Data Center Asset Tracking Using a Mobile Robot

John Nelson* Jonathan Connell† Canturk Isci† Jonathan Lenchner†

ABSTRACT

Management and monitoring of data centers is a growing field of interest, with much current research, and the emergence of a variety of commercial products aiming to improve performance, resource utilization and energy efficiency of the computing infrastructure. Despite the large body of work on optimizing data center operations, few studies actually focus on discovering and tracking the physical layout of assets in these centers. Such asset tracking is a prerequisite to faithfully performing administration and any form of optimization that relies on physical layout characteristics.

In this work, we describe an approach to completely automated asset tracking in data centers, employing a vision-based mobile robot in conjunction with an ability to manipulate the indicator LEDs in blade centers and storage arrays. Unlike previous large-scale asset-tracking methods, our approach does not require the tagging of assets (e.g., with RFID tags or barcodes), thus saving considerable expense and human labor. The approach is validated through a series of experiments in a production industrial data center.

Categories and Subject Descriptors

I.2.9 [Robotics]: Commercial robots and applications

Keywords

Data center; asset tracking; robot

1. APPROACH

Our approach to autonomic data center asset tracking leverages a vision-based mobile robot for determining the data center layout, and the racks positioned within this layout. Previous studies have shown that a simple, low-cost robotic solution can be used to provide at least a coarse mapping of data centers, which, further, can be coupled with thermal monitoring and energy management capabilities [2, 3]. While such a robot can identify physical obstacles, and hence candidate rack locations within a discovered data center, the location of individual servers or storage modules cannot be readily discovered with such an approach. To bridge this gap in the robot's understanding of the data center layout, we leverage asset identification functionality that is built into a range of enterprise IT systems. In particular, our prototype implementation utilizes BladeCenters and BladeCenter Management Modules (BCMMs), which provide a rich and easy to use set of location capabilities via locator LEDs [1]. Running a mobile robot in coordination with the BCMM software, we can configure each BCMM to emit a special location LED signature, or a series of LED states over time, which can then be processed by the vision processing subsystem of the mobile robot. The management module can in-effect spell out precisely what assets it contains using a binary LED pattern. The information thus obtained can then be used to enter the location into an asset management system and to keep such asset data up-to-date in an autonomous fashion, without necessitating any end user involvement.

The main advantages of our proposed, and prototyped approach are, first, we provide a very cost-effective way of tracking assets that requires little administration; second, it is much more accurate and reliable than manual asset tracking; and third, unlike some of the solutions that rely on additional tags to be placed on the IT assets, our solution operates without any such prerequisites on the asset side, aside from the asset's ability to emit locator lights.

2. EXPERIMENTS

Encoding unique identification numbers such as the serial numbers of IT equipment using existing locator lights can be approached from several directions. Of the available methods, the most time-wise efficient, was to encode each blade center using a unique binary number known to the message sender, i.e., the application resident on the blade center management modules, where a given blade server LED represents a single bit. A single blade center can house at most 14 blade servers, and thus a blade center fully populated with operable blade servers, can represent an arbitrary 14 bit number. In some cases blade centers may be populated with double-wide blade servers, reducing the number of operable bits per message to as little as 7, and in either the single-wide or double-wide cases, it may be that not all blade slots are populated, even further reducing the expressive content of potential messages.

Most racks contain perforated doors to accommodate high volume air flow to and through the equipment housed in the racks. These perforations, conveniently, also allow for the transmission of light in a way that can be observed by the robot (or for that matter, by a human being). Using the BCMM software resident on these blade centers, the indicator LEDs of the individual blade servers can be set to an on or off position.

Through a series of experiments on three blade centers, we investigated the ability of the robot to identify the intended encodings associated with these LEDs, and thereby identify the blade centers and resident blade servers at a given location in

*IBM CIO Office, Southbury, CT, USA. jcnelson@us.ibm.com.
†IBM T.J. Watson Research Center, Yorktown Heights, NY, USA. {jconnell, canturk, lenchner}@us.ibm.com.

SIGMETRICS'13, June 17–21, 2013, Pittsburgh, PA, USA.
ACM 978-1-4503-1900-3/13/06.

