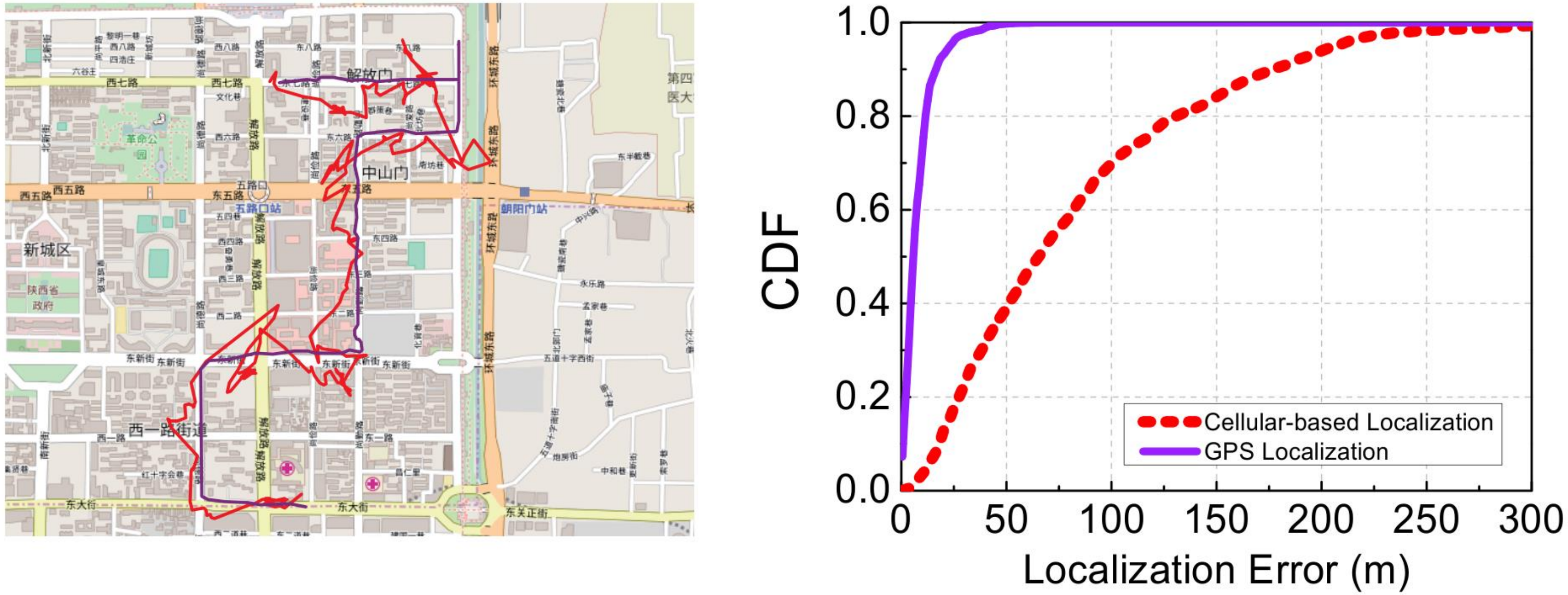
CLSTERS: A General System for Reducing Errors of Trajectories Under Challenging Localization Situations

HAO WU, Fudan University & Shanghai Key Laboratory of Data Science
WEIWEI SUN, Fudan University & Shanghai Key Laboratory of Data Science
BAIHUA ZHENG, Singapore Management University
LI YANG, HUAWEI Shanghai Research & Development Center
WEI ZHOU, HUAWEI Shanghai Research & Development Center

Presenter：Hao Wu
wuhao5688@fudan.edu.cn

GPS trajectory vs. cellular-based trajectory

- GPS: ~10m
- Cellular: several hundred meters



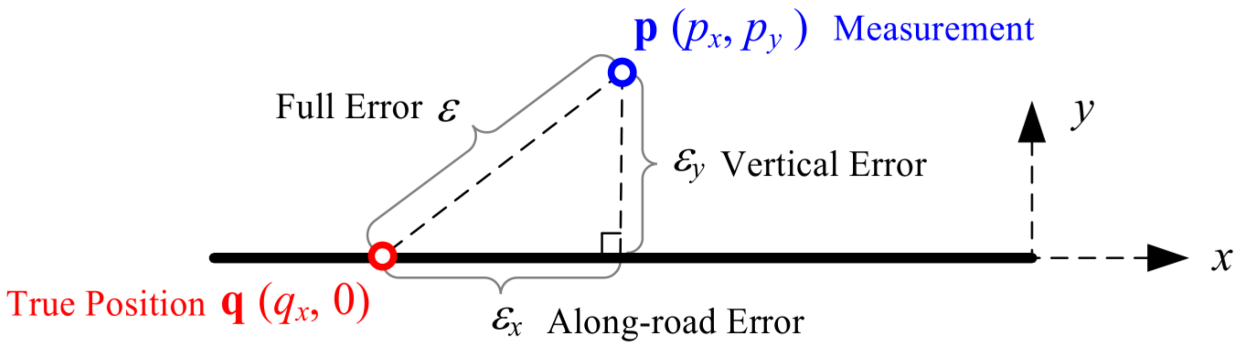
(a) Example of cellular-based trajectory

(b) CDF

Fig. 1. Visualization and statistics of a GPS-based trajectory vs. a cellular-based trajectory.

Existing works

- Directly reduce the error for cellular localization
  - Depend on the hardware environment (e.g., GSM, UMTS, LTE)
  - Need more information besides the (x, y, t) (e.g., RSSI, Cell-ID, TOA)
- Utilize the road network information (map matching-based)
  - Designed for low error trajectories (i.e., GPS trajectory)
  - Can not reduce the *along-road error*.



What we want to do

- Design a general framework
  - Specifically for those trajectories with large error scales
  - Only requires the (x,y,t) information which makes it independent on the hardware environment

What we want to do

- Design a general framework
  - Specifically for those trajectories with large error scales
  - Only requires the (x,y,t) information which makes it independent on the hardware environment

- Problem Formalization
  - Given a noisy trajectory $\mathcal{T} = \{p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_N\}$, calibrate the trajectory to $\tilde{\mathcal{T}} = \{\tilde{p}_1 \rightarrow \tilde{p}_2 \rightarrow \cdots \rightarrow \tilde{p}_N\}$, with minimizing the average error

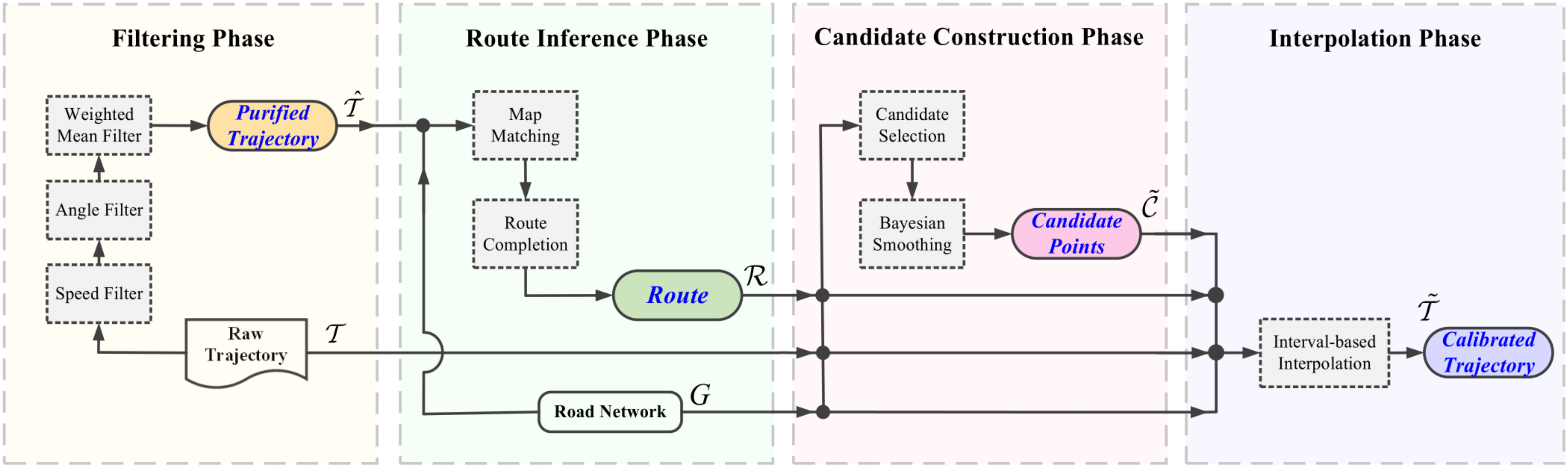
$$\varepsilon_{\tilde{\mathcal{T}}} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{\left(\tilde{p}_x^{(i)} - q_x^{(i)}\right)^2 + \left(\tilde{p}_y^{(i)} - q_y^{(i)}\right)^2}$$

where the ground truth is $\mathcal{T}_{gt} = \{q_1 \rightarrow q_2 \rightarrow \cdots \rightarrow q_N\}$

$$p_i = \{p_x^{(i)}, p_y^{(i)}, t^{(i)}\}$$

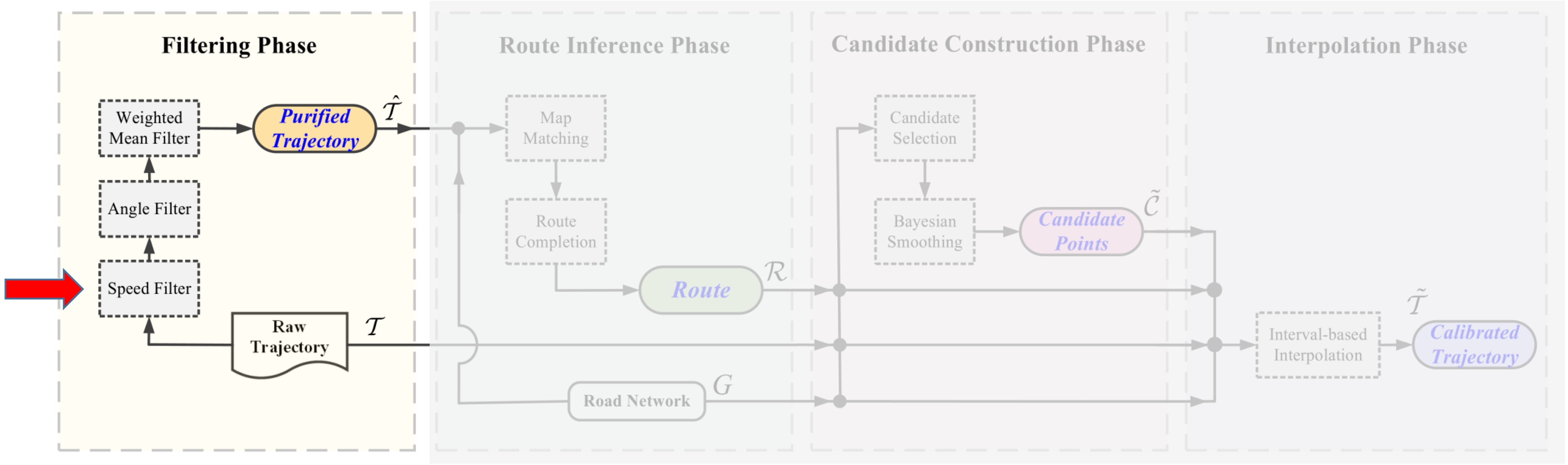
System overview

- CLSTERS - Challenging Localization Situation-aimed Trajectory Error Reduction System



