MicroCam: Leveraging Smartphone Microscope Camera for Context-Aware Contact Surface Sensing

YONGQUAN HU, University of New South Wales, Australia
HUI-SHYONG YEO, Huawei, China
MINGYUE YUAN, University of New South Wales, Australia
HAORAN FAN, University of New South Wales, Australia
DON SAMITHA ELVITIGALA, University of New South Wales, Australia
WEN HU, University of New South Wales, Australia
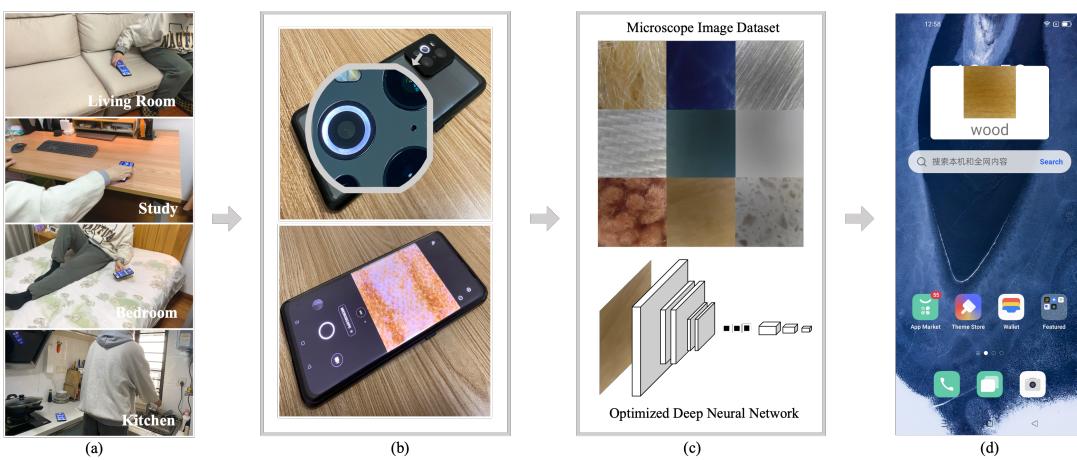AARON QUIGLEY, University of New South Wales, Australia

Fig. 1. The MicroCam system workflow and use case entail: (a) Placing mobile phones on various surfaces in different scenarios; (b) The upper subfigure displays the phone's rear with an active microscope camera and aperture, while the lower subfigure exhibits the captured microscopic image; (c) Collected images form a dataset (upper subfigure) for training an optimized deep neural network, designed for efficient microscopic surface texture classification (lower subfigure); (d) Surface detection triggers a backend service, launching relevant applications. For instance, if wood is detected, contextual inferences may suggest a study environment.

Authors' addresses: Yongquan Hu, yongquan.hu@unsw.edu.au, University of New South Wales, Australia; Hui-Shyong Yeo, yeo.hui.shyong@huawei.com, Huawei, China; Mingyue Yuan, mingyue.yuan@unsw.edu.au, University of New South Wales, Australia; Haoran Fan, haoran.fan@student.unsw.edu.au, University of New South Wales, Australia; Don Samitha Elvitigala, s.elvitigala@unsw.edu.au, University of New South Wales, Australia; Wen Hu, wen.hu@unsw.edu.au, University of New South Wales, Australia; Aaron Quigley, aquigley@acm.org, University of New South Wales, Australia.

arXiv:2407.15722v1 [cs.HC] 22 Jul 2024

The primary focus of this research is the discreet and subtle everyday contact interactions between mobile phones and their surrounding surfaces. Such interactions are anticipated to facilitate mobile context awareness, encompassing aspects such as dispensing medication updates, intelligently switching modes (e.g., silent mode), or initiating commands (e.g., deactivating an alarm). We introduce MicroCam, a contact-based sensing system that employs smartphone IMU data to detect the routine state of phone placement and utilizes a built-in microscope camera to capture intricate surface details. In particular, a natural dataset is collected to acquire authentic surface textures in situ for training and testing. Moreover, we optimize the deep neural network component of the algorithm, based on continual learning, to accurately discriminate between object categories (e.g., tables) and material constituents (e.g., wood). Experimental results highlight the superior accuracy, robustness and generalization of the proposed method. Lastly, we conducted a comprehensive discussion centered on our prototype, encompassing topics such as system performance and potential applications and scenarios.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; *User interfaces—Input devices and strategies.*

Additional Key Words and Phrases: Sensing; surface sensing; macro-camera; microscope camera; mobile interaction.

1 INTRODUCTION

Advances in pervasive computing, especially mobile-computing, have opened up the potential of context-aware interactions to become part of the fabric of our daily life [17, 49, 66]. However, many nascent forms of context-aware interaction still require extra learning, configuration and user maintenance to achieve the appearance of a seamless and carefree interaction, that many desire. An understanding of what the user wishes, as a by product of actions they are performing anyway, remains a challenge. In MicroCam, we design, develop and evaluate a form of mobile interaction, based on the placement action the user is going to perform anyway, through the use of a phone with a microscope camera. This work is situated within the wider advancements of positioning and sensing technologies which have afforded new interaction techniques based on context perception. More specifically, within these advancements, a range of discreet and subtle techniques have emerged [13, 30, 38].

Surface and material sensing technology [15, 20, 25, 34, 45, 47, 63, 66] offers the potential for more fine-grained mobile phone context awareness compared to other localization techniques reliant on communication signal detection (e.g., GPS [64], Bluetooth [41], GSM [42], WiFi [19, 28, 29], or multi-signal fusion [17]). In the "Placement Awareness" paradigm [20], a crucial facet of context awareness, users gain insight into the specific location of their mobile devices, such as on a bed, a desk, or in their pocket, rather than merely the general vicinity of a bedroom or living room, which may provide further inference regarding the user's current potential behavior. Among various sensing categories (e.g., optical [20, 45, 47, 66], acoustic [25], magnetic [15]), optical-based sensing methodologies [20, 45, 47, 66] have demonstrated superior performance in differentiating materials and achieving improved recognition outcomes. Despite minor variations in the devices utilized (e.g., SpeCam [66] employs the reflection of a mobile phone's front screen and front camera, while SpectroPhone [47] depends on the reflection of external LEDs and the rear camera), the fundamental sensing principle is consistent: unique materials yield diverse spectral reflectance. However, spectral discrimination may occasionally lead to information loss. In contrast, color images with an original optical base provide more comprehensive features and finer-grained information (e.g., vivid and distinct material textures) compared to reflected spectrum. This wealth of information can be harnessed for a wider array of application scenarios, such as recognizing exceptionally small QR code patterns on surfaces, as depicted in Figure 10, which is unattainable using surface perception based on spectral recognition alone. Consequently, acquiring this enriched information is essential for inferring the semantic context of user behavior in context-aware applications. Furthermore, with the growing prevalence of macro photography in commercial

mobile phones and the advent of devices featuring "ultra-close macro lenses" (microscope lenses) (e.g., Oppo Find X3 Pro, Realme GT2 Pro), the potential to capture RGB images of surfaces during camera placement arises. In an effort to capture the intricate texture details present in microscopic images, DNNs (Deep Neural Networks) have emerged as highly effective and prevalent methods in recent years [61]. However, they also encounter several technical challenges, such as limited training data, substantial computational complexity, and restricted robustness and generalizability. To address these issues, we conducted through a three-pronged approach: (1) collecting a custom dataset for training the network; (2) selecting a comparatively lightweight MobileNet architecture to balance performance and computational complexity; (3) incorporating continual learning into our system to enhance the algorithm's performance.

Meanwhile, beyond the algorithmic performance of the technology itself, we also place considerable emphasis on the practicality [52], usability [5, 57], and user-friendliness [9, 59] from a user-centered system design perspective [1, 16], aspects that have been limited addressed in previous work. In terms of practicality, accurate scene perception and precise action detection often necessitate the attachment of multiple sensors or devices, such as radar sensors [65], optical sensors [20, 47], or multi-spectral sensors [45]. For example, SpectroPhone achieved the inspiring 99% accuracy for 30 distinct materials using warm and cool white LEDs in conjunction with a smartphone's rear camera [47]. However, such strategies assume the presence of external hardware support and are incompatible with off-the-shelf devices. From the perspective of usability and user-friendliness, SpeCam [66], for instance, can detect surfaces using only the built-in front camera but requires users to place the phone face down with an appropriate bumper case (3mm thickness), which is not a typical orientation for screen usage and obstructs visibility of front display notifications. Conversely, methods that capitalize on pre-existing user interactions align more closely with user habits, enabling novel forms of mobile context-aware computing. While the future application potential of this technology is vast, users currently face challenges in readily acquiring the prototype and integrating it into their everyday routines. Overall, our objective is to leverage flexible, consumer-friendly approaches that are easily integrated, user-friendly, and compatible with popular devices, rather than relying on fixed and additional sensors or infrastructures. In this context, we seek to explore and expand the boundaries of interaction based on built-in sensors in existing commercial mobile phones without requiring additional attachments. Secondly, a diverse range of cameras integrated into assorted mobile devices (e.g., GoPro, smartwatches) are also consistently situated on surfaces in various environments, further substantiating the need for camera-based contact sensing. Additionally, the datasets obtained from real-world environments impose minimal constraints on users, which could be expected to further augment the versatility of our interactive technology to enhance the user experience. Most importantly, sensing detection should occur unobtrusively and be executed automatically without necessitating additional user learning, such as accompanying unintentional behavior, e.g., implicit interaction or subtle interaction [26, 32, 38, 46, 48]. Thus, we employ the mobile phone's IMU (Inertial Measurement Unit) data for loop detection, activating the sensing algorithm to capture surface images upon placement states. This quiet, non-disruptive method requires no additional learning costs.

In summary, this paper presents an efficient, lightweight context-aware sensing system utilizing the built-in microscope camera of a mobile device (as shown in Figure 1). Firstly, the IMU sensor is used to monitor the movement status of the mobile phone since people put them anywhere in daily life (Figure 1 (a)). Then, once the state of the mobile phone being placed still is detected, the "microscope camera" mode is activated to capture microscopic images of surfaces on which the phone is naturally or unintentionally placed (Figure 1 (b)). Next, we employ a deep neural network (MobileNet) optimized based on continual learning (Experience Replay method) for object and material recognition (Figure 1 (c)). Lastly, some context-aware information can be inferred based on detection results (Figure 1 (d)). Specifically, our contributions include:

- A contact sensing system that integrates built-in hardware (an IMU sensor and a microscope camera of smartphone) and software (an optimized deep neural network). This system detects the placement status of a smartphone and recognizes the surface texture beneath, aiding in mobile context-awareness;
- An open dataset of surface texture images, featuring object types and material properties, with rich color and detail, obtained through an object-oriented natural acquisition method during everyday smartphone placement activities to improve the generalization of the proposed approach;
- A improved MobileNet based on continual learning for microscopic images recognition, offering bolstered robustness and reinforced generalization of the algorithm.

2 RELATED WORK

Our related work spans multiple research areas including context-aware computing, surface and material sensing, in particular for input and interaction, and image recognition based on deep neural networks.

2.1 Context-Aware Computing

UbiComp (Ubiquitous Computing) represents a post-desktop paradigm of human-computer interaction, in which "context is any information that can be used to describe the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves" [8]. Mobile context-awareness encompasses information such as spatial detail, identity, user details, temporal details, environmental factors, social context, resources, computing factors, physiological data, activities, schedules, and agendas. It aims to provide services that are adaptive, proactive, and automatic, reducing the cognitive burden on users. However, existing work often necessitates explicit interactions, rather than capitalizing on interactions people would naturally undertake.

