Using Human Pose Estimation for User-Defined Indoor Location Sensing

Sinan Chen [1], Sachio Saiki [1], Masahide Nakamura [1,2]
[1]Graduate School of System Informatics, Kobe University, 1-1 Rokkodai-cho, Nada, Kobe, 657-8501, Japan
[2]RIKEN Center for Advanced Intelligence Project, 1-4-1 Nihonbashi, Chuo-ku, Tokyo, 103-0027, Japan
Email: chensinan@ws.cs.kobe-u.ac.jp, sachio@carp.kobe-u.ac.jp, masa-n@cs.kobe-u.ac.jp

***Abstract*—To implement indoor location sensing convenient and affordable for general households, this paper presents a novel method to simply realize user-defined indoor location sensing using fixed-point camera devices, human pose estimation technologies, and HTML canvas elements. Specifically, our method first captures images of a target space to the browser in real-time, then creates an HTML5 canvas element to draw the live image in the browser. Our key idea is to use the bounding-box-as-a-location (BBL) concept that users can define any rectangular area as a location in the live image with the computer mouse. Next, the coordinates of human joints in the live image are obtained using human pose estimation technologies. In this way, the system notifies the label of BBL, the sensing result of each body part ID, and the current time, as soon as the coordinates of human joints enter into any defined BBL. Users can develop various user-defined location-aware services using these notifications. Our method is possible to implement using inexpensive camera devices and normal computer browsers. Moreover, since all BBL is defined by each user freely, it does not require developers to perform any learning in each indoor environment. Therefore, general users can easily introduce it in the room. Using the proposed method, we conducted multiple experiments from one house to another. By performing the quantitative evaluation, the advantage and limitation are made explicit.**

***Index Terms*—location sensing, image, human pose estimation, HTML5 canvas element, bounding-box-as-a-location, smart home**

I. Introduction

With the rapid progress of Artificial Intelligence (AI) to its peak, developing an approach to adapt to different environments and users' needs is becoming urgent. Indoor location sensing is one of the key steps for extending *smart home* services, which has actively conducted for many years in the field of *ubiquitous computing*. In the conventional approaches for indoor location sensing, it is common to use sensors and/or signals such as Bluetooth [1], Beacon [2], Wifi [3], RFID [4], and wearable [5], in order to obtain location information of users. However, since they usually require a lot of devices and unique deployment for specific users and environments, the above existing technologies are yet far from simple implementation in users with special needs, such as detecting if the elderly have signs of falling off the bed.

Our interest is to implement convenient and user-operable location sensing that can allow custom locations by users for special needs. Unlike traditional indoor location sensing, the core of our thought is to regard each indoor location as a fixed area, then detecting which body key points exist in each

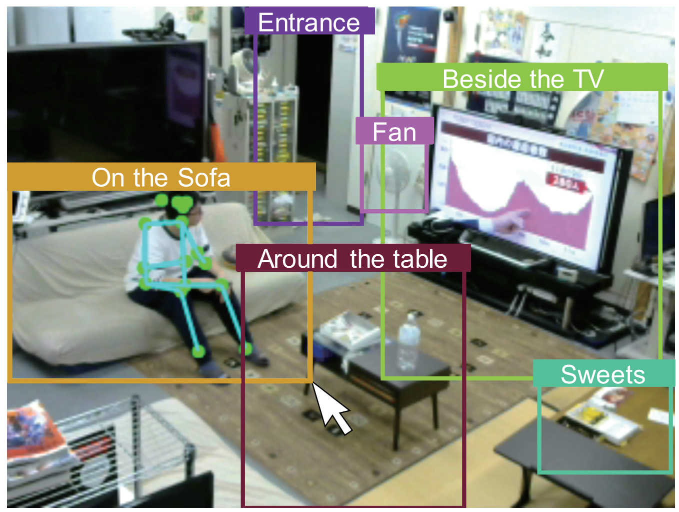

Fig. 1. Example of user-defined indoor location sensing with *BBL* concept

target area for realizing the location sensing. In human pose estimation technologies, we especially focus on the *PoseNet model* [6], which is a pre-trained and standalone pose estimation model performing in the browser with Tensorflow.js. Using it to estimate 17 body key points from the live image went well in the previous study [7], where the accuracy was about 70% within the error range of 10 pixels.

The goal of this paper is to propose a method that integrates fixed-point camera devices, human pose estimation technologies [8], and HTML canvas elements [9], for realizing *user-defined* indoor location sensing. In the proposed method, we first capture images of a target space to the browser in real-time. Afterward, we create an HTML5 canvas element to draw the live image in the browser. Meanwhile, we develop an HTML5 canvas drawing application that users can define any rectangular area (i.e., bounding box) in the live image with the computer mouse.

