RoFin: 3D Hand Pose Reconstructing via 2D Rolling Fingertips

Xiao Zhang, Griffin Klevering, Juexing Wang, Li Xiao, Tianxing Li
Michigan State University
{zhan1387,kleveri2,wangjuex}@msu.edu,lxiao@cse.msu.edu,litianx2@msu.edu

ABSTRACT

Smart homes, medical devices, and education systems, among other emerging cyber-physical systems, hold immense promise for sensing-based user interfaces, especially for using fingers and hand gestures as system input. However, vision approaches compatible with time-consuming image processing adopt low 60 Hz location sampling rate (frame rate) for real-time hand gesture recognition. Furthermore, they are not suitable for low-light environment and long detection range. In this paper, we propose RoFin, which first exploits 6 temporal-spatial 2D rolling fingertips for real-time 3D reconstructing of 20-joint hand pose. RoFin designs active optical labeling for finger identification and enhances inside-frame 3D location tracking via high rolling shutter rate (5-8 KHz). These features enable great potentials for enhanced multi-user HCI, virtual writing for Parkinson suffers, etc. We implement RoFin prototypes with wearable gloves attached with low-power single-colored LED nodes and commercial cameras. The experiment results show that (1) In flexible sensing distances up to 2.5 m, RoFin achieves an average labeling parsing accuracy of 85%. (2) In comparison to vision-based techniques, RoFin improves the tracking grain with 4× more sampled points each frame. (3) RoFin reconstructs a hand pose in real time with 16 mm mean deviation error compared with Leap Motion under flexible distance.

CCS CONCEPTS

• **Human-centered computing → Interaction devices**; **Ubiquitous and mobile devices**.

KEYWORDS

rolling shutter, sensing, hand pose reconstructing, deep learning, wearable device, optical wireless communication

ACM Reference Format:
Xiao Zhang, Griffin Klevering, Juexing Wang, Li Xiao, Tianxing Li. 2023. RoFin: 3D Hand Pose Reconstructing via 2D Rolling Fingertips. In *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys '23), June 18–22, 2023, Helsinki, Finland*. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3581791.3596838

1 INTRODUCTION

Human hands are not just crucial, vital organs for catching and grabbing; they have also long been used for communication, such as in greetings, sign language for the deaf, or hand signs in sports and wars. Hand poses have become direct, and cost-effective Human-Computer Interaction (HCI) across a wide variety of applications due to the fast development of computer technology and artificial intelligence (AI). For example, fingers and hands can be used in smart homes to control IoT devices (e.g., turning devices on/off), interact with video games for a user-friendly and immersive gaming experience (e.g., accelerating race cars), and in XR (AR, VR, and MR) enabled mobile applications to provide interactive operations that are close to reality (e.g., navigation)[10, 15, 28–33, 36, 37].

Some researchers attach on-body sensors (e.g., accelerators, gyroscopes) to each finger and joint to measure the spatial position variation of fingers. XSens[4] conducts 3D motion capturing enabled by specific and expensive miniature MEMS inertial sensors (e.g., MOVELLA DOT SENSOR price is $132). Other studies utilize wireless signals including radio frequency, sound, and lights (e.g., soli[16], FingerIO[17], and Ali[15]) for hand-free gesture recognition. However, these methods require the expensive or specific devices and have limited sensing distance less than 0.5m.

The vision-based hand gesture identification approaches are popular, which adopts the similar processing as human eyes to detect the morphology of hands with about 60Hz frequency of perception. The accuracy of vision-based hand gesture recognition achieves more than 80% with the help of deep learning[28]. However, vision-based approaches have the following drawbacks: (1) they can not work well in the low light conditions or long detection range due to the limited light amount reflecting from the hand to the camera's image sensor, (2) low sampling rate (e.g., 60 Hz) [9, 19, 25, 34, 35] of cameras when tracking fingers, the same as human eyes with limited perception ability can not see the detailed motion trajectory of Parkinson sufferers' trembling hands clearly, (3) high processing cost and latency because of their recognizing hand morphology with about 20 hand joints, and (4) the captured frame of the scene with hands also poses privacy concerns in sensitive circumstances.

Commercial cameras and LEDs are deployed everywhere, enabling optical camera communication (OCC) a reality in our daily lives. The **rolling shutter** in commercial cameras exposes one row of pixels and generates a whole image row by row. A clear **strip**

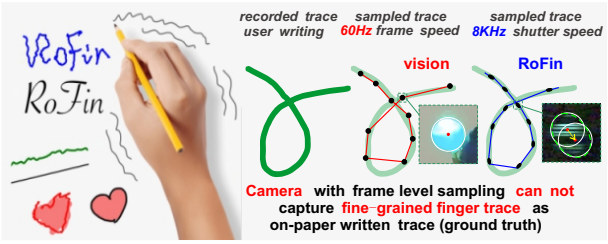
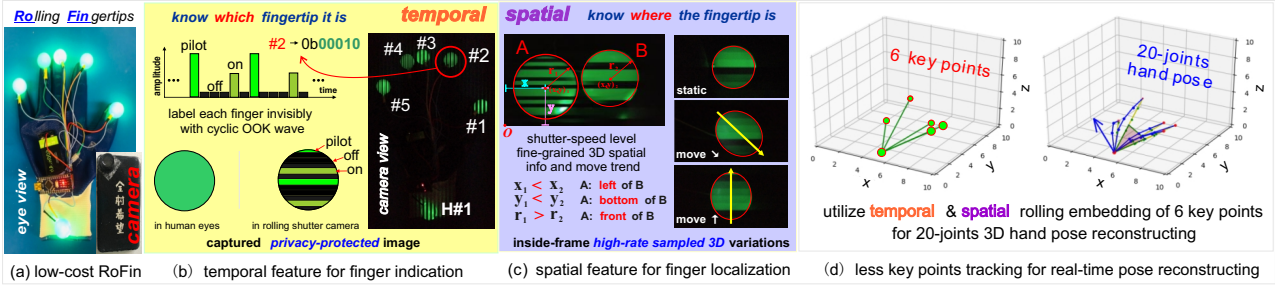


Figure 1: RoFin with shutter level sampling can better record jitters of hand trace compared with frame level sampling.

MobiSys '23, June 18–22, 2023, Helsinki, Finland

ACM ISBN 979-8-4007-0110-8/23/06…$15.00
https://doi.org/10.1145/3581791.3596838



(a) low-cost RoFin (b) temporal feature for finger indication (c) spatial feature for finger localization (d) less key points tracking for real-time pose reconstructing

Figure 2: 3D hand pose reconstructing via 6 temporal-spatial 2D rolling patterns from fingertips and the wrist.

effect appears when the switching speed of the light wave from the transmitter is equal to or slightly less than the rolling shutter speed. Many researchers have tried to improve data rates by collecting data in rolling strips rather than the entire image frame. However, these systems[25–27, 35, 36] only exploit rolling shutter for communication instead of sensing such as inside-frame fine-grained location tracking with high sampling rate (rolling shutter speed, e.g, 5 KHz) instead of one sample (1Hz).

In this work, we propose **RoFin**, which is a low-cost and privacy-protected approach for real-time 3D hand pose reconstructing with fine-grained finger tracking ability in flexible distance and varied ambient light conditions. RoFin consists of wearable gloves and a commercial camera, as shown in Figure 2 (a). Each glove finger and the wrist is attached to one low-power LED node controlled by Arduino Nano (<$10). The receiver is the commercial camera (e.g., a smartphone camera). Before employing RoFin, both frame and shutter speeds can be modified manually and quickly using a built-in camera app in professional mode or a third-party app on an Android or iPhone (e.g., Protake).

RoFin **first** exploits **2D temporal-spatial rolling** fingertips for ***(1)*** *active optical labeling for fingers/hands*, ***(2)*** *fine-grained inside-frame finger tracking with rolling shutter speed*, and ***(3)*** *real-time 3D hand pose reconstructing*, as shown in Figure 2 (b), (c), and (d) separately. Each LED node covered with same-size sphere emits distinct light waves as optical label invisible to human eyes but perceptive by rolling shutter cameras for robust finger identification. Based on the captured spots (deformed ellipses) via rolling shutter at high sampling rate (e.g., 5 KHz), RoFin can parse fine-grained 3D locations and inside-frame variations of fingertips (left/right, up/down, and front/rear). Finally, RoFin reconstructs 3D hand pose consisting of 20 points by tracking only 6 key points (5 fingertips and 1 wrist point) for less latency and computation overhead.

There are several commercial approaches closely related with RoFin. Leap Motion[3] utilizes infrared cameras with illumination LEDs to precept hands' morphological variation frame by frame. Luxapose[12] exploits fixed several LED landmarks on the ceiling to interact with rolling camera for indoor localization instead of hand pose reconstruction. In contrast, our RoFin hand gesture recognition has the benefits of: **(1) Less tracking overhead.** Instead of entire hand morphology (i.e., 21 joints), RoFin recognizes a hand with 6 nodes tracking. **(2) Massive optical labeling.** RoFin recognizes multiple hands with identities via high-capacity active optical labeling. **(3) Flexible usage scenarios.** Our RoFin is portable and can work at distance up to 3m in both day and night/dark. **(4) Fine-grained sampling.** RoFin can sample fine-grained 3D fingertips in dynamic at high rolling shutter rate. **(5) Mutual blockage avoidance.** RoFin can avoid most blockage issues because it only uses the fingertips' area rather than the entire hand morphology.

However, we must overcome three **technical challenges**: **C1:** Each finger from multiple hands requires a distinct label identifiable robustly in varied ambient light and long distances. **C2:** It is difficult to decipher the fingertip's fine-grained 3D fluctuation based on the 2D shape (i.e., a distorted ellipse) that was recorded during a frame period. **C3:** We merely track of a hand's 6 key points for reduced overhead. It is a challenge to reconstruct a 20-point 3D hand pose accurately from restricted 6 key points in real-time.

Our **contributions** can be summarized as follows:

(1) RoFin is the first work to exploit rolling shutter effect for 3D hand pose reconstructing via 2D rolling patterns of fingertips. We indicate each fingertip and wrist point with asynchronous cyclic optical labels. Then we adopt a lightweight CNN model with bounding boxes to identify fingertips and wrist points. Our active optical labeling overcomes the limitations of the vision-based technique and is appropriate for the identification of multiple hands in low-light and long-range detection scenarios.

