Smartphone-Based Crosswalk Detection and Localization for Visually Impaired Pedestrians

Vidya Murali, Ph.D.
James Coughlan, Ph.D.

THE SMITH-KETTLEWELL
EYE RESEARCH INSTITUTE

Traffic intersections

Traffic intersections are dangerous for everybody, but especially for **blind** and **visually impaired** persons

Orientation and **guidance**

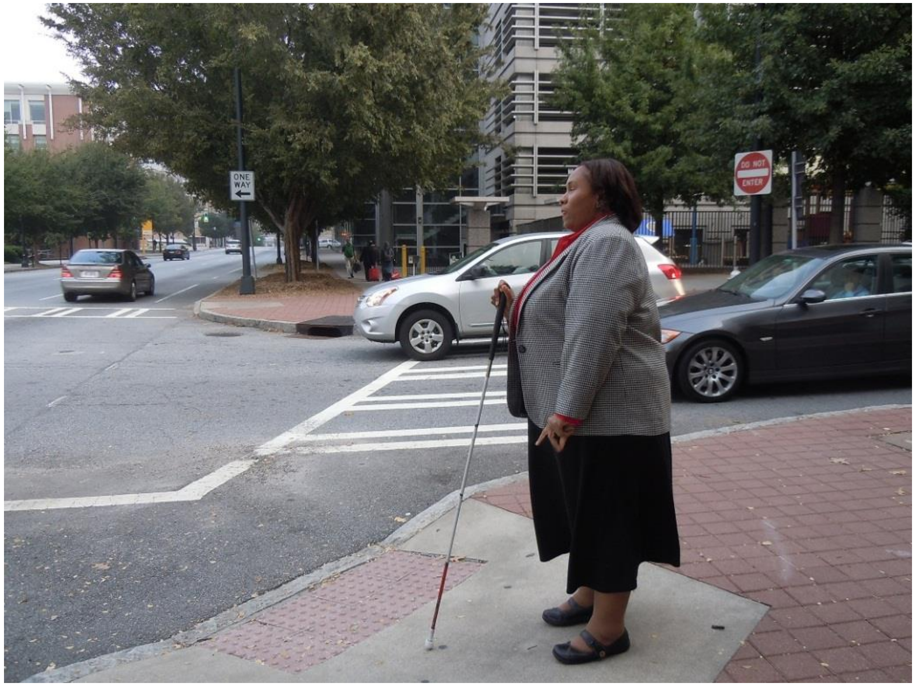
towards, and across,

intersections are sorely needed

Existing adaptations/techniques

Cane

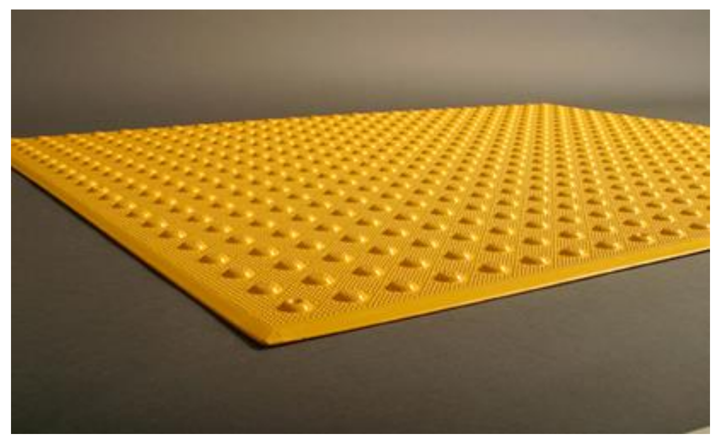
- can help traveler find curb or curb cut



