Department of Informatics
Technische Universität München

All Your Location are Belong to Us: Breaking Mobile Social Networks for Automated User Location Tracking by Li *et al.* (2013)

Sub-Topic: ***Privacy***

Benedikt Simon Beyer

Munich, May 16th 2019

Master-Seminar Internet of People: Connectivity, Mobility and Privacy

Leonardo Tonetto, Vaibhav Bajpai

Outline

- Introduction to the Topic
- Research Questions
- Classification of Location-based Social Networks
- Methodology
- Results
- How to mitigate the privacy threat by LBSNs?
- Critique and Implications

What are Location-Based Social Networks?

	Distance	Accuracy Limit	Coverage Limit	Number of Users (millions)	Platform or region	SDK	Category
Wechat	Y	100m	1km (Shanghai)	300 millions	iOS/Android/WP	Google	II
Skout	Y	0.5mile	N/A	5 millions	iOS/Android/WP	Google	II
Momo	Y	10m	N/A	30 millions	iOS/Android/WP	Baidu	II
Whoshere	Y	100m	N/A	5 millions in 2012	iOS/Android	Google	II
MiTalk	Y	100m	0.6km (Shanghai)	20 millions	iOS/Android	Baidu	II
Weibo	Y	100m	1600m	500 millions	iOS/Android/WP	Google	II
SayHi	Y	10m	1000km	500 thousands	iOS/Android	Google	I/II
iAround	Y	10m	N/A	10 millions	iOS/Android	Baidu	I/II
Duimian	Y	100m	N/A	500 thousands	iOS/Android	Google	II
Doudou Friend	Y	10m	N/A	1 million	iOS/Android	Amap	II
U+	Y	10m	N/A	10 millions	iOS/Android	Baidu	II
Topface	Y	100m	N/A	50 million	iOS/Android	Google	II
Niupai	Y	10m	N/A	61 thousands	iOS/Android	Google	II
LOVOO	Y	100m	27.8km (Shanghai)		iOS/Android	Google	II
KKtalk	Y	10m	N/A	320 thousands	iOS/Android	Google	II
Meet24	Y	0.5mile	N/A		iOS/Android	Google	II
Anywhered	Y	10m	N/A	750 thousands	Android	Baidu	II
I Part	Y	10m	1000m	8 millions	iOS/Android	Google	II
Path	N	N/A	N/A	10 millions	iOS/Android	Google	I
TweetCaster	N	N/A	N/A	10 millions	iOS/Android/WP	Google	I
Google Plus	N	N/A	N/A	10 millions	iOS/Android/WP	Google	I
eHarmony	N	N/A	N/A	5 millions	iOS/Android	Google	I
SinglesAroundMe	N	N/A	N/A	1 million	iOS/Android	Google	I

Table 1: Summary of Location-based Friend Discovery Apps

Research Questions

1. Is it possible to make an **involuntary localization** of a **random** LBSN user by exploiting the **public available** information only?
    - *No hacking*

2. Could we **freely track** a particular user within a reasonably short time period?
    - *Investigating* three most popular LBSN apps (Wechat, Momo and Skout)

The answer to the two questions is **yes**.

Classification of Location-Based Social Networks

Two categories:

1. LBSNs with **Exact** Location Sharing

2. LBSNs with **Indirect** Location Sharing

Classification of Location-Based Social Networks

LBSNs with Exact Location Sharing

a) Open access Location Sharing
(present the **exact** location without any restriction)

b) User Authorized Location Sharing
(users can **decide** with whom they share the exact location information)

Classification of Location-Based Social Networks

LBSNs with Indirect Location Sharing

→ Special *Location Hiding Techniques* are implemented to **obfuscate** exact locations**.**