Our key idea is to create a *bounding-box-as-a-location (BBL)* concept, to apply human pose estimation to see if the poses of users are contained in any bounding box. More specifically, the coordinate relationship is checked in real-time, between the poses and bounding boxes. That is, the coordinate values of each body part are compared with the maximum and minimum coordinate values of each bounding box. Figure 1 represents an example of user-defined location sensing with *BBL* concept. Our method is possible to implement using inexpensive camera devices and general computer browsers. Hence, users with special needs can easily apply it to the room.

Based on the proposed method, we present a *content-oriented* architecture that integrates the Pub/Sub model [10], *Event-Condition-Action (ECA)* [11], and HTTP request methods [12] for extending the proposed method with *BBL* concept. We also conducted multiple experiments in different houses. In the experiments, we performed the quantitative evaluation using the size of *BBL*, shortest response time, and parameters of camera view (i.e., height, angle, and coverage).

The experimental results showed that the shortest response time was 0.21 seconds that the label was "bed" and the pixel size was 145×66, the longest time was 1.47 seconds that the label was "jug" and the pixel size was 25×36. It can be seen that the small bounding box may make the response time slower. As for the parameters of camera view, it did not seem to affect the response time of each *BBL*. We inferred that the detection of body key points in each target area compensates for the lack of distance judgment in the 2D space.

II. Preliminaries

A. Indoor Location Sensing

The indoor locations refer to any location information in the room, including locations of daily activities of residents, locations of the environment, locations of the status of the room. Typical Indoor locations are, for example, "in front of the entrance", "beside the window that the light is bright", "around the stove that it is heating up", etc.

We use the term *user-defined* indoor location to represent an indoor location that is specifically defined by individual residents, rooms, and environments, for a special purpose of the application. For example, support that a son is worried about his father living in a remote place alone with mild dementia. Then, the indoor locations like "beside the TV", "around the table", "in the bed", are crucial information for the son. If these user-defined indoor locations can be sensed by an elderly monitoring system, and the information is regularly sent to the son, it would be a great value for the son.

Moreover, for each indoor location, it can realize a value-added smart service that uses the location information and other devices in the timeline. For example, by integrating it and a speaker, it can regularly notify the elderly to drink water for hydration, when he is around the water cooler. The same approach can be applied to notify the elderly to open the window for ventilation. These notifications are of great significance for elderly people who often forget things.

B. Technical Challenges

The most common approach has been to obtain high-accuracy location information is to use sensor devices. In this approach, the user installs a sensor device S_1 to the target location L_1, where the location information is automatically collected by the sensor device. However, since this approach requires a lot of devices and unique deployment for specific users and environments, it is not easy to implement it in users with special needs. We consider that there are two challenging points for indoor location sensing in smart homes.

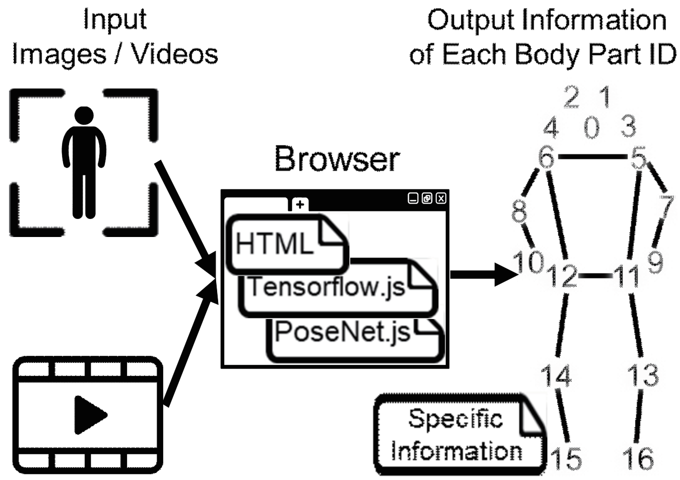

Fig. 2. Usage of the PoseNet model in single-person pose estimation

The first challenging point is *simplicity*. As seen in the above example, the indoor locations are defined by every user depending on a special purpose. Also, the layout, the environment, and the configuration of the target space vary from one house to another. Hence, the high-quality indoor location sensing service should easily construct a universal location sensing method that satisfies all users, houses, and environments.

The second challenge lies in *user operability*. Most existing technologies are developed and implemented in research labs or dedicated smart homes, and few of them are actually operated in real households. In order for the technology to be widely used, it should be easy to operate and maintain, should be affordable enough, and should not be intrusive to daily life. Of course, security and privacy issues influence acceptability. To improve them, in this paper, we present a method that all processing of raw data performed at the edge of devices without network connectivity.