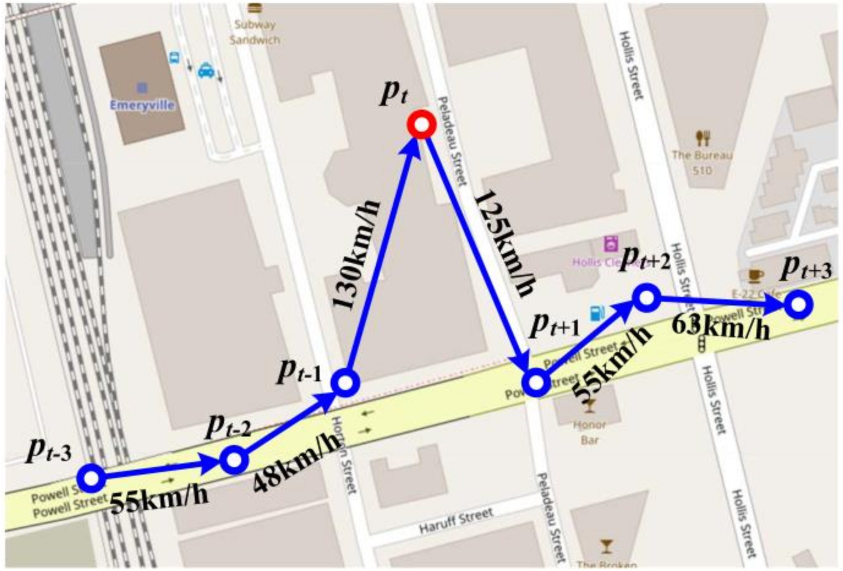
Phase 1: Filtering Phase

- Purify the raw noisy trajectory for the later map matching phase

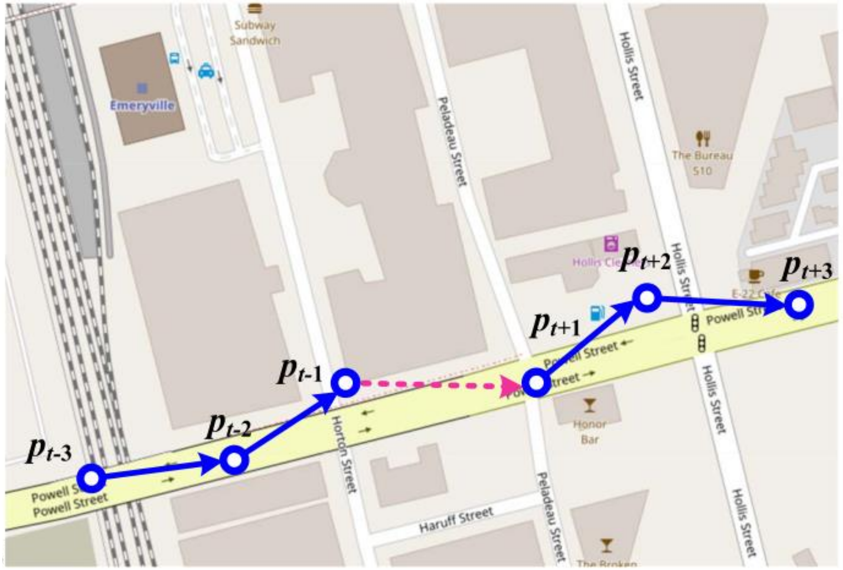


Speed Filter

- Filter the points having the irregular immediate speeds



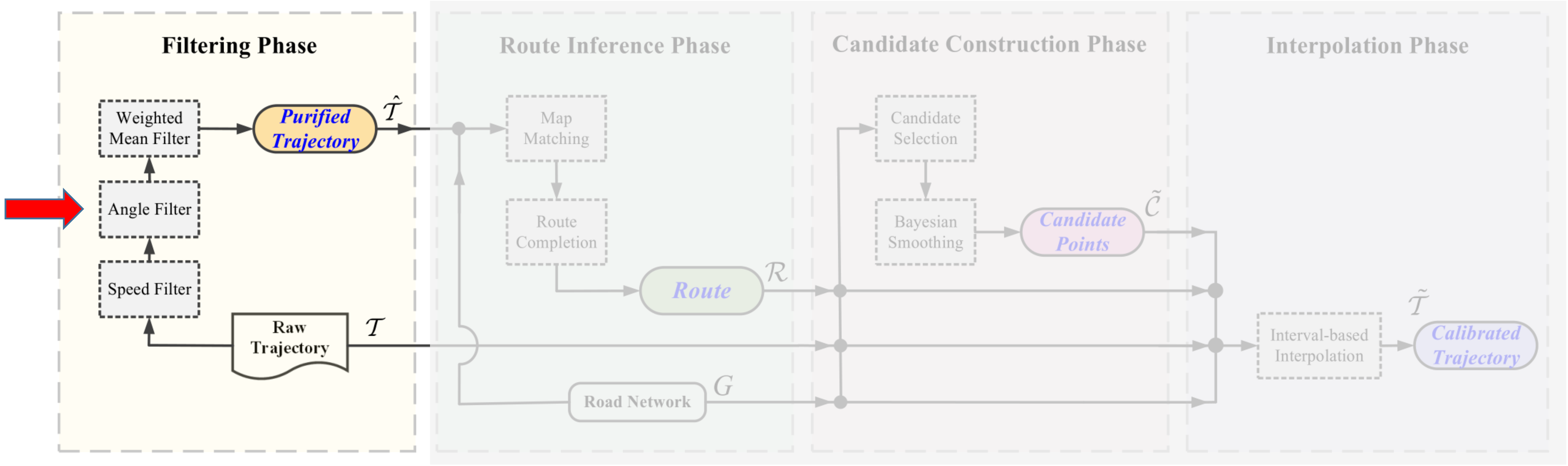
(a) Example of an original trajectory



(b) Result of the trajectory adopting the speed filter

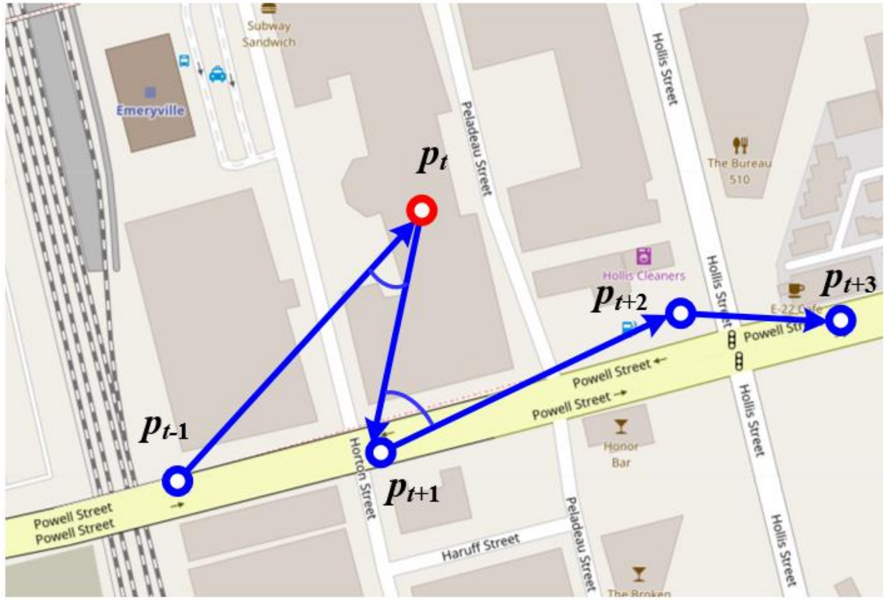
Phase 1: Filtering Phase

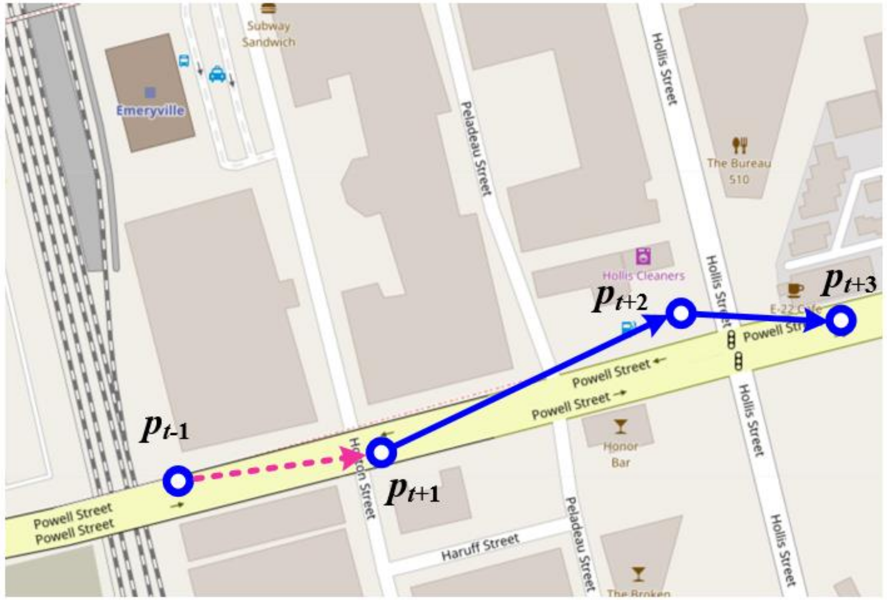
- Purify the raw noisy trajectory for the later map matching phase



Angle Filter

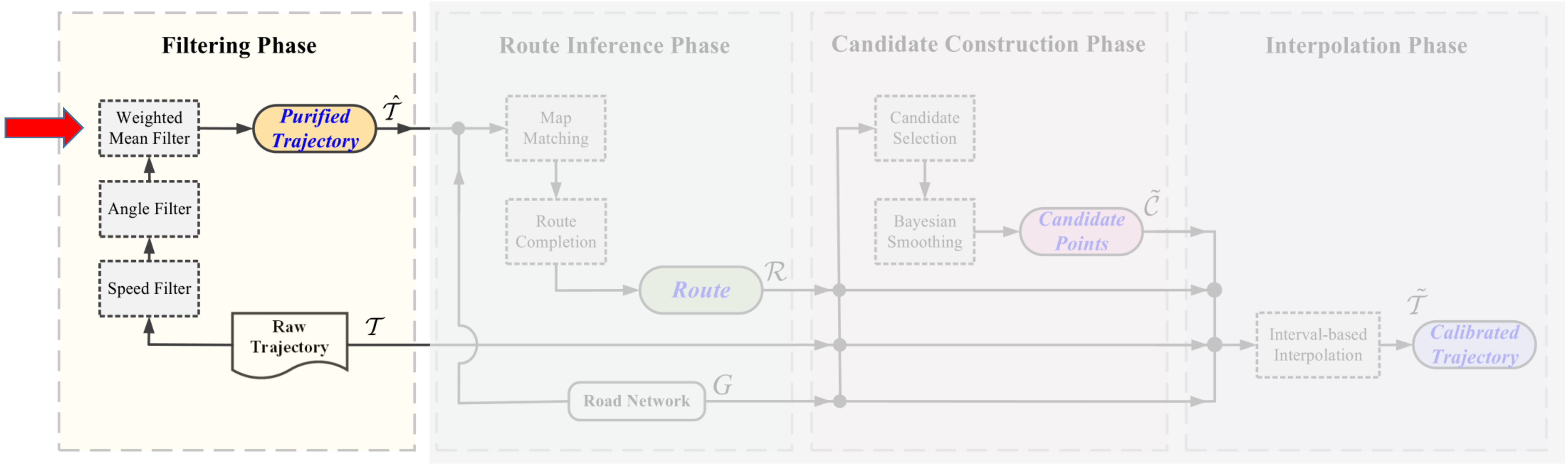
- Filter the points forming zigzag shapes.
  - Filter if $\angle p_{t-1}p_tp_{t+1} < \theta_{min}$ and $\angle p_tp_{t+1}p_{t+2} < \theta_{min}$





