Pattern Anal Applic (2017) 20:613–637
DOI 10.1007/s10044-017-0611-1

SURVEY

State-of-the-art in visual geo-localization

Jan Brejcha[1] · Martin Čadík[1]

Received: 7 September 2015 / Accepted: 16 March 2017 / Published online: 28 March 2017

Abstract Large-scale visual geo-localization has recently gained a lot of attention in computer vision research and new methods are proposed steadily. However, surveys of visual geo-localization methods are rare and they focus mainly on city-scale localization methods. We present a comprehensive and balanced study of existing visual geo-localization domains, including city-scale, global approaches and methods for natural environments. We describe the methods to show their pros and cons, application domains, datasets, as well as evaluation techniques. We categorize the reviewed methods by two criteria. The first is the type of data the method uses for geo-location estimation. The second criterion is the target environment for which the method has been proposed and validated. Based on this categorization, we analyze important conditions that must be considered while solving geo-localization problems. Each category is in a different state of research—while city-scale image-based methods received a lot of attention, other categories such as natural environments using cross-domain data sources are still challenging problems under active research. Future research of large-scale visual geo-localization is discussed, primarily the challenging and new research category—geo-localization in natural environments.

Keywords Visual geo-localization · City-scale localization · Natural environments · Image geo-location · Visual odometry · Geo-tagging · Image to model registration · 3D alignment · Cross-domain registration · Extrinsic calibration · 6 DOF

✉ Jan Brejcha
ibrejcha@fit.vutbr.cz

Martin Čadík
cadik@fit.vutbr.cz

[1] Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic

1 Introduction

Billions of images and videos on the Internet comprise big amount of valuable information covering ever growing geographic areas. Despite proliferation of GPS-equipped cameras and mobile devices, the majority of available media still lack the geo-tag information; according to Flatow et al. [24], there is around 2% of geo-tagged media on Twitter and 25% on Instagram.

Location gives the context and it is essential for image and video recognition. Important applications are crucially dependent on the location knowledge, e.g., model-based image enhancement [47], augmented reality [9, 59, 65] and self-driving vehicles [14, 52]. Additionally, visual geo-localization could help in existing non-visual localization systems to achieve higher precision and robustness.

Hays and Efros [33] introduces visual geo-localization as "… estimating a distribution over geographic locations from single image…," Zamir and Shah [93] define the problem as "… estimating the geo-location of a query image by finding its matching reference images,"[1] and Bansal et al. [10] say "Given a ground-level street-view (SV) image in an urban area, we want to determine the geo-location of the camera in the absence of any metadata (GPS or camera parameters)." In summary, we define visual geo-localization as finding the geographic coordinates (and possibly the camera orientation) for given query image.

[1] Author's note: in a reference database of geo-tagged images.

The problem has several variants—we can use initial GPS estimate in small-scale geo-localization problems or no initial estimate in large-scale geo-localization variant. Sometimes, there is an assumption that we know the camera intrinsics such as field of view (FOV), but in many practical scenarios this information may not be available making the geo-localization task even harder. Visual geo-localization in context of data mining from social media was reviewed by Ji et al. [39].

1.1 Classification of the visual geo-localization methods

We classify the works in this survey by two main criteria. The first criterion is based on type of input data. We recognize two main classes of methods—*image-based* methods and *methods utilizing data of multiple modalities*. ***Image-based*** methods use large GPS-tagged image databases to infer the location of the query image. These methods can be used to precisely locate (up to several centimeters in some cases) images mainly in highly urbanized areas, with high density of ground-level imagery available online. Methods utilizing ***data of multiple modalities*** use more information, beyond a simple image database. Mostly, the methods make use of digital elevation models (DEM) [7, 9, 65, 87], orthophoto maps, attribute maps [56] or satellite imagery [38]. Such methods were developed mainly for areas, where coverage by ground-level imagery is sparse, e.g., mountain areas, deserts and other places with low population density.

Categorization based only on type of data would not be enough, since the categories may overlap. In order to distinguish between the methods better, we add a second classification criterion—the environment for which the particular method was developed. We divide the environment criterion into three classes:

- **global**—unrestricted geo-localization at the planet scale (Fig. 1a),
- **city scale**—geo-localization in urban environments (Fig. 1b) and
- **natural**—geo-localization in natural (non-urban) environments—e.g., in the mountains (Fig. 1c).

The goal of global methods is to geo-localize query image without prior assumption about the environment type. The ability of geo-localizing single image in the whole world is appealing, but the existing methods provide poor accuracy. The localization is considered as successful, if the query image is localized within 200 km from the ground-truth position [33].

City-scale methods are designed to localize more precisely, assuming the query image resides in a specific urban area. Natural methods are specialized as well—the published methods are targeted to a specific natural environment such as deserts or mountains. There are principal differences at urban and natural environments that determine the complexity of the respective geo-localization problem:

Data Availability Dozens of photos of attractive places and landmarks in highly populated areas—Flickr API returns more than 200 K photos containing the tag “Eiffel Tower” and more than 100 K photos containing the tag “Statue of Liberty” (2016). Such abundance of

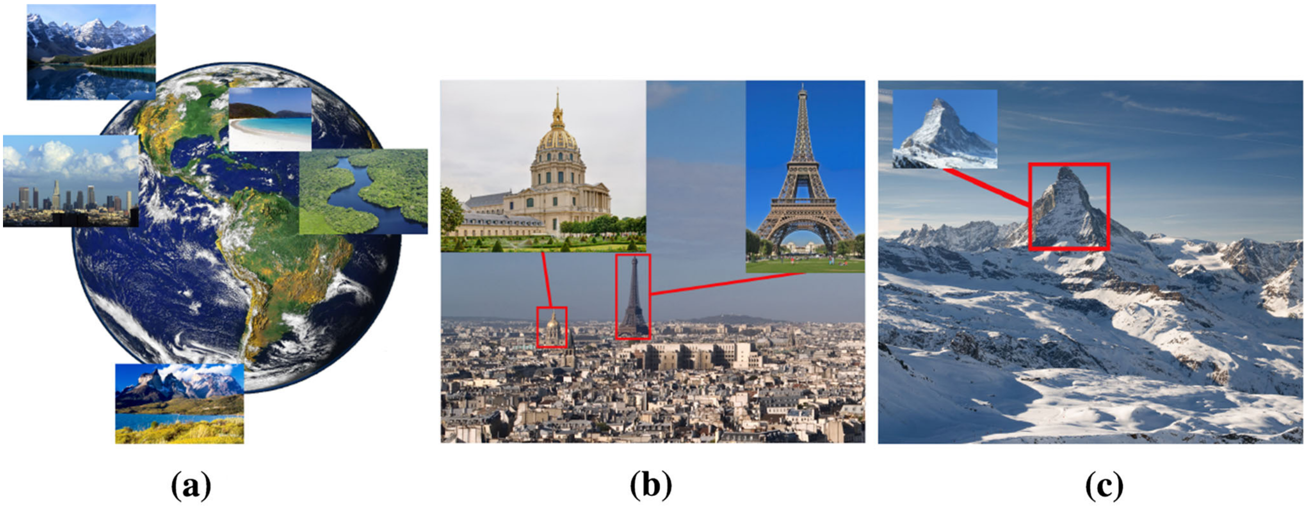

Fig. 1 Illustration of visual geo-localization categories. **a** Category: global (Credit: Neil Palmer (CIAT)—Amazonia, Michael Pazzani—Caribbean Island, Thomas Pintaric—Los Angeles Dowtown) **b** category: city scale (Credit: Myrabella—Paris from Notre Dame, Diliff—Les Invalides **c** category: natural (Credit: Felix Lamouroux—Zermatt Panorama, Marcel Wiesweg—Matterhorn)

data enables image to image search with Bag-of-Words, feature-based techniques and SfM model matching.

Well-Defined Objects Man-made objects with distinctive and stable appearance, such as buildings, bridges and road signs, can be well recognized and matched. Moreover, mutual arrangement of such objects in space is often unique, which can be used for localization. On the other hand, in natural environments, objects are rather unlikely to match well—e.g., mountains, foliage and clouds. Those are difficult to recognize because of inconsistent appearance (weather and illumination changes, vegetation growth) and frequent occlusion of such objects in the real world.

Repetitive and Self-Similar Patterns Urban environments contain repetitive objects such as windows, lamps and logos. In natural environments a lot of fractal and self-similar patterns can be found. All these aspects make the visual geo-localization difficult task.

Such specific issues narrow down the options for solutions of geo-localization in a particular environment. Broad overview of visual geo-localization methods in connection with the introduced classification is presented in Sect. 2. The environment specifics led to the development of various datasets, which we summarize in Sect. 3. In Sect. 4, we compare usual evaluation methods for visual geo-localization. We review the key geo-localization methods in Sect. 5. Finally, we summarize important visual geo-localization applications in Sect. 6, and discuss the future of visual geo-localization in Sect. 7.

2 Overview of the visual geo-localization methods

In this section, we briefly summarize the breadth of existing methods with respect to classification introduced in Sect. 1—*image-based methods* and *methods using data of multiple modalities*. For detailed review of the most influential methods, please refer to Sect. 5. All the overviewed methods are summarized in Table 1.

2.1 Image-based methods

Image-based methods are used when sufficient amount of reference images is available. *Image retrieval* methods use big databases of GPS-tagged images to infer the location of a query image by retrieving similar images using various matching algorithms. *Structure-from-Motion* localization methods use 3D reference model constructed using geometrical relationships between many overlapping images. Thanks to this fact, not all images need to contain explicit GPS tags.

2.1.1 Image retrieval

We identified two main approaches to visual geo-localization via image retrieval. The first (nonparametric) option is to search for similar images in large geo-tagged image database and to infer the query image location based on location of the most similar database images [40, 60, 69, 75, 93–95, 97]. The second (parametric) option is to train classifiers or regressors, given the geo-tagged database of images as a train set, to directly predict the geo-coordinates of the query image [44, 90].

First attempts to localization by image retrieval were published by Robertson and Cipolla [69]. They created a database of two hundred photos of rectified facades in Cambridge city center. For rectification, an automatic method by Kosecka and Zhang [48] finding vanishing points was used. Facade positions were manually annotated with respect to the 2D map to connect each facade with meaningful coordinates. For matching, sum of squared differences of patches centered around the Harris key points was used. The method does not scale well, since it matches the query image against all images in the database, which would lead to prohibitive run times on bigger image databases.

Zhang and Kosecka [97] extended the former approach by using database of SIFT feature descriptors [58]. Images were GPS-tagged, so no manual annotation of correspondences with map was needed. Still, the coarse matching stage was implemented as a simple voting to every document in the database causing high computational complexity. Best five candidates were verified and sorted by RANSAC [23], and the final location was found by triangulation of the best candidates.

One of the first methods for large-scale localization in a city was developed by Schindler et al. [75]. They tested the method on 20 km of street-side imagery, which was publicly released as a dataset. As we find this work very important, we add more description in Sect. 5.2.1.

The problem of place recognition was studied by Johns and Yang [40]. They improved the Bag-of-Words (BOW) technique [78] by clustering the image database of 200K images to visually similar scene models (landmarks). However, their results show only marginal improvement compared to standard BOW technique.

Zamir and Shah [94] used dataset of 100K geo-tagged images downloaded from Google Street View. They used a nearest neighbor tree search with additional steps of pruning and smoothing for better accuracy. Furthermore, they developed a measure called *Confidence of Localization* which quantifies the reliability of the localization of a particular query image using Kurtosis of a normalized voting space.

Table 1 Overview and properties of geo-localization methods

Method	Class	Environ.	Test area	Local. succ.	Max. err.
Robertson [69]	Image-based, retrieval	City	Single street	95%	N/A
Zhang [97]		City	City part	72% on ICCV 2005 Cont.	16 m
Schindler [75]		City	Single city	70%	10 m
Hays [33]		Global	Earth	16%	200 km
Zheng [98]		Global	Earth landmarks	Accuracy 80.8%	N/A
Li 09 [53]		Global	Earth landmarks	40.58% visual and tags	1 of 500 landm.
Zamir 10 [94]		City	240 km of street view	78%, vs. [75]: 39%	250 m
Chen 11 [17]		City	Single city	65%	N/A
Johns [40]		City	Landmark	N/A	N/A
Zamir 14 [95]		City	Several cities	N/A	N/A
Zamir 14a [93]		City	Several cities	44%	100 m
Mishkin [60]		Global	Place	P/R: $\frac{0.821}{0.825}$	1 frame
Weyand [90]		Global	Earth	37.6% on IM2GPS test set [33]	200 km
Irschara [37]	Image-based, SfM	City	Landmark	39% within top-10 candidates	N/A
Li 10 [55]		City	Dubrovnik [55], Rome [55], Vienna [37]	92.4% (Rome)	400 m
Sattler 11 [71]		City	Dubrovnik [55], Rome [55], Vienna [37]	97.6% (Rome)	400 m
Sattler 12 [72]		City	Dubrovnik [55], Rome [55], Vienna [37]	99.1% (Rome)	400 m
Sattler 12a [73]		City	Aachen [73], Vienna [37]	74–83%	N/A
Li 12 [54]		City	1 K of landm.	73% on Quad [20] 90%, images under 10 m	N/A
Hao [31]	Image-based, SfM	City	Landmark	N/A	N/A
Bergamo [12]		City	Landmark	95% on Lan.-3D [31] 63% on Lan.-620 [12]	N/A
Svärm [82]		City	Dubrovnik [55]	99.75% (Dubrovnik)	400 m
Sattler 15 [70]		City	San Fr. [54] Landmarks	62.5% (San Fr.)	N/A
Kendall [45]		City, indoor	City part building	2 m, 3° outd. 0.5 m, 5° ind.	N/A
Zeisl [96]		City	San Fr. [17], Dubrovnik [55]	99.75% (Dubrovnik)	400 m
Talluri [83]	Multi.DEM	Natur.	148 km^2	N/A	N/A
Stein [81]		Natur.	298 km^2	N/A	N/A
Naval 97 [64]		Natur.	N/A	N/A	N/A
Naval 98 [63]		Natur.	900 km^2	Avg. err. 393 m	N/A
Woo [91]		Aerial, natur.	2.28 km^2	N/A	N/A
Baatz [6]		City	Single city	35% or 85%	N/A
Ramal. [68]		City	Single city	Avg. err. 2.8 m	N/A
Baatz 12 [7]		Natur.	40,000 km^2	88%	1 km
Tzeng [87]		Natur.	10,000 km^2	N/A	N/A
Porzi [65]		Natur. (orient.)	100 places in the Alps	Avg. err. 1.87°	5.22°
Baboud [9]		Natur. (orient.)	28 photos in the Alps, Rocky Mnts.	86%	<0.2°
Hammoud [30]		Mainly natur.	20,000 km^2	49%	14 km
Chen 15 [18]		Natur.	10,000 km^2 (America, Asia)	60%	4.5 km
Hakeem [29]	Multi.-SLAM	City	Campus	Avg. err 6 m ICCV Cont. 2005	N/A
Conte [19]		Natur.	N/A (S. Sweden)	N/A	N/A

