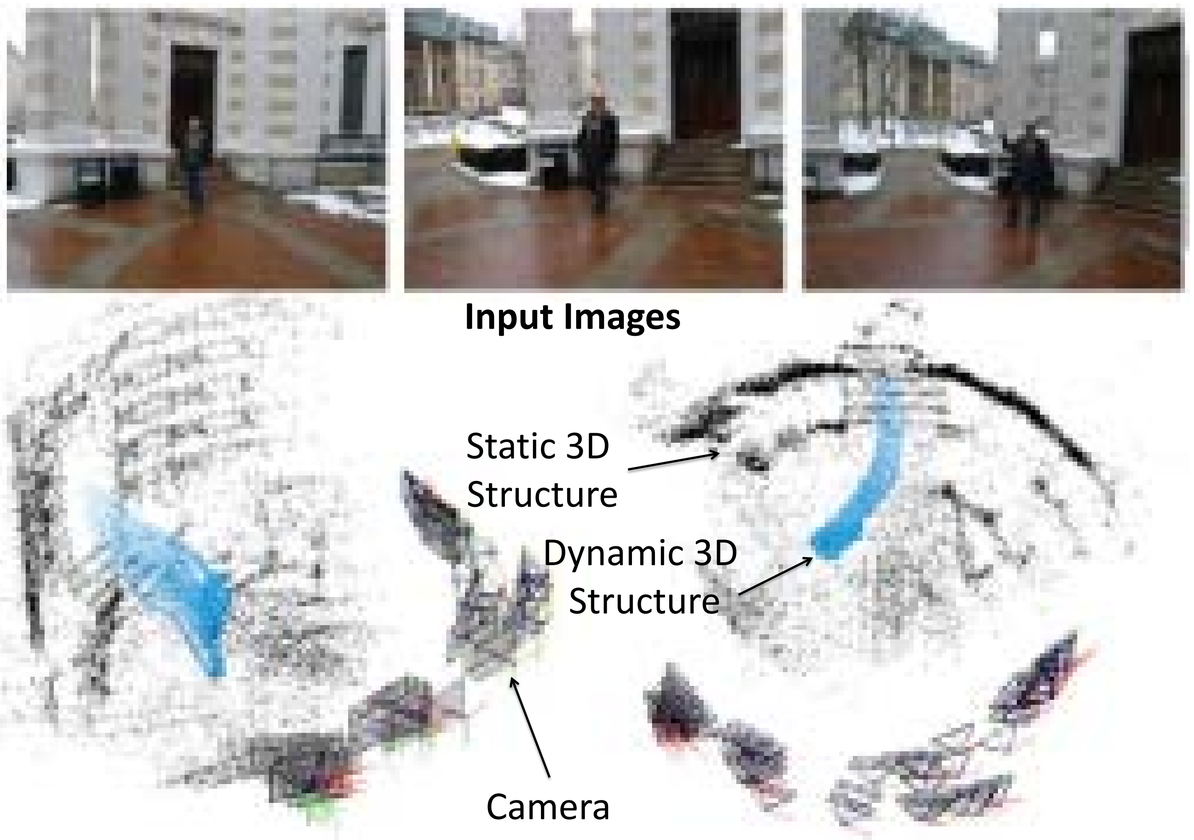
REALTIME 3D RECONSTRUCTION OF REALWORLD SCENES

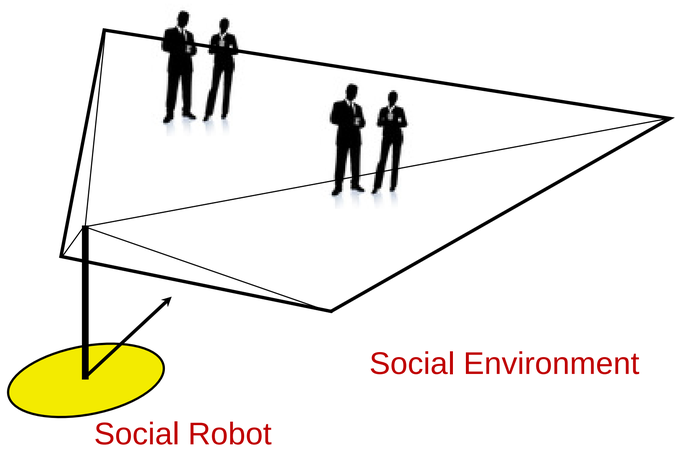
Yaser Sheikh (RI), Mei Chen (Intel), James C. Hoe (ECE)



The goal of this project is to produce a low energy, memory-bandwidth efficient embedded solution for realtime 3D reconstruction of dynamic environments from monocular video.

