openptrack

Original Project Proposal

October 4, 2013 - jburke@ucla.edu

(Updated October 2014 for posting)

Goal

- **Spatially scalable, open source person tracking library for interactive art, education, and entertainment applications**.

- We believe that a disruptive project is now possible due to enabling technologies and related open source software libraries, and that artists and creators across many disciplines can benefit from this work.

- Openptrack is founded with the following objectives:
    - Track people persistently in free space. Start with centroids.
    - Allow for a variety of commodity and industrial hardware inputs.
    - Provide track data outputs to contemporary creative software frameworks.
    - Enable straightforward and rapid calibration in a variety of environments.
    - Remain focused on artistic and creative applications.

Technical Approach – Year 1

- Modular architecture based on PCL approach.

- Process each camera individually, and then fuse in the centroid (feature) space, rather than fusing point clouds.

- Target Hardware Platform: Ethernet-connected, IP-networked depth cameras such as the Mesa Swissranger 4500, Kinect (with ethernet bridge), and stereo visible-light cameras such as Blackfly GigE.

- Target Software Platform: Cross-platform (Linux first, Mac OS X, Windows) based on Point Cloud Library (PCL)

- Data output: IP network server, JSON format "stream".

- Typical tracking volumes: 8m x 8m x 3m to 20m x 20m x 8m

- Lighting: Support a variety of lighting environments, including bright sunlight and film/theater lighting.

- Desired refresh rate and latency: 30Hz+, <50ms latency

Future Needs / Opportunities

- Distributed computation (for now, centralized)

- Fuse data with other tracking systems (e.g., ultra wideband by Zebra)

- Evolve to support markerless skeletal tracking.

- Provide (constrained) object or marker recognition and tracking in the same system.

- Provide recognition of individuals on repeated entries/exists from the tracking volume. (ie, template matching)

Participating organizations

UCLA Center for Research in Engineering, Media and Performance (REMAP)
Problem definition, fiscal sponsorship, project roadmap, testbed support
Initial user - application coding & testing

Electroland
Problem definition, requirements analysis, hardware guidance, community-building
Initial user - application coding & testing

Open Perception
3D Vision approach and implementation, open source project mgmt, community-building

Proposed Licensing

- Goal: Encourage both use in proprietary projects and re-contribution of library enhancement. Focus on support of art, entertainment and educational applications.

- Software License: *BSD 3-clause*
   http://opensource.org/licenses/BSD-3-Clause

- Documentation License: *Creative Commons BY-SA*

- Contributor policy: Mozilla-style, where copyright is retained by the contributor but contributions must be licensed with MPL 2.
   http://www.mozilla.org/hacking/committer/committers-agreement.pdf

Team

Technology platform steering committee

- Jeff Burke, Asst. Dean, Technology & Innovation, UCLA Theater, Film and Television
- Alex Horn, Lead Software Developer, UCLA REMAP
- Eitan Mendelowitz, Asst. Professor of Computing and the Arts, Smith College
- Radu B. Rasu, CEO, Open Perception
- Damon Seeley, Principal and Co-Founder, Electroland

Advisory committee

- Noel Enyedy, Professor Education and Director of Research, UCLA Lab School
- Fabian Wagmister, Professor and Vice-Chair, Department of Film, Television and Digital Media

Proposed Roadmap

openptrack 0 – **Proof of concept on existing codebase** (Oct '13?)

openptrack 1 – **Funded multi-camera proof-of-concept** (Oct-Dec'13)

openptrack 2 – **Robust multi-camera capability** (Jan'13 – Mar'14)

openptrack 3 – **Improved interface and analysis** (Mar'14 – Jul '14)

openptrack 4 – **Distributed algorithm fork** (Mar'14 – Jul'14)

openptrack 4 – **Body segmentation fork** (Mar'14 – Jul'14)

openptrack 0

Proof of concept on existing codebase

- Proof of concept: Single Mesa Swissranger 4500 based person tracking @ ~15 Hz, < 100 ms for small tracking volume
- Existing codebase, contributions from Electroland, NIST sprint, PCL
- Questions
  - Eitan/Damon – status of code?
  - Radu – status of NIST sprint?

openptrack 1

Funded multi-camera proof-of-concept

Prove basic functionality of core tracking algorithms and encourage artists to jump in and begin using the project. Incorporate into NSF STEP project demonstration. A focus on commodity hardware and "zero to Hello World" experience for creatives. (No body segmentation until ~v4)

- Proof of concept: Multi-camera RGBD tracking of people (centroids) in an 5m x 5m space @ min 30 Hz, < 100 ms latency. Centralized processing.
- Consider multiple imager/ranger types from the outset using PCL IO-style interface layer. High-end: Swissranger, low-end: Kinect, mid-end: custom stereo camera.
- Track data served as an abstract list of tracks in JSON format.
- Open source project branding and public launch
- Sample projects for Processing, OpenFrameWorks, Max/MSP and .js frameworks. (Probably started by REMAP/Electroland/others to help bootstrap.)
- Testbed setups by UCLA, Electroland, OP.

openptrack 2

Robust multi-camera capability

Enable use in larger applications and mission critical contexts for creative projects. 3rd party creative partners will be courted to encourage library use outside of academia.

- Add multi-camera input and registration for tracking in much larger spaces. Focus on calibration ease and performance.
- Roll into real-world classroom tests with NSF STEP project at UCLA.
- Group tracking algorithms / data output extension at application layer.
- Multi-camera tracking in the same space with overlapping camera positions.
- Mix camera technologies in one tracking environment.
- Record and playback tracking data for off-site development.
- Enhance tracking accuracy.
- Push to kickstarter for subsequent phases

openptrack 3

Improved interface and analysis

Enhance existing functionality and adds polish to the library and associated utilities.

- Consider running a tracking lab / class at UCLA.
- Enhance tracking accuracy.
- GUI tools for setup and registration.
- Tracking to include non-overlapping camera positions.
- Track people in non-planar space.
- Autocalibration by analysis of raw point clouds over time.
- Analysis and provision of track properties (eg: "active," "sitting," "stationary," etc)

openptrack 4

Distributed algorithm fork

- Distributed algorithm, decentralized platform

- Funding from broader consortia

Body segmentation fork

- Body segmentation, export of pose data

- Funding from broader consortia