Table 1 continued

Method	Class	Environ.	Test area	Local. succ.	Max. err.
Larnaout 12 [50]	Multi.-SLAM	City	City center	N/A	N/A
Larnaout 13 [51]		Rural, city	Rural, city	N/A	N/A
Middel. [59]		City	40 km^2	<1 m	N/A
Jacobs [38]	Multi.-other	Global	Pennsylv., Maryland	Avg. err. 71.8 km	N/A
Gallagher [26]		Global	Earth	33% on IM2GPS test set [33]	200 km
Kalogerakis [41]		Global	Earth	58% on IM2GPS test set [33]	400 km
Baatz 10 [6]		City	Single city	Earth mine 85% Navteq 35%	N/A
Kelm 11 [43]		Global	Earth	10%	1 km
Kelm 11a [42]		Global	Earth	35%	1 km
Lin 13 [56]		Global	1 600 km^2	17.37%	N/A
Aubry [4]		City	Landmark	55% good matches	18% no match
Viswanathan [89]		Aerial, natur.	c. 0.1 km^2	31% matches for top 10% cand.	N/A
Ardeshir [3]		City	10 km^2 Washing- ton DC	60% for top 20% cand.	N/A
Lin 15 [57]		City	Several cities	80%	20% of cand.
Workman [92]		Global	40,000 km^2	22.7%	N/A

Test area defines the area on which the method has been tested in original publication, localization success (local. succ.) denotes the best result achieved with given method, and maximum error denotes the maximum distance from the ground-truth position which is considered to be correct localization

multi. methods using data of multiple modalities, *cont.* contest, *cand.* candidates, *P/R* precision/recall, *landm.* landmarks, *mnts.* mountains, *tags* method uses also user-defined tags for localization, *San Fr.* San Francisco

The problem of using image database with noisy GPS tags was also studied by Zamir et al. [95]. For a query image, several matches from image database are found. Triplets of the query image and two database matches are formed. From these triplets, the geo-location can be estimated directly for correct locations of image pairs. The authors propose a method using random walks to correct the geo-locations of noisy GPS positions.

Zamir and Shah [93] used dataset from Google Street View, which is a superset of 100K dataset presented in their previous work [94] (for more information about datasets please refer to Sect. 3). They aimed to further improve nearest neighbor matching by pruning outliers and by incorporating approximate feature matching using generalized minimum clique problem (GMCP). The authors compare their method to Schindler et al. [75] and to their previous work [94]. They show that the new method has lower localization error; it was able to localize more than 55% of the query images within the error of 250 m, whereas their previous method [94] localized 50% and Schindler et al. [75] localized only 46% within the same error.

An interesting problem of place recognition in changing conditions, such as changes between day and night or winter and summer, was explored by Mishkin et al. [60]. They adopted a BOW method with multiple detectors, descriptors, view synthesis and adaptive thresholding to cope with large visual changes of the environment.

First global visual geo-localization method was published by Hays and Efros [33, 34]. They created a database of various features from 6 million images distributed around the whole Earth. Density of retrieved nearest neighbors was used to estimate the location of a query image. For more information, please refer to Sect. 5.1.1.

Global geo-localization was also recently studied by Weyand et al. [90]. They developed a technique using Convolutional Neural Networks that are trained to directly estimate the geo-location of a query image. For this purpose a dataset of 126 million images was used. More details about this work can be found in Sect. 5.1.4.

At the border of our interest lie landmark recognition techniques, which we review briefly. Li et al. [53] use BOW technique combined with multiclass SVM to learn landmark classification. Zheng et al. [98] combine GPS-tagged images from online services and a textual tour guide with unsupervised learning to build a world-scale landmark database. Avrithis et al. [5] studied the problem of separating landmark and non-landmark images using improved

BOW method. Chen et al. [17] studied a problem of landmark detection on mobile devices using onboard GPS estimates. The main contribution of their work is publicly available dataset for landmark recognition and localization (see Sect. 3).

2.1.2 *Structure from motion*

Structure-from-Motion (SfM) is a set of techniques to reconstruct a 3D scene from a set of overlapping 2D images depicting the same scene from different viewpoints. The correspondence of 2D imagery with the 3D model allowed researchers to develop techniques for localization and camera orientation (camera pose estimation) from a single image and camera pose tracking from a continuous series of images. The reconstructed 3D model consists of 3D points, positions and orientations of the cameras of the source imagery. An advantage of such datasets is that the camera pose reconstruction of the query image can be very precise, with error up to units of meters.

Building 3D models from tens of million images using SfM techniques have been thoroughly studied [1, 13, 20, 28, 32, 35, 79, 80]. Heinly et al. [35] automatically created models of many places around the whole world from 100 million photos from a Yahoo image dataset [85] in six days on a single computer. SfM models are usable in highly urbanized areas and near important landmarks. Irschara et al. [37] used several hundreds of photos to create an SfM model of the most famous landmarks in Vienna. Relevant photos in the SfM model were searched for by the standard image retrieval (BOW) approach. They successfully registered majority of frames of four test videos and test images. The authors also presented a compression technique to reduce the number of images needed to cover the 3D scene.

Li et al. [55] developed a location recognition approach which prioritizes features from a SfM model to be matched against query features. They show that defining priorities based on properties of features in the SfM model and application of feature-to-point (2D-to-3D) matching play a key role in improvement of matching performance on Dubrovnik and Rome dataset (see Sect. 3.2).

Sattler et al. [71] proposes a technique of direct 2D-to-3D matching. They assign feature descriptor to each visual word and match directly query feature descriptors to descriptors in relevant visual words. They show improvement in matching performance, while keeping reasonable response times (fractions of a second). In a follow-up work by Sattler et al. [72] the ideas of 2D-to-3D and 3D-to-2D were combined and formulated into an *Active Correspondence Search*, which improved both time and matching performance.

Problems of image retrieval methods connected to localization were studied in the next work by Sattler et al. [73]. Algorithms using direct feature descriptor matching outperform classical image retrieval approach by 15%. The problems in image retrieval approach causing this performance gap were identified and addressed by introducing selective voting for image retrieval approach. This method slightly outperformed the direct descriptor matching.

Among the first works addressing large-scale localization based on SfM model was an approach by Li et al. [54]. They presented a method able to cope with hundreds of thousands of images using *a cooccurence prior for RANSAC* and *bidirectional matching of image features with 3D points*, which is a similar idea to the *Active Correspondence Search* presented by Sattler et al. [72].

Bergamo et al. [12] used SfM model to learn random forest codebook for Landmark classification. The problem of landmark classification was further approached by other authors [31, 67], but it is out of the scope of this survey.

Swärm et al. [82] incorporated the knowledge about gravity direction in the query image obtained from gravitational sensors. Their method can handle large amount (up to 99%) outliers.

Localization on large datasets (hundreds of thousands images in the SfM model) poses new problems, namely large memory footprint of the model and strictness of the SIFT descriptor ratio test. These problems are approached by Sattler et al. [70], by quantizing descriptors to reduce the search space, while incorporating a new voting strategy to remove ambiguous matches.

The work by Zeisl et al. [96] on large-scale geo-localization using SfM model also tackles the problem of large fraction of outlier matches. The authors build on Svärm et al. [82], utilizing geometrical constraint of gravity direction on camera and incorporate them as well as additional constraints into the camera pose voting.

Kendall et al. [45] used SfM model to train a convolutional neural network for large-scale camera relocalization. Their experiments operate on 50,000 km^2 and have very accurate results—the authors report 2 m and 3° in outdoor areas, and 0.5 m and 5° in indoor areas.

2.2 Methods using data of multiple modalities

Unlike image-based methods, methods leveraging data of multiple modalities use additional input data to find camera location for a query image. A popular choice is cross-domain matching of a query image and a *terrain model*, with utilization of features such as horizon lines, ridges and edge maps. *Simultaneous localization and mapping* aims to localize a camera in unknown environment and to simultaneously create a map of that environment. *Methods using other input data* such as orthophoto maps combined with attribute maps, bird's-eye or satellite weather imagery were exploited. In this domain, mainly methods for outdoor, non-urban environments are considered. This is due to the nature of areas with lower density of population—for image-based methods there is not so many ground-level imagery, so other sources are used.

2.2.1 *Methods using terrain models*

The main motivation for the first visual geo-localization works was the need for localization of mobile robots and planetary rovers in outdoor environments. One of such works was presented by Talluri and Aggarwal [83, 84]. They use DEM model, and the robot is equipped with digital compass, altimeter and a monocular camera that can be panned and tilted. The localization is achieved by matching horizon lines extracted from a query image against those rendered from DEM. The authors conducted experiments on 1.41 km^2, with the area sampled uniformly with the distance of samples of 30 m.

Stein and Medioni [81] use horizon lines for localization as well. They create a database of synthetically rendered 360° horizon lines using DEM. Horizon lines are approximated by polygons, from which the database is created. Horizon from an input query image is extracted semiautomatically, encoded into the same format as horizons in the database against which it is matched. The best candidates are verified geometrically.

Localization using horizon line was further studied by Naval et al. [63, 64]. In these works, the skyline from query image is extracted by multilayer perceptron neural network classifier. As local feature points, the peaks are used—they are detected in both query image and the DEM. Pose of the query is calculated using three feature points from the database via minimization of error function using nonlinear least squares.

Woo et al. [91] studied navigation of UAV in mountain areas using DEM and infrared (IR) images with known altitude using altimeter. Infrared spectrum was used to tackle the visibility problems during the night and bad weather conditions. From a series of frames, the peaks were extracted and spatial reconstruction of peaks in 3D was achieved by the factorization method. Next, 3D positions of peaks from query frames are matched to peaks extracted from DEM, and the pose is hypothesized. Finally, synthetic horizon from DEM at hypothesized location is aligned with query horizon to confirm or reject the estimated location.

City-scale visual geo-localization method based on fish-eye images of the urban canyons was published by Ramalingam et al. [68]. The method takes an omni-skyline image, extracts the skyline defined by buildings and matches this skyline into database of synthetically rendered skylines. The method is usable in cities such as New York which have very tall buildings.

Hammoud et al. [30] extend the extracted horizon line from query image by LIDAR and Hyper-Spectral Land Use/Cover imagery. They match the inputs separately and combine them by linear fusion into single probability map. The authors validated their approach on 100 query images on two world regions, each of area 10,000 km^2.

Baatz et al. [7, 74] were the first to develop factually large-scale visual localization solution in outdoors (on an area of 40,000 km^2). The method uses large database of extracted features from horizon lines, called contourlets. The contourlets are dense representations of normalized and smoothed horizon lines stored as a single integer. For localization, they use Bag-of-Words-like approach to retrieve the best 1000 candidates, which are geometrically verified to find the best matching locations. Thanks to direction and location voting strategy and geometrical verification of horizon lines, the method is able to estimate both location and coarse heading of the camera. For detailed description please refer to Sect. 5.3.1.

Tzeng et al. [87] presented a similar work to Baatz et al. [7]. The idea of using database of horizon features generated from rendered DEM and searching for horizon features from query image is the same. The difference is that concavities of horizon line parts were used as local features.

An advanced approach based on horizon lines was presented by Chen et al. [18]. The authors build on the approach presented by Saurer et al. [74], and they extend the local feature descriptor utilizing multiple ridge lines, not only the horizon line. The feature extraction is the same as in Saurer et al. [74]. The key difference is in the voting stage of BOW, where the documents are voting not only for horizontal, but also for vertical direction. The authors tested their method on 10,000 km^2 and showed that their results were better than the results of Saurer et al. [74].

2.2.2 *Simultaneous localization and mapping*

Visual simultaneous localization and mapping (V-SLAM) is also relevant to the topic of visual geo-localization when performed outdoors. Generally, SLAM methods make use of various inputs, such as RGB image combined with depth, stereo, lidar sensors, GPS, etc. We focus on the works relevant to visual geo-localization, surveying the works utilizing only the single camera input. Since SLAM methods are focusing on continuous localization in time, we separated these methods from the problem of visual geo-localization of a single image.

An approach by Middelberg et al. [59] for 6-degrees-of-freedom (6-DOF) localization on mobile devices uses large offline SfM point cloud at server and small keyframe-based SLAM [46] model on the mobile device. The keyframes are matched against the offline SfM model to avoid drift, while normal frames are processed on the device to estimate the motion frame by frame.

