PoseFly: On-site Pose Parsing of Swarming Drones via 4-in-1 Optical Camera Communication

Xiao Zhang, Griffin Klevering, Li Xiao
Michigan State University, MI, East Lansing
{zhan1387, kleveri2}@msu.edu, lxiao@cse.msu.edu

***Abstract*—Drones are gaining more interest from the industry and the research community as a result of their many advantages, including low cost, small size, adaptability, and ease of use, as well as their potential applications. However, current control of swarming drones relies on stand-alone modes and centralized radio frequency control from a base station on the ground which is devoid of drone-to-drone communication. This method has drawbacks, including a crowded RF spectrum with mutual interference, high latency, and a lack of on-site drone-to-drone interactions. Because of its high spatial multiplexing capability, Line of Sight (LoS) security capabilities, broader bandwidth, and intuitive vision manner, Optical Camera Communication (OCC) is considered to be a potential alternative for sensing and communication in drone clusters. In this paper, we first utilize the rolling shutter effect in drone sensing and communication and propose *PoseFly*, a 4-in-1 AI-assisted OCC with drone identification, on-site localization, quick-link communication and lighting. We implement PoseFly prototypes on commercial drones, cameras and LEDs. Our experiments show our PoseFly achieves nearly 100% accuracy for distance estimation (20m), drone identification (12m), angle and speed estimation (4m) and 5 Kbps average quick-link throughput at up to 4 m on current prototypes.**

I. Introduction

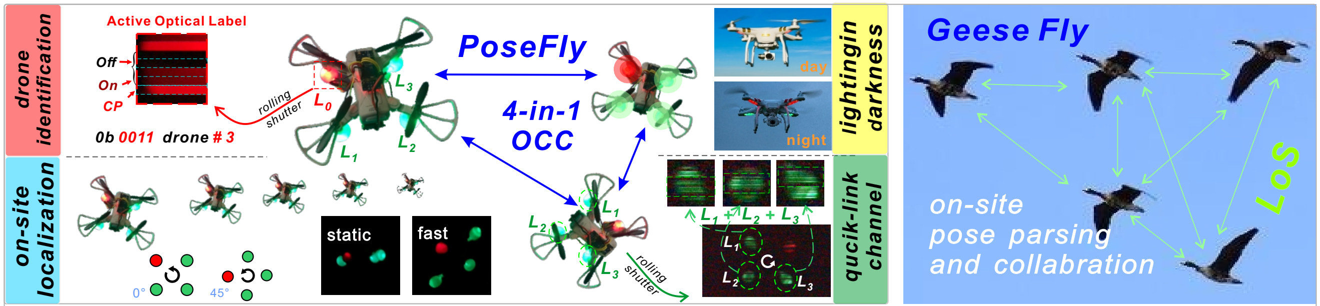
Drones are a type of unmanned aerial vehicle (UAV), that attract more attention because of their advantages over manned aircraft, including small size, low cost, simplicity of operation, and broad potential applications [1], [2]. Drones are now used in a variety of fields, such as aerial photography, plant protection, express deliveries, transportation, animal monitoring, surveying and mapping, power inspection, disaster relief, news reporting, selfies, film and television production etc. Drones are projected to play significant roles in integrative development for sensing, communication, and computing in the near future due to ongoing advances in Artificial Intelligence and their superior mobility. According to Verified Market Research, the size of the global drones market, which was expected to be worth USD 19.23 billion in 2020, would increase to USD 63.05 billion by 2028 with a Compound Annual Growth Rate (CAGR) of 16.01 percent between 2021 and 2028 [3].

