Georg Schroth, Robert Huitl, David Chen,
Mohammad Abu-Alqumsan, Anas Al-Nuaimi, and Eckehard Steinbach

Mobile Visual Location Recognition

Low-latency and robust visual localization

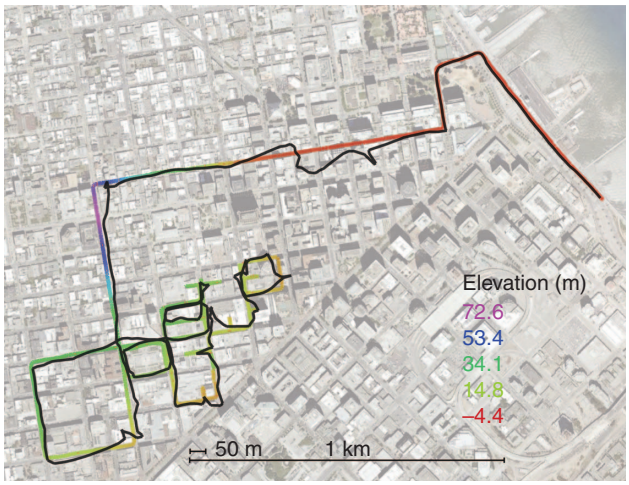
Mobility in Media Search

Information about the location, orientation, and context of a mobile device is of central importance for future multimedia applications and location-based services (LBSs). With the widespread adoption of modern camera phones, including powerful processors, inertial measurement units, compass, and assisted global positioning system (GPS) receivers, the variety of location- and context-based services has significantly increased over the last years. These include, for instance, the search for points of interest in the vicinity, geotagging and retrieval of user generated media, targeted advertising, navigation systems, social applications such as Foursquare [1], and many more.

Digital Object Identifier 10.1109/MSP.2011.940882
Date of publication: 15 June 2011

While satellite navigation systems can provide sufficient positioning accuracy, a clear view of at least four satellites is required, limiting its applicability to outdoor scenarios with few obstacles. Unfortunately, most interesting LBSs could be provided in densely populated environments, which include urban canyons and indoor scenarios. Figure 1 shows the GPS recordings (black line) of an iPhone 4 while driving a car through downtown San Francisco. Although a state-of-the-art-assisted GPS Broadcom chip is used, the phone mounting ensures the best signal reception, and a motion model is applied to filter out large deviations; the localization error is in the range of 50–100 m. This is caused by multipath effects, which are even more severe if the user is traveling on the sidewalks and not in the middle of the street. Here, an initial positioning

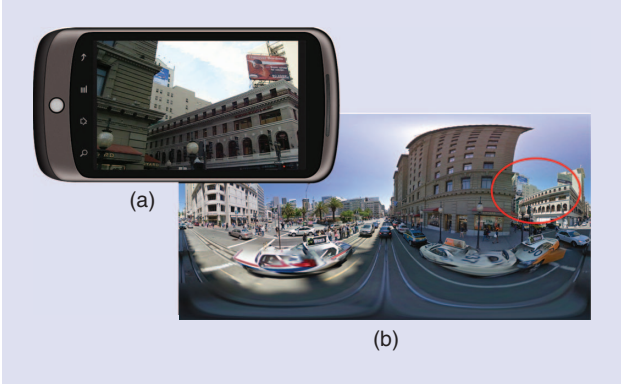
1053-5888/11/$26.00©2011IEEE



[FIG1] Track of video recordings in downtown San Francisco. The ground truth is shown as a colored path (the color corresponds to the elevation level), whereas the GPS recordings of the iPhone 4 are shown in black.

can take up to 40 s. The ground truth, as shown in Figure 1, has been recorded with the aid of a high-precision inertial measurement unit (IMU) and wheel odometry fused with the GPS signal. Clearly, the largest error is in the financial district, which is characterized by a dense collection of high buildings.

As GPS is virtually not available in indoor environments and the localization accuracy in urban canyons is insufficient, alternative positioning mechanisms, which can complement the available systems, are required. Adapted from human orientation, images recorded on the mobile device can be used as a visual fingerprint of the environment and matched to an existing georeferenced database such as Google Street View [2] or Microsoft Street-Side Views [3] (Figure 2). This allows us to derive the pose of the device in a very natural way. In contrast to WiFi-based indoor localization systems, no infrastructure, which grows in complexity with the size of the environment, is required. Further, LBSs do not only rely on a precise location and orientation information to determine the user's actual field of view but also benefit from information on its content such as exhibits, store names, trademarks, and so on, which can be derived from the images the user is intentionally recording. Ideally, the pose information from visual localization is fused with all other available sensor data providing location or orientation such as GPS, IMU, WiFi, or Cell-IDs if available. While it is only a matter of time until dense visual reference data will be available indoors [4], [5], this article focuses on outdoor scenarios using Google Street View panoramas. We review the specific challenges associated with mobile visual location recognition and discuss possible solutions. Experimental results illustrate the feasibility of a mobile visual location recognition system under realistic conditions.



[FIG2] Google Street View panorama matched to a low-resolution video recording (Union Square, San Francisco) using the visual location recognition system proposed in this article (a) video frame, and (b) database panorama image. (Figure used courtesy of Google.)

CHALLENGES OF MOBILE VISUAL LOCATION RECOGNITION

The main challenge for visual localization is the rapid and accurate search for images related to the current recording in a large georeferenced database. This task, known as content-based image retrieval (CBIR), has been an area of intensive research for the last few decades [6]. Objects, recorded at different sizes, poses, and with varying backgrounds, have to be distinctively described and efficiently retrieved from a database. The application of CBIR to location recognition complicates these requirements, as will be discussed later.

As illustrated in Figure 2, images captured with a mobile device are used to retrieve the spatially closest image from a georeferenced data set. This could, for instance, include the 360° panoramic images from Google Street View [2], which are used throughout this article. Typically, only sparse reference data can be assumed. For instance, Street View panoramas are available online with varying interpanorama distances, typically in the range of 12–17 m. The three exemplary panoramas shown in Figure 3 illustrate the associated problem of wide baselines. Although distant buildings can be well associated among the views, close objects are difficult to match even for a human observer. The description of distinct objects is complicated due to the three-dimensional structure of the environment and the resulting occlusions and overlaps. Further, different lighting conditions between the query and database image, which cause shadows and reflections, can change the visual appearance of the scene. In addition, both query and database images typically contain dynamic objects, such as cars or pedestrians, which lead to significant differences between matching views. As advertisements or even buildings alter over time and seasons significantly change the appearance, a dynamic update process for the database is required. Because of the properties of mobile device cameras, query images are typically affected by motion blur and provide a limited field of view, which makes it difficult to match them against high-resolution panoramas. Additionally, limitations on the processing power, battery capacity, and network performance require low-complexity approaches on the mobile device and efficient communication including data compression.

Finally, very low retrieval times are an essential prerequisite for most LBSs because of the rapidly changing field of view of the mobile device caused by the user's motion and constantly changing user's attention. On the other hand, vague prior knowledge of the location of the mobile device can always be assumed, e.g., derived from Cell-IDs, which should be exploited to increase precision, reduce complexity, and limit the requirements on the scalability of the approaches.