Hakeem et al. [29] proposed an offline method for estimating trajectory of a moving camera. They use a database of GPS-tagged photos to match keyframes with, and from the best matches they calculate essential and fundamental matrices to recover camera pose. Triangulation step is used to disambiguate the scale. To obtain smooth trajectory, the obtained locations are interpolated using B-splines.

Conte and Doherty [19] used geo-tagged image database in combination with KLT feature tracker [86] to address the problem of GPS signal outages of unmanned area vehicle (UAV). The visually tracked position was fused with the inertial measurement via onboard sensors through Bayesian framework.

Method by Vaca-Castano et al. [88] for trajectory estimation in a city is built on top of localization method by Zamir and Shah [94]. Each keyframe is localized using the discussed method, and Bayesian filtering enforcing temporal coherency is used. As the results are often noisy and exhibit false loops, the final trajectory is constructed using minimum spanning tree (MST)-based algorithm.

Larnaout et al. [50, 51] combine classical SLAM methods with elevation constraint taken from digital elevation map (DEM), because the height of the SLAM vehicle is constant. They also add a 3D buildings model as a constraint to the reconstructed 3D point cloud.

2.2.3 Methods using other input data

Baatz et al. [6] researched a method for localization in urban environment. They use panoramic street-view images and extruded floorplans of buildings to build a database of rectified images (by mapping the facades onto the extruded 3D models). Query image is also rectified based on vanishing points, which reduces the matching problem to 2D homothety. Detailed description of this method can be found in Sect. 5.2.2.

Data-driven solutions aim to learn the relationship between a photograph and the land cover appearance based on a geo-tagged ground-truth dataset. As in the case of Lin et al. [56], the geo-database is created from several corresponding data sources. The idea is to match input query photo against the database created by the triplets of ground-level image, aerial orthophoto map and attribute map. Detailed description of this method can be found in Sect. 5.1.2.

The idea of cross-view matching was researched by Workman et al. [92], who approached the problem by adapting a convolutional neural network (CNN) (pre-trained on Places [99] dataset) to extract similar features from ground-level photographs and aerial orthophoto maps. Nearest neighbors were used as candidates ranked by calculating euclidean distance between the ground-level and aerial features. The authors also developed a large dataset having over 1.5 million geo-tagged matching pairs. The authors claim their method is the state-of-the-art in cross-view geo-localization, which is supported by 6% improvement in comparison to previous work by Lin et al. [56].

Lin et al. [57] presented similar work to Workman et al. [92]. They use CNN for the cross-view matching, but they use Google Street View and aerial “bird’s-eye” imagery, which is captured tilted compared to classical aerial orthophoto imagery taken orthogonally to the terrain. They used CNN pre-trained on ImageNet and Places [99] databases. The results are currently far from the practical application; 80% of correctly localized queries are contained in 20% of top candidates.

Aubry et al. [4] developed a method to register an artistic painting with a 3D model, which also implies pose of the camera. For matching, they mention the possibility to use exemplar-based SVM classification introduced by Shrivastava [77]. Based on this approach, they developed a new method to avoid training SVM classifiers. They tested the method on a variety of historical paintings, which they successfully registered with the 3D model.

Viswanathan et al. [89] developed a method for robot localization by matching Google Street View panorama to aerial orthophoto map. They warp the street-view panorama to bird’s-eye view (top-down) and they use standard matching techniques using various features such as SIFT, SURF and FREAK. In their scenario, SIFT proved to have stable performance throughout the test set.

Ardeshir et al. [3] exploit semantic GIS information from a GIS database, such as locations of fire hydrants, traffic signals, road signs and other objects to improve object detection. Image metadata as GPS location, FOV and heading are used as a hypothesis to match the objects in the query image against the objects obtained from the GIS database under given viewpoint. Based on the object detection, the authors also developed a method of camera localization. Location hypotheses are generated on uniformly sampled grid, excluding the areas covered by buildings (only streets are considered). For each hypothesis, the object detection method is used and location-orientation score is calculated.

The geographic coherence in image sequences may be also used for camera localization. Jacobs et al. [38]

exploit sequences of frames from static outdoor cameras correlated with satellite imagery for location estimation. Kalogerakis et al. [41] learn human travel priors from 6 million database of images from Flickr Web service. Their approach is able to geo-localize image sequences from user gallery, with the use of timestamps to calculate probable locations based on the learnt prior. Kelm et al. [42, 43] use video keyframes combined with textual features to find the most probable regions of origin.

Multimodal approaches exploiting textual tags or other information also exist. Global geo-localization method by Gallagher et al. [26] used database containing over million of geo-tagged images and user-defined textual tags. The user tags from a query image are used in the matching process in parallel with several other visual features such as GIST, color histograms, tiny images and bag of textons.

2.3 Camera orientation estimation

Camera orientation estimation problem is also related to visual geo-localization. Some visual geo-localization methods are designed to retrieve the camera orientation, especially SfM [37, 45, 54, 59, 82] or horizon-based and DEM matching approaches [18, 30, 66, 74, 87]. However, some methods, such as image-based and cross-view visual geo-localization methods are unable to deliver camera orientation [33, 56, 57, 90]. In such cases, the geo-localization and camera orientation methods could be used together in order to retrieve full 6-DOF pose.

Kosecka and Zhang [48] presented algorithm for camera orientation estimation based on vanishing points. This method is suitable for urban indoor and outdoor scenes, as the detection of vanishing points is based on line segments. The line segments can be detected in urban scenes easily, while in natural scenes they are present sparsely.

Several approaches for camera orientation estimation for natural scenes exist. Behringer [11] matches synthetic panoramic horizon line to horizon line detected in query image. This approach was extended by Baboud et al. [9], who presented an algorithm for robust silhouette matching. Since it matches the synthetic and the query edge maps, it is much more robust to occlusion than methods using horizon line only. More details about this method can be found in Sect. 5.3.3. Baatz et al. [7] published camera orientation algorithm based on matching semantic areas in the image, such as forests or rivers. Efficient camera orientation refinement was approached by Porzi et al. [65]. They use smartphone sensors as an initial estimate, which is refined by silhouette matching algorithm similar to [9].

2.4 Summary

The surveyed methods were classified as *image-based methods* and *methods using data of multiple modalities*. The *image-based methods* were used mainly for urban areas, while *methods utilizing data of multiple modalities* were used mainly for localization problems outside city borders—in natural environments.

For *image-based methods*, three main methods for finding database matches for a given query image can be identified. The first is *nearest neighbor search* [75], the second is *Bag-of-Words* approach, [78], and the third approach is *Structure-from-Motion* [79], which reconstructs 3D model from many overlapping images, without the need of knowing GPS position of the images.

For *methods using data of multiple modalities*, the approach of horizon line matching is a popular technique [7, 18, 63, 64, 74, 81, 83, 84]. Another popular technique is a *cross-view* matching approach introduced by Lin et al. [56] and further studied in other variants [57, 92].

While image-based solutions are well established and achieve precise results, their use is limited. For urbanized areas, there are algorithms for fast and precise geo-localization, without the need of GPS sensor [59]. This is obviously different, when we travel outside the borders of the cities. In natural environments, there is still lack of algorithms that are fast and, more importantly, precise. Still a lot of work has to be done, in order to have at least similar precision as in the urbanized areas. For example, in the results of Saurer et al. [74], distance under which the query is considered as correctly localized is 1 km, which is still far from the results obtained by Middelberg et al. [59], who report the localization error in meters. In case of horizon-based localization proposed by Saurer et al. [74], 40% of query images need user interaction for discovering horizon line, mainly due to tree occlusions which arise in real-world photos quite often. Furthermore, horizon occlusion by fog or clouds cannot be addressed in this scenario. For such situations, more robust features for matching such as edges or semantic segments such as areas of forests, glaciers, grassland, rocks and stonefields are needed.

3 Datasets

Methods mentioned in this survey use several datasets that can be used to measure and compare existing and novel approaches to visual geo-localization. For *city-scale environments*, there exist various datasets acquired from online Web services such as Flickr, Panoramio or Google Street View. On contrary, datasets for visual geo-localization in natural environments are only sparse.

3.1 Image-based datasets

Google Maps Street View Dataset[2] introduced by Zamir and Shah [93] contains 102K images acquired automatically from Google Street View Web site, from Pittsburgh, PA and Orlando, FL. The dataset contains full 360° panoramic images with distance of about 12 m between consecutive locations. This dataset is suitable for precise localization and camera orientation estimation in urban areas.

IM2GPS test sets IM2GPS approach by Hays and Efros [33] was trained using 6 million geo-tagged images acquired from Flickr Web service. From this large dataset, only the test sets containing several hundred of photos is available online.[3]

YFCC100M: The New Data in Multimedia Research. A total of 100 million datasets of Yahoo Flickr images[4] were published by Thomee et al. [85]. The images and videos in this dataset are under Creative Commons licenses, making the data easily usable for anyone. Compressed metadata for this dataset consists of 13 GB of data, and contain GPS locations (for 48 million of photos and for 100K videos), tags, time span and camera information.

San Francisco Landmark Dataset Dataset of 1.7 million street-level images[5] with ground-truth labels, geo-tags and calibration data was provided by Chen et al. [17]. Challenging query set of 803 cell phone images taken few months after the first part of the dataset is also present.

Visual Place Recognition in Changing Environments VPRiCE dataset[6] for changing environments from VPRiCE challenge 2015.

Alps100K dataset Dataset called Alps100K[7] composed by Čadík et al. [15] was used in the original paper for elevation estimation. The fact that it contains data from Flickr which contains GPS coordinates makes it usable also for geo-localization tasks.

3.2 SfM datasets

Datasets for large-scale SfM and location recognition/pose estimation. Several datasets by Li et al. [55] for SfM problems are publicly available online.[8] The biggest are the Rome16K and Dubrovnik6K, covering the most famous parts and landmarks of the cities. Also variety of smaller datasets for famous landmarks such as Notre Dame Cathedral, Tower of London, Trafalgar Square, Vienna Cathedral and many more are available.

Quad dataset Crandall et al. [20] provide also SfM datasets. The Quad dataset[9] consist of 6514 images, about 5000 images which originates from iPhone 3G contains GPS information and 348 images contain very precise GPS coordinates (accuracy about 10 cm).

Landmark 3D Hao et al. [31] introduced dataset called *Landmark 3D*[10] which contains 45K images of 25 landmarks collected from Flickr Web service. Besides the landmark photos the dataset also contains reconstructed 3D landmark models. It is suited mainly for landmark recognition.

Cambridge Landmarks dataset Kendall et al. [45] recently published their dataset for 6-DOF camera relocalization using CNN. Training and testing images are available online,[11] as well as SfM models used for the camera pose training. The dataset contains 12K images with full 6-DOF camera pose.

3.3 Datasets for methods using data of multiple modalities

Dataset for horizon-based localization Two datasets for horizon-based localization were published online[12] by Saurer et al. [74]. The two datasets contain over 1000 images with verified GPS position and FOV for every image. For 203 images the horizon segmentation is available.

Cross-view dataset CVUSA dataset was introduced by Workman et al. [92] comprising 1.5 million geo-tagged image matched pairs of ground-level and aerial orthophoto map. It was created from Flickr photos and Google Street View. The dataset can be obtained directly from authors,[13] but it is not available online.

Raw mapping data of multiple modalities Raw mapping data are available through USGS[14] where various mapping data such as topo maps, aerial photographs or satellite images are available. The DEM[15] data are also available [27]. NLCD provides data[16] such as land cover attribute maps or tree canopy maps. Maps containing the change between consecutive published versions of land cover maps (last two are from years 2006 [25] and 2011[36]) are also available.

[2] http://crcv.ucf.edu/projects/GMCP_Geolocalization/.

[3] http://graphics.cs.cmu.edu/projects/im2gps/.

[4] http://webscope.sandbox.yahoo.com/catalog.php?datatype=i&did=67.

[5] https://purl.stanford.edu/vn158kj2087.

[6] https://roboticvision.atlassian.net/wiki/pages/viewpage.action?pageId=14188617.

[7] http://cphoto.fit.vutbr.cz/elevation/.

[8] http://www.cs.cornell.edu/projects/bigsfm/#data.

[9] http://vision.soic.indiana.edu/projects/disco/.

[10] https://landmark3d.codeplex.com/.

[11] http://mi.eng.cam.ac.uk/projects/relocalisation/#results.

[12] http://cvg.ethz.ch/research/mountain-localization/.

[13] http://cs.uky.edu/~scott/research/deeplyfound/.

[14] http://nationalmap.gov/elevation.html.

[15] http://nationalmap.gov/elevation.html.

[16] http://www.mrlc.gov/nlcd11_data.php.

4 Evaluation methods used in visual geo-localization

Several common methods for evaluation of geo-localization systems exist. Usually authors use similar evaluation methods for related visual geo-localization topics, so they are able to compare with competitive methods. We review most common methods found in the surveyed works.

4.1 Top-*k* candidates/percentage of localized images

Popular evaluation technique used by state-of-the-art geo-localization methods is the plot of number of the candidates (horizontal axis) against fraction of query images from evaluation set that were localized within the given number of candidates [3, 7, 56, 74, 92]. In other words, when the method returns ordered list of candidate locations, we count how many query images were localized correctly using fixed number of candidates. The image is considered as localized correctly if at least one candidate out of top-*k* candidates is located within defined distance from the ground truth. The curve has ROC-like, non-decreasing shape.

This method clearly shows how much candidates one has to inspect, in order to find at least given number of correctly localized query photos. It also illustrates that precise geo-localization is hard task, since the methods are often unable to provide good localization accuracy for the top-*1* candidate. However, practical usability of this metric is limited. Usually, the user is interested in top-*1* candidate, since it is not practical to verify several candidates of possible locations. To address this problem, following evaluation method can be used.

