Orientation Data Correction with Georeferenced Mobile Videos

Guanfeng Wang[†], Yifang Yin[†], Beomjoo Seo[§], Roger Zimmermann[†], Zhijie Shen[‡]
[†]School of Computing, National University of Singapore
[§]School of Games, Hongik University [‡]Hortonworks Inc.
[†]{wanggf,yifang,rogerz}@comp.nus.edu.sg
[§]bseo@hongik.ac.kr [‡]zshen@hortonworks.com

ABSTRACT

Similar to positioning data, camera orientation information has become a powerful contextual feature utilized by a number of GIS and social media applications. Such auxiliary information facilitates higher-level semantic analysis and management of video assets in such applications, *e.g.*, video summarization and video indexing systems. However, it is problematic that raw sensor data collected from current mobile devices is often not accurate enough for subsequent geospatial analysis. To date, an effective orientation data correction system for mobile video content has been lacking. Here we present a content-based approach that improves the accuracy of noisy orientation sensor measurements generated by mobile devices in conjunction with video acquisition. Our preliminary experimental results demonstrate significant accuracy enhancements which benefit upstream sensor-aided GIS applications to access video content more precisely.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Spatial databases and GIS; I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis - Sensor Fusion

Keywords

Orientation sensors, georeferenced mobile video, data correction, digital compass

1. INTRODUCTION

The multimedia content generated from smartphones and tablets has become one of the primary contributors to the media-rich web and its underlying databases. The top three most popular cameras in the Flickr community are smartphone models[1]. Meanwhile, from the significant number of sensors integrated into these devices, an increasing amount of geospatial sensor information in conjunction with still images and video frames is available for both research and commercial utilization. As an example, the convenient acquisition of time-series data from digital compasses integrated in mobile devices has enabled camera orientation data, in addition to traditional position measurements, to become important contextual information.

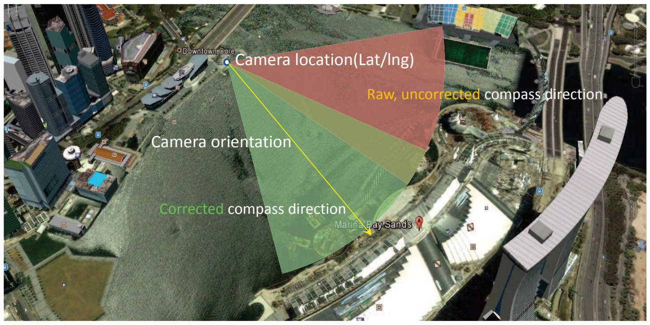


Figure 1: Example of a comparison of inaccurate, raw camera orientation data (red) with the ground truth (green).

Recently an increasing number of GIS and social media applications utilize diverse auxiliary sensor information as complementary features to improve multimedia content analysis performance. Such surrounding meta-data provides contextual descriptions at a semantically interesting level and enables ingenious and efficient management of mobile videos [10]. The scenes captured in images or videos can be characterized by a sequence of camera position and orientation data. These geographically described (*i.e.*, georeferenced) media data contain significant information about the region where they were captured and can be effectively processed in various GIS applications, *e.g.*, for visual navigation and geospace queries. Both camera position and orientation sensor data are also employed by various GIS and social media applications such as street navigation systems [5], photo organization and management [3], video indexing and tagging [6, 13], video summarizations [17, 4], video encoding complexity reductions [2, 15], and others.

However, unfortunately most geospatial sensor information (including positioning and orientation data) collected from current phones or tablets is not highly accurate due to varying surrounding environmental conditions during data acquisition, and the use of consumer-grade sensors. For GPS this issue is well-known in the research community, and thus a number of approaches have been proposed for data correction. In contrast, the accuracy of orientation data acquired from digital compasses, which is also increasingly used in many applications, has not been studied extensively. We

[1] http://www.flickr.com/cameras

SIGSPATIAL'13 Nov 05-08 2013, Orlando, FL, USA
ACM 978-1-4503-2521-9/13/11.
http://dx.doi.org/10.1145/2525314.2525445.