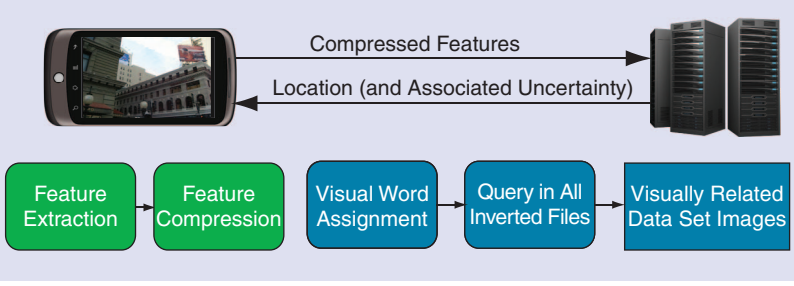
[FIG3] **Sample images from the Google Street View data set of San Francisco; online available panoramas are on an average 14 m apart from each other. (Figure used courtesy of Google.)**

BASIC SYSTEM ARCHITECTURE

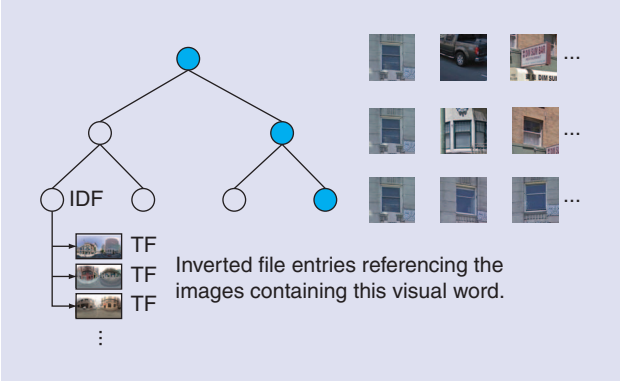
The most basic architecture of a visual location recognition system is, in principle, identical to that of a mobile visual product search system [7] (Figure 4), with the difference that not a single image but a continuous sequence of images is used to query a reference database. In the following, we will describe the fundamental modules (which are known to some extent to experts) because a good understanding of those is required to understand our extensions and adaptions (to facilitate location recognition).

As a first step, robust local features are extracted from the individual video frames to distinctively describe their content. During the last decade, various feature detectors have emerged, which provide some invariance with respect to scale changes, illumination conditions [8], [9], and affine transformations [10], [11]. As described above, these properties are of particular importance in location recognition. Mikolajczyk et al. provide a comprehensive evaluation of affine region detectors in [12] and conclude that maximally stable extremal regions (MSERs) [11] perform the best in most cases. Optimized implementations such as the approach proposed in [13] allow us to process an image of size 640×480 in 30 ms on a 3-GHz CPU, which can be further improved with the aid of parallel processing approaches on the graphics processing unit (GPU) of the mobile device. Tracking these regions and computing additional MSERs only from the previously unseen boundary textures allows for a five- to sixfold speedup [14], [15]. Altogether, approximately 10 frames per second (fps) can be processed on a Nexus S phone with a 1-GHz CPU. Descriptors of the image patch at these interest regions are computed to add additional robustness against view point changes and lighting conditions. The 128-dimensional scale-invariant feature transform (SIFT) [8] descriptor, its extension, gradient location, and orientation histogram [16], and its 64-D version speeded up robust features (SURF) [9] are considered state of the art. A comprehensive overview is provided in [16]. In this article, we use the MSER as the feature detector and either SURF or compressed histogram of gradients (CHoGs) [17] as descriptors. While CHoG achieves almost identical performance as SURF, it allows for a more than eightfold rate reduction. Once the features are extracted, they are sent to the server where state-of-the-art CBIR algorithms are employed to identify the visually most similar reference image in the database.

The popular bag of feature (BoF) approach proposed by Sivic and Zisserman [18] reformulates the image retrieval



[FIG4] **An overview of a basic visual location recognition system.**



[FIG5] Illustration of a tree-based BoF quantization and indexing structure. Image patches assigned to a single node decrease in diversity with increasing descriptor quantization. The leaf nodes are defined as visual words with associated inverted files. These reference the images to be scored when a query image patch is quantized to the respective word.

problem into a text retrieval one by quantizing the high-dimensional feature descriptors into the so-called visual words with the aid of the *k*-means algorithm. At fine quantization, descriptors associated with a word follow a texture pattern that is sufficiently represented by their mean (Figure 5). Thus, an image is no longer represented by its descriptors but by a visual word frequency histogram, the so-called BoF vector. The similarity between the two images is measured by the distance between their BoF vectors, which can be efficiently computed using inverted files [19]. Recently, several extensions and improvements have been proposed, which aim at a robust and efficient quantization of feature descriptors into visual words [20]–[26]. The major advantages of BoF-based image retrieval are its compactness, the reduced storage requirements, and the low query times. The interested reader is referred to "BoF-Based Location Retrieval" for more information and a comparison of the most prominent approaches with respect to their applications to visual location recognition. Although BoF-based algorithms allow us to cope with databases of approximately 1 million images, the very recently proposed approach in [27], which is based on vectors of locally aggregated descriptors (VLADs), achieves a significant reduction in storage requirements, facilitating retrieval within 10 million images. However, the search accuracy of BoF-based algorithms is not outperformed. Further, we found that VLAD faces difficulties when matching a regular image to all features extracted from one complete location as the aggregation of descriptors is sensitive to the image section. Since in visual localization the size of the reference databases is limited due to the availability of weak prior knowledge on the location, we employ BoF-based approaches to cope with the challenging retrieval task of location recognition.

Since wide baseline matching, an optimal adaption of the visual vocabulary to the properties of the reference database, and a flexible tradeoff between computational complexity and retrieval performance are of major importance in location recognition applications, the approximate k-means (AKM) (see "BoF-Based Location Retrieval") quantization turns out to be most suitable. Here, the number of kd-trees and considered leaves (the length of the priority queue) allows us to adapt to the available computational power. Nevertheless, the techniques proposed in the subsequent sections are very general and can be applied to any of the algorithms reviewed in "BoF-Based Location Retrieval" as well as VLAD-based approaches.

As the BoF representation ignores the layout of the features in the image, exploiting the spatial relation among them can improve the retrieval performance. Recently, several approaches have been proposed, which either postprocess a subset of the retrieved images to verify and rerank the results or integrate the information into the retrieval step [18], [21], [23], [29]–[31]. This optional geometric verification and reranking step concludes the image retrieval process on the server, and information on the estimated location and associated uncertainty are sent back to the mobile device.

While AKM requires about 200 ms on a 3-GHz server CPU to retrieve the visually most similar images (without geometric verification), the major bottleneck for this basic location retrieval architecture is the communication delay introduced by feature uploading. Utilizing low bit rate descriptors such as CHoG [17] allows us to send visual information from the client to server on a third-generation (3G) network in about 3 s as described in [7]. Including the network delay, communication timeouts, and retrieval itself, the delay until the client receives the results from the server is almost 6 s as reported in [7]. Clearly, this is not sufficient for most LBSs due to user motion and dynamically changing user attention.

In the subsequent sections, we will introduce approaches to cope with the two central challenges of mobile visual location recognition, which are the complex retrieval task and communication delay.

FEATURE EXTRACTION AND SELECTION

While local feature detectors such as SIFT, SURF, or MSER are designed to find distinctive image patches, they base their decision on the texture rather than high-level information such as the frequency of a feature in the database. Experiments show that most detectors produce a high number of features belonging to plants (e.g., foliage), which are unreliable due to the constantly changing texture and seasonal changes. Retrieval performance is improved by excluding those features, based on the color distribution of the corresponding image patch, which are mostly green (Figure 6). Hence, not only the quantization, indexing, and verification but also the selection of features is a major issue in visual location recognition. The selection of the most relevant features allows us to increase the performance and reduce the amount of data to be transmitted. Further, rotation invariance of the descriptor is not required in our application as it can be assumed that database as well as query images are recorded upright or rotated 90°. This not only

significantly reduces the extraction time on the mobile device but also increases the distinctiveness of the descriptors. In the following, we discuss appropriate feature extraction and selection approaches separately for query and database images.