(2) We creatively utilize inside-frame high sampling via rolling shutter to track several fingertips' 3D location variation instead of only one 2D location sample in a frame to enhance tracking granularity further while vision-based approaches only use one 2D location sample during one frame period. The improved finger tracking ability has potentials for the virtual writing for Parkinson's suffers, better user experience for virtual writing/painting in AR/VR/MR, etc. Our RoFin can also be adopted in telesurgery by transferring the fine-grained tracked finger traces of multiple doctors and nurses robots.

(3) Based on the finger identification and parsed 3D location info of 6 key points (5 fingertips and 1 wrist point) from (1) and (2), we design a real-time and lightweight 20-point 3D hand pose reconstructing algorithm HPR from tracked 6 key points. HPR can efficiently reconstruct a 3D hand pose by direct calculation instead of redundancy points' tracking while not sacrificing the reconstructing accuracy. Even a if specific fingertip is blocked by others, RoFin can easily infer which finger it is as well as its location due to the fingers relation of a users' hand fixed structure.

(4) We implement RoFin with commercial devices and evaluate its performance of (i) finger identification performance in different settings, (ii) inside-frame tracking enhancement in comparison to the vision-based approach, and (iii) hand pose reconstructing error with Leap Motion as the benchmark and its reconstruction latency. End-to-end and dynamic-scenario evaluations of RoFin

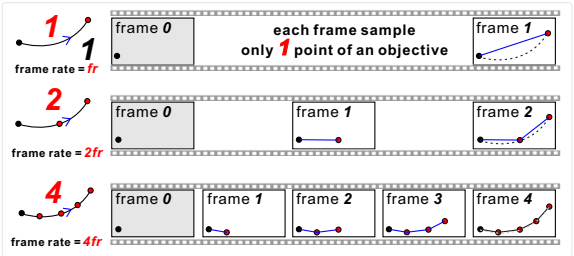


Figure 3: Frame rate vs. tracking granularity.

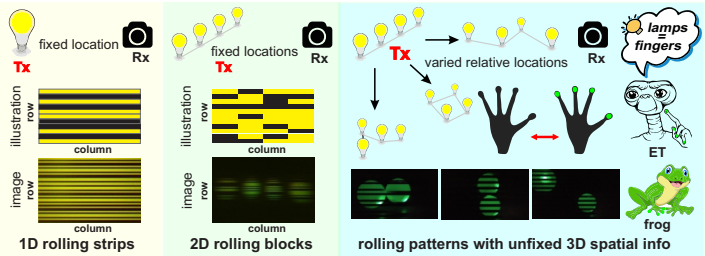


Figure 4: Strip effect from single to multiple light sources.

are conducted. We also discuss potential use cases such as multi-user interaction for meta, virtual writing or health monitoring for Parkinson suffers, hand pose commands for smart home, etc.

2 BACKGROUND AND RELATED WORK

2.1 Vision-based 3D Hand Pose Recognition

Numerous works adopt cameras to recognize hand poses. In general, these computer vision approaches can be classified into 2 categories. (1) Hand image searching in pre-computed databases with machine-learning assistance. These methods capture hand images and then query pre-computed 3D hand models to determine the best-matched hand pose[6, 14, 22]. (2) Calculate 3D coordinates of hand joints directly and then identify the hand pose by optimizing an objective function. These methods represent the hand with a 3D hand model and adopt an optimization strategy to speed up hand pose prediction[18, 20, 28] However, these existing vision-based hand pose recognition methods are based on complete hand morphology such as hand silhouettes and numerous joints (e.g., 20 joints) with non-trivial tracking and computation overhead.

Furthermore, vision-based approaches sample the location variation at the frame update level shown in Figure 3 while the frame rate is set $\leq$ 60 fps instead of higher to be compatible with time-consuming image processing. In contrast, RoFin enables 3D hand pose reconstruction via incomplete hand morphology, relying on capturing only six 2D rolling spots (5 fingertips and the wrist point) of a hand. With fewer tracking points and a lightweight pose reconstruction algorithm HPR, RoFin achieves the real-time reconstruction of a hand with an average 13.8 ms time cost.

Moreover, even with limited 60 fps frame rate, RoFin can sample numerous inside-frame points thanks to the higher rolling shutter rate instead of only 1 in vision approaches, thus enhances the finger tracking granularity which is explained below.

2.2 Strip Effect in Rolling Shutter Camera

Cameras in our daily-used smart devices adopt low-cost **rolling shutter** to reduce the readout time of pixels from the whole image frame. Rolling shutter camera exposes one row of pixels and generates a whole image row by row. A clear **strip effect** appears when the switching speed of the light wave from the transmitter is equal to or slightly less than the rolling shutter speed, as shown in Figure 4. As a result, optical signals containing transmitted data in a symbol period may be sequentially captured in these rolling strips, which enables optical camera communication (OCC) such as CASK, ColorBar, etc[25, 26, 36].

However, the high-rate sampling ability of rolling shutter camera is wasted in current vision-based finger tracking and hand pose recognition. These approaches only sample one location of specific objective (e.g., a fingertip) in a frame despite the rolling shutter camera can capture numerous location samples during a frame period. In contrast, RoFin fully utilizes these numerous location samples to enhance the inside-frame finger tracking granularity via active LED spheres attached on fingertips. The deformed ellipse will be generated when the finger is in high-motion status (e.g., shaking) and records the location variation of centre of LED sphere in 3D space during a frame period, as shown in Figure 2 (c) and 4.

As result, RoFin can improve virtual painting and writing user experience, especially for Parkinson suffers. Parkinson sufferers' tremor frequency (10Hz[21]) indicates how frequently their hands return to the same spot (i.e., the hand shakes from center to left and then to right, then back to the center). Although the frame rate of 60Hz can catch one cycle of tremor with 60Hz/10Hz=6 sampled points, RoFin samples more points at higher shutter speed (e.g., 10KHz). Considering these tremor cycles with random 3D motion routes, more sampling points for each cycle are indeed needed for fine-grained random tremor trajectories tracking and further trace optimization rather than vision-based coarse-sampled traces.

3 ROFIN SYSTEM OVERVIEW

In this section, we briefly introduce the composition, workflow and 3 main tasks of RoFin, as illustrated in Figure 5.

Composition. RoFin system consists of two parts. **(1) RoFin gloves** are commercial insulating gloves where each fingertip and the wrist are attached with a low-power LED component covered with a plastic ball. These LED components are controlled by an Arduino Nano to generate identical LED waves to indicate different fingertips. **(2) RoFin reader** is based on commercial cameras (e.g., smartphones, web cameras, etc). These cameras use adjustable focal length lenses and rolling shutters with configurable shutter rates.

Workflow. (i) The user puts on RoFin gloves and makes some hand poses. **(ii)** After setting the rolling shutter rate and focal length, RoFin reader captures the continuous 2D rolling spots of six key points (5 fingertips and 1 wrist point) frame by frame. **(iii)** RoFin reader identifies each fingertip/wrist point via lightweight CNN model with bounding boxes (i.e, YOLOv5). **(iv)** RoFin parses the 3D location variations of each key point based on captured deformed ellipses in each frame with the granularity of strip width. **(v)** Finally, RoFin reconstruct 3D hand pose via lightweight HPR algorithm based on the parsed label and its fine-grained 3D location.

3 Main Tasks. At the high level, RoFin responds to two questions: **(1) identify which fingertip** (*Section 4*) it is, and **(2) locate position and its inside-frame variation** (*Section 5*) of this fingertip with sampling rate at rolling shutter speed. RoFin further **(3)**

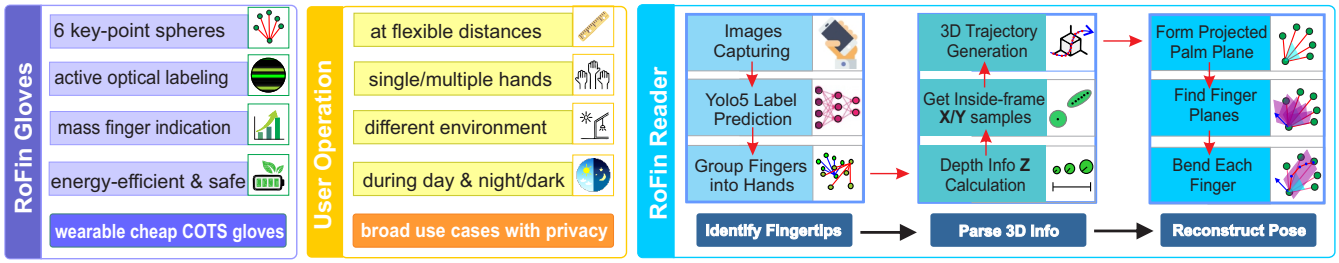


Figure 5: RoFin system overview: composition, workflow and three main tasks.

reconstructs 3D hand pose (*Section 6*) via HPR algorithm based on outputs from (1) and (2).

4 ACTIVE OPTICAL LABELING

4.1 Temporal Rolling Patterns

The light source emits optical signals varied with time sequences at rolling shutter speed level during one frame period can be recorded row by row in the captured image frame by the rolling shutter camera, as illustrated in Section 2.2. In our prototype, the Arduino Nano allows us to control and configure the ON/OFF switching rate of the LEDs by running executable codes onboard. The shutter rate is configured manually on smartphone. We set rolling shutter rate (i.e., the parameter `shutter speed`) via the professional mode of the system camera for Android smartphones and the Protake app for iphones.

Only when the rolling shutter rate is similar to the transmission frequency, however, can we clearly see the distinct rolling strips, as illustrated in Figure 6 (a). We can utilize captured rolling spots with distinct strip textures as active optical labels to **indicate fingertips**. However, optical signals have multiple light features varied with temporal sequences such as different amplitudes, transmission frequencies or colors can be used for indication of different fingertips. Which ought to be used in rolling patterns for RoFin?

Our design exploration is shown in Figure 6 (b). **(1) Amplitude.** We can adjust brightness of the light source with time sequences[8, 23, 38]. The light amplitude fluctuation is vividly captured sequentially. **(2) Transmission Frequency.** We may also alter the ON/OFF switching speed of the light wave. **(3) Color Spectrum.** We could transmit the light with different wavelengths. The captured rolling strips are colourful and vary in the same way of color fluctuation with time sequences as the light source does.