Currently, drones are mostly controlled by a centralized base station (CBS), such as a drone pilot on the ground or a satellite in orbit, using the radio frequency (RF) spectrum [4], [5]. These centralized controlling techniques would, however, restrict the use cases for drones due to their lack of mutual communication among drones, such as on-site sharing data directly without the assistance from the centralized base. This is due to the requirement that each drone in the drone cluster acquires the command from the CBS and then transmits its status, including its surroundings and posture state as measured by its inner sensors such as the IMU (Inertial Measurement Unit), back to the CBS. Thus, the back-and-forth communication latency caused by the centralized drone controlling mechanism, particularly in high motion scenarios, might result in significant localization errors. For instance, the 0.25s location computation and communication cost for two drones moving at 20m/s in opposing directions will result in a 10m localization mistake ($0.25 \times 20 \times 2$). Moreover, as drones amount increases, the constrained capacity of the RF spectrum gets significantly more crowded, which may result in bit errors during re-transmissions and further localization errors.

The transmission between drones and CBS in centralized control can naturally be avoided by on-site interactions among drones in a distributed manner. We could use RF to establish distributed drone-to-drone communication. However, due to Non-Line-of-Sight (NLoS) propagation, eavesdroppers can easily detect RF signals, and there is nontrivial multi-path effects and caused mutual interference [4]–[6]. Even though there is no back-and-forth communication cost between drones and the CBS in RF based distributed drone-to-drone communication, the growing drone population may cause the RF spectrum to become crowded, which could lead to more localization errors owing to re-transmission and lag.

There are two main issues for localization of drones with high mobility: **(1)** computing a drone's appropriate localization information including distance, posture, speed, and so on; and **(2)** promptly receiving the computed localization information. Actually, we can use on-site posture features of a drone (transmitter) and compute at the receiving side (another drone) instead of computing at the transmitter's side IMU to reduce transmission overhead. For instance, when a flock of geese are flying together, goose A (receiver) observes goose B (transmitter) and processes B's posture features in A's brain rather than goose B computing its own position and notifying A, as shown in Figure 1 (b).

Optical Camera Communication (**OCC**) has attracted more attention due to the popularity of commodity mobile devices with built-in cameras and its low interference with ambient light compared to photodiode based techniques such as LiFi. OCC also provides location-based services (LBS) such as fine-grain AR navigation with the association of data from a visible transmitter in a flexible communication range [6]–[15].



(a) PoseFly enables 4 integrated functions (b) geese fly scenario

Fig. 1: PoseFly: 4-in-1 OCC for swarming drones, similar to geese flying and their relative localization and collaboration.

Motivation: (1) Current control of swarming drones relies on stand-alone modes and centralized radio frequency control from a base station on the ground which faces severe interference when the number of drones increases. **(2)** Existing drone localization includes drawbacks such as requiring communication time cost to send out the computed localization data, which may cause non-trivial localization errors due to latency and the high mobility of drones. **(3)** Commercial drones are more suited for optical camera communication among drones since they are equipped with cameras for photographing and LEDs for indicating and lighting. **(4)** Finally, drones can share their computed on-site postures to other drones via secure OCC quick-link for collaboration.

Our Approach: In this paper, we propose **PoseFly**, which first exploits 2D spatial diversity of rolling shutter cameras in on-site drone positioning. PoseFly is a 4-in-1 cheap and efficient approach for high-capacity drone identification, on-site pose parsing for drones, quick-link communication, and lighting, as shown in Figure 1 (a). PoseFly requires only 4 cheap, single-color LED components with plastic covers (One is red, three are green) controlled by an Arduino Nano MCU (<\$10) and a commercial camera mounted on the drone. The red-colored LED mounted on the drone's left-front corner emits distinct cyclic OOK (On-Off Keying) waves that serve as its unique invisible optical identification label. Therefore, drones can reliably identify other drones via camera. When paired with the other three green LEDs, the red led can also be utilized as a positioning element. Based on the variations in the geometry of the quadrangle generated by these four LEDs, PoseFly can parse the drones' poses and share such data among other drones via OCC quick links.