QUERY IMAGE

Important information about the reliability of features can be derived from the query itself. As described above, we track features across several video frames, which allows for fast feature extraction [14], [15]. On the basis of time span, a feature is successfully tracked, and its stability with respect to view point changes and its importance regarding the actual pose are estimated. Further, moving objects such as cars or persons can be detected as the trajectory of associated features is not consistent with the majority of the features. Most importantly, the limited view of a single mobile phone image recording can be extended by generating virtual feature panoramas. These are generated by aggregating all reliably tracked features extracted from images within a short period of time and thus a limited local range. As users tend to pan their camera while recording, a significantly more distinctive query can be composed, which can be considered as a wide-angle recording.

Tracking the features on the mobile device also allows us to send only those that have not been transmitted before and the indices of those that disappeared. This allows for a further significant reduction of the data rate and thus network latency. This concept is also applied to reduce the computational complexity of the BoF-based image retrieval. As the image scores are composed of individual feature scores (see (3) in [7]), only these features that have not been assigned to a visual word before need to be quantized. The contribution of an individual feature to the score $S(i)$of database image i is shown in (1), where w_k is the inverse document frequency, c_{ki} the frequency of the corresponding visual word k in database image i, and Σ_i, Σ_q normalization factors for the database and query BoF vectors. The updated score $S_{\text{new}}(i)$ is composed of the score belonging to the last frame $S_{\text{old}}(i)$ minus the scores caused by disappeared features plus the scores caused by newly assigned features, as shown in (2)

$$S_{\text{feat}}(i) = \frac{w_k^2 c_{ki}}{\Sigma_i \Sigma_q}, \tag{1}$$

$$S_{\text{new}}(i) = S_{\text{old}}(i) - \sum_{\text{disappeared}} S_{\text{feat},d}(i) + \sum_{\text{appeared}} S_{\text{feat},a}(i). \tag{2}$$

As a result, for limited motion, the overall latency caused by transmission delay and feature quantization is significantly reduced.

DATABASE

In contrast to product recognition applications, we match regular images to spherical panoramas, which depict the complete 360° scene. Thus, only a fraction of the features that appear at a given location (usually around 3,000) can be found in the query. Further, as panoramas are usually provided in the form of equirectangular projections, objects are considerably distorted as shown in Figures 2 and 3. Thus, 12 rectilinear, partially overlapping views are generated via gnomonic projection as illustrated in Figure 6. Features extracted from these views are fused to represent a single location.

The selection of database features is based on the information gain of each visual word with respect to a certain location [22]. This gain is maximized if the visual word appears in neighboring locations but rarely at any other location. This approach implicitly filters unreliable features such as those resulting from moving objects and too generic textures such as windows. Our experiments show that comparable retrieval performance can be achieved at 20% of the original database size. Thus, a location can be represented with less than 1,000 features, which is comparable to the number of features required for a single query image.

For location recognition in dynamic environments that include cars, pedestrians, and advertisements, the use of static reference data is suboptimal. A dynamic update of the database can be achieved by integrating new features recorded by users. This requires that the position of the user has been estimated with sufficient reliability, which can be determined by the use of probabilistic filtering such as particle filters including motion models. Further, a weighting of features based on how often the feature has been matched at a given location or pruning of features that do not get matched can be performed. Altogether, the database can be adapted to local changes and hence the influence of dynamic objects reduced.

PRIOR KNOWLEDGE ON QUERY LOCATION

Ideally, a system for mobile location recognition based on visual data should not be bound to a confined area but allow for continuous and precise pose estimation at any place. However, the scalability of state-of-the-art BoF-based location recognition systems is limited to a few square kilometers and performance notably decreases with increasing database size [21]. Further, the size of the inverted file system scales linearly with the number of reference images and would require large amounts of RAM on the server. Covering a city like San Francisco with more than 100 km^2 would be an enormous challenge, however, is not necessary in practice as some coarse prior knowledge on the location usually exists. This information could for instance stem from the Cell-ID of the network provider, allowing us to determine a position estimate in the range of some hundred meters at most. Integrating this prior knowledge into the location recognition process reduces the required resources with respect to memory as well as query time and increases precision.

We propose to segment the search area, i.e., a large environment like a city, into several overlapping subregions for which individual quantization structures and associated inverted file sets are generated. The size of the subregions can be set to approximately 4 km^2 depending on the expected accuracy of the prior location information. In the example in

BoF-BASED LOCATION RETRIEVAL

The assignment of high-dimensional feature descriptors to visual words is a central and challenging task in BoF-based image retrieval algorithms since the quantization step reduces the distinctiveness of the descriptors. Although textual words can be unambiguously associated with their stems, the variance of the descriptors extracted from similar textures can be very large and significantly overlap with descriptor clusters of different textures. Thus, choosing the coarseness of quantization is a tradeoff between discriminative power and false association. This is particularly severe in high-dimensional spaces, as the number of adjacent quantization cells escalates with the dimensionality. Thus, the probability of assigning two matching descriptors, i.e., two descriptors representing the same physical structure, to the same cell is significantly reduced. Further, the time required for determining the cluster centers and the complexity of assigning a query descriptor increase with the increase in the size of the vocabularies.

As described in [7], a popular approach to cope with large vocabularies is the vocabulary tree (VT) proposed by Nistér and Stewénius [20], which hierarchically quantizes the descriptors using *k*-means clustering. This allows for increasing the number of visual words (e.g., 1 million) and, thus, the distinctiveness while significantly reducing the query time. In Figure 5, the decrease in diversity of image patches with increasing descriptor quantization can be seen.

While the *k*-means algorithm minimizes the total distortion between the data points and their cluster centers, its repetitive application in a treelike structure only minimizes the distortion within each node, i.e., only locally but not within the entire tree. Thus, Philbin et al. [21] propose using nonhierarchical *k*-means clustering, which results in minimal total distortion. To cope with the associated computational complexity, they construct eight randomized kd-trees [28] over the cluster centers at the beginning of a *k*-means iteration and use these to assign the data points to the (approximately) closest *k*-means (AKM) center.

To achieve the required low query times in location recognition, Schindler et al. [22] approach the quantization problem by considering multiple branches at each traversal step through the VT (greedy search).

Jégou et al. [23] follow a different approach by performing a hamming embedding (HE) of the feature descriptors to allow for a differentiation among them within a visual word. This scheme can be interpreted as an approximate nearest neighbor search in a dimensionality reduced and strongly quantized space.

A fair evaluation and comparison among the described algorithms can be achieved by adopting evaluation metrics from information retrieval research. Two evaluation measures are generally accepted in the CBIR community and are widely used: *precision* is the percentage of retrieved images (locations) that are relevant to the query and *recall* is the percentage of all the relevant images (locations) in the database, which are retrieved. Relevance is defined in location recognition by a given radius around the query location. The results can be summarized as precision–recall curves, where the number of images considered as retrieved is varied from the count of relevant images to a certain percentage of the overall database, e.g., 5%. In location recognition, high-precision values are of particular importance as postverification steps should be avoided to limit the overall processing time. In addition to these measures, the computational complexity and memory requirements, which are of particular importance in mobile visual location recognition, have to be considered.

