E^{2}C^2: Efficient and Effective Camera Calibration in Indoor Environments

†Huan Li, †Pai Peng, ‡Hua Lu, †Lidan Shou, †Ke Chen and †Gang Chen

†College of Computer Science and Technology, Zhejiang University, China

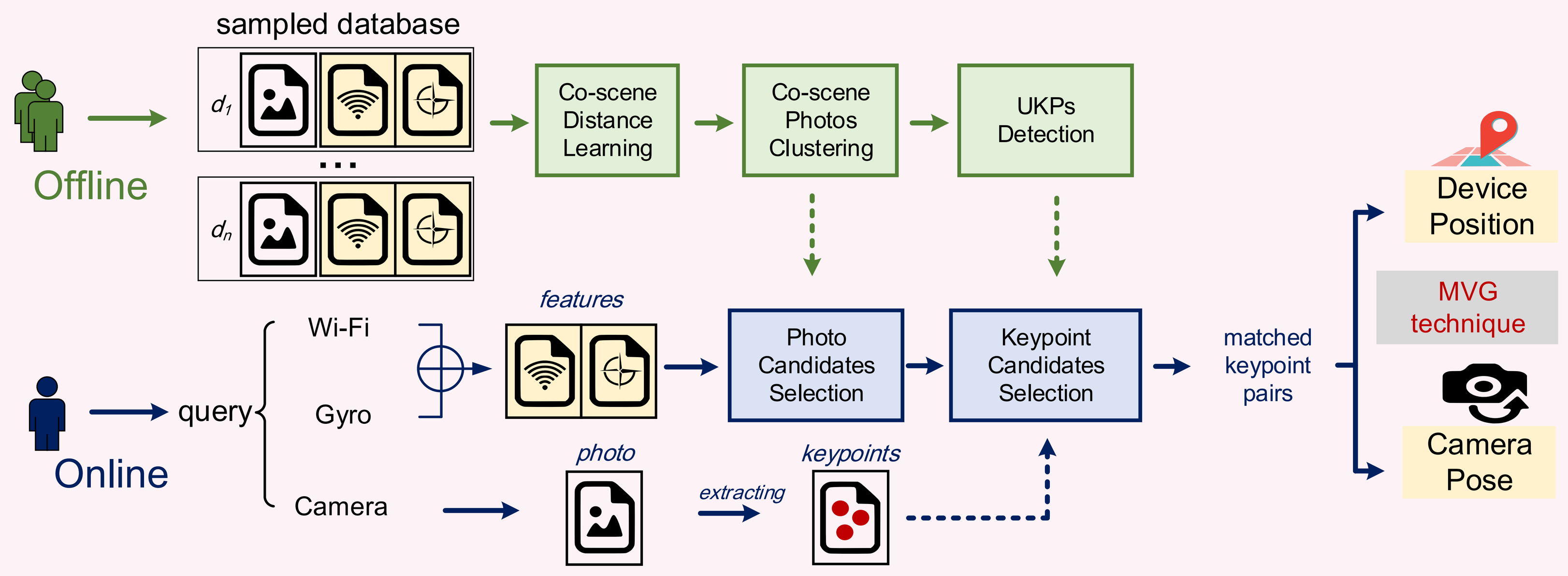
‡Department of Computer Science, Aalborg University, Denmark

Introduction

- Mobile AR applications help people better interact with complex indoor environments, e.g., navigation in a museum.
- **Camera Calibration** requires sufficient keypoints distributed in other photos matched with the current camera view.
- Pairwise visual matching is computationally expensive.
- We are motivated to take advantage of embedded rich sensors as a way of pruning candidate photos to accelerate the calibration.