There are three main **technical challenges**, as shown in Figure 2 and illustrated below. **C1**: Robust identification in long distance for drones. It is not as easy for drones to recognize other drones with similar appearances via vision recognition as it is for geese. Instead, we can attach optical marks or labels on drones. However, static marks or existing bar/QR codes are passive, they can only work within limited recognition distance such as 1 m. **C2**: Lightweight but precise localization (distance, speed, angle). Geese can sense the posture of other geese via many vision features such as the head, wings and feet, etc. If we sense the drone posture with the same method, it will introduce non-trivial computation overhead. **C3**: Decoding asynchronized rolling strips in rolling spots with random locations in a frame. Generated rolling strips in each rolling spot are not synchronized for decoding with flying drones.

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

(1) This is the first work to exploit rolling patterns for on-site drone posture parsing, including relative distance, speed and angle estimation, which was solely used for optical camera communication before.

(2) We thoroughly investigate the spatial rolling patterns and design the 4-in-1 PoseFly, an AI-assisted approach for drone identification, drone localization, drone communication, and lighting with commercial LEDs and cameras.

(3) We address challenges via cyclic pilots and OOK for active optical labeling and robust quick-link communication. We adopt CNN models for accurate and robust identification, localization at the receiver side.

(4) We evaluate PoseFly on our implemented prototypes in both day and night with varying distance and motion speed. Experiment results show that PoseFly can identify drones with nearly 100% accuracy within 12 m while providing accurate pose parsing (100% distance estimation within 20 m, 100% speed and angle estimation within 4 m). Besides, PoseFly provides 5 Kbps quick-link channel up to 4 m on average.

The rest of the paper is organized as follows: Section II introduces background and related work. Section III gives the system overview. Sections IV-VI illustrate 3 functions except PoseFly's lighting function: *Drone Identification*, *Drone Localization*, and *Drone Quick-Link*. Section VII presents PoseFly implementation. Section VIII reports the performance evaluation of PoseFly. Finally, we have some discussion and conclude the paper in Section IX and Section X.

II. Background and Related Work

A. Drone Identification

Vision based methods could be used to identify drones. For example, a camera can take an image of a drone and identify it based on its shape and features. The reader then uses the gray-scale image of the scene and detects the drone based on its silhouette [16]. However, these systems cannot work well at night, as the captured images of drones are not clear enough, nor do they work at longer distances. RF systems can identify drones in a few ways: Drones typically communicate at a

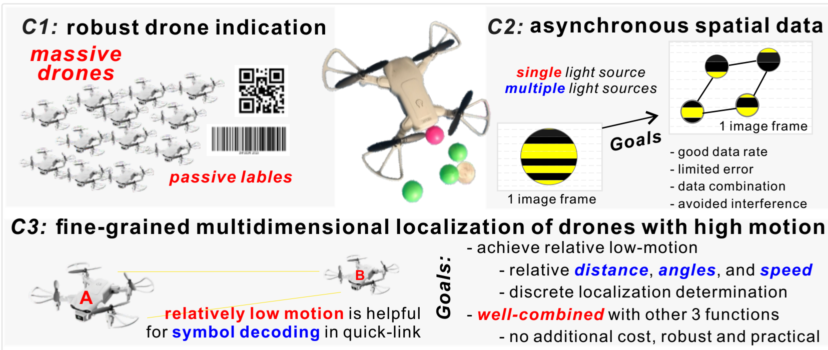


Fig. 2: Three main challenges in PoseFly: robust drone indication, asynchronous spatial data combination, and localization under high motion.