Our choice: Amplitude. It requires RGB LED and complicated modulation to achieve **color spectrum** diversity. Complex modulation and a longer time period to present complete frequency variation (i.e., only partial of the complete pattern could be presented on the captured spot of sphere with limited width) are both necessary for **transmission frequency** diversity. To indicate multiple fingertips, **amplitude variation** is more suitable compared with different colors or transmission frequency which require more complex devices (i.e., multi-color LED, high-clock-rate MCU) and control overhead. To create a low-cost design while remaining robust, RoFin uses single-color (i.e., green, or other color options) LEDs and Pulse Width Modulation (PWM) based amplitude configuration.

4.2 Fingertip and Hand Indication

Each attached LED element can emit the different amplitude waves as the active optical labels. However, we can not synchronously control each light source to let them start temporal rolling pattern at the same time. Additionally, because of their various positions inside the field of view (FOV) of the camera, the recorded rolling strip may begin at a different time. These asynchronous problems make it difficult for the RoFin reader (i.e., camera) to recognize the embedded identification information from different light sources (i.e., LEDs). Thus, we design asynchronous Cyclic-Pilot On-Off-Keying (CP-OOK) labeling scheme for different fingertips from multiple hands and wrist-assisted hand indication.

4.2.1 CP-OOK based Fingertip Indication.

The optical label consists of two parts: **(1) CP (cyclic pilot)**, takes one symbol period at the beginning, and **(2) indication sequences**, formed via 5 (can be extended) OOK (On-Off Keying) symbols, as shown in Figure 7 (a). Aside from the Off symbol (dark), the optical label design has two amplitude levels: the CP symbol has the highest brightness, whereas the On symbol has the lower brightness of CP.

Instead of the normal very long preamble[1], we designed a short pilot (i.e., CP). Because the number of rolling strips revealed in the finger pattern (the circle or ellipse) is restricted, we must ensure that at least one complete optical label is shown in each rolling pattern

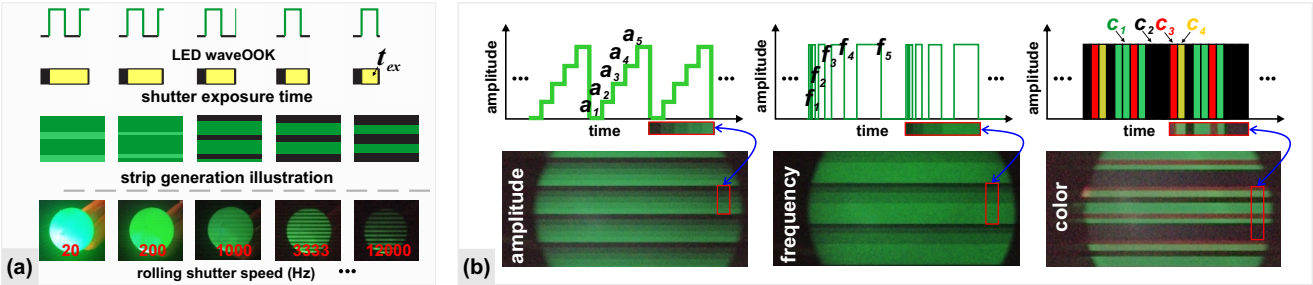


Figure 6: (a) Captured strips impacted by shutter speed. (b) Light feature selection for temporal rolling patterns.

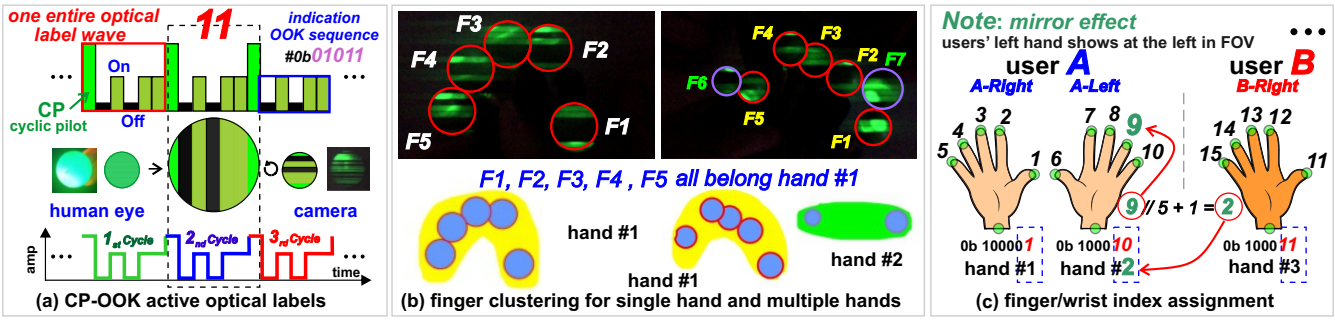


Figure 7: The scheme design of CP-OOK fingertip indication and wrist point labeling. Based on parsed indication number of fingertips and wrist points, the different hands are clustered for further finger tracking and hand pose reconstruction. The indication capacity of RoFin is flexible and massive by easily adjusting the number of the indication sequence.

for further decoding. Furthermore, to improve the robustness of these optical labels in variable environment, we set a total of 2 non-dark amplitudes ($\mathtt{Am}_{CP}$, and $\mathtt{Am}_{On}$) instead of additional amplitude levels (e.g., 5 amplitude levels in amplitude shift keying).

We encode the index of each finger with its binary number into OOK symbols, as shown in Figure 7 (a). When the finger index is 11, for example, the binary number is *01011* and the indication sequence is [*Off, On, Off, On, On*]. The length of the indication sequence is determined by the number of fingers being tracked. 3 OOK symbols can represent up to 8 fingers, enough for 1 hand. 4 OOK symbols can represent 16 fingers, enough for 3 hands. In general, N OOK symbols can represent 2^N fingers that are appropriate for $2^N/5$ hands. The transmission frequency of light waves have the same or slightly slower frequency than the rolling shutter, and thus these optical labels are clearly recorded for further finger identification, as shown in Figure 7 (b). RoFin has great potential of massive indication capacity. The number of indication sequences can easily be adjusted by setting proper rolling strip width (i.e., configure transmission frequency and shutter speed). For example, N can be set to 7 to indicate 2^7 = 128 fingers of 12 users. In contrast, Leap Motion is not extendable and is unable to distinguish between the hands of various players in multi-user games and HCI. For instance, 4 Leap-Motion users pay $4 \times 250 = 1000$, while 4 RoFin users cost \$150 (\$50 of a commercial camera plus $4 \times$ \$25 of gloves).

4.2.2 *Wrist-assisted Hand Indication.*

We assign each finger from multiple hands of multiple users with a finger index as illustrated in Figure 7 (c). For example, there are users A, B, and so on. We assign the A's right hand as the hand #1, A's left hand as hand #2. And we assign the B's right hand as hand #3, and the rest can be done in the same manner. In this paper, we evaluate three hands (A's right hand and left hand, B's right hand). We assign these fingers with indication index from 1 to 15 finger by finger as shown in Figure 7 (c).

However, only 5 fingertips are not enough to determine a hand in a 3D space. Besides five fingertips of a hand, we also attach an LED node covered with same-size sphere at the end of the wrist. This additional wrist point has more vital meaning for hand pose reconstructing in comparison to other five fingertips (discussed in Section 5). Furthermore, different hands should have distinct indications for these 6 key points of each hand (5 fingertips and 1 wrist point) for correct reconstructing each hand pose when they are shown in the camera view at the same time.

Based on the analysis above, the indication of the wrist should have more significant indication than the fingertips but not introduce additional non-trivial overhead (e.g, use different light features: colored-LED, different modulation schemes: FSK, etc). To achieve this design goal, we use the same CP-OOK modulation in fingertip indication, but set the **leftmost** indication bit as **1** while the remaining bit sequence as the wrist indication for differentiation.

Given three hands #1, #2, and #3, it requires 4-bit indication sequence to denote 15 fingers. Thus, originally, the binary number for finger #11 is 1011. But we set its indication sequence as **0**1011, which is [**Off**, *On, Off, On, On*], to make it compatible for wrist point indication. For the wrist point from #2, its binary number is 10. Following the rule above, the indication sequence of this wrist point is set as 1001**0**, which is [**On**, *Off, Off,* ***On, Off***].

5 3D SPATIAL PARSING

Although vision based approaches can use higher frame rate (e.g., 120 fps or 240 fps) for sampling, the image processing is still time-consuming. Thus, vision based approaches can not achieve the fast hand pose reconstructing with the same speed of faster frame rate. Thus, vision based approaches normally set the frame rate at about 60 fps for real-time user experience. Different from vision based approaches which only use one 2D location sample (x, y) in each image frame, RoFin has two specific abilities (Note: cameras in RoFin can be set with fixed location or in mobile):

(1) Ability of tracking more 2D location samples (*multiple X and Y values of the center point of the sphere in motion, i.e., inside-frame X/Y sampling*) in each image frame thanks to its sampling at shutter rate instead of frame rate. This ability can enable RoFin's **first** core function: **finer-grained finger's tracking (Section 5.1)**. We introduce finer-grained X/Y tracking inside one frame and among continuous frames in Section 5.1.1 and 5.1.2 separately.

(2) Ability to reflect depth info Z besides X and Y (Z value: from the sphere to the camera) via perspective principle based on the captured sphere's size variation. Combined with a YOLO's bonding box parsed one X value and Y value (*the center point of the sphere, i.e., inter-frame X/Y sampling*) in each image frame at 60 fps frame rate, the pair of X,Y, and Z values can be used as the **3D location input** for RoFin's **second** core function: **real-time 3D hand pose reconstruction (Section 6)**. We present YOLO enabled rolling label identification and inter-frame X/Y position in Section 5.2.1 and Z calculation in Section 5.2.2.