Existing adaptations/techniques

Detectable warning surfaces

- e.g., bumps on curb ramps



Existing adaptations/techniques

Aural cues

- listen to traffic sounds to infer when to cross an intersection

Existing adaptations/techniques

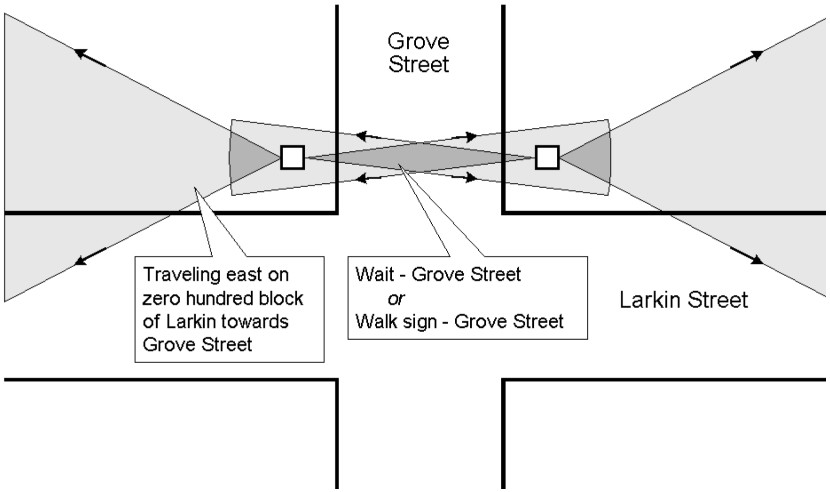
Accessible Pedestrian Signals (APS)

- indicate (e.g., chirping sound) when walk light is illuminated

Existing adaptations/techniques

Talking Signs

- Active beacons at selected landmarks and signs
- Information broadcast from infrared transmitters and converted to voice output by hand-held receiver



Existing adaptations/techniques

GPS

- Can localize user to within about 10 m, i.e., “what intersection am I at?”
- Available on smartphones
- Very useful for blind travelers

Existing adaptations/techniques

Specific apps

- Sendero GPS LookAround: find nearby streets, etc.
- Intersection Explorer: explore neighborhood offline
- Intersection: find nearest intersection, etc.

Crosswatch project: guidance at traffic intersections

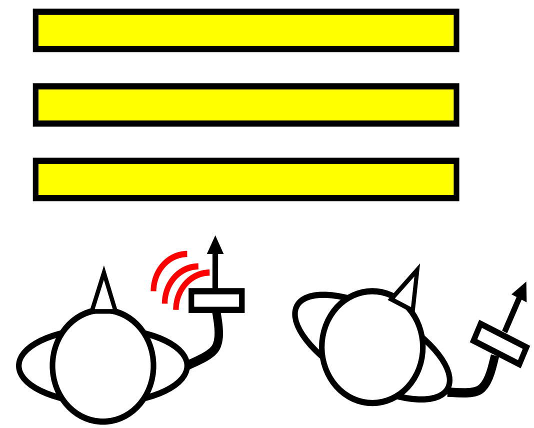
Crosswatch goal: address need for real-time information based on traveler's current location, *without needing to add physical infrastructure*

Past work relied on computer vision to perform two functions:

(a) Detect crosswalk and determine user's location/orientation relative to it

(b) Detect walk light

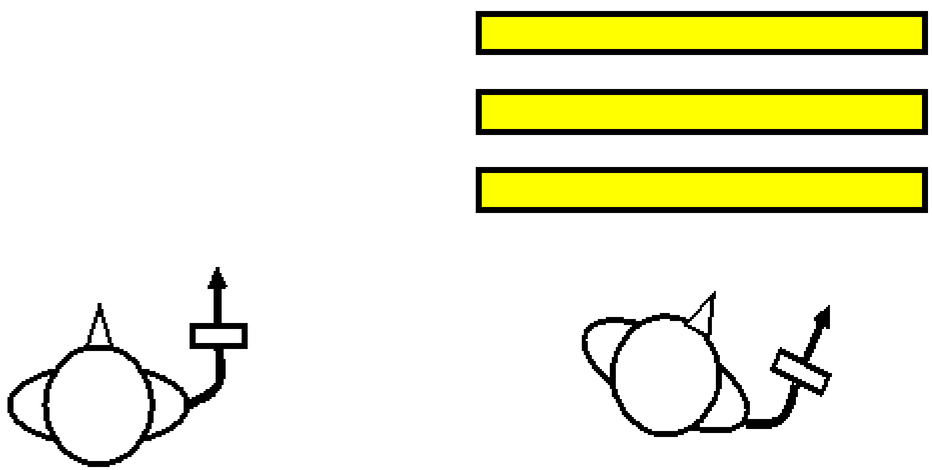
(a) Past work – localization



System finds crosswalk if there is one, and beeps to inform user

(a) Past work – localization

Note: two ways to be mis-aligned:



1.) Parallel but far away

2.) Close but not not parallel

(b) Past work – walk light detection

Zoom in →

Detect walk light in real time and report whenever it's illuminated

Recent work

Traveler needs 3 kinds of information about intersection:

What: type, layout of intersection

Where: location and orientation relative to crosswalks in real time

When: status of Walk light (or other traffic lights) in real time

Recent work

Original concept: rely solely on computer vision to recognize everything “from scratch”

Recent work

Problems:

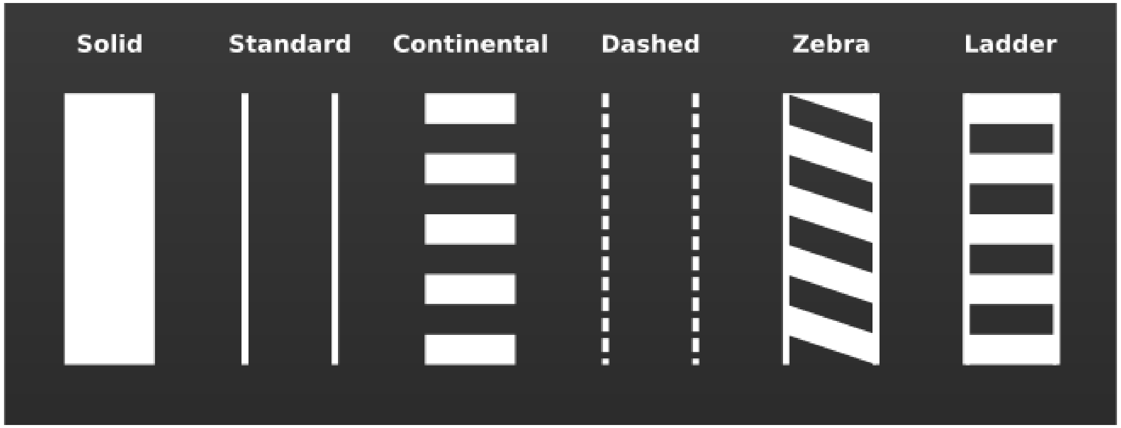
- enormous variability in the appearance of intersections, walk lights, etc.
- view from traveler's camera is very limited

Recent work

E.g., median strip: very difficult to detect reliably without accurate depth information

Recent work

E.g., many different types of crosswalk markings



Recent work

Solution: augment computer vision with sensors and other information (work done with Dr. Vidya Murali)

- **GPS** determines which intersection user is standing at

- **GIS** (geographic information system) database contains detailed information about this intersection, including a **template** of the intersection

Recent work

Armed with GPS and GIS, the traveler can determine lots of information (even without computer vision), including:

- Intersection type (four-way; T junction, etc.), presence of median, etc.
- Traffic controls, presence of walk lights, presence and location of pushbutton, etc.