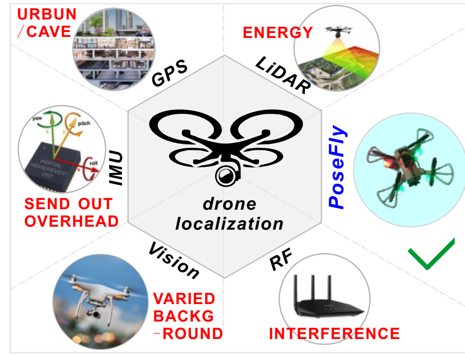


Fig. 3: Drone localization approaches: GPS, IMU, vision, RF, LiDAR, and PoseFly.

much higher frequency than other mobile devices. If the RF connection is monitored, the used frequency could be utilized to determine if a device is a drone or not. However, other wireless devices could communicate at the same frequency and thus it couldresult in a mis-identification [17]. Instead of the clear images with complete morphology needed by computer vision or confused RF spectrum indication, PoseFly simply requires one active LED node which holds the indication sequences and can work well in both day and night.

B. Drone Localization

We present the related work of drone localization below and illustrated in Figure 3. **(1)** Current **RF**-based drone localization methods are based on received signal strength (i.e., RSSI, CSI) or time difference of arrival. By monitoring the signal strength of an emitter or the change in time of its arrival, a receiver could determine the direction and speed of the drone. However, interference in the path can corrupt the localization results [18]. **(2) Vision** based localization approaches use cameras to record several frames of a scene, then detect a drone and calculate its velocity and future position [19]. While this is certainly effective, it has non-trivial processing overhead, especially for image processing of morphology with varied background when the drone is flying. **(3) IMU.** Drones can also measure their own localization date (e.g., position, and velocity) via an inertial measurement unit (IMU) and send them out to other drones. However, these messages would need to be sent constantly and received over long distances. Thus, the IMU based methods have non-trivial send-out communication overhead and time delay, especially when there are numerous drones with severe interference [20]. **(4)** Although **GPS** systems can provide accurate location information, they also have send-out cost and cannot work well in urban areas, caves, tunnels, etc. **(5) LiDAR** systems can provide on-site localization of nearby drones. However, laser generation and scan processing cost non-trivial energy [21].

In contrast to the above mentioned drone localization approaches, PoseFly only requires one frame image to determine velocity and orientation. PoseFly uses 4 LED nodes to illustrate which direction the drone is facing, allowing orientation to be found. Velocity can also be found through the orbs, as the faster the drone moves, the more the orbs will deform in one direction. It is free from interference from multiple drones thanks to the spatial diversity of millions of pixels from the camera to capture them into different image zones. The illuminated balls allow PoseFly to work during day and night over flexible distances. Considering these energy efficient LED balls also provide lighting function, PoseFly is a green localization approach. Moreover, the localization of PoseFly does not have a send-out cost as the reader captures the drone's image (the light propagates at high speed of 3×10^8 m/s) and then processes it locally. Besides, PoseFly's on-site localization only relies on the drones themselves and thus can work in caves/tunnels where GPS can not work.

C. Drone Communication

Today, most drones communicate via the radio frequency medium. RF signals can travel over relatively long distances. However, RF systems can be prone to eavesdroppers, jammers, and interference [22]. The RF signal is sent through the open space and anybody can listen or send their own confounding signals. PoseFly utilizes Line-of-Sight propagation and thus the signals from the swarming drones can be blocked out from attackers which makes it more secure than RF-based communication. Similarly, jammers must send more light directly into the receiver to jam the camera.

III. System Overview

Our proposed 4-in-1 optical camera communication, **PoseFly**, is composed of two parts, as illustrated in Figure 4: (1) commercial LED based PoseFly Transmitter, (2) AI-assisted commercial camera based PoseFly Reader. One drone can equip both transmitter and receiver as a transceiver.

PoseFly transmitter: PoseFly transmitter consists of 4 commercial low-power LED components attached on each corner of a four-rotor drone. Of these 4 LEDs, one is red while the others are green, and are covered with plastic balls of the same color and controlled by an Arduino Nano.

PoseFly receiver/reader. PoseFly reader is based on commercial cameras, which can be the mounted cameras on the drones. These cameras use adjustable focal length lenses and configurable rolling shutter rates and frame rate.

