Exploiting Geospatial Properties for Efficient Visual Data Management and Learning

Seon Ho Kim, Ph.D.

Integrated Media Systems Center

Viterbi School of Engineering

University of Southern California, CA, USA

seonkim@usc.edu

Outline

- Motivation
- Modeling Spatial Property of Visual Content
  - Point Location
  - FOV Model
  - Image Scene Location
- Harnessing Spatial Property in Data Management
  - Spatial Coverage Measurement
  - Efficient Data Collection
  - Access Method
  - Image Machine Learning with Edge Computing
- Conclusion

Outline

- **Motivation**
- Modeling Spatial Property of Visual Content
  - Point Location
  - FOV Model
  - Image Scene Location
- Harnessing Spatial Property in Data Management
  - Spatial Coverage Measurement
  - Efficient Data Collection
  - Access Method
  - Image Machine Learning with Edge Computing
- Conclusion

Visual Data

- Visual data: images, videos, and feature vectors
- Visual data is one of the biggest data
- Analytics on video or image data, either off-line or streaming, have become prolific across a wide range of application domains [1].
  - due to the growing ability of machine learning techniques to extract information
- Despite this rich and varied usage environment, there has been very little research on the management of visual data [1].
  - Ad-hoc collection of tools, unique and individual solutions
  - Seeks for new approaches

[1] NIPS, 2018

Why Location with Visual Data?

- Many visual data (images & videos) are *naturally tied with geographical information*
  - Surveillance, traffic, real estate, leisure, to name a few
- To better organize images & videos in large datasets
  - Indexing, searching
- Integrate visual data with other information
- Machine learning (spatial visual correlation)

So, tag and utilize the location of image

How to Get Location?

- Easier to capture location using GPS-equipped Cameras
  - E.g., smartphone, table, digital camera, GoPro

Enable taking photos which are tagged with location

Already Lots of Geo-tagged Image Datasets

- **User-generated Images**
  - Ubiquity of smartphone users
    - Billions of mobile subscriptions
  - Network bandwidth improvements
  - Growth of image sharing online services

flickr

E.g., Flickr collected 200+ million geo-tagged images

- **Professional-generated Images**

E.g., Google Street View Project collected photos for 3000 cities

Urban streets are being documented with geo-tagged images & videos

Many Applications: e.g., Smart City

Motivation

- State of the art techniques utilize camera location (e.g., GPS input) for organizing and searching images/videos
  – However, only camera location is not enough
- Location data do not have ***human viewpoints***
  – Viewing direction, Distance between camera and object (appearance of object), Semantics
- Any methods to potentially help managing visual data at the high semantic level preferred by humans?

Issues to Consider

- More and more visual data are generated
  - More amount than human can physically watch → machine watches
- Still, visual data collection is expensive and in adhoc manner
  - Size of data, orchestration of collection, timely collection
- Systematic use of visual data is not much available
  - Search, index, sharing, annotation, etc.
- Preparation for AI and machine learning is needed
  - Machine automatically selects dataset for learning?

The Question

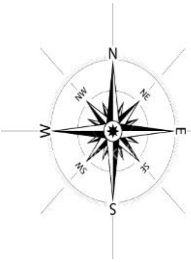
- For efficient visual data collection, indexing, searching, and furthermore diverse image machine learning, how can we use *geographical properties of visual data*?

Outline

- Motivation
- **Modeling Spatial Property of Visual Content**
  - **Point Location**
  - **FOV Model**
  - **Image Scene Location**
- Harnessing Spatial Property in Data Management
  - Spatial Coverage Measurement
  - Efficient Data Collection
  - Access Method
  - Image Machine Learning with Edge Computing
- Conclusion

Location Data Collection

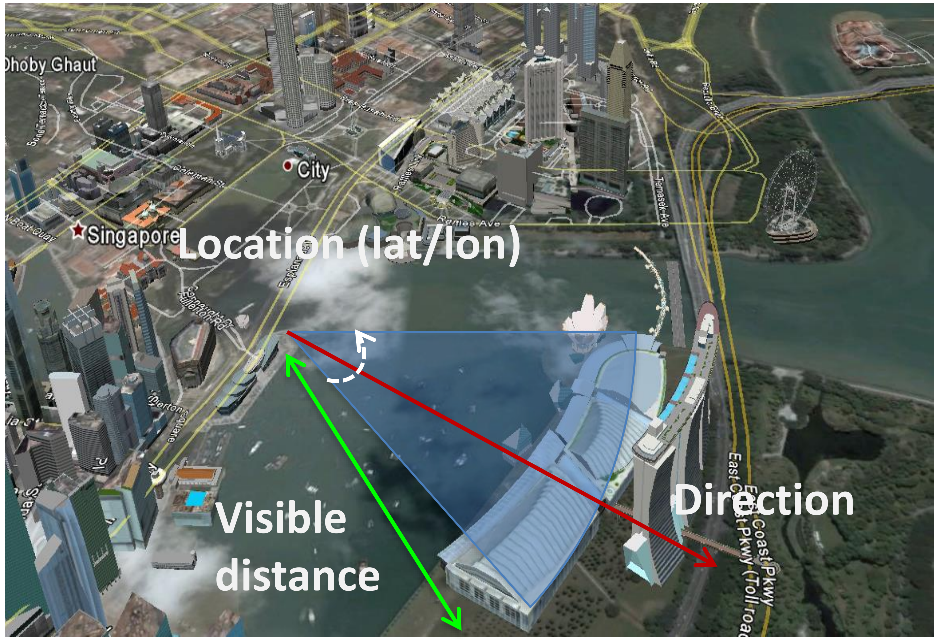
GPS → Latitude/Longitude

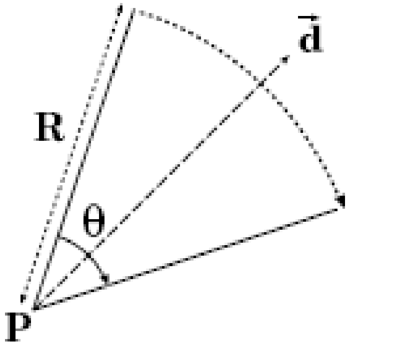


Compass → Direction

Modeling Viewable Scene of Image

- Accurately describe visual content through ***field-of-view (FOV) model***



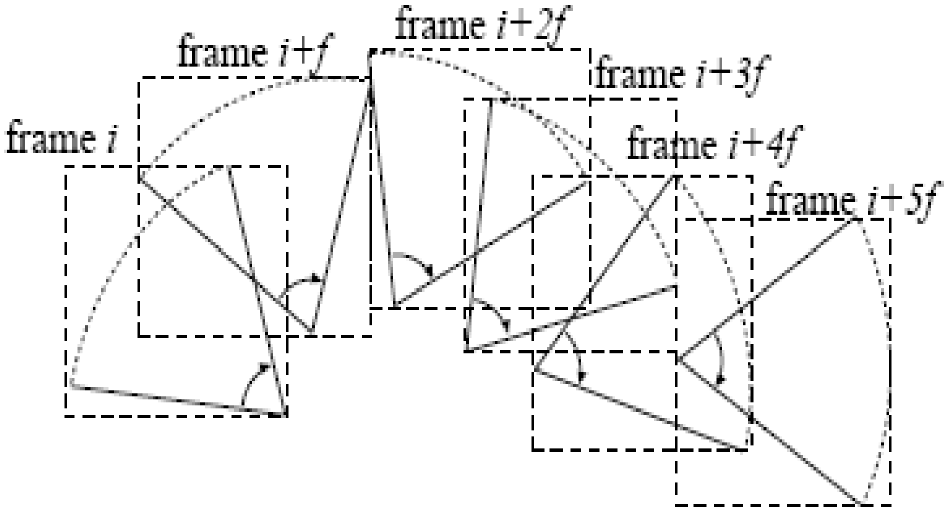


P <longitude,latitude>: camera location

θ : viewable angle

$\vec{d}$: camera direction vector

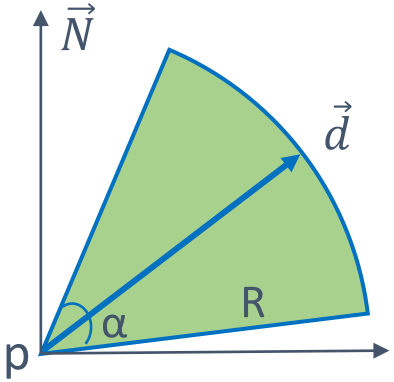
R : visible distance



FOVScene description is generated every *f* frames

Extended – Field of View (FOV) Model

Geo-tagged Images (or Video Frames)



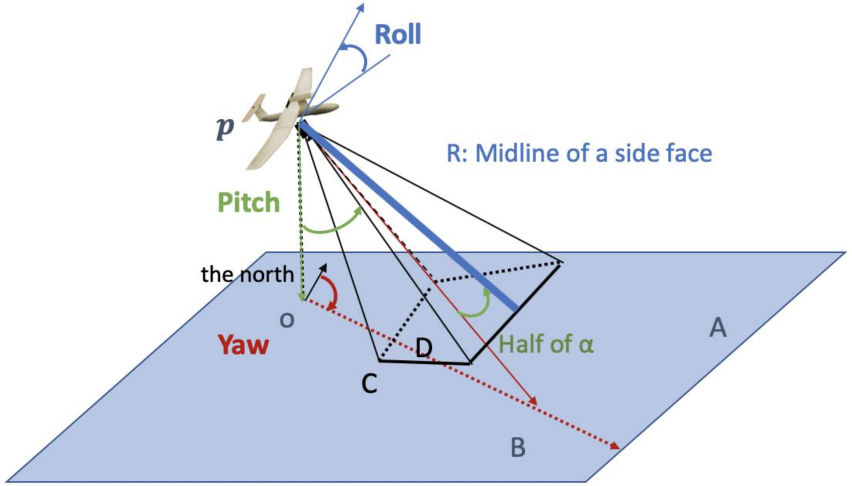
p: *camera location*

$\vec{d}$: *camera view orientation*

α: *viewable angle*

R: *viewable distance*

2D FOV Model [2]



3D FOV Model [3]

[2] ACM MM, 2008. [3] MMM, 2020.

Meaning of Viewable Scene Models

- What does this mean?

New Visual Data
Management Solution

Image/Video as
Spatial objects

Spatial Database
Technologies

Challenging Video Problem → Known Database Problem

More Precise Model

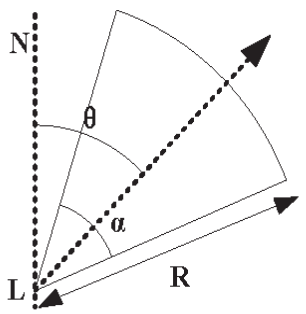
- **Spatial Representation**

Geo-location Point → Field of View (FOV) → Scene Location [4]

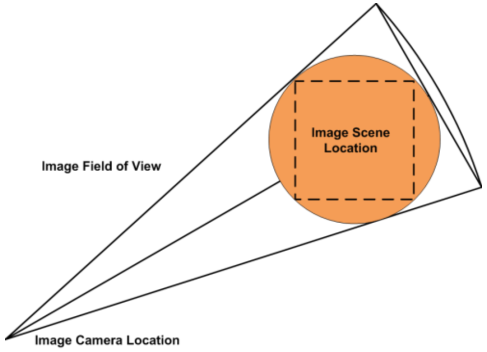
Geo-location Point
- ✗ imprecise

Field of View (FOV)
- ✓ Better spatial representation
- ✗ Loosely representing spatial extent

Scene Location [4]
- ✓ Precise spatial representation
- ✓ Tightly representing spatial extent



L :<longitude,latitude>
α :viewable angle
θ :camera direction
R :maximum visible distance



[4] ACM MM, 2018.

Spatial Keywords

- Searching videos using geo-coordinates (figure) is good and effective, BUT…

"Is this the best?"
People are already familiar with keyword-based search!

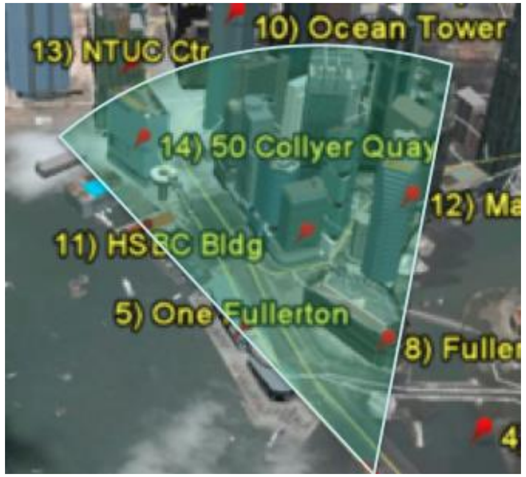
We have far more information in the virtual world
-Geographic Information Systems

Any way to utilize textual keywords?

Retrieving Spatial Keywords

Geographic Information Systems

Geo-Coordinates



Spatial Keywords

Visual Features & Keywords

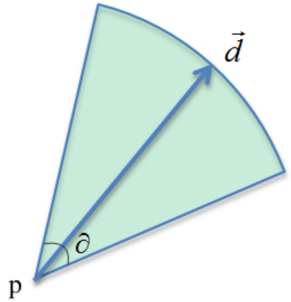
Metadata from Visual Analytics

Record video using
Camera with sensors

→ **Video/Image Analysis** →

Buildings,
Tommy Trojan

Visual Data Organization and Indexing



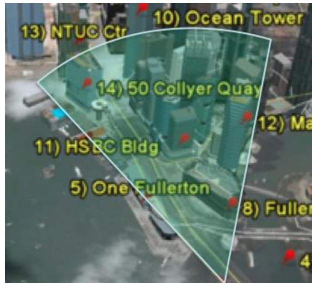
p : camera location

$\vec{d}$: camera direction vector

∂ : viewable angle

t : timestamp

Database Technologies
(hybrid storing and indexing, dynamic update, searching, etc.)



Tommy Trojan
USC

John

Outline

- Motivation
- Modeling Spatial Property of Visual Content
  - Point Location
  - FOV Model
  - Image Scene Location
- **Harnessing Spatial Property in Data Management**
  - **Spatial Coverage Measurement**
  - **Efficient Data Collection**
  - **Access Method**
  - **Image Machine Learning with Edge Computing**
- Conclusion

Spatial Coverage Model [4][5]

[4] MMM, 2020. [5] IEEE BigMM, 2018.

Basic Question

- When we have thousands of geo-tagged images in an area (e.g., Los Angeles downtown), how do we measure how much they visually cover the area?
  - Human perception → e.g., direction
  - Completeness → how much is enough? (for human and AI)
- How do we identify areas with no visual information?
  - Automatic crowdsourcing data for complete coverage

Problem Definition

- Given a field-of-view dataset $\mathcal{F}$ and a query range R_q, the spatial coverage measurement (SCM) problem is formulated as

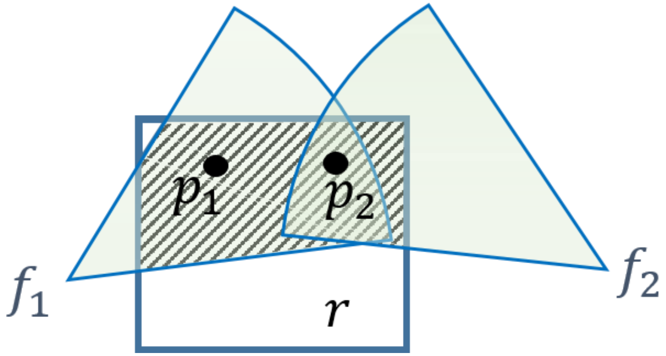
$$\boldsymbol{SCM}(\boldsymbol{\mathcal{F}}, \boldsymbol{R_q}) = \boldsymbol{\rho}$$

- ρ is the geo-awareness percentage of $\mathcal{F}$ to the visual space located in R_q.