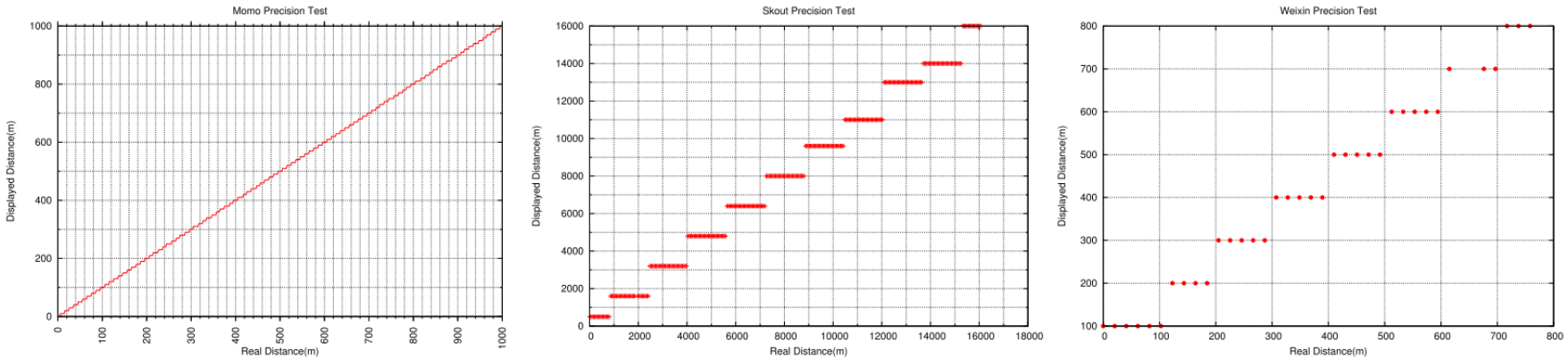
a) **Relative Distance Only** (Wechat, Skout, Momo)
- no geographical coordinates, only geographical distances

a) **Setting the Minimum Accuracy Limit**
- accuracy is not better than 1 mile in Skout, Wechat 100m, and Momo 10m

b) **Setting the Localization Coverage Limits**
- only users within a certain range or region (e.g. Weachat 1000m)
A maximal number of users visible is also possible

Classification of Location-Based Social Networks

Location Hiding Techniques in Practice



(a) Momo precision test (b) Skout precision test (c) Wechat precision test

Figure 1: Updating Strategy Evaluation Results

- Relative Distance only

- Relative Distance and
- Minimum Accuracy Limit

- Relative Distance,
- Minimum Accuracy Limit and
- Localization Coverage Limits

Methodology

Attack Methodology

Realtime experiment:

- With an automated user location tracking system for mobile social networks that tracks Wechat, Skout, and Momo users without any awareness.

- Attack towards 30 volunteers in a three-week from United States, China and Japan.

Accuracy:

→ Top 5 locations of one user

Methodology

Attack Methodology

Realtime experiment:

- With publicly available information provided by the LBSN app

- No hacking of internal operations

- User location is based on the relative distance information

- Exploiting the returned information of relative distance by using virtual **Anchor Points**
  - → Launch localization algorithms and geo-locate the victim
  - → Break the accuracy limit