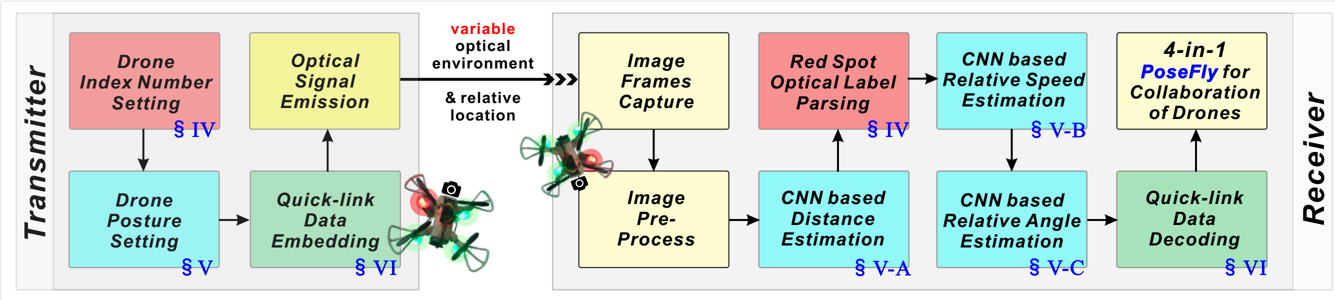


Fig. 4: The system overview including the transmitter and receiver, and the workflow of PoseFly.

4-in-1 Illustration: (1) Drone identification: The red LED generates OOK waves with cyclic pilots to indicate the index of a drone in the drone cluster. For example, the OOK wave [on, off, off, on] indicates the index of the drone is 0b1001, which is # 9. **(2) Drone on-site localization**: The PoseFly reader can `estimate distance` from the transmitter to the reader based on the size of captured four LEDs. Furthermore, the reader can conduct `on-site angle parsing` based on generated shape and color pattern of four LEDs. Additionally, the shape of the rolling spot varies from normal circle to ellipse with different motion speed of drones, which can help the reader to conduct `speed estimation`. **(3) Drone quick-link**: At the same time, the other three green LEDs create the quick-link channel among nearby drones by fast on-off switching. **(4) Lighting**: These LED components provide a lighting function in a dark environment or at night.

Workflow: As shown in Figure 4, these four functions are achieved at different distances between two drones step by step. **(1)** Firstly, when a drone, Drone A, notices there is a bright spot, which is another drone, Drone B, based on B's **lighting function** in long distance (>20m) via its camera. **(2)** Then Drone A will fly closer to B based on its **distance estimation** (<20m) function and conduct the **drone identification** (<12 m) to know the index number of Drone B in the cluster of drones. **(3)** Later, Drone A flies closer to B and performs finer-grained localization of B such as the estimation of **motion speed and posture angle** of B. **(4)** When these two drones require mutual data sharing, they can fly closer within 4 m and utilize the **quick-link** channel to share information such as fly instructions, on-site posture info of other drones.

IV. Drone Identification

For drone interactions, drone detection is critical. However, current optical labels like barcodes and QR codes are passive and only function at close ranges of a few centimeters. To overcome this limitation, we design active optical labels for drone identification in long distance (up to 12 m). We present our active optical label design at transmitter side and the CNN based robust label parsing solution below.

A. High-capacity Optical Labeling

Rolling Shutter strip Effect. The global shutter exposes the entire scene at once. The rolling shutter in commercial CMOS cameras, in contrast, expose only one row of pixels while concurrently creating an entire image row by row. Figure 5 illustrates the rolling shutter strip effect, which happens when the rolling shutter speed and the switching speed of the light wave from the transmitter are about equal. Thus, temporal optical signals carrying transmitted data during symbol periods can be successively collected as rolling strips.