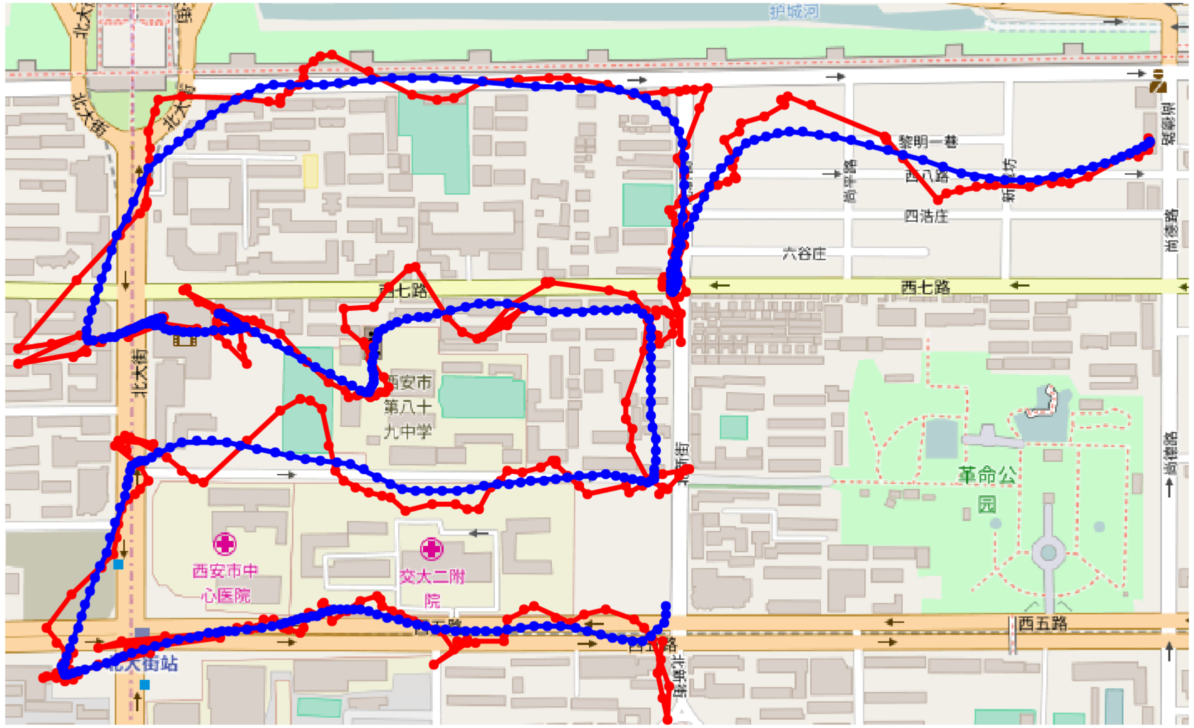
Phase 1: Filtering Phase

- Purify the raw noisy trajectory for the later map matching phase



Mean filter

- Use the average value among a context window to adjust the point
- Smooth the trajectory & reduce the outliers

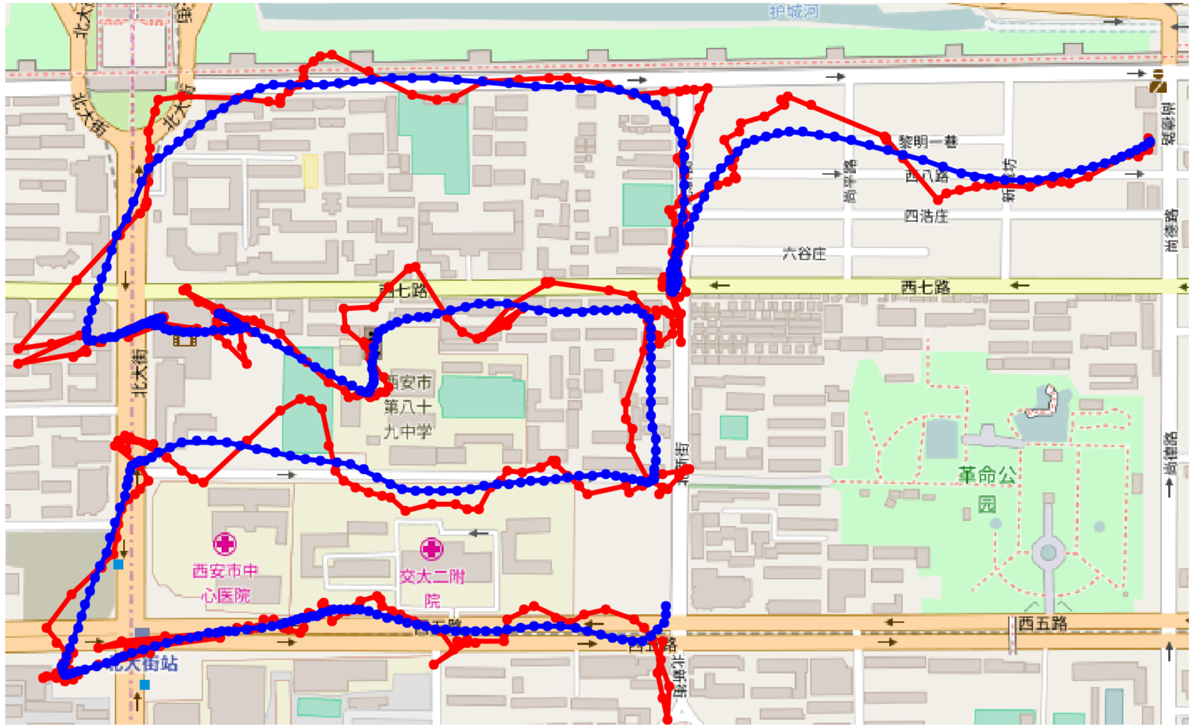


Red: Raw trajectory

Blue: Trajectory adopted mean filter

Mean filter

- Use the average value among a context window to adjust the point
- Smooth the trajectory & reduce the outliers



Red: Raw trajectory
Blue: Trajectory adopted mean filter

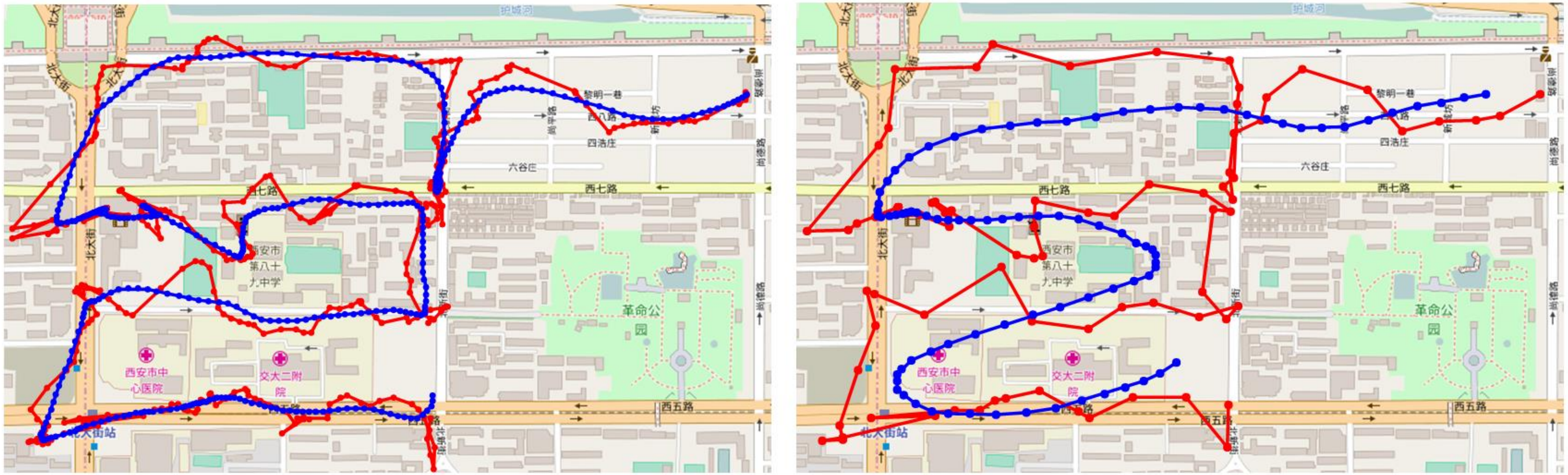
Problem:

How to decide the context window size?
The window size depends on the sampling rate

Mean filter

- Use the average value among a context window to adjust the point
- Smooth the trajectory & reduce the outliers

Red: Raw trajectory
Blue: Trajectory adopted mean filter



(a) Sampling interval = 2s, traditional mean filter (b) Sampling interval = 10s, trandition mean filter

with the same window size (15 in this case)

Weighted Mean Filter

- A point far away from the center of the window, a lower weight will be assigned

$$w(p_i, p_j) = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left[-\frac{(t_j - t_i)^2}{2\sigma_m^2}\right]$$

- Then the adjusted coordinate of p_i will be

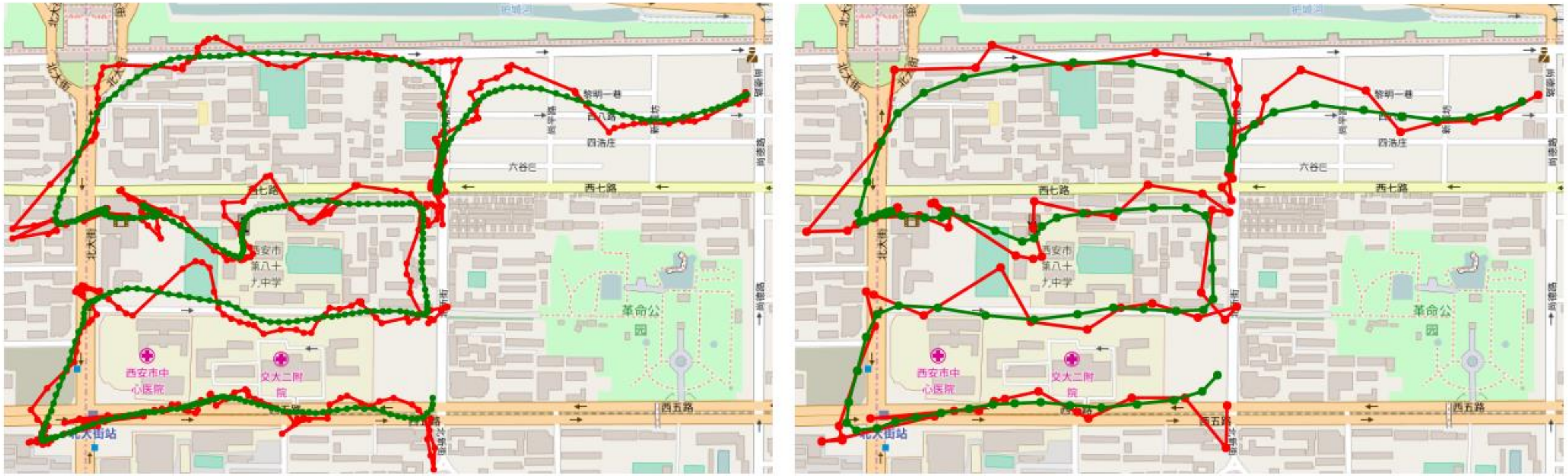
$$\hat{p}_x^{(i)} = \frac{1}{\sum_{k=2}^{\eta-1} w\left(p_x^{(x_k)}, p_i\right)} \sum_{j=2}^{\eta-1} w\left(p_i, p_x^{(x_j)}\right) \cdot p_x^{(x_j)}$$

$$\hat{p}_y^{(i)} = \frac{1}{\sum_{k=2}^{\eta-1} w\left(p_y^{(y_k)}, p_i\right)} \sum_{j=2}^{\eta-1} w\left(p_i, p_y^{(y_j)}\right) \cdot p_y^{(y_j)}$$