Pervasive interface usability metrics usually emphasize learnability, efficiency, memorability, error resistance, satisfaction, conciseness, transparency, and invisibility [40, 43]. The final metric, invisibility, assesses the degree to which the interface remains unobtrusive when it could have inferred or deduced the answer. MicroCam exemplifies this principle, harnessing insights gleaned from natural interactions to deduce contextual state information (e.g., activity, location, social status). By merely requiring users to place their phones on various surfaces as they typically would, without necessitating additional learning, MicroCam effectively facilitates mobile context awareness.

The concept of "placement-awareness" [20] emerged as a noteworthy manifestation of context-awareness, demonstrating the value of material identification for both location and activity recognition through placement detection. This approach sought to differentiate materials when a custom-built multispectral optical sensor is placed on various surfaces. Although Harrison et al. acknowledged the significance of material and surface perception based on placement for contextual situation inference, the interactive intent and experience of the "placement action" itself are not comprehensively addressed. Moreover, limited by early technological advancements, the prototype is not extensively integrated with mobile devices, leading to the development of a self-contained, lightweight, and portable standalone unit. While this design is versatile, it may not be ideally suited for specific scenarios.

In contrast, the widespread use of mobile phones and wearables today enables the proposed MicroCam system to offer unique application scenarios while permitting implicit interaction with computing systems based on routine activities.

2.2 Contact Surface and Material Sensing

Contact surface sensing has emerged as a burgeoning research topic, spanning the fields such as UbiComp, HCI (Human-Computer Interaction), IoT (Internet of Things) and robotics [21, 25, 51, 60, 63, 65]. The capacity to

sense contact surfaces or their materials can facilitate a variety of applications, encompassing but not limited to manufacturing, robotic gripping, context-aware computing, and household surface interactions [58]. Microwave, acoustic and optical sensing are the most commonly used methods for surface sensing [15, 20, 25, 28, 45, 47, 63, 65, 66]. Microwave sensing of surface materials offers advantages such as the ability to penetrate through various weather conditions and detect objects at long distances with high resolution. Additionally, it can provide valuable data on material properties and subsurface features [65]. However, disadvantages include signal interference from other sources and potential difficulties in distinguishing between similar materials. Moreover, the technology can be limited by high power requirements and the need for specialized equipment[28]. Moreover, acoustic sensing uses sound waves to measure surface properties such as thickness and stiffness [53], and is often used to detect defects in materials. However, it may not be suitable for all surfaces. Furthermore, spectroscopy is highly accurate in identifying a material's chemical composition but can be expensive [39]. Meanwhile, optical inspection is a widely used non-destructive method for surface sensing that measures the reflectivity and color of a surface using light-based sensors [20, 45, 47, 66]. It is relatively inexpensive and accessible to a wider range of users. This method can provide highly detailed and accurate information about surface properties, including texture, roughness, and reflectivity.

Our study primarily focuses on optical or visual surface sensing methods. A notable example is the work by Harrison et al. [20] which utilized a custom-built sensor comprising a photoresistor and light-to-frequency converter, achieving a material accuracy of 94.4% for 27 placements. SpecTrans [45] extended this concept by incorporating multi-spectral sensing, obtaining 99.0% accuracy even for transparent materials. In addition, Erickson et al. [11] employed a spectrometer, achieving a material classification accuracy of 94.6%. Schrapel et al.'s SpectroPhone [47] identified 30 distinct materials using warm and cool white LEDs in conjunction with a smartphone's rear camera. Conversely, SpeCam [66] repurposes a smartphone's built-in sensor, the front camera, alongside the screen as a multi-spectral emitter, facilitating surface sensing with 99.0% accuracy for 30 materials, while SpectroPhone [47] leverages the rear camera and flashlight to attain comparable results. However, regardless of whether it's SpecTrans [45], SpectroPhone [47], or SpeCam [66], the multispectral-based identification approach is primarily suited for classification tasks and struggles to identify patterns on material surfaces. In contrast, RGB-based microscopic image textures offer a more versatile tool, capable of distinguishing between different materials while also capturing finer patterns. Although MagicFinger [63], utilizing microscopic images, has demonstrated impressive test results on 22 textures, its grayscale image input omits certain color information, thus limiting its efficacy for more complex texture classification tasks, such as materials with similar textures but different colors.

Non-vision approaches are also feasible, such as radar [65], vibration absorption [25], sound echo [21], or a fusion of sensors [4, 60]. In the literature, Magic Finger by Yang et al. [63] bears the closest resemblance to our work. However, their system necessitates a micro camera connected to a sizable optical processing unit, and the camera can only capture black and white images, which limits the available information about surfaces.

2.3 Deep Learning and Continual Learning

Deep Neural Networks (DNNs) have been extensively employed in a multitude of multimodal tasks, encompassing data analysis, natural language processing, and image processing [6, 54, 56]. The preeminence of Convolutional Neural Networks (CNNs) in computer vision is attributed to their revolutionary advancements compared to traditional machine learning algorithms. Beyond image classification and recognition, researchers have adapted neural networks for additional computer vision tasks such as semantic segmentation, object detection, and video analysis. Specifically pertinent to our research, neural networks have been utilized to recognize surface textures of clothing [31, 50], distinct areas of the palm [51], and household materials [10–12]. The groundbreaking work of ResNet [22] incorporates residual connections between blocks, enabling the training of exceptionally

deep networks and the enhanced ability to learn abstract representations. Recently, efforts have concentrated on rendering deep neural networks lightweight for deployment on mobile devices with limited computational resources while maintaining comparable performance. For instance, MobileNet [24] employs depthwise separable convolution, achieving similar performance with fewer parameters than conventional CNN models. We have integrated this lightweight network into our system.

Additionally, continual learning, also known as lifelong learning or incremental learning, is an essential aspect of deep learning that focuses on the ability of a model to adapt to new tasks or knowledge while retaining and utilizing prior knowledge effectively [62]. In real-world applications, data is often non-stationary, and new information becomes available over time. Continual learning aims to enable deep learning models to learn from such evolving data streams without suffering from catastrophic forgetting, a phenomenon where the model's performance on previously learned tasks degrades while learning new tasks [23]. Several strategies have been proposed to tackle the catastrophic forgetting problem in deep learning, such as ER (Experience Replay) [44], EWC (Elastic Weight Consolidation) [27], and Synaptic Intelligence [67]. These methods aim to mitigate the interference between old and new knowledge by either reusing stored past experiences, regularizing the model's weights, or selectively updating the model's parameters.

To sum up, our design, delineated in the subsequent section, derives inspiration from existing research on contact surface and material sensing, context-aware computing, and continual learning for deep learning.

3 SYSTEM DESIGN

In photography, the concept of "macro photography" involves the close-up capture of small subjects, such as flowers, rain droplets or insects. The interest in this form of photograph stems from an interest in what one cannot normally see with the naked eye. Mundane objects can appear magical when inspected in detail and the proximity to the surface of objects reveals a world of hidden textures and details which may delight the eye of the viewer. An alternative term, "micro photography", broadly refers to the same concept as "macro photography". While there is no unified scientific definition to distinguish these terms, generally speaking, the degree of magnification is the main basis for judgment. Some literature [14, 55] suggests that the magnification of macro photography is around 20x or lower, while the magnification of micro photography is typically higher. While this is not an absolute it does suggest a useful threshold to distinguish the two terms here.

As a result of this interest, today clip-on macro lens for smartphone are common and cost only a few dollars. Indeed, newer models of smartphone are starting to include built-in macro-lens cameras. Furthermore, such macro camera technology has been advancing and it can now capture surfaces that are so close to the camera, that it is quite literally touching the lens, with a zoom factor up to 30x or even 60x. As a result, such types of macro cameras can broadly be considered as "microscope cameras" (e.g., Oppo Find X3 Pro, Realme GT2 Pro) based on the threshold noted previously. It's worth noting that such "microscope cameras" are still far from the magnification power of optical microscopes, which range from several 100x to the low 1000x. While the magnification factor of 30x to 60x of smartphone is relatively low compared to a lab microscope, it is much larger than an ordinary macro lens, and the sharper texture details and ultra-close shooting distances (only 1-2mm) have made mobile phone microscope cameras interesting. Figure 3 and 4 are example images taken by an actual smartphone with a built-in microscope camera (Oppo Find X3 Pro). Here, we have captured images in minute and fine texture detail of a particular surface. Considering that there are so many details about a surface that are captured by this camera, we see an opportunity in leveraging this "macro lens" or "microscope camera" to achieve new sensing capabilities for mobile devices. In particular, we are interested in surface sensing, to determine the material of the surface, in a mobile context for input and interaction.

In addition, in terms of algorithm implementation, we chose to adopt deep learning for image identification and classification. Microscopic images can reflect extremely detailed textures of objects, some of which are very close,

which requires recognition algorithms with extremely strong resolution and accuracy. With the improvement of computing power, we have witnessed the advancement of deep learning in image classification and recognition in recent years. At the same time, to be more easily deployed on mobile devices with less computing power, some lightweight neural networks have emerged and achieved good performance. Hence, we designed and implemented an approach to recognize the surface, based on these microscopic images, using MobileNet. Also, we incorporate the ER method, a continual learning method, into the foundation of this network to enhance recognition robustness and generalization substantially.

Moreover, our research aims to not only evaluate the technical feasibility and efficiency of surface sensing using the microscope camera but also to improve the user experience of the whole system based on context inference. The design of our system is inspired by how users interact with their smartphone throughout the daily life. For example, we consider the natural placement of a smartphone on a surface while it is not in active use, which occurs many times throughout a day (like when we are in the shower or when the mobile is casually set aside). Consequently, such "placement actions" can be characterized as unobtrusive, non-intrusive, and requiring minimal learning. Specifically, the IMU, a common built-in sensor in mobile phones, is frequently employed to determine the motion state of the devices. We propose utilizing the two-dimensional acceleration data (comprising linear acceleration and angular acceleration) to discern the placement state of mobile phones. Upon detection of this state, the microscope camera is invoked for a single instance of surface capture and processing.

Finally, we further consider various real-life scenarios and the locations where the phone will be placed, such as on a desk, on a bed, on a sofa, on the kitchen counter or even the edge of a sink or pool. After several brainstorming sessions and preliminary experiments, we identified six types of objects (bed, desk/table, sofa, cabinet/shelf/closet, sink/pool/bath, counter) where people often place their mobile phones in daily life. As shown in Figure 2, these objects often correspond to typical life situations. For example, a counter usually signifies a kitchen setting, and a bed typically represents a bedroom, each of which carries implications for contextual awareness. We discovered that the six common objects found in every household are composed of diverse materials; for instance, the desk in one participant's home might be wooden, whereas in another's, it might be made of fiberboard. Consequently, we required each participant to collect data from all six objects, a task that is easily manageable. However, the types of material varied due to the inherent difficulty and impracticality of requiring participants to gather specific materials. Beyond these criteria, no additional collection requirements are imposed. Participants simply placed their mobile phones on various surfaces as they normally would. This approach, which reflects natural user habits, is referred to as the "object-oriented" natural collection method, with the objective of augmenting the generalization of system recognition outcomes.

4 DATASET

To develop a neural network adept at discerning distinct surfaces utilizing these image categories, we opted for the independent collection of sample images due to the scarcity of existing datasets in the literature and open-source resources. Notably, outstanding network performance is intimately connected to the quality of the dataset employed.

