A Travel Planning System Based on Travel Trajectories Extracted from a Large Number of Geotagged Photos on the Web

Kohya Okuyama and Keiji Yanai

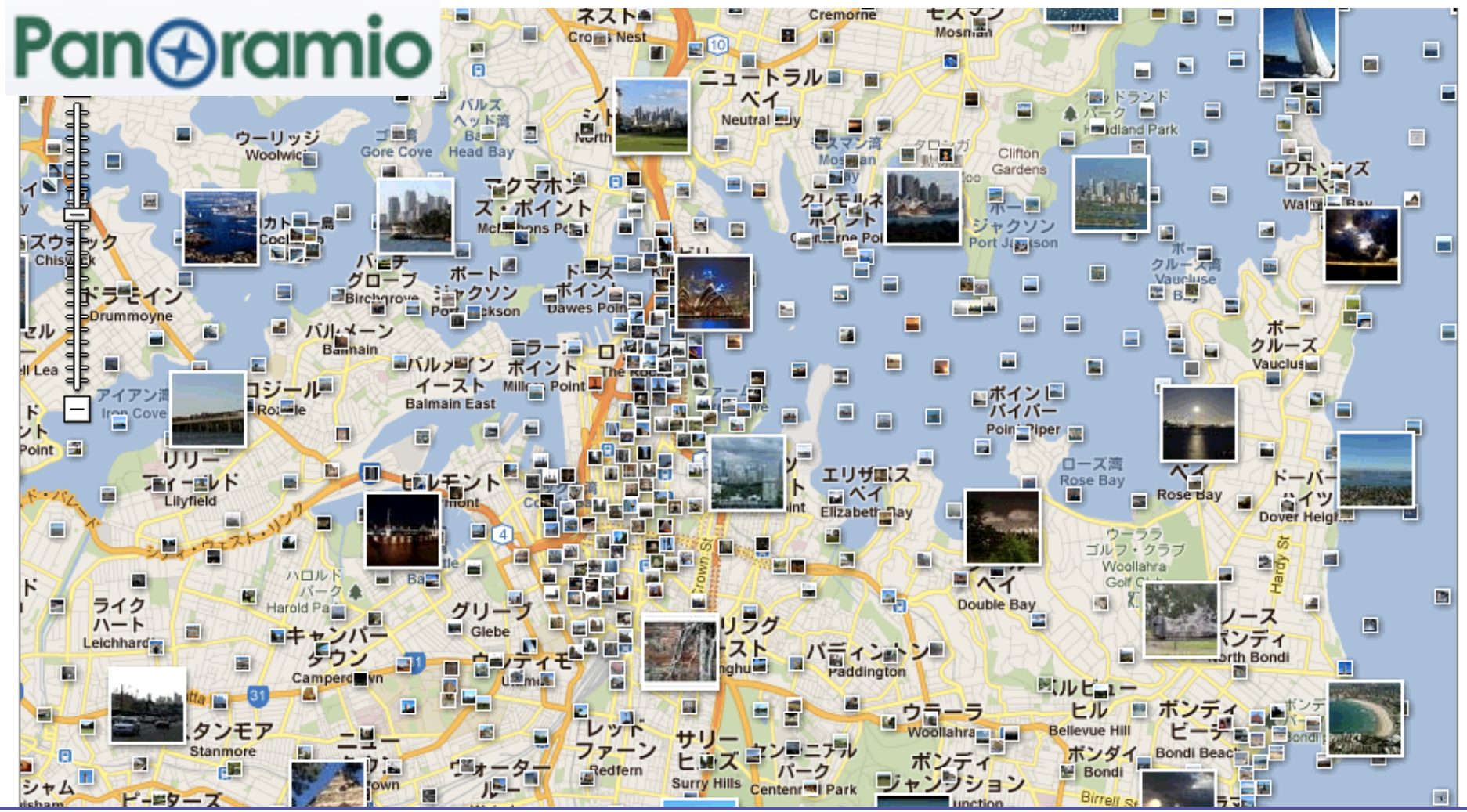
The University of Electro-Communications, Tokyo, Japan

Background

- Geotagging photos became common
  1. Camera with GPS (e.g. iPhone, Android phones)
  2. Photo sharing sites (e.g. Flickr, Panoramio)
  3. Applications showing gettagged photos on the map

so many geotagged Photos on the Web

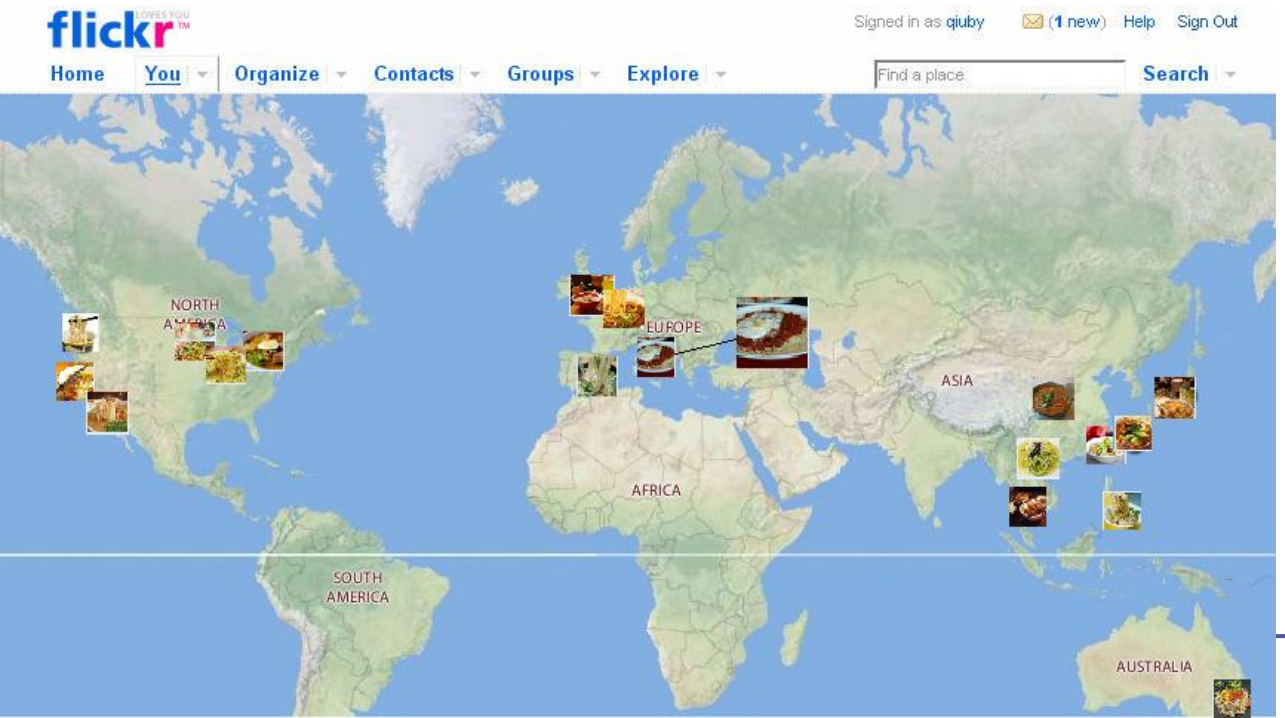
So many geotagged photos !



