Automatic Computer Detection and Power Estimation in Indoor Environments from Imagery

Satarupa Mukherjee, Hariprasad P.S., Omar Oreifej, Brian Pugh, Eric Turner, Avideh Zakhor

Department of Electrical Engineering & Computer Sciences,

University of California, Berkeley USA

{Satarupa,elturner,brian.a.pugh,avz}@berkeley.edu, {harips1992,oreifej}@gmail.com

***Abstract* – In this paper, we describe an automated technique for estimating power consumption by computers in buildings by processing visual imagery collected during a walkthrough of the building. This is an important problem since desktop and laptops are the largest contributors to electricity consumption in most large commercial buildings. The images used for estimation are obtained by a backpack equipped with a suite of sensors such as laser scanners, cameras, and an IMU, carried by a human operator walking at normal speed inside buildings. In addition, the operator carries a handheld infrared (IR) camera to take pictures of the CPU box of the desktops. We take a two step approach to this problem. First, we develop a technique based on convolutional neural networks to detect and count computers, which results in 90% accuracy as tested on a three story building with over 100 machines. Second, we develop an SVM based power estimation algorithm for computers using a handheld IR camera which captures both IR and visible light imagery simultaneously. The average power estimation error over 101 computers is around 8%. Combining these two algorithms, it is possible to accurately power consumption due to computers in commercial buildings.**

I. INTRODUCTION

A major source of power consumption in commercial office buildings is desktops, laptops and computers. As such, automatic detection of computers in open offices can be helpful in estimating a major portion of electricity consumption in multi-storied commercial buildings. Such estimates are not only useful by themselves, but also can be used in conjunction with energy simulation engines such as energy plus [14,20] to model energy consumption inside buildings.

Analysis of energy in multi-storied buildings involves three stages of building inspection: basic evaluation, diagnostic measurement, and advanced analysis. At each stage, an expert needs to walk through, inspect, and analyze energy consumption of different devices. The process is time consuming, laborious and error prone. In this paper, we utilize the advantages of sensing technology and computational algorithms to generate 3D models of building interiors which can then be used for energy analysis of buildings. Specifically, we use a novel system based on state-of-art backpack with a suite of sensors as shown in Fig. 1 in order to create 3D thermal models of buildings [17]. A human operator wears the backpack and walks through the building at a normal speed e.g. between 0.5 to 1 meter per second. Once the data is collected, it is processed offline and entirely automatically in order to generate the 3D path the operator traversed [1, 2,5,6,7,19], the optical 3D point cloud, the thermal/IR point cloud co-registered with the optical point cloud, 3D models with surface reconstruction of walls and floors, and photo realistic rendering of the internal structure of multi-storied buildings [8,9,10,11].

In this paper, we utilize the above data products generated from the backpack to estimate power consumption due to computers in buildings. To do so, we solve two distinct technical problems: first we use the imagery collected during the walkthrough to detect and count the number of computers in each room. Second, we develop an algorithm that uses infrared (IR) and visible light images to estimate power consumption for each computer. By combining the above two estimates, it is possible to estimate electricity consumption due to computers in commercial buildings. The outline of this paper is as follows: In Section II, we describe our approach to computer detection from visual images. In Section III, we describe our power estimation algorithms for computers. Section 4 includes experimental verification of our algorithms, and Section 5 includes conclusion.