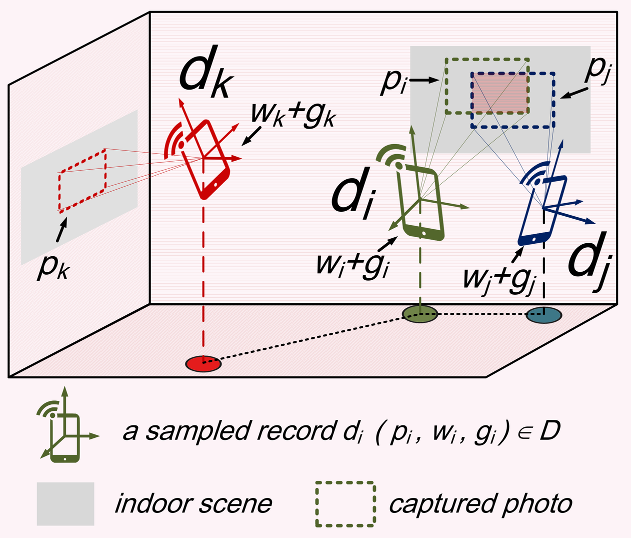
Framework Overview

- Our framework only needs a few sampled photos beforehand labeled with **Wi-Fi+gyro** information and can quickly select a small but sufficient set of SIFT points for calibration.



- **In the offline phase:**
  - capture a set of photos along with associated Wi-Fi and gyro sensor data.
  - cluster the captured photos into co-scenes according to a learned distance metric.
  - detect a subset of keypoints that are frequently appeared in each photo cluster.
- **In the online phase:**
  - select a set of photos in the nearest co-scene as photo candidates.
  - obtain several keypoint matches by only comparing with a few “useful keypoints”.
  - infer the extrinsic parameters (camera pose & position) with Multiple View Geometry(MVG) technique.

– Data Structure



- A database D, each record $d \in D$ has three fields:
  - a phone-captured photo p.
  - a set of consecutive Wi-Fi signals w.
  - device’s gyro information g.
- A constraint set C, each entry $c(d_i^{(1)}, d_i^{(2)}, y_i)$ determines if $d_i^{(1)}$ and $d_i^{(2)}$ are similar or not ($y_i \in \{-1, +1\}$). C is small-sized.

Offline Phase

- Learn a novel distance metric in order to bridge the gap between the query and the database.
- Cluster photos into distinctive clusters (each forms a *co-scene*).
- Find out “useful keypoints”(UKPs) located in each co-scene.

– Co-scene Distance Learning

- Create a synthetic feature vector ϑ by a linear combination of Wi-Fi+gyro

$$\vartheta = (w, \lambda g) = (w_1, ..., w_n, \lambda\alpha, \lambda\beta, \lambda\gamma, \lambda\omega) \quad (1)$$

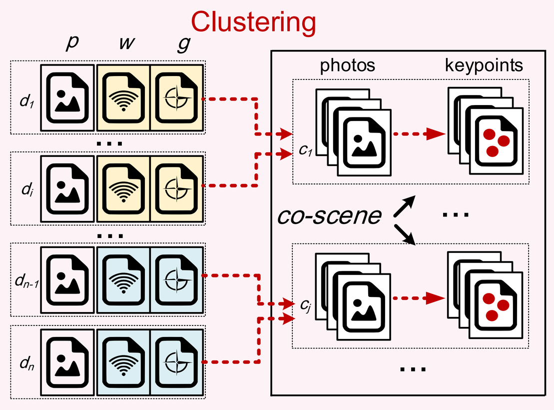
- Define the *co-scene distance* between two photos p_i and p_j by the definition of *Mahalanobis distance* as:

$$D_{cs}(p_i, p_j) = \sqrt{(\vartheta_i - \vartheta_j)^T \ M_\vartheta \ (\vartheta_i - \vartheta_j)} \quad (2)$$

- ITML is effective to optimize the distance, which is to regularize the parameter M_ϑ to be as simple as possible while satisfying the similarity and dissimilarity constraints in the set C.

– Co-scene Photos Clustering

- Cluster the photos in our database into k groups, photos in each cluster form a “co-scene”.
- k is a hyperparameter that needs to be tuned by cross validation.



– UKPs Detection

- Co-scenes help reduce photo candidates, still expensive if we iterate through all keypoints located in co-scenes.
- Detect “useful keypoints”(UKPs):
  - count the frequency of each keypoint by matching photos pairwisely in the same co-scene.
  - only frequently appearing ones are useful for further calibration.
- Consequently, UKPs (red) are selected from the whole local features (blue) by UKPs Detection.