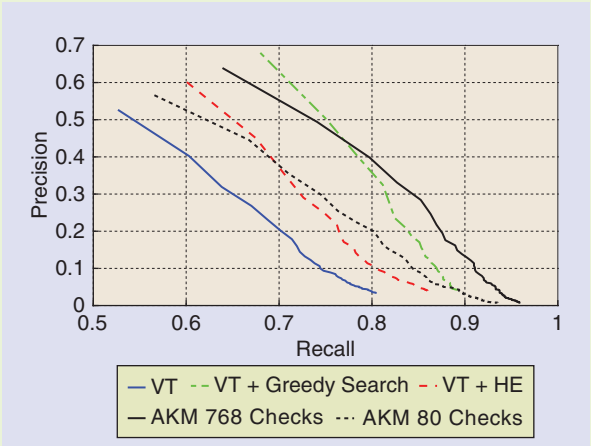
We compare the different approaches based on a georeferenced database extracted from Google Street View of an area of about 4 km^2, which consists of 5,000 panoramas at a distance of 10 m, each of them composed of 12 rectified images. As a query, we use views extracted from 500 panoramas with an opening angle of 70° and a resolution of 800×480 pixels. They are placed halfway between those panoramas included in the database, and their field of view is shifted by 45° to the left with respect to the direction of the street. For a maximum recall of one, all panoramas within a given radius around the query location have to be retrieved. This very challenging scenario allows us to effectively evaluate the properties of the individual approaches.

Figures S1 and S2 show averaged results for all 500 query images. According to Figure S1 where all panoramas within 10 m have to be retrieved, the basic VT quantization with 1 million leaves, branching factor $k = 10$, and tree depth $L = 6$ seems to be clearly inferior to the other approaches. However, as described above, only kL L_2 distance computations are required per query, which renders the approach very fast. Applying a greedy search [22] at query time significantly boosts the performance while requiring $k + kN(L - 1)$ L_2 distance computations, where N

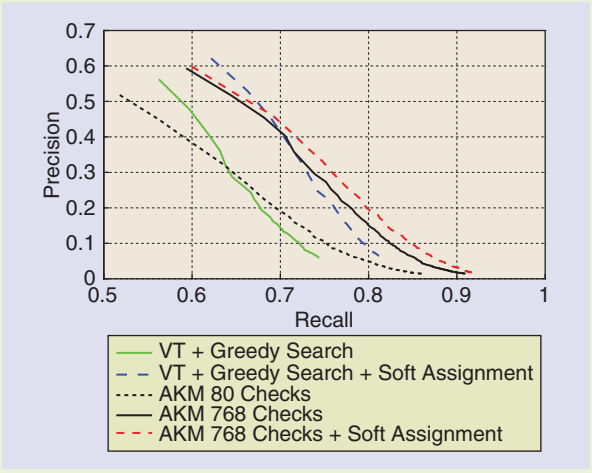
Figure 7, the area is covered by four subregions, overlapping with horizontally and vertically adjacent regions by 50%. Thus, every distinct location in the double shaded center is covered by four quantization structures. The pattern is continued accordingly across the whole city or even larger areas, such that for every location in the search area four separate vocabularies exist.

In the localization process only those, usually four, vocabularies are queried that cover locations within the area of the current uncertainty about the position of the user (Figure 7). However, in case of large areas of uncertainty or overlaps with the borders of a subarea, at most nine vocabularies have to be queried, where at least one covers the whole area of uncertainty.

For one database image in the area of uncertainty, this always results in four redundant similarity estimates. Since these similarity estimates are determined by quantization structures built with a significantly differing image set, the



[FIG S1] Comparison of state-of-the-art quantization and indexing structures based on a Google Street View database. Panoramas within a radius of 10 m around the query location are considered relevant.



[FIG S2] Evaluation of soft assignment [24] applied to AKM and VT quantization structures. Panoramas within a radius of 20 m around the query location are considered relevant.

is the number of considered branches. In this configuration, $N = 10$ results in 510 L_2 distance computations. The HE [23] with 64 b signatures requires only about one third of the computations; however, this comes at the cost of significantly increased memory requirements to store the signatures and the projection matrices. The AKM approach [21] is set to perform 768 and 80 L_2 distance computations as part of the backtracking in eight randomized kd-trees to query a flat vocabulary of 1 million visual words. While AKM with 768 checks cannot outperform the greedy search when all panoramas within 10 m have to be retrieved, flat vocabularies can cope well even with larger baselines. In Figure S2, all panoramas within a radius of 20 m are considered as relevant to compare the robustness of the descriptor quantization with respect to perspective transformations. Here, the flat vocabulary-based AKM with 768 checks can clearly outperform the greedy search, which makes the AKM particularly suited for retrieval tasks where only limited reference data is available.

SOFT ASSIGNMENT

As discussed above, there exists a clear mismatch between the hard visual word assignment and the continuous feature descriptor space. While descriptors extracted from different images but identical physical textures can be generally assumed to be in the vicinity of the descriptor space, the actual distance can significantly vary. Because of the scale, rotation, and affine-invariant coordinate system of the region detectors, the variance can be hardly estimated as it depends on the image texture.

An approach to cope with the tradeoff between fine and coarse quantization is the soft assignment of descriptors to visual words. Philbin et al. [24] propose to assign a descriptor not only to the closest visual word but also to words in its vicinity. The corresponding BoF entries are additionally weighted with respect to the distance between the word centers and descriptor according to an exponential distribution. Thus, the chances that matching query and database descriptors are assigned to one of the visual words are significantly increased, while the costs on the query time are very limited.

In Figure S2 where all panoramas within 20 m have to be retrieved, we apply the soft assignment approach proposed in [24] to the AKM and VT with greedy search. While only a limited additional increase in performance is achieved for the AKM, the relative improvements are more significant for VT. This clearly shows that less optimized vocabularies as those of the VT can be efficiently augmented with the aid of soft assignment.

quantization results of each are slightly different from the others. Nevertheless, the similarity scores are comparable and can be fused for the actual retrieval as described in the following. This is similar to techniques like locality sensitive hashing [32] or randomized kd-trees [28], where the results of a large number of random quantization structures are fused. However, we do not randomize our quantization structure directly, instead we compute tailored quantization structures but for different image sets. Thus, combining the redundant results allows us to significantly increase the precision at the cost of increasing the query time by the number of considered subareas.

In addition to reduced complexity and improved precision, this approach further allows us to provide a seamless handover to adjacent regions. Once a user is approaching the border of the currently covered search space, i.e., the area covered by four quantization structures per subregion, adjacent structures and associated inverted files are

[FIG6] **Gnomonic projections of a Google Street View panorama recorded in San Francisco. Ellipses are fitted around affine covariant MSER [11] regions. Features attached to foliage are successfully detected and removed. (Figure used courtesy of Google.)**

preloaded to the RAM of the server. Thus, the position of the user is always close to the center of one of the trees.

The fusion of the retrieval results can be based on the returned similarity scores and the spatial relationship of the retrieved locations. As a first step, the top K results for each query are selected, and the fused similarity scores are calculated by concatenating the BoF vectors of all queried vocabularies. This is efficiently done by simply summing over all individual similarity scores.

Even though the database images are typically more than 12 m apart, adjacent images are likely to share parts of the scene. Thus, the similarity scores of reference images close to the selected result images indicate whether query features are also matched at different viewing angles. The information implied in local clusters of high similarity is utilized by weighting the scores with the similarity scores of reference images in the direct vicinity.

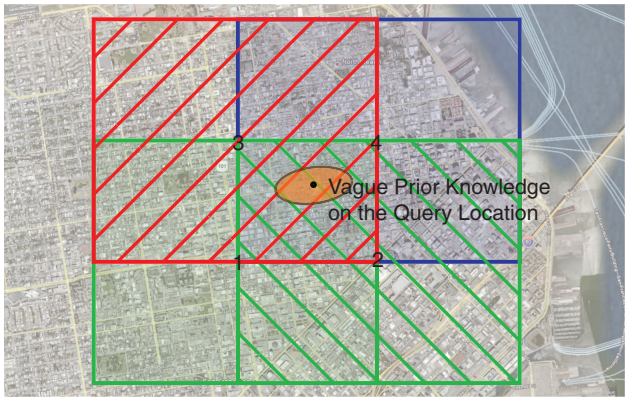
Finally, at most, K locations, sorted with respect to the final score, are returned. This algorithm effectively increases the precision of the returned results by fusing the information of multiple quantization structures at very low computational complexity. On the basis of these results, a geometric verification and the estimation of the pose of the mobile device via image registration can be performed.