4.2 Percentage of images/localization error

Another option is to plot the localization error (distance of the location estimation to the ground truth; horizontal axis) against percentage of images that were localized with the same or lower error. This method was used mainly by global geo-localization methods [33, 90], which retrieve the most probable location (1 candidate) and measure number of queries that were localized at least with error of given threshold.

Advantageous on this method is that we directly know how accurate is the method for given fraction of query images from the evaluation set.

4.3 Position, orientation error per video frame

Visual geo-localization methods based on SfM technique [55, 59, 71, 73], and reviewed SLAM methods [29, 51, 59, 88] usually evaluate their methods per frame basis. Number of correctly matching query images/frames is usually presented in the form of a table. Distance to ground truth and camera orientation error are usually calculated per video frame and plotted, frames on horizontal axis and the error on vertical axis. Average distance from ground truth is also often calculated as a measure of accuracy.

Since methods using this evaluation technique are usually verified on exactly the same datasets, it is very easy to compare the method performance to competitors. Furthermore, the methods aim to localize in real time, the computation time is also related metric.

4.4 Geo-localization area/region of interest

Similar measure to top-*k* candidates (Sect. 4.1) is the measure of geo-localization area (GA) over the region of interest (ROI, the total area of the search space) [18, 87]. Candidate positions in the search space have assigned its area (which is usually uniform for all the candidates). The candidates are sorted according to the method confidence. For each query, the GA is calculated as a sum of areas preceding the candidate that contains the ground truth, and divided by ROI. The graph is plotted with respect to changing GA/ROI measure.

In case of uniform candidate areas, this method is the same as top-*k* candidates. The method would be more informative for non-uniform sampling of the search space, since it would penalize wrong estimations with large area.

4.5 Precision/recall

Precision/recall is standard metric used for evaluation of classification and retrieval methods. In reviewed visual geo-localization, it is in fact very sparsely used method for evaluation. It was used for evaluation of object detection and place recognition in several methods [3, 60].

5 Influential methods of visual geo-localization in detail

In *Introduction* (Sect. 1.1), we presented the classification of the methods based on two criteria, namely the *data type* they are using, and the environment for which they are developed (see Table 1). In *Overview of the visual geo-localization methods* (Sect. 2), we summarized many existing approaches with respect to the first criterion (*data type*). In this section, we describe selected state-of-the-art visual geo-localization methods in a more detail and classify them according to the second criterion (*environment*).

5.1 Global image geo-localization methods

Global image geo-localization becomes possible thanks to growing number of publicly available photos. Unfortunately, by far not all of the available photos contain noise-free GPS coordinates. Other useful information might be missing as well; in particular the field of view (FOV), which helps to disambiguate the scale and position of the camera in 3D space, thus reducing the camera pose estimation complexity. Methods estimating the location without the camera orientation may therefore be advantageous in case of noisy metadata or their complete absence. As such methods are independent of metadata, environment and other assumptions, they can operate at global scale, around the whole world.

The idea of worldwide single image geo-localization was introduced in IM2GPS by Hays et al. [33], who use a database of several types of visual features for large-scale image retrieval (Sect. 5.1.1). Difficulties given by non-uniform distribution of locations of photos in a training set were addressed later by Lin et al. [56] in cross-view image geo-localization (Sect. 5.1.2). They use land cover attribute maps and aerial view maps, which are correlated with query photo and the database of ground-level photos. Jacobs et al. [38] introduced an idea of geo-localization by comparing the natural scene variations from static cameras with variations in weather satellite imagery (Sect. 5.1.3).

5.1.1 IM2GPS: estimating geographic information from a single image

One of the first methods concerned with the visual geo-localization was IM2GPS by Hays et al. [33]. IM2GPS is a data-driven approach that compares the query photo with a large database of features extracted from ground-based photos captured all around the Earth. Feature extraction of the image database takes approximately 3 days on a large cluster with 400 processors.

The method is suitable for highly populated places, where large number of photos with a GPS-tagged position is available. The test set consists of various photos from distinct places (not only highly urbanized areas), which is extremely challenging. Around 16% of photos from this dataset are successfully localized within 200 km. The percentage of landmarks in the dataset was 5%, and the distance of the 5% of successfully localized photos was not more than a few kilometers. For famous landmarks, a city from which they originate can be often found. For 16% of photos, the method succeeds to find the country from which they originate. 84% of photos were not localized at all. Most problematic are obviously the photos from locations with small density of photos in the database (forests, mountains, deserts, etc.).

5.1.2 Cross-view image geo-localization

Cross-view image geo-localization method by Lin et al. [56] searches for correspondences between the query photo, ground-level photos, aerial view map (Fig. 2a) and land cover attribute map (Fig. 2b). The main observation is that the attribute map and an aerial view have some geo-location-sensitive information in common with the ground-level photos. The database is therefore created from the land cover attribute map (freely available from USGS GAP Land Cover Data Set[17]), an aerial map (from Bing maps) and the ground-level photos. The query photos are matched with the ground-level photos. If not enough ground-level photos are available, the cross-view localization is used. This is the biggest improvement over the previous IM2GPS method, which is not suitable for such isolated images.

The method is evaluated on an area of 1600 km^2 around Charleston, SC. Although the region is highly diverse and exhibits several different kinds of scenes (urban, agricultural, forest, marsh, beach, etc.), it is much smaller than the area tested in IM2GPS [33].

5.1.3 Geo-locating static cameras

Another approach to global geo-localization has been studied by Jacobs et al. [38]. They approach the problem of geo-locating static webcameras in outdoor. Such cameras are freely available for observing weather conditions and other purposes. Neither landmarks, field-of-view overlaps, nor any other prior information about the environment of the camera is known. Therefore, this work builds on temporal variations in natural environments—day and night, illumination changes due to changes in cloudiness or seasons. Several settings are tested—the camera is localized according to the weather satellite imagery or imagery obtained from other static cameras with known location. Generating satellite images from geo-located static cameras has been also studied.

The method is based on estimation, which pixel in the geo-aligned satellite map corresponds to the query image time series. It is shown that particular components contain scene-dependent and scene-independent coefficients. These coefficients encode mainly illumination changes due to sun position, day and night and weather conditions. The query image time series $I \in \mathbb{R}^{p \times \tau}$, where p is the number of pixels and τ is number of temporal frames, is decomposed using PCA, so that $I = U \Sigma V^T$. Pixels of satellite images are rearranged to matrix $S^{p \times \tau}$. To estimate the location, k PCA coefficients in matrix $V^{\tau \times k}$ are correlated with pixels of satellite time series imagery stored in matrix S. For location estimation, satellite weather imagery maps, as well as sun

[17] http://gapanalysis.usgs.gov/gaplandcover/.

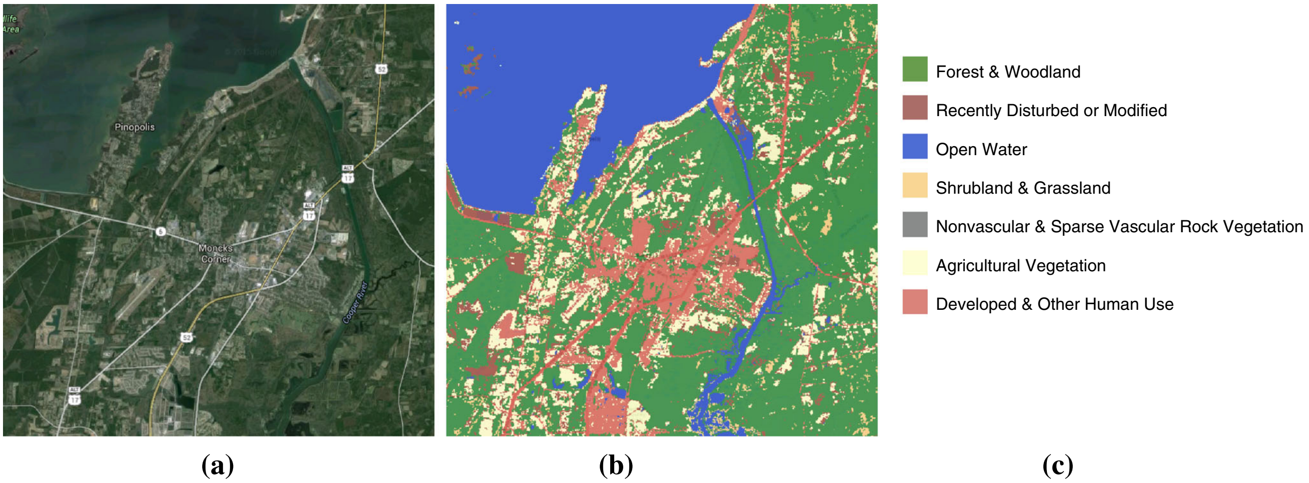
Pinopolis
Moncks Corner
Cooper River
(a)
(b)
(c)
🟩 Forest & Woodland
🟥 Recently Disturbed or Modified
🟦 Open Water
🟨 Shrubland & Grassland
⬛ Nonvascular & Sparse Vascular Rock Vegetation
🟨 Agricultural Vegetation
🟥 Developed & Other Human Use

Fig. 2 Cross-view image geo-localization: Illustration of the map data. **a** Aerial map, **b** USGS attribute map, **c** Attribute map legend

illumination maps, or other cameras with known location can be used.

The experiments were done with a database of 538 static outdoor cameras (with published GPS position) located across the USA. Localization using satellite imagery was done on a subset of these cameras located in Pennsylvania and Maryland. Each region contained approximately 50 cameras. Mean localization error was 44.6 miles (71.77 km), but without the worst 8 outliers it improved to 23.78 miles (38.27 km). The main disadvantage of this method compared to other global geo-localization methods is that there is the need for time series imagery from the camera, i.e., localization of a single image is impossible. On the other hand, the localization precision is much better compared to IM2GPS [33] or cross-view image geo-localization [56].

5.1.4 *PlaNet—photo geo-location with convolutional neural networks*

Weyand et al. [90] approached visual geo-localization over the whole Earth by training Convolutional Neural Network model. The problem is defined as classification problem over geospatial areas—cells of irregular grid. The non-overlapping cells cover the globe and are adaptively divided according to the number of training images available in given area. The denser the coverage with ground-level images, the smaller are the cells. For this partitioning, the authors use Google's open source S2 geometry library.[18] Using this library, the sphere is hierarchically partitioned by projecting its surface on an enclosing cube. Total number of the cells was 26 263.

The authors use 125 million images with GPS tag from exif; 91 million for training, and 34 million for validation. The images were acquired from online photo sharing services. Only minor filtering of non-photographic images such as graphs or diagrams was applied; the dataset therefore contains a lot of location-unrelated images, such as food, pets or products. The model was trained for 2.5 months on 200 CPU cores using DistBelief framework [21]. The authors evaluate their work quantitatively and qualitatively, The quantitative study was 2.3 million photos from Flickr, only containing exif GPS tag, and 1–5 textual tags; therefore, a lot of the images do not contain any cue about its location. The error is calculated from the GPS-tag location to the center of the predicted cell. Authors report that the system was able to localize 3.6% of images with the error at most 1 km and 10.1% with the error up to 25 km. In the qualitative study, the results of the neural network were compared to human results using GeoGuessr Web site.[19] The authors showed that the network was consistently deliver results with lower error than human.

Applying the CNN to the visual geo-localization problem instead of IM2GPS method has its advantages. First, the network saw much more training images than IM2GPS, which should be advantageous. Second, as IM2GPS uses nearest neighbor search, size of the model grows with the size of training set; on the other hand, the model of CNN is independent of the training size and in this case uses only 377 MB. Further advantage is that for the classification framework is natural to express its uncertainty across the classes, so it is easy to build probability distribution of likely locations in case the method is unsure of exact location.

5.2 City-scale image geo-localization methods

City-scale geo-localization is focused on densely populated areas, e.g., large cities. These areas, especially the landmarks, are nowadays covered with many overlapping

[18] https://code.google.com/p/s2-geometry-library/.

[19] http://www.geoguessr.com.

photos. This allows utilization of approaches such as SfM [79], which has been used in city-scale environment by Argawal et al. [1]. Methods using SfM model in combination with SLAM techniques can be very accurate, the error of localization presented by Middelberg et al. [59] is below 1 m (Sect. 5.2.6). This undisputed advantage is compensated by very high computational complexity of the SfM point cloud construction. Problems of large-scale localization using large SfM models are further studied by Li et al. [55], who introduced a new method of prioritized feature matching (Sect. 5.2.4). Even larger problem of pose estimation in several cities using 3D SfM point clouds was presented in a follow-up by Li et al. [54] (Sect. 5.2.5). Another approach using image retrieval was presented by Schindler et al. [75]. This work studies vocabulary trees for effective searching in high-dimensional data. Proposed algorithms are tested on problem of city-scale localization (Sect. 5.2.1). Image retrieval methods for city scale were studied deeper by Baatz et al. and Chen et al. [6, 17]. They show how to use street-view imagery along with rough 3D city models for landmark identification (Sects. 5.2.2, 5.2.3). The localization in the city by Ramalingam et al. [68] uses skylines of high buildings (Sect. 5.2.7). Skylines turn out to be intuitive and natural choice for local features. Similar idea of using horizon lines for matching and localization in the mountainous terrain was used by Baatz et al. [7] (Sect. 5.3.1).

5.2.1 City-scale location recognition

One of the first works dealing with city-scale outdoor visual localization was presented by Schindler et al. [75]. They propose a method based on vocabulary tree for city-scale location recognition. The main issue here is an efficient search in large spaces of SIFT descriptors. A method creating static and dynamic vocabulary trees for efficient searching is presented. To address performance issues on such a big database, new algorithm called Greedy N-Best Paths Search improving the basic Best Bin First (BBF) algorithm is presented.