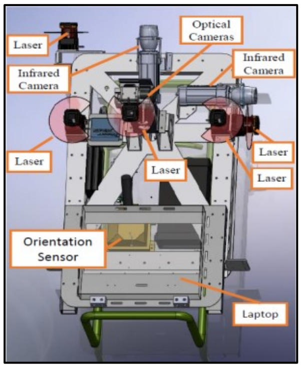

Fig. 1. CAD diagram of ambulatory Backpack

II. DETECTING AND COUTING COMPUTERS

We start by generating depth and normal maps for optical and IR images by using the 3D models generated from the backpack [8,15,19]. Fig. 2 shows an example of visible light image, IR image, and the corresponding depth and normal maps associated with the visible light image. The normal maps are used to remove the floors, walls and ceilings from the detection process thus limiting the search region to furnished portions of the rooms where computers are likely to be located. Next, we apply selective search algorithm [13] to generate candidate windows in the visible light image. Selective search is a useful tool for object detection as it possesses the attributes of both exhaustive search and segmentation. Similar to exhaustive search, selective search also tries to capture all possible object locations whereas like segmentation, it uses the structure of the image to direct the segmentation process. Finally, selective search is capable of capturing objects at different scales. Each window generated by the selective search method, is passed through a regional-convolutional neural network (R-CNN) [4]. R-CNN solves two purposes simultaneously. On one hand, it localizes and segment objects by applying high capacity convolutional neural networks to bottom up region proposals. On the other hand, it boosts significant performance by domain specific fine tuning when labeled training data is scarce.

The output from the first fully connected layer fc6, which is a sparse vector of 4096 values [4], is passed through a support vector machine (SVM) [3] model trained beforehand. To train the SVM model, 1070 images of desktops were downloaded from ImageNet [21] to serve as positive examples, and 8116 images of cabinets, door frames, window frames to serve as negative examples. Each of these images is mean subtracted and passed through Alex-net [16] after being resized to 227 × 227. Finally, the output of the first fully connected layer fc6, is used to train the linear SVM model. We than apply non-maximum suppression to the output of the SVM for all candidate windows in order to eliminate redundant windows detected as object. Finally, the depth map for the visible light image is used to project the detected machine in the visible light domain onto the IR domain. Thus we obtain the final detected objects both in optical and IR domain.

The output of the above algorithm is one candidate window per input visible light image. In order to count the total number of computers present in an area, we need to remove redundantly detected computers across multiple consecutive images. To do so, we first project the window corresponding to detected computer i in one frame onto the next frame with detected computer j by exploiting the knowledge of relative pose of successive images during the backpack localization step [2,19]. The goal is to determine whether these windows both correspond to the same computer or different ones. To do so, we apply the procedure described in [12]. Let b_i and b_j denote the set of all pixels in the rectangular window associated with detected computers i and j respectively. Define two scoring functions:

$$S(i,j) = \frac{|b_i \cap b_j|}{|b_i|}, \qquad S(j,i) = \frac{|b_i \cap b_j|}{|b_j|}$$

where the absolute sign is used to count the number of elements in a set. Fig. 3 shows visual representation of the scoring function for computers i and j denoted with boxes of size A and B respectively. Intuitively S(i,j) represents the ratio between the size of the overlap region between i and j, and the size of i.

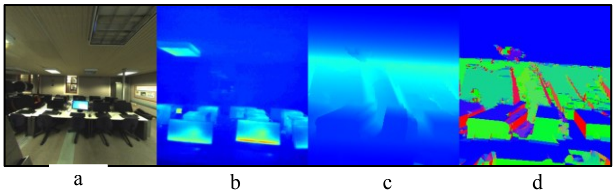

Fig. 2. (a) Original visible light image and corresponding (b) IR image (c) depth map; (d) normal map

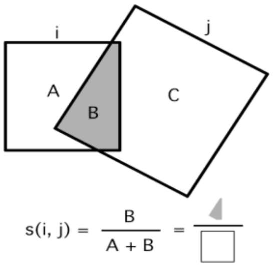

Fig. 3. Visual representation of scoring function

Once the above scoring functions are calculated for detected computers i and j, we apply the following logic to discard redundant computers across consecutive frames:

(a) If S(i,j) and S(j,i) are both smaller than a prespecified threshold, both detected computers i and j are kept.

(b) If S(i,j) is greater than the threshold and S(j,i) is less than the threshold, detected computer i is kept and detected computer j is discarded, and vice versa.

(c) If both the scoring functions are greater than the threshold, the detected computer with a greater area is kept and the smaller one is discarded.