C. Human Pose Estimation

Human pose estimation is one of the most challenging tasks in computer vision. It is common to estimate two-dimensional (2D) human poses that aim to obtain the localization of human joints (e.g., shoulders, elbows, knees, etc) from input images and videos [8]. Although 3D human pose estimation techniques have become increasingly popular over the last few years [13] [14] [15], the *expandability* and *practicability* of these techniques in smart homes still remain to be further studied.

The PoseNet model is a pre-trained 2D human pose estimation model based on machine learning that can perform in the browser with Tensorflow.js [6]. After importing the Tensorflow.js and PoseNet libraries, data input using an HTML element, such as an image or video tag can be performed.

Figure 2 represents the usage of the PoseNet model in single-person pose estimation.

A PoseNet model receives an image from the browser, detects 17 keypoints from a single person in the original input image, and returns the specific information of each key point in the browser console, such as "{'score': 0.81,'keypoints': [{'score': 1, 'part': 'nose', 'position': {'x': 200.02, 'y': 113.51}},...}", including accurate score, name, and 2D coordinates of the human joint. We consider that using the PoseNet model for smart homes is promising since data input and output via the browser, that can simply extend to various value-added smart services. Our previous study using the PoseNet model for home care monitoring is described in [7].

D. HTML5 Canvas Element

HTML5 is the fifth major version of *HyperText Markup Language (HTML)* appearing over ten years ago, to bring a mature web application platform that standardized all interaction with the computer, such as videos, audios, and images. One of the most powerful new elements in HTML5 is arguably the canvas element that allows developers to draw directly onto the defined 2D area via the HTML5 canvas element and JavaScript codes in the browser. The code sample of drawing a rectangle on canvas is shown in Listing 1.

The related work is common to use for visualizing data over the web [16], platform independent development [17], and creating 2D animations [18]. Unlike the conventional approach, we aim to establish the coordinate relationship between human pose estimation and user-defined rectangle via HTML5 canvas element in the browser.

Listing 1. Code sample of drawing a rectangle on canvas

```
<canvas id="canvas" width="320" height"="240">
</canvas>
<script type="text/javascript">
    var canvas =
            document.getElementById("canvas");
    var context = canvas.getContext("2d");
    context.storkeStyle = "red";
    context.strokeRect(50, 150, 100, 50);
</script>
```

III. User-Defined Indoor Location Sensing

In this section, we propose a pragmatic method that implements user-defined indoor location sensing. In the proposed method, we capture the live image of the target area as essential data in the browser. In order to achieve convenient and affordable indoor location sensing, we extensively introduce the *bounding-box-as-a-location (BBL)* concept using human pose estimation and the HTML5 canvas element.

A. Bounding Box as a Location (BBL)

We consider implementing indoor location sensing using the live image by installing a camera in the target space. This is because camera devices (e.g., USB camera) are already so common that the general users can easily introduce the cameras in the room.

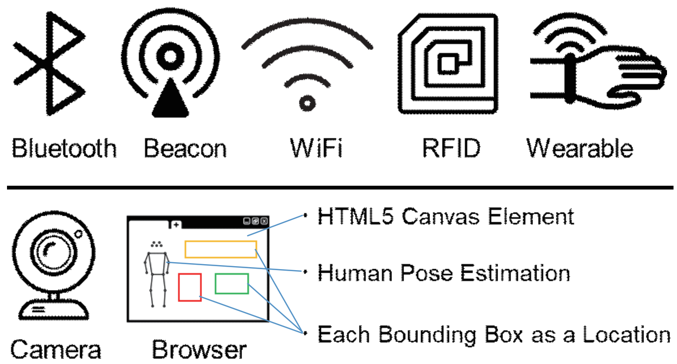

Fig. 3. Comparison of materials between sensor-based and proposed methods

Instead of using sensor devices, the proposed method extensively utilizes human pose estimation and the HTML5 canvas element (see Section II-C and II-D). In the proposed method, we acquire the live image of the target space and create an HTML5 canvas element to draw the live image in the browser. Then, the live image is input into the PoseNet model. As an output, the PoseNet Model returns the coordinates of human joints, which represent human poses that the PoseNet model estimated within the live image. Our key idea is to develop an HTML5 canvas drawing application that users can define any rectangular area (i.e., bounding box) as a location in the live image. Thus, each bounding box (as a range of coordinate pixels) is defined as a target location L_i. This is the *bounding-box-as-a-location* concept.