The method is evaluated on a dataset that consists of 30K images automatically captured by a vehicle driving through a city. Each photo has latitude, longitude and compass heading information. Localization is considered successful when the distance of a top match is within 10 m from the ground truth. Besides the performance experiments, it is stated that the method localizes successfully 80% of query images which were taken one year after acquisition of the training dataset. Proposed method outperforms the k-d tree with BBF search strategy.

The most important contribution of this work are the experiments with large vocabulary trees. The novelty is application of image retrieval methods with vocabulary trees on such a large problem. Moreover, practical advices to obtain best performance for searching in large vocabularies are given.

5.2.2 Handling urban location recognition as a 2D homothetic problem

Several methods for city-scale location recognition make use of street-view panoramas [6, 75, 93, 94]. The idea of using street-view panoramas in accord with 3D models obtained by extruding the floorplans of known buildings is followed by Baatz et al. [6]. The query image captured by mobile device is searched in the database of images. Novelty of this approach resides in preprocessing the database and query images. The images are transformed, so that the relation between two matching images is homothety, which rapidly reduces the search space for geometrical verification step.

The street-view panoramas were mapped on rough 3D model of Places-of-Interest (POI)—extruded floorplans of buildings. From the whole panorama were extracted smaller rectified images with little field-of-view overlaps to sparsely cover the whole POI. DoG key points and SIFT descriptors with *k*-means clustering were used for vocabulary tree creation. Rectification of query image was done by finding vanishing points from strongest line segments. For each vanishing point the rectified image was calculated by removing the perspective. On rectified image the upright SIFT features were calculated to query the vocabulary tree. Best 50 candidates were tested with geometrical verification step. Since the matching correspondence between two rectified images is simple homothety, a simple voting principle was proposed to find the scale and translation of the query image. The candidate list was reranked according to the results of the voting stage.

Several setups for image matching were tested on three datasets. The experiments demonstrated that usage of upright SIFT features improves the recognition. There were big differences in reported performance on each dataset. The easiest dataset consisted of training and testing images taken under the same time and conditions. On this dataset, 85% of top-ranked candidates were correct. On more challenging datasets, with test photos taken under different conditions than training images, there were around 35% of correct top-ranked candidates.

5.2.3 City-scale landmark identification on mobile devices

Landmark identification is related to localization—when a landmark is identified, the search space reduces rapidly to the area in the neighborhood of that landmark. Chen et al. [17] studied the landmark identification problem by fusing ideas of Schindler et al. [75] and

Baatz et al. [7]. The main contribution of this work is a publicly available dataset[20] of street-view imagery captured by moving vehicle around San Francisco. The landmark identification on mobile devices has been tested on this dataset as a first benchmark.

3D city model captured by LIDAR sensors and street-view imagery taken by moving vehicle are stored in a database. The images of the streets are taken as panoramas recalculated to perspective images. Position of buildings in the street-view panorama is calculated based on the 3D model. With this known position, the image is segmented and cropped to contain building centers only. This new image is called Perspective Central Image (PCI). For each PCI, a Perspective Frontal Image (PFI) without perspective distortion is calculated. Next, the PCI and PFI images are treated in two parallel branches. For both branches, vocabulary trees on SIFT descriptors of Upright Feature Key points are trained. According to the GPS readings distant images are excluded. A query image is matched against PCI's and PFI's independently, for each branch different matching scheme is used. Matching results from these two branches are merged to obtain final result, which by is far more precise than separate PCI or PFI matching result.

According to the experimental results, the combination of PCI and PFI matching improves the final result around 10%, with 65% of correctly matched candidates. Despite good matching precision, which is advantageous in this approach, complicated and tedious data collection remains the main problem. This fact is addressed in the following localization methods that often use publicly available photos.

5.2.4 Location recognition using prioritized feature matching

Li et al. [55] create a 3D model using SfM from a large set of photos downloaded from the Internet (see Fig. 3a). For that purpose, a standard feature matching with SIFT descriptors is used. As the highest density of photos is near famous landmarks, this paper brings some basic ideas about compressing (pruning) the image database to contain more even distributed photos. It is shown that the compression brings speedups in the localization phase without loss of precision. The localization compared to other methods is fairly precise—mean of the error is 18.3 m and the median is 9.3 m.

An efficient search of the feature correspondences is the key problem. To address it, two search strategies were developed. SIFT descriptors in the image are denoted "features" and SIFT descriptors in the model are called "points." Two search strategies are presented—"Feature to Point" (F2P) and conversely "Point to Feature" (P2F). In F2P strategy, query features are searched in the model. This is a standard, but inefficient approach. More powerful P2F strategy takes model features and compares them against query image features. To avoid comparing all points to each feature, a smart prioritization is introduced—the points are treated with priority. It is shown that usually a small number of features need to be searched to find a feasible match. Further contribution is the clever selection of the right features to be evaluated first, which lowers number of feature comparisons needed to find the correct match.

According to the presented results, the true advantage is the speed—the result of the localization is obtained by a single-threaded process within 4 s. However, the method is also precise, the localization error is 18.3 m on the test dataset, but exceptions with error of 400 m exist. The disadvantage is the need of SfM model, which requires lots of images for construction, making this method efficient only in largely populated areas.

5.2.5 Worldwide pose estimation using 3D point clouds

In the follow-up work by Li et al. [54], the authors study image geo-localization in the worldwide scale with a fine pose estimation. As in their previous work [55], the SfM model of 3D points of various places and landmarks is built. Query photos are localized by matching to the visual descriptors stored in the 3D model. Each 3D point contains a pointer to all SIFT descriptors, from which the point has been constructed.

The algorithm consists of two main stages. In the first stage, some matches are found by forward matching. This basic set of matches is augmented in the second stage with an inverse matching (image features from the model near the points found during forward matching are searched in the query image). This yields a broader set of correspondences which are examined to find the camera pose. Final matches are obtained by new technique called "Sampling with Cooccurrence Prior" which is, unlike RANSAC, designed to cope with many outlier candidates. The idea is that the set M of matching points is divided into smaller sets of points that appear together. The probability of selecting such a subset for RANSAC round is measured by magnitude of the intersection of images where the points are present. Intersections of K-image subset are calculated in advance, so the probability of selecting each point is quickly available at runtime. Subsets with high probability are inspected with RANSAC with 3-point or 4-point algorithm for camera pose estimation. Finally, the bundle adjustment is used to fine tune the final pose.

[20] http://purl.stanford.edu/vn158kj2087.

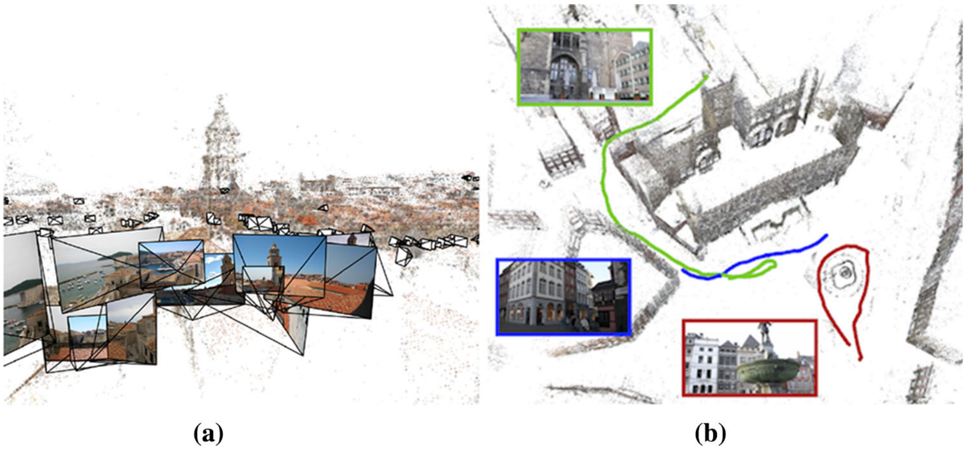

Fig. 3 Illustration of SfM and SLAM methods. **a** 3D SfM point cloud with poses (Image credit: Li et al. [55]) **b** SLAM using 3D SfM model (Image credit: Middelberg et al. [59])

This method succeeds to register 73% of the images from the challenging Quad dataset [20] within a few seconds. The mean localization error on Quad dataset is 5.5 m, median is 1.6 m. 90% of the query images were localized with location error smaller than 10 m. Most of the errors are caused by wrong matching of very similar parts of the image such as logos or flags.

5.2.6 Scalable 6-DOF localization on mobile devices

Recently proposed method by Middelberg et al. [59] is concerned with the localization and camera pose estimation on mobile devices. Due to memory and performance constraints of the mobile platform, the solution is decoupled to the device and the server parts. The mobile device computes local pose estimation, the server backend calculates city-scale localization and global pose estimation. Finally, the results are merged to obtain precise localization and pose (Fig. 3b).

More specifically, the method makes use of local and global 3D model. The local one is created on the fly on the mobile device—several keyframes are needed to create the model using SLAM technique. The global model is created by standard SfM technique on the server. For a good accuracy, the technique for merging global and local models is crucial, and therefore, two methods of merging are introduced.

The proposed methods are accurate in both localization and camera orientation estimation. Localization error is usually smaller than one meter. Furthermore, the camera orientation estimation error is up to several degrees. The system is highly responsive; the query frame is processed in approximately 50 ms.

5.2.7 SKYLINE2GPS: localization in urban canyons using omni-skylines

SKYLINE2GPS by Ramalingam [68] is an innovative approach to image localization in big cities. It attempts to localize the image using the shape of urban canyons (i.e., narrow streets with tall buildings) when looking upwards.

A query photo of the urban canyon from the street level while looking upwards is captured and segmented to obtain a skyline. Candidate skylines are rendered on a skydome and matched to the query skylines with the following two methods. The first method is based on the segmentation of the sky using a minimal cut in a graph. The input labeling for minimal cuts was trained for automatic estimation. After the skyline is obtained, chamfer distance from the synthesized skylines is calculated. The second approach is based on a shortest paths algorithm. The max–min operator for every pixel is applied—big value is assigned to edges far from other edges. The shortest paths algorithm is used to obtain final matching cost, which is minimized over matching candidates.

This method achieves good localization results. The mean error in the author's experiments is 2.8 m while the GPS is 29 m due to tall buildings and therefore reduced reception of the GPS receiver. However, experiments were done on relatively small area (part of Boston, New York and Tokyo), 6 km in total. The segmentation using minimum cuts did not perform well—in Tokyo there were problems with trees, in New York the system was not able to find correct matching for night photos. Similar idea of using horizon line for geo-localization in the mountains was also used by several other authors [18, 30, 63, 64, 81, 83, 84, 87] (Sect. 2.2.1). Matching using the skylines feels natural and intuitive, but it has limitations. One of the biggest problems is that the matching performance is directly dependent on the quality of the skyline recognition. Occluded or wrongly recognized skyline often leads to a failure of the matching system.

5.2.8 Accurate image localization based on Google Maps Street View

Zamir and Shah [94] published method based on nearest neighbor tree search with custom pruning and smoothing steps for better accuracy and to lower storage complexity. The method is trained by computing SIFT descriptors for detected interesting points by SIFT detector. The descriptors are then stored with their corresponding GPS tags in a tree using FLANN library [62]. For a query image, nearest neighbors are found. The matches are then pruned by step function which uses the nearest neighbor ratio test for each nearest neighbor that is farther than a constant. After the pruning stage, the nearest neighbors vote for location in the search space. The votes are further smoothed by gaussian function to prevent votes scattering. The voting function is normalized and considered as a probability distribution function. The authors propose to use the Kurtosis of this distribution as the measure of the confidence of localization. The authors also proposed method for localization of group of photos for better robustness.

Google Street View dataset consisting of 100K images was used in this work. The test set consisted of 521 GPS-tagged images. For evaluation purposes the whole dataset was divided into 5 trees. The results illustrated improvement over method by Schindler et al. [75] by large margin of almost 50%; the single image geo-localization method was able to localize almost 80% of images within the 250 m error, while the method by Schindler et al. [75] was able to localize only 40% of images within the same error. Group image localization method showed even better results with growing number of photos in the group up to 5 images.

5.2.9 *Image geo-localization based on multiple nearest neighbor feature matching using generalized graphs*

In a follow-up, Zamir and Shah [93] further improved their previous method [94] for visual geo-localization in urban areas. The pipeline of the geo-localization system is similar to the previous one [94], but instead of smoothing step they use a multiple nearest neighbor feature matching based on Generalized Minimum Clique Graphs. Local features are extracted from the training set and organized into a tree for nearest neighbor search. The global features comprising of color histograms and GPS position are extracted as well; they are used in a cost function of the following problem.

The method is formulated as an optimization task over graph $G = (\mathrm{V}, E, \omega, w)$, where V is set of nodes (database features), E is set of edges, ω defines node costs an w defines edge weights. Nodes are clustered so that each cluster contains k-nearest neighbors for each query feature. Nearest neighbor for each query feature is represented by a subgraph of G, with one node selected from each cluster. Such a subgraph is a complete graph. The cost function is defined so that it takes into account local feature similarity, as well as global feature similarity between the database images. As the problem is NP-hard [22], the authors proposed to use approximate solver based on local neighborhood search.

The results illustrated that the proposed method is much more accurate in terms of localization error compared to the previous works [75, 94]. The method was able to localize around 10% more images within the 250 m error compared to Schindler et al. [75]. The authors also published their dataset of 102K Google Street View images (see Sect. 3).