CP-OOK Label Wave Design. In PoseFly, each drone is identified by an optical label that regularly emits distinct amplitude waves that are invisible to human eyes (the On-Off switching rate is too high in terms of the KHz frequency to be sensed by human eyes [11], [14], [23]). The optical label is comprised of two components: **(1) CP (cyclic pilots)**, which begin with one symbol period with an adjustable symbol period (strip width) and is used to distinguish an entire optical label, and **(2) indication symbols**, which are made up of four (or more) OOK (On-Off Keying) symbols. There are two amplitude levels besides darkness in the Off symbol, generated by PWM (pulse width modulation) control: the On symbol has a lower brightness than the CP symbol while the CP symbol has the highest brightness.

High Indication Capacity. We embed a drone's binary index into OOK indication symbols. The binary number is *1001* when the drone index is 9 with indication symbols of [*On*, *Off*, *Off*, *On*]. The amount of drones in the drone cluster determines how long the indication symbols are. 4 OOK symbols can indicate up to 16 drones. In general, N OOK symbols can represent 2^N numbers for 2^N drones, which is promising for high-capacity indication and identification of drone swarms. Although some drones may be very close and appear in the FOV of the camera at the same time, different optical labels can notify the observing drone who they are.

B. CNN based Robust Label Parsing

Traditionally, the amplitude threshold was used to decode these optical labels. But it is difficult to configure the threshold dynamically due to the drones' nonlinear movement, long distance and the dynamic optical environment. For the following reasons, we adopt Convolutional Neural Network (CNN)-based label parsing in PoseFly to avoid the complexity and decoding overhead: (1) Online identification and offline training can reduce latency for real-time drone label parsing; (2) the CNN model can learn the features in the repeated dark and bright rolling strips even in conditions where it is difficult to distinguish the amplitude of CP and On.

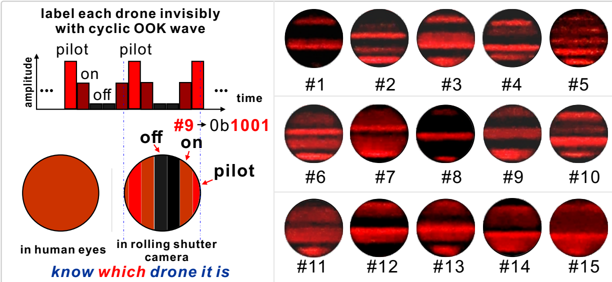


Fig. 5: Rolling strip effect and cyclic CP-based active optical label design: 4 OOK symbols denote up to 16 drones.

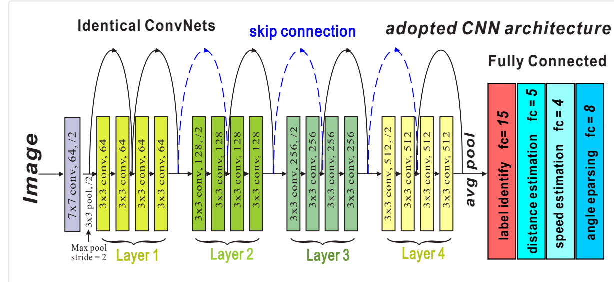


Fig. 6: Adopted CNN networks in PoseFly: ResNet-18 with modified fully connected layers.

We capture real images of optical labels from 15 drones at various distances during day and night to use as training data. The CNN models adopted in PoseFly shown in Figure 6 use the ResNet-18 architecture. They are the Drone Identification Model (**DIM**), Distance Estimation Model (**DEM**), Speed Estimation Model (**SEM**), and Angle Parsing Model models (**APM**). PoseFly has demonstrated exceptional performance on image classification tasks , which is extremely appropriate for our objective of identifying rolling strip patterns and the created shape with color patterns in Section V. The last fully connected layer's output feature is modified to meet the number of options (e.g., 15 in DIM, 5 in DEM, 4 in SEM, and 8 in APM) while keeping other layers the same.