Once the coordinates of human joints enter into any defined *BBL*, the system is possible to notify the label of *BBL*, the body part ID of human joints, and the time, which are much less expensive than sensor devices. There are various methods that draw rectangles onto the HTML5 canvas element, including [19], [20], and [9]. Using one of these methods, each *BBL* is defined onto a live image. By notifying a private (specific) location information to users, the proposed method extends a value-added smart service of custom indoor locations.

Figure 3 shows a comparison of materials between the conventional sensor-based method and the proposed bounding-box-as-a-location method. The upper part of the figure shows materials in the conventional method, where often install many sensors in the room, such as Bluetooth [1], Beacon [2], Wifi [3], RFID [4], and wearable [5].

The lower part of the figure shows the proposed method with the camera device and the browser. The captured live image is drawn in the HTML5 canvas that users can define any *BBL* in the live image. For each bounding box, the system notifies automatically when the coordinates from the live image-based human pose estimation existed. In this way, we aim to realize user-defined indoor location sensing in different households by installing a single camera and the browser instead of many sensor devices.

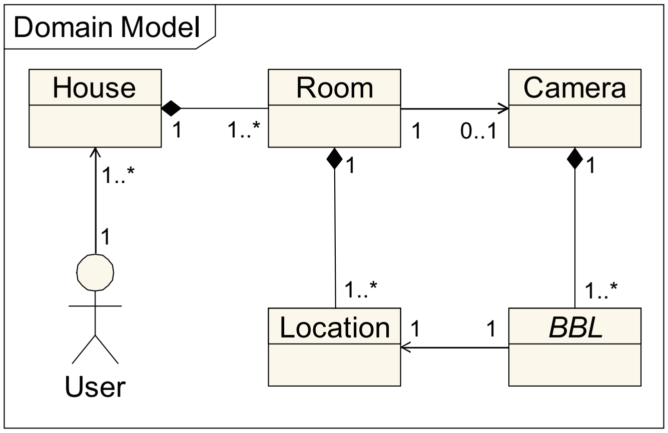

Fig. 4. Domain model in the proposed method

B. Proposed Method

Now, we produce a complete description of the proposed method using the *bounding-box-as-a-location (BBL)* concept. The domain model is shown in Figure 4. More specifically, the proposed method consists of the following five steps:

Step 1: Acquiring the live image

The user deploys a camera device (e.g., USB camera) in a target room of the house. Using a camera, the system periodically draws the live image of the target space onto the HTML5 canvas element in the browser.

Step 2: Defining *BBL* to label

In the HTML5 canvas element , the user draws n *BBL*s (rectangles), i.e., $L = \{L_1, L_2, ..., L_n\}$ with the computer mouse. For each target location L_i $(1 \leq i \leq n)$, the user defines a unique *Label*(L_i) that represents the location L_i, and obtains the maximum and minimum values of 2D coordinates X and Y in L_i, i.e., $maxX$, $minX$, $maxY$, and $minY$.

Step 3: Calling human pose estimation

The specific information of human joints in the live image is estimated by the single/multiple-person PoseNet model in the browser, including the coordinates and the body part name for each person pose. Using information of the estimated poses $P = \{P_1, P_2, ..., P_m\}$, each pose P_j $(1 \leq j \leq m)$ is drawn in the HTML5 canvas element. Note that m is a changing number depending on the real number of people in the target space. For each pose P_j, the 2D coordinate X and Y of 17 keypoints $P_{jk} = \{P_{j1}, P_{j2}, ..., P_{j17}\}$ are contained (see Figure 2).

Step 4: Analyzing output coordinates

For each custom *BBL* L_i, the coordinates of each pose P_j is checked if it existed in L_i. Since there are n bounding boxes, m poses, and 17 keypoints contained in each pose, this step checks totally $m \times n \times 17$ sets of coordinates. Comparing each set $Coordinates(L_i, P_j, P_{jk})$ to check if any keypoint P_{jk} exists in *BBL* L_i. That is, to confirm if both of $\{minX \leq X \leq maxX\}$ and $\{minY \leq Y \leq maxY\}$ are trued in each set $Coordinates(L_i, P_j, P_{jk})$.

Step 5: Designing notifications

Using a binary $Array(L_i, P_j, P_{jk})$ arranged by the body part ID that the number 0 means "no exist" and the number 1 means "exist", the expression of the sensing result for each *BBL* L_i is so clear. As the notification from the system, *Label*(L_i), $Array(L_i, P_j, P_{jk})$, and $Time(L_i, P_j, P_{jk})$ are described with the object type, as soon as the keypoint P_{jk} enter into any defined *BBL* L_i. Listing 2 depicts an example of the notification when the left wrist exists in this *BBL*.