Online Phase

- The online query contains the same structure of a sampled record d in the offline phase.
- Pick out the keypoint pairs between the query and other photos by a two-level selection.

– Photo Candidates Selection

- Compose the synthetic feature ϑ_q for the query according to Equation 1.
- Find the nearest co-scene by comparing ϑ_q with all cluster centroids based on the learned distance metric.
- As a result, all photos in that co-scene are considered as photo candidates.

– Keypoint Candidates Selection

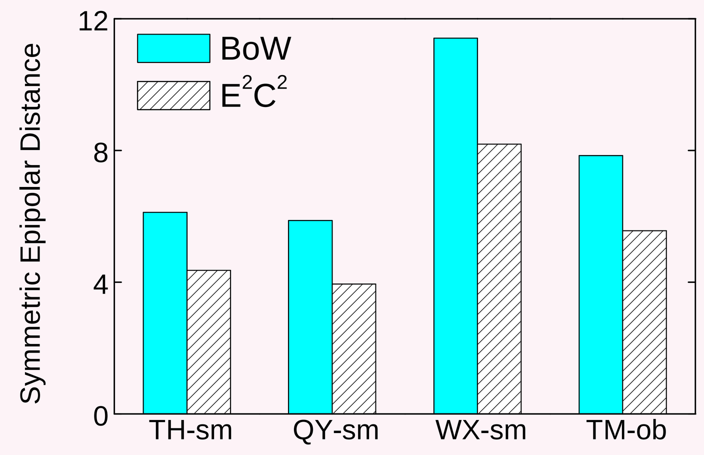
- Only need to compare keypoints in the query with these UKPs to confirm the ultimate matched keypoint pairs.
- The UKPs are only a small portion of all the keypoints, the online matching is accelerated.
- Finally, we use those few good matched keypoint pairs to estimate the query photo’s camera pose as well as the position by a MVG process.

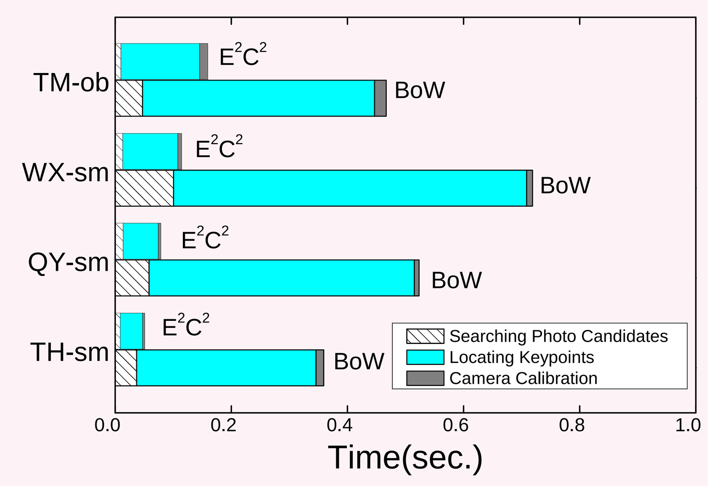
Experiments

dataset	#images	#floors
TH-sm	968	6
QY-sm	1151	6
WX-sm	1674	7
TM-ob	823	3

- An Android app was developed to collect data.
- 4 volunteers, 3 shopping malls and 1 office building.
- They were also required to indicate a few similar and dissimilar pairs to construct the constraint set C.
- 2 other volunteers collected a query set, 50 queries for each place.

– Experimental Studies





- Baseline approach:
  - a state-of-the-art bag-of-visual-words(BoW).
  - cluster all the extracted SIFT points into a dictionary with 5K visual words.
- Effectiveness of our framework:
  - lower values of SED indicate a better calibration accuracy.
  - our proposed approach outperforms BoW in each dataset.
  - the detected UKPs prove to be more robust and discriminative than simple brute-force matching.
- Efficiency of our framework:
  - our framework beats the baseline with a significantly reduced online processing cost.
  - we lower the cost of searching photo candidates since our searching space is reduced to cluster centroids instead of complete photos.
  - the cost of searching keypoints is decreased remarkably due to the UKPs detection.

UbiComp/ISWC 2015 Poster

Contact: lihuancs@zju.edu.cn