Methodology

Overview

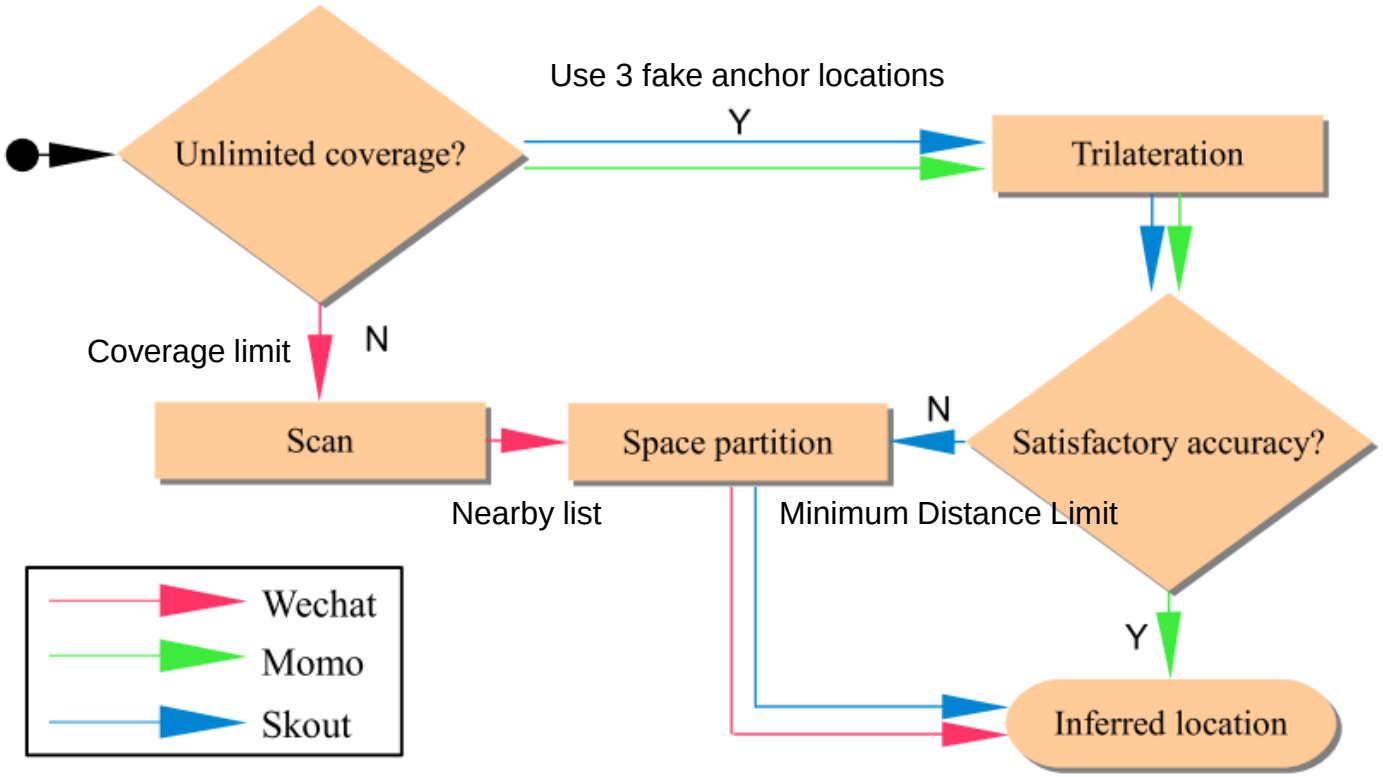


Figure 2: The Attack Flow

Methodology

Breaking Minimum Distance Limit

- Dividing a space into two or more non-overlapping regions

- Locating any point in the space to exactly one of the regions

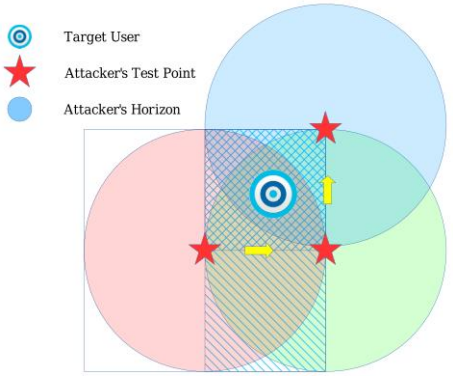


Figure 3: Illustration of Space Partition Attack

Breaking Localization Coverage Bound

- Social Popularity Index (**Zipf's law**)

- Higher priority to places with higher user population

  → Speed up location process

Implementation

Involves two key modules:

- Location Spoofing
- Location Reading

- System is implemented in Clojure (Programming Language)
  → Cope with MonkeyRunner
  → Control Android Virtual Machines
  → Send commands

- LocationFaker app to set the location in Android

Implementation
Spoofing Location

Add location providers in Android:

- Allow mock location
  - → Name to “gps”
  - → Feed fake location information

- Location faking components need to satisfy a certain accuracy to be not rejected
  - → if inaccurate, apps may return error messages (WeChat)
  - → Change the Android framework with ApkTool

- Accepting the fake location as the real location

Implementation
Fetching Location

Distance Reading based on fake locations:

- Simulate user input
  - → Perform tests on apps
  - → Integrating API
  - → Mimic user behavior
  - → Trigger a location information update
  - → Read out all items

- Read distance from the apps
  - → Filter log level to matching regular expression patterns

Real-World Experiment

30 volunteers for the 3 LBSN apps WeChat, Skout, and Momo:

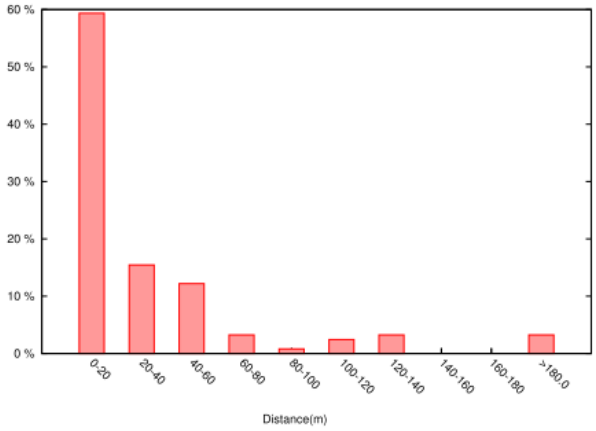
- Localization Accuracy
  - → Compare distance between real and inferred Locations
  - → Measure latency of launched attacks for different apps
    - → Localization Efficiency

- How many top locations could be recovered by using 3-week track?
  - → Filter log level to matching regular expression patterns

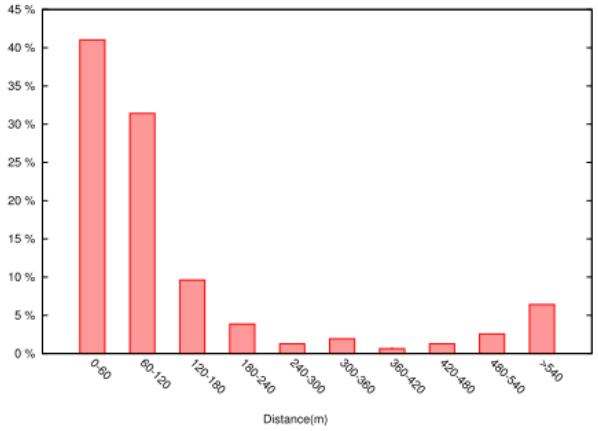
Real-World Experiment

350 reports and attacks from 30 volunteers in comparison:

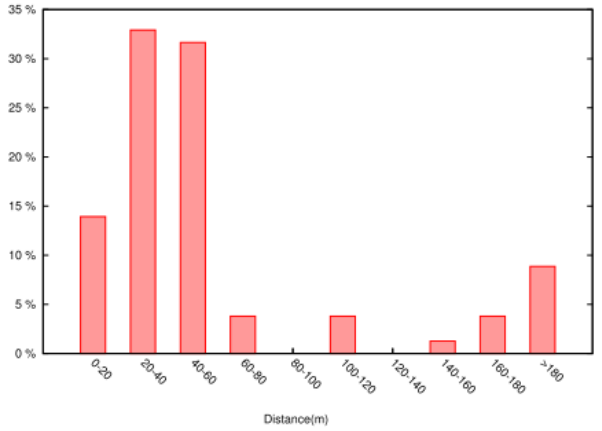
→ Localization differs in **Accuracy**



(a) Momo's Localization Accuracy



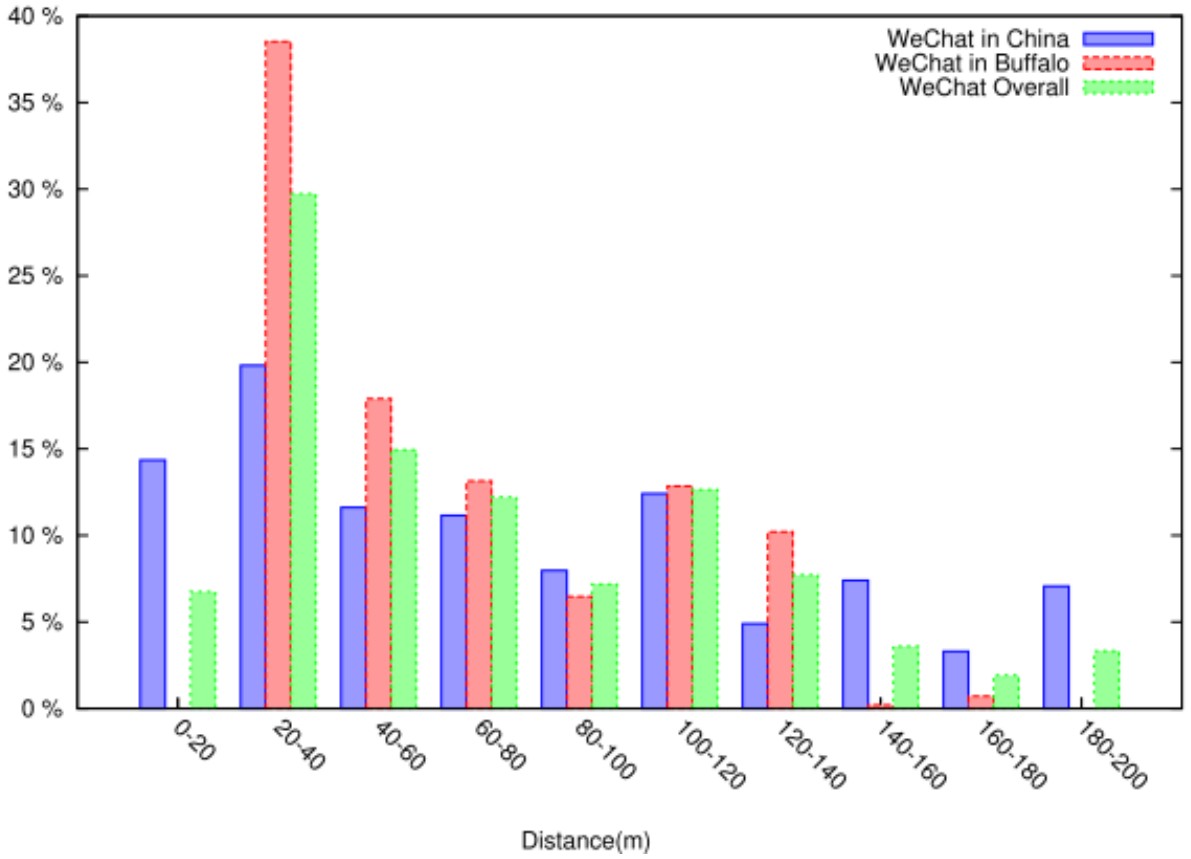
(b) Skout's Localization Accuracy



(c) Wechat's Localization Accuracy

Real-World Experiment

WeChat Accuracy Comparison



Real-World Experiment

Most Visited Places: N=5

top location	one week			two weeks			three weeks		
	Momo	Wechat	Skout	Momo	Wechat	Skout	Momo	Wechat	Skout
1	92.3%	50.0%	20.0%	100.0%	57.1%	60.0%	100.0%	71.4%	60.0%
2	46.1%	21.4%	0.0%	46.1%	21.4%	40.0%	69.2%	21.4%	40.0%
3	30.7%	21.4%	20.0%	46.1%	28.5%	60.0%	38.4%	28.5%	80.0%
4	23.0%	35.7%	20.0%	30.7%	35.7%	40.0%	38.4%	35.7%	40.0%
5	23.0%	21.4%	0.0%	15.3%	21.4%	40.0%	15.3%	14.2%	40.0%
	of 30 volunteers			of 30 volunteers			of 30 volunteers		

Results

1) An attacker could perform a range-free, involuntary user localization attack with high localization accuracy;

2) Furthermore, it can successfully establish very accurate user location profile

Attacker can easily identify top 5 locations

How to mitigate the privacy threat by LBSNs?