found that orientation data is also in need of correction prior to upstream application use. As exemplified in Figure 1, the red pie-shaped slice represents the raw, uncorrected orientation measurement while the green slice indicates the corrected data. Here we use the pie-shaped slice to present one orientation value instead of one single vector since the pie-shaped slice indicates the field-of-view (FOV) of one video frame. We detail this part in Section 2. As illustrated, the user is recording the tall Marina Bay Sands hotel structure towards the southeast direction, while the direct, raw sensor measurement from the mobile device indicates an east direction and hence may later lead to a completely incorrect scene expectation of a bridge (the Helix Bridge). We found in our real world measurements that in some cases the discrepancy is more than 90 degrees from the ground-truth value. Moreover, we believe that the research studies and applications employing geospatial analysis of sensor data outlined earlier are mostly not error resilient with respect to incorrect sensor data input. Consequently, one important and urgent requirement to facilitate upstream research activities and applications is the availability of well-corrected orientation data.

To aid in this effort we introduce a framework which corrects orientation data measured in conjunction with mobile videos based on image processing techniques. Our system assumes the roughly true orientation value of the first frame in a georeferenced video can be estimated and provided by users manually. We subsequently propagate the manually estimated orientation value from one specific frame to the rest of the video. By leveraging the content consistency in the temporal domain, we compute the horizontal motion flows to interpolate accurate orientation data for every frame.

2. SENSOR DATA MODEL

The target sensor data of our correction approach is a series of contextual camera orientation descriptions of mobile video content based on the geospatial properties of the scenes it captures. Here we provide a brief summary of background information on our georeferenced video annotation model that collects and manages sensor data in conjunction with video contents during the recoding phase.

To let users conveniently and efficiently acquire sensor-annotated videos, we have made two custom recording apps publicly available for the Android and iOS platforms, namely the *GeoVid* apps [16]. When a user begins to capture a video, the GPS and compass sensors are turned on to *continuously* record location and orientation information of the (moving) camera. All collected sensor data (*i.e.*, camera location and orientation, the corresponding frame timecode and video ID) are combined into a JSON format and uploaded to a portal, to which users can also submit various spatial and contextual queries, browse, and retrieve the videos with their sensor data via web APIs[2].

We adapt the field-of-view (FOV, also called the *viewable scene*) model introduced by Arslan Ay *et al.* [1]. An FOV describes a scene area captured by a camera positioned at a given location. The description of a camera's viewable scene consists of three parameters: the camera location $\mathcal{L}$, the camera orientation θ, and the viewable angle α ($FOV \equiv \langle \mathcal{L}, \theta, \alpha \rangle$). The camera position $\mathcal{L}$ is composed of latitude and longitude coordinates provided by a positioning device (*e.g.*, GPS receiver) and the camera orientation θ is obtained based on the direction angle value from a digital compass. The viewable angle α is calculated based on the camera and lens properties at the current zoom level. Note, each mobile device model may use different sampling frequencies for different sensors. Ideally we acquire one *FOV triplet per frame*. If that is not feasible and the granularity is coarser due to the device limits, we perform linear interpolation to generate triplets for every frame.

3. ORIENTATION DATA CORRECTION

To correct the noise, we propagate the manually estimated orientation value from one specific frame to the whole video based on optical flow estimation. We begin by describing the problem formally.

3.1 Problem Formulation

In our context, orientation data of a mobile video is a time-series dataset consisting of compass reading values. Let $\Theta = \{\theta_1, \theta_2, \cdots, \theta_n\}$ and $\mathcal{F} = \{f_1, f_2, \cdots, f_n\}$ be the sequences of compass readings and their corresponding video frames for every time instance $\mathcal{T} = \{t_1, t_2, \cdots, t_n\}$, respectively. The value θ_i is measured by how many degrees a northward unit vector needs to rotate to current vector clockwise in 2D geospace. For example, if the camera is facing due east, then $\theta_i = 90$. The tilting operation of the camera is not covered in this study. We plan to further elaborate on 6 degrees of freedom (DOF) camera pose correction techniques in 3D geospace as part of our future work. We denote the ground truth of the orientation sequence data as $\mathcal{G} = \{g_1, g_2, \cdots, g_n\}$. Both g_i and θ_i have values in the range 0 to 360 degrees. The direction measurement error for θ_i is the angle difference between its true and measured orientation $\delta_i = min(\|g_i - \theta_i\|, 360 - \|g_i - \theta_i\|)$. The direction error of Θ is the average of every sample's direction error, *i.e.*, $E_\Theta = \frac{1}{n}\sum_{i=1}^{n}\delta_i$. The words *orientation* and *direction* are used interchangeably in our study.