Background: geotagged photos

- The number of geotagged photos on the Web grows rapidly: Flickr, panoramio
  - Flickr has 100,000,000 geotagged photos.

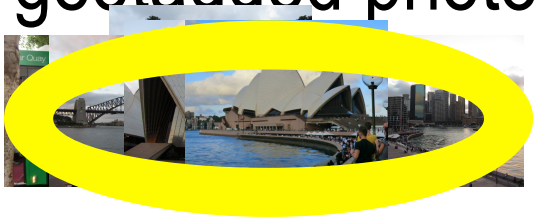
(Feb. 2009)



A "geo-tag" represents the coordinates (latitude,longitude) of a location where a photo are taken.

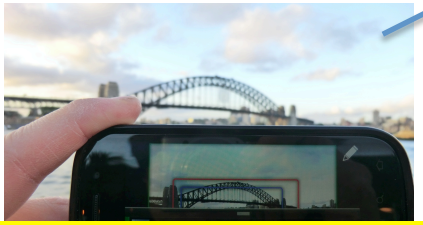
Sequences of geotagged photos = travel paths

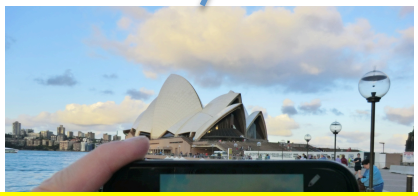
- People using photo sharing Web sites such as Flickr and Panoramio usually upload not just one photos but sequences of geotagged photos.



- By tracing user IDs, we can obtain travel trajectories as well as photo locations.

My travel path (Dec . 20th)





Sequence of geotagged photos ⇒ Travel path

Objective

- We propose a **Travel Planning System** using geotagged photos and travel trajectories extracted from the Web.

The system uses collective knowledge on the Web.

It has two functions:

(1) Travel route recommendation of both places and routes before traveling

(2) Travel reviewing of your travel after traveling

- You should have visited OOO places in addition to OOO. (e.g. botanical garden, Operahouse)

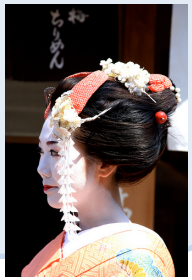
(1) Travel Route Recommendation for using before traveling

User inputs

User selects landmark photos he want to visit

The system outputs

The system shows several routes and popular places along the route



(2) Travel Reviewing for using after traveling

- Automatic organizing travel recoreds
  - Upload geotagged photos after trip
  - (optional) Upload GPS traces
  - Summarize route and show photos on map
  - Show not-visited but popular places.

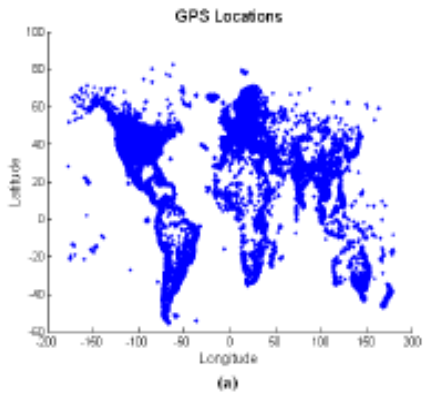
Re-organize

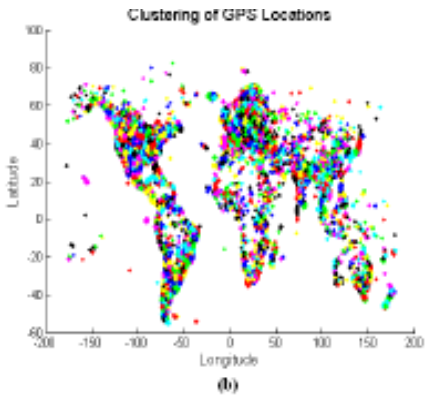
Related work

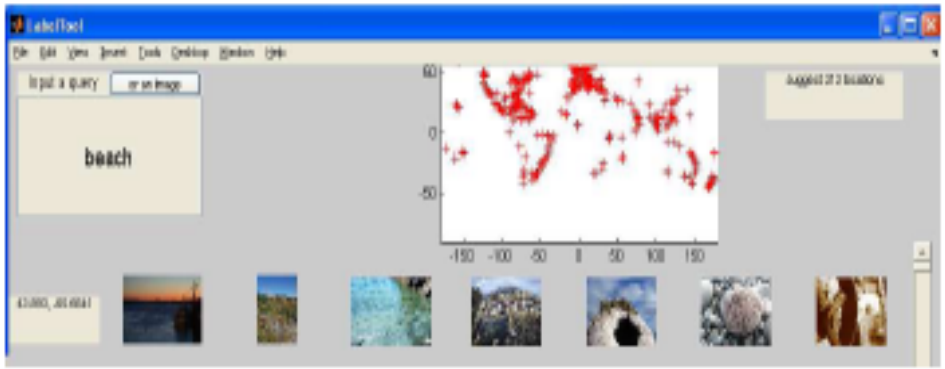
- So many works on geotagged photos
- Focus on works with travel paths
  - Travel route recommendation [X. Lu et al.ACM MM2010]
    - Combine travel paths of several people into travel routes
    - No clustering regarding places
    - No information on landmarks and sightseeing spots
  - Broad area route recommendation [Y. Arase et al.ACM MM2010]
    - No route combination
    - Not within a city, but among cities e.g. Sydney->Brisbane->Caens

