CSE 4/60373: Multimedia Systems

- Outline for today
  - Kulkarni, P., Ganesan, D., Shenoy, P., and Lu, Q. SensEye: a multi-tier camera sensor network. In Proceedings of the 13th Annual ACM international Conference on Multimedia (Hilton, Singapore, November 06 - 11, 2005)
  - Liu, X., Corner, M., and Shenoy, P. SEVA: sensor-enhanced video annotation (best paper award @ACM MM)

System setup

- Use video sensors to track suspects
- Steps:
  - Detect objects: know that an object is there
  - Recognize objects: See if it interesting
  - Track objects: Track its motion
- Approach 1: Single tier
  - One sensor that can perform all the tasks
- Approach 2: Multi-tier
  - Three tiers in this paper where each tier has increasing amounts of resources. Judiciously mix these tiers to achieve overall benefits
- Constraints:
  - Cost (reliability and coverage) and energy consumption

Applications

- Environment monitoring to track exotic animals
- Search and rescue missions
- Baby monitor (for toddlers)

- Design principles:
  - Map each task to the least powerful tier with sufficient resources (and conseve energy)
  - Exploit wakeup-on-demand higher tiers: (to conserve energy)
  - Exploit redundancy in coverage: If two camera can see the same object, then use this fact to localize the object in order to wake up the smallest set of higher tier nodes

Tier 1

- Lowest capability: Can perform object detection by using differencing between two frames (reference?)
  - CMUcam + mote: 136 ms (132 for camera), 13.4 J for mote and 153.8 J for camera
  - Cyclops + mote: 892 ms, 29.5 J
- Integrated platforms could be even more energy efficient

Platform	Type	Resources
Mica Mote	Atmega128 (6MHz)	84mW, 4KB RAM, 512KB Flash
Yale XYZ	OKI ArmThumb (2-57 MHz)	7-160mW, 32K RAM, 2MB external
Stargate	XScale PXA255 (100MHz–400MHz)	170-400 mW, 32MB RAM, Flash and CF card slots

Table 2: Different sensor platforms and their characteristics.

Tier 2

- Stargate

1 2 3 4 5 6
Time (seconds)

Mode	Latency (ms)	Current (mA)	Power (mW)	Energy Usage(mJ)
A: Wakeup	366	201.6	1008	368.9
B: Wakeup Stabilization	924	251.2	1256.5	1161
C: Camera Initialization	1280	269.6	1348	1725.4
D: Frame Grabber	325	330.6	1653	537.2
E: Object Recognition	105	274.7	1373.5	144.2
F: Shutdown	1000	153.7	768.5	768.5
G: Suspend	–	3	15†	–

Table 5: *SensEye* Tier 2 Latency and Energy usage breakup. The total latency is 4 seconds and total energy usage is 4.71 J.

† This is measured on an optimized Stargate node with no peripherals attached.

Comparison

- Multi-tier architecture is far more energy efficient with almost similar recognition ratios

Component	Total Wakeups	On Wakeup Object Found	On Wakeup No Object Found	Energy Usage (Joules)
Stargate 1	311	32	279	1464.8
Stargate 2	310	42	268	1460.1

Table 6: Number of wakeups and energy usage of a Single–tier system. Total energy usage of both Stargates when awake is 2924.9 J. Total missed detections are 5.

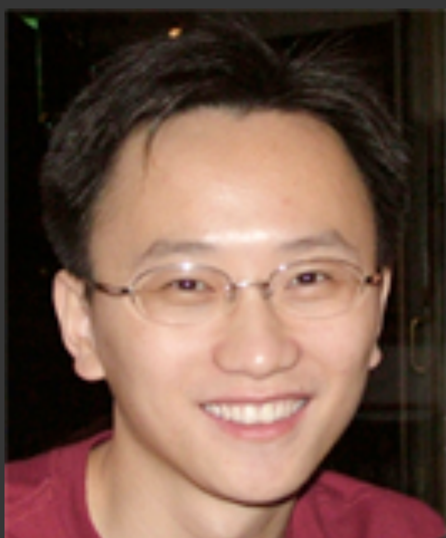
Component	Total Wakeups	On Wakeup Object Found	On Wakeup No Object Found	Energy Usage (Joules)	Cyclops Expected Energy(J)
Mote 1	304	15	289	50.7	8.96
Mote 2	304	23	281	50.7	8.96
Mote 3	304	27	277	50.7	8.96
Mote 4	304	10	294	50.7	8.96
Stargate 1	27	23	4	127.17	127.17
Stargate 2	29	25	4	136.59	136.59

Table 7: Number of wakeups and energy usage of each *SensEye* component. Total energy usage when components are awake with CMUcam is 466.8 J and with Cyclops is 299.6 J. Total missed detections are 8.