Recent work

Computer vision still essential for two pieces of information:

(a) user's precise location and orientation in intersection

(b) status of lights (walk or traffic lights)

New results: localization

- GPS localizes user to about 10 m: enough to determine the nearest intersection but not necessarily which corner (let alone detailed location relative to crosswalk)
- New: use computer vision to localize more precisely in both the x and y dimensions (latitude and longitude)

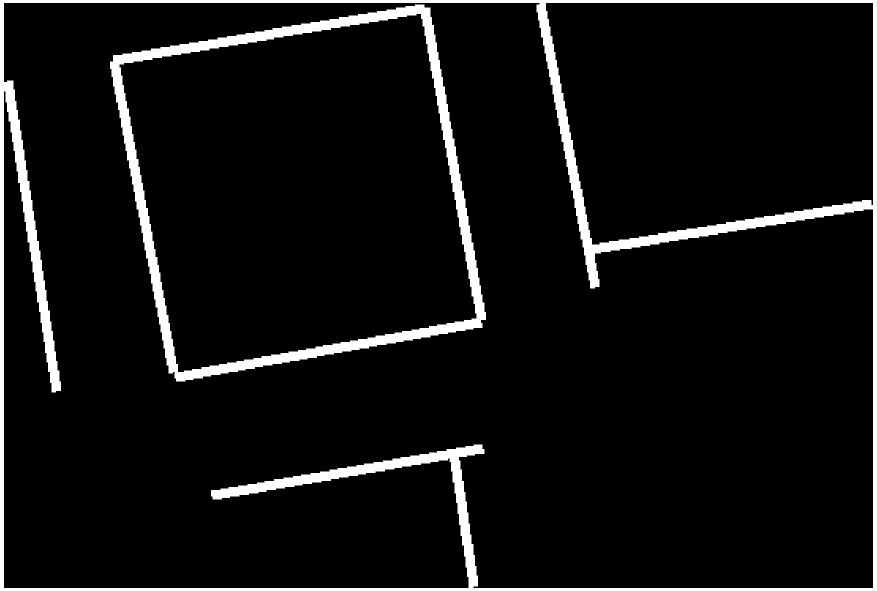
New results: localization

- By contrast, past work only estimated location along one dimension, the direction perpendicular to the crosswalk corridor

Localization

Google satellite image

Template of intersection (zoomed in), constructed manually



Localization

Where exactly am I in intersection?

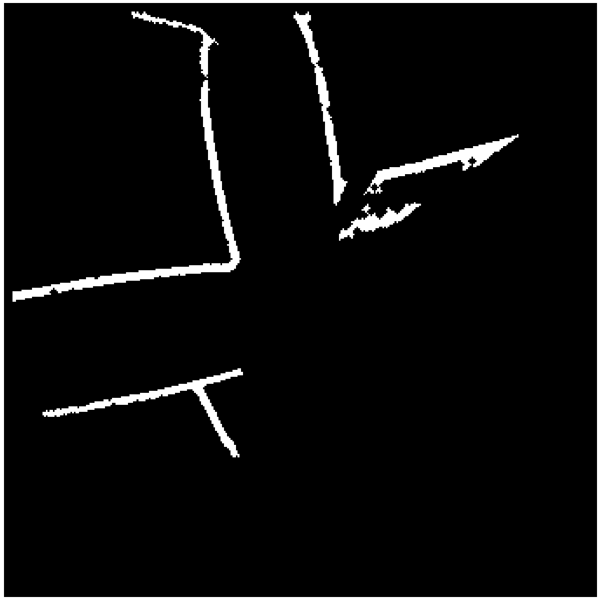
(a) Panorama acquired by smartphone: doesn't require user to be able to see anything

UI constrains orientation to be horizontal, to make it easy to aim camera

Localization

Where exactly am I in intersection?