III. POWER ESTIMATION

The power estimation algorithm for each computer uses the visible light and IR images of the computer captured with a handheld IR device taken close to the computer. In doing so, most of the pixels in the IR picture correspond to a computer rather than the background, resulting in more accurate power estimate than if we were to use the pictures from the IR cameras on the backpack. An example of an IR picture from the handheld device is shown in Fig. 4(b).

Our proposed power estimation algorithm is a two-step process, where in the first step we train a SVM to decide whether a machine is 'on' or 'off'. If the machine is detected to be 'on' in the first step, we pass it through a support vector regressor (SVR) to estimate the power consumed by it. We train the SVR using the following features: (a) Histogram of Oriented Gradients (HOG) for the detected machine on the IR image. (b) HOG for the detected machine on the optical image. (c) Ray features from the IR image designed to capture the temperature distribution around the hottest spot in the image. This is done by casting rays in random directions from the hottest point. Consequently, the feature is represented as a histogram of temperature difference between the hottest point and the points at the rays ends. (d) Temperature statistics, i.e. minimum temperature, maximum temperature, average temperature.

The features are computed at multiple scales around the hottest point in the image in order to achieve pose invariance. We refer to this approach as heat pyramid since it is similar to the spatial pyramid, except that the levels of the pyramid are constructed based on temperature. The top level of the pyramid is the smallest scale, where the features are computed within a small region around the hottest point. At the next pyramid level, the region around the hottest point is expanded, and the resulting features are concatenated with the features from the upper pyramid level. In our implementation, we use a two-level pyramid.

With the above mentioned features, a SVR is trained and in the testing phase, power consumption is estimated on the detected machines using a 'leave one desktop out' cross validation technique.

IV. RESULTS

To characterize the performance of our power estimation algorithm, we use a Kill-A-Watt device in series with a number of desktops in order to measure their actual power consumption. Figure 4(a) shows the plot of estimated versus actual power consumption over 101 computers.

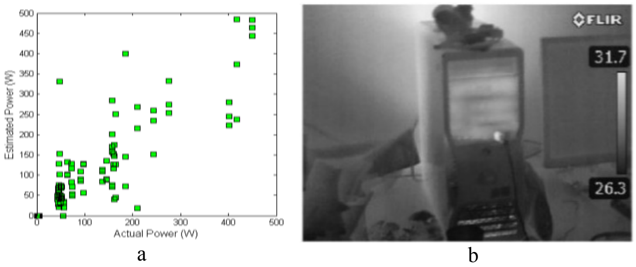

Fig. 4. (a) Estimated vs actual power consumption. (b) IR picture from the handheld IR device.

As seen, there is strong correlation between the estimated and actual power usage. The distribution of absolute estimation error over 101 desktops ranging in power from 50 to 500 watts is shown in Fig. 5. The average error in total power consumption over all 101 computers is only 8%. This is because the overestimation and underestimation errors over 101 computers cancel each other out, thus resulting in a low average error. Next we compute the average error over 10 computers rather than 101 to emulate performance in smaller buildings with fewer computers. To do so, we perform 20,000 trials whereby in each trial we randomly draw 10 out of 101 computers to compute the average percentage error for that trial. Averaging the absolute values of those over 20,000 trials, results in 21% error between actual and estimated power over 10 machines. This is to be expected since the larger the sample set over which we compute average error, the lower the percentage error. Finally, if we repeat the 20,000 trials over the 51 machines with actual power consumption between 50 and 200 watts, we find the average error over 10 machines to be around 11%. This improved estimation for lower power machines can be explained by the fact that temperature measurement through an IR camera is a better proxy for lower power machines than higher power computers. This is to be expected since in practice a myriad of methods are used for heat dissipation and temperature control of high end machines.