Discussion

- The claim is not that they invented new recognition algorithms
  - On the other hand, we need recognition algorithms which may not be as accurate as the state of the art but can fit into small devices and run for long durations

SEVA: Sensor-Enhanced Video Annotation



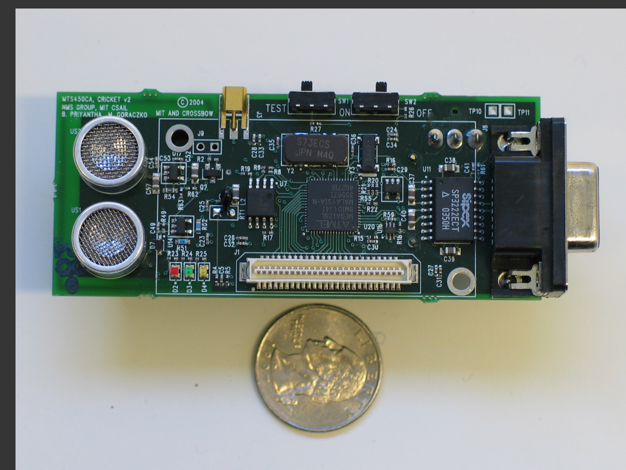
Xiaotao Liu,

Mark Corner, Prashant Shenoy

University of Massachusetts, Amherst

Pervasive Sensing and Location

- We are in the midst of a very exciting time
- Rapid advances in embedded sensor technology
  - wireless, processing, storage
  - battery-powered but long lasting
  - small-sized and inexpensive
- Similar trend in location systems
  - outdoor: GPS (<10m accuracy)
  - indoor: ultrasound (cm accuracy)
  - improvements in accuracy, deployment, and cost
- Hurtling towards pervasive sensing and location-based systems



Rapid Accumulation of Content

Vide

Ima

N

1

Lead

h

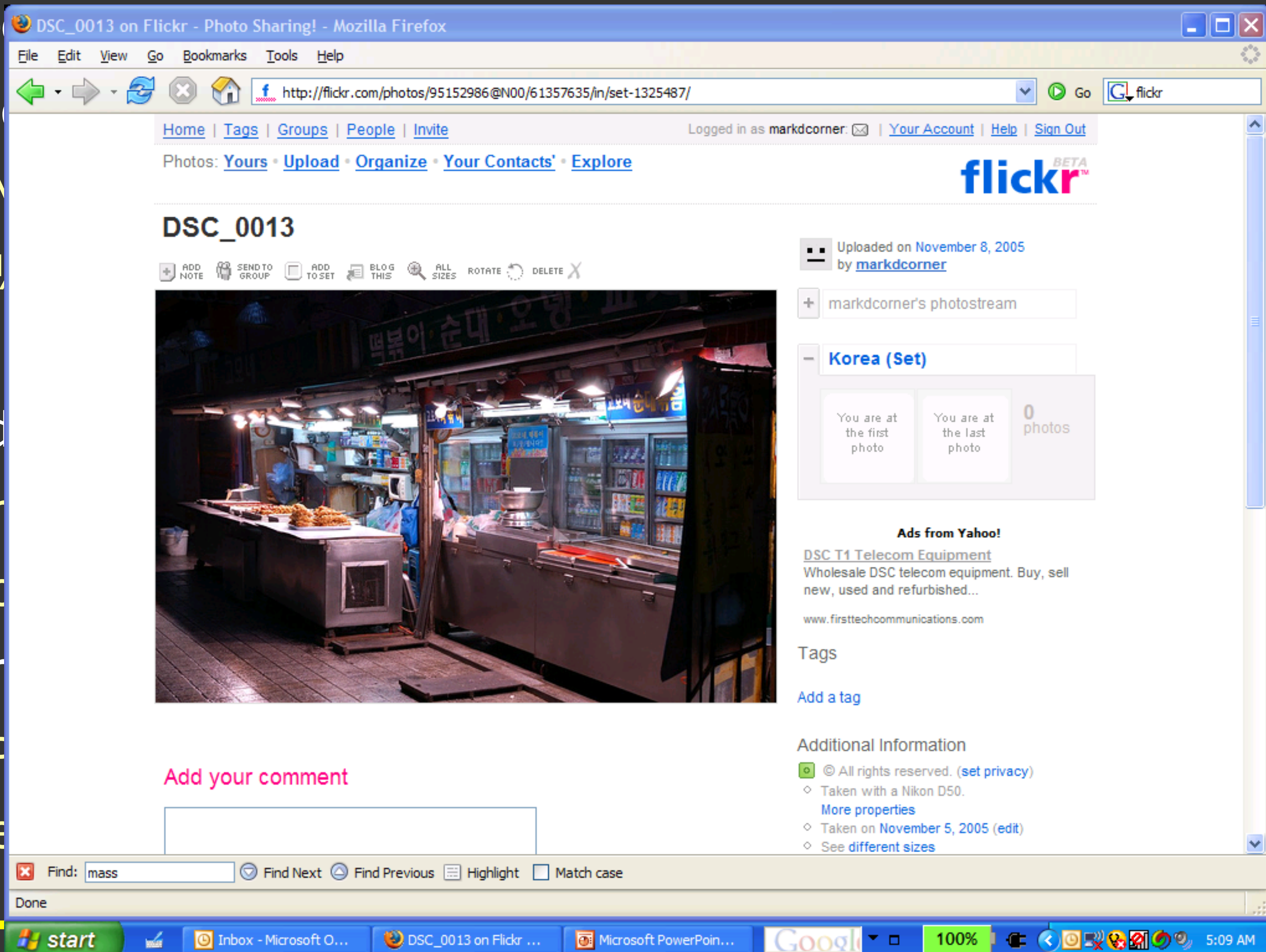
c

A pr

F

e

008



Content Organization and Retrieval

Organization and retrieval is the key to making multimedia useful

- depends on knowing what/where/when/who of my videos and pictures

Google, Flickr, .. all depend on manual or inferred text annotations

- annotations may be incomplete or inexact
- leads to poor precision and/or recall