4.1 Participants

We recruited a diverse group of 12 participants (6 males, 6 females), ranging in age from 18 to 39 years (mean = 25.23, SD = 5.2), for the dataset collection. Participants are designated P1 to P12 and represented various backgrounds: six participants are academic students from different disciplines, including computer science (2), mechanical engineering (1), mathematics (1), medical and health (1), and art and design (1); the remaining six participants are office workers from various industries, such as law (1), finance (1), accounting (1), UI/UX (2), and
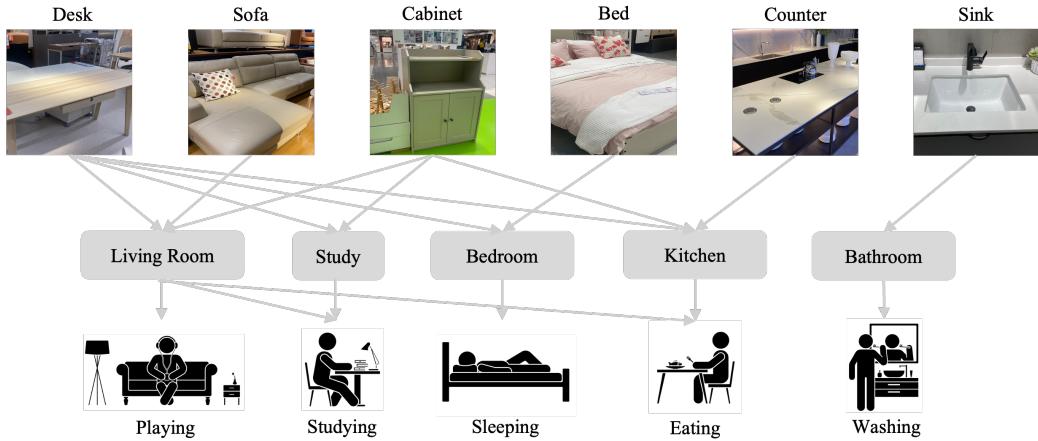
Fig. 2. Six objects that people usually place their phones on and some examples of context-awareness including corresponding scenes and activities.

telecommunications (1). All participants had at least two years of experience using smartphones in their daily lives.

4.2 Task and Procedures

In order to acquire authentic data from real-life situations, we invited the 12 participants to record their daily mobile phone usage. As previously mentioned in section 3, we identified six common object types on which mobile phones are typically placed in everyday life, which are closely related to various living environments such as the living room, bedroom, kitchen, study, and bathroom. This "object-oriented" approach to naturalistic data collection focuses on these six object categories.

To begin with, we presented participants with a comprehensive overview of the experimental procedure and demonstrated the appropriate use of the designated smartphone (Oppo Find X3 Pro). Subsequently, each participant is instructed to utilize the provided phone as they normally would for a period of three days, ensuring that they placed the device on the predetermined objects at least once per day to capture as diverse a range of surfaces as possible. While participants are encouraged to collect surface images at three distinct intervals throughout the day (morning, noon, and evening), this is not a strict requirement. Considering the availability of only one smartphone of the specified model, the device is allocated to each participant (P1 to P12) in a sequential manner, resulting in the whole data collection process spanning approximately one month.

It is crucial to highlight that although minimum placement requirements are set, there is no imposed upper limit; thus, the aforementioned six objects (bed, desk/table, sofa, cabinet/shelf/closet, sink/pool/bath, counter) are likely to be collected multiple times per day for each participant, with varying frequencies of repetition. During the recording process, participants are advised to randomly move and rotate their phones. Apart from these considerations, the entire use process remains unsupervised and uninterrupted, in line with the user's natural habits of using smartphones (such as how often the phone is placed on the surface, the scene where it is placed, etc.). Following data collection, the gathered videos are sampled at a rate of three frames per second to avoid extracting excessively similar images, and some images are discarded due to substantial motion blur (e.g., when the recording involved overly rapid rotation) that resulted in excessive distortion. Ultimately, the accumulated

Fig. 3. Some example images of 6 types of objects.

Fig. 4. Some example images of 9 types of materials (TC:Thread Count).

surface images are classified into 6 object categories (as shown in Figure 3) and their corresponding 9 material properties (as shown in Figure 4). Subsequent sections offer further details.

In short, through our natural, object-oriented data collection approach, the amassed surface images exhibit substantial diversity, encompassing a lengthy time span (over one month), various shooting angles (achieved by rotating the phone), and distinct lighting conditions (corresponding to different times of day). We posit that this diversity contributes to a dataset that more closely aligns with real-world conditions, thereby enhancing the generalization capabilities of the proposed system.

4.3 Data Statistics and Description

Figure 3 and 4 showcase examples of images captured using the microscope camera of the mobile device. It is important to note that the images for the surfaces of the six objects and nine materials are identical, with the

Table 1. The mapping relationship between object types and material types in our dataset.

Object Types	Material Types
1 Bed	1 Plush 2 Fabric (TC>100)
2 Desk/Table	5 Fiberboard/Particleboard 6 Wood/Wood-like Grain
3 Sofa	3 Fabric (TC<100) 4 Leather
4 Cabinet/Shelf/Closet	5 Fiberboard/Particleboard 6 Wood/Wood-like Grain
5 Sink/Pool/Bath	7 Ceramic 8 Stainless Steel
6 Counter	8 Marble/Quartz

distinction being in their respective property labels. Alternatively, it can be understood that the "material" label serves as a supplementary descriptive attribute to the "object" label. For instance, "sofa" is an object label, and in the data we collected, it could be either "fabric" or "leather". Consequently, both terms function as material labels, as they further specify the material composition of the sofa.

Specifically, surfaces categorized by object include: 1) Bed; 2) Desk/Table; 3) Sofa; 4) Cabinet/Shelf/Closet; 5) Sink/Pool/Bath; 6) Counter, while surfaces categorized by material consist of: 1) Plush; 2) Fabric (TC>100); 3) Fabric (TC<100); 4) Leather; 5) Fiberboard/Particleboard; 6) Wood/Wood-like Grain; 7) Ceramic; 8) Stainless Steel, 9) Marble/Quartz. The term "TC" is an abbreviation for Thread Count, which measures the number of threads woven into one square inch of fabric and is often utilized to describe fabric density. In actual tests, we discovered that both "sofa" and "bed" contain the material "fabric". Assuming we classify their "fabric" as the same material, even if our algorithm can accurately identify its material, it will be incredibly challenging to further discern whether it is a "sofa" or a "bed". We envision our material classification to be as precise and unique as possible, meaning that the same material frequently corresponds to only one object, but in reality, the same object may encompass multiple materials. Based on this observation, we employ TC as a metric to differentiate "sofa fabric" and "bed fabric" into two categories. For the purpose of this study, we determined 100 to be a suitable threshold (the physical size range of our square microscopic image in Figure 3 and 4 is approximately 3 mm^2). The density of bed fabrics is typically higher than that of sofa fabrics, which serves as the key factor for the neural network to distinguish between the two categories. Moreover, the mapping relationship between objects and materials is presented in Table 1 in detail.

We amassed a total of 35,284 images in this study. Furthermore, approximately 3,000 images are collected for each individual participant. For the six object types, each type had a minimum of 4,809 images and a maximum of 7,207, with approximately 12-16 surface cases per object type. Regarding the nine material types, each type had at least 2,014 images and no more than 6,850, with the number of corresponding surface cases for each material type ranging from 7 to 13. Additionally, our dataset includes metadata about the surface (type labels) and the folder is structured according to the "person-object-material" hierarchy. For example, "person 2-sofa-leather" refers to the leather sofa collected by the second participant. It should be emphasized that since our data collection is based on an object-oriented approach, each participant can gather data on six types of common objects; however, it is not guaranteed that all nine types of material data can be collected. This is consistent with our common understanding, as not all homes or companies will contain all these common objects due to material diversity. Consequently, participants are not required to deliberately search for specific materials but rather use the objects
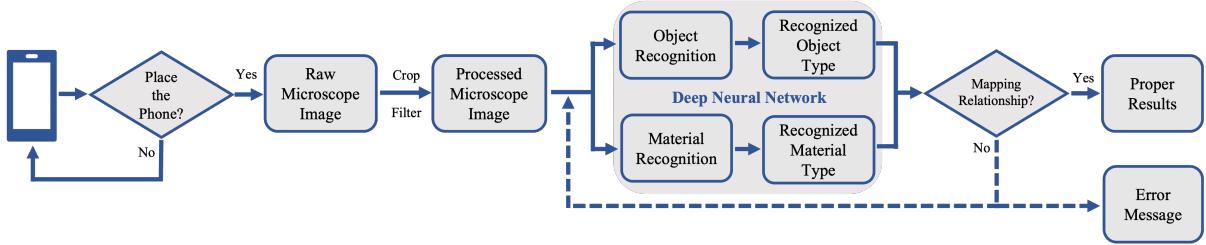
Fig. 5. The schematic representation of the MicroCam system pipeline.

as they typically would. This further underscores our characterization of this data collection method as "natural", requiring no additional learning or effort costs.

To summarize, utilizing this data collection methodology enables us to tackle some of the overfitting problems frequently observed in neural networks. The heterogeneity of the collected data, encompassing diverse participants, lighting conditions, time intervals, and angles, along with the sparse sampling density (3 fps), collectively assists in addressing these issues.

5 SYSTEM IMPLEMENTATION

5.1 Hardware Configuration

We employed the OPPO Find X3 Pro smartphone (256GB storage, 4500mAh and 8GB RAM) and its two built-in sensors in our study. The first, the IMU sensor, is a common component in smartphones that measures linear and angular motion utilizing accelerometers and gyroscopes. It offers data on device orientation, acceleration, and rotation for a variety of applications. In our research, we analyzed two-dimensional linear and angular acceleration data to ascertain whether the phone is situated on a surface. The second sensor, the microscope camera (magnification 30x or 60x), is a distinctive feature of this phone model. It permits users to capture close-up images at up to 30x magnification from extremely close distances. The camera also features a ring light (as shown in Figure 1 (b) upper subfigure) encircling the lens, ensuring consistent brightness for texture capture. Its short focal length (approximately 1mm) enables users to simply raise the phone's back using a standard case, facilitating effortless placement on surfaces for clear microscopic images (as shown in Figure 1 (b) lower subfigure) without suspending the device in mid-air to achieve focus. In the prototype configuration, we employed a transparent phone case with a thickness of 1mm to maintain the requisite sensing distance. In addition, all deep neural network training and testing is done on an Alienware X17 R2 laptop. The laptop's configurations are as follows: 1TB SSD, 32GB Memory, 12th Gen Intel Core i9-12900H CPU, NVIDIA GeForce RTX 3070 Ti Laptop GPU, and Windows 11 Home operating system. The pre-training time of MobileNet and ResNet we built on this laptop is about 3 hours and 8 hours respectively.

5.2 Software Algorithm

5.2.1 System Pipeline. As depicted in the Figure 5, we initially develop a simple Android program to leverage the IMU, allowing the mobile phone to unobtrusively capture surface images in a stationary horizontal state for automatic context awareness. Specifically, we assess the following three aspects: (1) Real-time IMU data monitoring with fixed thresholds for LA (Linear Acceleration) from the accelerometer and AA (Angular Acceleration) from the gyroscope; when both absolute values simultaneously fall below their respective thresholds, the mobile phone is considered stationary horizontal state (denoted as S1); otherwise, it is deemed in another state (denoted as S2); (2) Surface image sampling is activated only when the mobile phone transitions from S2 to S1; (3)

Repeated image sampling activation is avoided while the mobile phone remains stationary, and when the static duration exceeds a certain TT (Time Threshold), the program shifts from foreground to background processing, resuming foreground operation upon reactivation. In our practical tests, the triaxial thresholds for LA and AA are respectively set at $[0.04, 0.04, 0.04]$ (m/s^2) and $[0.02, 0.02, 0.02]$ ($°/s$), while the TT is established at 30 (s). It should be emphasized that the values of these thresholds may vary with different models of mobile phones and different test environments.