Weighted Mean Filter

- One window size will be enough to handle different sampling intervals

Red: Raw trajectory
Green: Trajectory adopted weighted mean filter

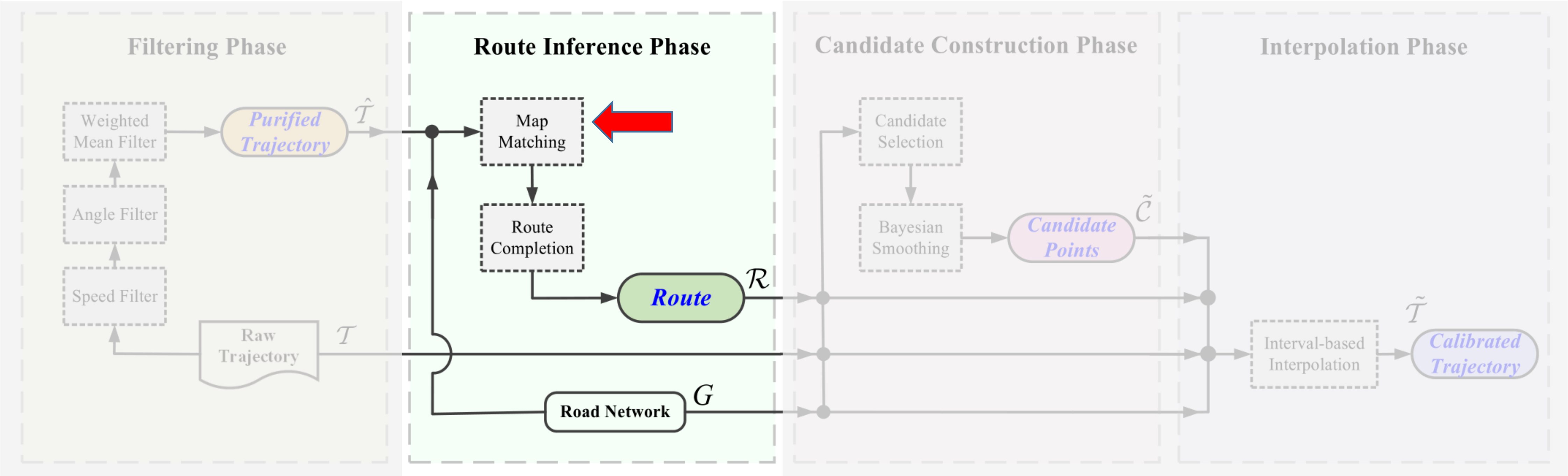


(c) Sampling interval = 2s, weighted mean filter (d) Sampling interval = 10s, weighted mean filter

with the same window size (15 in this case)

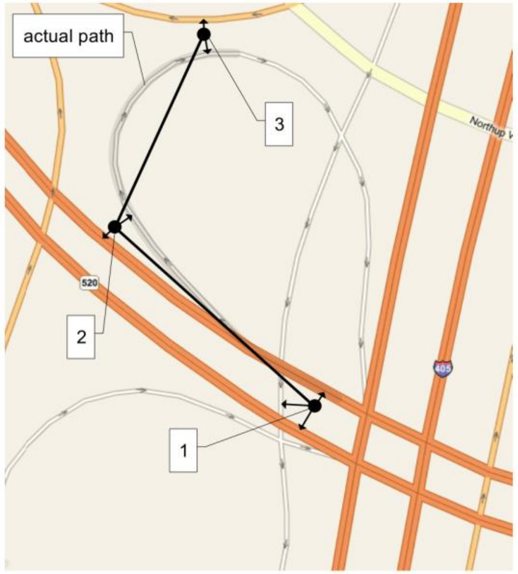
Phase 2: Route Inference Phase

- Use the purified trajectory $\hat{\mathcal{T}}$ to get the route $\mathcal{R}$ with the assistance of the road network G



Map Matching

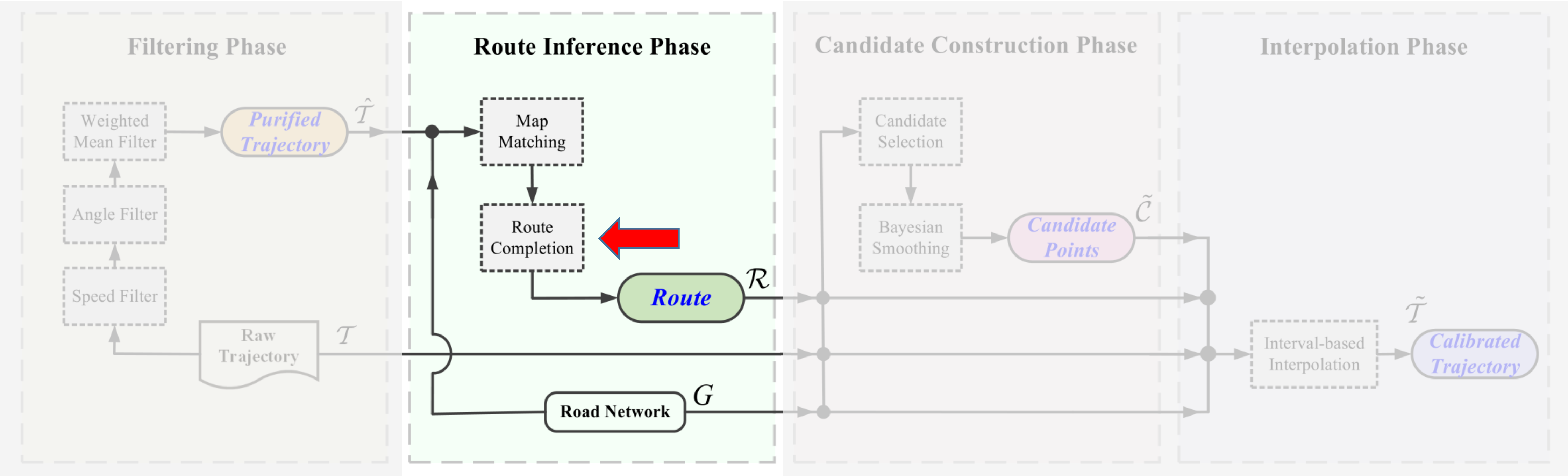
- Map each point to a road segment
- Hidden Markov Model-based Map matching algorithm



Paul Newson and John Krumm. Hidden Markov map matching through noise and sparseness, SIGSPATIAL'09

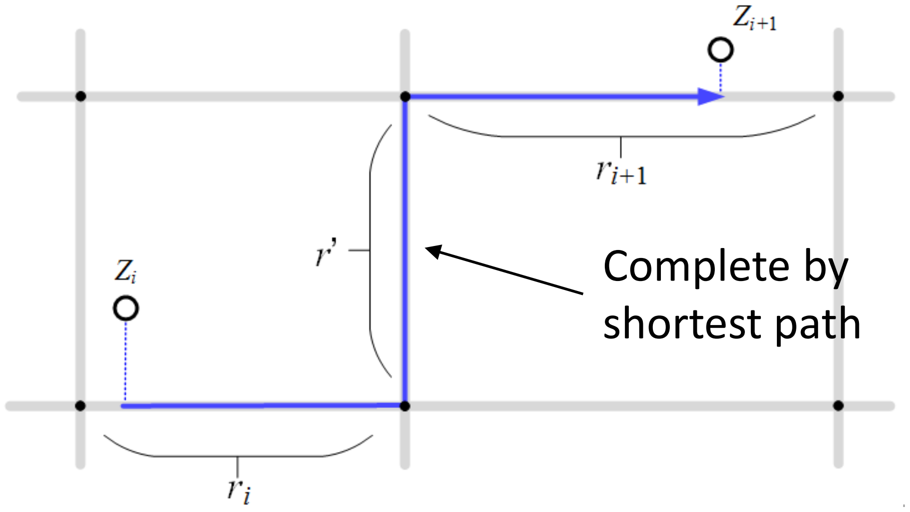
Phase 2: Route Inference Phase

- Use the purified trajectory $\hat{\mathcal{T}}$ to get the route $\mathcal{R}$ with the assistance of the road network G



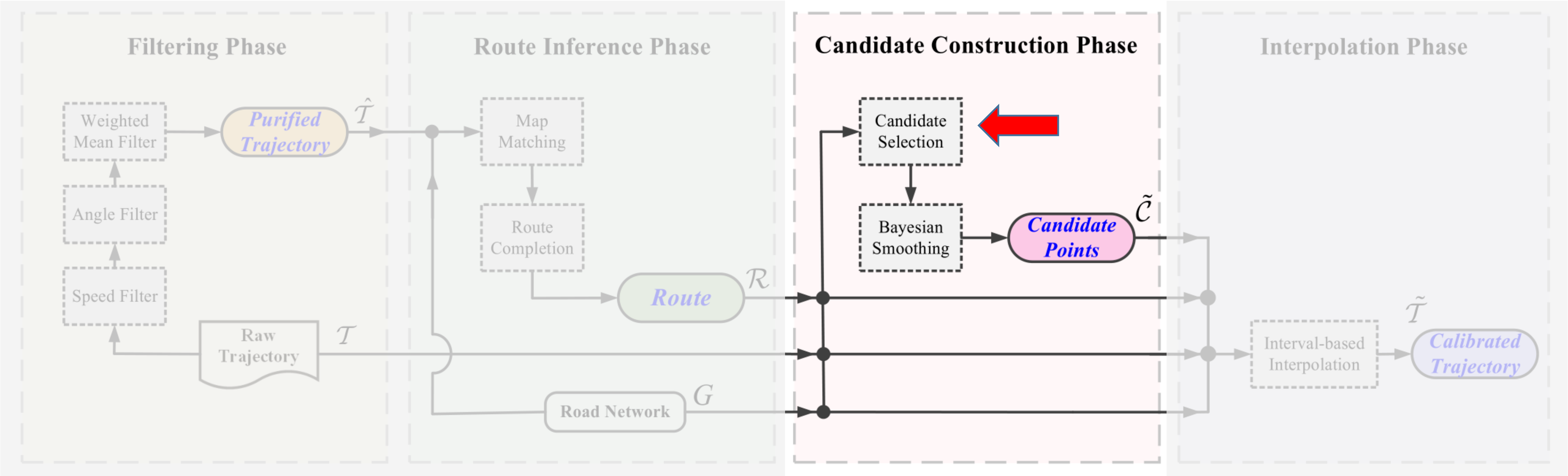
Route Completion

- Map matching: k points, k road segments
- Connect these road segments to form a complete route $\mathcal{R}$