Problem Statement. Given a sequence of orientation readings Θ and their related timestamps and frames $\mathcal{T}$ and $\mathcal{F}$, find a sequence of estimated directional values, $F : \theta_i \to \tau_i$, such that the accuracy of processed orientation sequence $\Theta' = \{\tau_1, \tau_2, ..., \tau_n\}$ is enhanced by having $E_\Theta \nleq E_{\Theta'}$, where $E_{\Theta'} = \frac{1}{n}\sum_{i=1}^{n}\delta'_i$ and δ'_i is each processed orientation's distance to the ground truth.

3.2 Landmark Tracking

Since the estimated directional value of a video's first frame is provided manually, $\theta_1 \to \tau_1$ at time $\mathcal{T} = \{t_1\}$, our system subsequently tracks the interesting feature points detected around the target landmark to continuously calculate the position of this building in the next several seconds of frames. The position of the target landmark in the image is also marked by users manually. Afterward we use an affine model to estimate the target landmark's 2D transformation in the image and extract motion vector information on the horizontal axis, termed $x_i \cdots x_j$, to compute camera orientation values $\tau_i \cdots \tau_j$ for the following portion of frames $f_i \cdots f_j$. Since the time duration of each visual feature tracking is relatively short (we perform one tracking process between every two GPS signal updates), it is reasonable to assume that the camera location does not move

[2] http://api.geovid.org

too much and the camera is approximately performing a panning operation during that short period. Thus, we can estimate the orientation values by the equation below,

$$\tau_i = \frac{-x_i}{R_h}\alpha + \tau_{i-t} \tag{1}$$

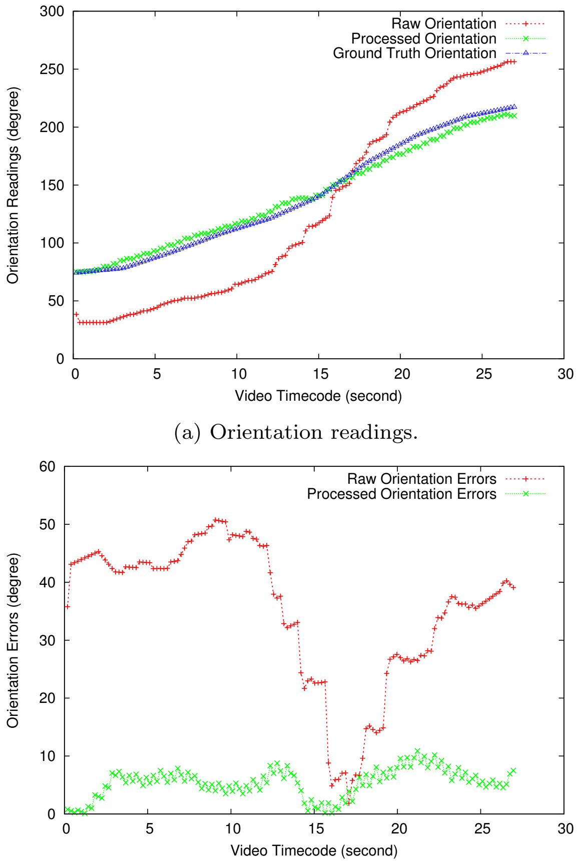
where R_h is the resolution value on either the $x-$ or the $y-$axis, depending on the recoding pose which could be *Portrait* or *Landscape* mode, and α is the viewable angle we introduced in Section 2. We calculate the relative camera rotation first and add it to the previous estimated directional value to obtain the current frame's camera orientation. In the convention of motion vector notation, if the reference object moves towards the right in the image, the motion vector should be positive. Similarly, in the orientation notation, this case indicates that the camera is panning left (rotating counter-clockwise from an aerial view), which generates a negative value accordingly. That is the reason we change the sign (-1) of the horizontal motion vector. Due to performance concerns, we do not compute the motion vector between every two consecutive frames. Instead, we perform such tracking computation every t frames.