5.2.10 *GPS-tag refinement using random walks with an adaptive damping factor*

Zamir et al. [95] further worked on the problem of automatic robust localization with a database contaminated by unknown number of noisy GPS tags. From viable matches, triplets of two matching database images and the query image are formed. The method utilizes SfM technique to provide camera pose estimation for large number of triplets. To filter out noisy triplets, random walks on the estimations are used. This provides subset of reliable triplets, from which the result location is inferred.

Again, the optimization problem is defined as a complete graph $G = (\mathrm{N}, \mathrm{E})$, where N corresponds to location estimations, and **E** represents edges. The probability of transition from one node to another is modeled according to their GPS distance. Initial score for random walks is modeled so that it penalizes images from spots with high density of images in the database, so the method does not necessarily converge to the densest place. To cope with noisy location estimations, the authors proposed to use adaptive damping factor instead of a constant damping factor. After the algorithm converges, the final location is calculated as a mean of estimated positions weighted with scores calculated by random walks.

The method was evaluated on a dataset of 18K GPS-tagged images. The authors manually verified the accuracy of the 8K GPS tags and showed that there is from 10 to 30% of inaccurate tags (error greater than 30 m). For evaluation, 500 of images from the dataset were manually annotated.

5.3 Geo-localization in natural environments

Recently, geo-localization in natural environments has gained attention, in particular in mountainous terrains and deserts. All the following methods use a digital elevation map (DEM) or 3D model of a terrain for cross-domain matching. Baatz et al. [7] use horizon lines to find the position and rough estimation of camera orientation (Sect. 5.3.1). Geo-localization of untagged desert imagery was studied by Tzeng et - al. [87], who proposed novel skyline-based feature based on concavities (Sect. 5.3.2). A method for estimating the orientation of the camera with a known position was proposed by

Baboud et al. [9] (Sect. 5.3.3). Pose estimation on mobile devices has been approached by Porzi et al. [65] (Sect. 5.3.4).

5.3.1 *Large-scale visual geo-localization of images in mountainous terrain*

Baatz et al. [7] use horizon lines (see Fig. 4a) to construct local features (contourlets), which are stored in a large database and utilized by a Bag-of-Words-like approach for matching. The search space is represented by publicly available DEM of Switzerland, which has an area of 40,000 km^2. Visualization of such a DEM map is shown in Fig. 4b.

The elevation map is sampled with the resolution of 111 m in N–S direction and 115 m in E–W direction. In each sample point, the horizon line is rendered in the form of a cubemap. From each cubemap, contourlet descriptors are extracted and stored in the database of visual words. Several metrics for fast comparison of query and database contourlets were introduced. Voting for best candidates from the database is performed. Top candidates are geometrically verified using iterative closest points algorithm (ICP), which aims to register the query horizon line to a database horizon line. Result list is created from candidates sorted according to the error reported by ICP.

The main contribution is the efficient encoding of horizon lines to contourlet descriptors. Since each contourlet is encoded using only one integer, all the necessary comparisons can be performed efficiently. Further, the proposed design of voting for candidates reflects approximate orientation of each contourlet. Therefore, this method can find approximate direction of the camera.

Experiments suggest that for 88% of images in the test dataset the top candidate was within 1 km radius from the ground-truth position, which is good result. Interesting on this work is the fact that horizon line feature is often sufficient to obtain viable localization. However, there is the possibility to extend this approach by secondary horizons, depth information, sun position, surface normals, natural landmarks as rivers, lakes, glaciers or forests and other information. Using only single horizon line faces the same problems that were discussed earlier with the SKYLINE2GPS [68]. In the context of mountains, the risk of wrong horizon line detection is even bigger. Distant hazy horizons with ill-defined boundaries or partially occluded horizon by clouds or trees are the most frequent reasons.

5.3.2 *User-driven geo-location of untagged desert imagery using digital elevation models*

A user-aided geo-localization approach was presented by Tzeng et al. [87]. Similarly to the method by Baatz et al. [7], the authors utilize horizon lines (here called skylines). The user needs to sketch the skyline, which is automatically improved by the system. The novelty resides in the features extracted from the skylines: instead of contourlet features, the authors propose new concavity-based features. The features are normalized to gain scale and in-plane rotation invariance.

The query skylines are matched against skyline database generated from a DEM. The DEM is evenly sampled in the same way as in Baatz et al. [7]—in the form of a 2D grid with resolution of 1000 m along both north–south and east–west directions. In each sample point on the grid, the DEM is rendered to full 360° panorama, from which the skyline is extracted. The actual matching of query features to the database consists of two steps. In the first step only the feature endpoints are matched to prune the search space. During the second step more exhaustive search on the pruned search space is done. Final candidates are aligned and sorted according to the alignment error.

As the mentioned methods [7, 87] use different datasets for testing and different metrics for measuring the geo-localization quality, it is difficult to compare them in terms of the precision. According to the results, to obtain 50% of correctly localized images, there is the need to visit 10% of the search space. As the DEM is sampled sparsely (each 1000 m0), correct location can exhibit the error up to 1 km. As a future work, the authors propose to use secondary

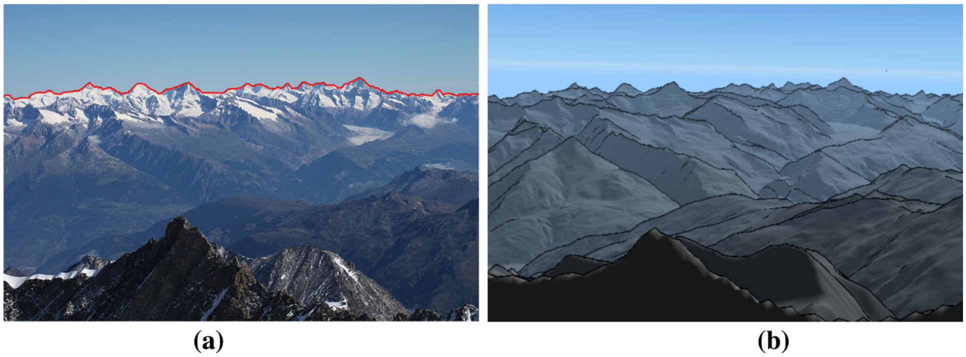

Fig. 4 Example of query image and corresponding DEM visualization. **a** Example of *horizon line* (*red*) **b** synthetic DEM (color figure online)

horizon skylines, because they contain abundance of distinctive information.

5.3.3 *Automatic photo-to-terrain alignment for the annotation of mountain pictures*

Baboud et al. [9] proposed a camera orientation estimation algorithm for mountain environments. The problem is defined as 3-degrees-of-freedom estimation—yaw, pitch and roll angles are subject to estimation. The authors are assuming the GPS position and FOV (or focal length along with the size of the camera sensor) are known. The algorithm utilizes the publicly available DEM rendered to synthetic panorama (portion of such a panorama is shown in Fig. 4b). From the synthetic panorama, the edges are extracted (using a depth of the scene). An edge map extracted from the query image (Fig. 5a, b) is matched against the synthetic silhouette map (Fig. 5c).

For most precise results the authors propose a robust silhouette map matching metric. Parallel overlapping edges contribute with positive value, while the crossing edges cause negative contribution to the matching score. Match with the highest score is considered the best. As the computation of this metric throughout the whole search space of three parameters would be too costly, custom vector-field cross-correlation (VCC) has been introduced to rapidly prune the search space. The cross-correlation also favors the edges with the same direction while edges crossings are penalized. Since the VCC can be calculated in a Fourier domain, it allows for fast approximation. The matching metric is finally evaluated on search space encapsulating the best candidates obtained by VCC.

According to the authors, the pure matching algorithm VCC with the matching metric without the edge detection runs approximately one minute on computer with two six-core Intel Xeon processors, one GeForce GTX 480 GPU and 23 GB RAM. The technique was able to successfully align 86% of query photos (28 photos from Flickr have been tested in total). The correctly aligned photos exhibit very small orientation error below 0.2°. While the results are very promising, problems can arise due to occlusion by clouds or trees, or incorrect EXIF data, as this method is very sensitive to wrong FOV. The authors report that the method is robust to small GPS position deviations, up to few hundreds meters.

5.3.4 *Learning contours for automatic annotations of mountains pictures on a smartphone*

Porzi et al. [65] propose a fast method of automatic photo-to-terrain alignment for precise augmented reality (AR) on a mobile device. The photo-to-terrain alignment was also studied in [9], but is unsuitable for usage in mobile environment since it is computationally demanding.

The query photo along with the GPS and rotation information from gyroscope and accelerometer sensors is captured using a mobile device. The mobile device extracts contours from the image, while the server renders the panorama silhouettes from DEM at given GPS coordinates. Query image contours are registered to the rendered edge map acquired from the server. Having precise camera pose, the query image is augmented with a metadata of the captured environment. The contour detection is approached as a classification problem—edges corresponding to edges on the mountain terrain shall be finally reported. Due to the resource constraints of the mobile platform, simple Random Ferns classifier is used, despite the random forrest classifier performed better in the evaluation.

Using only the information from onboard sensors and the Canny edge detector, the performance is almost real time. However, the registration error is above 5° making it unsuitable for AR applications. With the use of Random Ferns classifier, the registration error is around 1.3° with computational time of a few seconds (tested on Sony XPERIA Z).

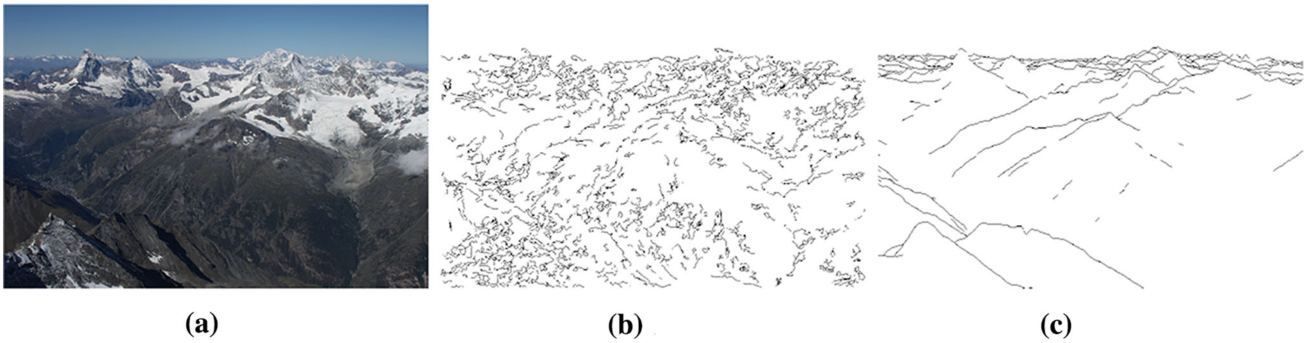

Fig. 5 Example of query image and corresponding silhouette maps. **a** Query image, **b** detected edges, **c** synthetic edges

5.3.5 Camera geo-location from mountain images

Chen et al. [18] extend the idea of visual geo-localization via horizon matching by using secondary ridge lines. They use semiautomatic approach to detect query ridge lines—the user selects the most important areas of the ridges and the system completes the line around using detected edges.

The authors incorporate similar voting process utilizing BOW-like approach on contourlet features [74]. The voting stage of the algorithm is extended for voting in both horizontal and vertical directions. This introduces problem with dimensionality of the voting array, which grew from 3D to 4D. After the accumulation stage only bin containing highest vote in vertical direction is kept. The authors also sample the FOV (0°–70°) and the roll angle ($-6°$ to 6°).

The method was tested on five 10,000 km^2 regions in the North and South America, and Asia. Digital elevation models were used from USGS NED [27]. Authors evaluate their results using fraction of query images within given geo-localization area (GA) (sum of all candidate areas that scored higher than are containing ground truth) and total area (TA) (sum of all candidate areas). The authors compare their approach to the approach using horizon line only and show that using multiple ridge lines is beneficial; over 80% of tested images lay within area of 10 km^2, which means that maximal error for more than 80% of images is 14 km at maximum. Method using only horizon lines localized 50% of images with the same error.

6 Applications

A number of works in this survey suggest many interesting applications. We give a review of applications mentioned in the surveyed articles. Visual geo-localization is in fact an application itself; from query image or video, we obtain a geographic location where the material has been captured.

In online applications, people can try their visual geo-localization abilities. *GeoGuessr*[21] site uses Google Street View panoramas as a query images, and people are supposed to make guess, where the panorama has been captured. *View From Your Window Contest*[22] is a similar Web site, where challenging sets of images are prepared to be geo-localized by people. Weyand et al. [90] has recently published evaluation of their geo-localization system, which was able to systematically beat geo-localization estimations made by people.

With the knowledge of camera pose of given image, systems for organization and visualization can be created, such as Photo Tourism [79]. With such an application, people can visit locations they have never been to and inspect the photos in their original pose at the time they were captured.

Various methods for digital photo enhancement were presented in Deep Photo [47]. The knowledge of location and orientation is crucial for methods such as model-based haze removal. Also another trick can be attained—illumination in the original image can be altered with the synthetic one, and the image can be augmented by labels or artificial segments such as paths or motorways.

Kendall et al. [45] recently published nice demo of their relocalization framework.[23] This online application can estimate the precise pose of the query image in the trained area. With such an application, people can localize themselves using their smartphone even without GPS.

Autonomous vehicles such as Junior [61] or UAV's are indeed another application of visual geo-localization. Such devices use several inputs, such as LIDAR, GPS, video and more to preserve robustness of location recognition. The vehicles actually need to solve many problems that are aimed by state-of-the-art in computer vision, such as pedestrian and traffic sign detection, self-localization, localization of other cars in traffic, reference speed measuring and more.

Google Goggles[24] is a mobile application from Google. It is able to recognize objects and identify landmarks as pointed out by Chen et al. [17].