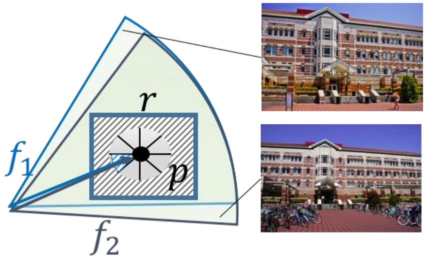
Baseline: Area Coverage Model (ACM)

- ACM estimates the percentage of the area R_q covered by $\mathcal{F}$.

$$ACM(R_q, \mathcal{F}) = \frac{Area(Overlap(R_q, \mathcal{F}))}{Area(R_q)}$$

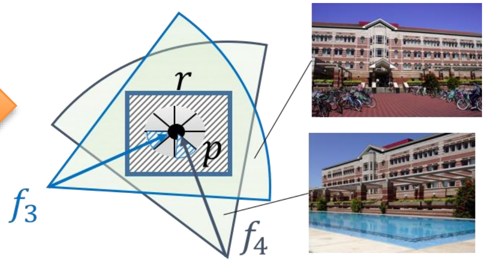


✗ ACM does not consider the directional property of $\mathcal{F}$.



ACM(R_q, $\mathcal{F}$) = 100%

Visual Perception of R_q by $\mathcal{F}$ is different!

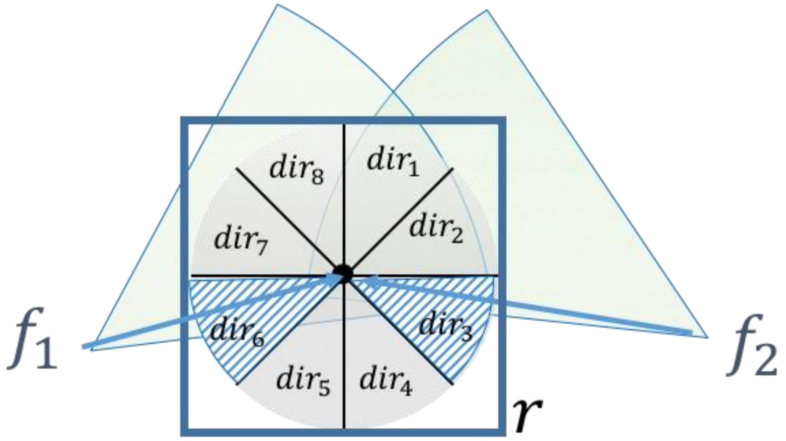


Directional Coverage Measurement Model (DCM)

- DCM extends ACM to measure the visibility of the area R_q from various directions
  - Divide R_q into a set of directional sectors.
  - Calculate ACM for every sector and then aggregate the result.

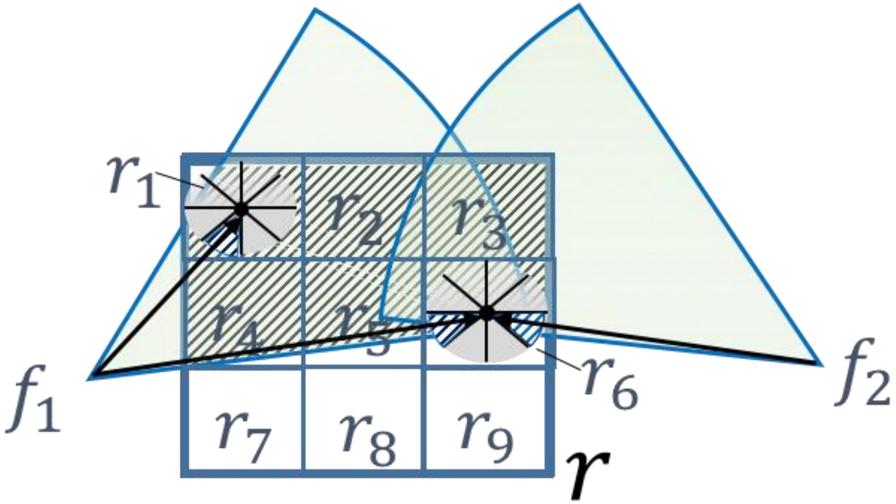
Define the maximum circle C inscribed in R_q where C.center = R_q.center.

$$DCM(R_q, \mathcal{F}, d) = \frac{\sum_{i=1}^{d} Coverage_{dir}(R_q, \mathcal{F}, s_i)}{d}$$



Cell Coverage Measurement Model (CCM)

- CCM extends DCM to measure the visibility at a finer granularity by dividing R_q into cells and evaluating the visibility of each cell, then aggregate the results. (Algorithm 1 in paper)

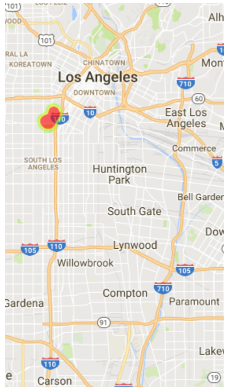


Experiments with Large Scale Datasets

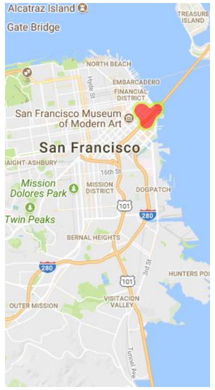
Range Query (R_q)	# of images	Area (km^2)	Avg. # images / km^2
Manhattan (MA)	21, 947	13.7 × 7.3	219
Pittsburg (PT)	5, 940	1.5 × 3.6	1, 100
Los Angeles (LA)	36, 624	0.9 × 1.7	23, 459
San Francisco (SF)	409, 862	4.4 × 4.2	22, 429