OVERVIEW

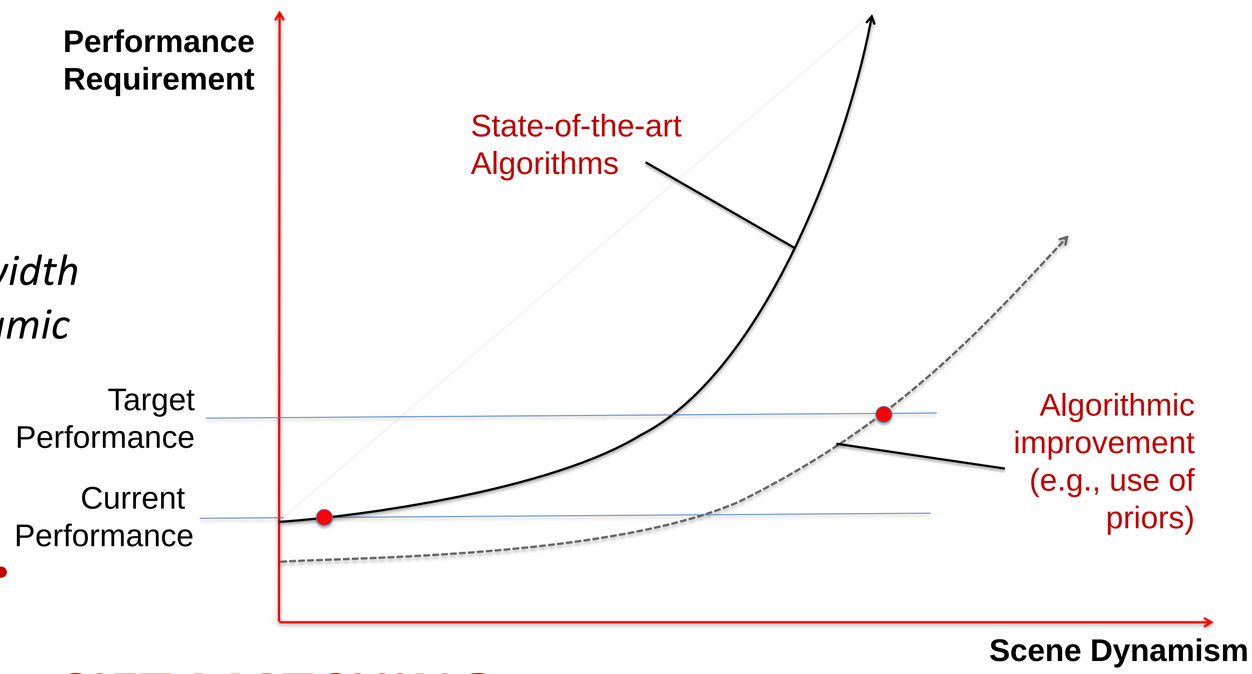
- **Problem**: Perceptually responsive system require realtime 3D knowledge of our social environment (e.g., pedestrians, cars)
- **Approach**: We will develop the theory and practice required to reconstruct, in realtime, the 3D scene structure and 3D camera motion from monocular video
- **Impact**:
  - Enables technology for perceptually aware robotics
  - Allows robots to safely co-habit environments with humans
- **Applications**:
  - 3D video "tagging": What is every pixel looking at
  - Collision avoidance: Spatial proximity of objects
  - Human robot interface: Safely co-habit the human world
- **Example of Long-term Success**: Every video camera with embedded capability to 'tag' videos in 3D



RESEARCH CHALLENGE

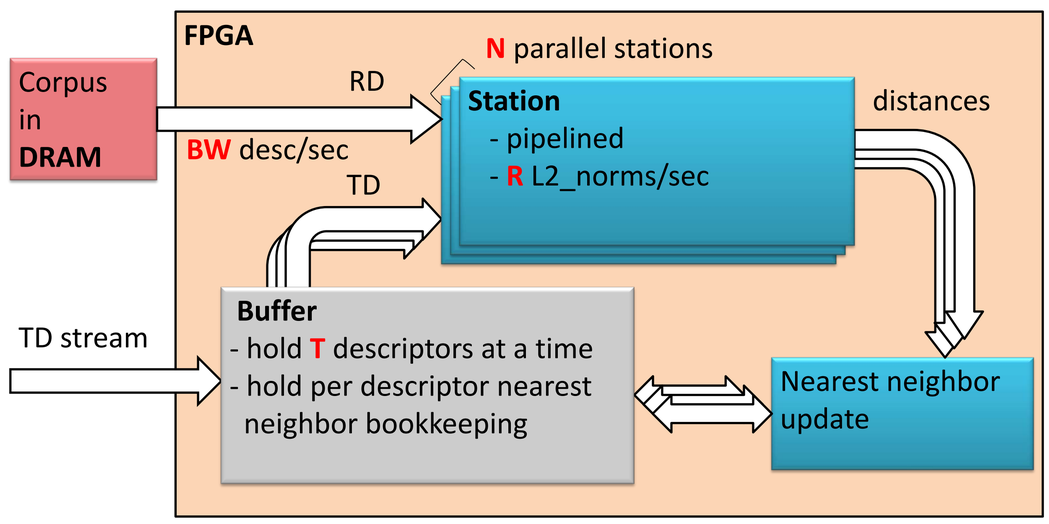
- Dynamic scene reconstruction from monocular view is a young research area with fundamental hard problems
- Performance-wise, post-processed offline—at least 3 orders of magnitude from realtime using 8x3-GHz Nehalem cores
- Develop for Intel Stellarton using FPGA acceleration
  - Current: 24-hours offline processing for 1 minute of video
  - To demonstrate: Real-time processing from video captured from a mobile camera

⇒Realworld scenes at realtime require combined, disruptive improvements from both theory/algorithm and platform



SIFT MATCHING

- *The* overwhelming performance bottleneck in reconstruction
  - A SIFT (Scale Invariant Feature Transform) descriptor is a 128-dimension vector
  - HD video generates ~30,000 *target descriptor* (TD) per sec.
  - Find for each TD its two nearest neighbors (by L2-norm) in a corpus of 10,000 to 1,000,000 (depending on complexity of scene) *reference descriptors* (RD)
  - Realtime requires 0.3-30 billion L2-norm calculations per second against a very large memory-footprint corpus
- **FPGA Accelerator Architecture:**



- **Throughput:**

Compute-bound: **R·N**/corpus_size TD per sec

Memory-bound : **T·BW**/corpus_size TD per sec