Phase 3: Candidate Construction Phase

- Select the candidate points to support interpolation

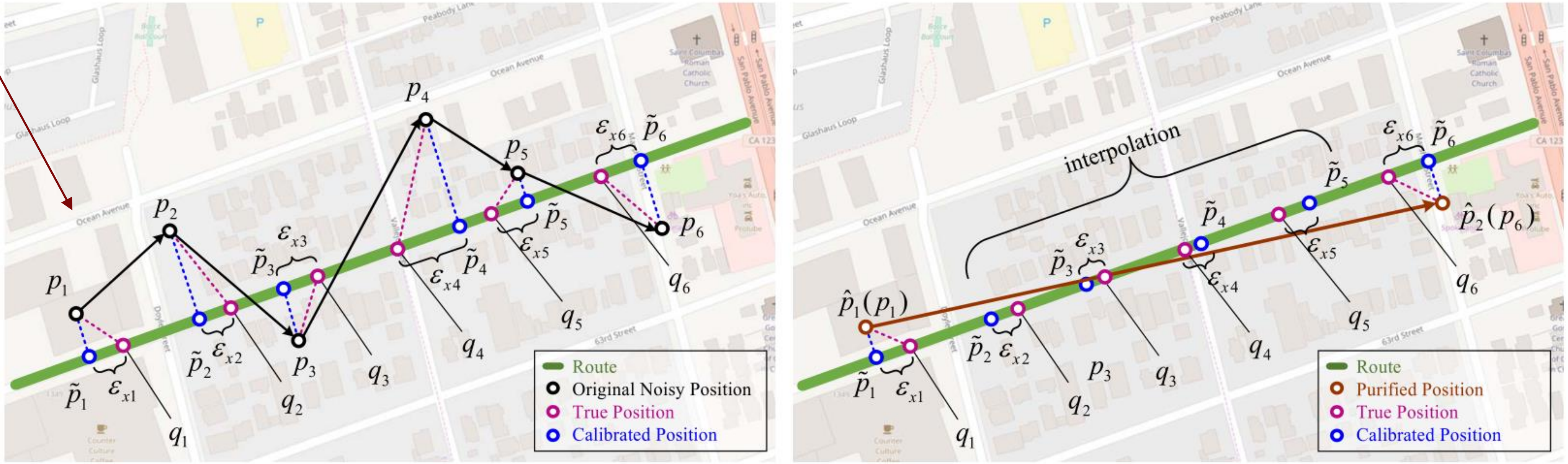


Now we have the route information

Can only reduce the vertical error!

Project the raw trajectory onto the road segments to do the calibration

Project part of trajectory points onto the road segments for calibration and calibrate the remaining points by interpolation



(a) Based on raw trajectory $\mathcal{T}$ (b) Based on purified trajectory $\hat{\mathcal{T}}$

Fig. 5. Leveraging route information to further reduce the error.

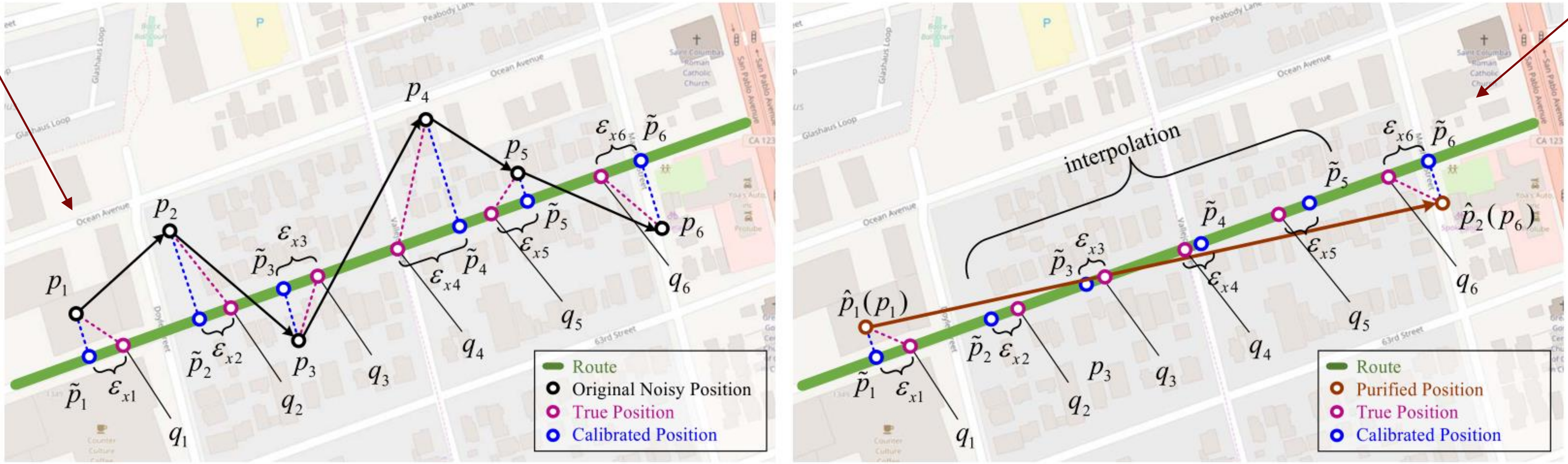
Now we have the route information

Can only reduce the vertical error!

Project the raw trajectory onto the road segments to do the calibration

Project part of trajectory points onto the road segments for calibration and calibrate the remaining points by interpolation

Have chance to reduce the along-road error!



(a) Based on raw trajectory $\mathcal{T}$

(b) Based on purified trajectory $\hat{\mathcal{T}}$

Fig. 5. Leveraging route information to further reduce the error.

How to select these “part of trajectory points”

- “part of trajectory points” - candidates
- Intuition
  - The error of interpolated points will be influenced by the error of candidates.
  - Select candidates with small errors

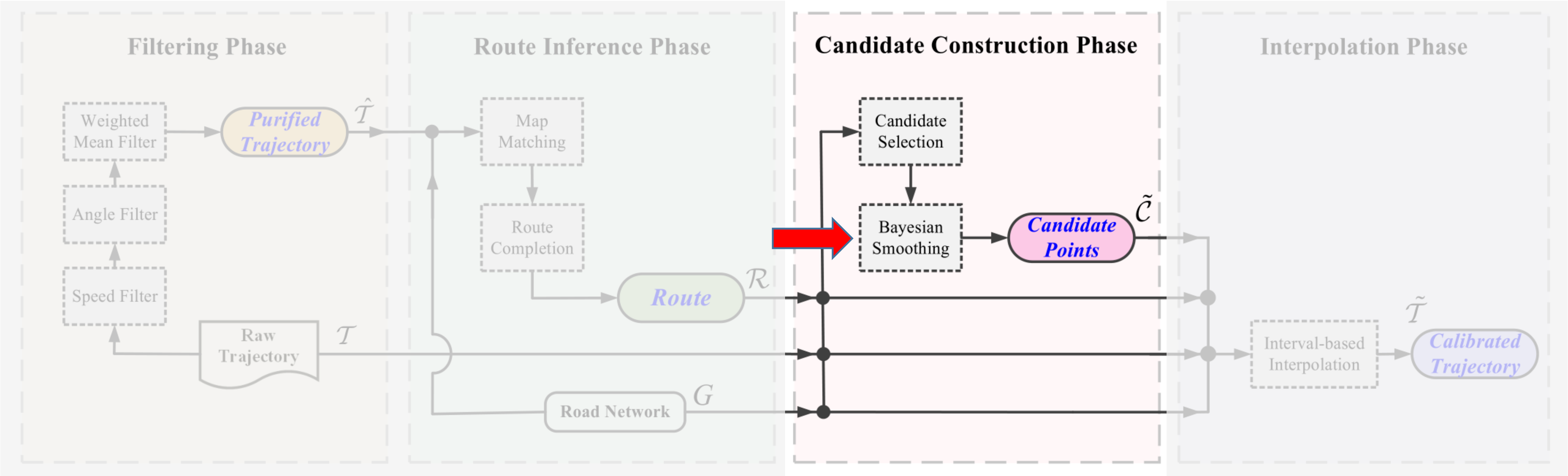
Theorem

THEOREM 4.1. *If the noisy measurement* $\mathbf{p} = \begin{bmatrix} p_x \\ p_y \end{bmatrix}$ *is generated according to a 2-D Gaussian distribution* $\mathcal{N}\left(\begin{bmatrix} q_x \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}\right)$, *suppose we have known the error* ε_y *in the y-direction of the measurement* $\mathbf{p}$, *the expectation of the squared full error of* $\mathbf{p}$, *i.e,* $\mathbb{E}[\varepsilon^2]$ *is* $\left(\left(\frac{\varepsilon_y^2}{\sigma_y^2} - 1\right)\rho^2 + 1\right)\sigma_x^2 + \varepsilon_y^2$.

- Smaller ε_y indicates a smaller full error ε
- Yields a simple but useful selection criteria
  - Selecting points having vertical error ε_y smaller than a given threshold ζ
  - Then the along-road error will be bounded by $\sqrt{\left(\left(\frac{\zeta^2}{\sigma_y^2} - 1\right)\rho^2 + 1\right)\sigma_x^2}$

Phase 3: Candidate Construction Phase

- Select the candidate points to support interpolation



Bayesian Smoothing

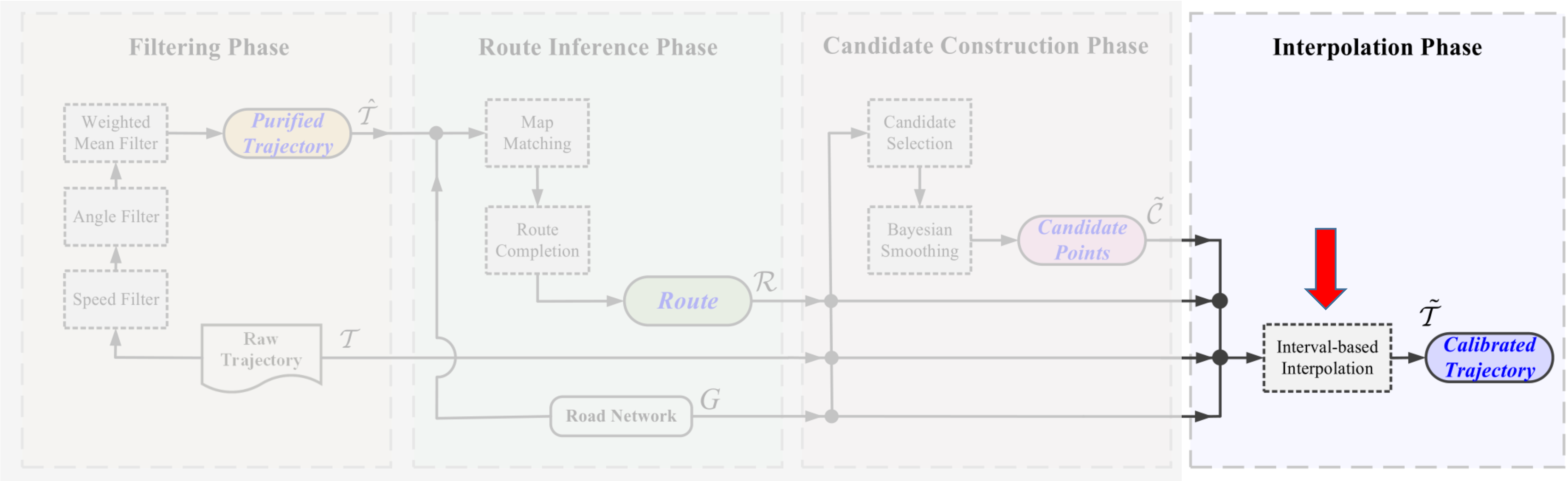
- Further smooth the candidates (further reduces the error of candidates)
- Rauch-Tung-Striebel Smoother (RTSS)