Manhattan,
NY, USA

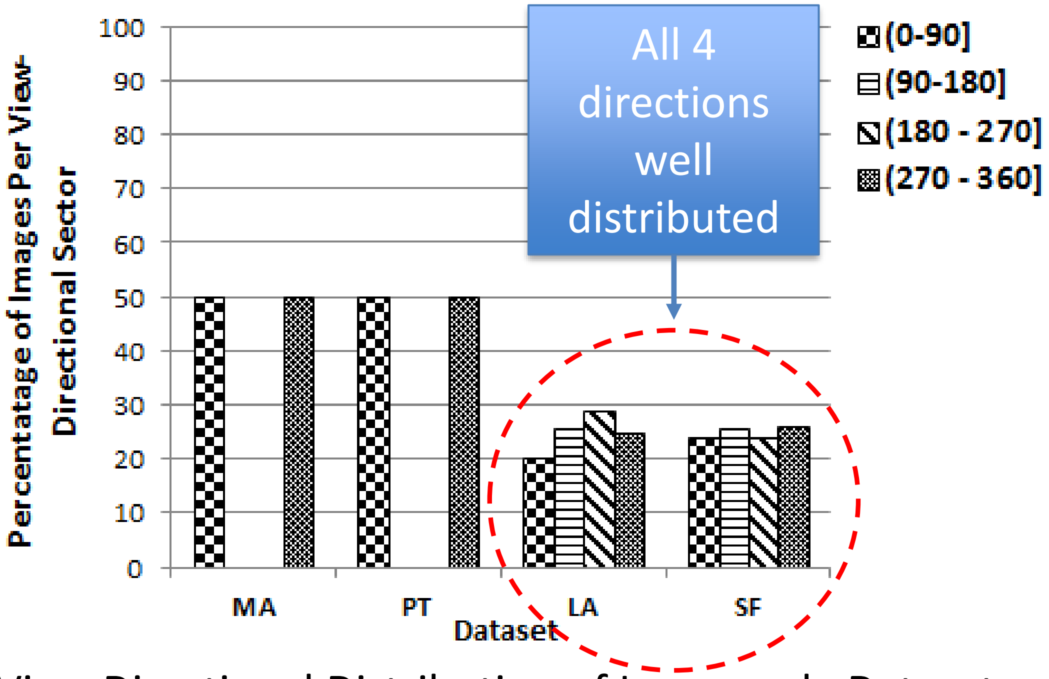
Pittsburg,
PA, USA



Los Angeles,
CA, USA



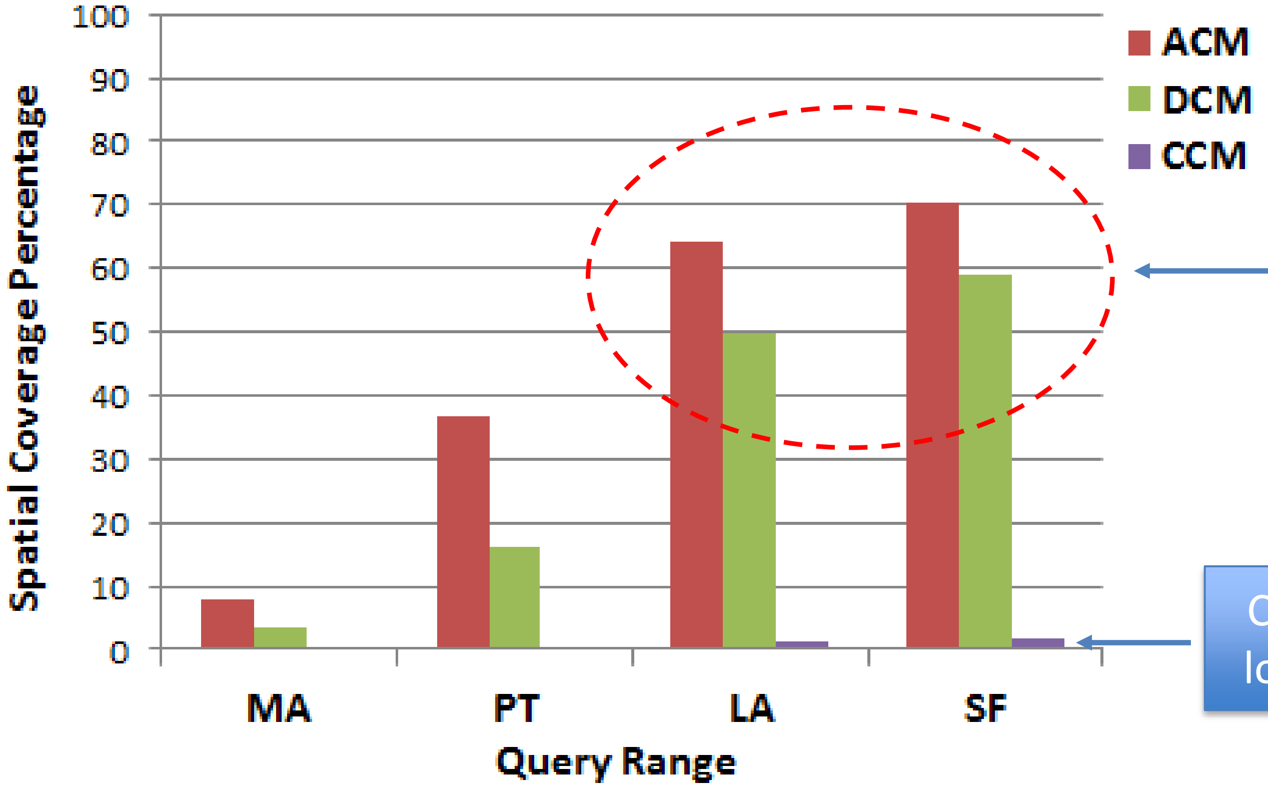
San Francisco,
CA, USA



View Directional Distribution of Large-scale Datasets

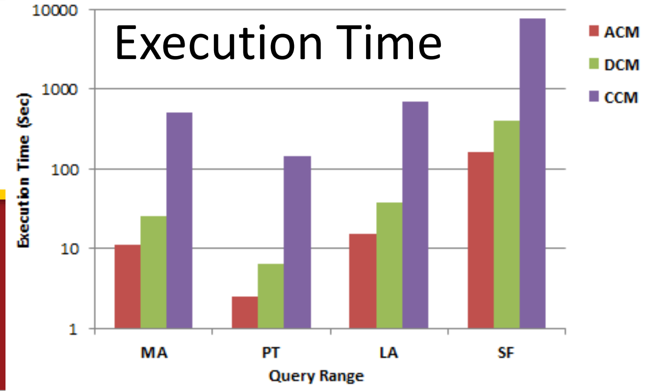
ACM, DCM, and CCM with Large Datasets

Overall coverage depends on image density

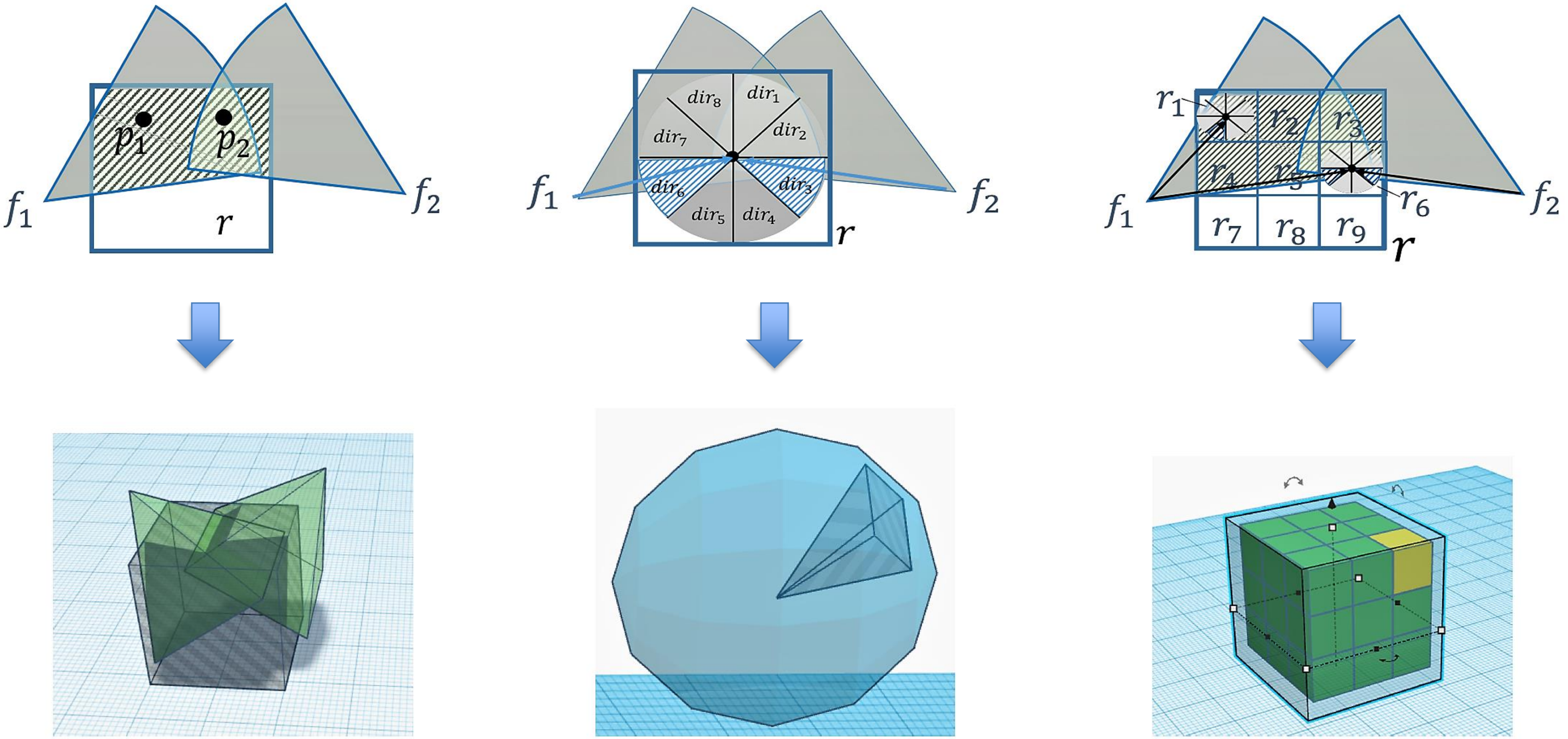


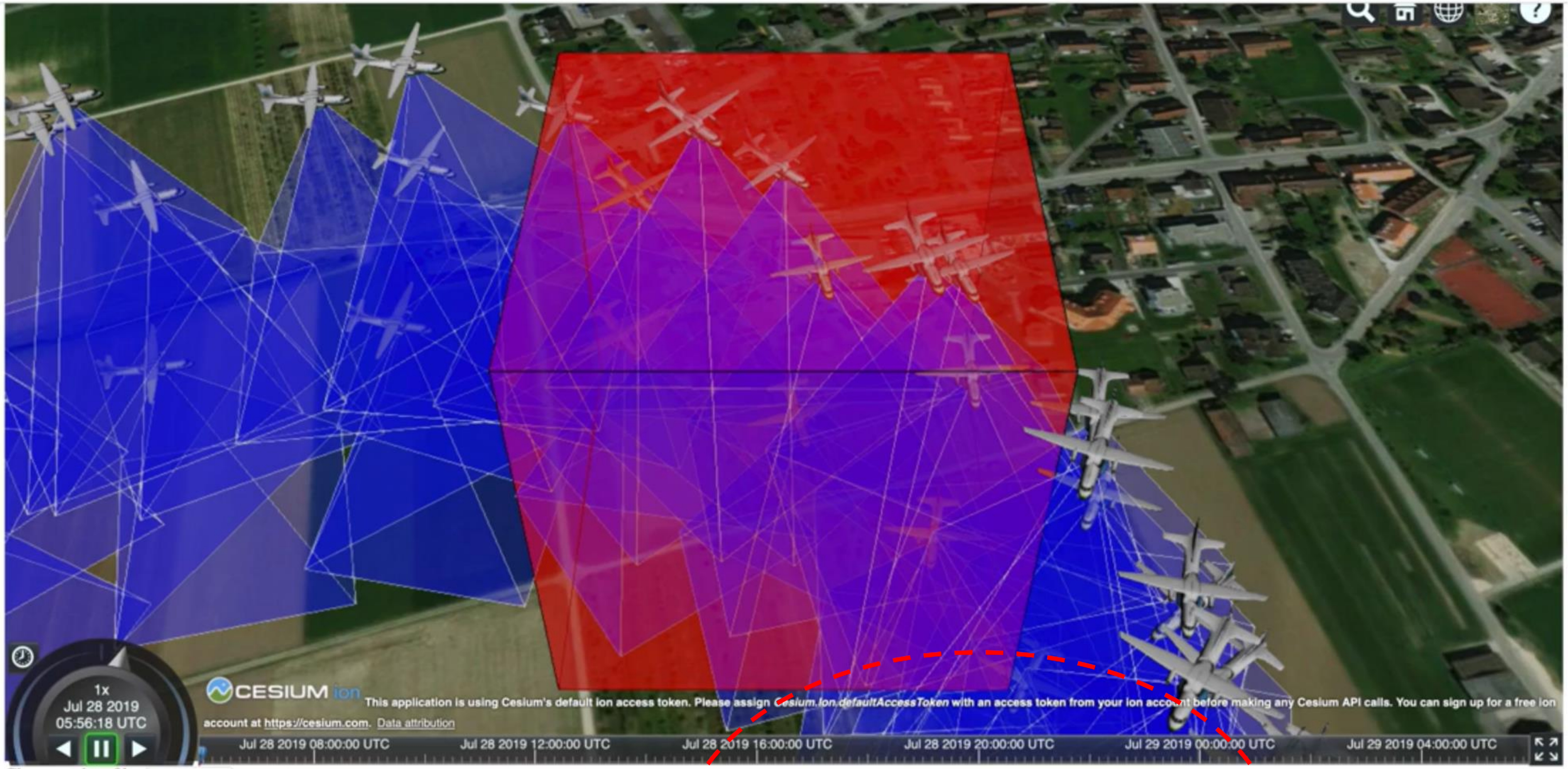
Images with well distributed directions

CCM requires lots of images



2D → 3D Spatial Coverage Model



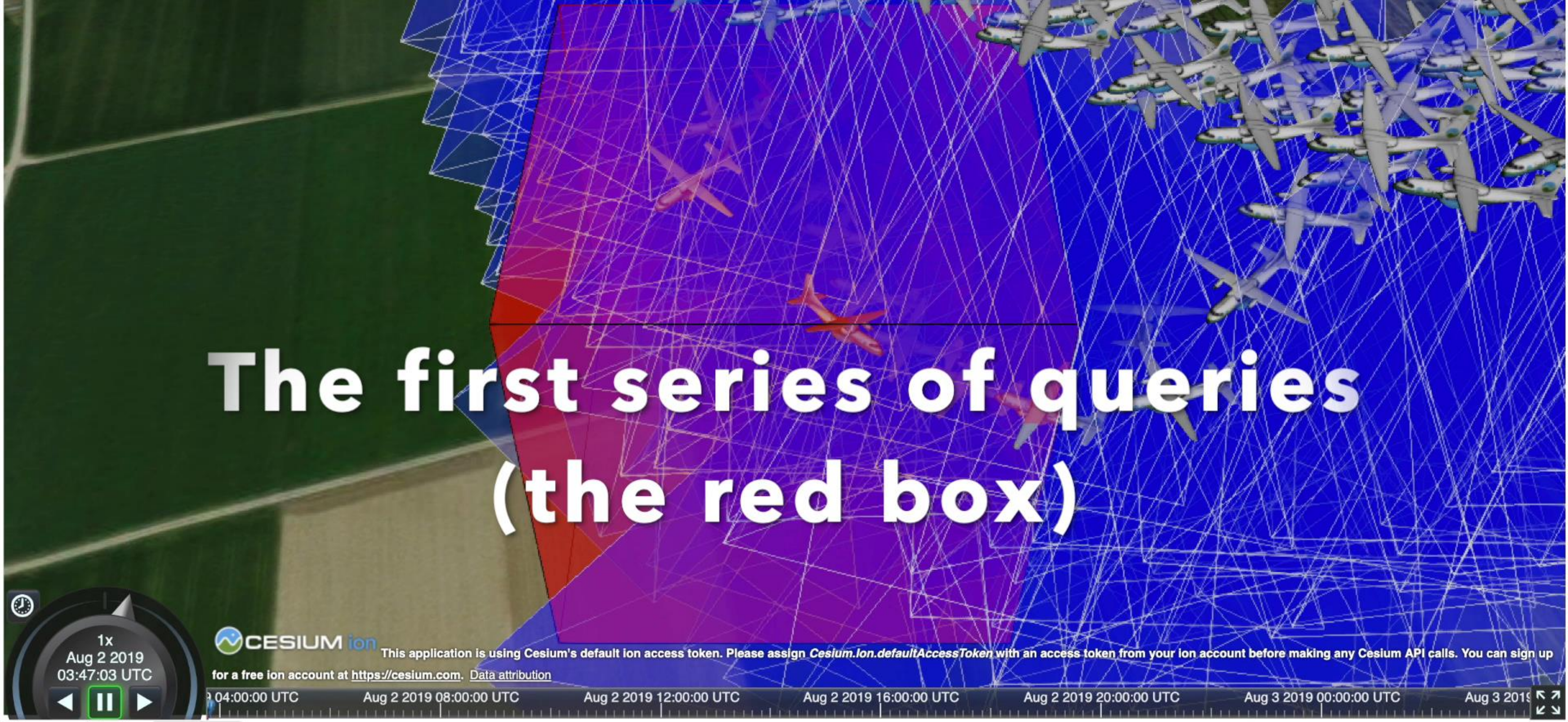


Choose a data file: Choose File extracted-data.csv
total number of frames: 645436
Start frame number 40000
End frame number 42000
select one from how many frames 10
R value 100
Alpha value 45
draw
Choose a query file: Choose File demoQuery.txt

Coverage using Alg1: 31.20%

Coverage using Alg2: 3.75%

Coverage using Alg3: 1.13%



Choose a data file: Choose File extracted-data.csv

total number of frames: 645436

Start frame number 40000

End frame number 42000

select one from how many frames 10

R value 100

Alpha value 45

No file chosen

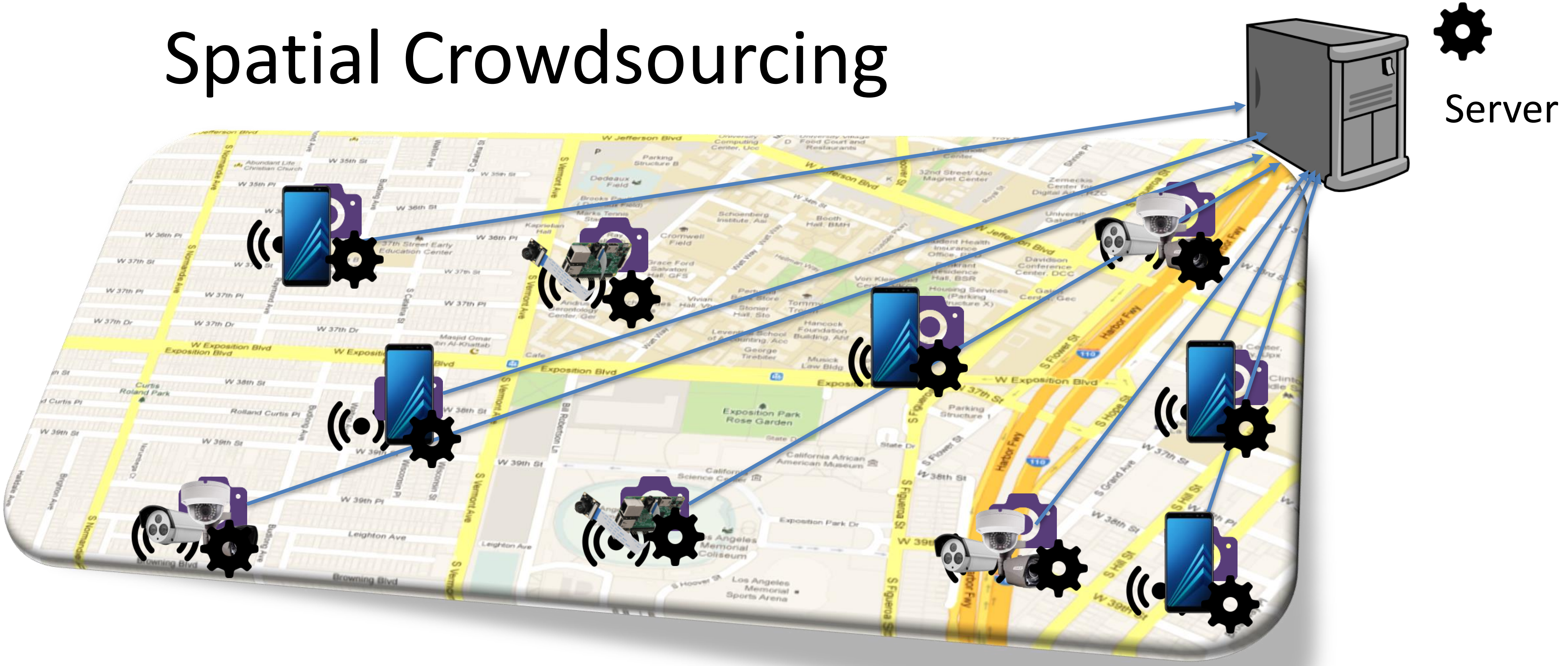
Coverage using Volume Coverage Model: 34.50%

Coverage using Euler-based Directional Coverage Model: 6.15%

Coverage using Weighted Cell Coverage Model: 1.55%

Efficient Data Collection

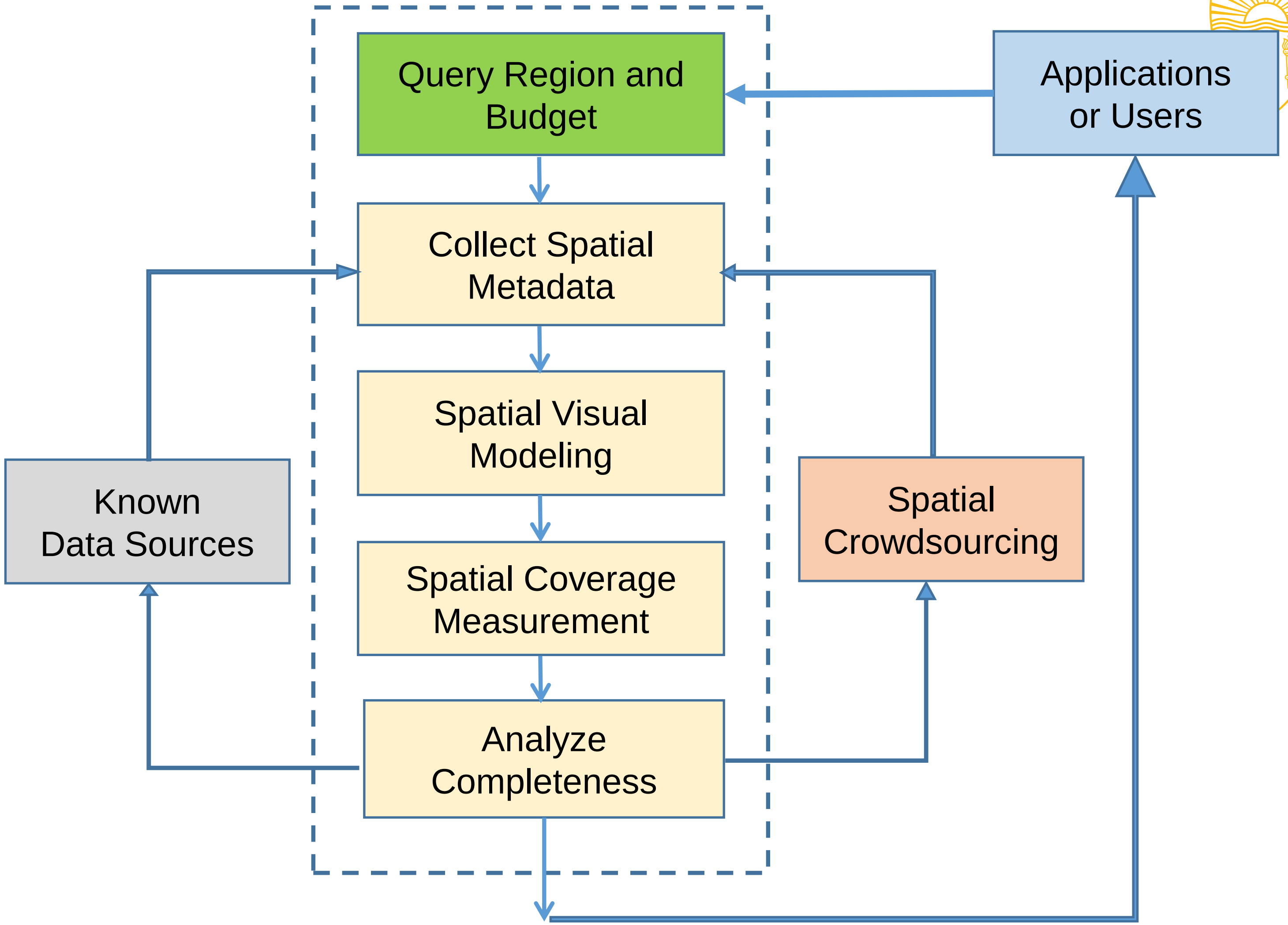
Spatial Crowdsourcing



Computing paradigm where humans are actively enrolled to participate in data collection (in our case, visual data w/ locations), especially at a certain location and time.