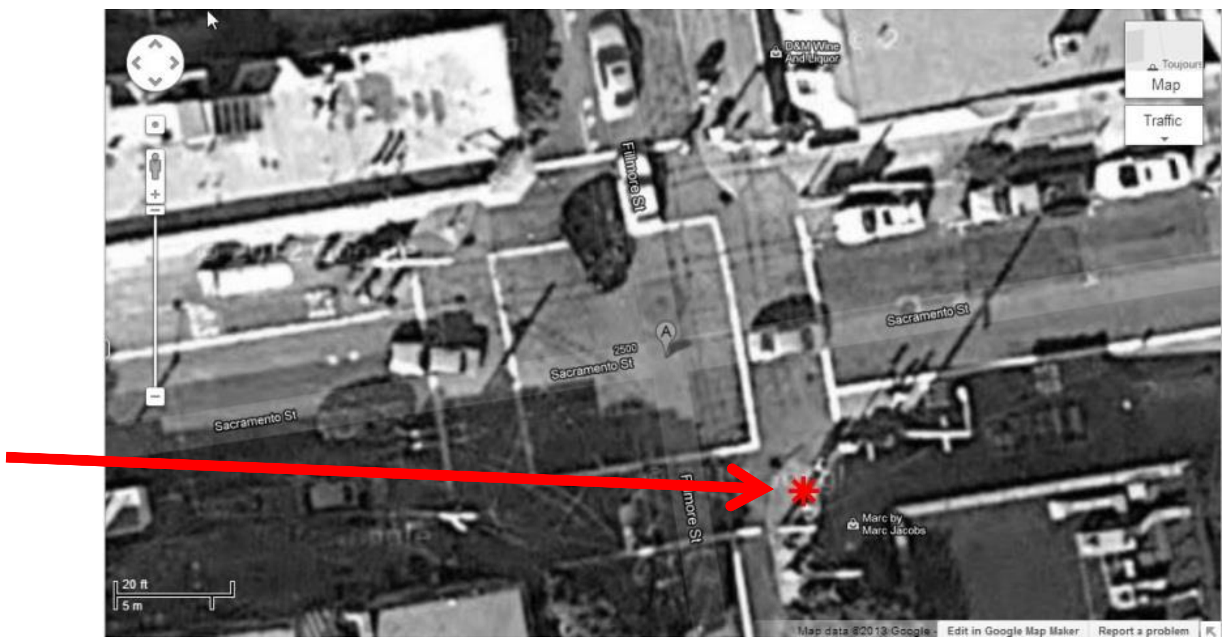
(b) Aerial reconstruction of nearby intersection, followed by segmentation of stripe features