Secondly, once surface acquisition is activated, the raw microscopic images collected are processed by cropping, and subsequently filtering out exceedingly blurry images based on the LoG (Laplacian of Gaussian) method [2]. The Laplacian of Gaussian (LoG) method is a technique for edge detection and feature extraction in image processing. It combines Gaussian smoothing to reduce noise and Laplacian edge detection to identify intensity changes, which is widely used for IQA (image quality assessment). By distinguishing true edges from noise artifacts, LoG effectively detects edges in noisy images while preserving important details. In short, this procedure results in the elimination of 31 images. Following this, the processed microscopic images are utilized as inputs for the neural network on a high-end PC during the training and inference phases.

Then, we implement the deep neural network part of the algorithm using the PyTorch framework [35]. Although the highest available resolution is 1920*1080, for the purpose of reducing the amount of calculation and speeding up the processing, the inputs to the network are normalized and resized to 3*224*224. Each input image comprises three channels of RGB, with each channel featuring a 224*224 two-dimensional spatial resolution. During training stage, the training images are augmented with horizontal flip, rotation and random shift. We set the batch size to 16 and use an Adam optimizer with a learning rate of 0.0001 and trained the network for 20 epochs. The trained models are saved and later loaded during real-time user testing. Furthermore, it should be emphasized that we build two identical parallel networks for two different tasks, namely: (1) object recognition (object classification); (2) material recognition (material classification). For object classification, we want to identify the object where the phone is placed on, out of the 6 possible objects, and use softmax function at the last layer for 6 outputs. For material classification, we want to identify the material of the object out of 9 possible classes, and use the same architecture, whereas we employ a softmax function at the last layer for 9 outputs.

Finally, it is worthwhile mentioning that we add a "validation" step after the network outputs the prediction results (as shown in the diamond box on the right in the Figure 5). According to Table 1, if the predicted results satisfy the mapping relationship between objects and materials, they would be output; if not, the recognition fails, and then our program could try the recognition again or terminate the recognition with an error message (in the case of multiple re-recognition failures). Through this way, some obviously wrong predictions can be ruled out.

5.2.2 Continual Learning. We notice that although deep DNNs could achieve satisfactory accuracy on established datasets, they struggle to maintain good performance when faced with continuous input of new data in practical applications. A typical issue is "catastrophic forgetting" for DNNs, where learning a new task (new distribution) leads to forgetting experiences from previous tasks (old distribution), resulting in performance degradation. CL (Continual Learning) is proposed to address this issue, but as it remains an ongoing challenge, we introduced a basic CL method: ER (Experience Replay) [44], to preliminarily enhance MicroCam's performance in practical applications. ER is a method designed to mitigate the issue of catastrophic forgetting by storing and reusing past experiences in memory buffers. The memory buffer, as the core component of the experience replay technique, is responsible for intelligent decision-making and model updating. As depicted in Figure 6, we assume that the network model has been trained well on our established microscopic image dataset at time t=0. At subsequent time steps t=1, 2, ..., new data samples are continually inputted. These samples consist of surface images automatically captured by the mobile phone and labels (ground truth) manually annotated by the user. Subsequently, the memory buffer adjusts the new input data and original data according to a predetermined strategy, such as extracting portions of both original and new data into the buffer and assigning them different training weights
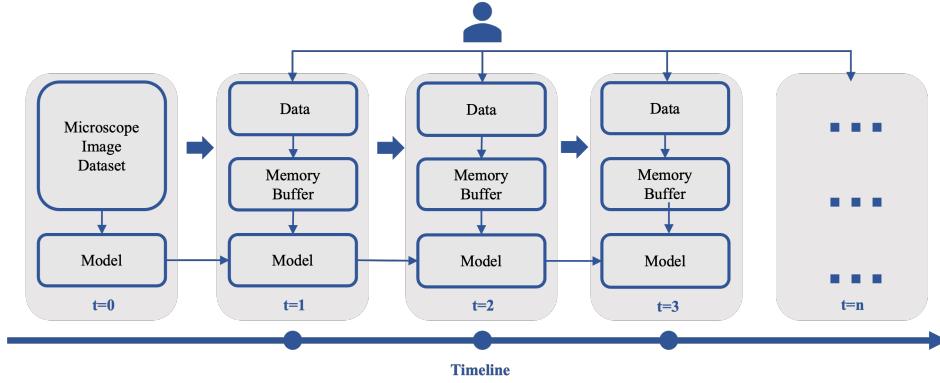
Fig. 6. The continuous learning (ER method) process with time changing for MicroCam system.

as per the strategy. The data in the memory buffer is then utilized to update the model parameters to optimize output performance.

Nonetheless, ER is dependent on meticulous design and fine-tuning to achieve exceptional results. In our experiments, we pre-test the performance of memory buffers with sizes of 120, 240 and 500 during the training process and ultimately selected 500 as the optimal buffer size. Furthermore, we also employ five optimization tricks to enhance experience replay in continual learning, drawing upon existing research [3]. The five tricks are as follows: (1) Independent Buffer Augmentation: This trick applies data augmentation techniques individually to each task-specific buffer instead of applying them jointly to the entire buffer, which increases the diversity and robustness of the replay experience, preventing overfitting and forgetting; (2) Bias Control: A strategy that modifies the neural network classification by adding new output units and adjusting old output units when learning new categories. The adjustment is performed by computing a bias correction term based on the statistics of the old and new categories. This approach balances classification scores between old and new classes, preventing classifier bias towards new classes; (3) Exponential Learning Rate Decay: Employing an exponential schedule with a smaller decay factor enables faster learning of new tasks by decaying the learning rate. This trick could effectively prevent the network from overwriting old knowledge with new knowledge by reducing weight update magnitudes; (4) Balanced Reservoir Sampling: This method samples experiences from the memory buffer in a balanced manner, ensuring each category has an equal probability of being selected, thus preventing class imbalances in memory buffers. However, this strategy may cause certain classes to be underrepresented or overrepresented during replay; (5) Loss-Aware Reservoir Sampling: A trick for extracting experiences from memory buffers using loss values as a measure of difficulty or information content, which is effective for replaying experiences with more challenging or correlated samples, improving learning efficiency.

In summary, through these strategies, the five aspects corresponding to continual learning — policy, frequency, batch, enhancement, and regularization — have all been improved in our MicroCam system. Moreover, we contend that ER is a promising method for mitigating catastrophic forgetting and enhancing forward transfer in continual learning scenarios, but there remains substantial room for improvement.

6 EVALUATION

We conducted a comprehensive evaluation of MicroCam from several perspectives. Firstly, we briefly compare MobileNet and ResNet-50. Although we currently deploy deep neural networks on high-end PCs, we anticipate

future transplantation to mobile devices for independent operation, which will enhance the system's portability and wearability, while balancing performance and computational requirements. Meanwhile, we compare to the results of MagicFinger [63]. Secondly, we performed a more detailed assessment of the lightweight MobileNet, including cross-validation and confusion matrix analysis. Subsequently, we present the performance optimization achieved through continual learning. Finally, we provide test results for systematic power consumption and latency to further demonstrate the system's merits.

6.1 Study 1: Performance Comparison

Firstly, it is noted that some analogous efforts are undertaken in prior research, such as MagicFinger [63], which employs microscopic images for surface classification and identification purposes based on an ultra-compact macro camera strapped to a finger. A significant distinction in our methodology is the utilization of color microscopic images as system input, in contrast to MagicFinger, which relies on grayscale images. Accordingly, we regard the classification of grayscale images mentioned in MagicFinger as a baseline, present the result comparison for analyzing the performance differences between color and grayscale images as input.

Secondly, we conduct a comparative performance analysis between ResNet-50 [22], a well-established deep CNN architecture, and MobileNet-v2 (version 2), a lightweight architecture. We utilize these two networks as feature extractors for both object and material classification tasks. In this expeditious comparison, the dataset is partitioned by randomly selecting 1/10 of the data as the test set, while the remaining data constitutes the training set. We employ top-1 classification accuracy as the evaluation metric.

Table 2. Performance comparison of ResNet-50 and MobileNet-v2 and a comparative analysis employing grayscale (baseline) and RGB images as inputs.

Model	Weights(M)	FLOPs(G)	Input Image Color Mode	Object Top-1 Acc(%)	Material Top-1 Acc(%)
ResNet-50	25.56	4.14	Grayscale	97.12	98.33
			RGB	99.30	99.47
MobileNet-v2	1.7	0.59	Grayscale	96.70	97.58
			RGB	98.23	99.15

The performance comparison results are compiled in Table 2. It is evident that classification outcomes utilizing RGB microscopic images as input surpass those using grayscale images, indicating the color RGB microscopic images contribute positively to object and material classification accuracy. This can be attributed to the richer information contained in RGB, which facilitates the network's ability to identify challenging examples. For instance, some sink and cabinet examples in Figure 3 exhibit similar textures but different colors, making it difficult to distinguish using grayscale images. Additionally, the material classification accuracy exceeds that of object classification, as some object categories encompass multiple materials, complicating the recognition process. For instance, the object category "table/desk", as illustrated in Table 1, corresponds to two distinct material textures, namely "fiberboard/particleboard" and "wood/wood-like grain".

Upon comparing the performance disparities between the two networks under the same color mode input, MobileNet demonstrates advantages in both performance and complexity. For example, with RGB microscopic image input, MobileNet-v2 achieves object and material classification accuracies of 98.23% and 99.15%, respectively, representing a decrease of 1.07% and 0.32% compared to ResNet-50. These results highlight that MobileNet attains comparable accuracy to ResNet-50 while maintaining merely 1/7 of its complexity. In conclusion, we select MobileNet-v2 as our primary prototype network and utilize color RGB microscopic images as input. Building upon these foundations, we proceed to deliver a more in-depth evaluation and a detailed optimization in the following sections.

6.2 Study 2: Cross-Validation

6.2.1 Cross-Validation Results. Following a comparative analysis of network architectures, we opt for MobileNet-v2 and conducted additional evaluations. In this section, we exclusively present the results of MobileNet-v2. The outcomes of the cross-validated tests are illustrated in Table 3. We partition the images into training and test sets using two distinct methods: time-based and person-based splitting.

Table 3. Cross-validation classification performance of MobileNet-v2 (RGB microscopic image inputs).

Classification	Time-based Split Acc(%)	Leave-1-Person-Out Split Acc (%)
Object	98.44	95.56
Material	99.25	96.96

In the time-based split, we divide data by collection time into 10 parts, and then apply 10-fold cross-validation to evaluate. The results of mean accuracy are 98.44% (SD = 0.17) (object) and 99.25% (SD = 0.39) (material). The accuracy of both object and material is very high, and the fluctuation of each test result is also quite small (The value of the two SDs is extremely small). Such a a situation is actually to be expected, because the test set considerably overlaps the training set. The way we solve overfitting for this kind of evaluation is to only take 3 frames per second.

For the person-based split method, we segregate the training and test sets according to each participant. A total of 12 participants are involved, with each individual's data serving as the test set in rotation, while the remaining 11 individuals' data composed the training set. Given that each person constitutes a basic unit, variations in usage habits among individuals result in differing objects and materials upon which the mobile is placed, as well as diverse test durations, lighting conditions, and shooting angles. Consequently, the training and test set data diverge. We argue that this approach mitigates overfitting and data leakage issues. Nonetheless, our results demonstrate a commendable average accuracy of 95.56% (SD = 5.97) (object) and 96.96% (SD = 2.82) (material). Analyzing the outcomes, the leave-1-person-out split method exhibits marginally lower performance than the time-based test, as the training set lacks the test case.

Additionally, we observe that material-based classification tasks are simpler than object-based tasks, yielding 0.81% (time-based split) and 1.40% (leave-1-person-out split) higher performance. This can be attributed to certain objects utilizing similar or identical materials. For instance, tables and cabinets may employ the same wood material, rendering the object classification task more challenging than that of material classification.