Spatial Crowdsourcing

- Continuous Collection and Management [6]
- Coverage Measure
- Spatial Visual Model
- Spatial Crowdsourcing
- Customized datasets
- Collaborative utility



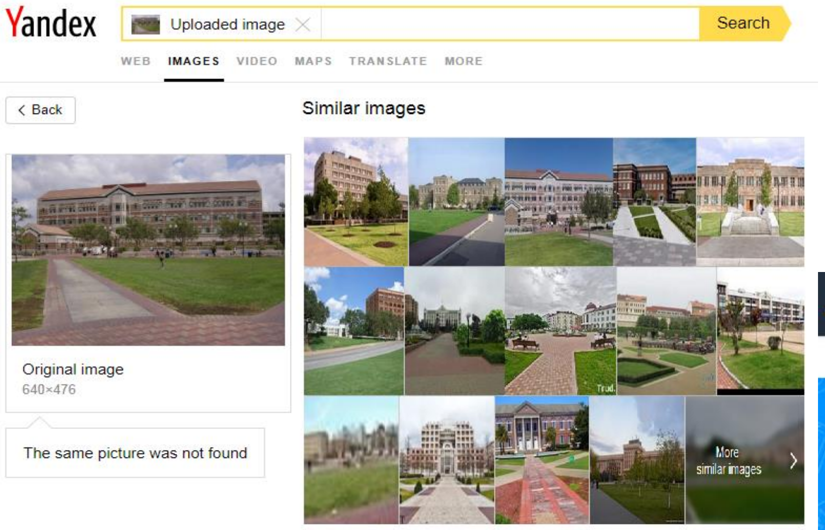
USC Viterbi School of Engineering Integrated Media Systems Center

[6] IEEE BigData, 2015.

Access Method

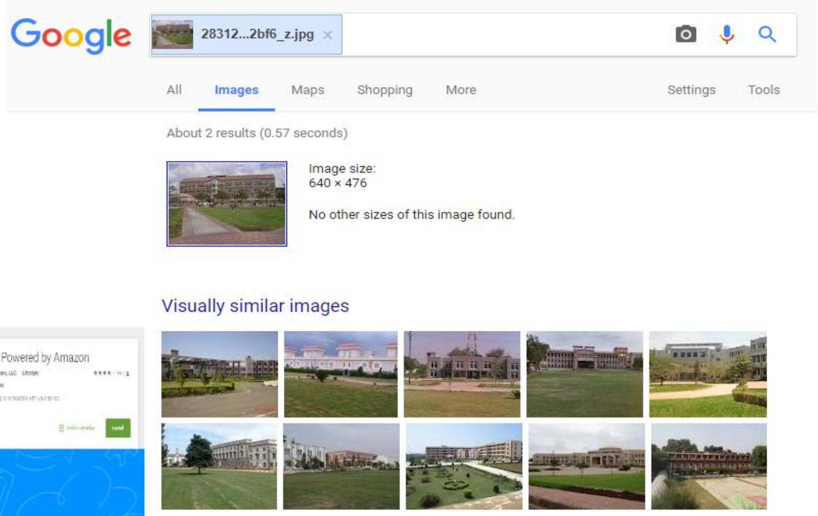
Conventional Image Search

- *Query By Example (QBE)*



https://yandex.com/images/

Amazon Flow Mobile App



https://images.google.com/

Conventional Image Search (Cont…)

- *Query by Geo-location*

Flickr Photo Search API

flickr Sign Up Explore Create Photos, people, or groups Sign In

The App Garden

Create an App | API Documentation | Feeds | What is the App Garden?

flickr.photos.search

Return a list of photos matching some criteria. Only photos visible to the calling user will be returned. To return private or semi-private photos, the caller must be authenticated with 'read' permissions, and have permission to view the photos. Unauthenticated calls will only return public photos.

Authentication

This method does not require authentication.

Arguments

`lat` (Optional)
: A valid latitude, in decimal format, for doing radial geo queries.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters — If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

`lon` (Optional)
: A valid longitude, in decimal format, for doing radial geo queries.

Geo queries require some sort of limiting agent in order to prevent the database from crying. This is basically like the check against "parameterless searches" for queries without a geo component.

A tag, for instance, is considered a limiting agent as are user defined min_date_taken and min_date_upload parameters — If no limiting factor is passed we return only photos added in the last 12 hours (though we may extend the limit in the future).

`radius` (Optional)
: A valid radius used for geo queries, greater than zero and less than 20 miles (or 32 kilometers), for use with point-based geo queries. The default value is 5 (km).

www.flickr.com/services/api/flickr.photos.search.html

Instagram Media Search API

Instagram Sandbox Invites Manage Clients Log in

Search Documentation
- Overview
- Authentication
- Login Permissions
- Permissions Review
- Sandbox Mode
- Secure Requests
- Endpoints
  - Users
  - Media
  - Comments
  - Tags
  - Locations

Media Endpoints

GET /media/search ··· Search for recent media in a given area.

GET /media/search

https://api.instagram.com/v1/media/search?lat=48.858844&lng=2.294351&access_token=ACCESS-TOKEN RESPONSE

Search for recent media in a given area.

REQUIREMENTS

Scope: public_content

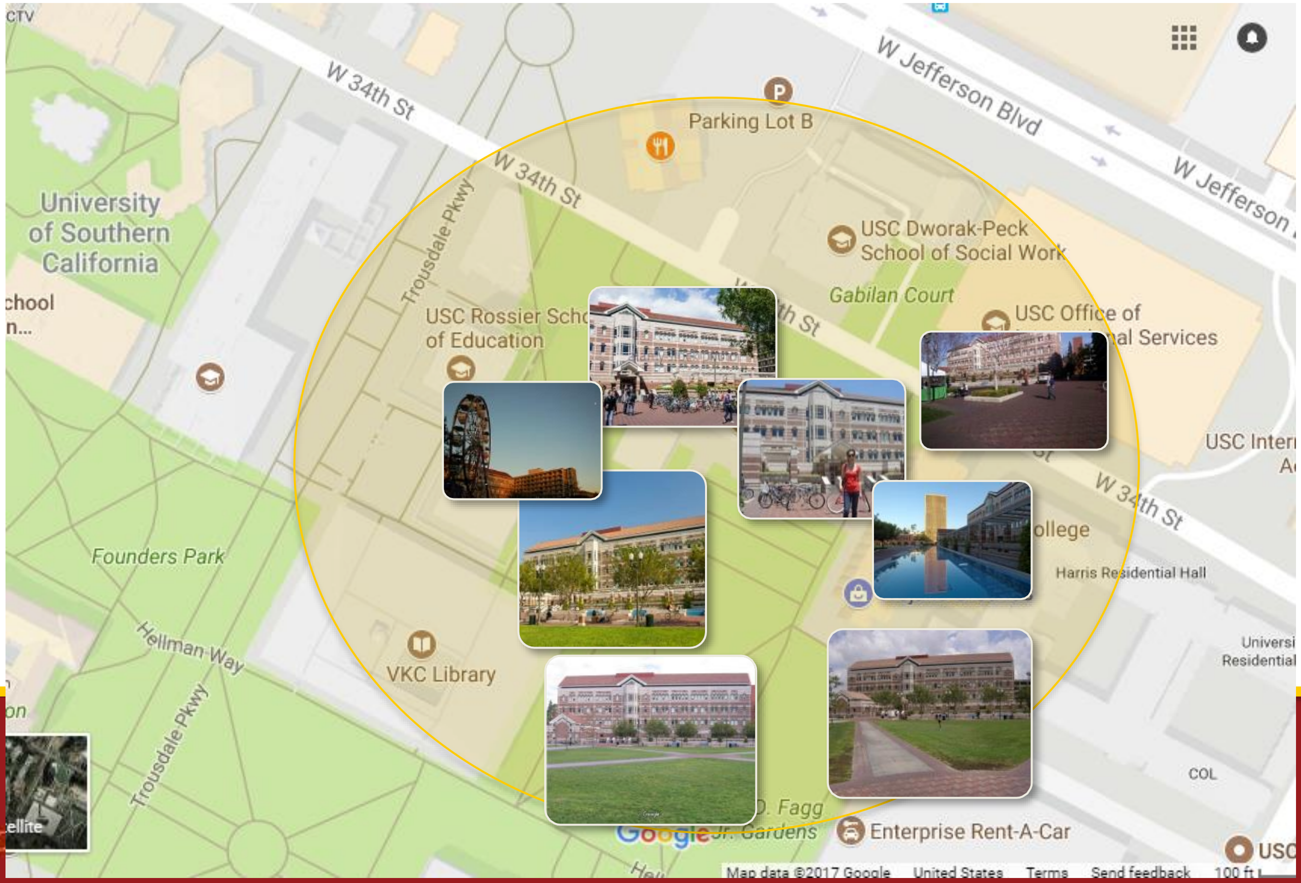
PARAMETERS

ACCESS_TOKEN	A valid access token.
LAT	Latitude of the center search coordinate. If used, lng is required.
LNG	Longitude of the center search coordinate. If used, lat is required.
DISTANCE	Default is 1km (distance=1000), max distance is 5km.

Spatial-Visual Search

Spatial-Visual Search: find similar images to a given query image and simultaneously within a given geographical area.

Query Spatial Range

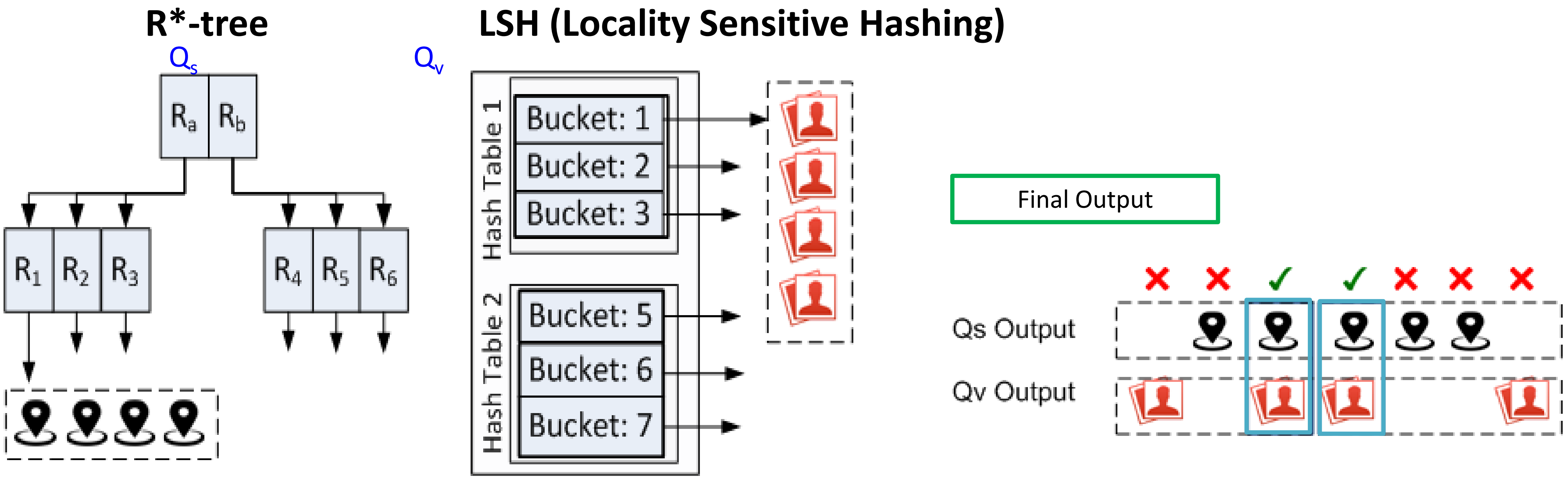


Query Image

Main Challenges with Spatial-Visual Search

- **Performance:**
  - Searching large volume datasets of geo-tagged images
- **Accuracy:**
  - Curse of dimensionality (high dimensional visual features)

Naive Indexes – Double Index (DI)



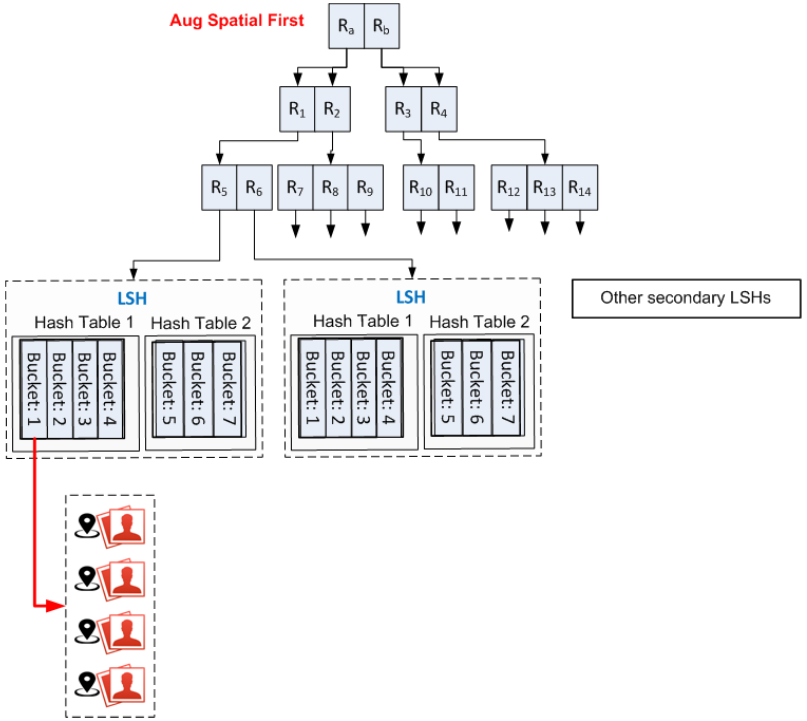
✗ Poor Performance: Execute query twice and intersect the results

Hybrid Index – Two Level

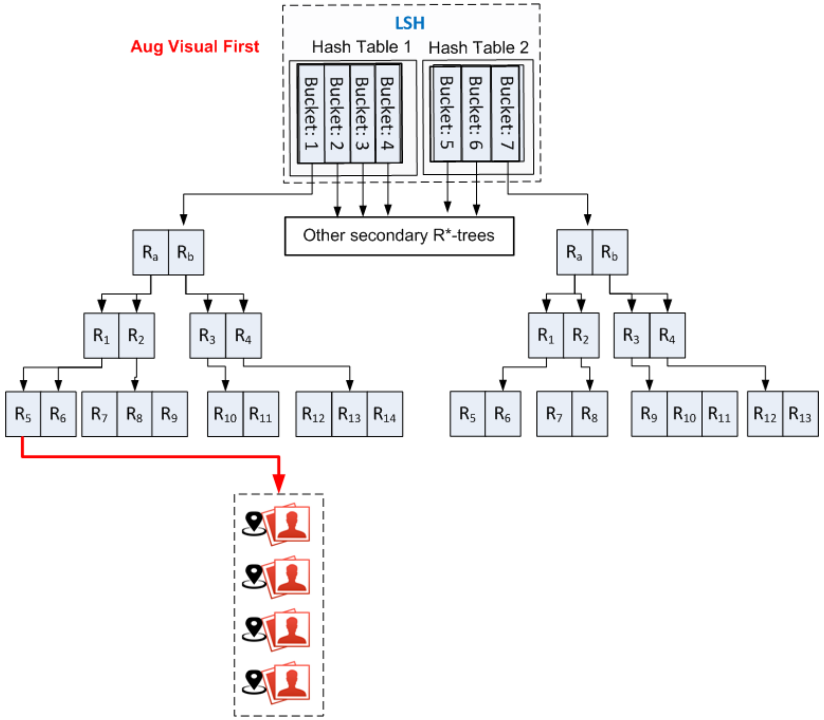
1) Augmented Spatial First Index (Aug SFI)

Primary Index

Secondary Indexes



2) Augmented Visual First Index (Aug VFI)