Limiting attacker's capability:

- Identifying potential anomalous users
  → Too fast changes of location

- Slowing down tracking process
  → But tracker can use multiple accounts

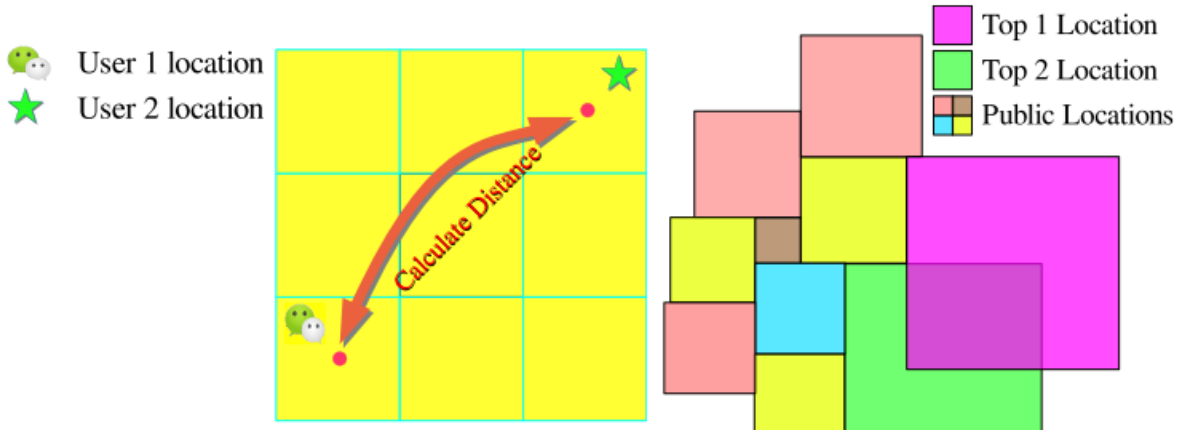
- Manual remove location access to the public

- Reducing accuracy
  → Adding more noise to the location management
  → Better privacy at the cost of users' utility

How to mitigate the privacy threat by LBSNs?

Introducing a Grid Reference System:

- Distance Obfuscation
  → prevent the attacker from using LBSN



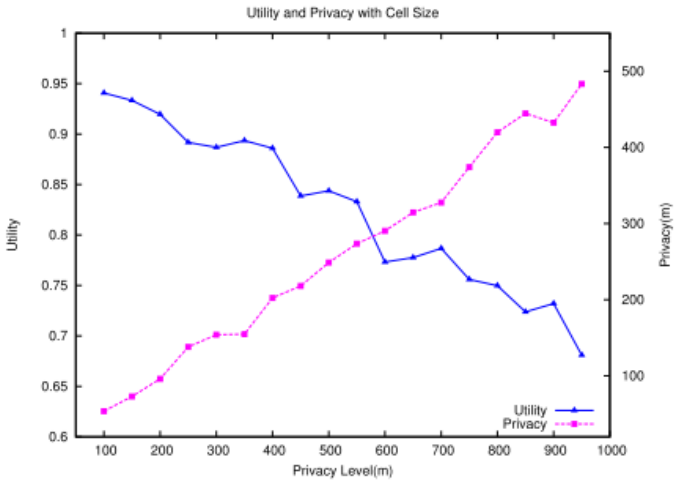
(a) Basic Grid Reference System

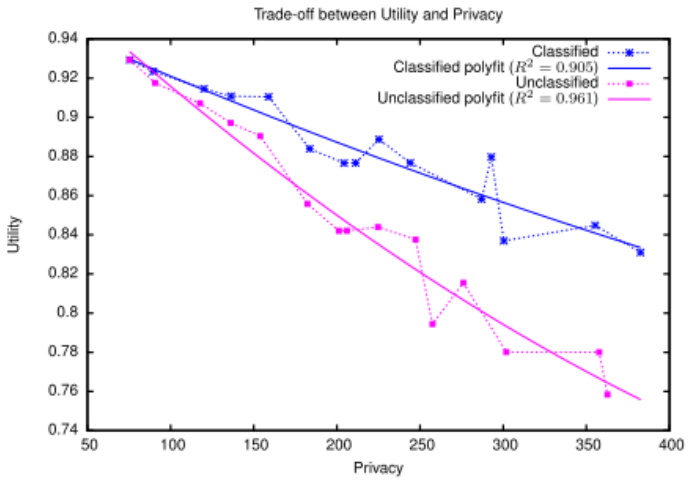
(b) Classified Grid Reference System

How to mitigate the privacy threat by LBSNs?