Content-based retrieval and image recognition aren’t 100% accurate

Google image search:
“Xiaotao Liu”

Sensor Enhanced Video Annotation

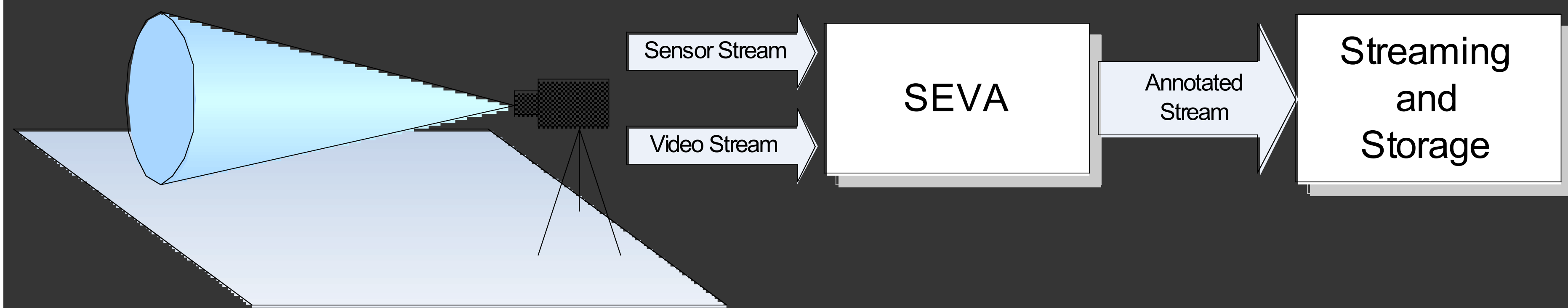
Our solution: Sensor Enhanced Video Annotation (SEVA)

- objects should be self identifying and videos self-annotating
- records the identity and locations of objects along with video
- does this frame-by-frame or for every photo

Video camera produces media stream

Camera queries nearby objects for identity and location

- produces a parallel sensor stream



Key Challenges

Mismatch in camera coverage and sensor range

- objects within radio range may not be visible

Objects, camera, or both may be highly mobile

- objects will move in and out of the field of view

Limitations of constrained sensors

- sensors can’t respond to every frame
- need slow query rate to scale system

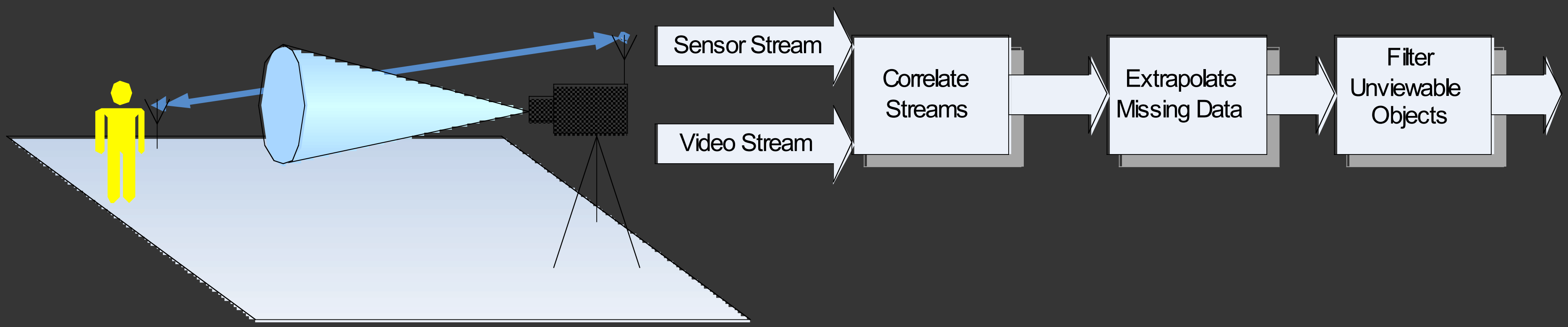
Limitations of location system

- location systems don’t update at same rate as video

SEVA Operation

SEVA operates in a series of stages:

- correlate data from sensor stream with video stream
- extrapolate and predict the locations of objects when missing
- filter out any unviewable objects from the annotations



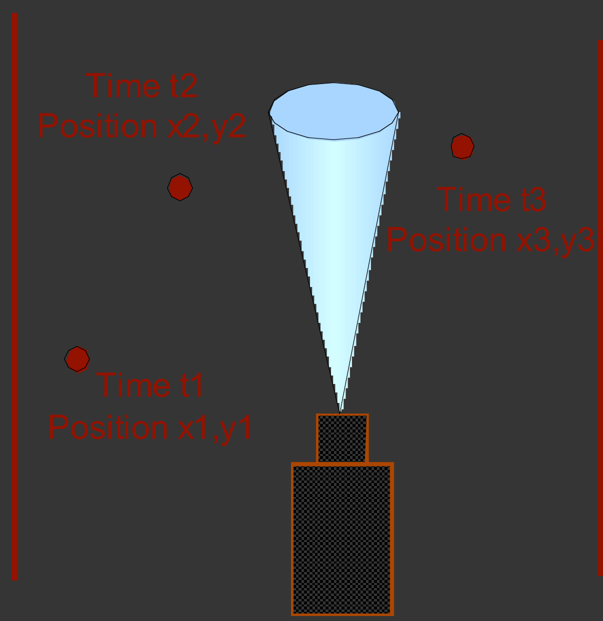
Stream Correlation

- SEVA must correlate sensor responses with frames
  - sensors may respond desynchronized with current frame
  - due to processing delays, power management, link-layer
- Two modes of operation:
  - synchronized clocks, but often not feasible in sensor
  - approximate based on MAC layer delays and processing
  - we currently use the later
- Produces a time-synched stream of video and locations