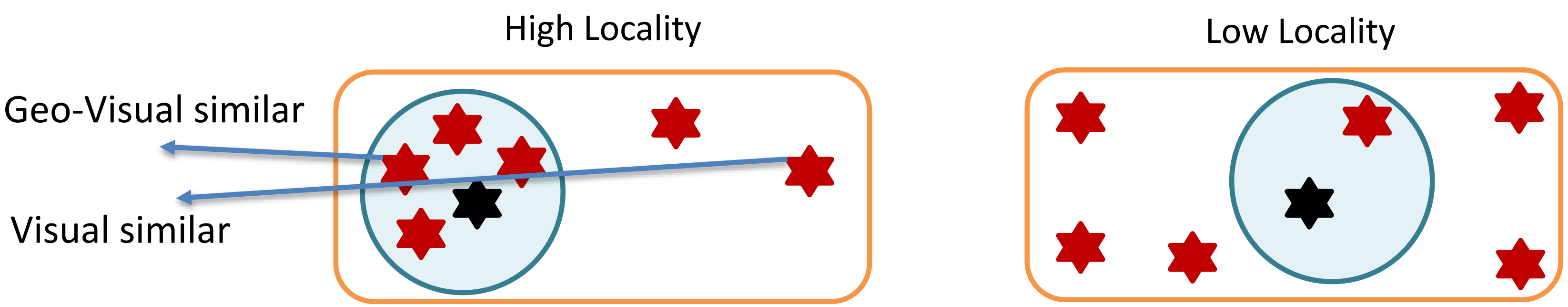
✓ Outperforms Double index

✗ Performance may suffer due to the bias towards spatial or visual dominance of the primary index

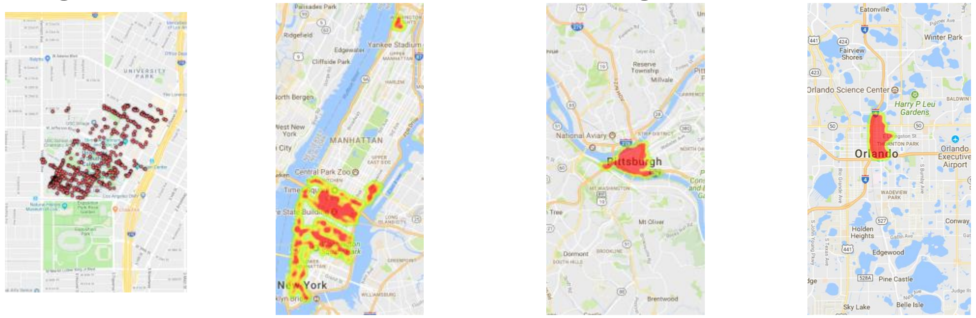
Observation: Locality of similar visual features of "street" images

Locality Analysis

Query Image   Similar Image   Spatial Range

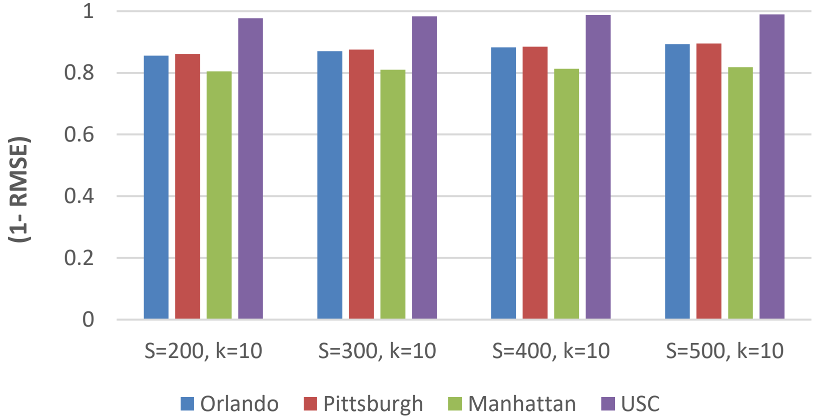


USC Neighborhood | **Manhattan, NY** | **Pittsburgh, PA** | **Orlando, FL**



Locality Analysis

Radius of Spatial Range (m) = {200, 300, 400, 500}, Visual Threshold (k)= 10.



Difference between Visual Similar Images and Geo-visual similar images of a query image

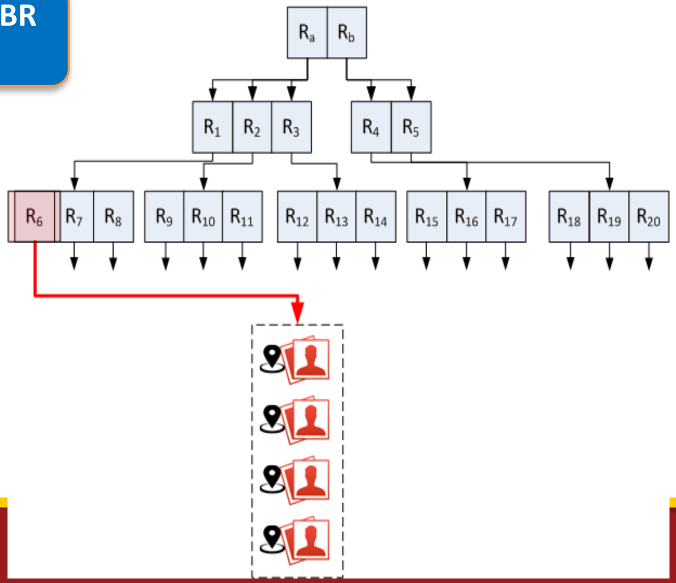
Nearby street images are also visually similar

Spatial-Visual Indexes using R*-tree **(Baselines)**

1) Spatial R*-tree Index (SRI)

- Organizes the dataset using "only" the **spatial** properties of the images.
- Each leaf node is augmented with both the spatial vector and visual vector of each image.

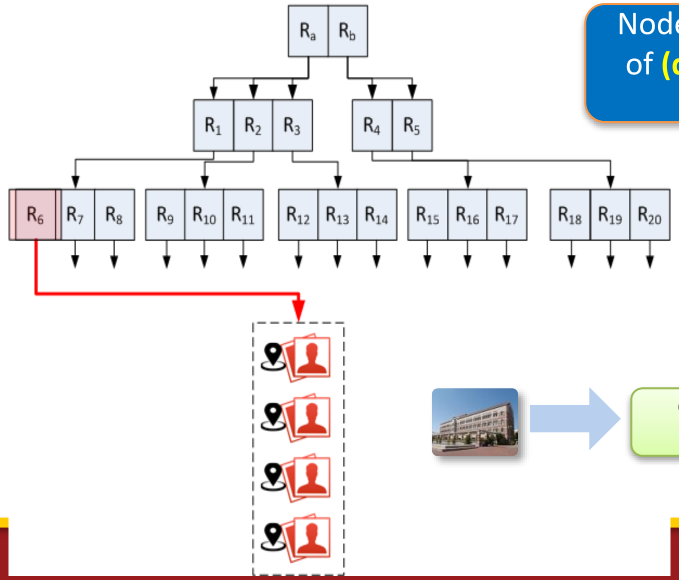
Nodes are associated with **MBR** of **spatial properties**

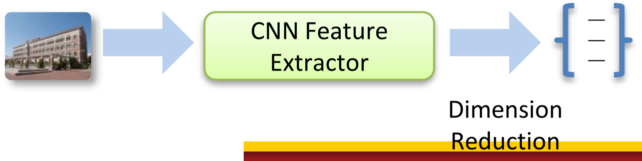


2) Visual R*-tree Index (VRI)

- Organizes the dataset using "only" the **visual** properties of the images.
- Each leaf node is augmented with both the spatial vector and visual vector of each image.

Nodes are associated with **MBR** of **(dimension-reduced) visual properties**

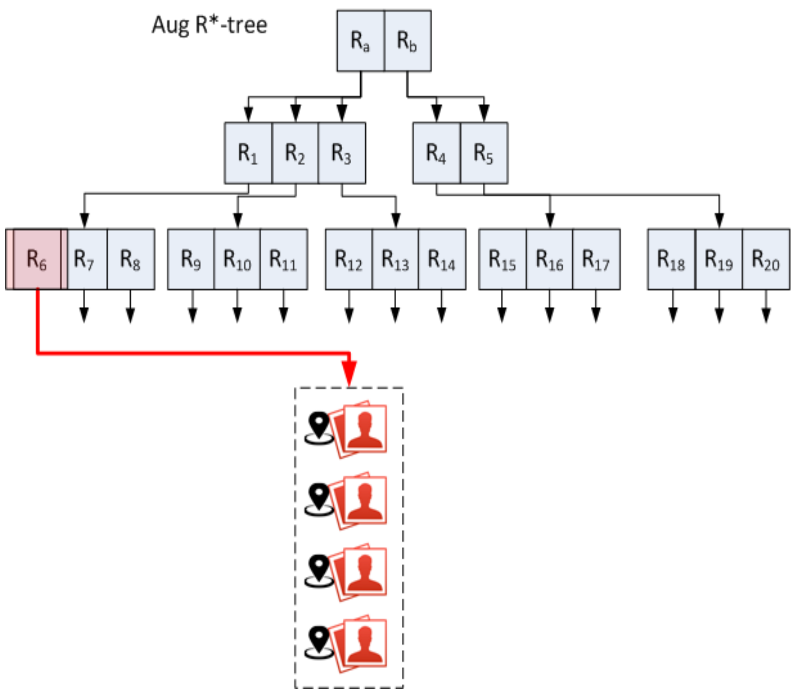




Hybrid Indexes - Plain Spatial-Visual R*-tree (PSV)

- Hybrid index structure which organizes images using their spatial and visual properties [7].

A node is associated with an **MBR** of both **spatial** and **(dimension-reduced) visual properties**



✓ Outperforms baselines

✗ PSV treats the spatial and visual properties of images equally; however, these are two different sets of features (might be treated differently)

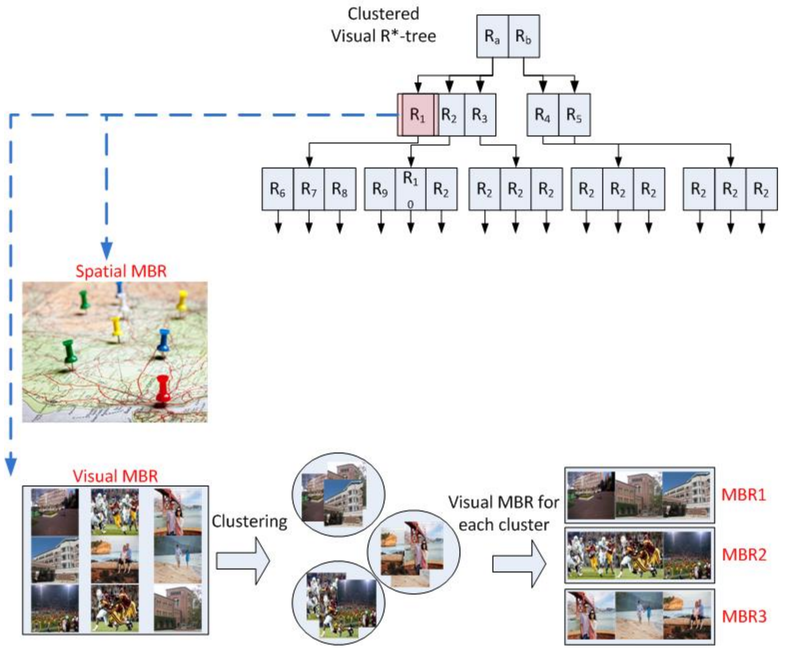
Hybrid Indexes: Adaptive Spatial-Visual R*-tree (ASV)

- Similar to PSV with the following changes:
  - Treat the spatial and visual properties *differently by creating spatial MBR and visual MBR for each node*.
  - Modify the underlying insert algorithm to accommodate the new design of each node. Hence, the goodness values used in the insert algorithm are modified to consider both MBRs.
    - $Margin = \alpha\ (margin_{spatial})/\max(margin_{spatial}) + \beta\ margin_{visual}/\max(margin_{visual})$
    - $Area = \alpha\ (area_{spatial})/\max(area_{spatial}) + \beta\ area_{visual}/\max(area_{visual})$
    - $Ovarlep = \alpha\ (overlap_{spatial})/\max(area_{spatial}) + \beta\ overlap_{visual}/\max(area_{visual})$

$$\left\{\begin{array}{c} \alpha = 1, \beta = 0 \rightarrow Spatial\ R*-tree\ Index \\ \alpha = 0, \beta = 1 \rightarrow Visual\ R*-tree\ Index \\ Otherwise \rightarrow Spatial - Visual\ Index \end{array}\right\}$$

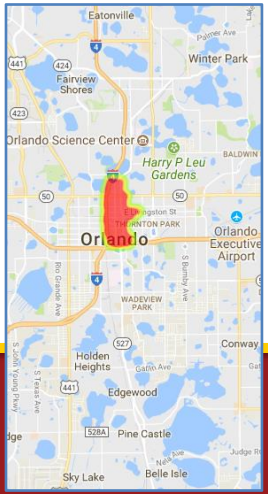
Hybrid Indexes: Clustered ASV R*-tree (CSV)

- ASV organizes a dataset by considering simultaneously the spatial and visual sub-division for the global area.
  - ✗ The visual MBR in a node can loosely represent the contained images.
- After ASV is constructed, *<u>for each node we can cluster the contained images</u>* (using k-means) and create a set of visual MBRs.
- While searching the tree, for each node
  - – The Spatial MBR is used to prune the search space of a query spatially.
  - – The bundle of Visual MBRs are used for pruning the search space of a query visually.