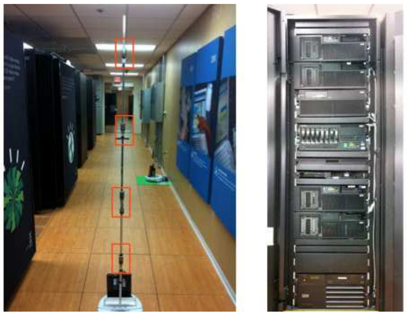

Figure 1: Our robot equipped with four forward-pointing cameras in addition to its downward-pointing camera, used for navigation, on the left, and the results of stitching images from the four cameras, on the right. The view is of a rack with rack door removed.

the data center. For all experiments the robot was equipped with a forward-facing Logitech C905 camera which we rotate by 90° from its standard orientation to maximize the vertical field of view. In addition, in the various experiments, atop the Logitech C905 lens, we added one of two inexpensive snap-on wide angle attachment lenses to expand the robot's field of view even further. Experiments were conducted only on the bottom three blade center chassis locations. To discover blade centers and blade servers at additional rack positions one would have to mount additional forward-facing cameras. The forward-facing camera of these experiments is in addition to the existing downward-facing camera the robot uses for its basic tile-by-tile navigation. Details of the robot's basic vision-guided navigation can be found in [3, 2].

2.1 Detecting and Counting Lights

The first experiment designed was to test the viability of encoding static ID numbers using each of the available bits. The experiments were universally successful but pointed to the fact that just two chasses were sufficiently clear in the image with a wide-angle or fisheye lens to faithfully discover all available lights. Thus the final robot would have to be equipped with several cameras. Figure 1 shows such a robot.

2.2 Framing and Reading Signatures

Our approach to framing and reading chassis signatures was to first set all of the lights on, so that the robot could determine the bits available to it. After processing the locations of the available bits, the actual message could then be sent. Figure 2 shows an initial binary image with all the LEDs of a blade center turned on. This image, in conjunction with the analogous image with all LEDs turned off, allows the robot to ascertain which LEDs are available for encoding.

In some cases a message may take multiple "words," but since the robot can process images in milliseconds, even the blade centers with few available bits can process their entire signature with negligible overhead. Note that both the sender (the asset management software in conjunction with the BCMMs) and the receiver of the message (the robot) know the number of available bits by virtue of the initial framing exercise. Only BCMMs with available bits matching those found by the robot will broadcast, and further, under the assumption that the serial numbers are of uniform length, all messages will be of the same length and, hence, the same number of "words" and so the sequential transmission of these words can by readily time synchronized.

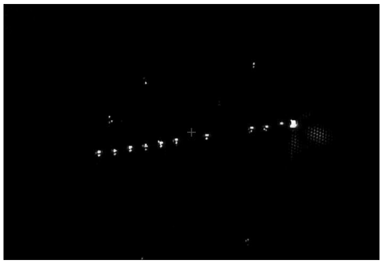

Figure 2: A two-bit (black and white) image of one of the blade centers used in our experiments with all available LEDs turned on – one of two views used by the robot to determine the number of bits available for encoding. Blades 1 – 6, 8, and 11 – 14 are available.

The second experiment involved first allowing the robot to identify the set of available bits by alternately setting all available indicator LEDs to the "off" position, and then to the "on" position, before performing the actual encoding. 15 trials were performed with each blade center utilizing distinct encodings. Again the encodings were hand generated with an attempt to balance the utilization of each bit, and all tests were successful.

2.3 Acceptable Robot Positioning

Up until this point in our experimental protocol, the robot had been manually placed and oriented towards the front of each rack, producing very promising results. The question we therefore turned to was examining the degree of accuracy the robot required in terms of centering in front of the rack in order for visual identification of the LED states to succeed. The next experiment established the region of confidence for accurately reading a blade center's encoding, a region we ultimately referred to as the "critical teardrop" because of its roughly teardrop shape.

2.4 Ability to Achieve a Position

After identifying the "critical teardrop," i.e. getting a precise picture of where the robot must be in order to accurately read the binary encodings, a final experiment was designed to discover how accurately the robot could reach a requisite coordinate and orientation, again with successful results achieved.

.

3. CONCLUSION

We present a simple, automated and low-cost approach for asset tracking in data centers using blade center indicator LEDs as asset signatures. Our approach uses a vision-guided robot and leverages existing data center management capabilities to eliminate the need for prior processing, tagging of assets, additional hardware, or manual inspection. Our experimental evaluations underline the feasibility and accuracy of our method.

4. REFERENCES

[1] IBM Corporation. IBM BladeCenter advanced Management Module: Command-Line Interface Reference Guide. `http://publib.boulder.ibm.com/infocenter/bladectr/documentation/topic/com.ibm.bladecenter.advmgtmod.doc/kp1aupdf.pdf`, 2009. [Online; accessed 31-October-2012].

[2] J. Lenchner, C. Isci, J. Kephart, C. Mansley, J. Connell, and S. McIntosh. Towards data center self-diagnosis using a mobile robot. *Proceedings of the 8th International Conference on Autonomic Computing (ICAC)*, pages 81–90, 2011.

[3] C. Mansley, J. Connell, C. Isci, J. Lenchner, J. O. Kephart, S. McIntosh, and M. Schappert. Robotic mapping and monitoring of data centers. *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.