Bayesian Smoothing Technique

- Dynamical state space model
  - Similar to HMM but Bayesian Smoother is continuous.
- Rauch-Tung-Striebel Smoother (RTSS)
  - $$P(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\mathbf{A}_t\mathbf{z}_{t-1} + \mathbf{B}_t, \mathbf{Q}_t)$$
    $$P(\mathbf{c}_t|\mathbf{z}_t) = \mathcal{N}(\mathbf{H}_t\mathbf{z}_t + \mathbf{C}_t, \mathbf{R}_t)$$
  - $$\mathbf{z}_\mathrm{t} = \left[q_x^{(t)}, q_y^{(t)}, v_x^{(t)}, v_y^{(t)}\right]^\top, \mathbf{p}_\mathrm{t} = \left[c_x^{(t)}, c_y^{(t)}\right]^\top$$
  - $$\mathbf{q}_t = \mathbf{q}_{t-1} + \delta_t\mathbf{v}_t$$

Phase 4: Interpolation Phase

- Interpolate the remaining points for calibration

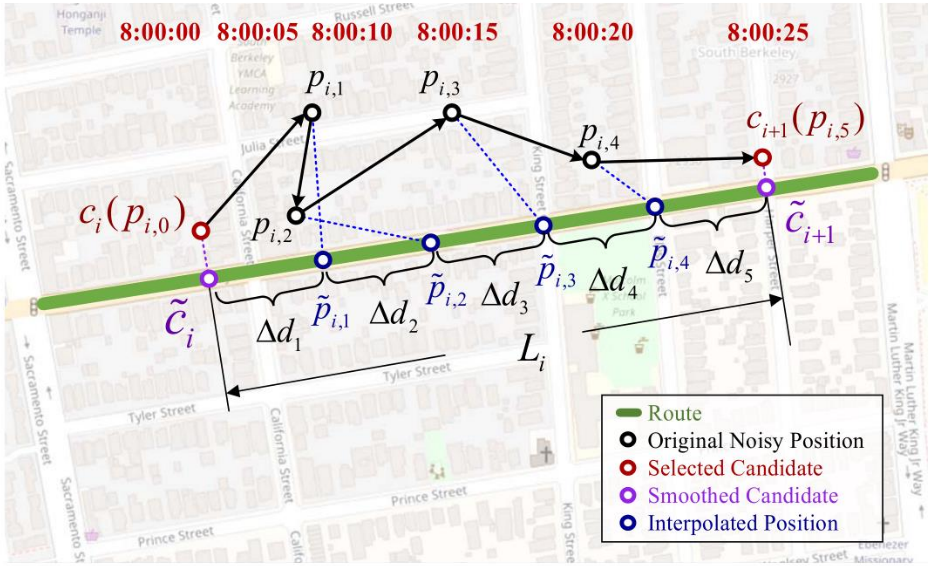


Interval-based Interpolation

- Spatial measurement is noisy, while temporal measurement is accurate
- Assuming the car is moving in a constant speed in a short interval

$$\Delta d_1 : \Delta d_2 : \Delta d_3 : \Delta d_4 : \Delta d_5 = (t_{i,1} - t_{i,0}) : (t_{i,2} - t_{i,1}) : (t_{i,3} - t_{i,2}) : (t_{i,4} - t_{i,3}) : (t_{i,5} - t_{i,4})$$

$$L_i = \Delta d_1 + \Delta d_2 + \Delta d_3 + \Delta d_4 + \Delta d_5$$



Experiment

- GPS as the ground truth
- Cellular-based positioning as the raw noisy trajectory

	Xi'an	Beijing	Kuwait City
# Points	10018	1338	1188
# Road Segments Passed	276	151	110
Duration	3h13m49s	1h19m	28m28s
Sampling Interval (s)	1.16	3.57	1.44
Length (km)	35.96	36.25	21.19
Average Speed (km/h)	10.81	27.33	45.20
Original Localization Error (m)	82.86	212.95	167.57
Moving Style	covering road network	covering road network	normal driving
Cellular Network	4G (LTE)	3G (UMTS)	4G (LTE)
Raw Signal	RSRP	Cell-ID	RSRP
Localization Technique	RFPM[34]	WCCL[30]	RFPM[34]

Experiment

- GPS as the ground truth
- Cellular-based positioning as the raw noisy trajectory

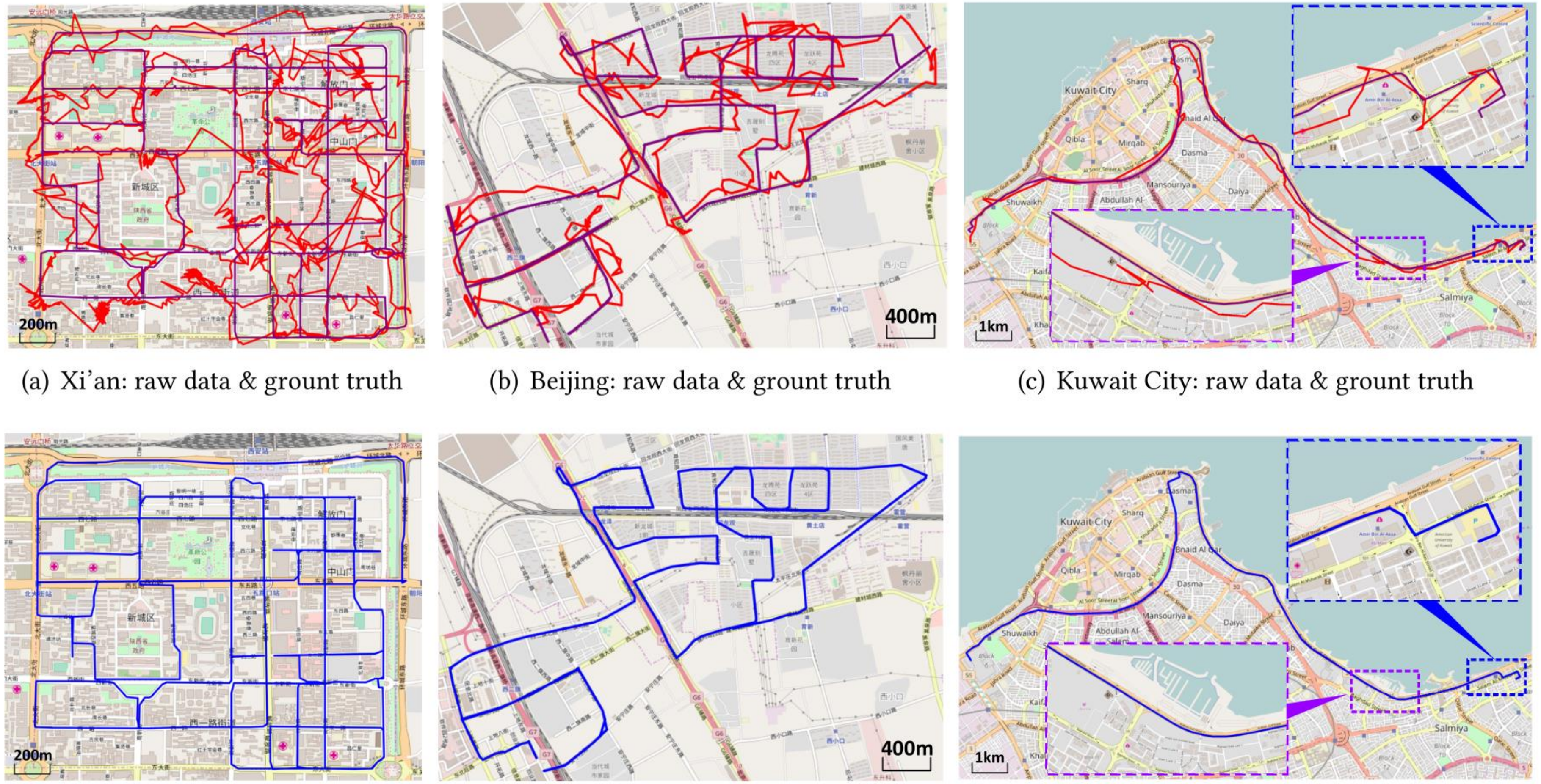
(a) Xi'an: raw data & grount truth (b) Beijing: raw data & grount truth (c) Kuwait City: raw data & grount truth

Baselines

- SnapNet — The map matching approach designed for cellular-based trajectories
- HMM-MM (HMM-based map matching)
- RTSS
- KF (Kalman filter)
- MF (Mean filter)
- RTSS+PJ
- KF+PJ
- MF+PJ
- PJ

RTSS+PJ, KF+PJ, MF+PJ: +*PJ version*: Project the calibrated trajectory points onto the nearest road segments

Visualization Results

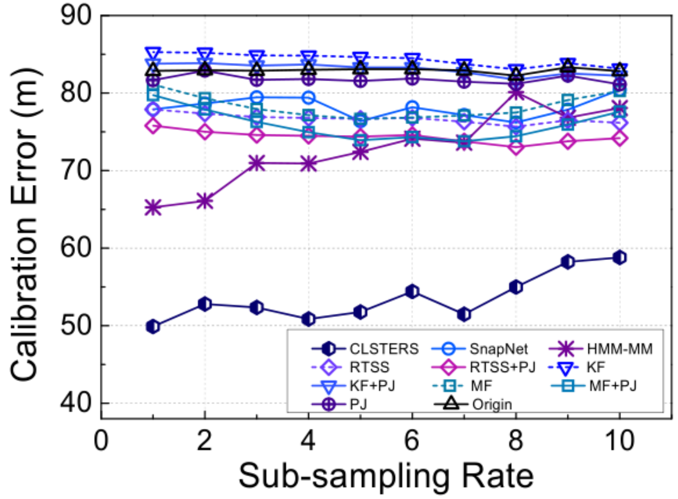


(a) Xi'an: raw data & grount truth (b) Beijing: raw data & grount truth (c) Kuwait City: raw data & grount truth

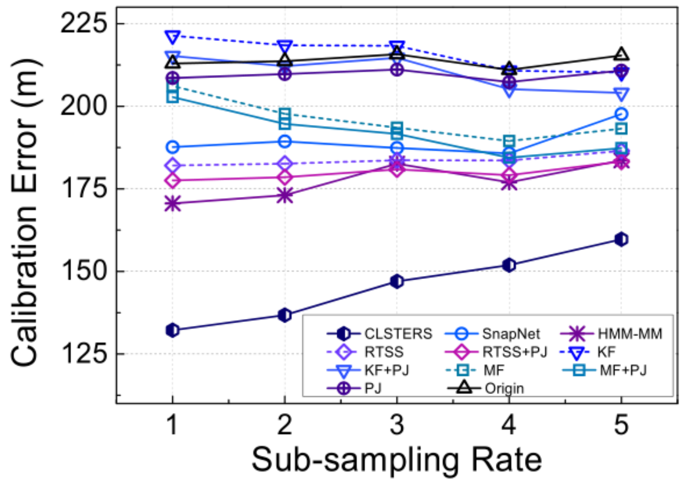
(d) Xi'an: the calibrated result of CLSTERS (e) Beijing: the calibrated result of CLSTERS (f) Kuwait City: the calibrated result of CLSTERS

Fig. 8. The visualization of the three datasets. Trajectories in red are generated by cellular-based localizations. Those in purple are generated by GPS and mapped to the road network which can be regarded as the ground truth and those in blue are the calibrated trajectories output by CLSTERS. Note that the measure scales of these maps are different.

