.e., the ratio of best to second-best matching feature vector distances was less than or equal to 0.8). Additionally, the maximum RANSAC iteration count was set to 90, and the maximum pixel distance between projected and corresponding positions was set to 60. In change region expansion, the initial seed point count ns for superpixel segmentation was 3000, the weight coefficient k was 1.5, and the seed point iteration count n was 10.

3.3 Evaluation Metrics

Four evaluation metrics were adopted: precision, recall, F1-Measure, and error rate. Precision refers to the proportion of correctly detected change region area relative to the total detected change region area within ROI images (the effective detection region, i.e., the overlapping area of two input images). Recall refers to the proportion of correctly detected change region area relative to the ground truth change region area within ROI images. F1-Measure is calculated from precision and recall. Error rate refers to the proportion of incorrectly extracted change region area relative to the ground truth non-change region area within ROI images.

To verify the proposed algorithm' s effectiveness, Otsu' s method [29] combined with image differencing served as the baseline method, and comparisons were made with methods from references [15–17].

3.4 Experimental Results Analysis

When training the ChangeNet model from reference [15] and the MDFNet model from reference [17], in addition to the PCD dataset mentioned in [17], 70 image pairs from our collected data were added (42 pairs from G1 and 28 pairs from G2), with data augmentation applied. The DeeplabV3+ network from reference [18] was trained using the Camvid dataset. The following visualization results were obtained from the model with the smallest test set loss across 50 training epochs.

The average precision, recall, F1-Measure, and error rate of the proposed method on datasets G1 and G2 are summarized in Table 1. For dataset G2, where image pairs contain no changes, precision, recall, and F1-Measure are not applicable and thus not evaluated.

Table 1. Experimental Results

Method	Precision	Recall	F1-Measure	Error Rate
Baseline (G1)				
Reference [15] (G1)				
Reference [16] (G1)				
Reference [17] (G1)				
Proposed (G1)				
Baseline (G2)				
Reference [15] (G2)				
Reference [16] (G2)				
Reference [17] (G2)				
Proposed (G2)				

Results show that for G1 data, the proposed method achieved an F1-Measure of 55.8%, representing improvements of approximately 38%, 24%, 7%, and 6% over the baseline, reference [15], reference [16], and reference [17] methods, respectively. The proposed method' s average error rate was 10.8%, decreasing by approximately 35%, 20%, and 24% compared to the baseline, reference [16], and reference [17] methods, respectively, while increasing by about 9% compared to reference [15]. For G2 data, the proposed method achieved an average error rate of 2.8%, decreasing by approximately 46%, 29%, and 28% compared to the baseline, reference [16], and reference [17] methods, respectively, while increasing by about 2% compared to reference [15].

For more direct comparison, Figures 3–5 present visualization comparisons of three randomly selected image pairs not used in training. When change content exists in image pairs (Figures 3 and 4), comparing detection results (c)(d)(e)(f)(g) with ground truth annotations (h) demonstrates that the proposed method detects changed objects in street view image pairs more accurately than comparative methods. When no change content exists (Figure

5), the proposed method produces no false detections, while other comparative methods (except reference [15]) still show large false detection regions.

Combining metrics and visualization results, the proposed method overall outperforms comparative methods and extracts changed objects more accurately. Reference [15] achieved the lowest error rate and relatively low recall on both datasets, demonstrating accurate detection of local changes but weaker capability in extracting larger changed objects. Although reference [16]' s method exhibits certain robustness to parallax through semantic segmentation, its pixel-wise comparison approach cannot achieve satisfactory results for distortions caused by image registration, and its error rate is higher. Incomplete semantic segmentation correctness in complex street view environments also contributes to this limitation. Reference [17]'s method ranks second to the proposed method; its extracted multi-layer difference features at different scales handle parallax issues reasonably well but exhibit suboptimal detail processing capability and higher false detection rates. The proposed method leverages registration feature point pair distribution information to better handle differences caused by imprecise pixel alignment. Finally, by exploiting the continuity property of object edges and focusing on extracting specific changed objects, the method improves precision, recall, and F1-Measure while reducing error rate.

---

4 Conclusion

This paper proposes a framework and method for detecting street view changes based on crowdsensing data. Experiments conducted on a campus street view image dataset captured by volunteers demonstrate that the method can currently detect street view change content with reasonable accuracy. The proposed method relies heavily on image registration feature point pair distribution information and improves registration accuracy through multi-feature point combination, though this increases computational cost. The APAP image registration algorithm used in this paper warrants further improvement. Future work will investigate how to combine feature point spatial distribution information with lightweight neural networks to better accomplish street view image change detection tasks based on crowdsensing.

References

[1] Huang Jiong, Hu Sheng, Wang Yun, et al. GrabView: a scalable street view system for images taken from different devices [C]// Proc of the 5th International Conference on Multimedia Big Data (BigMM). IEEE, 2019.

[2] Vishnani V, Adhya A, Bajpai C, et al. Manhole detection using image processing on Google Street View imagery [C]// Proc of the 3rd International

Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2020: 684-688.