Extrapolation and Prediction

Not every frame contains a location for every object

- want to maintain object information for every frame
- objects may have entered/left view between responses
- similarly, the camera may have moved, or both



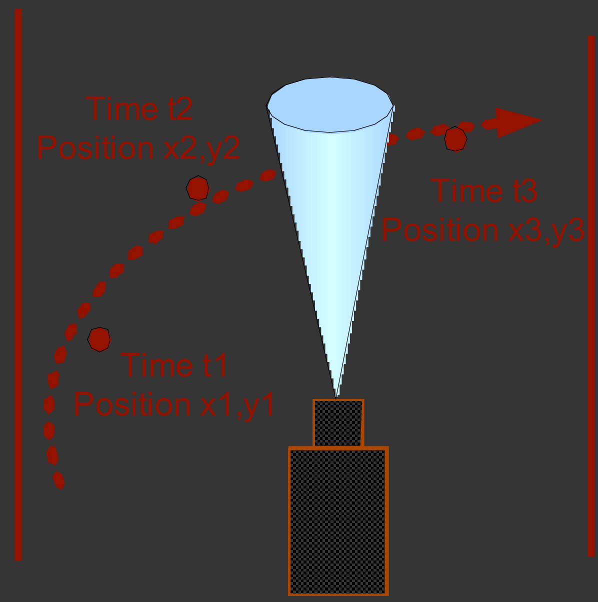
Extrapolation and Prediction

Apply a least squares regression technique to find object path

Search kth degree polynomials, of increasing degree, for each axis

$$X(t) = a_0 + a_1 t + a_2 t^2 + \ldots + a_k t^k$$

Can extrapolate or predict location for every frame



Filtering and Elimination

Need to determine which objects are visible in each frame

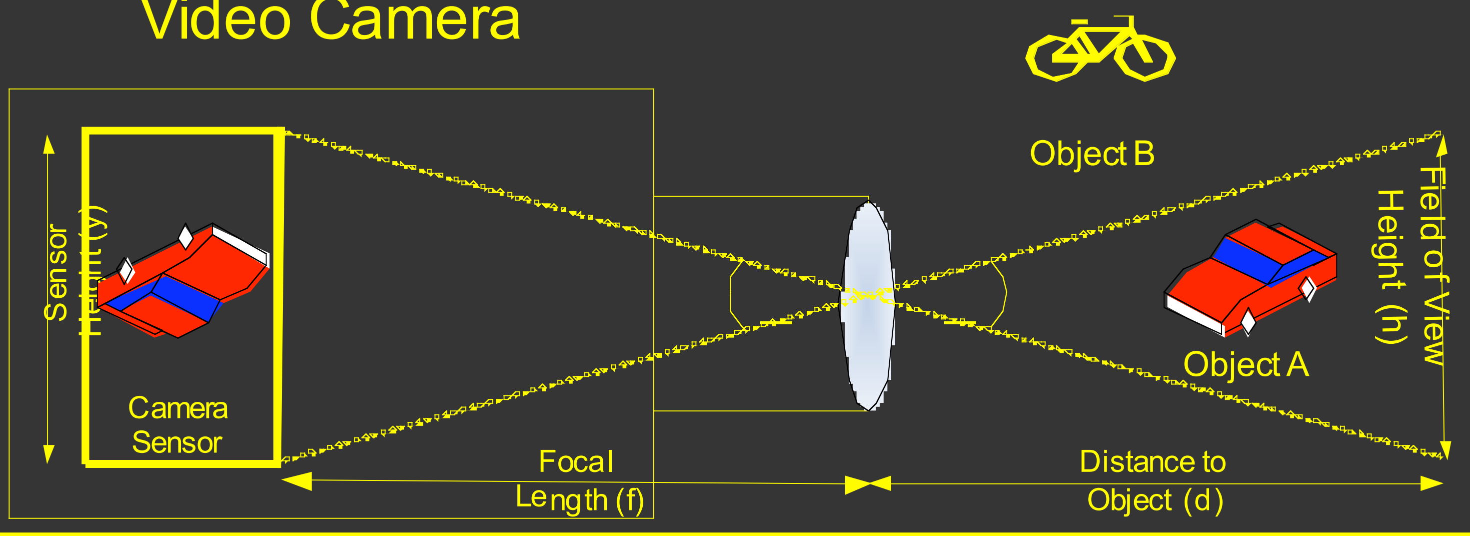
Use object locations with optics model

- combination of the focal length and sensor size
- does not take obstructions into account: bug or feature?

What about partially viewable objects?