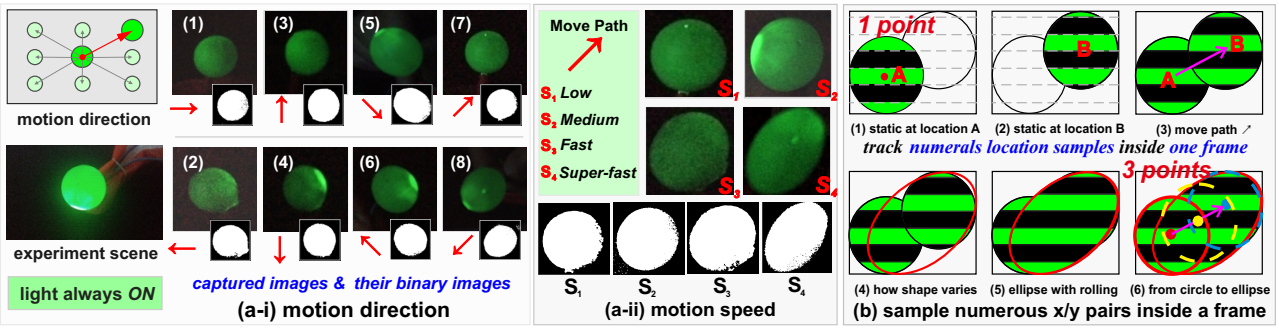


Figure 8: (a) Impacts of the shape variation of deformed ellipse: (i) motion direction and (ii) motion speed. (b) The sphere center's location variation is recorded in the deformed ellipse with the granularity of strip width.

5.1 Finer-grained X/Y Tracking

5.1.1 *Inside One Frame.*

Why high-rate inside-frame sampling? The objectives are mostly in mobile with random trajectory in real-life situations (e.g., vehicles, drones, or fingers). For example, it is required for numerous location samples in unit time to recover the real trajectory of fingertip as brush in virtual writing/painting. Either a long, random curve that is drawn quickly or a small curve requires more samples to capture more details. However, existing vision-based approaches sample the location variation at the level of frame update. Besides, the frame rate is set to about 60 fps instead of higher frame rate considering the time-consuming image processing. To break this gap, we creatively propose to utilize rolling shutter effect for numerous inside-frame location samples.

Impact of Motion Direction. We move the light source with different directions with constant distance from the light source to the camera plane and the motion speed. For example: (1) and (2) from left to right (→) and reversed (←); (3) and (4) from bottom to top (↑) and reversed (↓); (5) and (6) from upleft to bottomright (↘) and reversed (↖); and (7) and (8) from bottomleft to upright (↗) and reversed (↙). As shown in Figure 8 (a-i), the captured spot shape changes to an ellipse rather than the previous circle and its long axis can reflect the moving direction of the light source.

Impact of Motion Speed. We set 4 levels of motion speed of the light source (i.e, low, medium, fast, and super-fast) with the same motion direction (↗) and fixed distance to the camera plane. As the motion speed increases, so does the length of the ellipse's long axis, as shown in Figure 8 (a-ii).

Numerous Inside-frame X/Y Location Samples. The captured circle or ellipse's pixel index range in columns and rows reflects the horizontal and vertical location information independently. The circle shape means the fingertip/wrist point is not moving or moving slow in the image plane during the entire frame period, and its center location (x, y) can be treated as its location in horizontal and vertical directions. The deformed ellipse records the detailed inside-frame motion with the sample rate at rolling shutter speed, as illustrated in Figure 2 and Figure 9 (b).

5.1.2 *Among Continuous Frames.*

As shown in Figure 8 (a-i), the opposite moving direction of the light source has the same rolling pattern shape (i.e, the ellipse with similar long axis direction). For example, there are 3 frames in Figure 9 (a): `frame1`, `frame2`, and `frame3`. In `frame2`, the light source may move with possible trends as (↗) or (↙) and thus we can not determine fingertip's motion with separate frame.

Moving Trend Determination. However, if we combine the inside-frame moving direction candidates with two continuous frames, we can know the finger's moving trend. Because these frames are continuously generated with time sequences, the end position of finger pattern in previous frame will be close to the start position of finger pattern as shown in Figure 9 (b). Thus, we can determine the finger's moving trend by finding the closest positions of finger pattern in two continuous frames. In this example, the position point RL_{1-end} and $\mathrm{RL}_{2-start}$ are the closest position points between two continuous frames `frame1` and `frame2`.

Moving Trajectory Generation. More importantly, the moving trend determination method is a one-time initialization phase that only requires one frame duration to determine the end positions of each finger pattern and record them as the start positions for the next frame. In this example, using the finger pattern position in `frame3`, we can know the point $\mathrm{RL}_{3-start}$ is the start point. Then we can track finger locations by combining these numerous inside-frame samples and updating them frame by frame, as illustrated in Figure 9 (b). Finally, we can generate a finer-grained moving trajectory in RoFin than the vision-based approach.

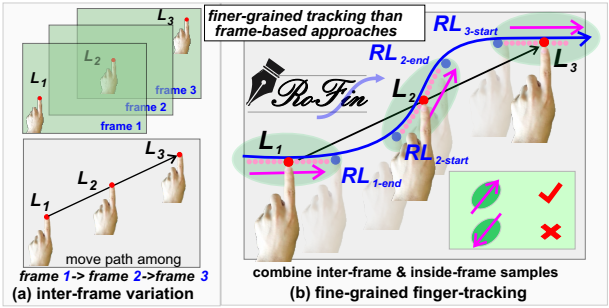
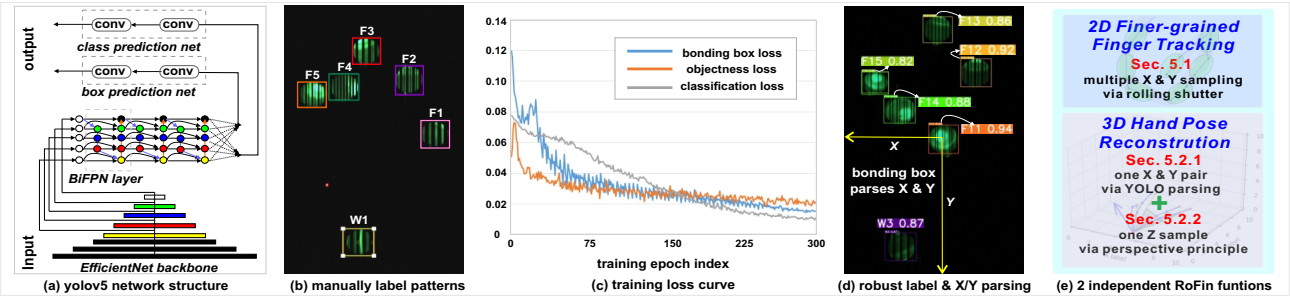


Figure 9: RoFin's Finger Tracking among frames combined with numerous inside-frame samples.

5.2 3D Location Pair as HPR Input

5.2.1 *Rolling Label Identification and X/Y Parsing via YOLO.*

Traditionally, we could decode these optical labels based on the amplitude thresholds via computer vision tools . However, due to the variable optical environment, it is difficult to configure the thresholds dynamically. Even in the same ambient light settings, the captured rolling pattern for each finger requires different thresholds



(a) yolov5 network structure (b) manually label patterns (c) training loss curve (d) robust label & X/Y parsing (e) 2 independent RoFin funtions

Figure 10: (a) The adopted CNN network architecture. (b) Example of manually label key points. (c) The training loss curves of bonding box loss, objectness loss (confidence of object presence), and classification loss with 300 epochs. (d) Example of YOLO parsed key point label and X/Y pair. (e) Two independent functions in RoFin and related section content.

for decoding. Convolutional Neural Networks (CNN) are widely applied in computer vision object classification due to their great robustness and accuracy. The benefits include: (1) Offline training and online identification can reduce latency for real-time finger label parsing; (2) even in high ambient light and difficult to distinguish CP and On, the CNN model can learn the features in the repeating dark and bright rolling strips.

We adopt YOLOv5 for optical label identification with related bounding boxes. YOLO (You Only Look Once) models are commonly used for objects detection since their fast inference with high accuracy. The network structure of YOLOv5 consists of EfficientNet backbone structures, BiFPN (Bi-directional Feature Pyramid Network) layers to extract object's features effectively, as shown in Figure 10 (a). Then these features are fed through the prediction nets for both objective's class and location of boxes as output.

We capture 90 images of 3 RoFin gloves in 3 different ambient light strengths with 10 images for each setting. Then, we manually label each rolling pattern with 18 class labels (i.e., F1-F15, W1-W3). One example is as shown in Figure 10 (b). In the training, we assign 62 images in the Train dataset, 18 images in the Valid dataset, and 10 images in the Test dataset. We adopt data augmentation via the gray-scale modification to 60% of images and output 3 images per training example to increase the size of training dataset.

As shown in Figure 10 (c), the training losses decrease significantly after 300 epochs of training. We present 3 different training loss results: (1) bonding box loss, which demonstrates that generated bonding boxes are accurate, (2) objectness loss, which indicates the confidence of prediction for object presence in the image, and (3) classification loss, which can reflect the accuracy of classification of different fingertips/wrist points. Finally, we use the trained model to infer the rolling pattern's label with bounding boxes. As an example shown in Figure 10 (d), RoFin outputs labels accurately with high confidence. Besides, these outputted bounding boxes include each sphere's x,y, and radius, which are used for further hand pose reconstructing in Section 6.

5.2.2 *Depth Info Estimation: Z.*

Perspective Principle. We keep the LED light source fixed in the FOV while moving the light source closer or farther to the camera. The size of captured spot grows and shrinks separately due to perspective principle. Then, we could calculate the depth info Z (i.e., the front and back), as shown in Figure 11 (a).

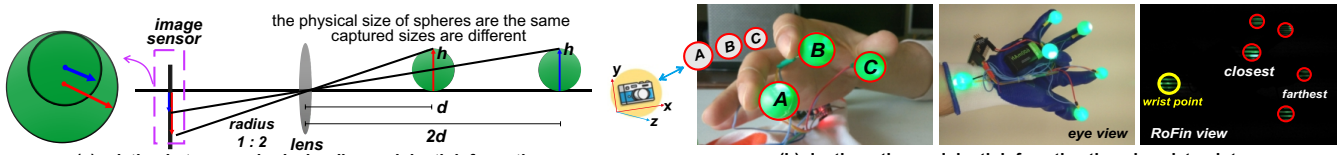
Absolute Depth Calculation. The wrist point is designed not only for assistance for hand indication illustrated in Section 4.2.2, its captured diameter ϕ_w (unit: pixel) can also be used to calculate the absolute distance of key points d to the camera. As shown in Figure 11 (b), the distances to the camera has the relations: $\frac{1m}{d} = \frac{\phi_w}{\phi_{1m}}$. Thus, the absolute distance d from the wrist point to the camera can be formulated as $d = \frac{\phi_{1m}}{\phi_w}$. To do so, we measure and store the captured spot diameter of wrist point at 1m as reference for depth info estimation of all six key points using the same manner.