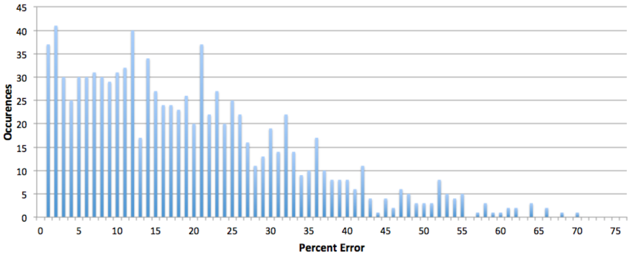

Fig. 5. Distribution of absolute error over 101 desktops.

Next we apply our computer detection algorithm to the visible light images from a laboratory with 12 machines. Our proposed method estimates the number of computers at 11. The estimated power consumption for these 12 computers, together with the ground truth obtained with a power-meter put in series with the machines is shown in Table 1. The actual and estimated total powers are 542 and 597 watts respectively, resulting in relative error of about 10%.

In practice, taking close up IR pictures of all computers during a walkthrough could substantially increase the acquisition time using the backpack. One way to alleviate this, would be to take close up IR pictures of a small sample of computers in an office building and use the average power for those as a proxy for power consumption for all other detected computers.

Computer Id	Actual Monitor Power	Actual CPU power	Estimated Power	Absolute Error	% Error
105-22	21.1	31.0	57.26	5.16	9.90
105-21	22.4	31.2	63.14	9.54	17.80
105-20	23.2	33.4	62.22	5.62	9.93
105-19	23.6	32.2	58.26	2.46	4.41
105-18	23.3	32.3	56.70	1.10	1.98
105-16	21.8	31.1	66.74	13.84	26.16
105-15	20.0	31.5	58.55	7.05	13.69
105-14	22.3	31.6	62.22	8.32	15.44
105-13	21.8	33.4	62.03	6.83	12.37
105-12	21.7	32.7	50.31	4.09	7.52

Table 1. Quantitative results for computer power estimation

Figure 6 shows two examples of results of the computer detection algorithm on the laboratory dataset. The left picture in each pair, corresponds to visible light and the right one to IR. The detected computers are represented by green boxes on the optical images which are then projected on the IR images. The advantage of detecting computers in the visible light domain is that even when a machine is off and hence not prominent in the IR domain, our proposed method based on visible light imagery can be successful in detecting in.

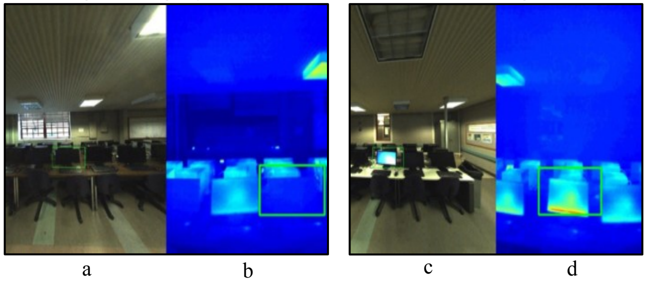

Fig. 6. Visual results on detected computers in laboratory dataset (a) & (c) visible light domain (b) &(d) IR domain.

In addition to the above laboratory dataset, we have validated our computer detection algorithm on Mulford Hall, which is a 3 story building on U.C. Berkeley campus. Table 2 shows the manual and algorithmic count of machines for each floor indicating that our algorithm is more than 90% accurate for this large dataset.

Floor	Manual Count of Computers	Algorithmic Count of Computers	Accuracy (%)
First	30	32	93.75
Second	57	53	92.98
Third	24	26	92.31

Table 2. Quantitative Result on Mulford Dataset

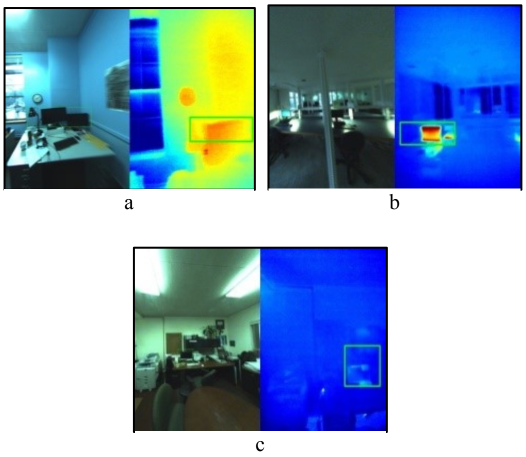

Fig. 7. Visual results on (a) first floor, (b) second floor and (c) third floor of Mulford Hall.