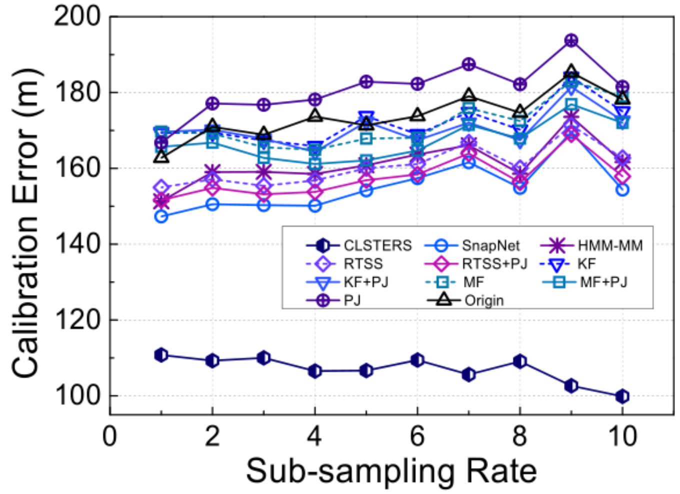
Overall Comparison (Lower better)



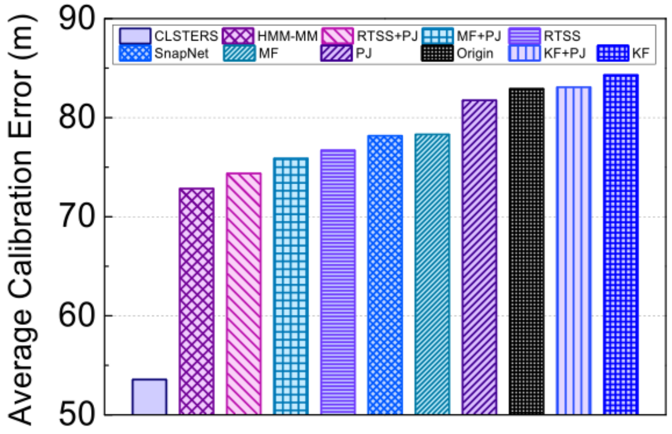
(d) Xi'an: error vs. sub-sampling rates



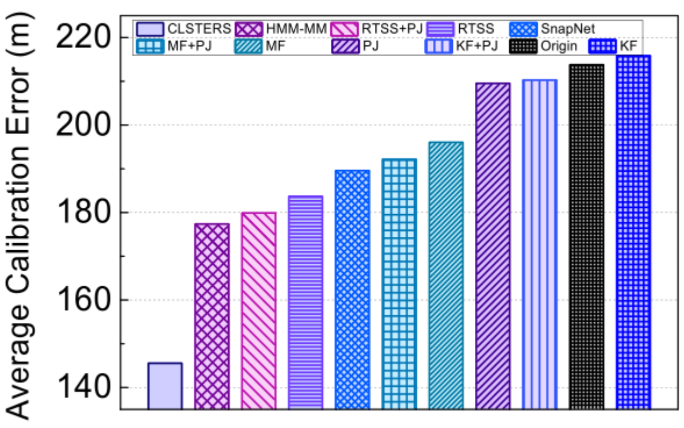
(e) Beijing: error vs. sub-sampling rates



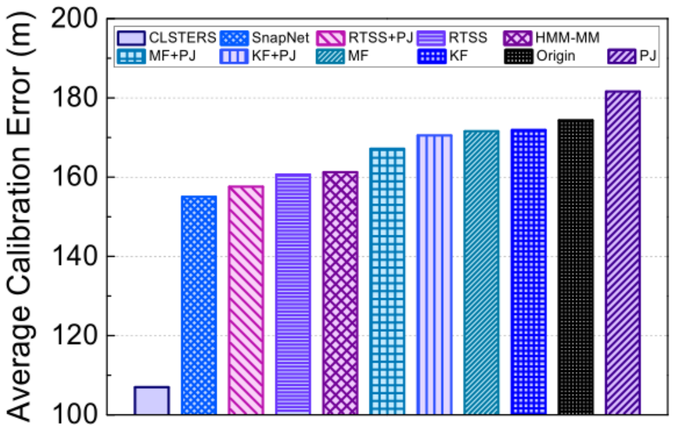
(f) Kuwait: error vs. sub-sampling rates