Localization

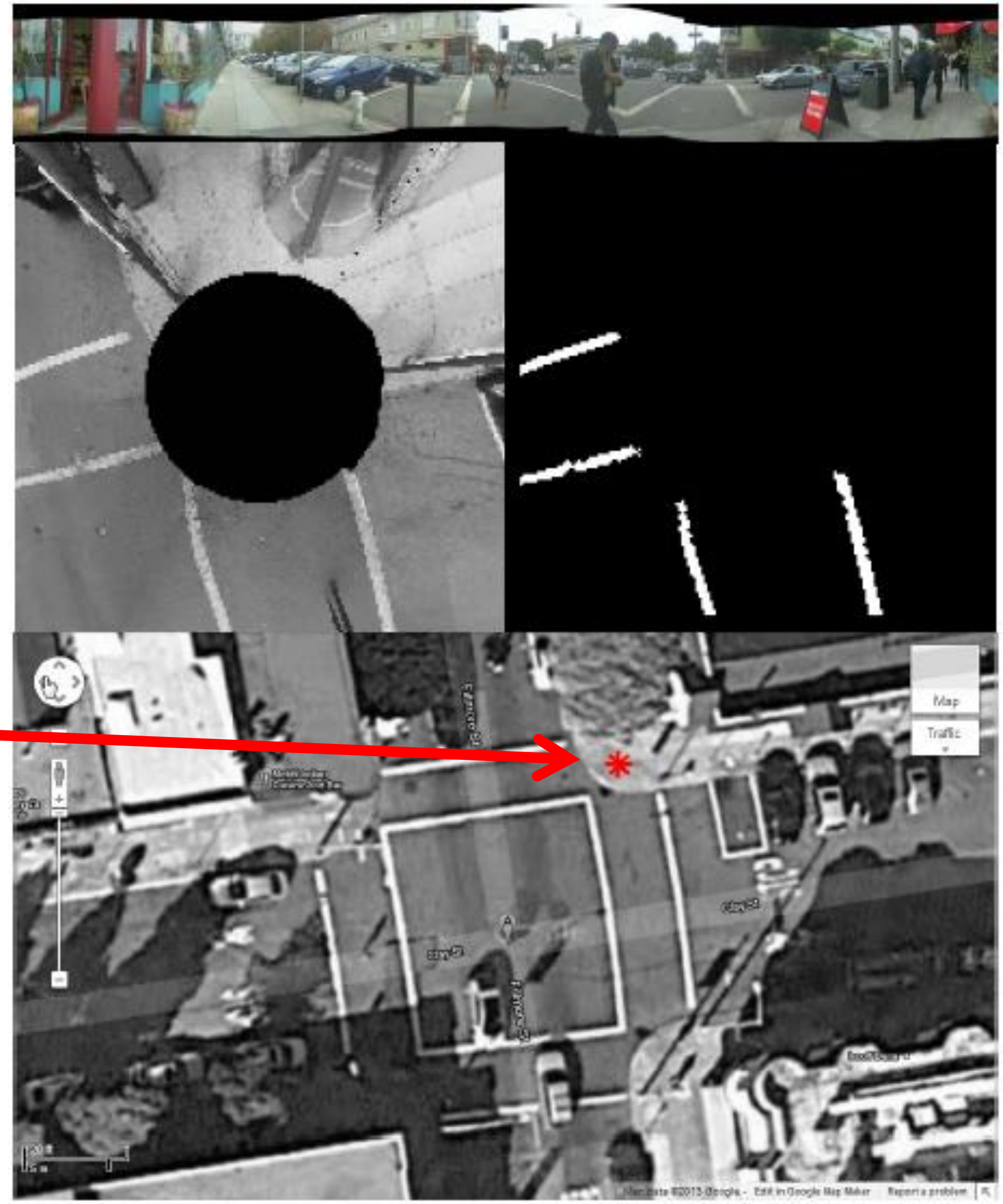
Where exactly am I in intersection?

(c) Match aerial reconstruction with template to estimate current location (red dot)