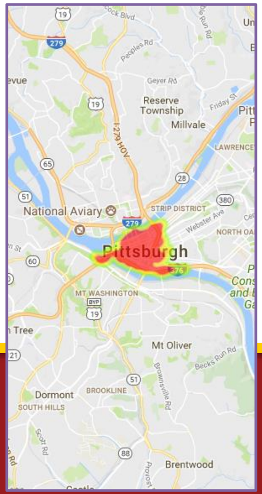


Experiments -- Geo-tagged Image Datasets

	Dataset	# of images	Size of Spatial Descriptors (MB)	Size of Visual Descriptors (MB)	Spatial Region (W * H) (km^2)	Spatial Density (# of images per km^2)
Orlando Downtown	OR	3,204	1	20	2.1 * 1.2	1,271
Pittsburgh Downtown	PT	4,825	1	30	1.5 * 3.6	893
Manhattan	MA	17,825	2	106	13.7 * 7.3	178
USC	LA	24,345	3	140	1.4 * 1.0	17,398
San Francisco	SF	520,623	51	3,607	6.0 * 8.1	10,712

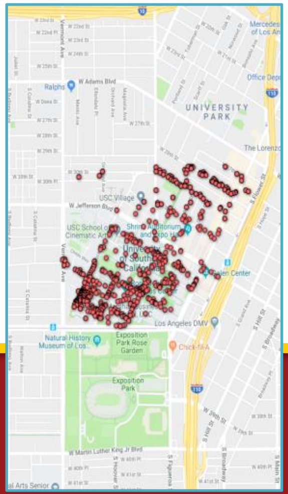


Orlando Downtown



Pittsburgh Downtown

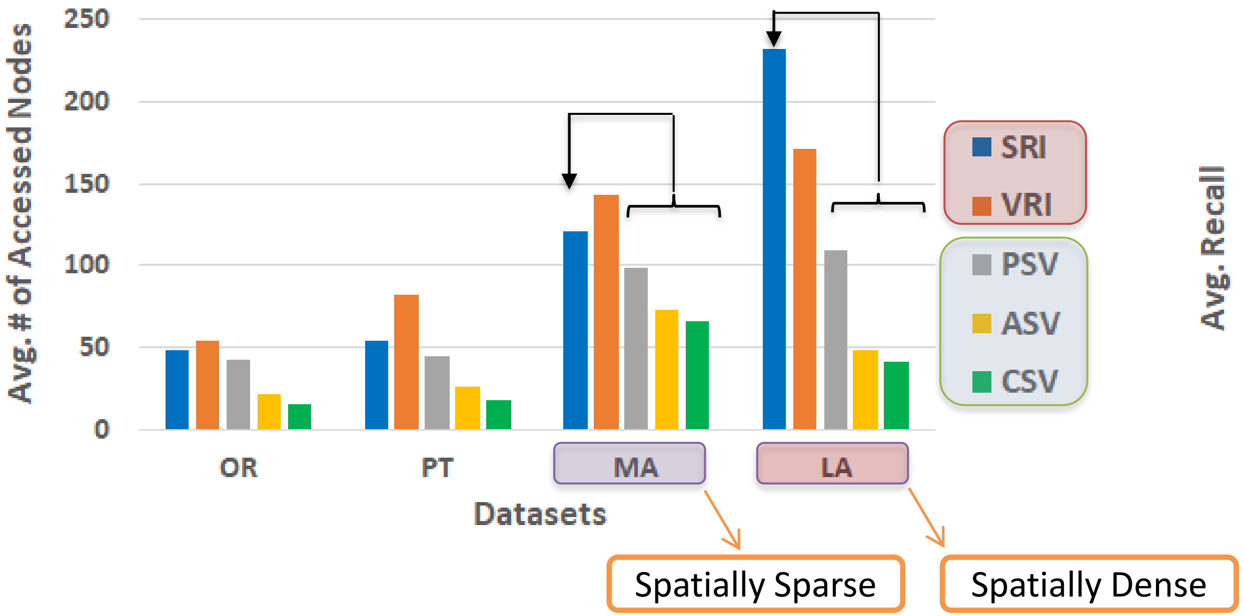
Manhattan



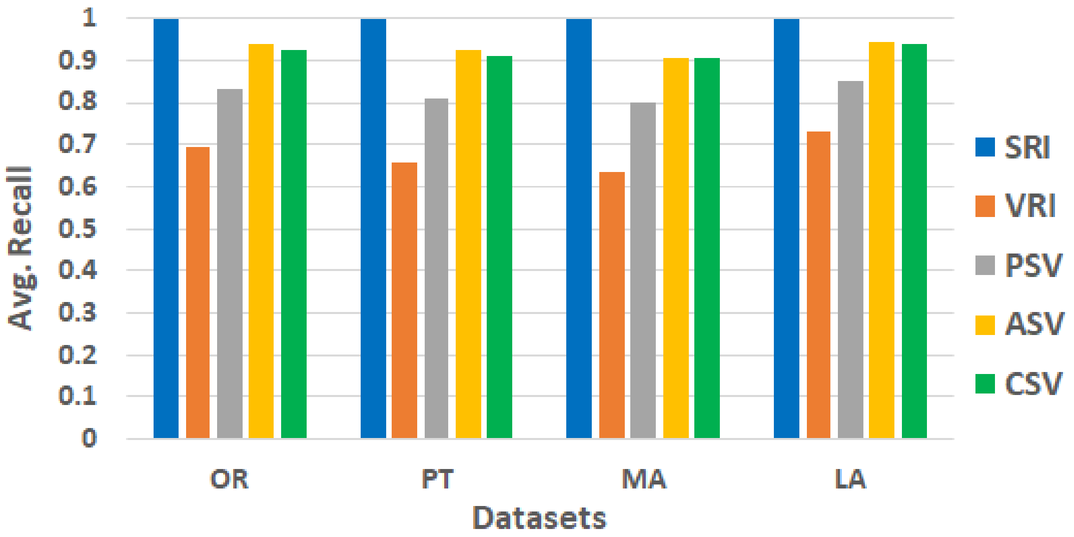
LA (USC)

Baseline vs. Hybrid

Efficiency



Effectiveness



Using MA With respect to SRI, the speedup factor of PSV, ASV, and CSV reached up to 1.2x, 1.6x, and 1.8x.

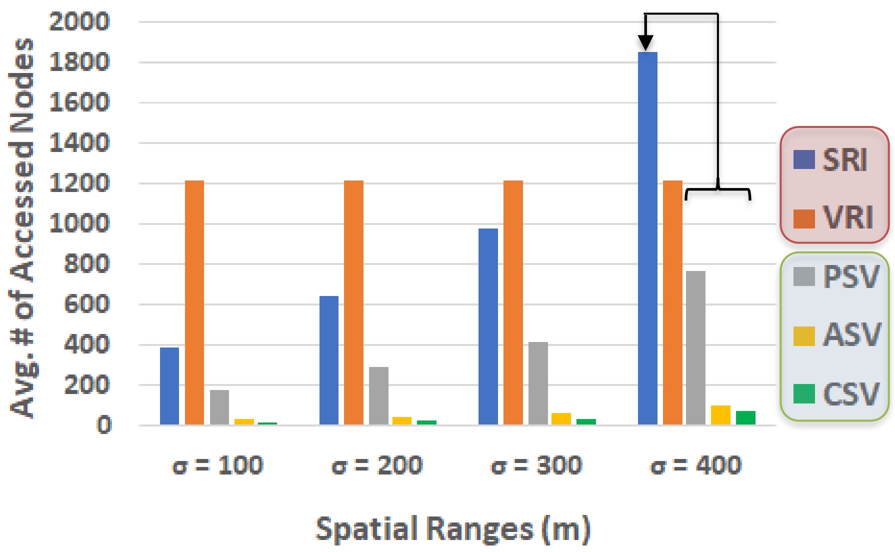
Using LA With respect to SRI, the speedup factor of PSV, ASV, and CSV reached up to 2.1x, 4.8x, and 5.6x.

SRI achieves a perfect recall score

Among Hybrid, recall of ASV and CSV is better than PSV

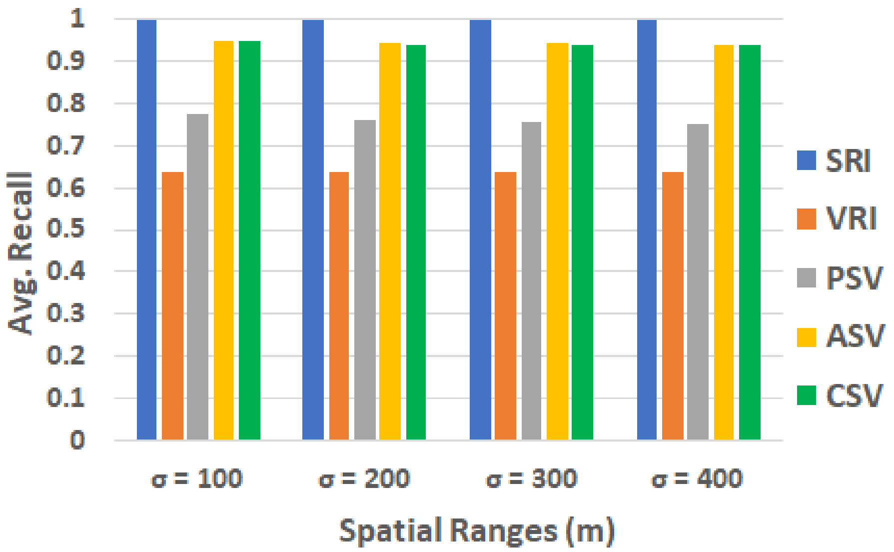
Scalability: Evaluation on a large-scale dataset

Efficiency



With respect to SRI, the speedup factor of ASV, and CSV reached up to 18x and 25x, respectively.

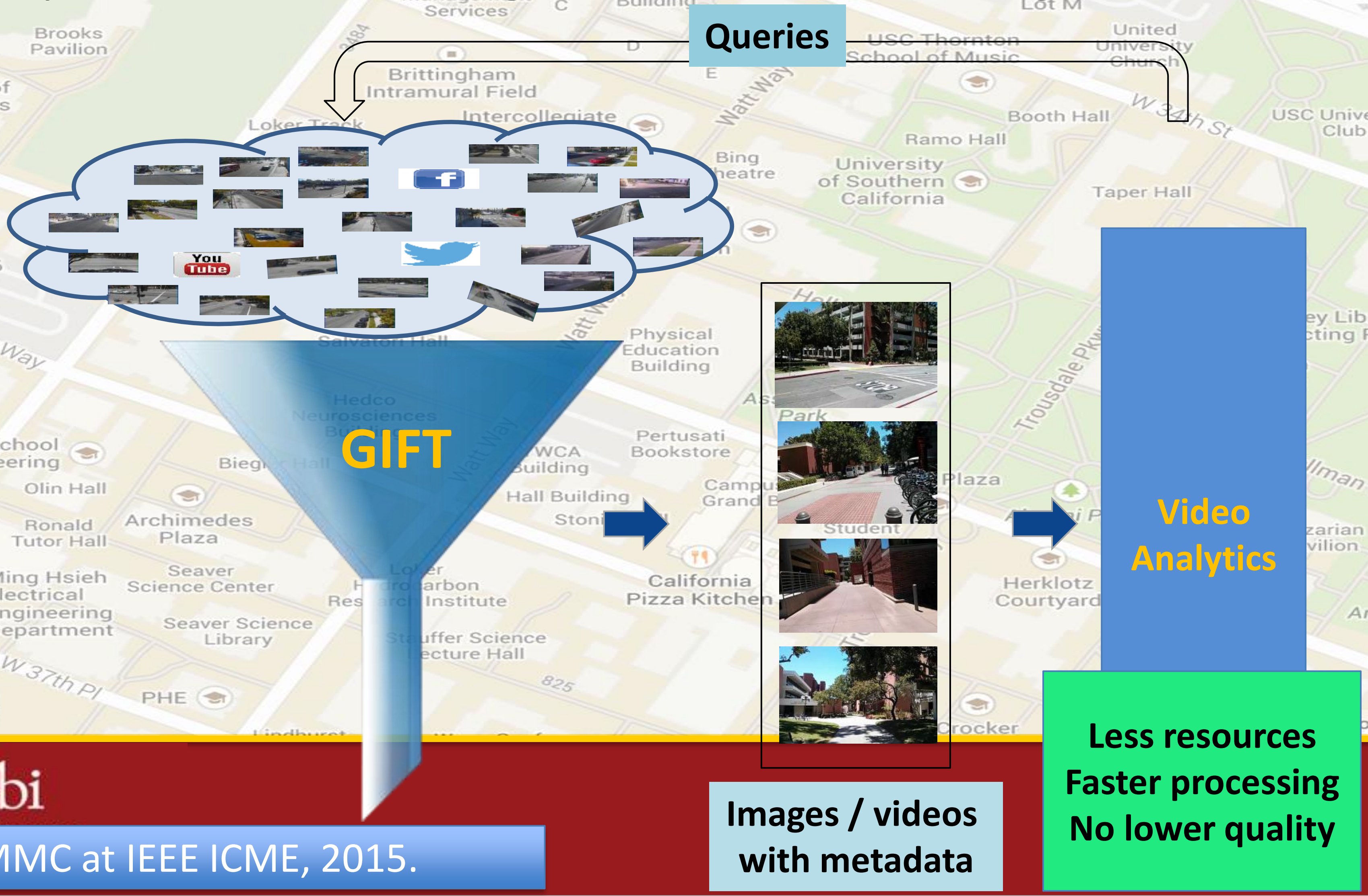
Effectiveness



Filtering for Computer Vision Applications

Geospatial Image and Video Filtering Tool (GIFT) [8]

- ➢An efficient tool to **organize, index and search** spatio-temporal image/video data.
- ➢A fast way to select related image/video frames according to user demands.



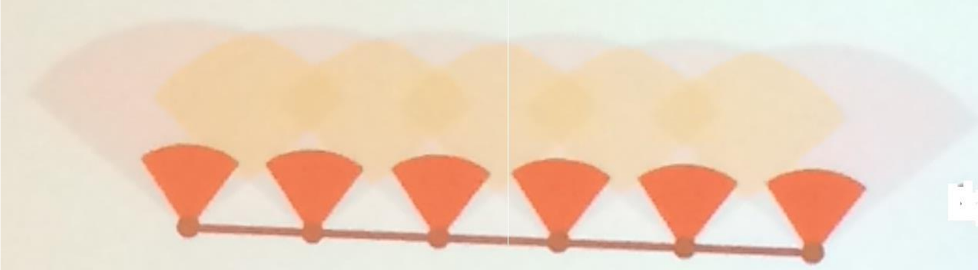
[8] MMC at IEEE ICME, 2015.

Automatic Generation of Panoramic Images

Panorama Problem

Point panorama

Route panorama

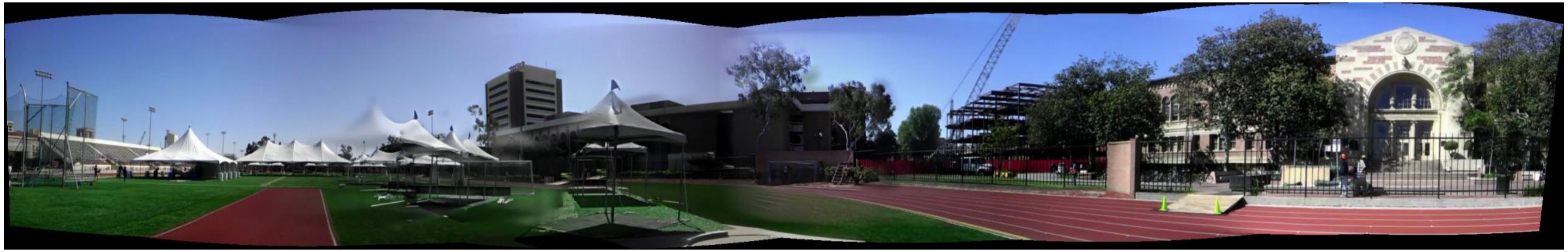


Small number of well selected input images would be fine!

How to automatically select the minimum number of image frames for panorama generation?