7 The future of visual geo-localization

The ultimate goal of the visual geo-localization is to precisely estimate the *position* or even *orientation* of the camera, given a query image or a series of images. In order to analyze and discuss possible future research directions, let us summarize what actually is considered to be a *well-studied* problem and which problems are still *open* research challenges.

7.1 Well-studied problems

We consider problems which were addressed by a number of researchers as *well-studied* problems. However, there is still a chance that algorithms for these problems can be improved slightly. Expected improvements in this field are rather technical: more training data, and a faster training of machine learning techniques due to faster hardware, etc.

[21] https://www.geoguessr.com/.

[22] http://dish.andrewsullivan.com/vfyw-contest/.

[23] http://mi.eng.cam.ac.uk/projects/relocalisation/#results.

[24] http://www.google.com/mobile/goggles.

Generally, image-based visual geo-localization methods in urbanized areas are *well studied* these days. The two major branches of image-based algorithms for geo-localization or place recognition are based on: (I) image retrieval [2, 17, 75, 93–95]; and (II) Structure-from-Motion (SfM) [37, 45, 54, 55, 70–73, 82, 96]. The recent image retrieval-based paper, utilizing a VLAD layer inside a convolutional neural network (NetVLAD [2]), sets a promising state-of-the-art result with almost 90% of correctly found top-1 candidates. The core finding is that the CNN is able to learn the invariance to scene appearance changes such as night vs. day, winter vs. summer, with the use of Google Time Machine feature of a street view. Location estimated using this method is dependent on the precision of locations in Google Street View, which is around 30 m. An even more accurate position estimation, up to units of meters, can be achieved by SfM methods. Such methods need a large sparse 3D point cloud constructed from many images. This is quite a strict shortcoming of the method—the sufficient model is not available for the entire cities, usually just for city parts, and building such models is highly demanding on computational resources and time. Ideally, both approaches might be exploited at the same time, so as to estimate rough position, and refine it using an SfM camera pose estimation when a sufficient model is available. Visual SLAM is a *well-studied* topic as well [19, 29, 50, 51, 88], which is now able to localize a mobile device almost in real time [59].

7.2 Open problems

Open problems are less studied and usually more complicated than the *well-studied* problems, and so a major improvement in current research in this area may be expected in the future. The global localization [33, 90] is still an open problem. Recent work by Weyand et al. [90] utilized a massive CNN training on a dataset of hundred million images to predict likely localization using classification. While the experiments suggest that the method slightly outperforms humans in the task of visual geo-localization, only around 10% of query top-1 candidates were localized with the distance below 25 km, and around 15% were localized with the distance below 200 km from the ground truth. However, with respect to the overall complexity of the whole task, these results are promising. Despite the fact that machine learning-based cross-view approaches [56, 57] for matching different modalities exist, the possibility of utilizing multimodal data on the scale of the whole world has not been explored yet. Multimodal data for such a large scale might include orthophoto maps, satellite imagery, weather, digital elevation models or attribute maps.

Rapid scene appearance changes and self-similar, repeating patterns are still the largest obstacles for place recognition in the natural environment. Geo-localization in this domain was addressed mainly by retrieval using horizon lines and edge features [18, 30, 74, 87]. However, not only edge and horizon line features are descriptive in the natural environment. Semantic segmentation, an estimated depth of scene, normals or sun position [49] might be used as well. Geo-localization based on semantic segmentation has already been studied by several authors [3, 16, 76], but in urban areas only. Specifically for natural environments, semantic segmentation identifying forests, bodies of water, glaciers or rocks can significantly help to prune the search space in mountainous areas. However, this direction of research has been explored only slightly by Baatz et al. [8]. They developed a method for camera orientation estimation with a known location, based on an alignment of semantic segments detected in the query image and semantic segments rendered from a digital model.

While approaches for place recognition in urbanized areas, such as the NetVLAD [2], do not rely on the field of view of a camera at all, systems developed for localization in mountains often depend on it. Both Saurer et al. [74] and Chen et al. [18] need to sample the field of view for each round of the matching process, which is costly. Tzeng et al. [87], on the other hand, built the field-of-view estimation into the matching process of horizon line descriptors. Camera orientation estimation methods, such as the method by Baboud et al. [9], need to know the camera field of view as well. In future work, the need for the explicit field of view shall be eliminated.

An approach to geo-localize static cameras [38] based on weather patterns from satellite maps has been proposed. However, we are not aware of any work using a similar principle for a single image. Since weather is *time* and *location* dependent, we believe that such an approach could be adopted for single images as well. Furthermore, additional information such as the sun's position might be estimated from a single image and used with this approach. However, more prior information about the ground-level photos would be needed, such as the time when the image was taken.

8 Conclusion

Visual geo-localization is a new research topic which has recently been studied extensively. Initially, the idea of large-scale visual geo-localization has been explored by global localization methods. These methods take advantage of machine learning techniques combined with large image databases. Although large number of images is available worldwide, reliable global geo-localization (across whole world), it is still not enough accurate for practical use.

During the past few years, most of the research has been focused on city-scale localization. City-scale localization is often based on classical image retrieval methods which are applied in large-scale city environment. This became possible due to abundance of freely available images, which allows researchers to create extensive 3D point clouds with corresponding databases of images. Search in the large databases became (and remains) the key problem in this direction of visual geo-localization research.

Visual geo-localization in natural environments is far less explored category than city scale. However, the research activity in this field is rising. Outdoor localization is considered challenging due to the nature of the problem. First, natural environments are way larger than urban areas. Second, since the density of people in natural environments is minimal compared to the residential areas in cities, fewer images of natural landmarks are available. On the other hand, various kinds of data may be used—land cover attribute maps, aerial maps and DEM's are the primary source of information. Since not much is known about correspondence between real-world photos and these abstract data sources so far, a lot of future research resides in this field. Self-similarity and fractal nature of the outdoor scenes are another cues which may be utilized by future localization methods.

Acknowledgements This work was supported by SoMoPro II grant (financial contribution from the EU 7 FP People Programme Marie Curie Actions, REA 291782, and from the South Moravian Region). The content of this article does not reflect the official opinion of the European Union. Responsibility for the information and views expressed therein lies entirely with the authors. This work was also supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center - LM2015070".

References

1. Agarwal S, Snavely N, Simon I, Seitz SM, Szeliski R (2009) Building Rome in a day. In: Proceedings of the 2009 IEEE 12th international conference on computer vision. IEEE, New York, NY, USA, pp 72–79
2. Arandjelović R, Gronat P, Torii A, Pajdla T, Sivic J (2016) NetVLAD: CNN architecture for weakly supervised place recognition. In: Proceedings of the 2016 IEEE conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 5297–5307
3. Ardeshir S, Zamir AR, Torroella A, Shah M (2014) Gis-assisted object detection and geospatial localization. In: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (eds.) Computer Vision—ECCV 2014: 13th European conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part VI. Springer, pp 602–617
4. Aubry M, Russell BC, Sivic J (2014) Painting-to-3D model alignment via discriminative visual elements. ACM Trans Graph 33(2):14:1–14:14
5. Avrithis Y, Kalantidis Y, Tolias G, Spyrou E (2010) Retrieving landmark and non-landmark images from community photo collections. In: Proceedings of the 18th ACM international conference on multimedia, MM '10. ACM, New York, NY, USA, pp 153–162
6. Baatz G, Köser K, Chen D, Grzeszczuk R, Pollefeys M (2010) Handling urban location recognition as a 2d homothetic problem. In: Daniilidis K, Maragos P, Paragios N (eds) Computer vision—ECCV 2010: 11th European conference on computer vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part VI. Springer, Berlin, pp 266–279
7. Baatz G, Saurer O, Köser K, Pollefeys M (2012) Large scale visual geo-localization of images in mountainous terrain. In: Fitzgibbon A, Lazebnik S, Perona P, Sato Y, Schmid C (eds) Computer vision—ECCV 2012: 12th European conference on computer vision, Florence, Italy, October 7–13, 2012, Proceedings, Part II. Springer, Berlin, pp 517–530
8. Baatz G, Saurer O, Köser K, Pollefeys M (2012) Leveraging Topographic Maps for Image to Terrain Alignment. In: 2012 second international conference on 3D imaging, modeling, processing, visualization and transmission. IEEE, New York, NY, USA, pp 487–492
9. Baboud L, Čadík M, Eisemann E, Seidel HP (2011) Automatic Photo-to-terrain alignment for the annotation of mountain pictures. In: Proceedings of the 2011 IEEE conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 41–48
10. Bansal M, Sawhney HS, Cheng H, Daniilidis K (2011) Geo-localization of street views with aerial image databases. In: Proceedings of the 19th ACM international conference on multimedia, MM '11. ACM, New York, NY, USA, pp 1125–1128
11. Behringer R (1999) Improving registration precision through visual horizon silhouette matching. In: Proceedings of the international workshop on augmented reality: placing artificial objects in real scenes, IWAR '98. A. K. Peters Ltd, Natick, MA, USA, pp 225–232
12. Bergamo A, Sinha SN, Torresani L (2013) Leveraging structure from motion to learn discriminative codebooks for scalable landmark classification. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 763–770
13. Brown M, Lowe DG (2005) Unsupervised 3D object recognition and reconstruction in unordered datasets. In: Proceedings of international conference on 3-D digital imaging and modeling, 3DIM. IEEE, New York, NY, USA, pp 56–63
14. Brubaker MA, Geiger A, Urtasun R (2013) Lost! leveraging the crowd for probabilistic visual self-localization. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 3057–3064
15. Čadík M, Vašíček J, Hradiš M, Radenović F, Chum O (2015) Camera elevation estimation from a single mountain landscape photograph. In: Jones MW, Xie X, Tam GKL (eds) Proceedings of the British machine vision conference (BMVC), pp 30.1–30.12. BMVA Press
16. Castaldo F, Zamir A, Angst R, Palmieri F, Savarese S (2015) Semantic cross-view matching. In: 2015 IEEE international conference on computer vision workshop. IEEE, New York, NY, USA, pp 1044–1052
17. Chen DM, Baatz G, Köser K, Tsai SS, Vedantham R, Pylvänäinen T, Roimela K, Chen X, Bach J, Pollefeys M, Girod B, Grzeszczuk R (2011) City-scale landmark identification on mobile devices. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 737–744
18. Chen Y, Qian G, Gunda K, Gupta H, Shafique K (2015) Camera geolocation from mountain images. In: 2015 18th international