Related work: travel places

- Travel place recommendation system
  [L. Cao et al.ACM MIR2010]
  - Recommend travel places
  - No route recommendation





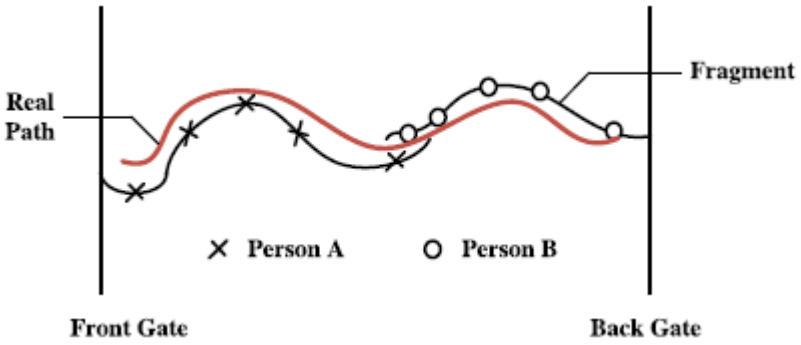


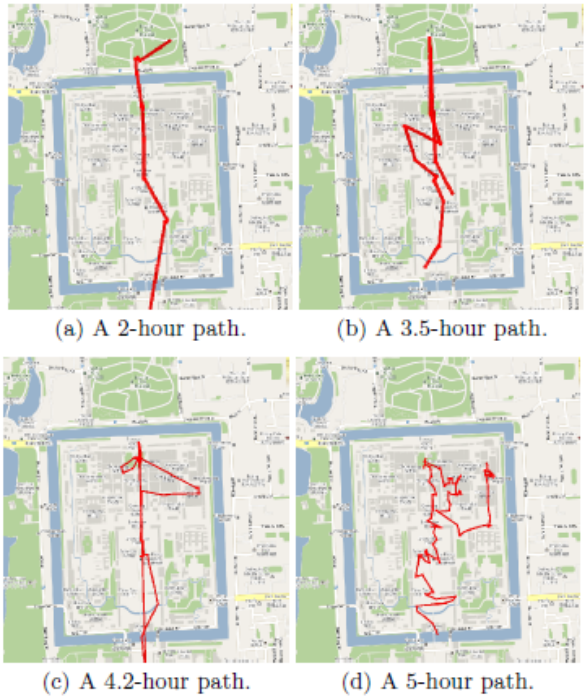
Related work: travel route (1)

- Travel route recommendation

  [X. Lu et al. ACM MM2010]

  – Combine travel paths of several people into travel routes
  – No clustering regarding places
  – No information on landmarks and sightseeing spots

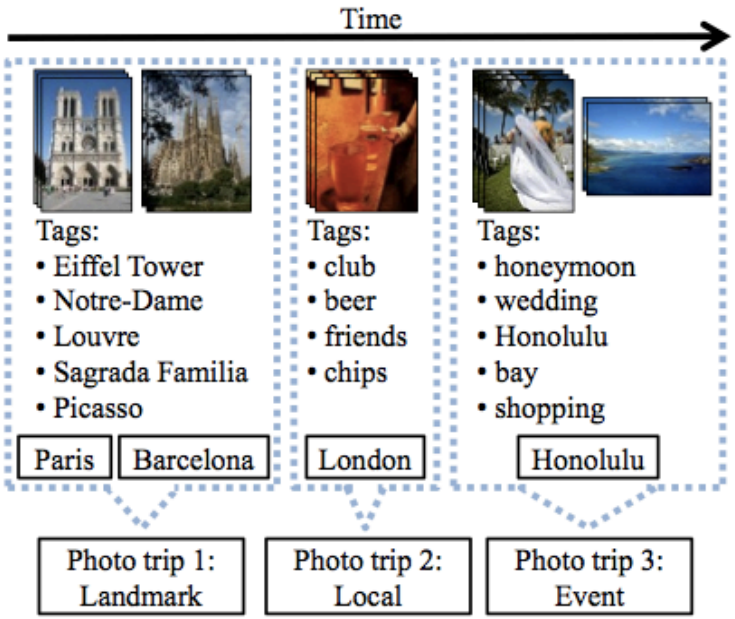



(a) A 2-hour path. (b) A 3.5-hour path.
(c) A 4.2-hour path. (d) A 5-hour path.

Related work: Travel route (2)

- Broad area recommendation [Y. Arase et al.ACM MM2010]
  - Recommend the order of visiting cities and some places within each city
  - Travel plan of wide area

    e.g. Sydney->Brisbane->Cairns



Features of our system

- **Offline** processing : data preparation
  - Detect popular places automatically
  - Extract many travel trajectories and generate many travel path candidates

- Two kinds of **online** systems
  - Travel planning before traveling
    - Recommend both places and routes
  - Travel reviewing after travelling
    - Teach places you should have visited for the next trip
  - (Realtime travel suggestion system for future work)

Overview of the offine processings

Data Gathering
- Gather geotagged photos from Flickr
- Select “good” data

Place Clustering
- Hierarchical clustering
- Detect popular places

Analyzing Place Clusters
- Extract tags for each place
- Select representative photos

Extraction of travel paths
- Extract travel paths
- Generate travel route candidates

Overview of the offine processings

Data Gathering
- Gather geotagged photos from Flickr
- Select “good” data

Place Clustering
- Hierarchical clustering
- Detect popular places

Analyzing Place Clusters
- Extract tags for each place
- Select representative photos

Extraction of travel paths
- Extract travel paths
- Generate travel route candidates

1) Data gathering

- Collect geotagged photos via FlickrAPI
- Select geotagged photos
  - Remove unreliable photos in terms of latitude and longitude
    - Geotagged on the large-scale map without GPS
    - Many photos have exactly the same location
  - Remove single photos (not in sequences)