Example: Point Panorama Results [9]

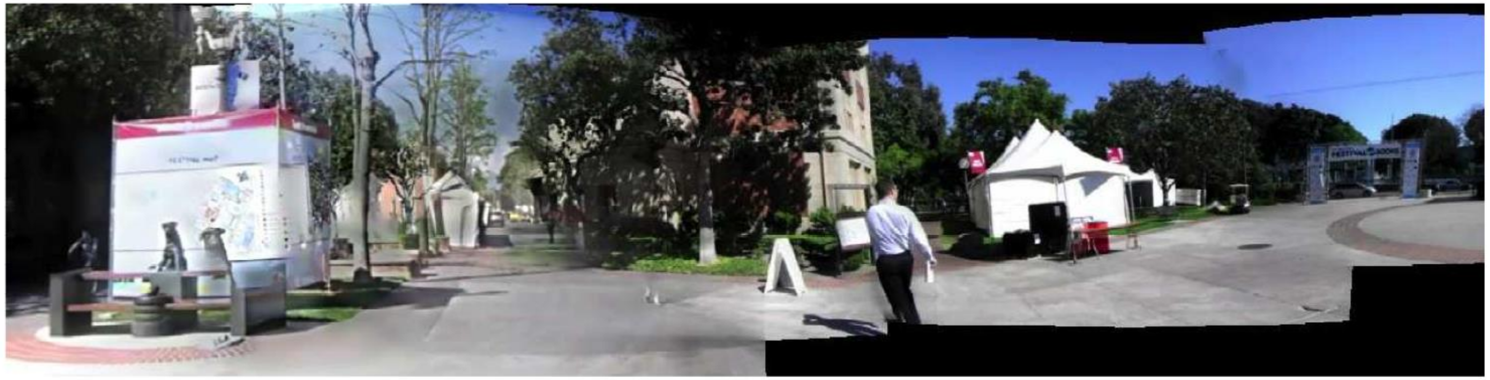
BA-P: Selected FOV# = 228, Video #= 3, Stitching time = 148.5 seconds



DA-P: Selected FOV# = 17, Video #= 2, Stitching time = 8.51seconds

DLA-P: Selected FOV# = 13, Video #= 3, Stitching time = 8.65seconds

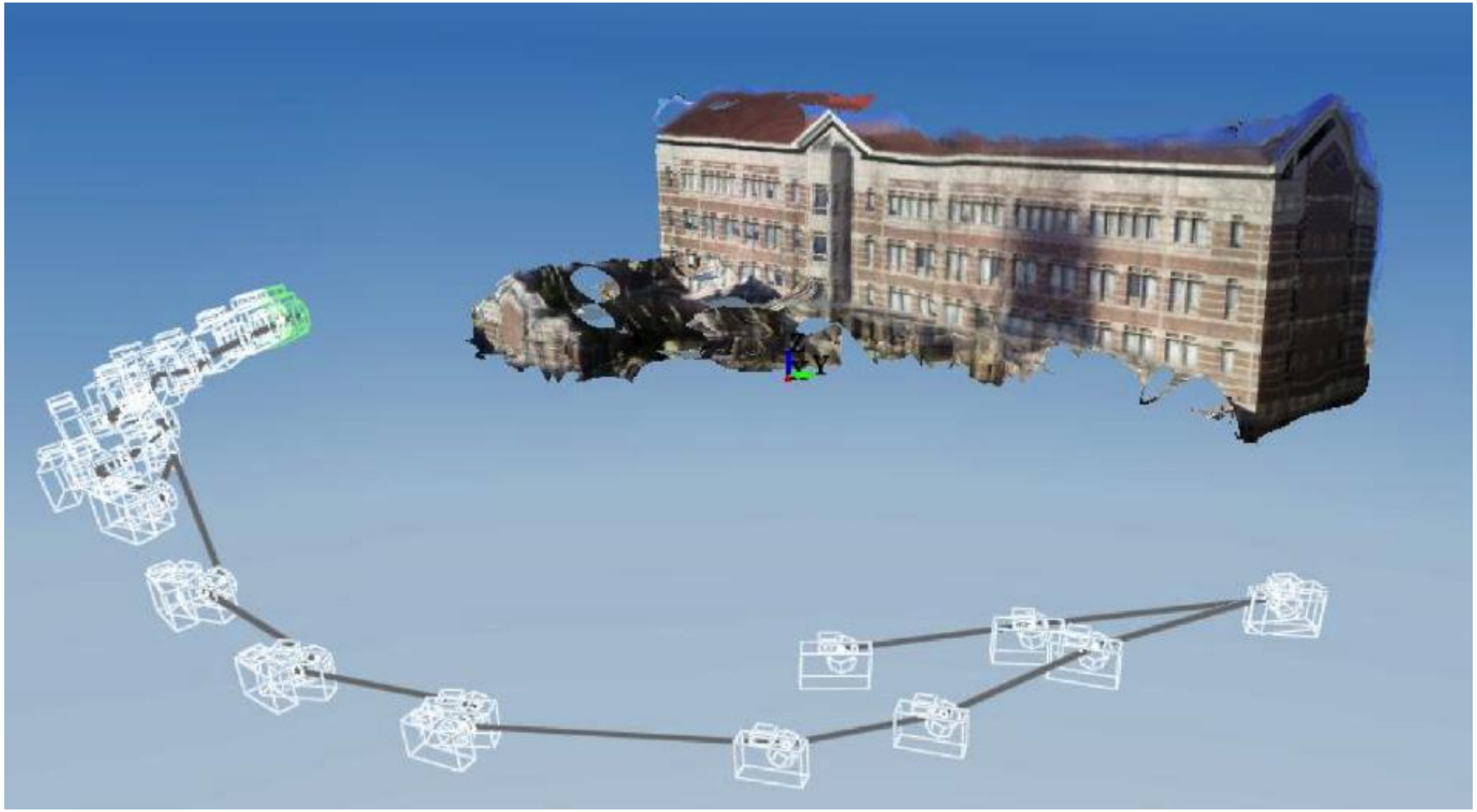
(a) Algorithm *BA-P*, SelectedFOV# = 77, Video# = 4.



(b) Algorithm *DA-P*, SelectedFOV# = 14, Video# = 4.

(c) Algorithm *DLA-P*, SelectedFOV# = 14, Video# = 2.

Automatic Generation of 3D Models [10]



[10] IEEE ICME, 2014.

Image Machine Learning with Edge Computing

Image Machine Learning Example

- In collaboration with the Sanitation Department of Los Angeles, monitor LA streets for cleanliness using images.
- Currently, data are manually collected and evaluated: inefficient, costly →automate!
- Goal: *automatically detect the cleanliness of streets as well as any special objects in need of removal.*

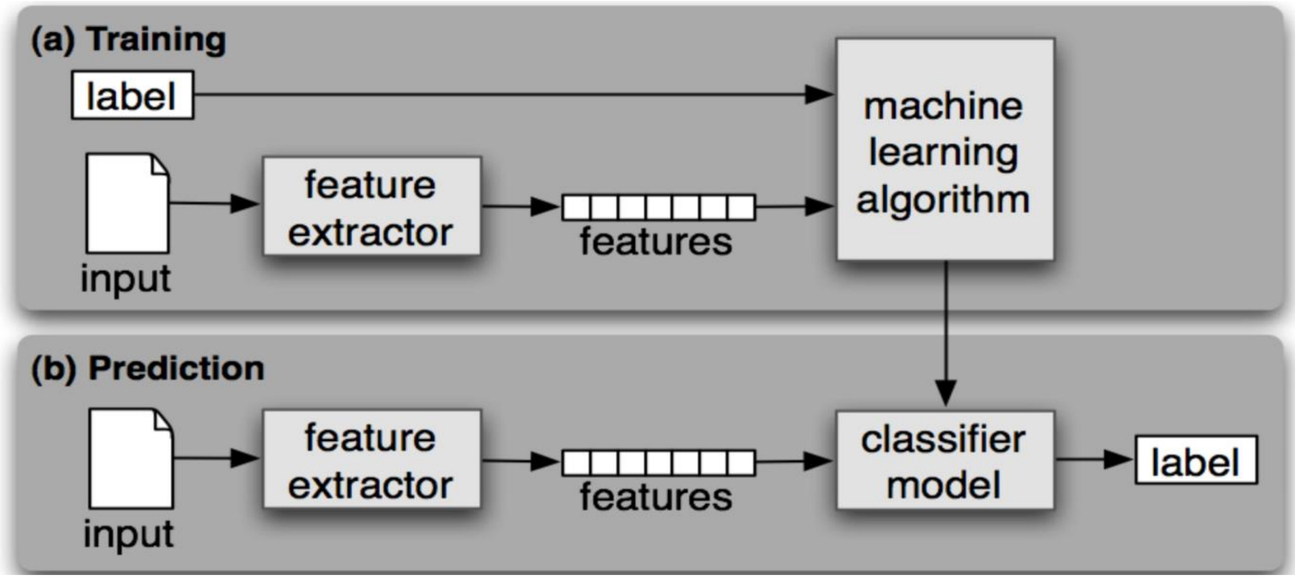
Image Classification of Street Cleanliness

Image Label	Description	Examples	Image Label	Description	Examples
Bulky Item – Few	1 to 3 items (e.g., coach, desk, mattress, and tire) are thrown on the street.		**Encampment**	A tent for people who live in streets.	
Bulky Item – Many	More than 3 items are thrown on the street.		**Overgrown Vegetation**	There is extra vegetation on the streets.	
Illegal Dumping	There is an area which is full of waste which needs special equipment to remove.		**Clean**	The street is clean ☺	

20K+ Images collected by the Sanitation Department, City of Los Angeles

Image Based Classifier

- Achieved 80 – 90% accuracy (depending on class): stable and practical
- The more images in an area, the higher the accuracy becomes: promising



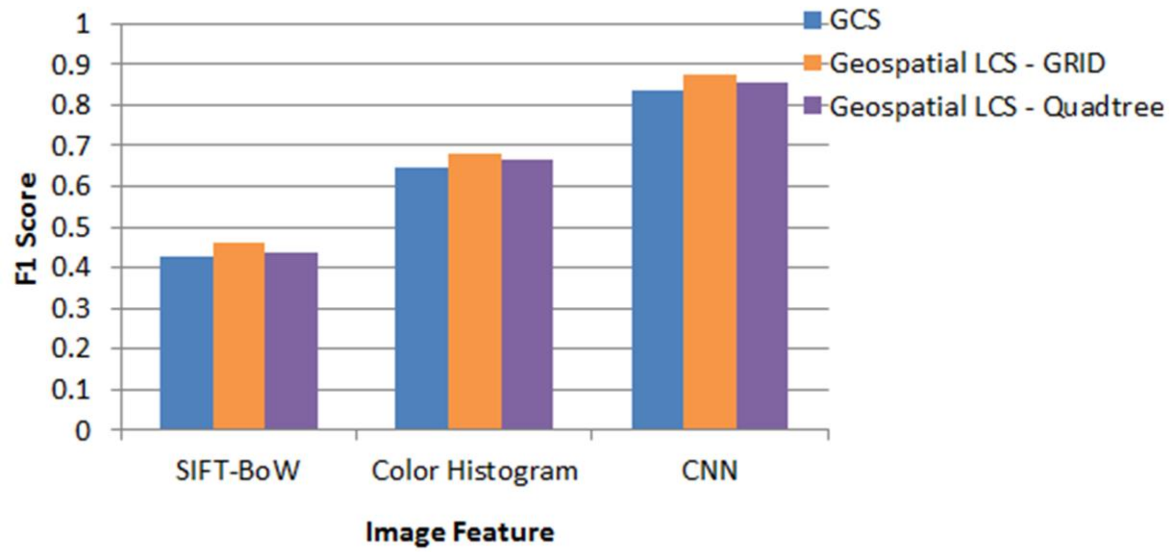
[11] IEEE Big Multimedia, 2018

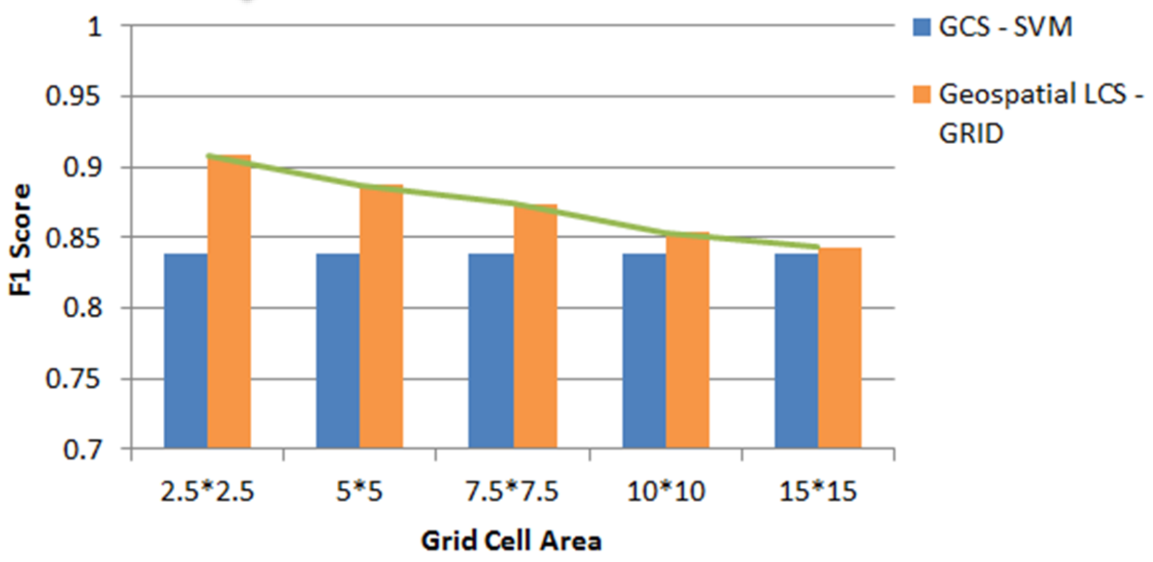
Global vs. Local Classifier

- Global Classification Scheme (GCS)
  - This approach constructs one single trained model that learns the image features throughout the overall geographical region in a dataset.
  - Street scenes have visual differences across geographical regions → Classification accuracy decreases
- Geo-spatial Local Classification Scheme (LCS)
  - Utilizing the geo-properties tagged with the images
  - Partition the overall geographical area into sub-regions using Grid or Bucket Quadtree.
  - For every sub-region, construct a local trained model.

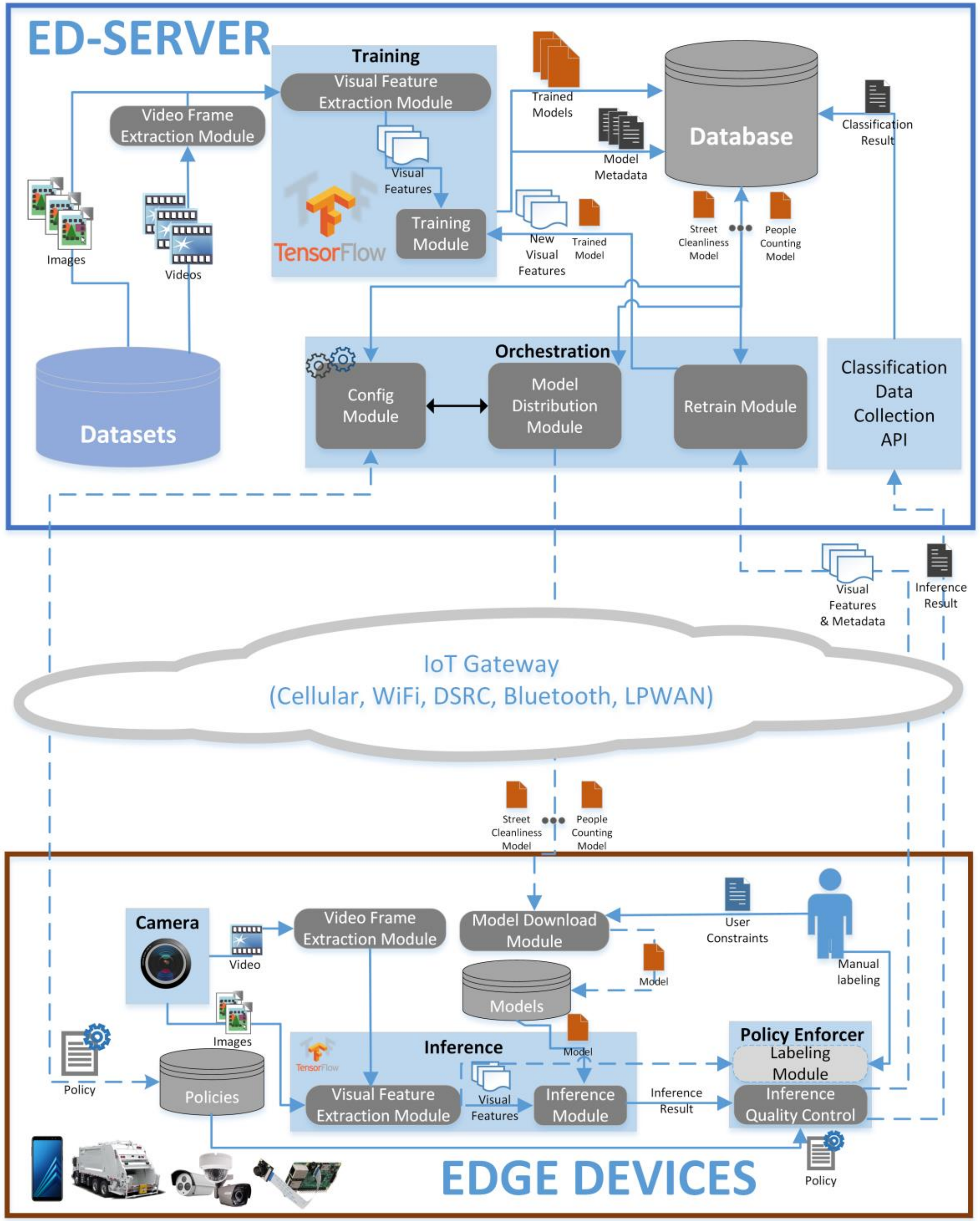
Global vs. Local Classifier

Smaller Cell → Higher Score





How about supporting city scale image learning? → Not a centralized system



Edge Computing

- Train machine learning models on the server (initial model)
- Distribute the models to edge devices (e.g., smartphones, smart cameras)
- Inference happens on edge devices using CPU on edge
- Report selected results to involved agencies
- Improve models iteratively

