Kitchen Occupation

People counting using depth sensors.

Department of Electrical Engineering, Linköping University

Objectives

1. Create a system to monitor room usage intensity, primarily focusing on student kitchens.
2. The system needs to be cheap and easy to install and maintain.
3. The system should provide real-time information about room usage intensity.

System

- Hardware setup (figure 1)
  - One or several Microsoft Kinect cameras and a computational device with Internet connection is required, as well as access to the power grid.
- Portability
  - Cross platform in the sense that we support most UNIX-like systems and Windows.

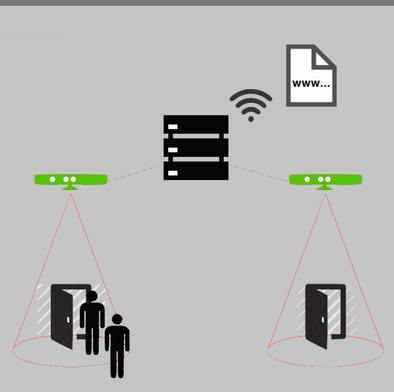

Figure 1: Rough system overview with main components.

Image Processing

- Human segmentation
  - Human segmentation is based on the assumption that human heads are distinguishable modes in the depth image and that people moving very close to each other seldom differ radically in height (figure 3). It is realized by a series of threshold and morphological operations, Gaussian blurring, contour drawing and searches for local maxima's (figure 2).

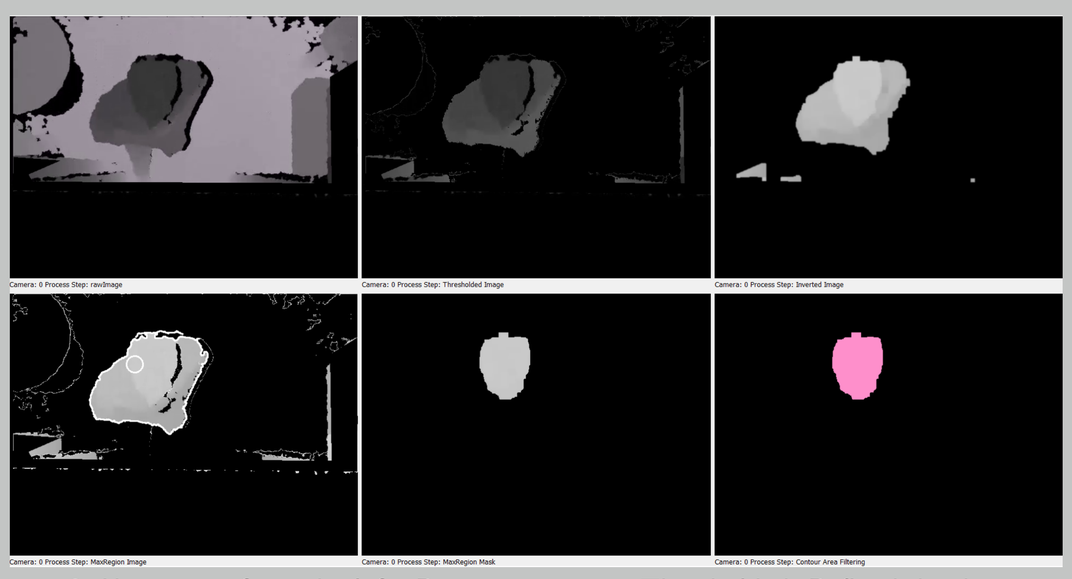

Figure 2: Upper row from the left: Raw input image, Thresholded, Refined depth image. Bottom row from the left: Local Maxima detection, Head thresholding, Segmented heads.

Figure 3: From left: Raw imput image, Occlusion after thresholding, Successful segmentation

- Tracking and counting (figure 4)
  - The tracker pairs objects with each other from previous frame to the next. Pairs closest matching objects. Handles occlusion, outliers and noise.
  - Counting is done using user specified checkpoint lines and a door area.
- Queue detection (figure 5)
  - Objects connected by short Bézier curves, which has been fit to the objects' positions and velocities, are considered in a queue.
  - Queue severity is determined by queue detection frequency.

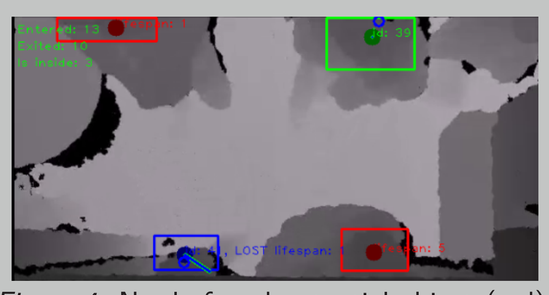

Figure 4: Newly found potential objects(red), lost objects(blue) and objects(green)

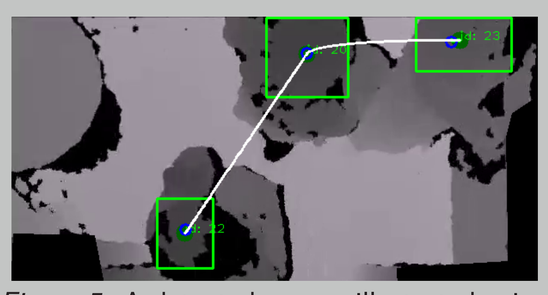

Figure 5: A detected queue, illustrated using the spline connecting the persons.