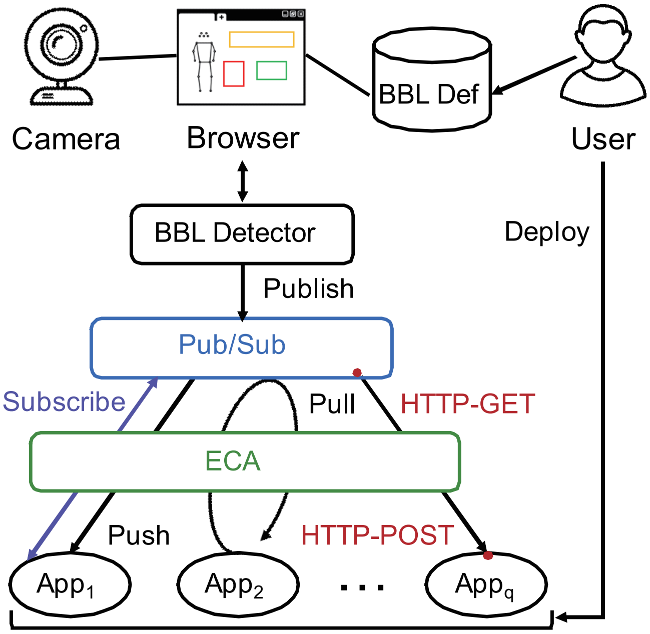

Fig. 5. Content-oriented architecture for extending the proposed method

Listing 2. Example of the *BBL* notification when the left wrist exists

```
{
        "label": "in front of the entrance",
        "sensing result": [
                0, // nose           (ID: 0)
                0, // leftEye        (ID: 1)
                0, // rightEye       (ID: 2)
                0, // leftEar        (ID: 3)
                0, // rightEar       (ID: 4)
                0, // leftShoulder   (ID: 5)
                0, // rightShoulder  (ID: 6)
                0, // leftElbow      (ID: 7)
                0, // rightElbow     (ID: 8)
                1, // leftWrist      (ID: 9)
                0, // rightWrist     (ID: 10)
                0, // leftHip        (ID: 11)
                0, // rightHip       (ID: 12)
                0, // leftKnee       (ID: 13)
                0, // rightKnee      (ID: 14)
                0, // leftAnkle      (ID: 15)
                0  // rightAnkle     (ID: 16)
        ],
        "time": "2020-11-10T10:51:01.75"
}
```

C. Extension

We provide a *content-oriented* architecture that extends the proposed method with *bounding-box-as-a-location (BBL)* concept, for a delivered notification of indoor locations.

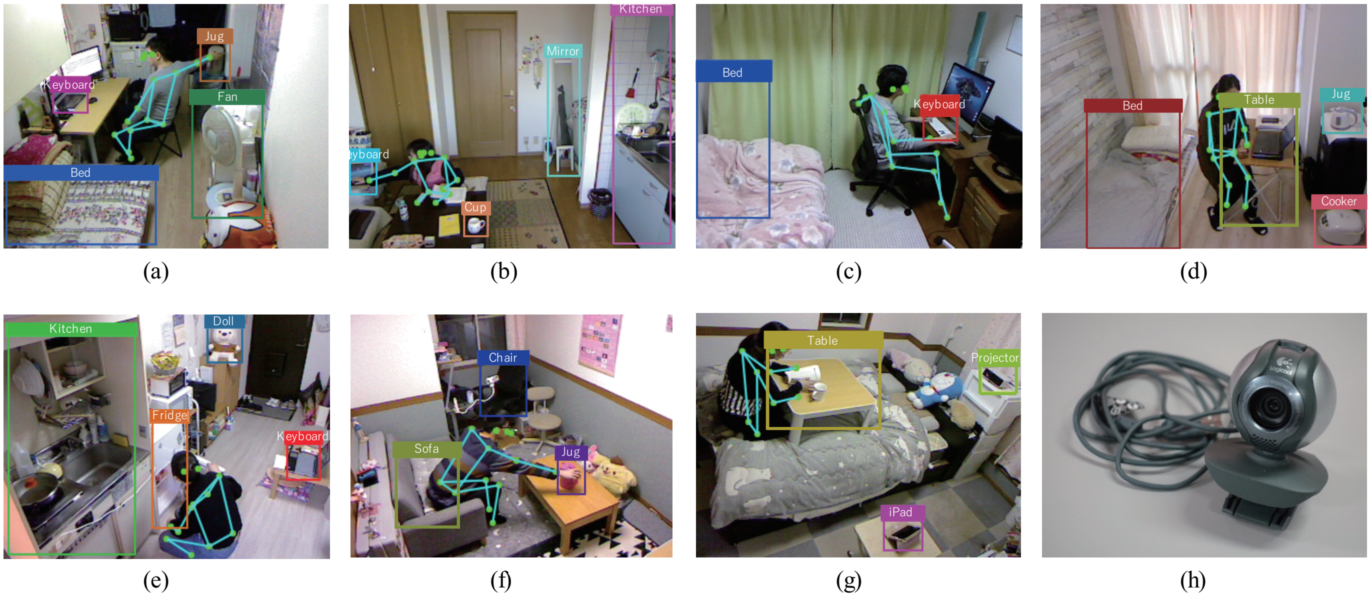

Fig. 6. Representative images: (a) - (g) Experiments conducted in 7 houses. (h) A USB camera.