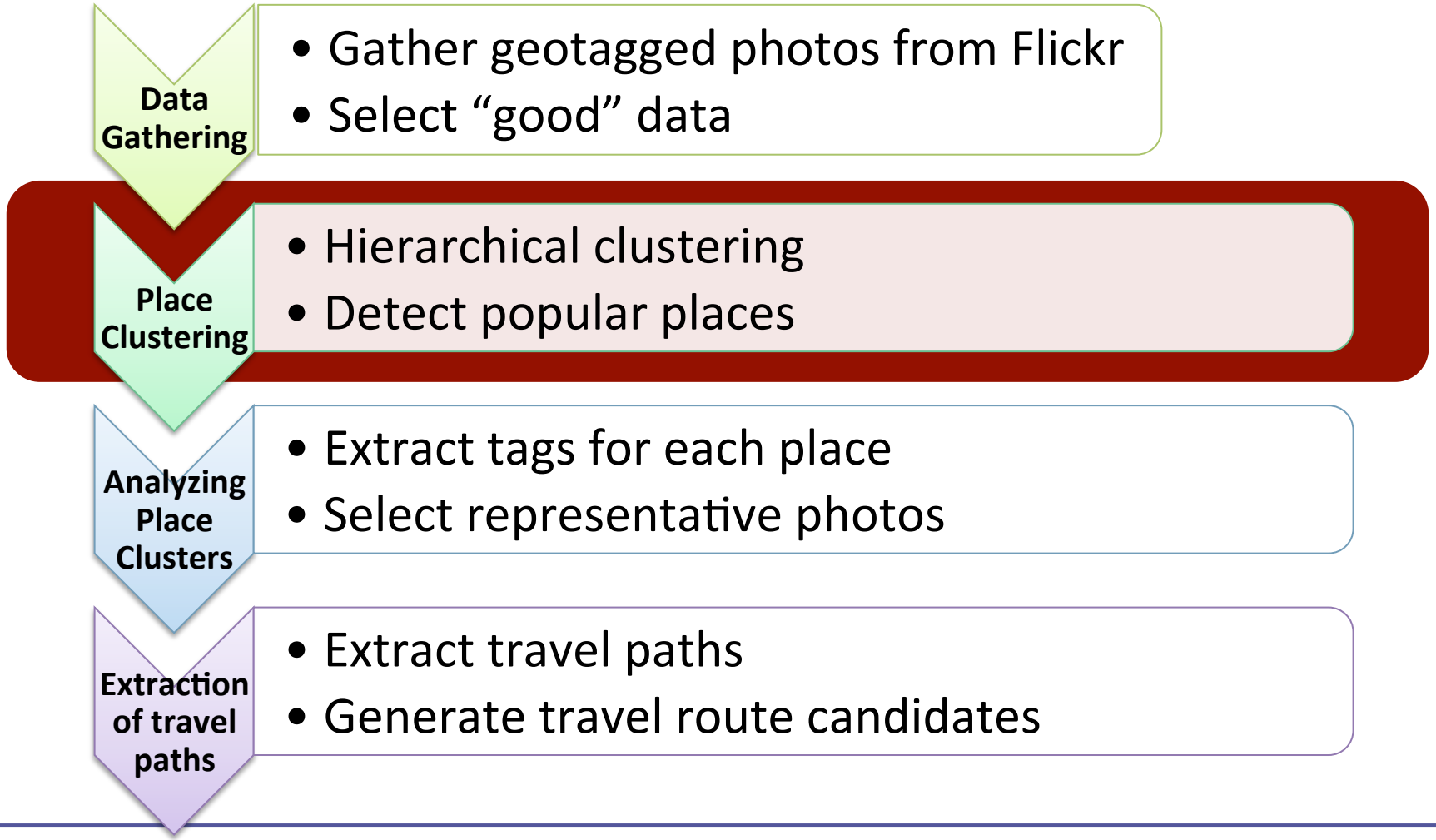
	Original # photos	After noise removal
User A	25189	192
User B	4793	4766

Metadata obtained via FlickrAPI

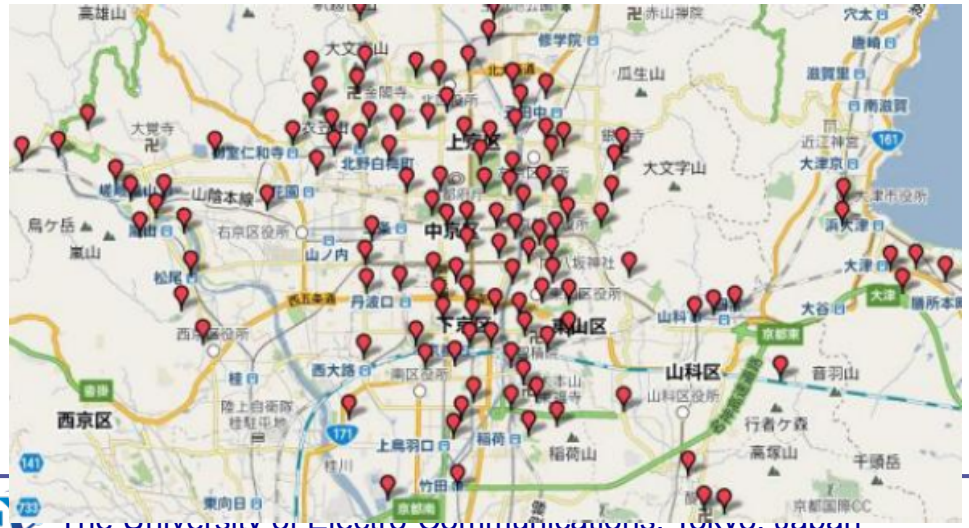
- Flickr API provides various information on Flickr photos.
  - Geotag (latitude and logitude)
  - Accuracy of geotag *(only use more than 8)*
  - UserID (owner ID) *(we can track user's trace)*
  - Text tags *(add text tags to each places)*

<photo id="5017594322" **owner="75008966@N00"** secret="f8d231e2cd" server="4154"
farm="5" title="The Kyoto Tower from Kyoto Station" ispublic="1" isfriend="0"
isfamily="0" **latitude="34.986409543437" longitude="135.75977325439453"**
accuracy="16" datetaken="2009-10-01 10:23:45" tags="kyoto japan kyoto tower " .../>