Coordination Transformation. We set the center of wrist point is the origin of 3D coordinate system. As shown in the right of Figure 2, the z value of the five fingertips is set as their physically relative depth distance value to the wrist point. The center (x, y) of each fingertip's spot shown in the image plane is the pixel value which we need to convert to the physical distance as well. We also use the pixel value range of the wrist point's diameter which maps to the 19 mm of the plastic sphere as the reference to convert the relative X/Y value of each fingertip's center into their relative physical distance to the wrist point separately.

6 HAND POSE RECONSTRUCTING

6.1 Cluster Fingers and Wrists into Hands

Grouped 6 Key Points of a Hand. Based on the identified fingertips and wrists from multiple hands above, we can calculate their hand belonging separately. And then we can easily cluster fingertips and wrist points from one hand together. For example, the fingers which have indication numbers in [1, 2, 3, 4, 5] and the wrist point with an indication number of 1 should be grouped in hand #1 due to their calculated hand index being the same, which is 1. As shown in Figure 12 (a), the wrist labeling avoids the wrong



(a) relation between spherical radius and depth information (b) depth sorting and depth info estimation via wrist point

Figure 11: Absolute depth calculation via perspective principle by using wrist point as the reference.

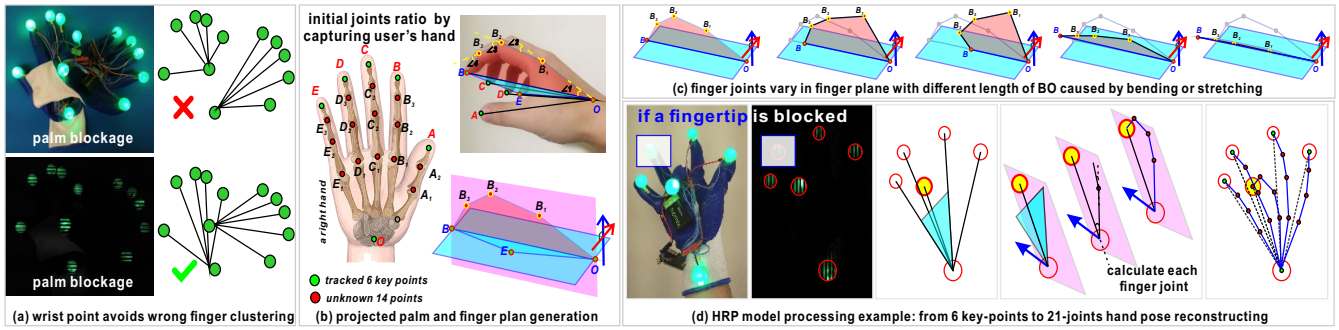


Figure 12: Finger clustering with the correct wrist point into a hand and the illustration of the HPR (hand pose reconstructing) model for hand pose reconstruction via six tracked 3D key points. Palm or fingertip blockage situation and RoFin initialization.

finger clustering with the wrist point from another hand and thus guarantees further accurate hand pose reconstruction.

3D Coordinates of 6 Key Points. The 6 key points with 3D coordinates clustered into one hand will be input into the HPR model and then the HPR model outputs the reconstructed 3D hand pose in real time. Different from fine-grained finger tracking with numerous inside-frame sampled points in an image frame (Section 5), the real-time hand pose reconstructing requires only one 3D location sample for each of six key points per frame for processing.

6.2 Lightweight HPR Model

RoFin is initialized by taking one/several photos of a specific user's open and unobstructed hand to obtain joint ratios before the usage, as shown in the left of Figure 12 (b). Our goal is to calculate the unknown joints according to 6 key points with 3D coordinates (the wrist point p_O, the tip of thumb p_A, the tip of index finger p_B, the tip of middle finger p_C, the tip of ring finger p_D, and the tip of little finger p_E), as shown in Figure 12 (c). However, how can we reconstruct a 20-joints hand pose? The intuitive answer is to calculate 3D locations of the other 14 points: p_{A_1}, p_{A_2} (i.e., we simplify the thumb finger with 2 joints), p_{B_1}, p_{B_2}, p_{B_3}, p_{C_1}, p_{C_2}, p_{C_3}, p_{D_1}, p_{D_2}, p_{D_3}, p_{E_1}, p_{E_2}, and p_{E_3}.

6.2.1 *Projected Palm and Finger Planes.*

The Plane of Projected Palm. As shown in Figure 12 (b)-(d), the fingers and the palm can be projected on the plane which we defined as projected palm P_{palm}. Actually, the index finger tip, the little finger tip and the wrist point consist of the P_{palm} (i.e., P_{BOE}).

The Plane Formed by Finger Joints. The joints of a finger form a finger plane. These finger planes (except the thumb finger plane) are perpendicular to the plane P_{palm}. For example, joints of the index finger: p_{B_1}, p_{B_2}, p_{B_3}, p_B, and the wrist point p_O generates the finger plane $P_{OB_1B_2B_3B}$. And $P_{OB_1B_2B_3B} \perp P_{palm}$ (i.e., $P_{OB_1B_2B_3B} \perp P_{BOE}$). In contrast to finger planes above, the thumb finger plane is almost parallel to the plane P_{palm} (i.e., $P_{OA_1A_2A} \perp P_{BOE}$). Thus we can find these 5 finger planes based on the known plane P_{BOE}, as shown in Figure 12 (b)-(c).

6.2.2 *Bending Fingers in Finger Planes.*

Given the 5 connection lines between each fingertip to the wrist point (i.e., l_{OA}, l_{OB}, l_{OC}, l_{OD} and l_{OE}) and the calculated finger planes $P_{OA_1A_2A}$, $P_{OB_1B_2B_3B}$, $P_{OC_1C_2C_3C}$, $P_{OD_1D_2D_3D}$, and $P_{OE_1E_2E_3E}$, we can determine the unknown 14 joints (underlined, the red joints in Figure 12 (b)) on the finger planes via two rules.

R1: We can simplify the finger bending because each finger section from one finger bends with a same or proportional angle.

R2: The length from fingertips to the wrist l_{con} equals to the sum of each finger section's projection to the line l_{con}. We can calculate the bending angle and further find each unknown joint location.

As shown in Figure 12 (c), the finger joints of the index finger vary in its finger plane with different lengths of l_{con} (i.e., l_{OB}). Thus, given a value of variable l_{con}, the 3D locations of other joints from this finger are fixed and can be calculated. For example, we know the length of each finger section of the index finger (i.e., l_{OB_1}, $l_{B_1B_2}$, $l_{B_2B_3}$, and l_{B_3B}) by the initial measurement step. Given the calculated l_{OB} (i.e.,l_{con}) from Section 5, the unknown bending angle for index finger $\angle\alpha$ can be calculated by the equation below:

$$l_{OB_1} \times \cos 2\underline{\alpha} + l_{B_1B_2} \times \cos\underline{\alpha} + l_{B_2B_3} \times \cos\underline{\alpha} + l_{B_3B} \times \cos\underline{\alpha} = l_{OB}.$$

6.2.3 *Hand Model Comparison, Tradeoff, Blockage and Limitation.*

Hand Model Comparison. Similar to the standard IK (inverse kinematic) model[5], which is to decide the parameters to control each joints of the end effector (e.g., robot hand/arm) to present a specific gesture in a physical or virtual 3D space, our RoFin significantly simplified the calculation via 2 rules above instead of Jacobian matrix which considers the joint speed and moving power. Besides, IK calculates joints from the root joint to the end joint (i.e, wrist to the fingertip) while our RoFin calculate middle joints based on the wrist and the fingertip.

Key Points Tradeoff. There are other options of the number of tracked key points to reconstruct a hand, which is a tradeoff problem between hand pose reconstruction accuracy and latency. For example, if we track A,B,C,D,E,O and other joints such as joints A_2 of thumb, E_2 of the little finger in the top left of Figure 12 (b), it will definitely improve the hand pose reconstruction accuracy by avoiding the calculation error based on the simple R1 and R2 rules. More accurate reconstruction with more tracked points means more tracking overhead. How many proper tracked joints to be set should be suitable to specific application goal and precision requirement.

Blockage and Limitation. Current RoFin can not work when the whole hand is blocked by other hands or objectives, the same as vision approaches in which the limitation is caused by the optical nature of LoS propagation. However, if the hand parts except the 6 key points are blocked, as shown in Figure 12 (a), our RoFin can still work while vision approaches can not. RoFin can determine which/where a blocked finger is by looking at other unblocked fingers because blockage also represents relative spatial placements of fingers in a hand, as shown in the bottom left of Figure 12 (d).

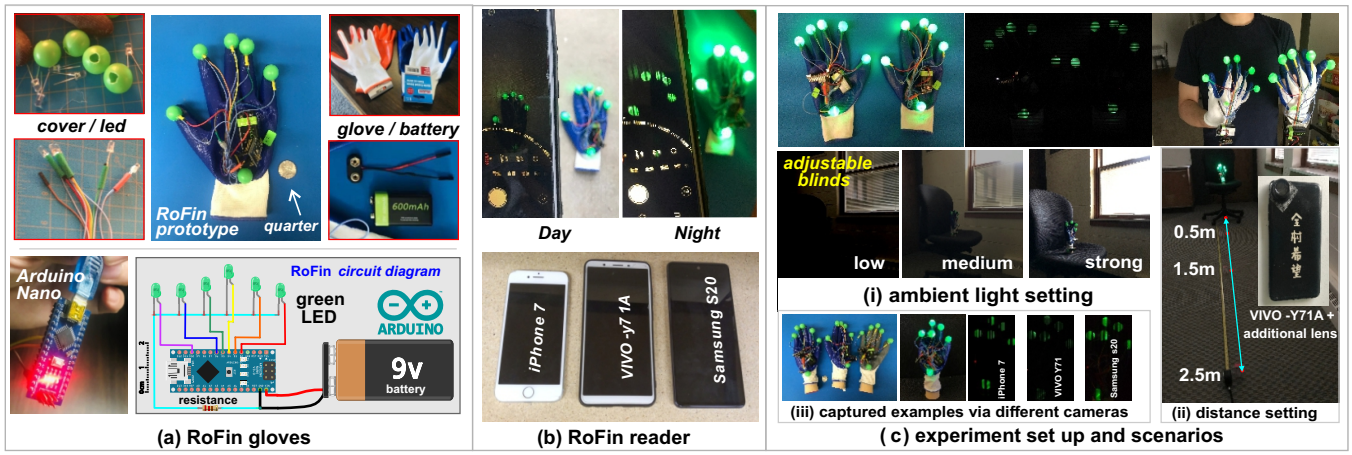


Figure 13: System implementation: RoFin gloves (prototype & circuit diagram), RoFin reader (commercial cameras) and experiment scenarios (varied ambient light strength and distances from 0.5m to 2.5m).