- visibility is in the eye of the beholder

Video Camera



Prototype Implementation

To provide a test platform we constructed a prototype

Based on a Sony Vaio laptop

- contains a 320x240, 12fps, CMOS based camera

Two location systems

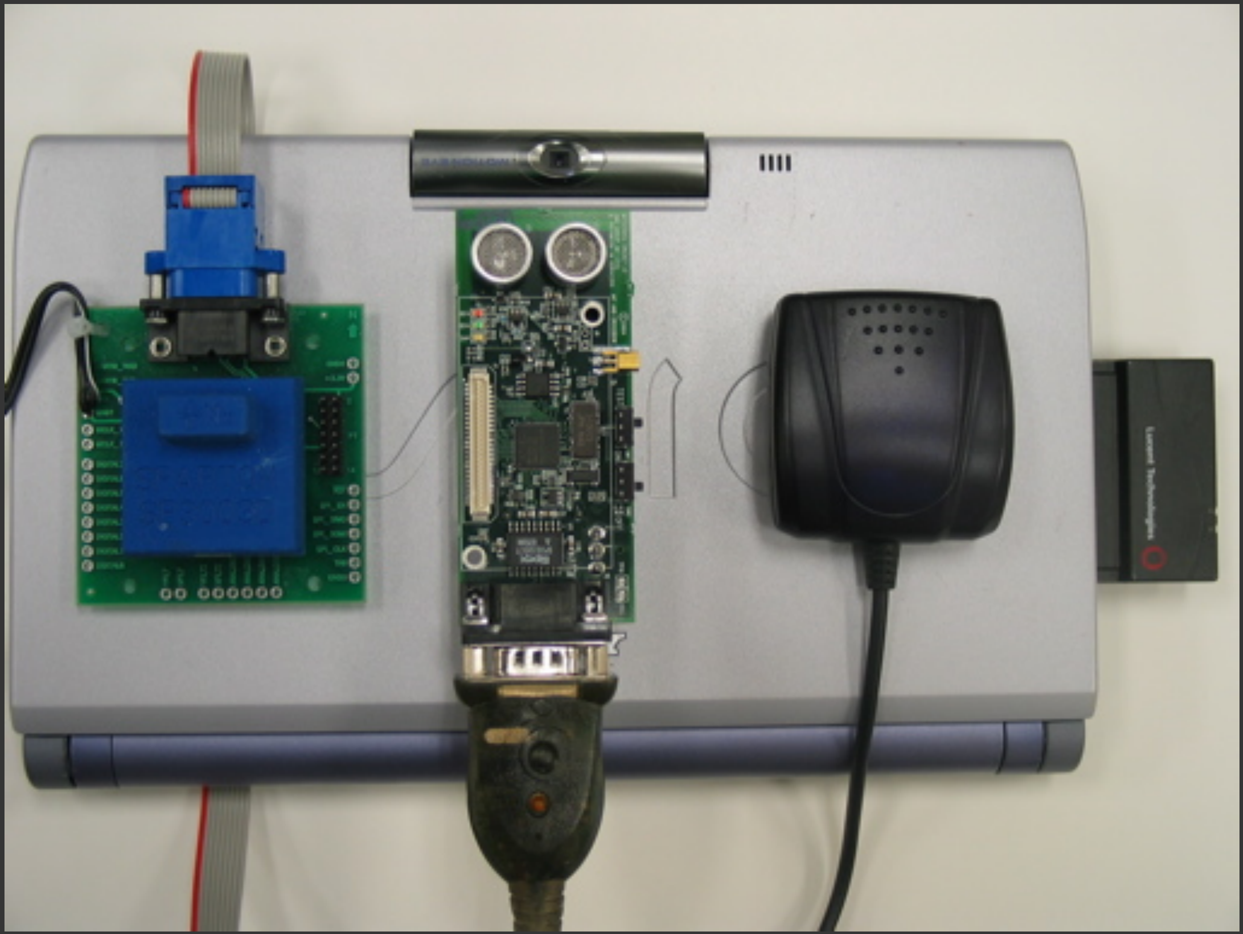
- outdoors: GPS w/land-based correction (accuracy: 5-15m)
- indoors: Cricket ultrasonic location system (accuracy: 3cm)

Augmented with digital compass for orientation

Pervasive Identification System

- outdoors: 802.11 ad-hoc mode
- indoors: sensor wireless interface

Prototype Implementation (cont.)



Laptop with: Digital Compass, Cricket Ultrasound, Camera, GPS, WiFi

Evaluation

In evaluating SEVA we sought to answer several key questions:

How accurate is SEVA is tagging frames?

- static experiments
- moving objects/camera: stresses extrapolation system
- report results from Ultrasound location system (GPS in paper)

How well does SEVA scale?