Figure 5 depicts the essential part of the established architecture. In the figure, *BBL Def* represents a defined set of indoor locations by the user. Each output coordinates are analyzed via a *BBL* detector in the browser. Then, the *BBL* notification as one of the topics is published into a compact Pub/Sub model, as soon as the *BBL* sensing changes existed. The Pub/Sub model defines how to publish and subscribe messages toward every content nodes. The related work includes [10] and [21], and [22].

The user deploys multiple value-added smart services $App = \{App_1, App_2, ..., App_q\}$ based on the published *BBL* notification. For each App_r ($1 \leq r \leq q$), the designate *BBL* notification is subscribed or filtrated, which depending on the *Event-Condition-Action (ECA)* rules [11] and HTTP request methods (i.e., *POST* and *GET*) [12]. *BBL* notification as a topic makes each App_r (subscriber) and *BBL* detector (publisher) perform independently of one another, where the messaging without contact. In this way, it is promising that not only implement various location-based value-added smart services in a single room but also satisfy the needs of different users and indoor environments.

IV. Evaluation

A. Experimental Setup

We aim to evaluate the *user-defined* indoor location sensing from several practical aspects. Specifically, for the proposed method presented in Section III-B, we define three kinds of basic metrics: *accuracy*, *applicability* and *portability*. The accuracy means the ratio of the number of detected body key points in the defined area to the total number of key points when the defined area completely covers the body. The applicability means whether the proposed method responds to users' needs, and we use the shortest response time to reflect this. The portability means the applicability of using the same method in different environments. For this, we add three parameters of the camera view (i.e., height, angle, coverage) to the experimental evaluation.

TABLE I
Experiment settings

Experimental subject	Single-person
Target space	Single-room
Number of target space	7
Shooting device	USB camera (Logitech OEM B500)
Shooting interval	30 fps
Computer browser	Chrome (Version 84.0.4147.125)
HTML5 canvas size (*pixel*)	320×240
Human pose estimation	PoseNet model

Table I summarizes the universal settings of every experiment. To perform experiments simple and effective, we deploy the proposed method to the computer browser. Using the camera device and computer browser, then opening a specific file, users can directly define any *BBL* in the live image. As for experimental data, all notifications and image files are saved in the local automatically. Figure 6 shows the representative images. Note that each *BBL* was defined by users, including its label name and size.

B. Evaluation method

In the metric of *accuarcy*, we consider that the accuracy of the proposed method completely depends on the PoseNet model. As described in Section I, we previously evaluated the *accuracy* of *PoseNet model* was about 70% within the error range of 10 pixels [7]. In the metric of *applicability*, considering that the layout and environment are different in

TABLE II
PARAMETERS AND RESULTS OF BOUNDING BOXES IN 7 TARGET SPACES

Target space	Bounding box		
	Label	Size (*pixel*)	response time (*s*)
(a)	Keyboard	32×24	0.93
	Jug	27×40	0.60
	Bed	145×66	0.21
	Fan	67×114	0.35
(b)	Keyboard	24×35	1.44
	Cup	21×24	1.36
	Mirror	29×119	0.66
	Kitchen	54×226	0.22
(c)	Bed	69×136	0.39
	Keyboard	31×26	0.77
(d)	Bed	89×139	0.37
	Table	72×113	0.13
	Jug	36×32	1.25
	Cooker	49×38	0.58
(e)	Kitchen	122×216	0.32
	Fridge	32×105	0.65
	Doll	32×36	1.43
	Keyboard	30×37	1.05
(f)	Sofa	59×71	0.58
	Jug	25×36	1.47
	Chair	44×53	0.73
(g)	Table	109×82	0.31
	iPad	36×31	0.63
	Projector	33×29	0.84

seven target spaces, we give users full autonomy to decide the location label, size, and the number of the bounding boxes.

Regarding the process of every experiment, for each defined location, the user needs to approach and leave it, using any body parts. We regard the *shortest* time interval of notifications as the response time of that location, which is caused by each change in the sensing result (see Listing 2). In the metric of *portability*, we summarize parameters of the camera view in each target space, including the camera height, view angle, and coverage size. Combining the above two metrics, we evaluate the reliability of the proposed method in different houses.