Overview of the offine processings



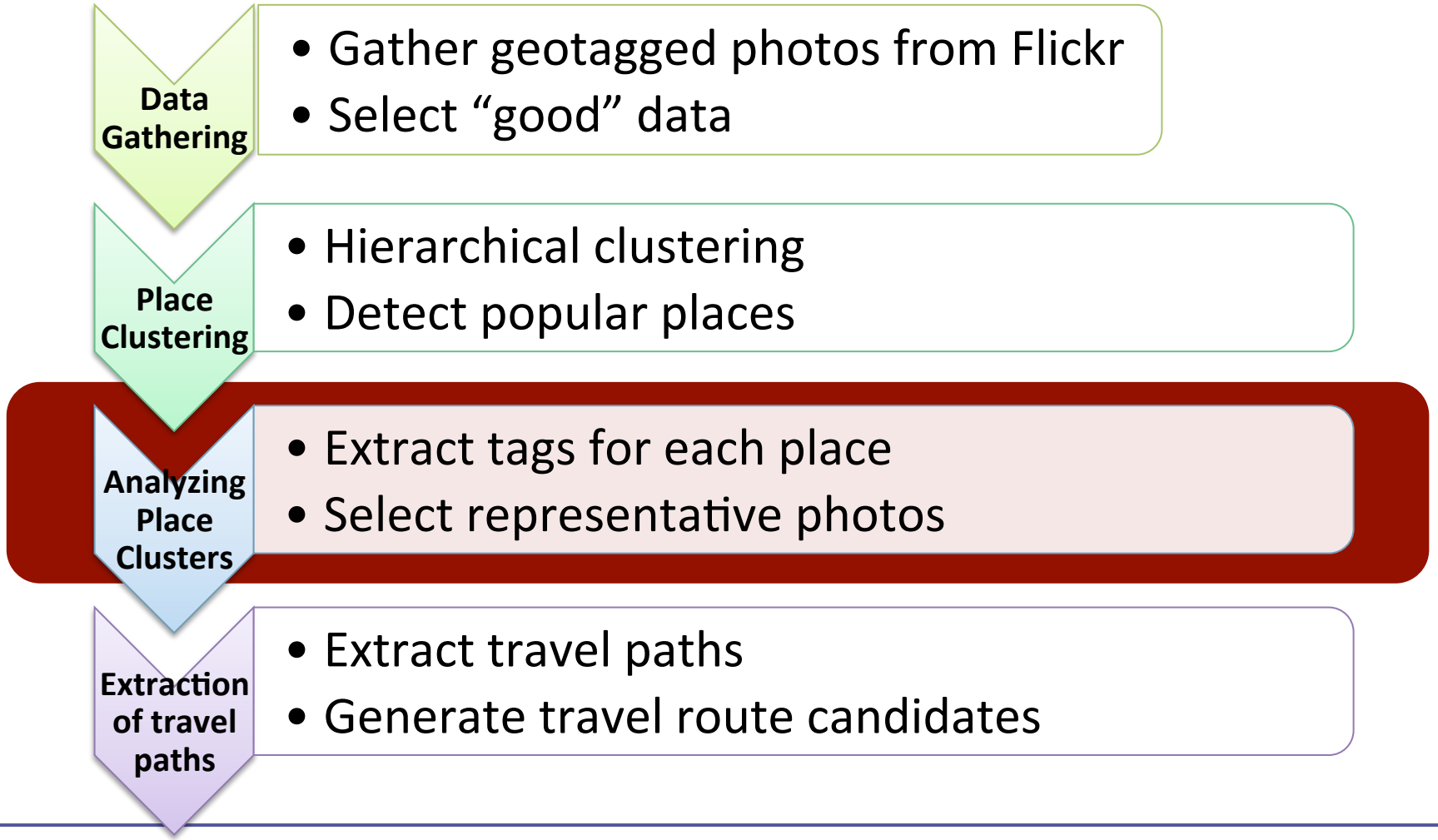
2) Place Clustering



To detect popular places, hierarchical clustering is carried out.

Cluster centers = popular tourist places

Overview of the offine processings



3-1) Add tags to each place

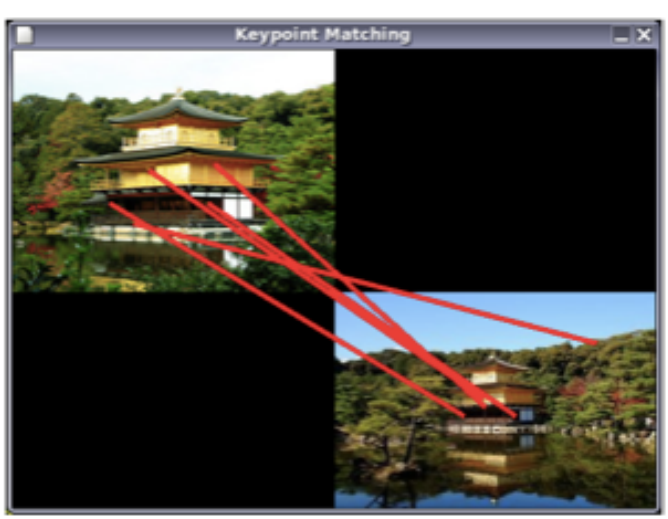
- Extract tags to explain the places
  - Extract all the tags within a place cluster
  - Remove stop words (e.g. Canon, Nikon,..)
  - Select only the tags where the following score is more than the pre-defined threshold.

$$score = \frac{\text{Number of the tag within the clutesr}}{\text{Total number of the tag over all the photos}}$$