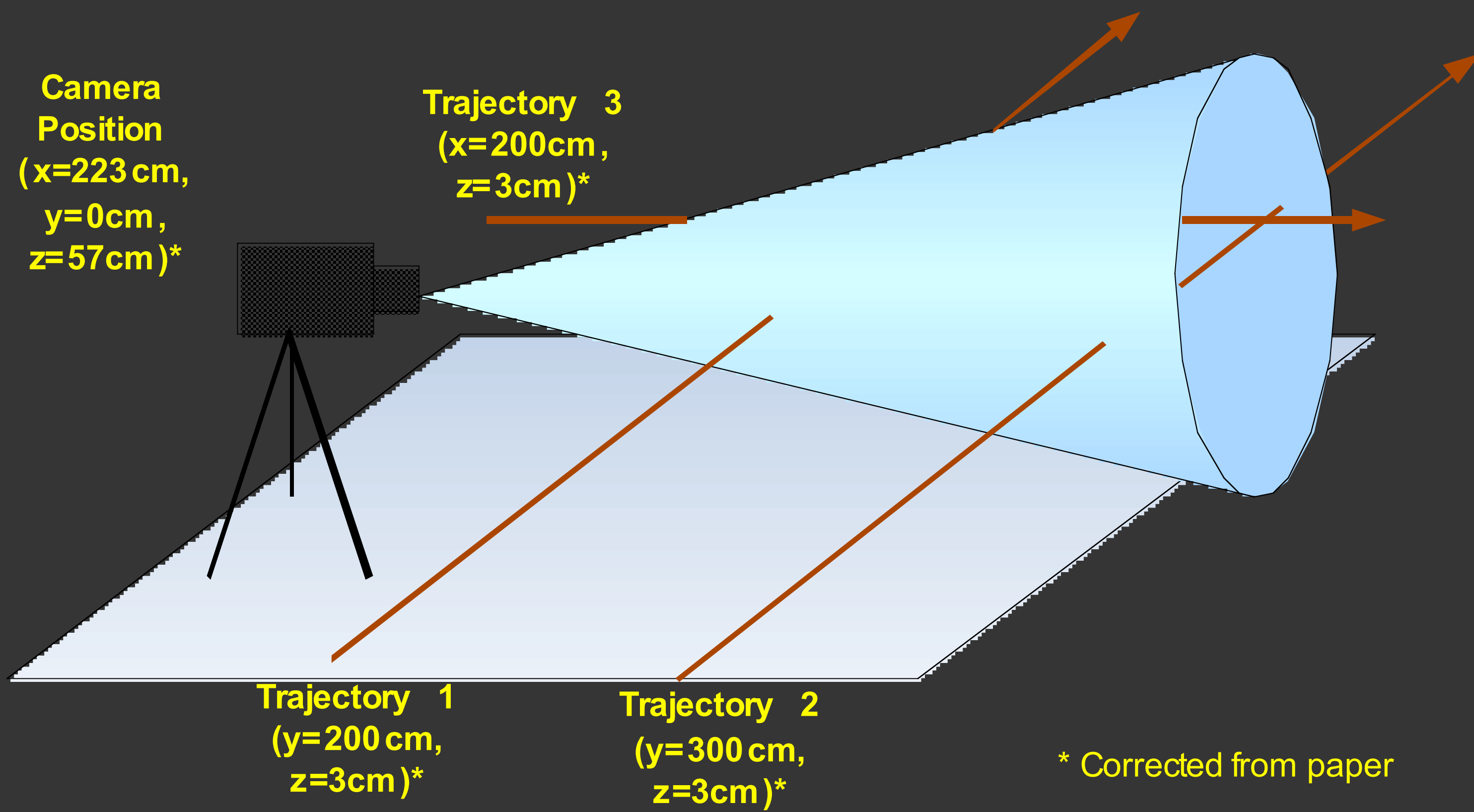
What is SEVA's computational overhead?

Static Objects

Place object (film canister) along trajectories through the viewable area

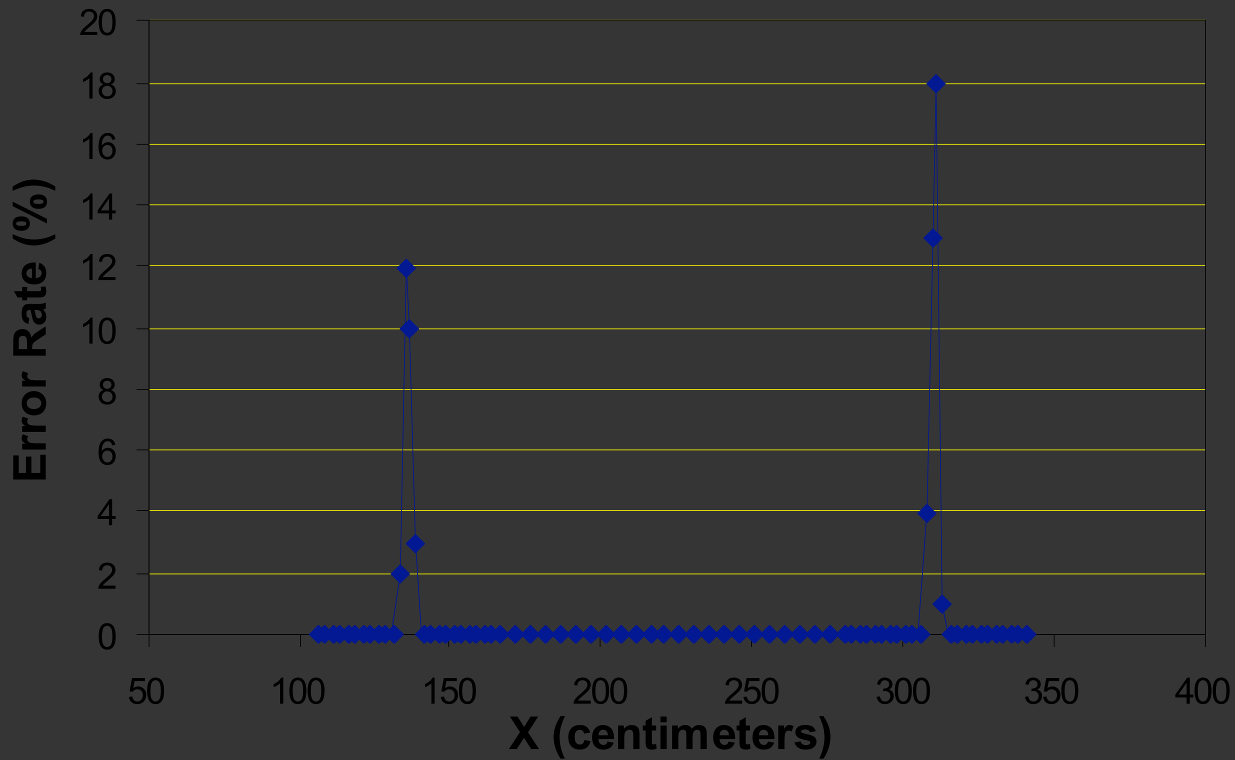
Take 100 frames at each location, and manually verify accuracy

