Poster Abstract: Intuitive Appliance Identification using Image Matching in Smart Buildings

Kaifei Chen*, John Kolb*, Jonathan Fürst†, Dezhi Hong‡, Randy H. Katz*
*University of California at Berkeley †IT University of Copenhagen ‡University of Virginia
{kaifei, jkolb}@berkeley.edu, jonf@itu.dk, hong@virginia.edu, randykatz@berkeley.edu

ABSTRACT

Identifying an appliance for interaction in commercial buildings becomes non-trivial as the number of smart appliances explodes. We present a system for users to intuitively "look up" appliances using image matching-based technique on a pre-constructed and annotated visual model of building interiors. It matched 98% images on a public robot-collected dataset and achieved 100% recall and precision. Our lab experiments with human captured videos and images also show the feasibility of real world deployments.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation (I.7)**]: User Interfaces (D.2.2, H.1.2, I.3.6)— *Interaction styles (e.g., commands, menus, forms, direct manipulation)*; I.4.9 [**Image Processing And Computer Vision**]: Applications

Keywords

Identification; Structure from Motion; Image Matching

1. INTRODUCTION

There are many smart home appliances emerging today, such as programmable thermostats, light bulbs, and fridges. The intent is to make everyday objects software-controllable and connected to the Internet, forming the Internet of Things. However, as the number of smart appliances grows, identifying which appliances to interact with through software becomes non-trivial and time-consuming for users.

Previous efforts are not intuitive for two major reasons. First, some approaches require users to describe the appliance in cumbersome ways. For example, users need to query appliances using a statement with strict syntax in pre-populated appliance directories [1]. Second, many approaches require the deployment of additional infrastructure, such as infrared transceivers [3]. Given these drawbacks, we employ a vision-based approach, which is intuitive for users and introduces minimum deployment overhead.

BuildSys'15, November 4–5, 2015, Seoul, South Korea.
ACM 978-1-4503-3981-0/15/11.
DOI: http://dx.doi.org/10.1145/2821650.2830299 .

Figure 1: Modeling Phase Overview

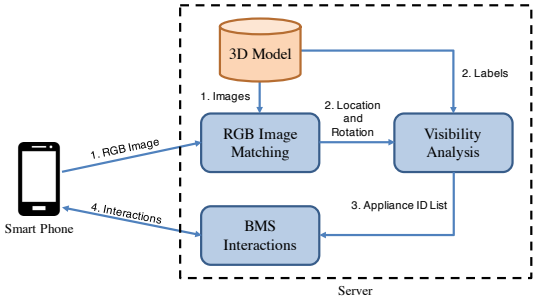
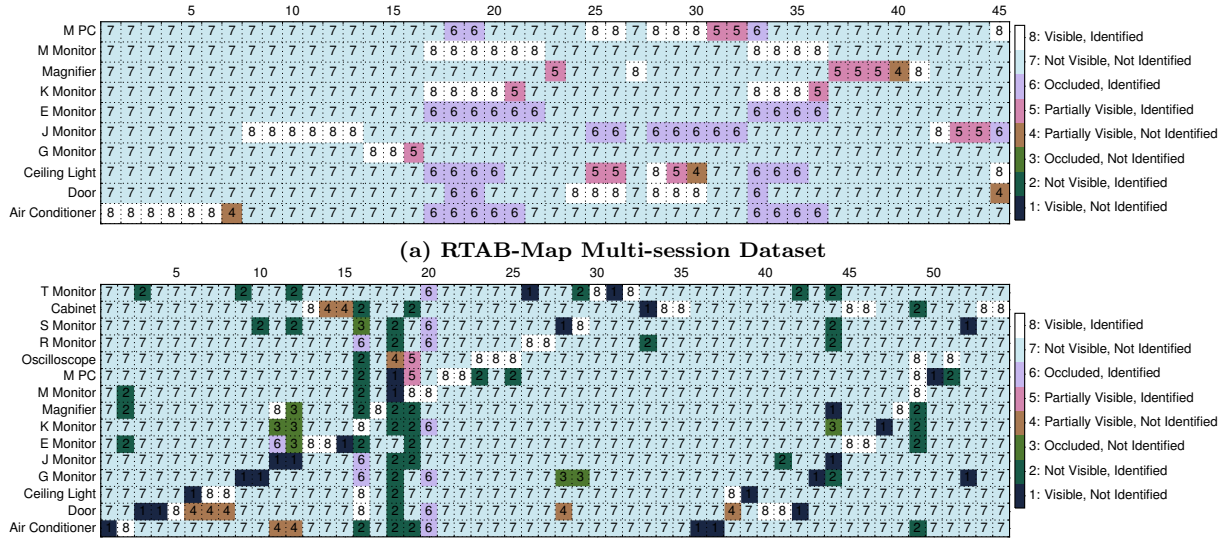


Figure 2: Matching Phase Overview

This work explores an intuitive way of identifying smart appliances: *what you see is what you interact with.* We argue that with computer vision-based techniques, we should be able to identify the objects more easily and interact with them in a straightforward manner. To achieve this, we develop a system comprised of two steps: a modeling phase and a matching phase. In the modeling phase, the building manager needs to reconstruct a visual 3D building model and mark the appliances inside it. In the matching phase, users can identify appliances in the 3D model and use a smart phone to interact with them.