Text tags associated with “Kiuomizu temple” cluster

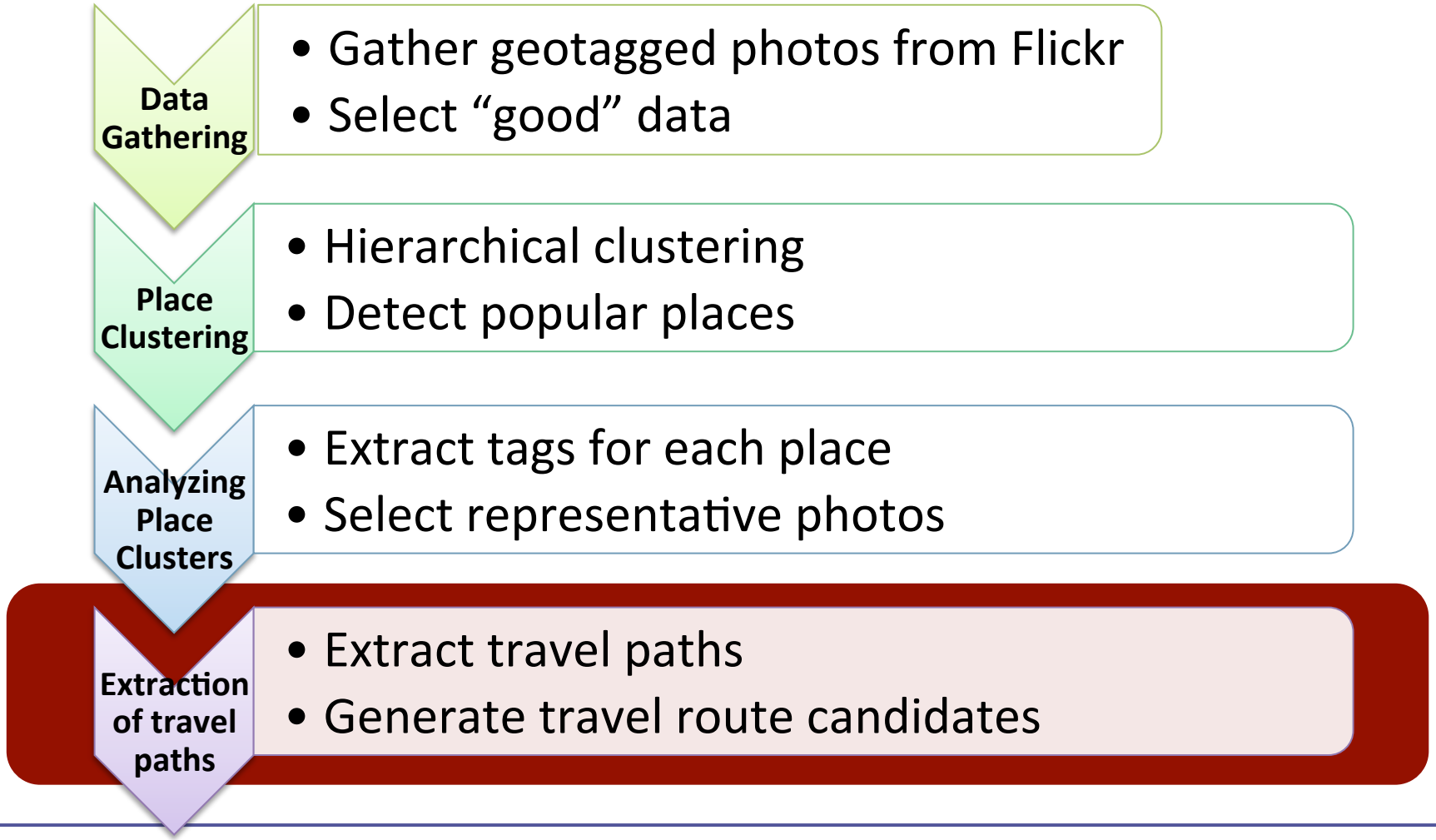
Before	**october japan oktober kyoto kiyomizudera 2009 kiyomizutemple**
After	**kiyomizudera kiyomizutemple**

3-2) Select representative photos



- Based on the number of matching points with SURF features.
- The photo gathering the most number of matching is regarded as a representative one.

Overview of the offine processings

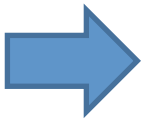


Extract traces based on userID, and simplify them

- Summarize gettag sequences into sequences of place clusters
  - Associate extracted geotags with place clusters

TRIP

34.9864095434377 ,135.7597732543
34.9878159325440 ,135.7596874237
34.9894332501478 ,135.762305259
34.9893629326552,135.7638072993
:

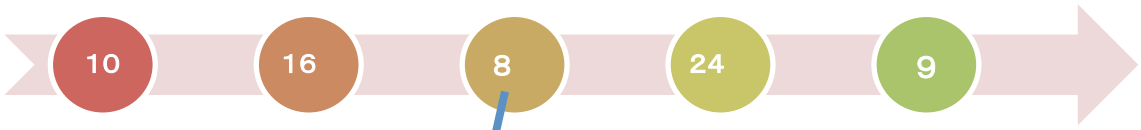


TRIP MODEL

A->B->C->D->E->F

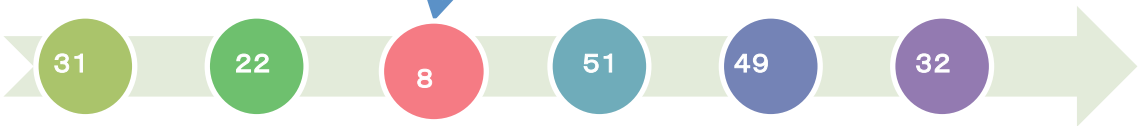
Combine “Trip models”

Trip Model A

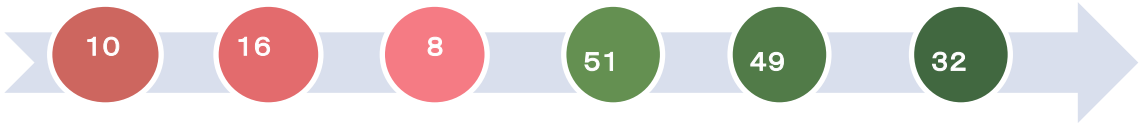


The same place was found !