We use the Kanade-Lucas-Tomasi Feature Tracker [14] to infer motion vector x_i between f_i and f_{i-t}. Our system chooses and locates features by examining the minimum eigenvalue of each 2×2 gradient matrix, and features are tracked using a Newton-Raphson method of minimizing the difference between the two windows. An affine transformation is fitted between the image of the currently tracked feature f_i and its image from a non-consecutive previous frame f_{i-t}. If the affine compensated image is too dissimilar, the previously extracted features are dropped and new qualified features are selected based on the same algorithm for substitution. Therefore in each motion vector calculation, we maintain a consistent number of tracking feature points FN through abandonment and replacement operations. Our implementation uses values of $t = 15$ and $FN = 150$ by considering both image size and performance.

By aggregating x_i at the landmark's position indicated by the users in the beginning, we are able to know whether the landmark still appeares within the image (*i.e.*, within the FOV geographically). When we detect that the target landmark is moving out of the viewable scene, our system changes to track the feature points detected from the whole frame and to extract motion vectors based on these extended features. Lastly, all camera orientation values between τ_i and τ_{i-t} are estimated by linear interpolation. Our system continues such operations until the end of the video sensor file.

4. EXPERIMENTAL EVALUATION

In our experiments, we utilize the publicly available real-world georeferenced video dataset from the GeoVid website[3], process the corresponding sensor data of the videos with our proposed method and compare the results with the ground truth in terms of the accuracy enhancement. Since the manual ground truth annotation of the orientation value is extremely time consuming, in this paper we apply our approach on only one georeferenced video with erroneous directional sensor data, which consists of 131 raw orientation values and 870 frames in total. We plan to conduct more experiments on other mobile videos in the future.



(a) Orientation readings.

(b) Camera orientation error.

Figure 2: (a) Raw, processed and ground truth camera orientation reading results of one sensor data file (θ_i, τ_i, and g_i). (b) Raw and processed camera orientation error of each sample in one sensor data file (δ_i and δ'_i).

To obtain the ground truth data we provide two alternative ways for users to manually annotate true camera orientation values. For a given video frame, we first provide multiple Google Street View images and a Google Earth 3D synthesized view from the current GPS location. Users can compare the visual contents between the frame and the referenced views to determine the orientation value. In addition, we also allow users to indicate the geographic object that appears in the frame center on the Google Earth interface. The coordinates of the indicated object as well as the camera location are later entered into the Geotools library[4] to calculate the true camera orientation. For each experimental video, we sample frames every 3 seconds for users to perform the ground truth annotation. We interpolate the orientation degrees between sampled frames for later comparisons.

The first video frame of each video clip is treated as a still image and ask users to manually estimate a likely direction for this image via the above method. To the following frames, we apply our landmark tracking approach. As will be shown in our experimental results, the system works well for georeferenced mobile videos. We employ the *FFmpeg*

[3] http://api.geovid.org/v1.0/web/viewer

[4] http://geotools-php.org

library[5] to extract frames from the video dataset at a chosen resolution of 360×240 per frame to reduce the time consumption for image processing.

In Figure 2, we present the results of one camera orientation sequence, which consists of 131 camera orientation values. Figure 2a illustrates θ_i, τ_i, and g_i of the tested mobile video and Figure 2b illustrates the errors δ_i and δ'_i accordingly. As shown, the raw orientation readings along the whole file are very much incorrect, which is causing a drift phenomenon when displayed on a map interface. After the correction with our algorithm, we find a distinct improvement such that the processed orientation data approaches the ground truth values and the error of each sample is considerably reduced.

5. RELATED WORK

Researchers have leveraged various content-based computer vision techniques to estimate the viewing direction of photos. *E.g.*, they geo-locate a photo and then estimate the camera orientation by registering the image onto street level panoramas [7], or Google Street View and Google Earth Map [11]. Luo *et al.* utilize a SIFT Flow to match a photo in a database followed by image geometry calculation, to determine and filter the viewing direction [9]. However, these methods can be applied only to individual photos. They all require either a constrained camera location (photos taken on or near a road network) or a relatively large image database to perform the matching phase.

Recently, the Structure from Motion (SfM) technique has been extensively exploited to reconstruct 3D models from a collection of images [12, 8]. The images are later registered to 3D scenes by feature point matching and the camera pose (including location and orientation) of each image is estimated by image geometry calculation. Since these algorithms were not devised for a dedicated sensor data correction purpose, they ignore all contextual geo-information. As a result, the preliminary dataset requirements and processing time make these methods unsuitable for large-scale camera orientation correction. In our framework, we only process the frames in one single video without requiring any third party image database.