C. Results

Table II represents the parameters and results of bounding boxes in seven target spaces. From the results of the response time, the shortest time is 0.21 seconds that the label "bed" in the target space (b), the longest time is 1.47 seconds that the label "jug" in the target space (f). From the overall results, defining the small bounding box may make the response time slower, where the threshold pixel value (width/height) is about 40. Table III summarizes the parameters of camera view in 7 target spaces. From the description of Table III and II, we found that the relatively long response times happened more in the big view area and small bounding box. Moreover, it did not seem to affect the response time of each bounding box that even tilt the proper angle or adjust the height of the camera.

TABLE III
PARAMETERS OF CAMERA VIEW IN 7 TARGET SPACES

Target	Camera View		
space	Height (m)	Angle	Coverage (m^2)
(a)	1.9	Front	6
(b)	1.7	Front	9
(c)	1.8	Front	4
(d)	1.7	Front	4
(e)	1.9	Slope	8
(f)	1.8	Slope	4
(g)	1.8	Slope	6

V. DISCUSSION

A. Advantage

The advantages of the proposed method are summarized as follows:

- Since all *BBL* is defined by each user freely, the flexibility in different users and indoor environments is improved. It does not require developers to perform any learning in each indoor environment.
- Since user-defined indoor location sensing is realized by a inexpensive camera device and a normal computer browser, the proposed method does not require sensor devices to be installed in each location. Hence, it can be easily applied to the room for users with special needs.
- Since the PoseNet model can be performed in the browser without network connectivity, the security and privacy issues when processing raw image data is well improved.

B. Limitation

As the drawback in user-defined indoor sensing with *BBL* concept, the following issues are currently anticipated:

- For users whose indoor environment changes frequently, such as locations of furniture, it takes time to redefine *BBL* every time. The proposed *BBL* concept is more applicable in each location that seldom changes its meaning.
- Due to using a single camera in the target room, the viewing angle of estimating human pose is limited. Hence, to avoid users covering the camera at short range when defining dense *BBL*, the location and height of installing the camera will be a big challenge.

C. Camera Specification

We just simply used a general USB camera for the above experiments. However, for the diversity of users' needs from different households, the general USB camera will have insufficient functions, such as image data capture in the dark space is not easy to achieve. For this, we recommend using an infrared camera if users need to shoot images at night. In addition, for users who need to shoot the image with big coverage, using a wide-angle camera may be a fine option.

D. Related Work

The research of indoor location sensing within smart homes is generally based on three ways as follows: energy, sensor, and signal. Deng et al. [23] provided data that shows energy-based methods of sound source localization can improve localization accuracy. In [24], a smartphone inertial sensor-based indoor localization with occasional iBeacon corrections is proposed. As for the signal-based approaches, Hyun et al. [25] presented the wireless UWB-based indoor localization method using the ray-tracing algorithm. They basically adopt assisted data that depend on machine learning, specific devices and environments. The major difference from their approaches is, we present a human-centric approach that deploying fewer devices and improving operability for users with special needs.

VI. Conclusion

In this paper, we have presented a method using fixed-point camera devices, human pose estimation technologies, and HTML canvas elements for realizing user-defined indoor location sensing with the *bounding-box-as-a-location (BBL)* concept. We have also presented a *content-oriented* architecture, conducted multiple experiments in different houses, performing the quantitative evaluation in *applicability* and *portability*. In our future work, we evaluate the limitation of the proposed method from a more comprehensive viewpoint. For this, we are currently extending it with more users and more types of cameras in the same target space. Based on the proposed method, we will also evaluate the feasibility of indoor human behavior detection in the time series. Inspired by an interesting topic called the *human-object interaction (HOI)* [26], we plan to extend the *BBL* concept to define any *one-off* rectangular area that contains a movable object (e.g., a cup).

Acknowledgment

This research was partially supported by JSPS KAKENHI Grant Numbers JP19H01138, JP18H03242, JP18H03342, JP19H04154, JP19K02973, JP20K11059, JP20H04014, JP20H05706 and Tateishi Science and Technology Foundation (C) (No.2207004).

References

[1] M. Terán, J. Aranda, H. Carrillo, D. Mendez, and C. Parra, "Iot-based system for indoor location using bluetooth low energy," in *2017 IEEE Colombian Conference on Communications and Computing (COLCOM)*. IEEE, 2017, pp. 1–6.