2. SYSTEM OVERVIEW

Modeling: Figure 1 shows the modeling phase. Structure from Motion (SfM) reconstructs a point cloud from an RGB-D video captured by the building manager. Given an RGB-D video, SfM assumes every two consecutive frames have a small difference in location and orientation, meaning they have enough overlap to be registered. SfM first computes all Speeded Up Robust Features (SURF) on both frames. For each feature in an image, SfM searches for a common feature in the other, therefore generating a list of matched points. With the matched points, SfM computes their corresponding 3D coordinates in their own image coordinate system using the depth values. SfM then uses the RAndom SAmple Consensus (RANSAC) algorithm to calculate the relative transformation between the two frames. After all consecutive frames are registered, their 3D points are



(a) RTAB-Map Multi-session Dataset

(b) Lab Deployment

Figure 3: Image matching results. Every column is a matched image, and every row is an appliance.

transformed and added to a global point cloud. In addition, we build a labeling tool that allows the building manager to label a 3D point by clicking on a registered 2D image.

Matching: Figure 2 shows the matching phase. The image matching component registers an RGB image onto the point cloud in the 3D model. We cannot simply reuse SfM here for two reasons. First, an image has to be compared with another registered image that shares enough common features. However, the new image is taken from an arbitrary location and orientation, so we do not know which registered image it should be computed with. Second, the RGB image does not have depth value, so we cannot use RANSAC to compute the relative transformation. We therefore adopted another approach. We first calculate features for the new image and find the registered image that has the most common features. Then image matching becomes a Perspective-N-Point (PnP) problem, which computes the image location and orientation given the 3D coordinate of a point and the 2D coordinate of its corresponding point in the image.

Given the location and orientation, we need to find visible appliances. We project all the labeled points in front of the camera onto the camera plane, and pick the 2D pixels on the image plane that are within range of the image.

3. EVALUATION

We implemented our system by extending RTAB-Map [2] and evaluated it on two datasets. One is the RTAB-Map multi-session dataset [2], which contains five videos of a building floor collected by a robot with a Microsoft Kinect. We used one video with 304 images to model and labeled 10 objects, shown in Figure 3a. We manually chose 46 RGB images that contain labeled objects from the other videos. The second dataset was collected by a human using a Microsoft Kinect in our lab, containing 1075 images and 15 labeled objects, shown in Figure 3b. We then took five or six pictures of each object after four days from different angles.

Every object in every test image is referred to as an instance, and marked based on human observation: *Visible*, *Partially Visible*, *Not Visible* (not in the viewing cone), and *Occluded* (in the viewing cone but occluded). Our system reports whether each object is identified in every matched image. Therefore, instances fall into eight categories given ground truth and identification result, as shown in Figure 3.

Figure 3a shows the results on the RTAB-Map multi-session dataset. Among the 450 instances, 392 are type 7 and 8, meaning our system reports correctly, and 40 are type 6, meaning the objects are occluded but in the viewing cone. Because we do not perform occlusion analysis, these results are expected. The other 18 instances are type 4 and 5, which are partially visible and not always identified. Because we only label several points on an object, our system does not identify the object if the labeled points are not in the viewing cone. This will not impact user experience because users intuitively capture the entire target object. We consider categories 4-8 to be successful. Therefore, we achieved 100% success on this dataset. The lab deployment results are in Figure 3b. 730 out of all 810 instances are type 4-8. Considering the noise caused by human and environmental changes, we argue that this result is promising for real deployment in spite of more erroneous instances.

Acknowledgements

This work is supported in part by the National Science Foundation under grant CPS-1239552 (SDB).

4. REFERENCES

[1] S. Dawson-Haggerty, X. Jiang, G. Tolle, J. Ortiz, and D. Culler. sMAP: a simple measurement and actuation profile for physical information. In *SenSys*. ACM, 2010.

[2] M. Labbe and F. Michaud. Online global loop closure detection for large-scale multi-session graph-based slam. In *IROS*. IEEE, 2014.

[3] B. Zhang, Y.-H. Chen, C. Tuna, A. Dave, Y. Li, E. Lee, and B. Hartmann. Hobs: head orientation-based selection in physical spaces. In *SUI*. ACM, 2014.