Software

- Built in C++ in a very modular, extendable and configurable fashion.
- Modular
  - Algorithms have a very clear general interface to the rest of the pipeline.
  - Multiple algorithms of the same type were developed in parallel, with minimal effort on interface compliance.
  - An entire new computer vision approach could be switched to at the end of the project with minimal effort.
- Configurable
  - The entire pipeline can be reconfigured from file. This includes addition, removal and rearrangement of algorithms, and parameter values.
  - It is possible to automate tests of hundreds of configurations, changing both algorithms and algorithm parameters without re-compiling.

Configuration & Debugging GUI

- Transparent and flexible pipeline overview (figure 7)
- Automatic on-line low-level profiling of every module (figure 7)
- Configuration interface (figure 6)
  - Doors (green)
  - Entering checkpoints (circles)
  - Excluded areas (red)

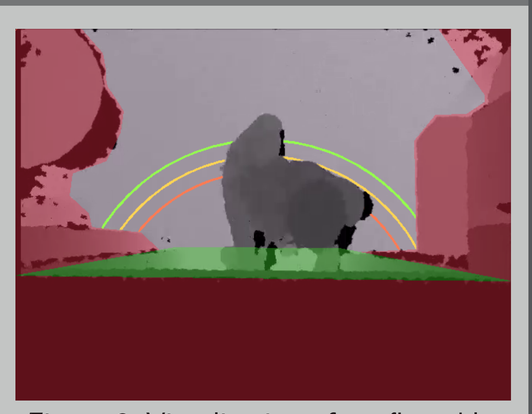

Figure 6: Visualization of configurables

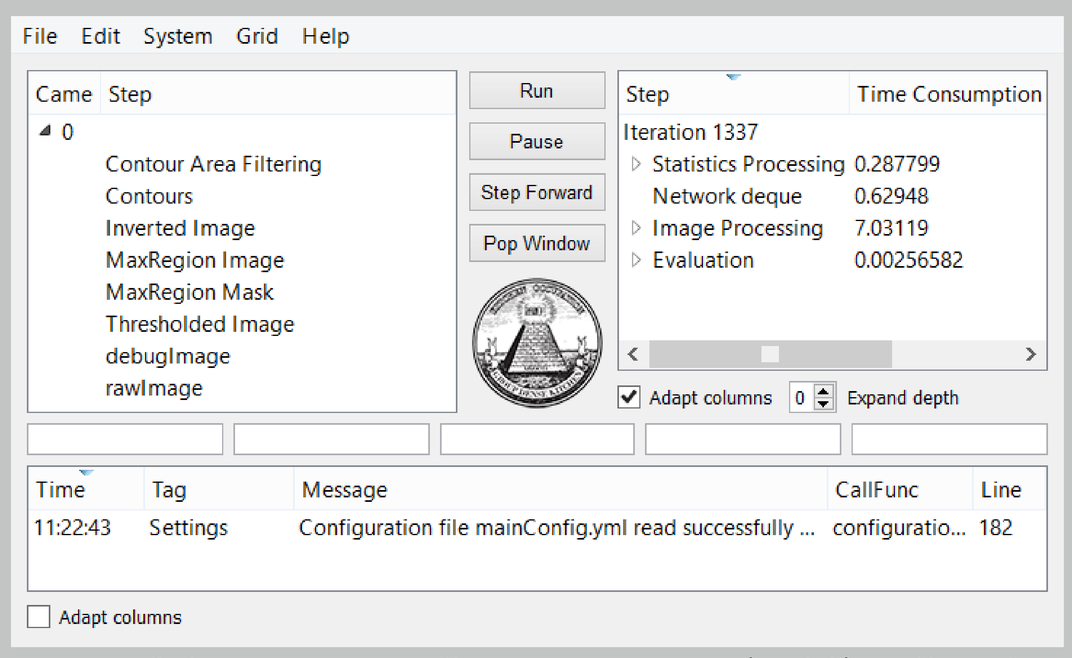

Figure 7: Main GUI window with the different processing steps, (top left), profiling information (top right) and log messages (bottom)

Results: Table

- Final system performance and equations used during evaluation.

$$A_{in} = 1 - \left|\frac{\sum_{frames} in_{Est} - \sum_{frames} in_{GT}}{\sum_{frames} in_{GT}}\right| \quad (1)$$

$$A_{out} = 1 - \left|\frac{\sum_{frames} out_{Est} - \sum_{frames} out_{GT}}{\sum_{frames} out_{GT}}\right| \quad (2)$$

Sequence Name	Total entered (GT)	A_{in}	Total exited (GT)	A_{out}
Data seq. 1	108 (108) people	99 %	101 (104) people	97 %
Data seq. 2	122 (141) people	87 %	77 (91) people	85 %

System performance in the two evaluation sequences

- Data seq. 1 & Data seq. 2 are two data sequences of 30 minutes each.

Conclusion

- The system provides high-precision real time people counting using the Microsoft Kinect sensor at low computational cost on a 2Ghz low-end CPU.
- The software architecture enables fast implementing and testing of different algorithms. It gives a very lightweight and high-performing vision pipeline.

Erik Fall Gustav Häger Malin Rudin Alexander Sjöholm Martin Svensson Mattias Tiger Nikolaus West