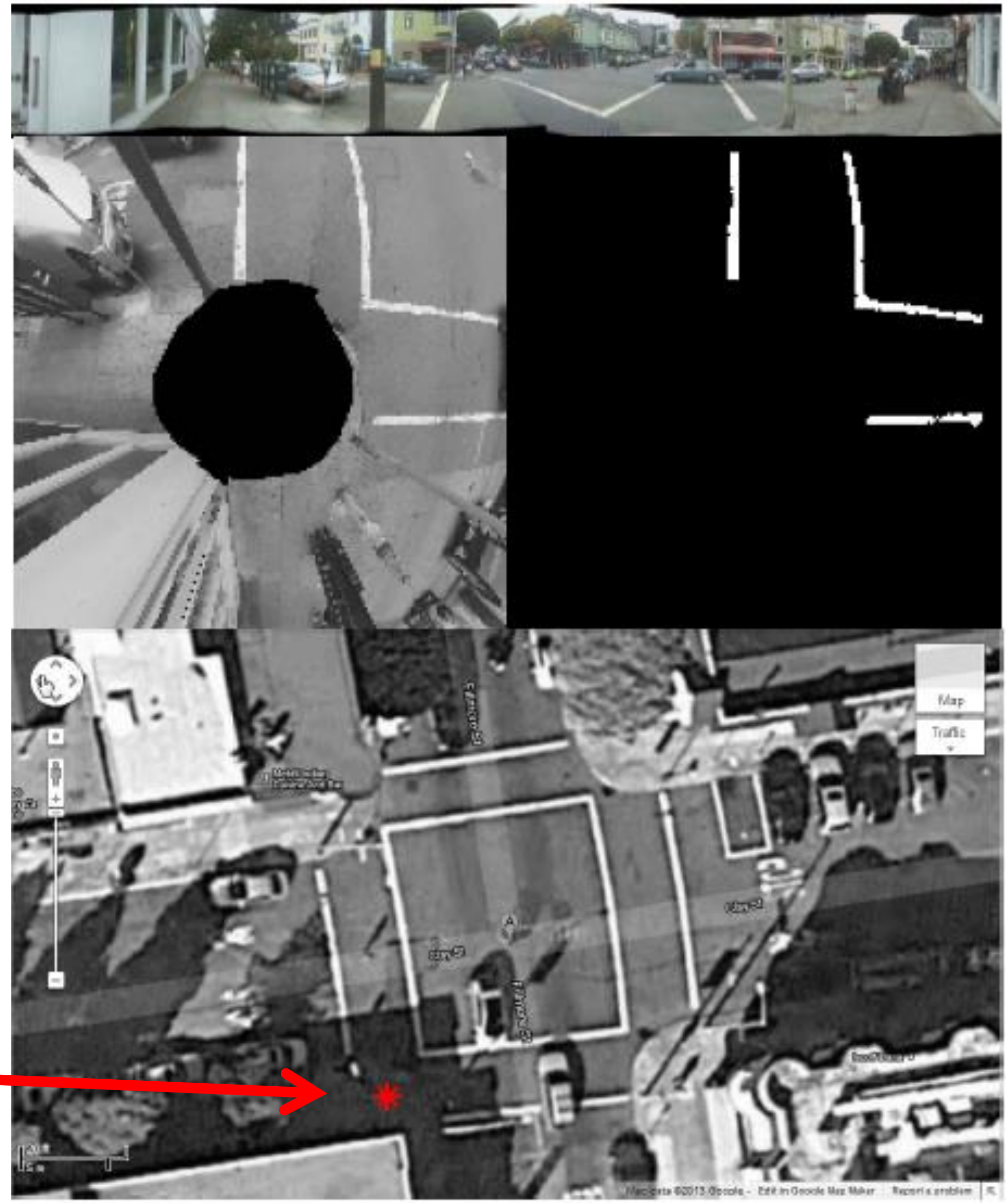
Localization

Example 2



Localization

Example 3



Note #1

- Model of intersection is 2D
- Benefits of 2D model:
  - simple, easy to acquire (e.g., satellite imagery), low memory/complexity;
  - some features are close to user → strong localization info.

Note #1 (con't)

Disadvantages of 2D:

- Vehicles/people often occlude features in image (though panorama often erases moving occluders); peeling paint can degrade features

Thus, may augment with 3D models in future

Possible benefits of 3D:

- Many 3D features (e.g., on buildings) less likely to be occluded or degraded
- Many more features can be used for greater robustness

Note #2

- In future, template constructed by crowdsourcing
- Crowdsourcing will allow visually impaired travelers to add many different kinds of information (e.g., how difficult is intersection to traverse? Road construction; etc.)

Note #3

In future, users may not have to hold smartphone to pan camera left and right – wearable cameras like Google Glass have the potential to greatly facilitate computer vision applications for visually impaired users

Conclusion

- Computer vision can be used to extract useful real-time information for blind/visually impaired travelers at intersections
- Technologies such as GIS and crowdsourcing are promising ways to **augment** and **complement** computer vision

Thanks to…

John Brabyn (S-K)

Richard Ladner (Univ. Wash.)

Megan Lawrence (S-K)

Roberto Manduchi (UC Santa Cruz)

Josh Miele (S-K)

Huiying Shen (S-K)

Ender Tekin (S-K)

David Vásquez (S-K)

Volunteer participants in experiments

Funding from NIH, NIDRR and Smith-Kettlewell