Figure 7 shows visual results of 3 instances of detected computers on the 3 floors. Again the left image in each paid corresponds to visible light image and the right one to IR.

Figure 8 shows few error instances of the computer detection algorithm. Figures 8(a) and 8(b) illustrate a missed detection instance with 2 computers on the two sides of the window. Our algorithm detects a large window corresponding to one computer, rather than two. Figures 8(c) and 8(d) on the other hand, show an instance of false detection where a microwave is detected as a computer as it appears visually similar to a computer.

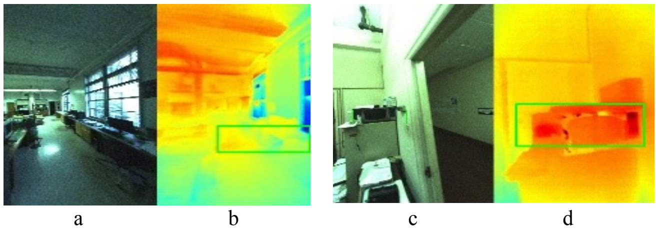

Fig. 8.Visual Representation of Error Instances(a) & (c) detections in visible light domain (c) & (d) detections in IR domain

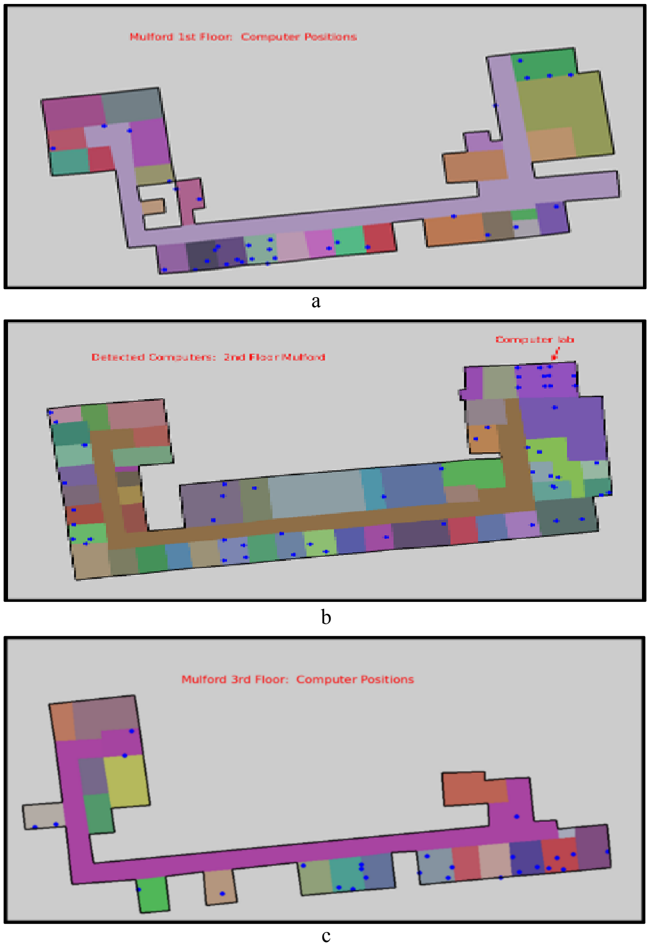
Mulford 1st Floor: Computer Positions
a
Detected Computers: 2nd Floor Mulford
Computer lab
b
Mulford 3rd Floor: Computer Positions
c

Fig. 9. Plots of detected computers on (a) first floor, (b) second floor and (c) third floor

For the detected computers on all the three floors, we can also calculate the 3D co-ordinates and visualize them on the respective floorplans as shown in Figure 9. The computers in the laboratory for Table 1 are in the upper right portion of the second floor.