APPLICATION SCENARIO

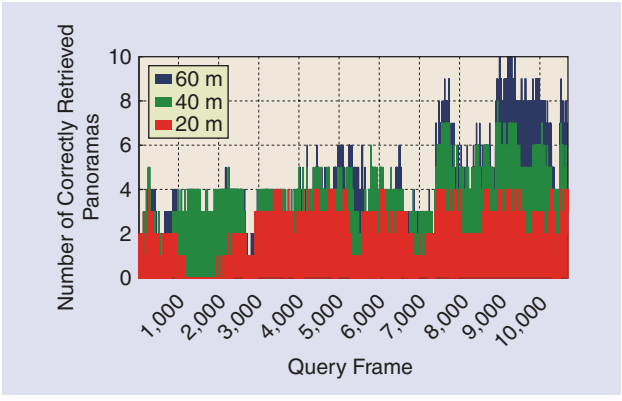
To illustrate the performance of a BoF-based mobile location recognition system based on the proposed feature selection and integration of prior knowledge, we query a Google Street View database of San Francisco comprising over 60,000 images at a spatial resolution ranging between 12 and 17 m. The realistic query video is recorded along a 0.7-km long section of the track, as shown in Figure 1, at a resolution of 800 × 480 pixels and an opening angle of 60°. Note that in some video frames, only trees or mostly clutters are visible. Further, occasional motion blur impairs the video quality.

A query is composed of tracked features within a time frame of 5 s. We conservatively assume an area of location uncertainty of a radius of 1 km. After 0.5 km, the track crosses the boundary of a subregion, and a handover occurs. The top ten results of the fusion of four subareas with a size of 4 km^2 each are considered. In this experiment, the retrieval is performed on the server, and every query is assumed to have the same coarse prior knowledge about the location of the user. The approach of spatial partitioning, as discussed above, is combined with the AKM algorithm.

Figure 8 shows the number of retrieved locations within a given radius around the true query location for the individual video frames. For 79.6% of the track, at least one panorama has been retrieved within 20 m and for 93.2% of the track within 40 m. Without the use of virtual feature panoramas and without the removal of green features, as discussed in the section "Feature Extraction and Selection," the performance significantly degrades. Only for 34.4% of the track at least one panorama is retrieved within 20 m. Note that on average three panoramas are located within a radius of 20 m, except for street junctions. Increasing the radius further results in a marginal increase in correct matches as the visual similarity is local. At frames

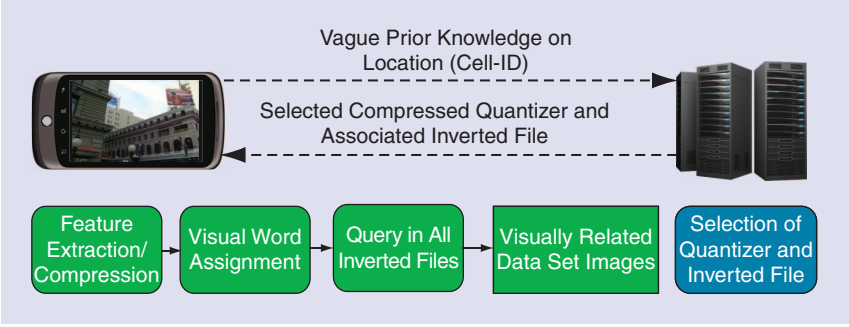


[FIG7] **Spatial partitioning of the search space in four regions/trees, which overlap by 50% in the horizontal and vertical directions. The double shaded area in the middle is covered by four trees.**



[FIG8] **Evaluation of retrieval performance for realistic query video recordings. The number of retrieved locations within a given radius around the true query location is plotted against the frame number.**

7,500–10,500, the video recordings show a square, such that distant locations are also visually similar. Unsuccessful localizations, which occur if the query feature set is degenerated, can be overcome with the aid of probabilistic filtering and visual odometry. Subsequently, an image registration can be performed to determine the pose of the mobile device with respect to neighboring panoramas.



[FIG9] Overview of a visual location recognition system where the retrieval task is shifted from the server to client by transferring the quantizer and compressed inverted file system of the most suitable subregion. Dashed lines indicate the data transmission during initialization. Once the transfer of the quantizer and inverted file is finished, no further network delay is introduced.

LOW LATENCY MOBILE LOCATION RETRIEVAL

With the above-described extensions of the basic system architecture in Figure 4, we significantly improve the localization performance with respect to precision and recall. Further, by transferring only selected tracked features from the client to server, the overall latency can be reduced to 2–3 s depending on the motion of the mobile device.

While this would be sufficient for mobile visual product search, it is not sufficient for visual location recognition applications. Ideally, a user is notified about an LBS the very moment the camera records the corresponding store, exhibit, and trademark. Because of the rapidly changing field of view and dynamically changing user attention, very low system latency is essential for the LBS to be perceived as useful. With the described basic system architecture in Figure 4, the response time always suffers from the round trip time, which ranges between 0.3 and 0.9 s on 3G networks, and time-outs in addition to the time required to actually transfer the data [7]. The architecture almost exclusively uses the slower uplink channel for uploading query features, whereas the downlink is usually more than five times faster. Further, the selection of features to be transferred can be hardly based on database statistics, as this would require the quantization into visual words.

A possible approach to solve this problem is to transfer the quantizer as well as the inverted file to the client and exclusively perform the location retrieval on the mobile device as shown in Figure 9. With the above-described approach to integrate prior knowledge, we limit the size of the inverted file system to about 5,000 panoramas and send the most suitable quantization structure. In our work, the quantizer is based on the AKM [24], which uses a flat vocabulary where descriptors are assigned to k-means centroids via randomized kd-trees, as described in "BoF-Based Location Retrieval." To reduce the download time, the centroids of the vocabulary are represented by CHoG descriptors requiring 60 b each. With the size of a kd-tree being less than 100 kB for 1 million centroids, the overall memory requirements range between 3.75 and 7.5 MB depending on the size of the vocabulary.

Additionally, the inverted file needs to be transferred, which can be efficiently compressed following the BoF compression approach in [33]. This allows us to reduce the size of the inverted file to about 12.5 MB. Further, spatially neighboring panoramas overlap with respect to the associated visual words by about 18% depending on their distance and the size of the vocabulary. We exploit this by transferring differentially encoded binary BoF vectors, which include only the difference to the visually most similar panorama. This allows for an additional reduction of the size of the inverted file by more than 10%.

On a Nexus S phone, the assignment of 1,000 descriptors to visual words requires about 180 ms using four kd-trees and 40 checks on a vocabulary of 500,000 words. This query time is reduced to about 50 ms by quantizing only the newly appearing descriptors depending on the amount of motion as described in the section "Feature Extraction and Selection." An even faster quantization (12 ms per 1,000 descriptors) is achieved using the approach proposed in [26], where fast rotation-invariant fast features descriptors [34] are combined with a binary VT.

Although this approach eliminates the network delay by performing the location retrieval step on the mobile device, the amount of data to be transmitted via the downlink would result in a time-consuming initialization of the system. During this phase, localization would need to be performed according to Figure 4, where features are sent to the server and the position can be updated approximately every 3 s.