[2] H. Hoshi, H. Ishizuka, A. Kobayashi, and A. Minamikawa, "An indoor location estimation using ble beacons considering movable obstructions," in *2017 Tenth International Conference on Mobile Computing and Ubiquitous Network (ICMU)*. IEEE, 2017, pp. 1–2.

[3] M. N. Husen and S. Lee, "Indoor location sensing with invariant wi-fi received signal strength fingerprinting," *Sensors*, vol. 16, no. 11, p. 1898, 2016.

[4] Y. Á. López, M. E. de Cos Gómez, and F. L.-H. Andrés, "A received signal strength rfid-based indoor location system," *Sensors and Actuators A: Physical*, vol. 255, pp. 118–133, 2017.

[5] J. Huang, X. Yu, Y. Wang, and X. Xiao, "An integrated wireless wearable sensor system for posture recognition and indoor localization," *Sensors*, vol. 16, no. 11, p. 1825, 2016.

[6] D. Oved, I. Alvarado, and A. Gallo, "Real-time human pose estimation in the browser with tensorflow. js," *TensorFlow Medium, May*, 2018.

[7] S. Chen, S. Saiki, and M. Nakamura, "Nonintrusive fine-grained home care monitoring: Characterizing quality of in-home postural changes using bone-based human sensing," *Sensors*, vol. 20, no. 20, p. 5894, 2020.

[8] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.

[9] R. R. Ganji, M. Mitrea, B. Joveski, and F. Preteux, "Html5 as an application virtualization tool," in *2012 IEEE 16th International Symposium on Consumer Electronics*. IEEE, 2012, pp. 1–4.

[10] R. S. Gargees and G. J. Scott, "Dynamically scalable distributed virtual framework based on agents and pub/sub pattern for iot media data," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 599–613, 2018.

[11] C. Vannucchi, M. Diamanti, G. Mazzante, D. Cacciagrano, R. Culmone, N. Gorogiannis, L. Mostarda, and F. Raimondi, "Symbolic verification of event–condition–action rules in intelligent environments," *Journal of Reliable Intelligent Environments*, vol. 3, no. 2, pp. 117–130, 2017.

[12] T. Hokaguchi, Y. Ohsita, T. Shibahara, D. Chiba, M. Akiyama, and M. Murata, "Detecting malware-infected hosts using templates of multiple http requests," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2020, pp. 1–2.

[13] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single rgb camera," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–14, 2017.

[14] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang, "3d human pose estimation in the wild by adversarial learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5255–5264.

[15] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7753–7762.

[16] C. A. Miller, J. Anthony, M. M. Meyer, and G. Marth, "Scribl: an html5 canvas-based graphics library for visualizing genomic data over the web," *Bioinformatics*, vol. 29, no. 3, pp. 381–383, 2013.

[17] S.-W. Jang and M.-H. Huh, "Platform independent game development using html5 canvas," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 18, no. 12, pp. 3042–3048, 2014.

[18] L. VÉGH and J. UDVAROS, "Possibilities of creating interactive 2d animations for education using html5 canvas javascript libraries." *eLearning & Software for Education*, vol. 2, 2020.

[19] A. Taivalsaari and T. Mikkonen, "The web as an application platform: The saga continues," in *2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications*. IEEE, 2011, pp. 170–174.

[20] M. Anttonen, A. Salminen, T. Mikkonen, and A. Taivalsaari, "Transforming the web into a real application platform: new technologies, emerging trends and missing pieces," in *Proceedings of the 2011 ACM Symposium on Applied Computing*, 2011, pp. 800–807.

[21] A. E. Redondi, A. Arcia-Moret, and P. Manzoni, "Towards a scaled iot pub/sub architecture for 5g networks: The case of multiaccess edge computing," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, 2019, pp. 436–441.

[22] T. Ding, S. Qian, J. Cao, G. Xue, and M. Li, "Scsl: Optimizing matching algorithms to improve real-time for content-based pub/sub systems," in *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2020, pp. 148–157.

[23] F. Deng, S. Guan, X. Yue, X. Gu, J. Chen, J. Lv, and J. Li, "Energy-based sound source localization with low power consumption in wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4894–4902, 2017.

[24] Z. Chen, Q. Zhu, and Y. C. Soh, "Smartphone inertial sensor-based indoor localization and tracking with ibeacon corrections," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1540–1549, 2016.

[25] J. Hyun, T. Oh, H. Lim, and H. Myung, "Uwb-based indoor localization using ray-tracing algorithm," in *2019 16th International Conference on Ubiquitous Robots (UR)*. IEEE, 2019, pp. 98–101.

[26] T. Wang, T. Yang, M. Danelljan, F. S. Khan, X. Zhang, and J. Sun, "Learning human-object interaction detection using interaction points," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4116–4125.