- error rate is the sum of false positives and negatives



* Corrected from paper

Static Objects



Errors only occur near the viewable boundary

- due to inaccuracies in location and filtering

The fact that the object is very small represents a worst case

- any object wider than 20cm will have zero error rate

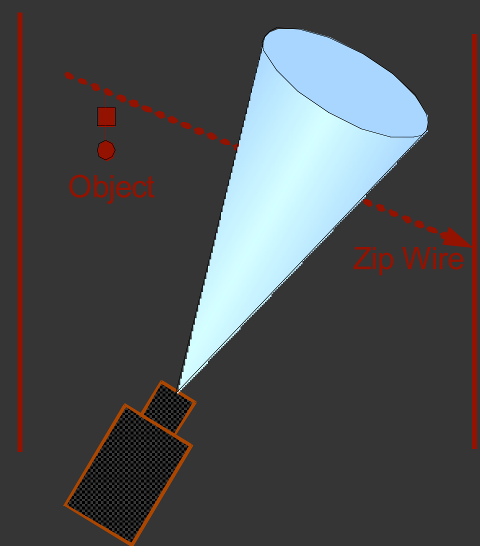
Dynamic Objects

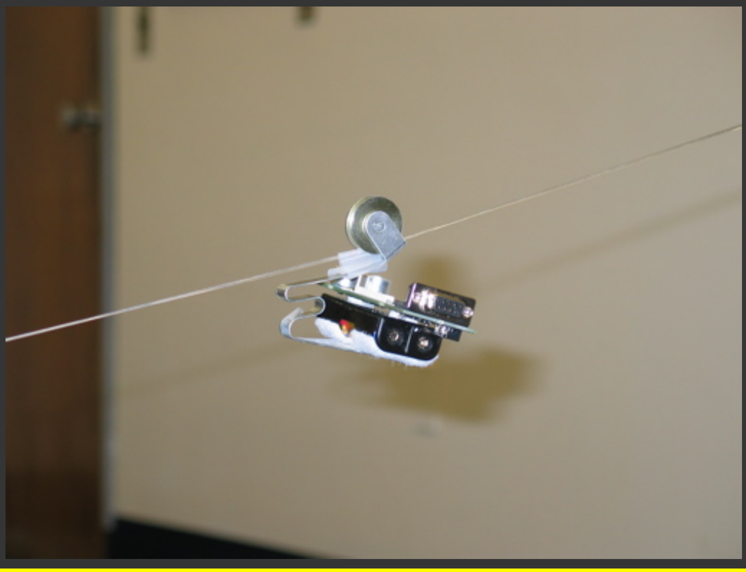
Attach object to a pulley and “zip wire”, crosses view at different speeds

Measures the effectiveness of our extrapolation method

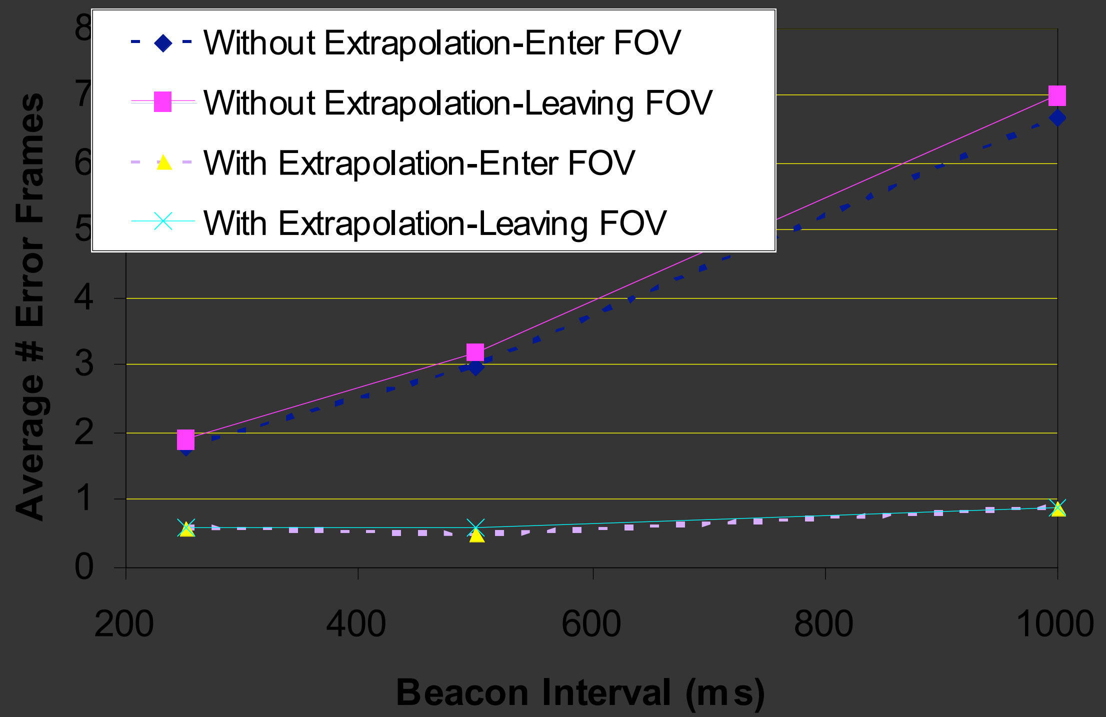
We compare system with and without extrapolation