6.2.2 Confusion Matrix Analysis. We generate confusion matrices under two evaluation methodologies for six objects (1 bed; 2 desk/table; 3 sofa; 4 cabinet/shelf/closet; 5 sink/pool/bath; 6 counter) and nine materials (1 plush; 2 fabric (TC>100); 3 fabric (TC<100); 4 leather; 5 fiberboard/particleboard; 6 wood/wood-like grain; 7 ceramic; 8 stainless steel; 9 marble/quartz).

Prior to examine the overall confusion matrix, it is valuable to focus on a single individual's confusion matrix to clarify the underlying calculations executed, such as the test results pertaining to the 7th participant, as a representative example. Figure 7 displays the object and material confusion matrices using a "leave-7th-person-out" evaluation. In other words, the data gathered by the 7th person constitutes the test set, while the data obtained by all other participants forms the training set. This "leave-one-person-out" approach is applied to each of the 12 participants in our study.

Each column represents the ground truth, and each row represents the result of model prediction. As we mentioned previously, when collecting data, we require each participant to collect at least 6 kinds of objects once a day. However, it is not guaranteed to cover all 9 kinds of materials, which is also the origin of our collection
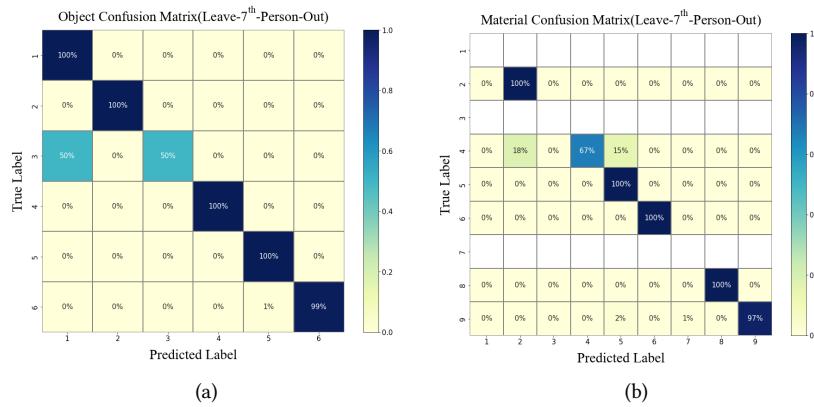
Fig. 7. Object and Material confusion matrix for leave-7th-person-out when using MobileNet-v2.

method, namely "object-orientation". When calculating the confusion matrix, for the leave-one-person-out evaluation method, a material may be missing from the material confusion matrix of a certain participant. For example, in Figure 7 (b), the 1, 3 and 7 types of material are not collected by the 7th participant. Therefore, when calculating the total confusion matrix for the leave-one-person evaluation, if a person lacks a certain material category, that person would not be counted in the calculation of the average for that category.

For analysis, in Figure 7 (a), we can see that the "1 bed" and "3 sofa" classes are often confused, and 50% of the sofas are mistakenly identified as beds, which is due to the similarity of the materials. The collected data is limited, and as a result, the model still does not generalize well enough for these two object labels. For Figure 7 (b), it can be seen that 4 types of leather are misclassified, 18% are incorrectly identified as category 2 (TC>100), and 15% are incorrectly identified as category 5 (wood board). We checked these misclassifications and found that these occurred when the color of the image we tested is too dark or too light. This occurs when the image is blurred, or the surface texture is not obvious, which can result in a misclassification. With our current prototype, users should ensure the device is placed in a stable manner to avoid such misclassification problems. This however, is a matter for future work as discussed in section 8. However, other categories can be well distinguished, indicating that there are still obvious differences in the surface texture between different material categories.

Another example can be seen in Table 1, where both of the objects of class 2 (desk) and class 4 (cabinet) correspond to two materials: class 5 (particleboard/fiberboard) and class 6 (wood/wood-like grain). We originally think that these two classes would be easily misclassified by the neural networks, however this approach continues to perform well. One possible explanation is that for our actual collection, the desks with wood grain predominate, while most of the cabinets/shelves/closets are made of particleboard. As a result, in most cases, they remain straightfoward to distinguish.

For the overall confusion matrix in Figure 8, we note that, under the evaluation of time-based split, both object and material recognition have high accuracy, as one might expect. Instead, let's focus more on the leave-one-person-out evaluation results. As shown in Figure 8 (c), in object recognition, 8.8% of category 1 (bed) is misidentified as category 3 (sofa), and 4.2% of category 3 (sofa) is misidentified as category 1 (bed). As noted previously, this is due to their similar material composition. For material classification in Figure 8 (d), 18.1% of class 1 (plush) are mistaken for class 3 (fabric (TC>100)). We can see from Figure 4 that they are indeed similar and therefore easily confused. We suggest that in practical applications, we can further improve the discrimination
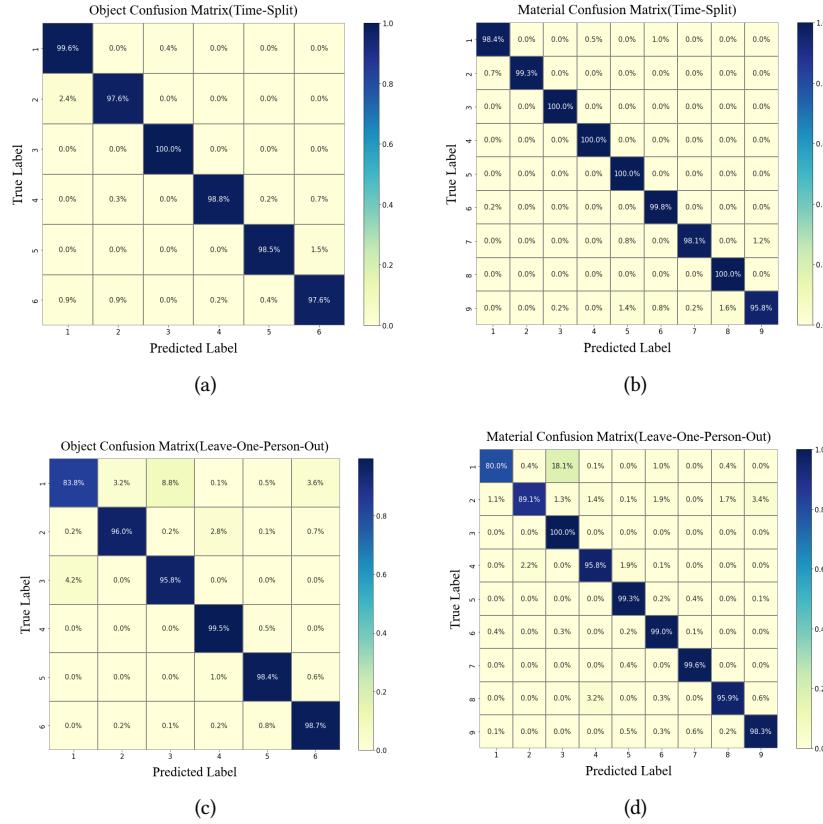
Fig. 8. Object and Material confusion matrix for two types of cross-validation of MobileNet-v2. (a) & (c) Object classification; (b) & (d) Material classification.

of the two classes by increasing the amount of training data for this class. In addition, other categories can be well distinguished, indicating that the surface textures of different object and material categories contain obvious differences. Finally, for the categories that are easily misclassified, we need to pay special attention in the application, and improve the accuracy through the retraining or adaptation of the model to the user data.

In summary, due to the promising results of leave-1-person-out split method, future work should investigate a larger training data to demonstrate the generalization performance of the model.

6.3 Study 3: Continual Learning

As outlined in section 5.2.2, we employ the continual learning ER method to optimize the MobileNet-based algorithm of MicroCam. Figure 9 presents a comparison of MobileNet and MobileNet+ER performance across three test datasets, utilizing the testing methodology described in section 6.1. Specifically, the three test datasets comprise: (1) 3528 images included, obtained by randomly sampling 1/10 of the original dataset; (2) 3000 images included, consisting of difficult instances with low recognition accuracy—some selected from the original dataset and others manually altered with interference processing to increase difficulty. For instance, the four example

Test Dataset	Example Images				MobileNet		MobileNet+ER		ΔScore	
					Object Classification Acc (%)	Material Classification Acc (%)	Object Classification Acc(%)	Material Classification Acc (%)	Object ΔScore (%)	Material ΔScore (%)
1 Original	Sofa-Fabric (TC<100)	Bed-Fabric (TC>100)	Desk-Wood	Counter-Marble	98.23	99.15	98.23	99.15	0	0
2 Original Hard Sample	Sofa-Fabric (TC<100)	Sink-Ceramic	Desk-Wood	Bed-Fabric (TC>100)	78.17	82.33	91.12	94.37	12.95	12.04
3 Wild New Sample	Skin-Skin	Desk-Glass	Jeans-Fabric (TC>100)	White Paper-Paper	22.86	23.01	84.04	85.68	61.18	62.67

Fig. 9. Performance comparison of integrating continual learning (ER method) into MicroCam. Example image captions follow a two-level label naming convention: object type-material type. For instance, "Desk-Wood" denotes a wooden desk.

images in Figure 9 exhibit reduced recognition rates due to lens defocus, negligible texture, artificially lowered image brightness, and increased noise; (3) 1000 images included, constituting newly acquired samples from real-world scenarios. Concerning the labels of these new sample images, two situations arise: part of sub-labels (object type/material type) of the two-level labels can be found in the original dataset, such as "Desk-Glass" and "Jeans-Fabric (TC>100)"; all sub-labels are entirely new and not included in the original dataset, such as "Skin-Skin" and "White Paper-Paper".

From the results depicted in Figure 9, the ER method does not yield improvement in the network performance on the original dataset, as this process has yet to involve the memory buffer update. However, for the second test dataset, the performance gains for object and material classification are 12.95% and 12.04%, respectively. Given that the data in the second test dataset originates from the original dataset without the introduction of new label categories, this outcome demonstrates that the ER method enhances the robustness of MicroCam. Moreover, the evaluation on the third dataset exhibits a substantial boost of 61.18% and 62.27%, respectively. It is attributed to the fact that the supervised learning approach based on the neural network is unable to identify newly added categories; however, upon incorporating ER, the model is capable of continually learning new experiences while retaining the previously acquired knowledge. In this process, we employ various techniques to mitigate catastrophic forgetting and optimize recognition outcomes. The presence of new label categories in this dataset suggests a considerable improvement in the algorithm's generalization. In summary, through the incorporation of the continuous learning ER method and the implementation of corresponding optimization techniques, we substantially augment the algorithmic robustness and generalization capabilities of the MicroCam system.

6.4   Study 4: System Overhead

To gain a more comprehensive understanding of the load situations of our MicroCam system in practical applications, we conducted tests focusing on two aspects: energy consumption and the response latency.

6.4.1   Energy Consumption. Considering that we currently possess only a single mobile phone with an integrated microscope camera, the OPPO Find X3 Pro, we have conducted power consumption tests for the MicroCam system solely on this device. Employing the BatteryHistorian power monitoring tool [7], we measured energy consumption and reported the average from an experiment involving 100 repeated inferences on the smartphone. It is important to note that, as previously mentioned, we presume that placing mobile phones on various surfaces represents typical user behavior in everyday life, which also aligns with the initial design intent of MicroCam. Consequently, here term "every time" running the program refers to the complete process of setting down the phone, capturing a microscopic image, and subsequently processing it. Based on our findings, the OPPO Find X3 Pro can support average 120 microscopic image acquisitions per 1% of battery power consumed by the smartphone for the MicroCam system. This outcome suggests that the power consumption of MicroCam is comparable to that of the standard camera mode on the mobile phone.