ADAPTIVE PARTIAL VOCABULARIES

On the basis of the features transmitted from the mobile device to the server, the location of the client is periodically estimated at the server. This allows for a high performance with respect to precision and recall as described in the section "Prior Knowledge on Query Location" and shown in Figure 8. Providing the mobile device with reference data within a certain vicinity of these periodic location estimates (every 3 s) allows us to autonomously continue the localization for a limited time period.

If we consider $F = \{f_1, f_2, \ldots, f_N\}$ to be the set of features of one query frame and $V = \{v_1, v_2, \ldots, v_L\}$ to be the set of visual words (i.e., the full vocabulary), the quantization function $q_V(f_i) = v_j$ assigns each feature f_i to a visual word v_j in the full vocabulary V. Hence, $Q(F|V) = \{v = q_V(f)|f \in F\} = V_F \subseteq V$, determines the subset of visual words, which represents a particular video frame. Instead of the full vocabulary if we use only a random partial set of visual words $V_R \subseteq V$, the feature

quantization result will be different in most cases. However, using the subset of visual words representing the frame itself $V_F = Q(F|V)$ as the partial vocabulary would not change the result of the quantization as expressed in the following equation:

$$Q(F|V_F) = Q(F|V). \tag{3}$$

Hence, only this part of the full vocabulary needs to be available at the client to obtain the same results as if the full vocabulary is being used. However, this equation holds only for a specific set of features F, and a partial vocabulary V_F would need to be sent to the client for each frame. Ideally, we would like to identify a partial vocabulary that includes the correct visual words to process multiple consecutive frames without the need to know their features a priori. Since V_F can be extended by other subsets of the full vocabulary ($S \subseteq V$) without changing the quantization result, as shown in (4), we can use partial vocabularies at the client that have a sufficiently high probability of including V_F.

$$Q(F|V_F \bigcup S) = Q(F|V). \tag{4}$$

To limit the amount of data to be transferred to the client, we seek for the smallest partial vocabulary that includes the unknown V_F with high probability.

On the basis of the periodic location estimates obtained at the server, we do have prior knowledge on the location in the form of the top K retrieved locations. Thus, the partial vocabulary can be based on the visual words that are part of panoramas located at these candidate locations. Features of frames recorded at these locations are expected to be quantized to the corresponding visual words. Hence, the probability that V_F of these frames is part of this partial vocabulary is very high.

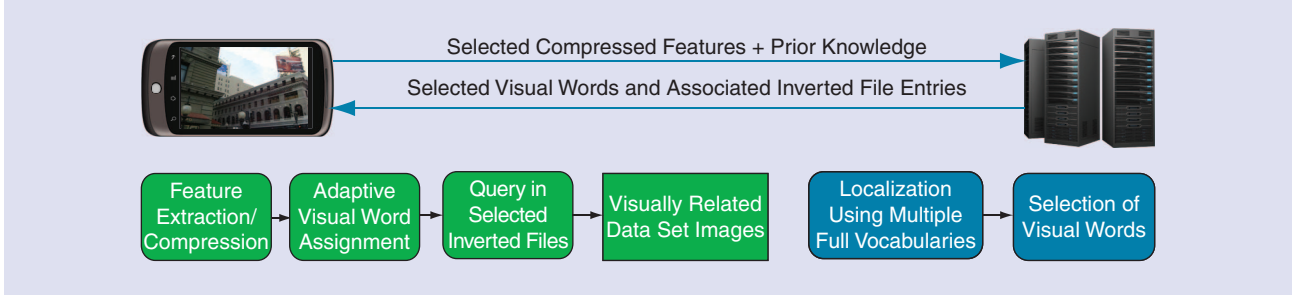
Hence, we propose to periodically send only these relevant visual words together with their associated inverted files to the client to allow for a local pose estimation on the mobile device within a limited area as shown in Figure 10. This is achieved by matching query features on the mobile device to the partial vocabulary, which can be performed at 10 fps. At an average motion of 1.2 m/s, the visual words of two neighboring panoramas (distance ranging between 12 and 17 m) allow us to navigate for at least 10 s or 100 frames without an update of the partial vocabulary. Because of the imperfect location retrieval at the server and/or degenerated query features sent, the probability that the first ranked panorama is located at the actual location is about 60% as shown in Figures S1 and S2 in "BoF-Based Location Retrieval." To achieve a probability of about 90%, the top five-ranked panoramas need to be considered to build a partial vocabulary.

Our experiments have shown that querying the partial vocabulary, which is based on the top K panoramas, actually results in an increased precision compared to the full vocabulary. This is due to the fact that with a high probability the correct location is among the top K results, but possibly not at the beginning of this ranking. As the partial vocabulary constrains the set of possible location candidates, the quantization of the features of a subsequent query frame to this vocabulary can be considered as a resorting among the original top K results. The panorama that fits best to both queries is implicitly ranked the highest.

As described above, features that vote for one of the preselected locations are matched to the same visual words as if the full vocabulary is being used. Only those features that refer to other locations with the full vocabulary can be matched to a different visual word when considering a partial vocabulary. Applying a threshold on the maximally accepted distance between the feature and visual word results in a marginal further improvement.

To make the best use of the available data rate of the downlink, a priority queue of locations and their visual words is updated every time a new set of features is sent from the mobile device to the server. Thus, we constantly send those visual words that belong to the panoramas with the highest probability of representing the actual location. This probability can be determined based on the score of the retrieval or with the aid of temporal filtering approaches such as Bayesian filters, which are out of the scope of this article. With a 3G downlink, the visual words and associated inverted file entries of at least two panoramas (each comprising approximately 1,000 visual words) can be transferred per second. Within the time a feature set is uploaded, we can download the visual words of about six locations.

As more visual words are transferred to the client, the chances that visual words in the priority queue are already



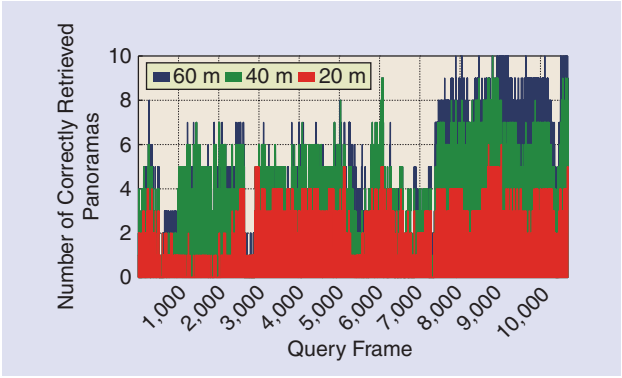
[FIG10] Overview of a visual location recognition system where relevant parts of the vocabulary are periodically pushed to the client. The selection of the transmitted visual words is based on the location retrieval performed at the server using the query features uploaded from the client. The pose estimation is performed on the mobile device utilizing these preloaded adaptive partial vocabularies. Blue lines indicate that the network delay does not influence the time required for the location recognition.

available at the mobile device increase, and hence, they do not have to be sent again. To increase the time the client can navigate based on the partial vocabulary, visual words of panoramas that are adjacent to the top K location estimates are added to the priority queue. Since they significantly overlap with respect to the visual words, only a limited amount of additional data has to be transferred. To speed up the quantization of feature descriptors to the adaptive partial vocabulary, which includes at most 50,000 visual words if the uncertainty about the actual location is large, randomized kd-trees are sent to the mobile device. As they require about 10 kB each (for 50,000 visual words), a frequent update can be carried out.

To evaluate the performance of this system, we use the same Google Street View data set as used for Figure 8. The realistic video recording along the 0.7 km long track is processed at 10 fps. Every 3 s, a new set of tracked query features is uploaded to the server and used to query the full vocabularies to retrieve the visually most similar locations. On the basis of these recognized location candidates, the priority queue is updated and approximately 5,000 selected visual words and their associated inverted files entries are downloaded to the client within 3 s. The pose estimation is continuously performed on the mobile device at 10 fps utilizing the adaptive partial vocabulary as illustrated in Figure 10.