V. Drone Localization

The on-site drone localization (pose parsing) in our proposed PoseFly consists of three parts: (1) distance estimation, (2) relative speed estimation, and (3) on-site angle parsing. We present challenges and design details below.

A. Relative Distance Estimation

For drone localization, the perception and estimation of distance is very important for the interactions among flying drones. For example, accurate estimation of distance between two drones can help avoid unexpected collisions and keep the specific flight formations similar to geese flying for complex collaboration and tasks. The quadrangle generated by the four LED spots in our PoseFly transmitter can give another drone a rough estimation of the distance between themselves. We use the rough size of the captured quadrangle of drone to infer the current relative distance between two drones.

As shown in the bottom of Figure 7, we can estimate the distance based on the captured drone size because the drone size increases when the drone is getting closer to the other drone due to the spatial perspective principle. We first collect the captured images (camera is set with fixed focal length) at different distances and use this data set to train the CNN model for classification offline. Then we can use the trained CNN model to predict and estimate the current relative distance between two drones in real-time.

To filter out the strong ambient light and emphasize the 4 colored spots, we set the rolling shutter with a high shutter speed such as 4000 Hz in our experiments. In our current version of PoseFly, we set 5 distances: 4 m, 8 m, 12 m, 16 m, and 20 m. The captured quadrangles in day and night with random poses are shown in Figure 12 (c).

B. Relative Speed Estimation

The same as distance estimation, the drone speed is critical for drones' collaboration and accident avoidance. In PoseFly, we exploit our discovered relation among motion speed and the varied shape of the spot generated by one of four LEDs.

First, we explore the relationship between different motion speeds and the captured spot shape at the same distance between the camera and the light source. We set different motion speeds for the light source to simulate the drone's different motion speeds and capture the shape of generated spot. As shown in Figure 8, we set 4 levels of movement speed of the light source (i.e, static, low, medium, and fast) and move the light source within the same movement path (↗). Without movement in the front and back direction, the shape of captured rolling patterns changes. As the speed of

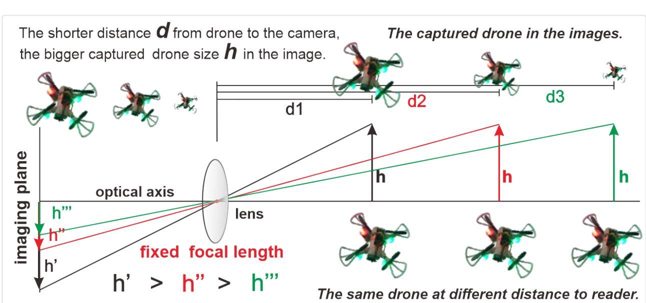


Fig. 7: Distance estimation via perspective principle: the longer the distance, the smaller the captured drone size.

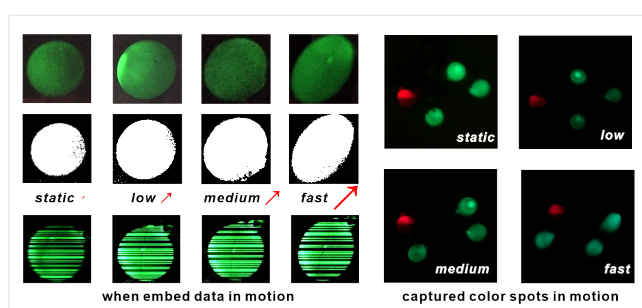


Fig. 8: Relations with motion speed and varied spot shapes: the greater the speed, the larger the spot shape variation.

the light source increases, the shape morphs from a circle to an oval with speed, so does the length of the ellipse's long axis for both light sources with and without embedding data.

In PoseFly, we captured images of the shapes of each spot generated by four LEDs speed estimation within 4 m. To make the SEM more robust, we capture these images during day and night with 4 different motion speeds with random moving paths and used these images as a training dataset for SEM.