(g) Xi'an: average error

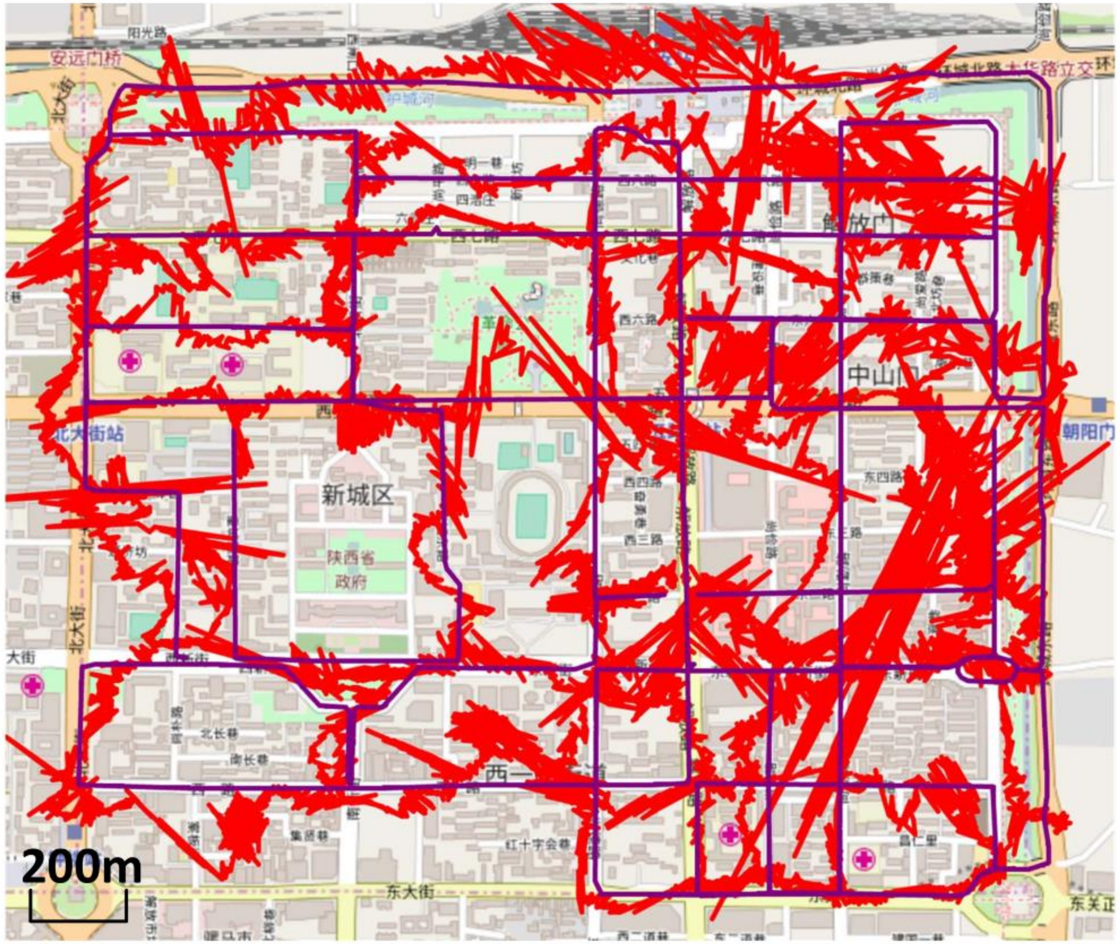


(h) Beijing: average error

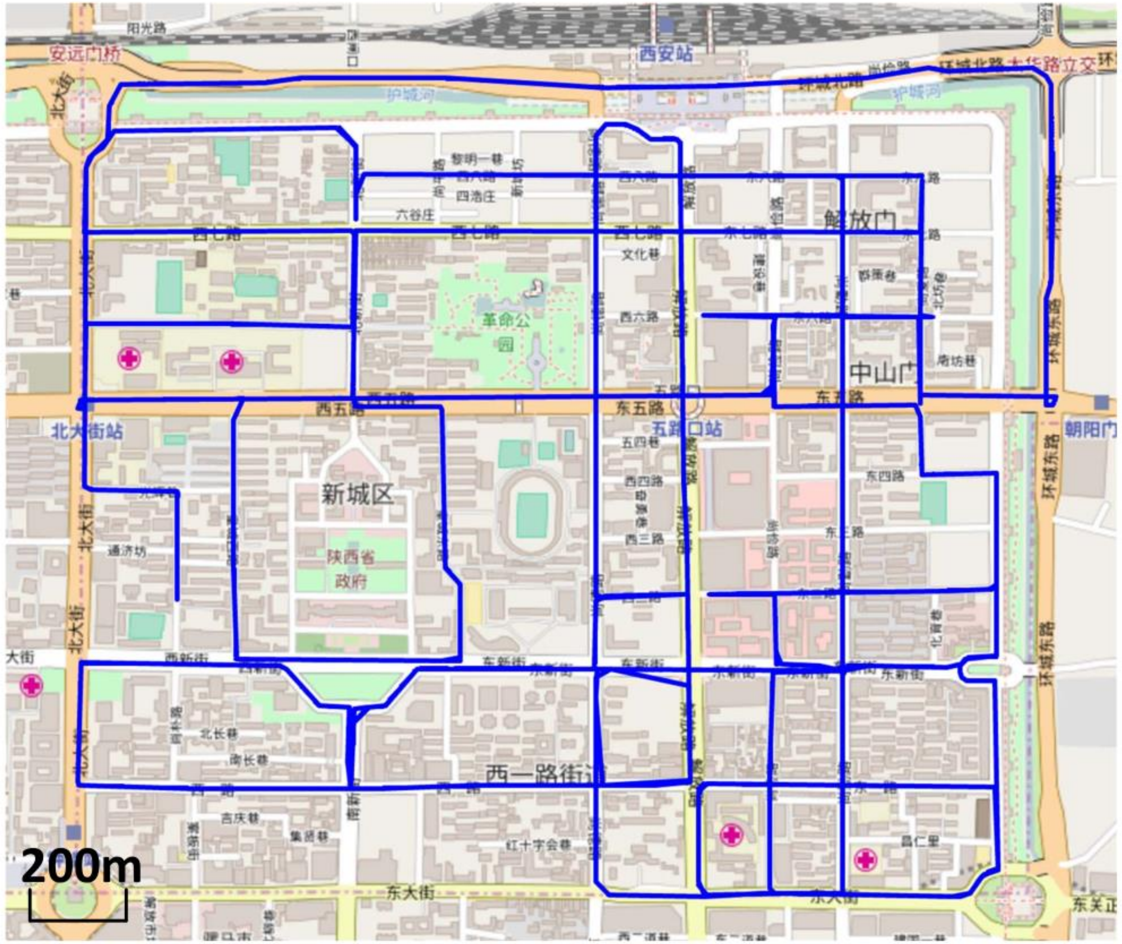


(i) Kuwait: average error

Performance Under Pedestrian Scenario

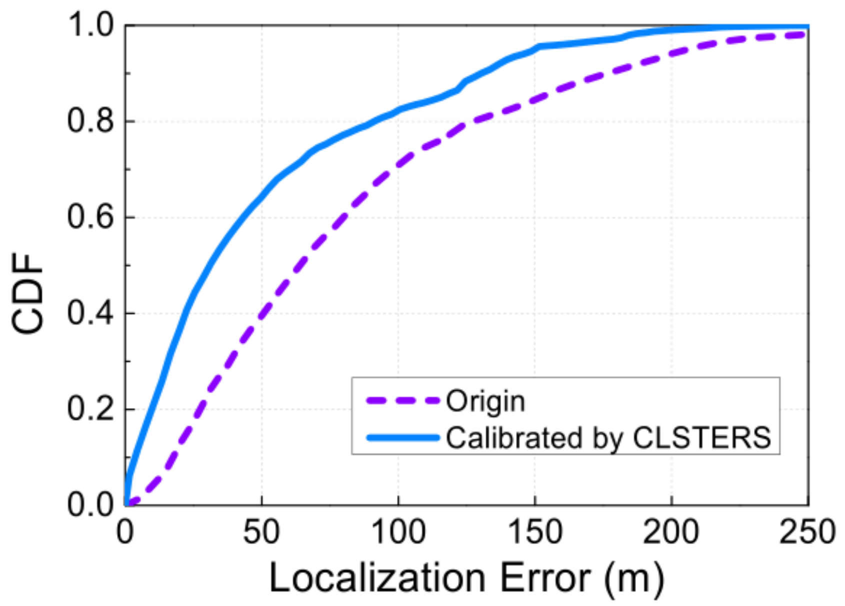


(a) Simulated pedestrian trajectory

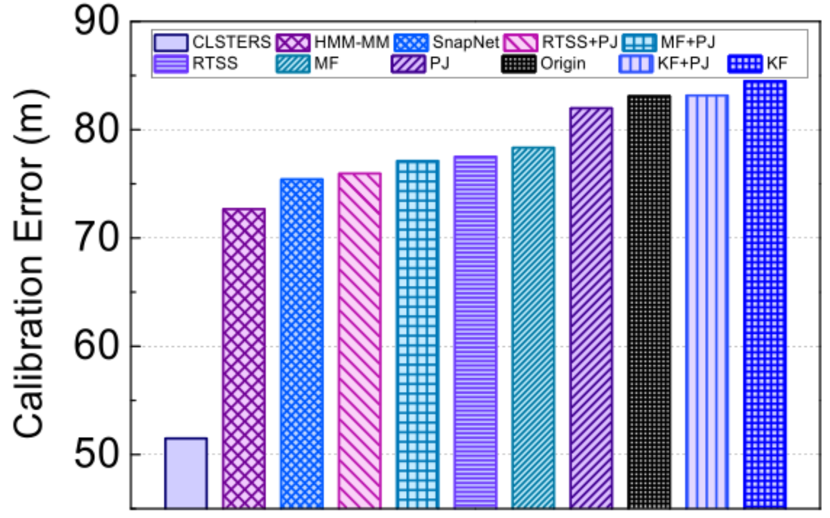


(b) The calibrated result of CLSTERS

Performance Under Pedestrian Scenario



(c) The CDF of error



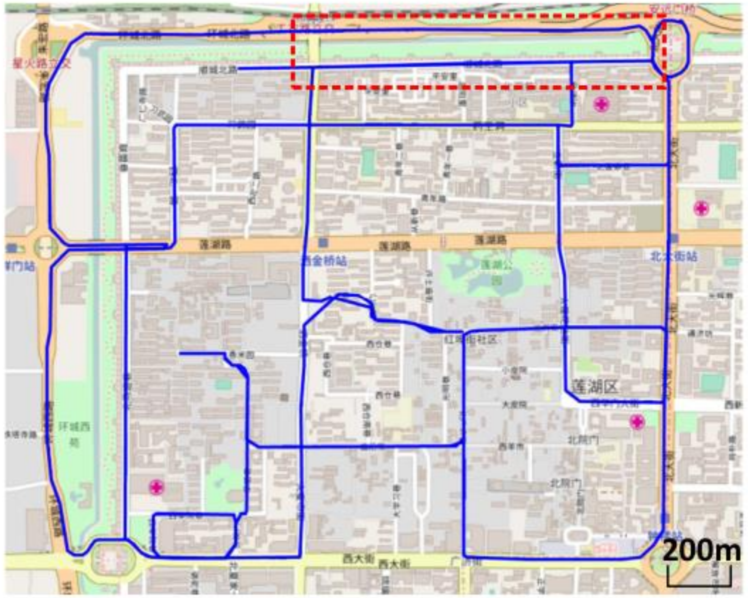
(d) Calibration error

Parameter Generalization

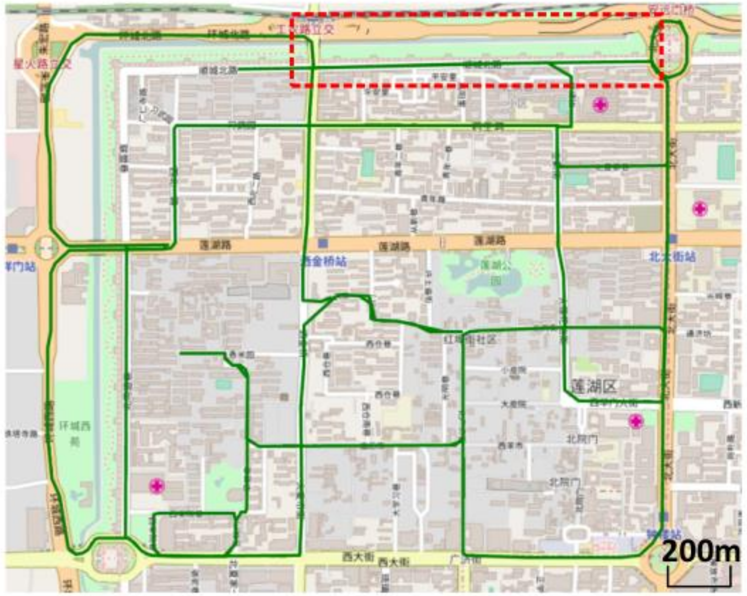
Collect a trajectory performed in another area in Xi’an

Raw trajectory & ground truth

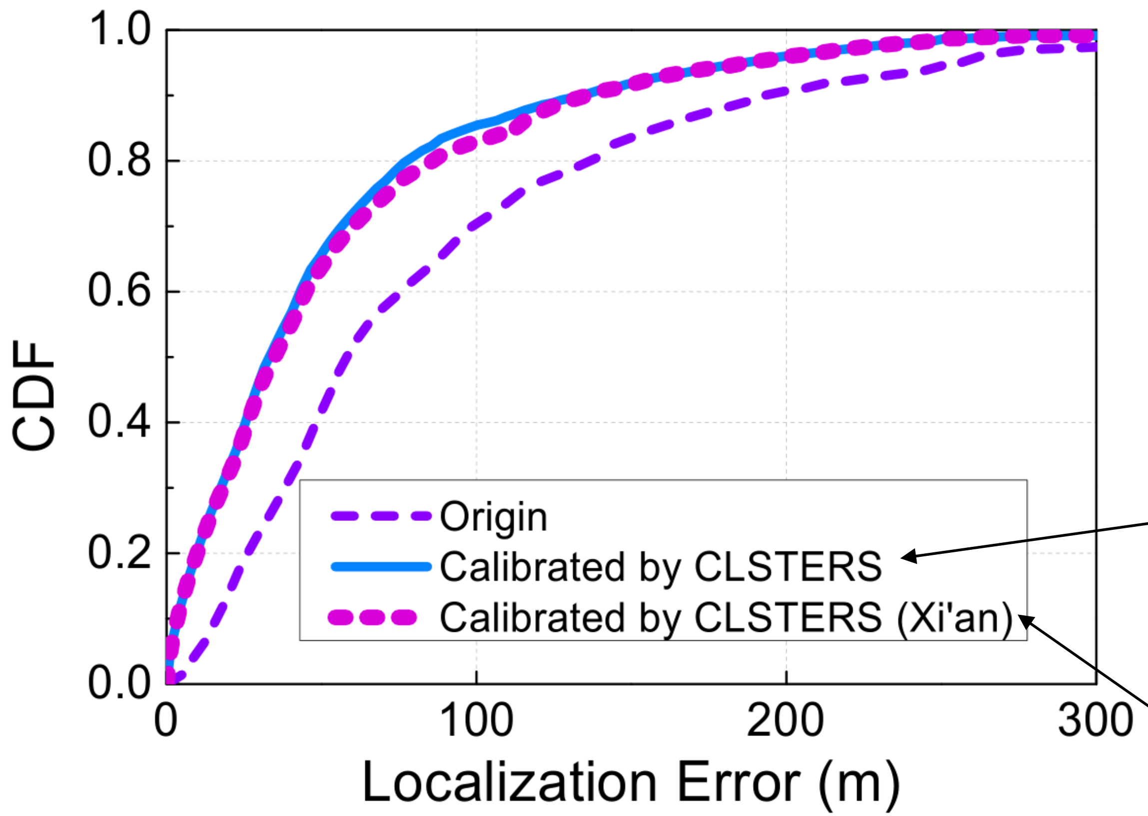
Performance under the best parameter setting



Performance under the parameter setting tuned by another area/dataset



Parameter Generalization



Performance under the best parameter setting

Performance under the parameter setting tuned by another area/dataset

Conclusion

- We proposed a general a general system for reducing error of trajectories localized in challenging localization scenarios.
- Independent on the hardware environment and localization techniques of the raw data
- Support to only use the (x, y, t) information
- Reduce about 40% error while baselines can only reduce 10% error

Thank you

- *Mail: wuhao5688@fudan.edu.cn*

- *Any question?*

Hao Wu (吴昊)
Computer Science and Technology, Fudan University
1159 CaiLun Rd., Shanghai, China, 201203
Tel: +8613817149574
E-mail: wuhao5688@fudan.edu.cn