Similar to Figure 8, we plot the number of retrieved locations within a given radius around the true query location for the individual video frames in Figure 11. The use of adaptive partial vocabularies on the mobile device allows us to retrieve at least one panorama within 20 m for 82.3% of the track and within 40 m for 91.0% of the track when considering the top ten retrieved locations. As shown in Figure 11, the number of retrieved locations within the given radius is significantly increased by adaptive partial vocabularies. No temporal smoothing approaches such as particle filters or geometric verification techniques are applied. The priority queue is exclusively based on the most recent set of features uploaded to the server.

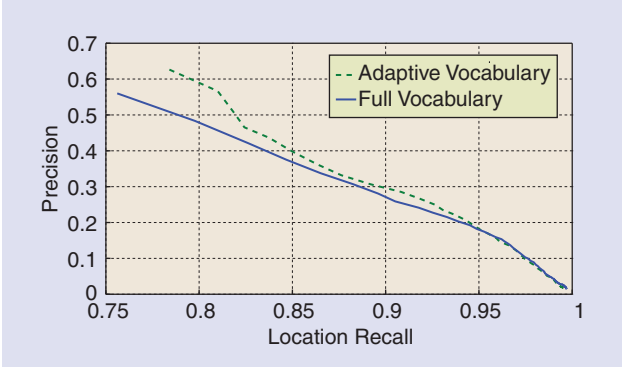
Further, we evaluate the performance improvement using precision and location recall metrics in Figure 12. Here, precision is defined as the percentage of retrieved locations that are within 20 m around the ground truth position. This measure is averaged over all queries. With this very realistic data set, the number of relevant panoramas, i.e., the ones within 20 m around ground truth, is different among the individual queries. Hence, the percentage of relevant locations retrieved (recall) cannot be averaged over the queries. Thus, we use location recall, which is a binary measure that is set to one if at least one relevant panorama is retrieved. By iterating over the number of considered retrieval results, multiple precision/location recall pairs are computed as shown in Figure 12. Clearly, a major performance increase with respect to the use of a full vocabulary can be observed. As discussed above, this is achieved by quantizing the features of a query frame to a preselected subset of visual words, which have a high probability of referring to the correct locations. As this preselection is based on a previous



[FIG11] Evaluation of the location recognition performance of adaptive partial vocabularies using the same data set and metrics as in Figure 10. A significant increase in the number of correctly retrieved panoramas per query location can be observed.

query, this can be interpreted as a resorting within the top results of this previous query.

With this proposed system, we facilitate a close to real-time pose estimation on the mobile device by eliminating the network delay, which would otherwise result in response times of 4–5 s when using the system architecture as shown in Figure 4 [7]. This allows the LBS to react to the constantly changing field of view and user attention. While still obtaining a location update by the server every 4–5 s, we additionally exploit the faster downlink to periodically transfer the vocabulary required to navigate within a limited area. As the visual words associated with the estimated and predicted locations are selected with respect to their information gain considering the overall database statistics, only the relevant information is downloaded. This stands in contrast to the uploaded query features where only limited knowledge on their information exists. Further, as humans usually move with about 1.2 m/s but rapidly change their orientation, the set of query features significantly alters over time, whereas the vocabulary required to cover the change in position is almost constant. Experiments show that after



[FIG12] Comparison of location recognition systems using full and adaptive partial vocabularies with respect to precision and location recall metrics. The use of adaptive partial vocabularies on the mobile device allows us to eliminate the network delay and moreover to increase the recognition performance.

about 4 min a significant fraction of the vocabulary has been downloaded, and more and more visual words in the priority queue do not need to be transferred. As selected features are uploaded to allow the server to select the right partial vocabulary, these features can be also used to update the database as part of the online process as described in the section "Feature Extraction and Selection."

CONCLUDING REMARKS

With recent advances in CBIR, mobile visual location recognition becomes feasible. Using video recordings of a mobile device as a visual fingerprint of the environment and matching them to a georeferenced database provides pose information in a very natural way. Hence, LBSs can be provided without complex infrastructure in areas where the accuracy and availability of GPS is limited. This includes indoor environments where georeferenced data are just about to become publicly available [4], [5].

However, the application of CBIR to mobile location recognition implies several challenges. The complex three-dimensional shape of the environment results in occlusions, overlaps, shadows, reflections, etc., which require a robust description of the scene. BoF-based representations allow us to match regular images to 360° panoramas. Dynamic objects such as cars, pedestrians, advertisements, and seasonal changes demand for a selection of features in both the query and the database. On the part of the query, this is achieved by tracking features to identify those that are most stable with respect to view point changes and those that are not attached to moving objects. Further, virtual feature panoramas, composed of multiple frames to extend the limited view of a regular camera, are used as a query. The statistics of the database can be exploited to identify those features that provide most information about their location. This allows us to reduce the size of the database by 80%. An online process is proposed to update the database and to exclude features attached to dynamic objects by integrating the information provided by the user recordings.

To increase the retrieval performance and to integrate coarse prior knowledge on the user location, we segment the search area, i.e., a large environment such as a city, into several overlapping subregions. Similar to techniques such as locality-sensitive hashing [32] or randomized kd-trees [28], this leads to redundant retrieval results, which are subsequently fused.

Because of the rapidly changing field of view and constantly changing user attention, a close to real-time location retrieval is essential for LBSs to be perceived as useful. With the data rate of current 3G networks, a major challenge of mobile visual location recognition is the transmission of data between the client and server. We propose an approach that eliminates the network delay by exploiting the typically about five times faster downlink to periodically preload the vocabulary required to navigate within a limited area. Only the relevant information is downloaded as database features are selected with respect to their information gain with respect to the overall database statistics.

Several challenges remain to facilitate robust and reliable mobile visual location recognition. With current robust feature detectors such as MSER [11], a major part of the available computational power of the mobile device has to be spent on the feature extraction stage. Hence, low complexity feature extraction and tracking pipelines are needed. As more applications based on features arise, a hardware integration of a feature extractor on the device could be a possible salvation. Further, a more distinctive description of the environment is required, especially in less feature-rich indoor scenarios. Here, a fast detection [35] and recognition [36] of textual information such as those on doors, signs, stores, and posters would provide additional visual information on the location.

With an increasing amount of information extracted from the recordings at low complexity, probabilistic temporal filtering, and fusion with other sensors such as WiFi and IMU, ubiquitous location recognition and mapping becomes feasible in mass market applications.

ACKNOWLEDGMENTS

This research project has been supported in part by the space agency of the German Aerospace Center with funds from the Federal Ministry of Economics and Technology on the basis of a resolution of the German Bundestag under the reference 50NA1107. We thank Google for generously providing a Street View data set with high spatial resolution that was used to produce the results in Figures S1 and S2.

AUTHORS

Georg Schroth (schroth@tum.de) holds a B.Sc. degree (2007) and a Dipl-Ing. Univ. (M.Sc.) degree (2008) in electrical engineering and information technology from Technische Universität München (TUM). He graduated with an honors degree in the master-level graduate program in technology management of the Center for Digital Technology and Management (CDTM), which is part of the Elite Network Bavaria. As a graduate visiting researcher, he joined the Global Positioning System Laboratory in 2007 and the Information Systems Laboratory in 2010 at Stanford University. He is a Ph.D. candidate working at the Institute for Media Technology at TUM. His research focuses on vision-based localization methods.