- vary the response frequency: measure of scalability and robustness
- error rate is reported as the number of frames mislabeled
- report error rates for entering and leaving field of view





Dynamic Objects (avg=1.5 m/s)



System with extrapolation mislabels less than one frame

Non-extrapolated system mislabels up to seven frames

- SEVA corrects for missing responses
- or scales well to larger number of objects

Random Dynamic Experiment

"Zip Wire" is a linear path

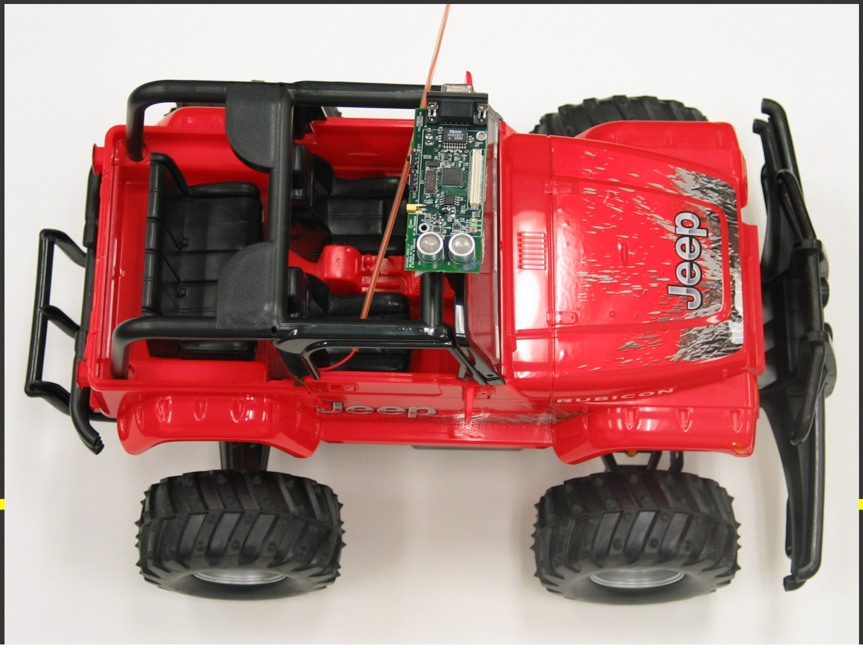
- provides repeatability, but straightforward extrapolation

Instead try experiments with "random" movement

- stresses higher-order regression

We drove a remote control car in and out of the camera's view

On average, SEVA only misidentifies 2 frames at boundaries



Scalability and Computation

System currently scales well to 10 moving objects

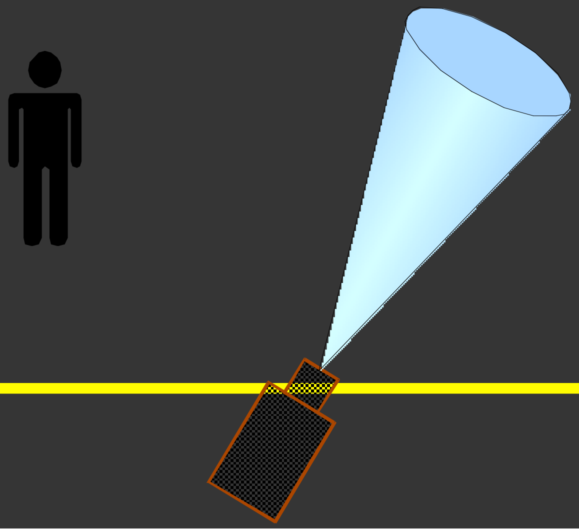
- limited by the available bandwidth of sensors

Computational load measured on laptop

- ultrasound location: 150 μs/object
- correlation and extrapolation: 100 μs/object
- filtering: 100 μs/object
- SEVA will work in realtime on more modest hardware

Other results

- GPS accuracy is still too poor to use with SEVA
  - results in paper
  - SEVA mislabels when object is 10s of meters from viewable
  - major improvements in GPS expected
- SEVA also works with a moving camera
  - used several repeatable movement patterns
  - makes few errors (< 2 frames on average)
  - performs worst when rotating camera quickly



Related Work

- Sensor-based annotation of video:
  - records where/when camera took picture: Aizama 2004, Davis 2004, Ellis 2004, Gemmell 2002, Naaman 2003, Toyama 2003.
  - in contrast, SEVA records what and where the object was
  - system for augmenting video studio with light sensors: Su 2004
- Sensor Systems and Location
  - Hill 2002: Mote sensor platform
  - Priyantha, Chakraborty, and Balakrishnan 2000: Cricket

Conclusions

Multimedia systems must utilize new sensor/location systems

SEVA provides a system for automatically annotating video

- records what, where, and when for visible objects
- enables later retrieval, or online streaming applications

A large set of experiments demonstrates that SEVA:

- can identify visibility of static objects with a few centimeters
- can extrapolate positions even with slow beacon rates