7 SYSTEM IMPLEMENTATION

7.1 RoFin Gloves

We implement three wearable RoFin gloves for experiments as shown in Figure 13. The main components in one pair of RoFin gloves are shown in Table 1: lightweight insulated breathable gloves, 2 Arduino Nano MCU, 12 green LEDs wrapped with 12 green plastic balls (ϕ = 19mm), and a 9V li-ion battery for power-supply. The spheres we used are industrially produced with the same size and standard sphere shape, thus there is no influence caused by the directivity of the LED and unbalanced brightness issue. The total weight of one pair of RoFin glove is **132g** (including two batteries' weight of 60g) while the total price is only **26.3**$.

Table 1: Components in one pair of RoFin gloves.

Component	Price (USD)	Details
insulated gloves	0.6 x 2 = 1.2	for each: 24cm x 15cm, 18g
Arduino Nano	10 x 2 = 20	ATmega328P, 5V, 16M
LED	0.02 x 12 = 0.24	5mm, green, 20000mcd, 20mA
plastic cover	0.08 x 12 = 0.96	19mm, green, lightweight
battery	2 x 2 = 4	2 rechargeable batteries 7x2 = 14$
Total price	26.3	mass produced, cheaper the price

7.2 RoFin Reader

There are numerous commercial smart devices widely available and reasonably priced that can be used as our RoFin reader including smart phones, drone cameras, and even underwater sports cameras. In our experiments, we use commercial smartphones with additional lens such as iPhone 7, VIVO Y71A, and Samsung s20, as shown in Figure 13 (b) and (c). The camera parameters including shutter rate, resolution and so on can be easily configured via camera apps on smartphones (e.g., Protake app).

8 PERFORMANCE EVALUATION

We evaluate the RoFin's performance in three folds. **(1) label identification** with different ambient light settings, distances, cameras. **(2) inside-frame tracking performance** in contrast to vision-based method. **(3) hand reconstruction performance** with Leap Motion as the benchmark. Then we also discuss about RoFin's use cases and other concerns such as privacy and power consumption.

8.1 Robust Label Parsing

(1) Impact of Ambient Light. We use trained YOLO model to predict labels in captured images by VIVO-Y71A in 3 different ambient light settings [low, medium, strong] at the same distance 0.5m of Hand #1. As shown in Figure 14 (a), the label parsing achieves the best accuracy under the strong ambient light at 0.94 and the average accuracy of 0.91. These results demonstrate RoFin's label parsing works robustly under varied ambient light even in the darkness and outperforms than vision-based approaches, which can not work in the darkness and lack of identification ability.

(2) Impact of Sensing Distance. We predict the labels in the captured images via VIVO-Y71A in 3 distance settings [0.5m, 1.5m, 2.5m] under strong ambient light of Hand #1. The average accuracy of label parsing is shown in Figure 14 (b). The accuracy of label parsing drops slowly with increased distance. RoFin achieves the best accuracy of 0.93 at 0.5m and 0.77 at 2.5m and an average accuracy of 85% label parsing. Although the 15% label parsing error within 2.5m may cause 6×15% = 0.9 key point label parsing error out of a hand, we can infer its label based on other parsed labels and improve label parsing accuracy by training with larger dataset. These results demonstrate RoFin works robustly under varied distance and outperforms than vision-based approaches (i.e., 1m).

(3) Impact of Different Hands. We also evaluate the label parsing performance of six labels from different hands captured via VIVO-Y71A at 0.5m under strong ambient light. As shown in Figure 14 (c), The hand #1 and #3 achieve the high prediction accuracy more than 0.96 while the hand #2 achieves the lowest accuracy of 0.77. The reason is the F6-F10 from hand #2 have more confused rolling patterns than hand #1 and #2. Even though, the average label parsing accuracy still achieves 0.91, which demonstrates the effectiveness of our optical labeling and parsing scheme.

(4) Impact of Different Labels. We also present the confusion matrix of the trained label parsing model for 18 different classes of 3 hands under the settings: VIVO-Y71A, 0.5m and strong ambient light in Figure 14 (d). These 18 classes are finger #1 to finger #15 (F1-F15) and the wrist point #1 to the wrist point #3 (W1-W3). It shows the labels from hand #2 are easier to be identified as other labels than hand #1 and #3, which is consistent with the

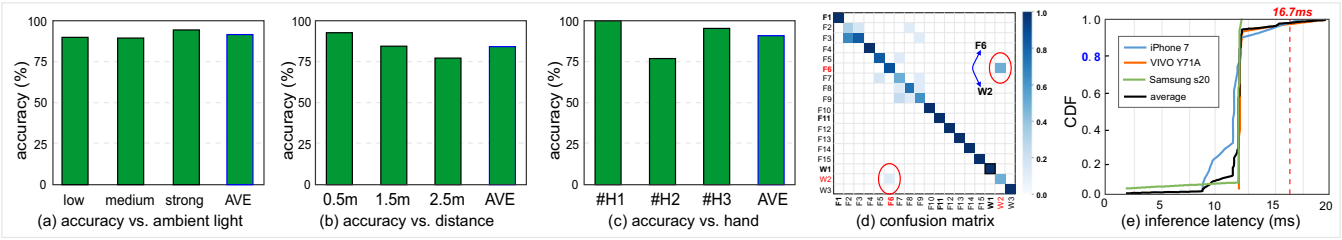


Figure 14: Label parsing accuracy performance in varied settings and latency evaluation.

results in Figure 14 (c). It also shows that the rolling pattern of W2 [CP, On, Off, Off, On, Off] is confused by F6 [CP, Off, Off, On, On, Off]. That is because the reversed rolling patterns of F6 [Off, On, On, Off, Off, CP] (i.e, [..Off, On, On, Off, Off, CP, Off, On, On, Off, Off, CP..]) has the high similarity with the W2 when the amplitude of CP symbol is similar to On symbol.

(5) Impact of Different Cameras. We use the trained model to parse the labels captured by different cameras of commercial smartphones [iPhone 7, VIVO Y71A, and Samsung s20] to measure their label parsing latency performance of hand #1 under strong ambient light at 0.5m with the same resolution of 640×640. As shown in Figure 14 (e), iPhone captured images can be parsed with the shortest time while the average parsing latency of these different cameras are about 12ms (83Hz), and less than the 16.7ms (60 Hz), which demonstrates RoFin achieves the real-time label parsing.

8.2 Enhanced X/Y Tracking and Depth Parsing

8.2.1 Inside-frame X/Y Tracking Performance.

Setup. We bond one fingertips of the RoFin glove with a pen (blue marker) and draw on the transparent plastic paper hanging parallel to the camera's image plane, as shown in Figure 15 (a). We also set two cameras at the fixed distance 0.5m when the user is drawing. One camera follows the traditional vision based approach which captures the video as usual with 60 fps frame rate while the other camera (RoFin reader) captures the video of the rolling patterns with the same 60 fps frame rate but with high rolling shutter rate (**8KHz**). Thus we track 3 traces of user's drawing at the same time: (1) ground truth on the plastic paper, (2) vision approach tracked trace, (3) RoFin tracked trace.

X/Y Tracking Enhancement. We ask a user to draw 3 different letters: (1) M with more straight lines, (2) C with curve, (3) a rotated α with more complex curve with two writing speed: (1) normal speed, and (2) faster speed. We process the captured videos with 6 above setting combinations via PotPlayer software to generate frames for measurement. As shown in Figure 15 (b) and (c), the RoFin tracked 4 times of location points for the same letter, which significantly enhances the granularity of tracking trace in compared to vision-based tracking. Besides, RoFin achieves more accurate trace tracking than vision based method among all three different letters due to its fine-grained inside-frame sampling. We also conduct experiments to figure out the up-sampling ability of RoFin via set higher rolling shutter rate at 12KHz, 16KHz and 20KHz while keep the motion speed the same and high. As shown in Figure 15 (d), with the increased transmission frequency with matched rolling shutter rate, the maximum number of sampled points in one frame increases from 7 sampled points to 8, 10, and 12 points separately because of the decreased width of a rolling strip.

8.2.2 Depth Z Estimation Performance.

Setup. We set a wooden hand model worn RoFin glove at the desk, as shown in Figure 15 (d). The fingertips are separated with different distances to the camera image plane (XY plane). The hand model keeps the same pose but with 3 different orientations to the camera. We also set camera with 3 different rotations to capture the RoFin glove. Then we measure the distance between the fingertips' projected points on the desk to the camera plane as the Z ground truth. We measure all 6 key points in evaluation under 9 different setting combination of orientation and rotation at distance of 0.5m.

Z Estimation Accuracy. As shown in Figure 15 (e), although the error of estimated depth info Z via RoFin (discussed in Section 5.2.2) varies with the different hand orientation and camera rotation, RoFin achieves the average estimation error of 1.6 cm.

Summary. The experiment results in Section 8.2.1 and 8.2.2 demonstrates that our low-cost RoFin provides enhanced X/Y tracking with up-sampling ability of 12 times of vision based approach and accurate Z estimation for depth parsing in 3D space.

8.3 Real-time Hand Pose Reconstruction

We define 10 hand poses as shown in Figure 16 for hand pose reconstruction evaluation. We capture the images of the wooden hand worn the RoFin glove with RoFin reader for different hand poses. Then we run HPR model and evaluate its accuracy and latency with Leap Motion as benchmark.