conference on information fusion. IEEE, New York, NY, USA, pp 1587–1596

19. Conte G, Doherty P (2009) Vision-based unmanned aerial vehicle navigation using geo-referenced information. EURASIP J Adv Signal Process 2009(1):10:1–10:18
20. Crandall D, Owens A, Snavely N, Huttenlocher D (2011) Discrete-continuous optimization for large-scale structure from motion. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 3001–3008
21. Dean J, Corrado GS, Monga R, Chen K, Devin M, Le QV, Mao MZ, Ranzato MA, Senior A, Tucker P, Yang K, Ng AY (2012) Large Scale Distributed Deep Networks. In: Proceedings of the 25th international conference on neural information processing systems, NIPS'12, pp 1223–1231. Curran Associates Inc
22. Feremans C, Labbé M, Laporte G (2003) Generalized network design problems. Eur J Oper Res 148(1):1–13
23. Fischler Ma, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applicatlons to image analysis and automated cartography. Commun ACM 24(6):381–395
24. Flatow D, Naaman M, Xie KE, Volkovich Y, Kanza Y (2015) On the accuracy of hyper-local geotagging of social media content. In: Proceedings of the eighth ACM international conference on web search and data mining, WSDM '15. ACM, New York, NY, USA, pp 127–136
25. Fry J, Xian G, Jin S, Dewitz J, Homer C, Yang L, Barnes C, Herold N, Wickham J (2011) Completion of the 2006 national land cover database for the conterminous United States. Photogramm Eng Remote Sens 77(9):858–864
26. Gallagher A, Joshi D, Yu J, Luo J (2009) Geo-location inference from image content and user tags. In: Proceedings of the 2009 IEEE conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 55–62
27. Gesch D, Oimoen M, Greenlee S, Nelson C, Steuck M, Tyler D (2002) The national elevation dataset. Photogramm Eng Remote Sens 68:5–11
28. Grzeszczuk R, Košecká J, Vedantham R, Hile H (2009) Creating compact architectural models by geo-registering image collections. In: Proceedings of the 2009 IEEE 12th international conference on computer vision workshops. IEEE, New York, NY, USA, pp 1718–1725
29. Hakeem A, Vezzani R, Shah M, Cucchiara R (2006) Estimating geospatial trajectory of a moving camera. In: Proceedings of the 18th international conference on pattern recognition, vol 2. IEEE, New York, NY, USA, pp 82–87
30. Hammoud RI, Kuzdeba SA, Berard B, Tom V, Ivey R, Bostwick R, Handuber J, Vinciguerra L, Shnidman N, Smiley B (2013) Overhead-based image and video geo-localization framework. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition workshops. IEEE Computer Society Press, Washington, D.C., USA, pp 320–327
31. Hao Q, Cai R, Li Z, Zhang L, Pang Y, Wu F (2012) 3D visual phrases for landmark recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 3594–3601
32. Hartley R, Zisserman A (2004) Multiple view geometry in computer vision. Cambridge University Press, Cambridge
33. Hays J, Efros AA (2008) IM2GPS: Estimating geographic information from a single image. In: Proceedings of the 26th IEEE conference on computer vision and pattern recognition. IEEE, New York, NY, USA, pp 1–8
34. Hays J, Efros AA (2015) Multimodal location estimation of videos and images. In: Choi J, Friedland G (eds) Multimodal location estimation of videos and images, chap. Large-scale image geolocalization, pp 41–62. Springer, Berlin
35. Heinly J, Sch JL, Dunn E, Frahm JM, (2015) Reconstructing the world* in six days. In: Proceedings of the 2015 IEEE conference on computer vision and pattern recognition. IEEE, New York, NY, USA, pp 3287–3295
36. Homer C, Dewitz J, Yang L, Jin S, Danielson P, Xian G, Coulston J, Herold N, Wickham J, Megown K (2011) Completion of the 2011 national land cover database for the conterminous United States-representing a decade of land cover change information. Photogramm Eng Remote Sens 81:345–354
37. Irschara A, Zach C, Frahm JM, Bischof H (2009) From structure-from-motion point clouds to fast location recognition. In: 2009 IEEE computer society conference on computer vision and pattern recognition workshops. IEEE Computer Society Press, Washington, D.C., USA, pp 2599–2606
38. Jacobs N, Satkin S, Roman N, Speyer R, Pless R (2007) Geolocating static cameras. In: Proceedings of the IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 1–6
39. Ji R, Gao Y, Liu W, Xie X, Tian Q, Li X (2015) When location meets social multimedia: a survey on vision-based recognition and mining for geo-social multimedia analytics. ACM Trans Intell Syst Technol 6(1):1:1–1:18
40. Johns E, Yang GZ (2011) From images to scenes: compressing an image cluster into a single scene model for place recognition. In: Proceedings of the IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 874–881
41. Kalogerakis E, Vesselova O, Hays J, Efros AA, Hertzmann A (2009) Image sequence geolocation with human travel priors. In: Proceedings of the IEEE international conference on computer vision, pp 253–260
42. Kelm P, Schmiedeke S, Sikora T (2011) A hierarchical, multi-modal approach for placing videos on the map using millions of Flickr photographs. In: Proceedings of the 2011 ACM workshop on social and behavioural networked media access, SBNMA '11. ACM, New York, NY, USA, pp 15–20
43. Kelm P, Schmiedeke S, Sikora T (2011) Multi-modal, multi-resource methods for placing Flickr videos on the map. In: Proceedings of the 1st ACM international conference on multimedia retrieval, ICMR '11. ACM, New York, NY, USA, pp 1–8
44. Kendall A, Cipolla R (2016) Modelling uncertainty in deep learning for camera relocalization. In: Proceedings of the international conference on robotics and automation (ICRA). IEEE, New York, NY, USA, pp 4762–4769
45. Kendall A, Grimes M, Cipolla R (2015) PoseNet: a convolutional network for real-time 6-DOF camera relocalization. In: Proceedings of the 2015 IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 2938–2946
46. Klein G, Murray D (2007) Parallel tracking and mapping for small AR workspaces. In: Proceedings of the 2007 6th IEEE and ACM international symposium on mixed and augmented reality, ISMAR '07. IEEE Computer Society, Washington, DC, USA, pp 1–10
47. Kopf J, Neubert B, Chen B, Cohen M, Cohen-Or D, Deussen O, Uyttendaele M, Lischinski D (2008) Deep photo: model-based photograph enhancement and viewing. ACM Trans Graph 27(5):1–10
48. Košecká J, Zhang W (2002) Video compass. In: Heyden A, Sparr G, Nielsen M, Johansen P (eds) Computer Vision – ECCV 2002: 7th European conference on computer vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part IV. Springer, Berlin, pp 476–490
49. Lalonde JF, Narasimhan SG, Efros AA (2010) What do the sun and the sky tell us about the camera? Int J Comput Vis 88(1):24–51
50. Larnaout D, Bourgeois S, Gay-Bellile V, Dhome M (2012) Towards bundle adjustment with gis constraints for online geo-

localization of a vehicle in urban center. In: 2012 Second international conference on 3D imaging, modeling, processing, visualization transmission. IEEE, New York, NY, USA, pp 348–355

51. Larnaout D, Gay-Bellile V, Bourgeois S, Dhome M (2013) Vehicle 6-DoF localization based on SLAM constrained by GPS and digital elevation model information. In: Proceedings of the 2013 20th IEEE international conference on image processing (ICIP). IEEE, New York, NY, USA, pp 2504–2508
52. Levinson J, Thrun S (2010) Robust vehicle localization in urban environments using probabilistic maps. In: 2010 IEEE international conference on robotics and automation. IEEE, New York, NY, USA, pp 4372–4378
53. Li Y, Crandall DJ, Huttenlocher DP (2009) Landmark classification in large-scale image collections. In: Proceedings of the IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 1957–1964
54. Li Y, Snavely N, Huttenlocher D, Fua P (2012) Worldwide pose estimation using 3d point clouds. In: Fitzgibbon A, Lazebnik S, Perona P, Sato Y, Schmid C (eds) Computer vision—ECCV 2012: 12th European conference on computer vision, Florence, Italy, October 7–13, 2012, Proceedings, part I. Springer, Berlin, pp 15–29
55. Li Y, Snavely N, Huttenlocher DP (2010) Location recognition using prioritized feature matching. In: Daniilidis K, Maragos P, Paragios N (eds) Computer vision—ECCV 2010: 11th European conference on computer vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, part II. Springer, Berlin, pp 791–804
56. Lin TY, Belongie S, Hays J (2013) Cross-view image geolocalization. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 891–898
57. Lin TY, Belongie S, Hays J (2015) Learning deep representations for ground-to-aerial geolocalization. In: Proceedings of the 2015 IEEE conference on computer vision and pattern recognition. IEEE, New York, NY, USA, pp 5007–5015
58. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
59. Middelberg S, Sattler T, Untzelmann O, Kobbelt L (2014) Scalable 6-dof localization on mobile devices. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer vision—ECCV 2014: 13th European conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, part II. Springer, Cham, pp 268–283
60. Mishkin D, Perdoch M, Matas J (2015) Place recognition with WxBS retrieval. In: CVPR 2015 workshop on visual place recognition in changing environments
61. Montemerlo M, Becker J, Bhat S, Dahlkamp H (2008) Junior: the stanford entry in the urban challenge. J Field Robot 25(9):569–597
62. Muja M, Lowe DG (2009) Fast approximate nearest neighbors with automatic algorithm configuration. In: International conference on computer vision theory and applications VISAPP ’09. SciTePress, Setúbal, Portugal, pp 331–340
63. Naval PC (1998) Camera pose estimation by alignment from a single mountain image. In: International Symposium on Intelligent Robotic Systems, pp 157–163
64. Naval PC, Mukunoki M, Minoh M, Ikeda K (1997) Estimating camera position and orientation from geographical map and mountain image. In: 38th Research meeting of the pattern sensing group, Society of Instrument and Control Engineers, pp 9–16
65. Porzi L, Buló SR, Valigi P, Lanz O, Ricci E (2014) Learning contours for automatic annotations of mountains pictures on a smartphone. In: Proceedings of the international conference on distributed smart cameras, pp 13:1–13:6. ACM, New York, NY, USA
66. Produit T, Tuia D, Golay F, Strecha C (2012) Pose estimation of landscape images using DEM and orthophotos. In: 2012 International conference on computer vision in remote sensing (CVRS). IEEE, New York, NY, USA, pp 209–214
67. Raguram R, Wu C, Frahm JM, Lazebnik S (2011) Modeling and recognition of landmark image collections using iconic scene graphs. Int J Comput Vis 95(3):213–239
68. Ramalingam S, Bouaziz S, Sturm P, Brand M (2010) SKYLINE2GPS: Localization in urban canyons using omni-skylines. In: 2010 IEEE/RSJ international conference on intelligent robots and systems. IEEE, New York, NY, USA, pp 3816–3823
69. Robertsone D, Cipolla R (2004) An image-based system for urban navigation. In: Proceedings of the British machine vision conference, pp 84.1–84.10. BMVA Press
70. Sattler T, Havlena M, Radenovi F, Schindler K, Pollefeys M (2015) Hyperpoints and fine vocabularies for large-scale location recognition. In: Proceedings of the 2015 IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 2102–2110
71. Sattler T, Leibe B, Kobbelt L (2011) Fast image-based localization using direct 2D-to-3D matching. In: Proceedings of the IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 667–674
72. Sattler T, Leibe B, Kobbelt L (2012) Improving image-based localization by active correspondence search. In: Fitzgibbon A, Lazebnik S, Perona P, Sato Y, Schmid C (eds) Computer vision—ECCV 2012: 12th European conference on computer vision, Florence, Italy, October 7–13, 2012, Proceedings, part I. Springer, Berlin, pp 752–765
73. Sattler T, Weyand T, Leibe B, Kobbelt L (2012) Image retrieval for image-based localization revisited. BMVA Press, Guildford
74. Saurer O, Baatz G, Köser K, Ladický L, Pollefeys M (2016) Image based geo-localization in the alps. Int J Comput Vis 116(3):213–225
75. Schindler G, Brown M, Szeliski R (2007) City-scale location recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 1–7
76. Senlet T, El-Gaaly T, Elgammal A (2014) Hierarchical semantic hashing: visual localization from buildings on maps. In: Proceedings—international conference on pattern recognition. IEEE, New York, NY, USA, pp 2990–2995
77. Shrivastava A, Malisiewicz T, Gupta A, Efros Aa (2011) Data-driven visual similarity for cross-domain image matching. ACM Trans Graph 30(6):1
78. Sivic J, Zisserman A (2003) Video Google: a text retrieval approach to object matching in videos. In: Proceedings of the Ninth IEEE international conference on computer vision, vol 2. IEEE, New York, NY, USA, pp 1470–1477
79. Snavely N, Seitz SM, Szeliski R (2006) Photo tourism: exploring photo collections in 3D. ACM Trans Graph 25(3):835–846
80. Snavely N, Seitz SM, Szeliski R (2008) Modeling the world from Internet photo collections. Int J Comput Vis 80(2):189–210
81. Stein F, Medioni G (1995) Map-based localization using the panoramic horizon. In: IEEE transactions on robotics and automation, vol 11. IEEE, New York, NY, USA, pp 892–896
82. Svärm L, Enqvist O, Oskarsson M, Kahl F (2014) Accurate localization and pose estimation for large 3D models. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 532–539
83. Talluri R, Aggarwal J (1992) Position estimation for an autonomous mobile robot in an outdoor environment. IEEE Trans Robot Autom 8(5):573–584
84. Talluri R, Aggarwal JK (1993) Image map correspondence for mobile robot self-location using computer graphics. IEEE Trans Pattern Anal Mach Intell 15(6):597–601

85. Thomee B, Shamma DA, Friedland G, Elizalde B, Ni K, Poland D, Borth D, Li LJ (2016) YFCC100M: the new data in multimedia research. Commun ACM 59(2):64–73

86. Tomasi C, Kanade T (1991) Detection and tracking of point features. Technical report, Carnegie Mellon University

87. Tzeng E, Zhai A, Clements M, Townshend R, Zakhor A (2013) User-driven geolocation of untagged desert imagery using digital elevation models. In: Proceedings of the 2013 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE, New York, NY, USA, pp 237–244

88. Vaca-Castano G, Zamir AR, Shah M (2012) City scale geospatial trajectory estimation of a moving camera. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 1186–1193

89. Viswanathan A, Pires BR, Huber D (2014) Vision based robot localization by ground to satellite matching in GPS-denied situations. In: IEEE international conference on intelligent robots and systems. IEEE, New York, NY, USA, pp 192–198

90. Weyand T, Kostrikov I, Philbin J (2016) Planet-photo geolocation with convolutional neural networks. In: Leibe B, Matas J, Sebe N, Welling M (eds) Computer vision—ECCV 2016: 14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, part VIII. Springer, Cham, pp 37–55

91. Woo J, Son K, Li T, Kim GS, Kweon IS (2007) Vision-based UAV Navigation in Mountain Area. In: Proceedings of the IAPR conference on machine vision applications (IAPR MVA 2007), pp 236–239

92. Workman S, Souvenir R, Jacobs N (2015) Wide-area image geolocalization with aerial reference imagery. In: Proceedings of the 2015 IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 3961–3969

93. Zamir AR, Ardeshir S, Shah M (2014) GPS-tag refinement using random walks with an adaptive damping factor. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition. IEEE Computer Society Press, Washington, D.C., USA, pp 4280–4287

94. Zamir AR, Shah M (2010) Accurate image localization based on google maps street view. In: Daniilidis K, Maragos P, Paragios N (eds) Computer vision—ECCV 2010: 11th European conference on computer vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, part IV. Springer, Berlin, pp 255–268

95. Zamir AR, Shah M (2014) Image geo-localization based on multiple nearest neighbor feature matching using generalized graphs. IEEE Trans Pattern Anal Mach Intell 36(8):1546–1558

96. Zeisl B, Sattler T, Pollefeys M (2015) Camera pose voting for large-scale image-based localization. In: Proceedings of the 2015 IEEE international conference on computer vision. IEEE, New York, NY, USA, pp 2704–2712

97. Zhang W, Košecká J (2006) Image based localization in urban environments. In: Third international symposium on 3D data processing, visualization, and transmission (3DPVT'06), pp 33–40. IEEE

98. Zheng YT, Zhao M, Song Y, Adam H, Buddemeier U, Bissacco A, Brucher F, Chua TS, Neven H (2009) Tour the world: building a web-scale landmark recognition engine. In: Proceedings of the 2009 IEEE conference on computer vision and pattern recognition. IEEE, New York, NY, USA, pp 1085–1092

99. Zhou B, Lapedriza A, Xiao J, Torralba A, Oliva A (2014) Learning deep features for scene recognition using places database. In: Advances in neural information processing systems 27. Curran Associates Inc, Red Hook, NY, USA, pp 487–495