6.4.2   Response Latency. Response latency is a critical element influencing user experience in smartphone applications. The delay primarily consists of two components: the duration required for the Android application to capture a microscopic image and the time needed for MobileNet to process the image. The former entails real-time monitoring of the mobile phone's motion status through loop detection of IMU data, followed by microscopic image sampling upon determining the phone's static state. The average duration for this process is approximately 2.1 seconds (calculated from 50 trials using the developed Android application). The latter refers to the inference time of the MobileNet network and the output of results, which averages less than 0.1 seconds (based on 100 trials using MobileNet on an Alienware X17 R2 laptop). The laptop's specifications could be found in section 5.1.

7   FINDINGS AND DISCUSSION

7.1   Discussion 1: Study Methods and Performance

In our dataset, as people moved and rotated the phone along the surface while collecting data, and we only extracted 3 frames per second, the images are different. What's more, each of the 12 participants used the phone for three days (spanning day and night) and their usage habits varied. All of these reduce the overfitting and enhance the generalization of our findings.

The average object and material recognition rates are 98.44~99.25% when using Time-Split and 95.56~96.96% when using leave-one-person-out evaluation. The latter is more realistic because the testing dataset is never seen during training. This is done to evaluate the robustness and generalization of the system on unknown surfaces. Since such a limited number of surfaces are collected, for example, only 5 people provided relevant data for class 1:plush in the material classification task. In the future, with a larger training dataset (e.g., the size of ImageNet), we believe that the results can be improved and generalized to wider, real-life surfaces.

However, collecting a very large dataset of surfaces is non-trivial. We realized that for general users, there can be infinite number of different surface textures.

Thus, to address this issue, we incorporate the concept of continuous learning to bolster the robustness and generalizability of the algorithm. Enhanced robustness guarantees improved performance even in the face of challenging sample recognition, while amplified generalizability ensures the model's capacity for learning and adaptation each time new data is introduced, thereby offering users the opportunity for data and label customization.

(a) Placement-Aware Computing

(b) On-Body Shortcut

(c) Fingerprint Swipe

(d) Customizable Hidden Tag

(e) Ambient Camouflage

Fig. 10. Examples of five potential applications for MicroCam.

We experiment with different neural network architectures, a complex and a lightweight architecture, where the results only differ by a little. This suggests that it is sufficient to use a lightweight neural network architecture (e.g., MobileNet, EfficientNet) to strike a balance between computing speed and accuracy, considering that the system should run in a smartphone itself. Furthermore, we incorporated continuous learning (ER method) to bolster the robustness and generalization of the algorithm, thereby significantly augmenting the practicality of the MicroCam system when encountering a wider array of data in real-world settings.

In addition, an IMU-based sensor fusion approach for horizontal stationary state detection of mobile phones further improves the deployment performance of MicroCam. It ensures a seamless integration of data sampling, image processing, and context awareness, functioning as a prime example of implicit interaction. Consequently, it facilitates a non-intrusive, low learning curve, and personalized user experience, optimized for user satisfaction.

7.2 Discussion 2: Potential Applications and Scenarios

Leveraging MicroCam surface sensing technology, we present a range of applications aimed at improving the overall user experience, as illustrated in Figure 10. To provide a detailed overview of these applications, we have devised a template encompassing the following aspects: application description, benefits, supported scenarios, target users, required computational/setup resources, and interpretation of the sketch shown in Figure 10.

7.2.1 Placement-Aware Computing. As mentioned in section 2.1, a core and pervasive application of context-aware computational MicroCam systems is leveraging surface sensing results for contextually informed decision-making. Specifically, for MicroCam, we envision its ability to discern the surface it is situated on and autonomously adjust the phone's settings accordingly, such as transitioning to silent mode or providing personalized recommendations. The advantages of such applications are evident: seamless adaptation to the user's environment without necessitating explicit actions, enhancing user experience, and minimizing interruptions in diverse situations. These applications can support an extensive array of scenarios, including home and work settings like offices or bedrooms. To implement this application, in addition to the MicroCam system, further algorithms and hardware based on data mining might be required, which could involve recording private information (user habits, browsing history, etc.) for tailored adaptation. In Figure 10 (a), we present two examples: the first detects a phone placed on a desk for an extended period, inferring that the user may be working, and consequently sets the phone to silent mode; the second example recognizes a living room sofa and automatically displays news or game suggestions for entertainment purposes.

7.2.2 *On-Body Shortcut.* The recognition outcomes of the MicroCam system can serve as a trigger for body shortcuts. By touching the phone to various body parts (arms, abdomen, legs, etc.), it can initiate specific commands or launch particular applications [18]. This is akin to scanning an NFC tag with your phone, but without the actual hardware tag. Distinct body parts can represent different actions or meanings. For instance, touching the abdomen might indicate that the user is hungry and wishes to order food, prompting the system to launch the food ordering app. Similarly, touching the pants could signify that the user is on the move and desires to initiate a map navigation application. A comparable application is proposed in RadarCat [65]. However, RadarCat necessitates supplementary hardware on top of standard mobile phones. In contrast, MicroCam utilizes the built-in macro camera of the smartphone. Moreover, considering the wealth of information provided by microscopy, more refined body shortcuts can be realized, such as estimating skin moisture content from microscopic images and further inferring human body conditions. Consequently, this application is well-suited for individuals seeking swift and convenient support from mobile devices. We suggest that, based on these advantages, the application can be employed more frequently in mobile scenarios and support auxiliary functions. During development, additional body recognition algorithms (e.g., gesture or posture recognition) and corresponding applications can be incorporated to extend the application's functionality. In the example depicted in Figure 10 (b), for preliminary demonstration purposes, we have simply set different response modes for the skin, upper body (hoodie) and lower body (jeans): vibration and two rings with distinct rhythms. The demonstration effect can be observed in the supporting demo video. More diverse and varied response methods can be added in subsequent development stages.

7.2.3 *Fingerprint Swipe.* Fingerprint swipe is proposed to facilitate microscope-based finger sensing. We envision that, under this application, the information obtained by the microscope can be utilized in two aspects: firstly, the detection of fingerprint texture, where more detailed fingerprint information can be employed to enhance the security of the device's authentication; secondly, detecting minute finger movements as a form of interactive input. Due to the microscope's high magnification, it is highly sensitive to extremely small movements, which can be harnessed for subtle and discreet interactions [38] as well as one-handed use. The aforementioned features are well-suited for privacy requirements in public places, enhancing the system's usability, privacy, and security. For instance, users can unobtrusively interact with their mobile phones in public spaces to initiate customized phone operations. Some complementary technologies for discreet and small interactions are considered for use with this application. In the example provided in Figure 10 (c), we implemented one of the application's functions, which is to determine the direction of finger movement based on the background subtraction method [37], subsequently triggering the downward slide of the front menu bar.

7.2.4 *Customizable Hidden Tag.* MicroCam can decipher custom tags embedded in everyday surfaces that are too minuscule to be detected by the naked eye, subsequently triggering specific actions or launching applications. For instance, micro-watermarks can also be concealed within various images and even texts, utilizing our encryption-oriented technology. Another example includes hiding images within human faces, thereby achieving nested images or mosaic photos. This method is characterized by its high concealment and security, enabling the concealment of interaction with physical objects and facilitating subtle or implicit interactions. The potential application scenarios for this method are vast, encompassing security, advertising, and property rights protection, among others. For example, merchants or enterprises can employ this application to embed "subtle" advertisements within products without disturbing consumers. To support these functionalities, specific recognition algorithms are required according to different using purposes. In Figure 10 (d), we provide a sketch of an implementation where we generated and printed an icon (a computer icon on a piece of paper, composed of numerous imperceptible QR codes. Aesthetically, it appears not much different from standard printed content. Touching the phone to the printed icon activates certain actions, such as turning on a PC. Ultimately, different QR codes, barcodes, and the

like can be generated for various purposes. In the provided sketch demo, we employed a QR code for simplicity, which could be replaced with custom encoding, such as Anoto's dot pattern.

7.2.5 Enhanced Ambient Camouflage. Blending smartphones with their surroundings as if they are imperceptible and seamlessly integrated into the background may offer enhanced aesthetics and social discretion. For instance, during an intimate dinner, a phone on the table could camouflage itself [36] and blend into the background, thereby improving the user experience and promoting social acceptance. This level of realism appears increasingly feasible as edge-to-edge phones with minimal bezels and under-display cameras become more commonplace. Detailed images captured by microscope cameras can provide a more sophisticated foundation for imitation in this application, allowing for the incorporation of richer texture details to augment the verisimilitude of the camouflage images. This application can be employed in social situations, meetings, and scenarios that necessitate discretion. In addition to the surface image captured by the MicroCam itself, some supplementary techniques such as edge alignment and image rendering are required to integrate local detail textures into the overall image. Moreover, to achieve superior real-time rendering effects, additional hardware, such as a high-performance cloud server, may be necessary. In our example in Figure 10 (e), for the purpose of sketch demonstration, the image captured by a conventional camera is displayed on the phone and manually adjusted (e.g., zooming in/out, alignment) to achieve the optimal camouflage angle. Subsequently, we employed manual photo-editing techniques to integrate the texture patterns of the microscopic images into photos of wooden materials, captured via conventional cameras. This procedure enhanced the texture details of the camouflage images, resulting in an augmented level of realism.

7.3 Discussion 3: User Experience Analysis

A plethora of intriguing and innovative sensing technologies have been consistently investigated for material and surface detection. However, limited research has concentrated on interaction factors beyond the sensor itself. In response, we strive to bridge these gaps by conducting a qualitative analysis of the positive and negative impacts of specific technical parameters on user experience. This approach will elucidate the differences between our contribution and prior work. Given the similarities in research context and technical direction, we have selected MagicFinger [63], SpectroPhone [47], and SpeCam [66] for comparative analysis.

As demonstrated in Table 4, both MicroCam and MagicFinger perform material classification directly based on microscopic images. The key distinction lies in MicroCam's processing of RGB images, while MagicFinger utilizes grayscale images. The advantages of employing RGB images are evident; for instance, in section 6.1, we assessed that classification results derived from RGB microscopic images surpass those obtained from grayscale images. This is attributable to the fact that color information is instrumental in distinguishing material textures with similar structures, which are lost in grayscale images. A compelling example involves the samples from objects cabinets (fiberboard material) and sinks (ceramic material), which are texturally similar and indistinguishable, yet their colors differ significantly. Moreover, we observed that our MicroCam possesses a higher resolution than MagicFinger, which proves beneficial for more precise context perception and broader application scenarios.

Moreover, both MicroCam and SpectroPhone utilize the rear camera instead of the front one. SpeCam requires the user to place the mobile phone face down, occupying the front display to showcase different lights, rendering the phone unusable during the sensing period. This approach is evidently less user-friendly and more challenging to use. Additionally, the focal length, or sensing distance, of MicroCam is approximately 1mm. This necessitates more support to maintain the requisite distance between the mobile phone and the surface, compared to other methodologies. However, MicroCam enables a more user-friendly solution by necessitating a less cumbersome setup. For instance, while SpectroPhone and SpeCam both mandate a custom but heavy mobile phone case, we merely requires a commonplace and lightweight 1mm commercial mobile phone case or the simple use of a coin for elevation. This design choice notably enhances the user experience. Ultimately, compared to other

Table 4. Technical parameter comparison between 3 surface and material sensing methods (MicroCam, MagicFinger [63], SpectroPhone [47] and SpeCam [66]). Note: numerical specifications outside resolution brackets represent cropped resolution, while figures within brackets indicate maximum resolution. For instance, 175*175 (248*248) means the maximum image resolution is 248*248 pixels and the cropped image resolution for algorithm processing is 175*175 pixels.