REFERENCES

[1] G. Chen, J. Kua, S. Shum, N. Naikal, M. Carlberg, and A. Zakhor, “Indoor Localization Algorithms for a Human- Operated Backpack System,” In 3D Data Processing, Visualization, and Transmission 2010, Paris, France, May 2010.

[2] N. Corso and A. Zakhor, “Indoor Localization Algorithms for an Ambulatory Human Operated 3D Mobile Mapping System,” In Remote Sensing 2013, vol. 5, pp. 6611-6646, Oct. 2013.

[3] C. Cortes, and V. Vapnik, “Support-Vector Networks, in Machine Learning,” 20,
1995.

[4] R. Girshick, J. Donahue, T. Darrell, J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” In CVPR 2014.

[5] J. Kua, N. Corso, A. Zakhor, “Automatic Loop Closure Detection Using Multiple Cameras for 3D Indoor Localization,” In IS& T / SPIE Electronic Imaging, Burlingame, California, January 22-26, 2012.

[6] T. Liu, M. Carlberg, G. Chen, Jacky Chen, J. Kua, A. Zakhor, “Indoor Localization and Visualization Using a Human-Operated Backpack System,” In International Conference on Indoor Positioning and Indoor Navigation, 2010.

[7] N. Naikal, J. Kua, G. Chen, and A. Zakhor, “Image Augmented Laser Scan Matching for Indoor Dead Reckoning,” In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), St. Louis, MO, October 2009.

[8] E. Turner and A. Zakhor, “Floor Plan Generation and Room Labeling of Indoor Environments from Laser Range Data,” In GRAPP 2014, Lisbon, Portugal, January 2014.

[9] E. Turner and A. Zakhor, “Watertight Planar Surface Meshing of Indoor Point-Clouds with Voxel Carving,” In 3DV 2013.

[10] E. Turner and A. Zakhor, “Watertight As-Built Architectural Floor Plans Generated from Laser Range Data,” In 3DIMPVT 2012.

[11] V. Sanchez and A. Zakhor, “Planar 3D Modeling of Building Interiors from Point Cloud Data,” In ICP 2012.

[12] G. Singh, M. Jouppi, Z. Zhang, and A. Zakhor, “Shadow Based Building Extraction from Single Satellite Image,” SPIE Electronic Imaging Conference, Computational Imaging, San Francisco, CA, February 2015.

[13] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, “Selective search for object recognition, In IJCV, 2013.

[14] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte, and J. Glazer. “EnergyPlus: creating a new generation building energy simulation program,” Energy and Buildings, vol. 33, pp. 443-457, 2001.

[15] J. Z. Liang, N. Corso, E. Turner, and A. Zakhor, "Reduced-Complexity Data Acquisition System for Image Based Localization in Indoor Environments," IPIN 2013, Montbeliard, France, October 2013.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[17] O. Oreifej, J. Cramer, and A. Zakhor, "Automatic Generation of 3D Thermal Maps of Building Interiors," submitted to ASHRAE annual Conference, Seattle, WA, June 2014.

[18] E. Turner, P. Cheng, and A. Zakhor, "Fast, Automated, Scalable Generation of Textured 3D Models of Indoor Environments," IEEE Journal on Selected Topics in Signal Processing, 2014.

[19] J. Z. Liang, N. Corso, E. Turner, and A. Zakhor, "Image Based Localization in Indoor Environments," International Conference on Computing for Geospatial Research and Applications, San Francisco, CA, July 2013.

[20] See, R., Haves, P., Sreekanthan, P., Basarkar, M., O’Donnell, J. and Settlemyre, K. Development of a Comprehensive User Interface for the EnergyPlus Whole Building Energy Simulation Program. Proc. Building Simulation ’11, 2011.

[21] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Competition 2012 (ILSVRC2012). http://www.image-net.org/challenges/LSVRC/2012/.