Framework to **train, distribute** and **adapt** models

[12] IEEE BigMM, Sep. 2019

Image Learning with Edge Devices

- Provide various model "flavors" of the same classification task
  - Choose the one that fits the application requirements and device capabilities → Resource based model building and dissemination
- Save bandwidth by reducing the amount of transmitted data
  - Resize image: smaller size of original image
  - Extract visual feature vectors on device and transfer
- Improve model by selecting images for retraining at *new locations* and time periods

Experiments

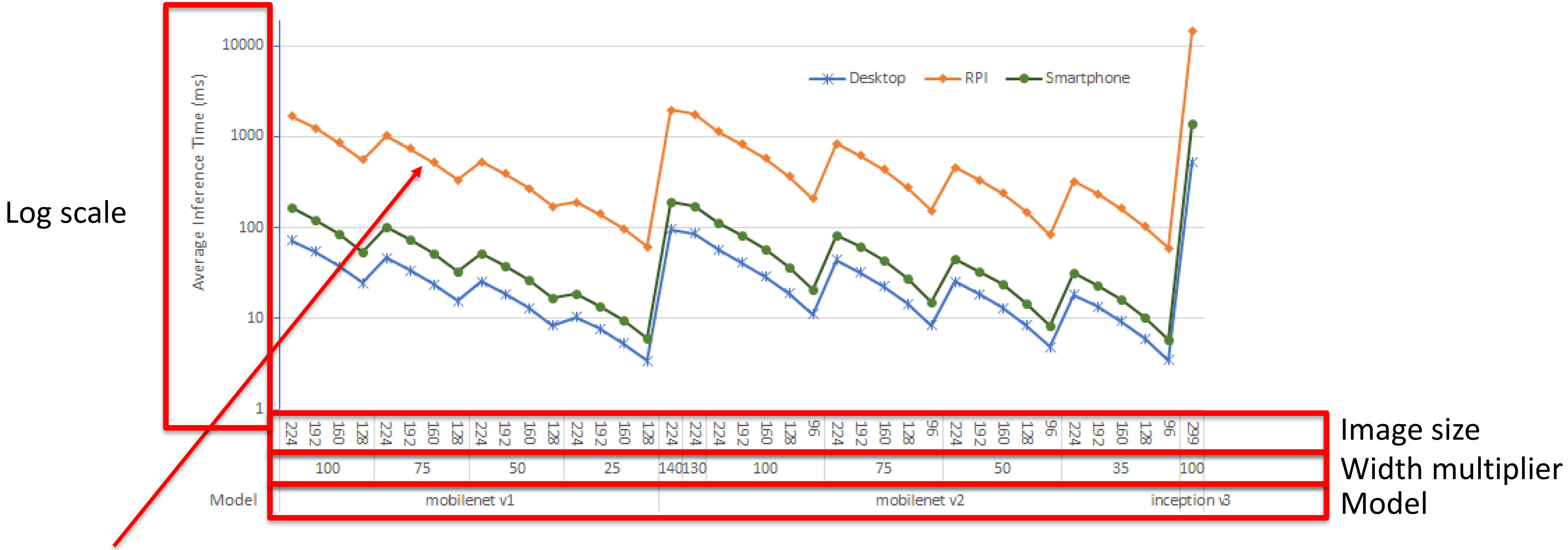
- Show the **trade-off** between inference accuracy and the resource constraints of the edge devices
  - models can be tuned to support wide classes of edge devices
- Three classes of edge devices

Device	Class	CPU	GPU	Memory
Raspberry Pi 3B+	Low	Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4 GHz	Broadcom VideoCore IV	1GB LPDDR2 SDRAM
Google Pixel 3	Medium	Qualcomm Snapdragon 845 8x Qualcomm® Kryo™ 385 CPU 64-bit @ 2.8 GHz	Qualcomm® Adreno™ 630 GPU	4GB
Desktop	High	56 (Intel(R) Xeon(R) Gold 5120 CPU @ 2.20GHz	2 Tesla P100-SXM2-16GB	187 GB

Experiments

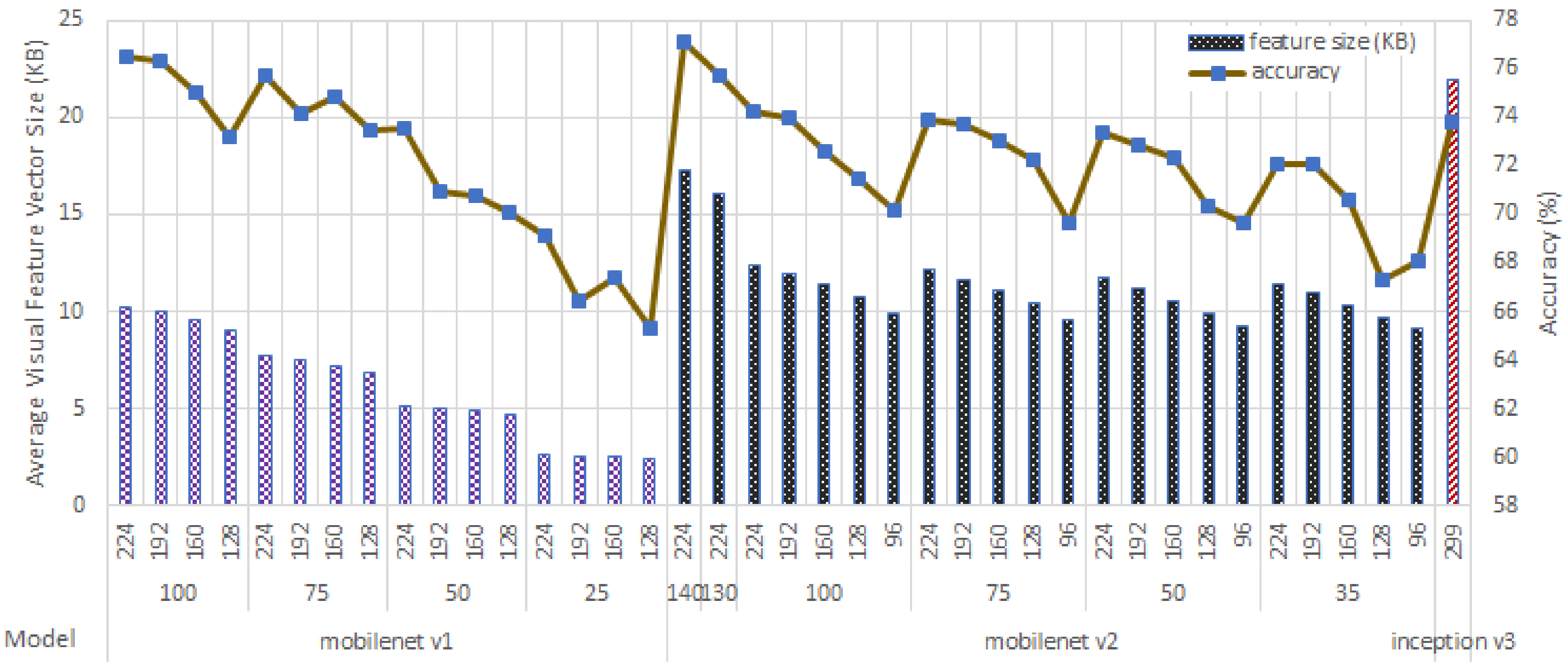
- **Datasets**:
  - D^{SC} geotagged labeled LASAN image collection for cleanliness classification
    - 42,331 images with 5 labels: 14,495 bulky items, 7,120 illegal dumping, 7,007 encampment, 6,982 overgrown vegetation, and 6,727 clean
  - D^{CAL256}: Caltech 256
    - 30,608 images with 256 labels, with a minimum of 80 images per label and 119 on average
- **Three pre-trained models**:
  - Inception V3, MobileNet V1, and MobileNet V2
  - Used transfer learning

Inference Time vs Model



- Raspberry Pi is 1.5x order of magnitude slower compared to desktop class devices

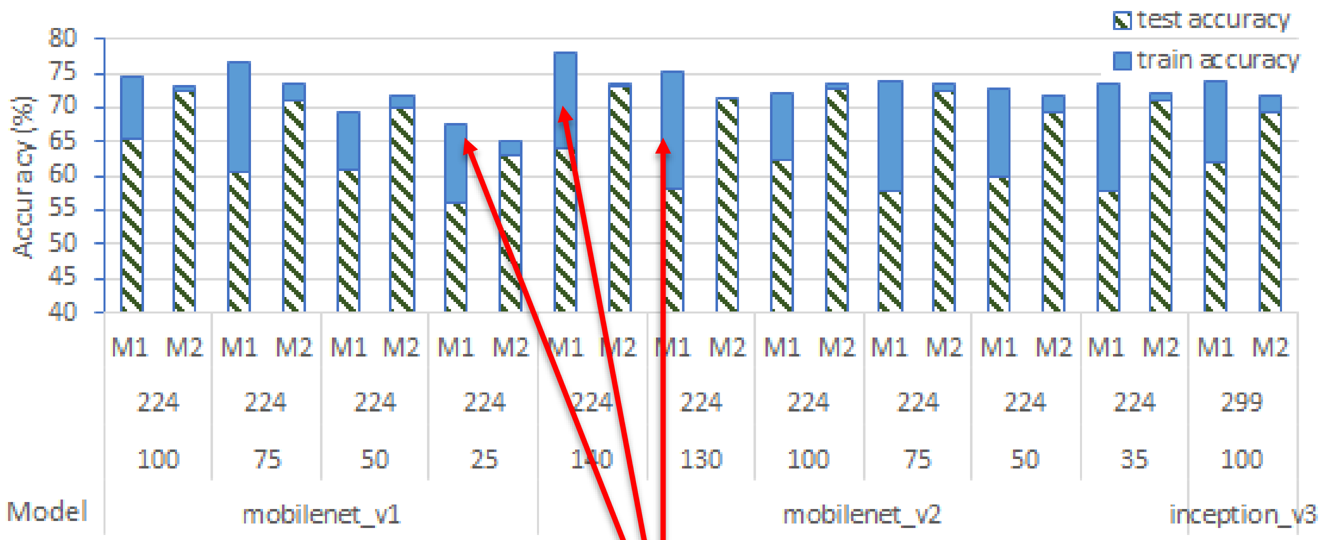
Feature Size vs Accuracy



- Usually, the larger the size of the VFVs, the higher the accuracy
  - they carry a more detailed summary of the image

Location-based Feature Selection

- **M1:** Excluded images from Downtown LA
- **M2:** Includes 50% of images from Downtown LA
- Accuracy tested on 50% images of unseen data in Downtown LA



- Under-represented regions significantly affect the accuracy
  - Sometimes with almost 15% drop of accuracy

Outline

- Motivation
- Modeling Spatial Property of Visual Content
  - Point Location
  - FOV Model
  - Image Scene Location
- Harnessing Spatial Property in Data Management
  - Spatial Coverage Measurement
  - Efficient Data Collection
  - Access Method
  - Image Machine Learning with Edge Computing
- **Conclusion**

Conclusion

- Provide an overview of 1) modeling spatial properties of visual data, and 2) various ways to harness spatial properties in visual data management and machine learning with examples.
- Spatial metadata are getting more important in many visual data applications including image machine learning.
- Proper consideration of spatial metadata would be useful in many visual data applications, especially where geographical information is critical.

References

[1] VDMS: Efficient Big-Visual-Data Access for Machine Learning Workloads. Luis Remis, Vishakha Gupta-Cledat, Christina R. Strong, Ragaad Altarawneh. Intel Labs. Workshop on Systems for Machine Learning and Open Source Software at NIPS, 2018.

[2] Sakire Arslan Ay, Roger Zimmermann, Seon Ho Kim. Viewable Scene Modeling for Geospatial Video Search. ACM Multimedia Conference (ACM MM 2008), pp. 309-318, Oct. 2008.

[3] Abdullah Alfarrarjeh, Zeyu Ma, Seon Ho Kim and Cyrus Shahabi. 3D Spatial Coverage Measurement of Aerial Images. 26th International Conference on Multimedia Modeling (MMM 2020), Jan. 2020.

[4] Abdullah Alfarrarjeh, Seon Ho Kim, Shivnesh Rajan, Akshay Deshmukh, and Cyrus Shahabi. A Data-Centric Approach for Image Scene Localization. IEEE International Conference on Big Data, 2018.

[5] Abdullah Alfarrarjeh, Seon Ho Kim, Akshay Deshmukh, Shivnesh Rajan, Ying Lu, Cyrus Shahabi. Spatial Coverage Measurement of Geo-Tagged Visual Data: A Database Approach. IEEE International Conference on Big Multimedia (BigMM 2018), Sep. 2018. (Winner of the best student paper award)

References

[6] Hien To, Seon Ho Kim, Cyrus Shahabi. Effectively Crowdsourcing the Acquisition and Analysis of Visual Data for Disaster Response. IEEE International Conference on Big Data (IEEE Big Data 2015), 2015.

[7] Abdullah Alfarrajeh, Seon Ho Kim, Cyrus Shahabi. Hybrid Indexes for Spatial-Visual Search. ACM International Conference on Multimedia (ACM MM), Oct. 2017.

[8] Abdullah Alfarrarjeh, et al. A Class of R*-tree Indexes for Spatial-Visual Search of Geo-tagged Street Images. 36th International Conference on Data Engineering (IEEE ICDE 2020), short paper, April 2020.

[9] Seon Ho Kim, Ying Lu, Junyuan Shi, Abdullah Alfarrarjeh, Cyrus Shahabi, Guanfeng Wang, Roger Zimmermann. Key Frame Selection Algorithms for Automatic Generation of Panoramic Images from Crowdsourced Geo-tagged Videos. Web and Wireless Geographical Information Systems (W2GIS), 2014.

[10] Guanfeng Wang, Ying Lu, Luming Zhang, Abdullah Alfarrarjeh, Roger Zimmermann, Seon Ho Kim and Cyrus Shahabi. Active Key Frame Selection for 3D Model Reconstruction from Crowdsourced Geo-tagged Videos. IEEE International Conference on Multimedia and Expo (ICME), pp. 1-6, July 2014.

References

[11] Abdullah Alfarrarjeh, Seon Ho Kim, Sumeet Agrawal, Meghana Ashok, Su Young Kim, Cyrus Shahabi. Image Classification to Determine the Level of Street Cleanliness: A Case Study. IEEE International Conference on Big Multimedia (BigMM 2018), Sep. 2018.

[12] Giorgos Constantinou, Abdullah Alfarrarjeh, Seon Ho Kim, Gowri Sankar Ramachandran, Bhaskar Krishnamachari, Cyrus Shahabi. A Crowd-based Image Learning Framework using Edge Computing for Smart City Applications. IEEE International Conference on Big Multimedia (BigMM '19), Sep. 2019.

Thank you!
Q & A

Seon Ho Kim, Ph.D.
University of Southern California, CA, USA
seonkim@usc.edu