Technical Parameters	MicroCam	MagicFinger	SpectroPhone	SpeCam
Sensing Principle	RGB microscopic image	grayscale microscopic image	multispectral feature	multispectral feature
Resolution (pixels)	224*224 (1920*1080)	175*175 (248*248)	640*480	3986*2976
Sensing Distance(mm)	1	<5	3	3
Front/Rear Camera	Rear	Tied to One Finger	Rear	Front
Internal/External Sensor	all internal	all external	external LEDs+interal camera	all internal

technologies, our MicroCam offers more detailed and fine-grained microscopic images, providing a broader range of context-aware information. For instance, multispectral imaging struggles to detect subtle differences within the same material, such as variations in surface texture. In contrast, our system can intuitively and easily discern these differences through microscopic images, as demonstrated by textiles with varying knitting densities in Figure 4. Additionally, for the potential application example provided in section 7.2.4, the multispectral-based method cannot identify specific surface textures such as ultra-small QR codes, while MicroCam's microscopic image can capture and display them intuitively.

Last but not least, in the final row of Table 4, both MagicFinger and SpectroPhone utilize external, custom-built hardware to support their approaches. For instance, SpectroPhone requires external LEDs to provide a more personalized light source. In contrast, MicroCam and SpeCam depend solely on off-the-shelf components found in commercial cell phones. While external hardware-based approaches have demonstrated feasibility, we focus more on the "user experience in the present". The development, commercialization, and public acceptance of a technology require time and resources. Our hardware solution has been applied to some commercial mobile phones, such as the Oppo Find X3 Pro and Realme GT2 Pro. With these devices, our method can be directly deployed on existing mobile phones without incurring additional hardware costs. For other mobile phones, users can also easily purchase various commercial microscope lens clips. Conversely, SpectroPhone's highly customized Bluetooth transmitter and receiver modules and LED lighting modules are currently not readily available on the market for users. Consequently, we contend that MicroCam offers distinct advantages in terms of practicality and convenience for user experience.

7.4 Discussion 4: Privacy Concern

We realize that camera-based sensing approaches, such as MicroCam, may raise potential privacy concerns for users. On one hand, compared to other camera sensing methods that capture larger scenes, MicroCam focuses primarily on surface detail information rather than the overall scene. This makes it difficult to infer macro information, such as an individual's identity or building structure. From this perspective, the privacy risk associated with microscope camera-based sensing is lower than that of conventional camera-based sensing methods, offering certain advantages. On the other hand, previous work has demonstrated that integrating cameras into various devices can mitigate some privacy and security issues [33]. For instance, some device manufacturers provide physical kill switches for laptop cameras. Similarly, in terms of future enhancements, we anticipate the incorporation of a "switching" mechanism to regulate the camera's functionality within the MicroCam system. This could potentially manifest as a physical switch integrated directly into the mobile device or a virtual switch implemented through software algorithms. For the latter, it may necessitate the amalgamation of and support from authentication algorithms to ensure effective on/off control. In addition, another possibility involves employing a low-resolution sensor. As demonstrated in the section 6.3, the incorporation of continuous

learning can enhance algorithm robustness and achieve better recognition results for challenging samples. Consequently, a lower-resolution sensor may increase the difficulty of reconstructing user information, thereby providing a degree of privacy protection. In conclusion, while some current approaches address the inherent privacy concerns of camera sensing methods, this remains an ongoing challenge that requires further research and development to ensure user privacy is adequately protected.

8 LIMITATIONS AND FUTURE WORK

In our dataset, we included only 6 common objects and 9 typical materials encountered in our everyday life. However, the diversity of objects and materials in the wild can be far more extensive. Therefore, in future work, we will consider more complicated situations. Additionally, we also plan to invite more users to further increase the dimension of the data, which is absolutely beneficial for future user-personalized context adaptation.

In the implementation of certain components within the system, we employ basic methods or algorithms. For instance, to detect the horizontal static state of a mobile phone, we simply compare the two data points obtained from the Inertial Measurement Unit (IMU), which are linear acceleration and angular acceleration, with a fixed threshold to determine the state. While this rudimentary approach proves to be highly effective, it is not without its limitations. The horizontal static state may trigger image sampling in some cases, which might not align with user expectations. To address this issue, we plan to consider the integration of additional sensors in future developments to enhance the system's performance in this regard. Additionally, we implemented our classifier using a high-end PC. Yet, it is possible to run the classifier on the phone itself, in real-time, since we are using a lightweight neural network architecture (i.e. MobileNet). Further, although we added a verification phase to check the "object and material mapping relationship" at the end of the system pipeline (as shown in Figure 5), it still cannot exclude some special error cases. For example, as mentioned earlier, we noticed that from Table 1, the object "desk" and "cabinet" contain two materials, "particleboard" and "wood/wood-like grain". Imagine for a moment, a wooden cabinet, where the object recognition result output by the model is incorrect ("table"), and the material recognition result is correct ("wood"), which will not be detected by our model. Therefore, in the future, we will try to incorporate data from multiple sensors (such as radar, etc.) into the system to adapt our prototype to more complex situations. Finally, the ER (Experience Replay) method utilized in the system optimization component represents a fundamental approach within the realm of continual learning. Our implementation of ER has indeed enhanced the system's robustness and generalization capabilities to a certain degree; however, there remains substantial potential for improvement. Continual learning is a dynamic and highly challenging field, offering numerous opportunities for further refinement and advancements in future research endeavors.

While macro cameras are becoming more common in smartphones today, not all devices feature a macro (microscope) camera. Therefore, we cannot claim that our sensing method will work on *all* smartphones. In addition, we require a phone case that raises the gap between the camera and the surface by approximately 1mm to achieve optimal focus. However, with the addition of an external microscope lens (such as a clip-on) and a suitable phone case designed to fit over the standard phone camera, we suggest that MicroCam may perform well.

At present, we have only explored a limited number of applications. In future work, we aim to explore additional applications, such as continuous movement tracking of the mobile phone on a surface with unique textures (e.g., a wooden desk), akin to using a computer mouse. By integrating more information, such as additional sensor data or activity tracking within the software, we can further improve the intelligence and sophistication of MicroCam's context inference capabilities.

9 CONCLUSION

We have introduced a method for contact-based surface sensing using the built-in macro (microscope) camera of a smartphone. In our approach, IMU-based detection of a horizontal static state identifies users' actions when they

naturally place the phone on any surface, anyway. Then, automatic surface recognition occurs via MobileNet-based microscopic image classification, further triggering corresponding actions such as placement-aware mode switching or body shortcuts. We collected a substantial dataset, which supported us in training a robust neural network model capable of recognizing different surfaces with a high degree of accuracy. Moreover, we applied continual learning to optimize the robustness and generalization of the algorithm. In conclusion, we anticipate that this sensing technique could, in due course, be available on any smartphone. The only requirement is a macro-lens feature, which is already emerging in the consumer market.

REFERENCES

[1] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. 2004. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications* 37, 4 (2004), 445–456.

[2] Raghav Bansal, Gaurav Raj, and Tanupriya Choudhury. 2016. Blur image detection using Laplacian operator and Open-CV. In *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*. 63–67. https://doi.org/10.1109/SYSMART.2016.7894491

[3] P. Buzzega, M. Boschini, A. Porrello, and S. Calderara. 2021. Rethinking Experience Replay: a Bag of Tricks for Continual Learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 2180–2187. https://doi.org/10.1109/ICPR48806.2021.9412614

[4] Rajkumar Darbar and Debasis Samanta. 2015. SurfaceSense: Smartphone Can Recognize Where It Is Kept. In *Proceedings of the 7th International Conference on HCI, IndiaHCI 2015* (Guwahati, India) *(IndiaHCI'15)*. Association for Computing Machinery, New York, NY, USA, 39–46. https://doi.org/10.1145/2835966.2835971

[5] Antonella De Angeli, Alistair Sutcliffe, and Jan Hartmann. 2006. Interaction, Usability and Aesthetics: What Influences Users' Preferences?. In *Proceedings of the 6th Conference on Designing Interactive Systems* (University Park, PA, USA) *(DIS '06)*. Association for Computing Machinery, New York, NY, USA, 271–280. https://doi.org/10.1145/1142405.1142446

[6] Shohreh Deldari, Hao Xue, Aaqib Saeed, Daniel V. Smith, and Flora D. Salim. 2022. COCOA: Cross Modality Contrastive Learning for Sensor Data. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3, Article 108 (sep 2022), 28 pages. https://doi.org/10.1145/3550316

[7] Android Developers. 2021. BatteryHistorian. https://developer.android.com/topic/performance/power/setup-battery-historian Accessed: 2021-07-17.

[8] Anind K Dey. 2001. Understanding and using context. *Personal and ubiquitous computing* 5, 1 (2001), 4–7.

[9] Andrew Dillon. 1987. A PSYCHOLOGICAL VIEW OF "USER-FRIENDLINESS". In *Human–Computer Interaction–INTERACT '87*, H.-J. BULLINGER and B. SHACKEL (Eds.). North-Holland, Amsterdam, 157–163. https://doi.org/10.1016/B978-0-444-70304-0.50034-0

[10] Zackory Erickson, Sonia Chernova, and Charles C. Kemp. 2017. Semi-Supervised Haptic Material Recognition for Robots using Generative Adversarial Networks. In *Proceedings of the 1st Annual Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 78)*, Sergey Levine, Vincent Vanhoucke, and Ken Goldberg (Eds.). PMLR, 157–166. https://proceedings.mlr.press/v78/erickson17a.html

[11] Zackory Erickson, Nathan Luskey, Sonia Chernova, and Charles C. Kemp. 2019. Classification of Household Materials via Spectroscopy. *IEEE Robotics and Automation Letters* 4, 2 (April 2019), 700–707. https://doi.org/10.1109/LRA.2019.2892593

[12] Zackory Erickson, Eliot Xing, Bharat Srirangam, Sonia Chernova, and Charles C. Kemp. 2020. Multimodal Material Classification for Robots using Spectroscopy and High Resolution Texture Imaging. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 10452–10459. https://doi.org/10.1109/IROS45743.2020.9341165

[13] Euan Freeman, Gareth Griffiths, and Stephen A. Brewster. 2017. Rhythmic Micro-Gestures: Discreet Interaction on-the-Go. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (Glasgow, UK) *(ICMI '17)*. Association for Computing Machinery, New York, NY, USA, 115–119. https://doi.org/10.1145/3136755.3136815

[14] Florian Fuchs, Andreas Koenig, David Poppitz, and Sebastian Hahnel. 2020. Application of macro photography in dental materials science. *Journal of Dentistry* 102 (2020), 103495.

[15] Kaori Fujinami, Satoshi Kouchi, and Yuan Xue. 2012. Design and Implementation of an On-body Placement-aware Smartphone. In *2012 32nd International Conference on Distributed Computing Systems Workshops*. IEEE, 69–74.

[16] Susan Gasson. 2003. Human-centered vs. user-centered approaches to information system design. *Journal of Information Technology Theory and Application (JITTA)* 5, 2 (2003), 5.

[17] Hans W Gellersen, Albrecht Schmidt, and Michael Beigl. 2002. Multi-sensor context-awareness in mobile devices and smart artifacts. *Mobile Networks and Applications* 7, 5 (2002), 341–351.