6. CONCLUSIONS

We presented an approach for camera orientation data correction based on the image processing techniques. We applied this method to estimate more precise orientation data for georeferenced videos. The preliminary experimental results demonstrate that our technique is reasonably effective compared with the ground truth. One limitation of our work is that the initial estimated orientation value of the first frame has to be provided manually, for our system to propagate the corrected values across all the video frames. As part of our future work we plan to investigate other visual features and sensors embedded in mobile platforms to help with camera orientation correction without extra user input.

Acknowledgment

This research has been supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

7. REFERENCES

[1] S. Arslan Ay, R. Zimmermann, and S. H. Kim. Viewable Scene Modeling for Geospatial Video Search. In *16th ACM Multimedia*, pages 309–318, 2008.

[2] X. Chen, Z. Zhao, A. Rahmati, Y. Wang, and L. Zhong. SaVE: Sensor-assisted Motion Estimation for Efficient H.264/AVC Video Encoding. In *17th ACM Multimedia*, pages 381–390, 2009.

[3] B. Epshtein, E. Ofek, Y. Wexler, and P. Zhang. Hierarchical Photo Organization Using Geo-relevance. In *15th International Conference on Advances in Geographic Information Systems*, 2007.

[4] J. Hao, G. Wang, B. Seo, and R. Zimmermann. Keyframe Presentation for Browsing of User-generated Videos on Map Interfaces. In *19th ACM Multimedia*, pages 1013–1016, 2011.

[5] L. Kaminski, R. Kowalik, Z. Lubniewski, and A. Stepnowski. "VOICE MAPS" - Portable, Dedicated GIS for Supporting the Street Navigation and Self-dependent Movement of the Blind. In *2nd International Conference on Information Technology*, pages 153–156, 2010.

[6] S. H. Kim, S. Arslan Ay, and R. Zimmermann. Design and Implementation of Geo-tagged Video Search Framework. *Journal of Visual Communication and Image Representation*, pages 773–786, 2010.

[7] M. Kroepfl, Y. Wexler, and E. Ofek. Efficiently Locating Photographs in Many Panoramas. In *18th International Conference on Advances in Geographic Information Systems*, pages 119–128, 2010.

[8] H. Liu, T. Mei, J. Luo, H. Li, and S. Li. Finding Perfect Rendezvous on the Go: Accurate Mobile Visual Localization and Its Applications to Routing. In *20th ACM Multimedia*, pages 9–18, 2012.

[9] Z. Luo, H. Li, J. Tang, R. Hong, and T.-S. Chua. ViewFocus: Explore Places of Interests on Google Maps Using Photos with View Direction Filtering. In *17th ACM Multimedia*, pages 963–964, 2009.

[10] H. Ma, R. Zimmermann, and S. H. Kim. HUGVid: Handling, Indexing and Querying of Uncertain Geo-tagged Videos. In *20th International Conference on Advances in Geographic Information Systems*, pages 319–328, 2012.

[11] M. Park, J. Luo, R. T. Collins, and Y. Liu. Beyond GPS: Determining the Camera Viewing Direction of a Geotagged Image. In *18th ACM Multimedia*, pages 631–634, 2010.

[12] T. Sattler, B. Leibe, and L. Kobbelt. Fast Image-based Localization Using Direct 2D-to-3D Matching. In *IEEE International Conference on Computer Vision*, pages 667–674, 2011.

[13] Z. Shen, S. Arslan Ay, S. H. Kim, and R. Zimmermann. Automatic Tag Generation and Ranking for Sensor-rich Outdoor Videos. In *19th ACM Multimedia*, pages 93–102, 2011.

[14] J. Shi and C. Tomasi. Good Features to Track. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[15] G. Wang, H. Ma, B. Seo, and R. Zimmermann. Sensor-assisted Camera Motion Analysis and Motion Estimation Improvement for H.264/AVC Video Encoding. In *22nd ACM NOSSDAV workshop*, pages 89–94, 2012.

[16] G. Wang, B. Seo, and R. Zimmermann. Motch: an Automatic Motion Type Characterization System for Sensor-rich Videos. In *20th ACM Multimedia*, pages 1319–1320, 2012.

[17] Y. Zhang, G. Wang, B. Seo, and R. Zimmermann. Multi-video Summary and Skim Generation of Sensor-rich Videos in Geo-space. In *3rd Multimedia Systems Conference*, pages 53–64, 2012.

[5] http://ffmpeg.org/