Trip Model B



Concatenate the first part of Trip model A and the latter part of Trip model B



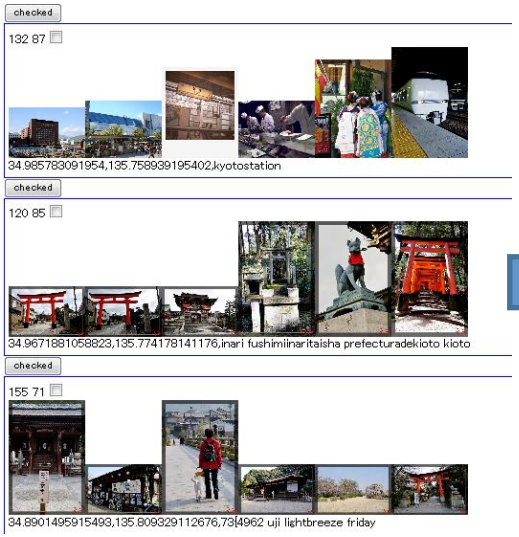
Newly generated trip model

Experiments

- 2million geotaggged photos from Flickr
  - To extract paths, a lot of getagged photos are needed.

- We focus on “Kyoto” area in Japan.
  - 162 unique userIDs.
  - 1805 geotag places
  - 154 place clusters
  - 18,752 routes

Online system(1): route recommendation system



Select landmark photos the user is interested in

Show travel route candidates

Showing the selected route on the map

Select landmarks to visit

Recommended routes 1

Recommended 3 routes

Table 4. The number of visiting tourist places, and the total and average distances of the three routes.

Route candidate	num. of places	Total distance (km)	Avg. dist. between places (km)
Trip 1	9	29.8	3.73
Trip 2	15	52.7	3.76
Trip 3	11	54.1	5.41

Online system(2): Travel reviewing system

Upload geotagged photos and GPS traces
after coming back home

Detect visited popular places and not-visited popular places

The system says
"You missed the famous places, OOO. Please visit there at the next time ! "

Conclusions

- We proposed a **Travel Planning System** using both geotagged photos and travel trajectories extracted from the Web.

The system has two functions:

(1) Travel route recommendation of both places and routes before traveling

(2) Travel reviewing of your travel after traveling

Future work

- Evaluation by user studies
- Experiments with other areas
- Detailed search option for routes
  - Duration, total distance of the route
  - Ranking of recommended routes
- Improvement of UI

- Implement a mobile-version real-time route recommendation system which can be used from smartphones while traveling

旅行記録の整理

1. ユーザの移動経路を読み込む
   a. GPSの軌跡
   b. 廻った観光地のマッピング
2. 観光地クラスタ間の移動に置き換える
   – 移動を要約する
3. そのルートで他にも行けた観光地を提示
   – 他ユーザの観光ルート情報を利用

まとめ

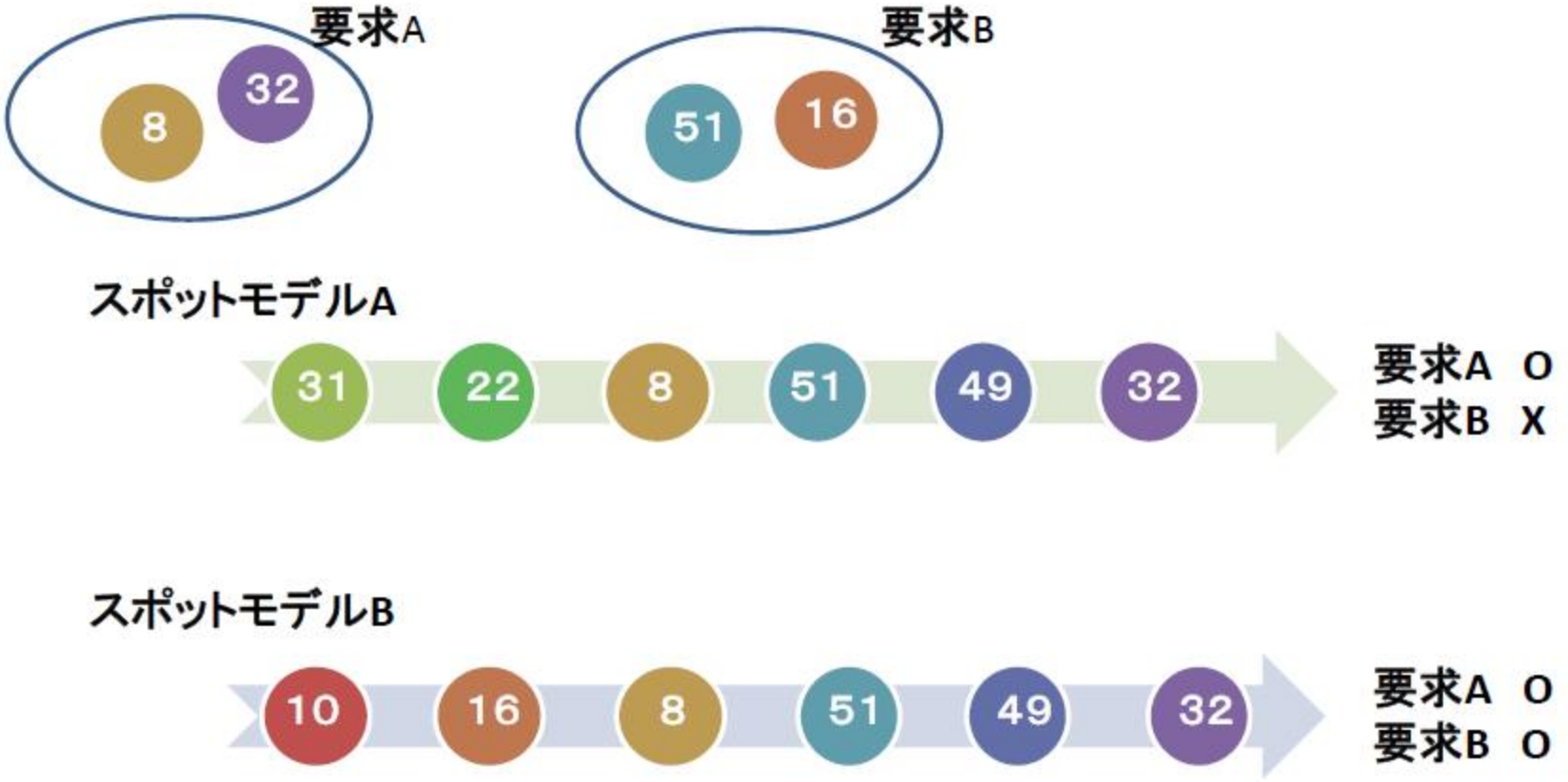
- 位置情報付き写真を利用した観光旅行支援のシステムを開発した

1. 仮想観光アルバム
   - 順序を意識した観光ルートの提案
2. 旅行アルバム補完
   - 旅行記録の要約と付加情報の追加

観光ルート推薦

1. ユーザの入力
   - クラスタのテキストタグ，代表画像を表示
   - いくつかの観光地を選択
2. 候補トリップモデルの検索
   - 選択されたものを含むトリップモデルを表示
3. 候補経路表示
   - Google maps APIを利用しルートを表示