[3] Xu Chi, Chen Qiao, Liu Jiangchuan, et al. Smartphone-based crowdsourcing for panoramic virtual tour construction [C]// Proc of IEEE International Conference on Multimedia & Expo Workshops (ICMEW). IEEE, 2018: 1-4.

[4] Peng Zhe, Gao Shang, Xiao Bin, et al. CrowdGIS: Updating digital maps via mobile crowdsensing [J]. IEEE Transactions on Automation Science and Engineering, 2017, 15(1): 369-380.

[5] Wang Qianru, Guo Bin, Wang Leye, et al. Crowdwatch: dynamic sidewalk obstacle detection using mobile crowd sensing [J]. IEEE Internet of Things Journal, 2017, 4(6): 2159-2171.

[6] Kong Yingying, Yu Zhiwen, Chen Huihui, et al. Detecting type and size of road crack with the smartphone [C]// Proc of IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). IEEE, 2017, 1: 572-579.

[7] Leonardi C, Cappellotto A, Caraviello M, et al. SecondNose: an air quality mobile crowdsensing system [C]// Proc of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational. 2014: 1051-1054.

[8] Chen Huihui, Guo Bin, Yu Zhiwen. Measures to improve outdoor crowdsourcing photo collection on smart phones [C]// Proc of IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE, 2019: 907-912.

[9] Chen Huihui, Cao Yangjie, Guo Bin, et al. LuckyPhoto: multi-facet photographing with mobile crowdsensing [C]// Proc of ACM International Joint Conference and International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers. 2018: 1865-1873.

[10] Radke R J, Andra S, Al-Kofahi O, et al. Image change detection algorithms: a systematic survey [J]. IEEE Trans on Image Processing, 2005, 14(3): 294-307.

[11] Zhao Xinwei, Li Haichang, Wang Rui, et al. Street-view change detection via siamese encoder-decoder structured convolutional neural networks [C]// VISIGRAPP (5: VISAPP). 2019: 525-532.

[12] Chen Shuo, Yang Kailun, Stiefelhagen R. DR-TANet: dynamic receptive temporal attention network for street scene change detection [C]// Proc of IEEE Intelligent Vehicles Symposium (IV). IEEE, 2021: 502-509.

[13] Lei Yinjie, Peng Duo, Zhang Pingping, et al. Hierarchical paired channel fusion network for street scene change detection [J]. IEEE Trans on Image Processing, 2020, 30: 55-67.

[14] Sakurada K, Okatani T. Change detection from a street image pair using CNN features and superpixel segmentation [C]// BMVC. 2015, 61: 1-12.

[15] Varghese A, Gubbi J, Ramaswamy A, et al. ChangeNet: a deep learning architecture for visual change detection [C]// Proc of European Conference on Computer Vision (ECCV) Workshops. 2018: 0-0.

[16] Li Wenguo, Huang Liang, Zuo Xiaoqing, et al. A street view image change detection method combining semantic segmentation model and graph cuts [J]. GNSS World of China, 2021, 46(01): 98-104.

[17] Zhan Rui, Lei Yinjie, Chen Xunmin, et al. Street-view change detection based on multiple difference feature network [J]. Computer Science, 2021, 48(02): 142-147.

[18] Bu Shuhui, Li Qing, Han Pengcheng, et al. Mask-CDNet: a mask based pixel change detection network [J]. Neurocomputing, 2020, 378: 166-176.

[19] Guo Enqiang, Fu Xinsha, Zhu Jiawei, et al. Learning to measure change: fully convolutional siamese metric networks for scene change detection [J]. ArXiv Preprint ArXiv, 2018, 1810.09111.

[20] Alcantarilla P F, Stent S, Ros G, et al. Street-view change detection with deconvolutional networks [J]. Autonomous Robots, 2018, 42(7): 1301-1314.

[21] Ku Tao, Galanakis S, Boom B, et al. SHREC 2021: 3D point cloud change detection for street scenes [J]. Computers & Graphics, 2021, 99: 1-16.

[22] Taneja A, Ballan L, Pollefeys M. City-scale change detection in cadastral 3D models using images [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition, 2013: 113-120.

[23] Schindler G, Dellaert F. Probabilistic temporal inference on reconstructed 3D scenes [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010: 1410-1417.

[24] Matzen K, Snavely N. Scene chronology [C]// Proc of European Conference on Computer Vision. Springer, Cham, 2014: 615-630.

[25] Coltuc D, Bolon P, Chassery J M. Exact histogram specification [J]. IEEE Trans on Image Processing, 2006, 15(5): 1143-1152.

[26] Zaragoza J, Chin T J, Brown M S, et al. As-projective-as-possible image stitching with moving DLT [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2339-2346.

[27] Zhang Yanli, Ke Xu. Research and implementation of image defogging algorithm combined with sky area detection [J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2017, 34(05): 37-42.

[28] Stutz D, Hermans A, Leibe B. Superpixels: An evaluation of the state-of-the-art [J]. Computer Vision and Image Understanding, 2018, 166: 1-27.

[29] Otsu N. A Threshold selection method from gray-level histograms [J]. IEEE Trans on Systems, Man, and Cybernetics, 1979, 9(1): 62-66.

Note: Figure translations are in progress. See original paper for figures.