Privacy vs. Utility

- Any obfuscation technique will reduce the users' utility.
  → Trade-off





(a) Relationship of Utility/Privacy with Cell Size

(b) Comparison of Utility/Privacy Trade-offs

How to mitigate the privacy threat by LBSNs?

Privacy vs. Utility – Possible Solution

- Let the user decide
  → different location privacy protection preferences
  → Classify locations into different categories

- Give the most frequent visited locations a higher privacy protection

- Use non-uniform grid reference system

Critique

- Possible trade-off solutions may be weak
  → Use it or leave it
- Simultaneous use of metric and imperial system
- Was the identification success of the users really that high as the authors claim?

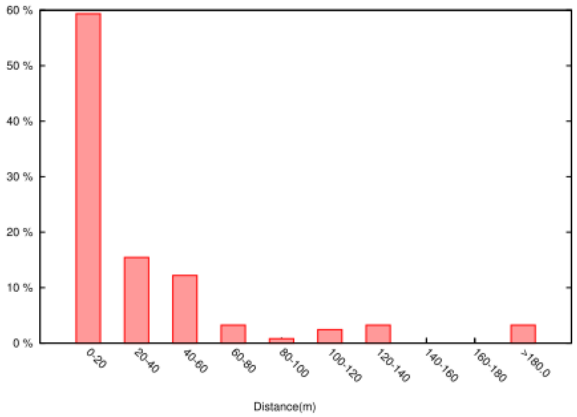
and Implications

- One location leakage of an LBSN-User is not a threat but the combination with other identifications measures can evolve to a serious threat.
- Users will not necessarily protect their own privacy if they need to trade-off utility.

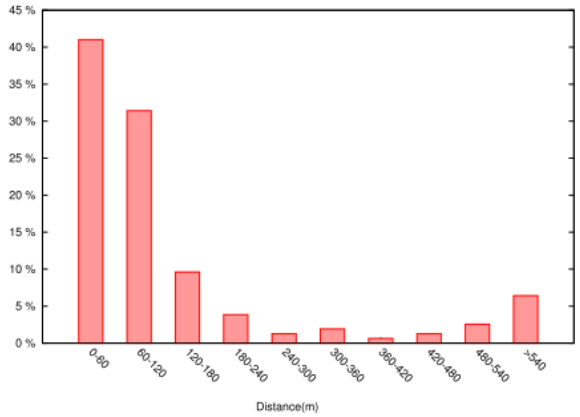
Critique - Real-World Experiment

350 reports and attacks from 30 volunteers in comparison:

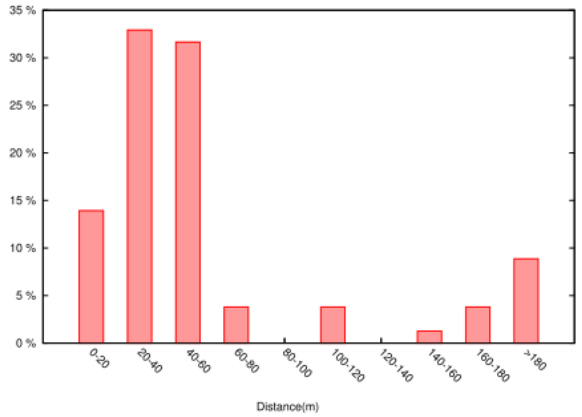
→ Localization differs in **Accuracy**



(a) Momo's Localization Accuracy



(b) Skout's Localization Accuracy



(c) Wechat's Localization Accuracy