候補の選択


要求A
8
32
要求B
51
16
スポットモデルA
31
22
8
51
49
32
要求A O
要求B X
スポットモデルB
10
16
8
51
49
32
要求A O
要求B O

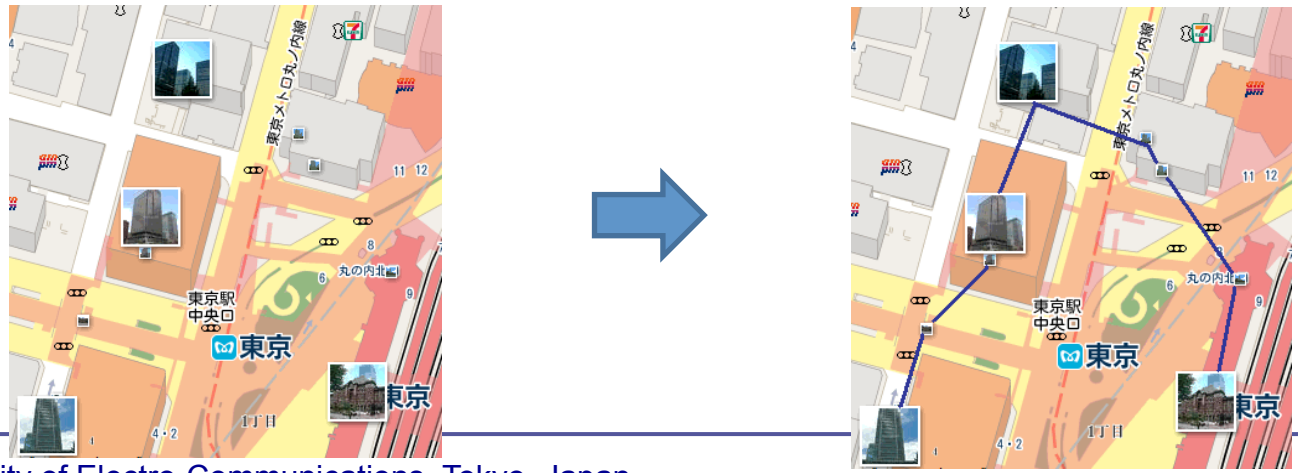
ランダムとの比較

トリップモデル数の結合数毎の内訳

結合数	トリップモデル数
単独	106
2	645
3	559
4	17,442

予備実験

- 軌跡の取り方
  1. 大量に画像情報を収集
  2. Geo-tag画像を選抜
  3. ユーザ毎のIDで時間順にソート
  4. 撮影位置をつなぎ移動の軌跡を求める



予備実験

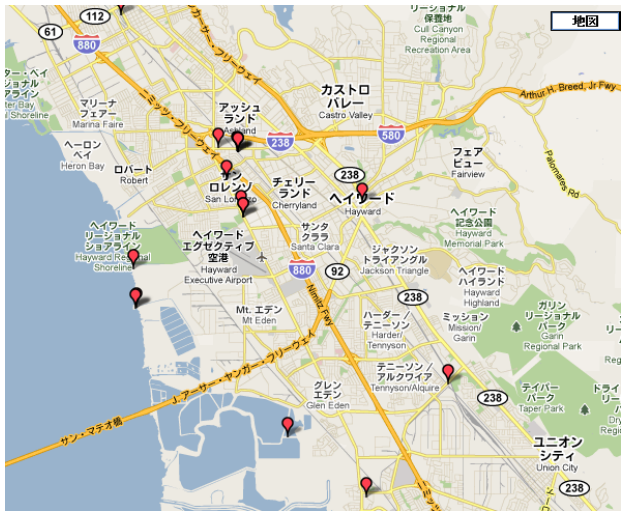
- 約200万枚の画像の情報を利用
- 位置情報が正確でなくてはならない
  - 重複した位置のものを除く（下表）
    - ユーザAのデータは位置情報が信用できない

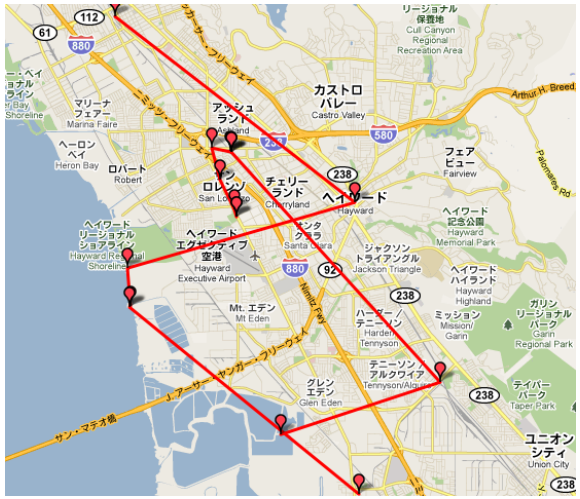
対象	元のデータ数	除外後のデータ数
ユーザA	25189	192
ユーザB	4793	4766

- 70万枚ほどに選抜された

予備実験

- あるユーザの撮影位置から軌跡を求めた
  - マーカが撮影位置
  - ラインが軌跡
    - 同じ日に撮影していったものを結んだ





予備実験

ある地域間の比較

- 地域で画像数に偏りがある
  - 観光地としての人気
  - 軌跡がとれない
- 地域の中でも偏りがある
  - 人気の観光スポットの範囲
  - 観光スポット内での移動は考慮する必要がある