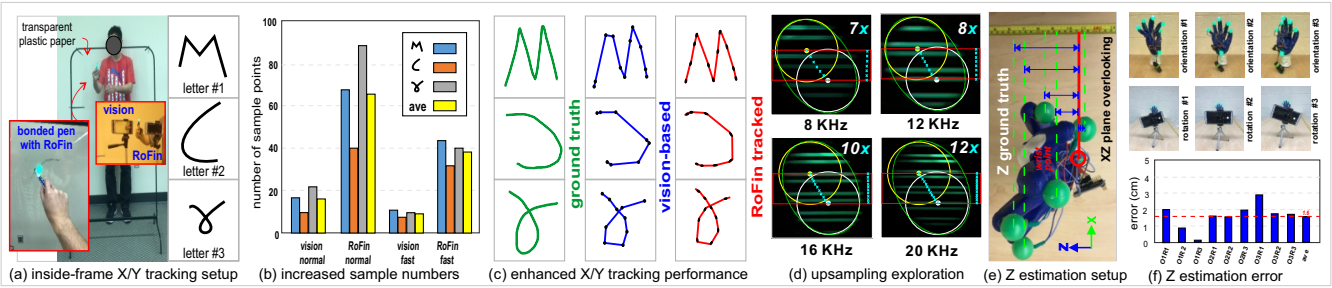


Figure 15: Enhanced inside-frame tracking with up-sampling exploration, and Z estimation performance.

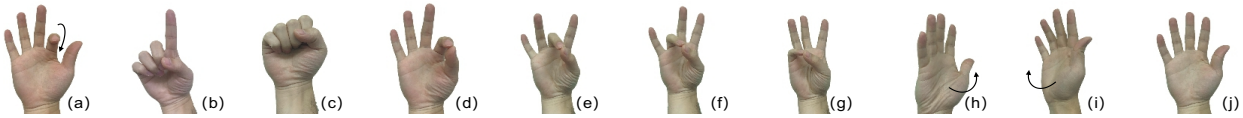


Figure 16: 10 defined hand poses: (a) bend index finger, (b) point with index finger, (c) close the fist, (d-g) pinch thumb with Index, Middle, Ring, and Little finger, (h) turn palm to the left, (i) turn palm to the right, (j) the palm.

8.3.1 Reconstructing Accuracy.

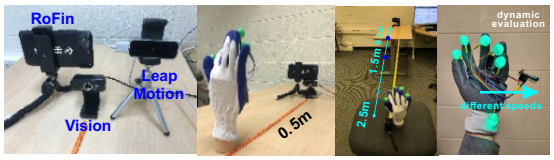


Figure 17: Experiment scenarios.

Impact of Ambient Light. We define the deviation error as the average difference of x,y,z between RoFin with Leap Motion. As shown in Figure 18 (b), the average deviation error of three ambient light settings [low, medium, strong] under 0.5m has the similar distribution and the most deviation error is distributed less than 22 mm. Among three ambient light settings, the medium ambient light achieves the best performance due to the RoFin reader can capture the most clear contours of six key points' spheres.

Impact of Sensing Distance. As shown in Figure 18 (c), the average deviation error of three distances [0.5m, 1.5m, 2.5m] are similar and are mostly distributed in 28 mm. The deviation error of 1.5m achieves the best performance with the average deviation error of 14 mm while the 2.5m setting achieves the largest average deviation error of 19 mm. These results demonstrate our HPR model works well up to 2.5 m while the vision approaches usually work within 1 m and Leap Motion works within 0.5m.

Impact of Different Poses. We also evaluate the reconstructing deviation error of 10 hand poses defined above at 0.5m. As shown in Figure 18 (c), the reconstructed y has the largest deviation error compared with x and z, especially for the hand pose (b), point with index finger. The reason is that the finger planes of the ring finger, the little finger are not exactly as assumed in our simplified HPR model that their finger planes perpendicular to the projected palm plane. Among 10 hand poses, the pose (j) achieves the lowest average deviation error in hand pose reconstructing of 7.6 mm.

Impact of Dynamic Scenarios. Besides the wooden hand based evaluation above, we ask the user to wear RoFin gloves for dynamic scenario evaluation, as shown in Figure 17. The user stands 0.3m far away from RoFin camera and Leap Motion and uses the fixed hand pose shown in the right of Figure 17 with four different motion speeds [static, slow, medium, fast]. As shown in Figure 18 (d), the average deviation error with different motion speeds are similar and less than 4cm. The reason is that the X/Y/Z pair for hand pose reconstruction are based on frame level one-sample parsing instead of multiple-sampled X/Y pairs for finger tracking.

8.3.2 Reconstructing Latency.

As for hand pose reconstructing, the main advantage of RoFin compared with vision-based approaches is its less tracked key points and flexible and long sensing distance. We evaluate the hand pose reconstructing latency and make comparison with the vision based approach Media Pipe ran on the same platform: Thinkpad T480 with Intel(R) Core(TM) i7-8650U CPU for different hand poses under the same 0.5m distance and strong ambient light setting.

As shown in Figure 18 (e), the latency of the RoFin HPR model is distributed less than 21 ms with the average latency of 13.8 ms (72Hz), which is less than 16.7 ms (60Hz). The vision based Media Pipe achieves 47.5 ms latency in average. Although the finger label parsing requires about 12ms for each image frame, the label parsing module and the HPR module can still run in pipe-line manner to achieve the real-time processing. These results demonstrate that our HPR model can achieve real-time hand pose reconstructing due to its only tracking 6 key points with simplified HPR model.

We also measure the end to end latency of entire steps for hand pose reconstruction including (1) finger identification, (2) YOLO based X/Y parsing, (3) Z estimation, and (4) time cost of HPR processing under 4 different motion speed of the same hand pose. (1)-(3) are processed together and the output will be the input of the step (4). As shown in Figure 18 (f), there is no significant difference in the end-to-end time cost among different motion speeds.

8.4 Use Cases

Multi-user interaction for AR/VR/MR. As demonstrated in Section 8.2.1, RoFin can track inside-frame X/Y location samples at rolling shutter rate and thus provide the ability of fine-grained finger tracking, especially for the high-speed motion or small-scale motion. Multiple users can use their fingertips to write or paint virtually at the same time in front of the camera. Thus, RoFin can be used as the user interface with better user experience for AR/VR/MR with privacy protection of users due to they only want the camera to capture the trace instead of the face, as shown in Figure 19 (a).

Virtual Writing or Health Monitoring for Parkinson's Suffers. Our RoFin system can track fine-grained writing trace including the subtle trembling while the vision-based approaches (1Hz

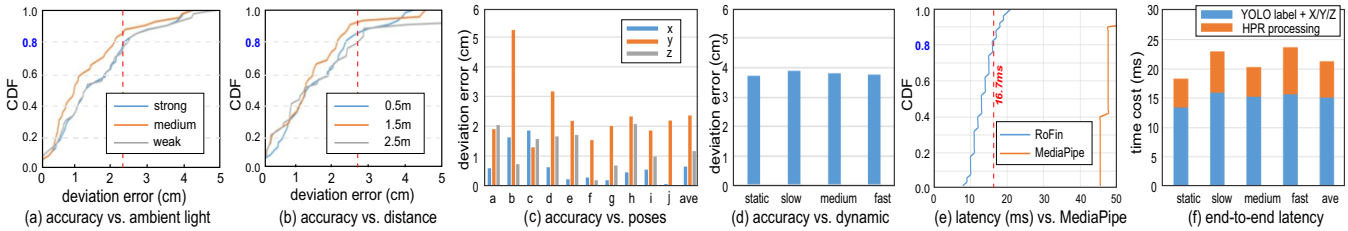
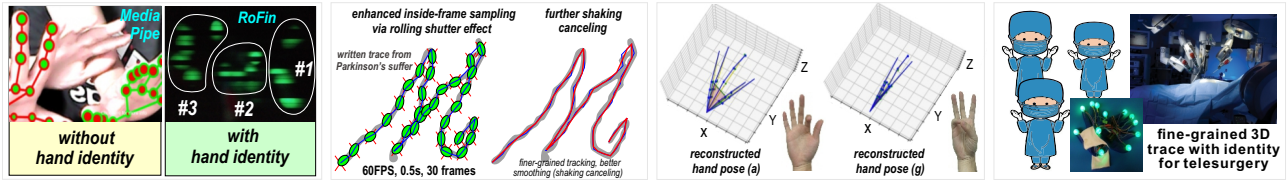


Figure 18: Hand pose reconstructing performance.



(a) multi-user MR interaction (b) fine-grained trampling tracking (c) hand pose commands (d) telesurgery interfaces

Figure 19: 4 possible use cases for our low-cost RoFin: (1) multi-user MR interactions with identification and protected privacy, (2) finer-grained tracking of writing of Parkinson's suffer[2], (3) real-time hand pose commands, and (4) telesurgery interfaces.

inside-frame sampling rate and about 60 fps frame rate) can not track it clearly as human eyes. The Parkinson's suffers can use our RoFin glove to virtually write characters. Then we can use RoFin tracked fine-grained trace to better smooth the trace (e.g, connect the middle points among two trace sub-lines), as shown in Figure 19 (b). Besides, the tracked fine-grained trace can be also utilized as the medical diagnosis and health monitoring.

Hand Pose Commands for Video Games/Smart Home. As demonstrated in Section 8.3, RoFin achieves real-time hand pose reconstructing with less computation overhead and high accuracy. Our low-cost RoFin system can be used as the hand pose command input interface for video games, smart home, etc. Figure 19 (c) shows reconstructed hand pose examples via RoFin's HPR model.

Telesurgery Gloves. There are multiple doctors, nurses and remote operation robots involved in a telesurgery. RoFin can differentiate whose hand it is with the fine-grained real-time finger traces and then send out the 3D traces to the remote Dr. Robots for operations, as shown in Figure 19 (d). Our goal is to sense the fine-grained 3D traces without sensing delay instead of operating surgery with RoFin gloves directly, thus spheres with future optimizations will not be the problems.

9 DISCUSSION

Non-vision based Solutions. There are two types of Non-vision based solutions: (1) on-body sensor based approaches[7, 11, 13], and (2) hand-free approaches[15–17, 24]. Compared with our RoFin, these approaches are limited in: (1) requirement of specific/expensive sensors and devices instead of commercial LED nodes, such as mmWave chips, FBG sensors, (2) sensing distances within the near hand area (i.e., within 0.5m), (3) lack of finger/hand identification and can not serve multiple users with identification, (4) wearable/IMU solutions can sample at a high frequency at the user side, but they must incur data-send-out overhead and latency for centralized control (e.g., in a situation where multiple users are actively engaged), whereas RoFin obtains these data directly at non-user sides without incurring data-send-out overhead.