Robert Huitl (huitl@tum.de) studied information technology at TUM and received his Dipl.-Ing. (Univ.) degree in 2010. He has published numerous articles in print and online magazines. He is currently pursuing his Ph.D. degree at the Institute for Media Technology as a member of the research and teaching staff. His research interests include computer vision and image processing, in particular methods for visual localization and navigation.

David Chen (dmchen@stanford.edu) received his M.S. and B.S. degrees in electrical engineering from Stanford University. He is a Ph.D. student in the Department of Electrical Engineering at Stanford University. His research focuses on the large-scale content-based image and video retrieval for mobile visual search applications. He received an Outstanding Teaching

Assistant (TA) Award when serving as a TA for the graduate-level digital image processing class taught at Stanford.

Mohammad Abu-Alqumsan (moh.marwan@mytum.de) received his B.Sc. degree in electrical engineering (specialization, communications) from Birzeit University (BZU), Ramallah, Palestine, in 2006, and his M.Sc. degree in communications engineering (specialization, communications systems) from TUM in 2010. He worked as a research and teaching assistant in the Electrical Engineering Department at BZU from 2006 to 2007. Currently, he is working as a research assistant in the Institute of Media Technology at TUM. His current research interests include CBIR and vision-based localization.

Anas Al-Nuaimi (anas.alnuaimi@tum.de) graduated with honors in the M.Sc. program in communications engineering at the TUM in 2009 and holds a B.Sc. degree in electrical and computer engineering from Hashemite University in Jordan as well as an honors degree in technology management from the CDTM. He is a member of the research staff of the Institute for Media Technology at TUM working toward his Dr.-Ing. degree focusing on topics related to computer vision and multisensor fusion.

Eckehard Steinbach (eckehard.steinbach@tum.de) studied electrical engineering at the University of Karlsruhe, Germany, University of Essex, Colchester, United Kingdom, and ESIEE, Paris, France. He received an engineering doctorate from the University of Erlangen-Nürnberg, Germany, in 1999. From 1994 to 2000, he was a member of the research staff of the Image Communication Group, University of Erlangen-Nürnberg. From February 2000 to December 2001, he was a postdoctoral fellow with the Information Systems Laboratory, Stanford University, California. In February 2002, he joined the Department of Electrical Engineering and Information Technology, Technische Universität München, Munich, Germany, as a professor of media technology. His current research interests are in the area of audiovisual-haptic information processing, image and video compression, error-resilient video communication, and networked multimedia systems. He is a Senior Member of the IEEE.

REFERENCES

[1] Foursquare. [Online]. Available: http://foursquare.com/

[2] Google. Google Street View [Online]. Available: http://maps.google.com/streetview

[3] Microsoft. Microsoft Street-Side Views [Online]. Available: http://www.bing.com/maps/

[4] Nokia. 3D indoor maps [Online]. Available: http://onsoftware.en.softonic.com/nokia-set-to-add-3d-indoor-maps-to-ovi

[5] Google. Google Art Project [Online]. Available: http://www.googleartproject.com

[6] R. Datta, D. Joshi, J. Li, and J. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv. (CSUR)*, vol. 40, no. 2, pp. 1–60, 2008.

[7] B. Girod, V. Chandrasekhar, D. M. Chen, N. M. Cheung, R. Grzeszczuk, Y. Reznik, G. Takacs, S. S. Tsai, and R. Vedantham, "Mobile visual search," *IEEE Signal Processing Mag.* (*Special Issue on Mobile Media Search*), vol. 28, no. 4, pp. 61–76, 2011.

[8] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Jan. 2004.

[9] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision – ECCV 2006* (Lecture Notes in Computer Science), A. Leonardis, H. Bischof, and A. Pinz Axel, Eds. Berlin: Springer-Verlag, May 2006, pp. 404–417.

[10] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, Jan. 2004.

[11] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vis. Comput.*, vol. 22, no. 10, pp. 761–767, Sept. 2004.

[12] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, vol. 65, no. 1, pp. 43–72, Nov. 2005.

[13] D. Nistér and H. Stewénius, "Linear time maximally stable extremal regions," in *Computer Vision - ECCV 2008* (Lecture Notes in Computer Science, vol. 5303), D. Forsyth, P. Torr, and Zisserman, Eds. Berlin: Springer-Verlag, Oct. 2008, pp. 183–196.

[14] D. Ta, W. Chen, N. Gelfand, and K. Pulli, "SURFTrac: Efficient tracking and continuous object recognition using local feature descriptors," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Miami, June 2009, pp. 2937–2944.

[15] M. Donoser and H. Bischof, "Efficient maximally stable extremal region (MSER) tracking," in *Proc. IEEE Int. Conf. Compter Vision and Pattern Recognition*, New York, June 2006, pp. 553–560.

[16] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[17] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod, "CHoG: Compressed histogram of gradients a low bit-rate feature descriptor," in *Proc. IEEE Int. Conf. Compter Vision and Pattern Recognition*, Miami, FL, June 2009, pp. 2504–2511.

[18] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Compter Vision*, Nice, France, Oct. 2003, pp. 1470–1477.

[19] I. Witten, A. Moffat, and T. Bell, *Managing Gigabytes: Compressing and Indexing Documents and Images*. San Mateo, CA: Morgan Kaufmann, 1999.

[20] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, New York, June 2006, pp. 2161–2168.

[21] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Minneapolis, June 2007, pp. 1–8.

[22] G. Schindler, M. Brown, and R. Szeliski, "City-scale location recognition," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Minneapolis, MN, June 2007, pp. 1–7.

[23] H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *Int. J. Comput. Vis.*, vol. 87, no. 3, pp. 316–336, Feb. 2010.

[24] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proc. IEEE Int. Conf. Computer and Vision Pattern Recognition*, Anchorage, AK, June 2008, pp. 1–8.

[25] J. van Gemert, C. Veenman, A. Smeulders, and J. Geusebroek, "Visual word ambiguity," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 32, no. 7, pp. 1271–1283, July 2010.

[26] G. Schroth, A. Al-Nuaimi, R. Huitl, F. Schweiger, and E. Steinbach, "Rapid image retrieval for mobile location recognition," in *Proc. IEEE Conf. Acoustics, Speech and Signal Processing*, Prague, May 2011.

[27] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, San Francisco, CA, June 2010, pp. 3304–3311.

[28] C. Silpa-Anan and R. Hartley, "Optimised KD-trees for fast image descriptor matching," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Anchorage, AK, June 2008, pp. 1–8.

[29] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling features for large scale partial-duplicate web image search," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Miami, FL, June 2009, pp. 25–32.

[30] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, New York, June 2006, pp. 2169–2178.

[31] S. Ober, M. Winter, C. Arth, and H. Bischof, "Dual-layer visual vocabulary tree hypotheses for object recognition," in *Proc. IEEE Int. Conf. Image Processing*, San Antonio, TX, Oct. 2007, pp. 345–348.

[32] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proc. Symp. Computational Geometry*, New York, June 2004, pp. 253–262.

[33] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, J. Singh, and B. Girod, "Tree histogram coding fpor mobile image matching," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2009, pp. 143–152.

[34] G. Takacs, V. Chandrasekhar, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod, "Unified real-time tracking and recognition with rotation-invariant fast features," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, San Francisco, June 2010, pp. 934–941.

[35] H. Chen, S. Tsai, G. Schroth, D. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," in *Proc. IEEE Int. Conf. Image Processing*, Brussels, Sept. 2011.

[36] S. S. Tsai, H. Chen, D. M. Chen, G. Schroth, R. Grzeszczuk, and B. Girod, "Mobile visual search on papers using text and low bit-rate features," in *Proc. IEEE Int. Conf. Image Processing*, Brussels, Sept. 2011. [SP]