[18] Tiago Guerreiro, Ricardo Gamboa, and Joaquim Jorge. 2009. *Mnemonical Body Shortcuts for Interacting with Mobile Devices*. Springer-Verlag, Berlin, Heidelberg, 261–271. https://doi.org/10.1007/978-3-540-92865-2_29

[19] Xiansheng Guo, Shilin Zhu, Lin Li, Fangzi Hu, and Nirwan Ansari. 2019. Accurate WiFi Localization by Unsupervised Fusion of Extended Candidate Location Set. *IEEE Internet of Things Journal* 6, 2 (2019), 2476–2485. https://doi.org/10.1109/JIOT.2018.2870659

[20] Chris Harrison and Scott E. Hudson. 2008. Lightweight Material Detection for Placement-Aware Mobile Computing. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. Association for Computing Machinery, New York, NY, USA, 279–282. https://doi.org/10.1145/1449715.1449761

[21] Tatsuhito Hasegawa, Satoshi Hirahashi, and Makoto Koshino. 2016. Determining a Smartphone's Placement by Material Detection Using Harmonics Produced in Sound Echoes. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services* (Hiroshima, Japan) *(MOBIQUITOUS 2016)*. Association for Computing Machinery, New York, NY, USA, 246–253. https://doi.org/10.1145/2994374.2994389

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778. https://doi.org/10.1109/CVPR.2016.90

[23] Shruthi K. Hiremath, Yasutaka Nishimura, Sonia Chernova, and Thomas Plötz. 2022. Bootstrapping Human Activity Recognition Systems for Smart Homes from Scratch. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3, Article 119 (sep 2022), 27 pages. https://doi.org/10.1145/3550294

[24] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).

[25] Sungjae Hwang and Kwangyun Wohn. 2013. VibroTactor: Low-Cost Placement-Aware Technique Using Vibration Echoes on Mobile Devices. In *Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Companion* (Santa Monica, California, USA) *(IUI '13 Companion)*. Association for Computing Machinery, New York, NY, USA, 73–74. https://doi.org/10.1145/2451176.2451206

[26] Wendy Ju. 2015. The design of implicit interactions. *Synthesis Lectures on Human-Centered Informatics* 8, 2 (2015), 1–93.

[27] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.

[28] Sunmin Lee, Jinah Kim, and Nammee Moon. 2019. Random forest and WiFi fingerprint-based indoor location recognition system using smart watch. *Human-centric Computing and Information Sciences* 9, 1 (2019), 1–14.

[29] Hang Li, Xi Chen, Ju Wang, Di Wu, and Xue Liu. 2022. DAFI: WiFi-Based Device-Free Indoor Localization via Domain Adaptation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 167 (dec 2022), 21 pages. https://doi.org/10.1145/3494954

[30] Nicolai Marquardt, Ken Hinckley, and Saul Greenberg. 2012. Cross-Device Interaction via Micro-Mobility and f-Formations. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 13–22. https://doi.org/10.1145/2380116.2380121

[31] Alexander J. Medeiros, Lee Stearns, Leah Findlater, Chuan Chen, and Jon E. Froehlich. 2017. Recognizing Clothing Colors and Visual Textures Using a Finger-Mounted Camera: An Initial Investigation. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility* (Baltimore, Maryland, USA) *(ASSETS '17)*. Association for Computing Machinery, New York, NY, USA, 393–394. https://doi.org/10.1145/3132525.3134805

[32] Florian Floyd Mueller, Pedro Lopes, Paul Strohmeier, Wendy Ju, Caitlyn Seim, Martin Weigel, Suranga Nanayakkara, Marianna Obrist, Zhuying Li, Joseph Delfa, Jun Nishida, Elizabeth M. Gerber, Dag Svanaes, Jonathan Grudin, Stefan Greuter, Kai Kunze, Thomas Erickson, Steven Greenspan, Masahiko Inami, Joe Marshall, Harald Reiterer, Katrin Wolf, Jochen Meyer, Thecla Schiphorst, Dakuo Wang, and Pattie Maes. 2020. Next Steps for Human-Computer Integration. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3313831.3376242

[33] José Ramón Padilla-López, Alexandros Andre Chaaraoui, and Francisco Flórez-Revuelta. 2015. Visual privacy protection methods: A survey. *Expert Systems with Applications* 42, 9 (2015), 4177–4195.

[34] Brice Parilusyan, Marc Teyssier, Valentin Martinez-Missir, Clément Duhart, and Marcos Serrano. 2022. Sensurfaces: A Novel Approach for Embedded Touch Sensing on Everyday Surfaces. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 2, Article 67 (jul 2022), 19 pages. https://doi.org/10.1145/3534616

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 8024–8035. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[36] Jennifer Pearson, Simon Robinson, Matt Jones, Anirudha Joshi, Shashank Ahire, Deepak Sahoo, and Sriram Subramanian. 2017. Chameleon Devices: Investigating More Secure and Discreet Mobile Interactions via Active Camouflaging. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 5184–5196. https://doi.org/10.1145/3025453.3025482

[37] Massimo Piccardi. 2004. Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, Vol. 4. IEEE, 3099–3104.

[38] Henning Pohl, Andreea Muresan, and Kasper Hornbæk. 2019. *Charting Subtle Interaction in the HCI Literature.* Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3290605.3300648

[39] Hongmei Qian, Meng Xu, Xiaowei Li, Muwei Ji, Lei Cheng, Anwer Shoaib, Jiajia Liu, Lan Jiang, Hesun Zhu, and Jiatao Zhang. 2016. Surface micro/nanostructure evolution of Au–Ag alloy nanoplates: Synthesis, simulation, plasmonic photothermal and surface-enhanced Raman scattering applications. *Nano Research* 9, 3 (2016), 876–885.

[40] Aaron Quigley. 2010. From GUI to UUI: Interfaces for ubiquitous computing. *Ubiquitous Computing Fundamentals* (2010), 237–283.

[41] A. Quigley, B. Ward, C. Ottrey, D. Cutting, and R. Kummerfeld. 2004. BlueStar, a privacy centric location aware system. In *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556).* 684–689. https://doi.org/10.1109/PLANS.2004.1309060

[42] Aaron Quigley and David West. 2005. Proximation: Location-awareness though sensed proximity and gsm estimation. In *International Symposium on Location-and Context-Awareness.* Springer, 363–376.

[43] Vaskar Raychoudhury, Jiannong Cao, Mohan Kumar, and Daqiang Zhang. 2013. Middleware for pervasive computing: A survey. *Pervasive and Mobile Computing* 9, 2 (2013), 177–200. https://doi.org/10.1016/j.pmcj.2012.08.006 Special Section: Mobile Interactions with the Real World.

[44] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. *Advances in Neural Information Processing Systems* 32 (2019).

[45] Munehiko Sato, Shigeo Yoshida, Alex Olwal, Boxin Shi, Atsushi Hiyama, Tomohiro Tanikawa, Michitaka Hirose, and Ramesh Raskar. 2015. SpecTrans: Versatile Material Classification for Interaction with Textureless, Specular and Transparent Surfaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15).* Association for Computing Machinery, New York, NY, USA, 2191–2200. https://doi.org/10.1145/2702123.2702169

[46] Albrecht Schmidt. 2000. Implicit human computer interaction through context. *Personal technologies* 4, 2 (2000), 191–199.

[47] Maximilian Schrapel, Philipp Etgeton, and Michael Rohs. 2021. *SpectroPhone: Enabling Material Surface Sensing with Rear Camera and Flashlight LEDs.* Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3411763.3451753

[48] Barış Serim and Giulio Jacucci. 2019. Explicating "Implicit Interaction": An Examination of the Concept and Challenges for Research. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19).* Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3290605.3300647

[49] Dai Shi, Dan Tao, Jiangtao Wang, Muyan Yao, Zhibo Wang, Houjin Chen, and Sumi Helal. 2021. Fine-Grained and Context-Aware Behavioral Biometrics for Pattern Lock on Smartphones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 1, Article 33 (mar 2021), 30 pages. https://doi.org/10.1145/3448080

[50] Lee Stearns, Leah Findlater, and Jon E. Froehlich. 2018. Applying Transfer Learning to Recognize Clothing Patterns Using a Finger-Mounted Camera. In *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility* (Galway, Ireland) *(ASSETS '18).* Association for Computing Machinery, New York, NY, USA, 349–351. https://doi.org/10.1145/3234695.3241015

[51] Lee Stearns, Uran Oh, Leah Findlater, and Jon E. Froehlich. 2018. TouchCam: Realtime Recognition of Location-Specific On-Body Gestures to Support Users with Visual Impairments. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 164 (jan 2018), 23 pages. https://doi.org/10.1145/3161416

[52] Constantine Stephanidis, Gavriel Salvendy, Margherita Antona, Jessie YC Chen, Jianming Dong, Vincent G Duffy, Xiaowen Fang, Cali Fidopiastis, Gino Fragomeni, Limin Paul Fu, et al. 2019. Seven HCI grand challenges. *International Journal of Human–Computer Interaction* 35, 14 (2019), 1229–1269.

[53] Hossein Taheri and Ahmed Arabi Hassen. 2019. Nondestructive ultrasonic inspection of composite materials: A comparative advantage of phased array ultrasonic. *Applied Sciences* 9, 8 (2019), 1628.

[54] Sasha Targ, Diogo Almeida, and Kevin Lyman. 2016. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029* (2016).

[55] Format Team. 2020. The Beginners Guide to Macro Photography. https://www.format.com/magazine/resources/photography/macro-photography-beginners-guide

[56] Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. *arXiv preprint arXiv:1905.05950* (2019).

[57] Manfred Thüring and Sascha Mahlke. 2007. Usability, aesthetics and emotions in human–technology interaction. *International journal of psychology* 42, 4 (2007), 253–264.

[58] Garreth W. Tigwell and Michael Crabb. 2020. *Household Surface Interactions: Understanding User Input Preferences and Perceived Home Experiences.* Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376856

[59] Lesley Trenner. 1987. How to win friends and influence people: definitions of user-friendliness in interactive computer systems. *Journal of information science* 13, 2 (1987), 99–107.

[60] Jason Wiese, T. Scott Saponas, and A.J. Bernheim Brush. 2013. Phoneprioception: Enabling Mobile Phones to Infer Where They Are Kept. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13).* Association for Computing Machinery, New York, NY, USA, 2157–2166. https://doi.org/10.1145/2470654.2481296

[61] Fuyong Xing, Yuanpu Xie, Hai Su, Fujun Liu, and Lin Yang. 2017. Deep learning in microscopy image analysis: A survey. *IEEE transactions on neural networks and learning systems* 29, 10 (2017), 4550–4568.

[62] Susu Xu, Shijia Pan, and Tong Yu. 2020. CML-IOT 2020: The Second Workshop on Continual and Multimodal Learning for Internet of Things. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers* (Virtual Event, Mexico) *(UbiComp-ISWC '20)*. Association for Computing Machinery, New York, NY, USA, 616–618. https://doi.org/10.1145/3410530.3414613

[63] Xing-Dong Yang, Tovi Grossman, Daniel Wigdor, and George Fitzmaurice. 2012. Magic finger: always-available input through finger instrumentation. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*. 147–156.

[64] Jiung yao Huang and Chung-Hsien Tsai. 2008. Improve GPS positioning accuracy with context awareness. In *2008 First IEEE International Conference on Ubi-Media Computing*. 94–99. https://doi.org/10.1109/UMEDIA.2008.4570872

[65] Hui-Shyong Yeo, Gergely Flamich, Patrick Schrempf, David Harris-Birtill, and Aaron Quigley. 2016. RadarCat: Radar Categorization for Input & Interaction. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 833–841. https://doi.org/10.1145/2984511.2984515

[66] Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, David Harris-Birtill, and Aaron Quigley. 2017. SpeCam: Sensing Surface Color and Material with the Front-Facing Camera of a Mobile Device. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Vienna, Austria) *(MobileHCI '17)*. Association for Computing Machinery, New York, NY, USA, Article 25, 9 pages. https://doi.org/10.1145/3098279.3098541

[67] Friedemann Zenke, Ben Poole, and Surya Ganguli. 2017. Continual learning through synaptic intelligence. In *International conference on machine learning*. PMLR, 3987–3995.