Comparison with Commercial Products. (1) To precept hands' morphological variation frame by frame, Leap Motion's [3] uses infrared cameras with illuminating LEDs. (2) XSens[4] performs 3D motion capturing made possible by specialized and pricey tiny MEMS inertial sensors. For example, the MOVELLA DOT SENSOR costs $132. (3) Instead of using hand posture reconstruction, Luxapose [12] utilizes many fixed LED landmarks on the ceiling to work with a rolling camera for indoor localization.

Privacy Leakage. Vision approaches (i.e., human eyes, Media Pipe, Leap Motion integrated with camera) can cause privacy leakages. As shown in Figure 20, the user conducts the hand pose command while his/her hand holds a bank card. The Leap Motion and the Media Pipe cause the leakage of sensitive information (i.e., cvv number) which may result in the property loss.

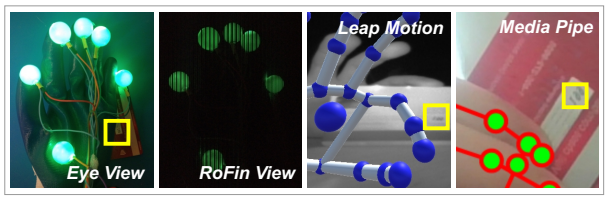


Figure 20: Sensitive data leakage of vision-SOTA.

Power Consumption and Safety. Our RoFin gloves are made of electric insulation rubber gloves, and the voltage at the LED node side is less than 3V, ensuring the safety of users who wear gloves. The current through one RoFin glove's circuit is **75** mA, and the power consumption is **225** mW. Based on our 600mAh and 9V li-ion battery, one RoFin glove can work for approximately 5.4 Wh / 225 mW = **24** hours before needing to be recharged.

Limitation of RoFin. Compared with hand-free approach (e.g., vision approaches), current RoFin prototype requires users to wear gloves attached with plastic spheres with wires and battery. We can overcome this limitation by ergonomic design, textile technique, energy harvesting, passive labeling, etc., in the future.

Future Direction. (1) optimize the spheres and explore backscatter based passive fingertips' labeling. We can decrease the sphere size and exploit energy harvesting techniques for decreased weight and ease to use. (2) update HPR model. We cam improve HPR for hand poses in which finger planes are not perpendicular to the projected palm plane (e.g., hand pose (b)). (3) extend RoFin for body gesture recognition. The core idea of RoFin can be extended for human body gesture reconstructing easily with predictable benefits.

10 CONCLUSION

In this paper, we first exploit the 2D temporal-spatial rolling to construct 3D hand pose. We address technical challenges in RoFin design and implementation, e.g, fingertips active optical labeling, fine-grained 3D information parsing of rolling fingertips, and lightweight 20-joints 3D hand pose reconstructing via 6 tracked key points. Then we undertake studies using RoFin gloves in a variety of circumstances. The results demonstrate RoFin can robustly identify fingers, parse fine-grained 3D info, and achieve real-time hand pose reconstruction. Our RoFin is a low-cost but effective solution for human computer interactions with promising use cases.

ACKNOWLEDGMENTS

This work was partially supported by the U.S. National Science Foundation under Grants CNS-2226888 and CCF-2007159.

REFERENCES

[1] 2019. IEEE Standard for Local and metropolitan area networks–Part 15.7: Short-Range Optical Wireless Communications. *IEEE Std 802.15.7-2018 (Revision of IEEE Std 802.15.7-2011)* (April 2019), 1–407.

[2] 2022. *Micrographia*. https://en.wikipedia.org/wiki/Micrographia_%28handwriting%29

[3] 2023. *Leap Motion Tutorial*. https://developer-archive.leapmotion.com/documentation/cpp/devguide/Sample_Tutorial.html

[4] 2023. *XSens*. https://www.movella.com/

[5] Andreas Aristidou, Joan Lasenby, Yiorgos Chrysanthou, and Ariel Shamir. 2018. Inverse kinematics techniques in computer graphics: A survey. In *Computer graphics forum*, Vol. 37. Wiley Online Library, 35–58.

[6] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. 2018. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 666–682.

[7] Weiya Chen, Chenchen Yu, Chenyu Tu, Zehua Lyu, Jing Tang, Shiqi Ou, Yan Fu, and Zhidong Xue. 2020. A survey on hand pose estimation with wearable sensors and computer-vision-based methods. *Sensors* 20, 4 (2020), 1074.

[8] Minhao Cui, Qing Wang, and Jie Xiong. 2020. Breaking the limitations of visible light communication through its side channel. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 232–244.

[9] Ander Galisteo, Qing Wang, Aniruddha Deshpande, Marco Zuniga, and Domenico Giustiniano. 2017. Follow that light: Leveraging leds for relative two-dimensional localization. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*. 187–198.

[10] Jun Gong, Yang Zhang, Xia Zhou, and Xing-Dong Yang. 2017. Pyro: Thumb-tip gesture recognition using pyroelectric infrared sensing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. 553–563.

[11] Jun Sik Kim, Byung Kook Kim, Minsu Jang, Kyumin Kang, Dae Eun Kim, Byeong-Kwon Ju, and Jinseok Kim. 2020. Wearable hand module and real-time tracking algorithms for measuring finger joint angles of different hand sizes with high accuracy using FBG strain sensor. *Sensors* 20, 7 (2020), 1921.

[12] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. 2014. Luxapose: Indoor positioning with mobile phones and visible light. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. 447–458.

[13] Yongjun Lee, Myungsin Kim, Yongseok Lee, Junghan Kwon, Yong-Lae Park, and Dongjun Lee. 2018. Wearable finger tracking and cutaneous haptic interface with soft sensors for multi-fingered virtual manipulation. *IEEE/ASME Transactions on Mechatronics* 24, 1 (2018), 67–77.

[14] Rui Li, Zhenyu Liu, and Jianrong Tan. 2019. A survey on 3D hand pose estimation: Cameras, methods, and datasets. *Pattern Recognition* 93 (2019), 251–272.

[15] Tianxing Li, Xi Xiong, Yifei Xie, George Hito, Xing-Dong Yang, and Xia Zhou. 2017. Reconstructing hand poses using visible light. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–20.

[16] Jaime Lien, Nicholas Gillian, M Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–19.

[17] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1515–1525.

[18] Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2021. Sign language recognition: A deep survey. *Expert Systems with Applications* 164 (2021), 113794.

[19] Sanjib Sur, Ioannis Pefkianakis, Xinyu Zhang, and Kyu-Han Kim. 2018. Towards scalable and ubiquitous millimeter-wave wireless networks. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 257–271.

[20] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. 2015. Robust articulated-icp for real-time hand tracking. In *Computer graphics forum*, Vol. 34. Wiley Online Library, 101–114.

[21] Mary Ann Thenganatt and Elan D Louis. 2012. Distinguishing essential tremor from Parkinson's disease: bedside tests and laboratory evaluations. *Expert review of neurotherapeutics* 12, 6 (2012), 687–696.

[22] Robert Wang, Sylvain Paris, and Jovan Popović. 2011. 6D hands: markerless hand-tracking for computer aided design. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. 549–558.

[23] Yue Wu, Purui Wang, Kenuo Xu, Lilei Feng, and Chenren Xu. 2020. Turboboosting visible light backscatter communication. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 186–197.

[24] Huichuan Xu, Daisuke Iwai, Shinsaku Hiura, and Kosuke Sato. 2006. User interface by virtual shadow projection. In *2006 SICE-ICASE International Joint Conference*. IEEE, 4814–4817.

[25] Y. Yang, J. Hao, and J. Luo. 2017. CeilingTalk: Lightweight Indoor Broadcast Through LED-Camera Communication. *IEEE Transactions on Mobile Computing* 16, 12 (2017), 3308–3319.

[26] Yanbing Yang and Jun Luo. 2018. Boosting the throughput of LED-camera VLC via composite light emission. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 315–323.

[27] Yanbing Yang and Jun Luo. 2020. Composite Amplitude-Shift Keying for Effective LED-Camera VLC. *IEEE Transactions on Mobile Computing* 19, 03 (2020), 528–539.

[28] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214* (2020).

[29] Xiao Zhang, Hanqing Guo, James Mariani, and Li Xiao. 2022. U-Star: An Underwater Navigation System based on Passive 3D Optical Identification Tags. In *The 28th Annual International Conference on Mobile Computing and Networking*.

[30] Xiao Zhang, Griffin Klevering, Xinyu Lei, Yiwen Hu, Li Xiao, and Tu Guanhua. 2023. The Security in Optical Wireless Communication: A Survey. *ACM Computing Surveys (CSUR)* (2023).

[31] Xiao Zhang, Griffin Klevering, Kanishka Wijewardena, and Li Xiao. 2023. Demo: Integrated On-site Localization and Optical Camera Communication for Drones. In *2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 1–3.

[32] Xiao Zhang, Griffin Klevering, and Li Xiao. 2022. Exploring Rolling Shutter Effect for Motion Tracking with Objective Identification. In *Proceedings of the Twentieth ACM Conference on Embedded Networked Sensor Systems*. 816–817.

[33] Xiao Zhang, Griffin Klevering, and Li Xiao. 2023. PoseFly: On-site Pose Parsing of Swarming Drones via 4-in-1 Optical Camera Communication. In *2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 1–10.

[34] Xiao Zhang, James Mariani, Li Xiao, and Matt W Mutka. 2022. LiFOD: Lighting Extra Data via Fine-grained OWC Dimming. In *2022 19th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 73–81.

[35] Xiao Zhang and Li Xiao. 2020. Lighting Extra Data via OWC Dimming. In *Proceedings of the Student Workshop*. 29–30.

[36] Xiao Zhang and Li Xiao. 2020. RainbowRow: Fast Optical Camera Communication. In *2020 IEEE 28th International Conference on Network Protocols (ICNP)*. IEEE, 1–6.

[37] Run Zhao, Dong Wang, Qian Zhang, Xueyi Jin, and Ke Liu. 2021. Smartphone-based Handwritten Signature Verification using Acoustic Signals. *Proceedings of the ACM on Human-Computer Interaction* 5, ISS (2021), 1–26.

[38] Shilin Zhu, Chi Zhang, and Xinyu Zhang. 2017. LiShield: Create a Capture-Resistant Environment Against Photographing. In *Proceedings of the 9th ACM Workshop on Wireless of the Students, by the Students, and for the Students*. 23–23.