C. Relative Angle Parsing

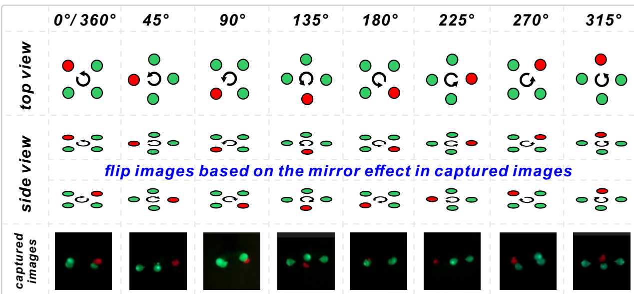


Fig. 9: On-site angle parsing via colored-arc variation.

We model the drone as a rigid body and use the plane generated by the four LEDs to denote the bottom plane of the drone. The red LED is mounted at the left-front corner of a drone and it can be treated as the positioning element to denote the facing angle of the drone.

As shown in Figure 9, we define the relative angle as $0°$ when the camera captures a drone's tail end. Then the captured red spot is rotated $45°$ in the clockwise direction. Using the same rule, we define 8 relative angle statues in total: [$0°$ or $360°$, $45°$, $90°$, $135°$, $180°$, $225°$, $270°$, and $315°$]. Naturally, we can determine the relative angle of the captured drone based on the position of the red spot in the color arc in images. However, due to the small size of LED spots in captured images, it is hard to judge the relative angle. Thus, we employ CNN models to learn the relative angle features offline and then predict the relative angle in the captured image in real-time, similar to the AI method used in previous optical label parsing, distance estimation, and relative speed estimation.

Similarly, we set high rolling shutter speed to avoid the ambient light when we capture the images of color arcs. The captured images for training at 4 m in day and night are shown in the bottom of Figure 9.

VI. Drone Quick-Link

The sensed postures of nearby drones can be stored locally for the usage of the drone itself. At the same time, this posture information can also be shared with nearby drones to extend the communication ranges by using some drones as the relay nodes. Thus, even if some drones are far away or blocked by other drones due to LoS (line-of-sight), they can still communicate with each other. To achieve this goal, we design a quick-link channel for data sharing and communication and present the details of the PoseFly quick-link below.

A. Modulation Design

Quick-link is one type of OCC, which provides a data sharing ability for a small amount of burst data [23]. In PoseFly, we design quick-link to provide a robust optical channel with a similar data rate level (hundreds of bps to several Kbps) besides other 3 functions synchronously. The challenge here is that the captured 3 green spots are randomly located in a captured frame due to the high speed of motion of the drone and variation among frames. Thus, even though we successfully recorded the data in one of the three green spots, we are unable to identify which spot it is and cannot eventually complete the correct decoding. Furthermore, in contrast to optical labels in Section IV-A, if we adopt PWM and use amplitude shift keying, it will sacrifice the transmission bandwidth and decrease data rate significantly.

In PoseFly, firstly we can determine which green spot (i.e., L_1, L_2, or L_3) based on the colored arc in the captured image. For the modulation in each green spot, we design CP (cyclic preamble) based on cyclic OOK data sequences with only bright and dark amplitude levels for a robust quick link. The CP takes the same duration with the CP in optical labels illustrated in Section IV. The CP in green spots are dark strips with adjustable width. The symbol length of OOK data sequences is set as 32 bits while setting the beginning symbol and the end symbol as On as gaps between CP and valid data symbols shown in Figure 10. The data sequence may contain the same length of dark strips as the CP which may make it hard to recognize the CP during rolling strips. Nevertheless, we can set the CP to have a long symbol length to prevent this from happening to confuse decoding. For example, if we set CP with 8 symbol periods, the possibility of the inside data sequence containing 8 continuous Off symbols is (30-8) / C_{30}^{8} $\approx$ 4x10^{-6}, which is low enough for potential conflicts. Thus,

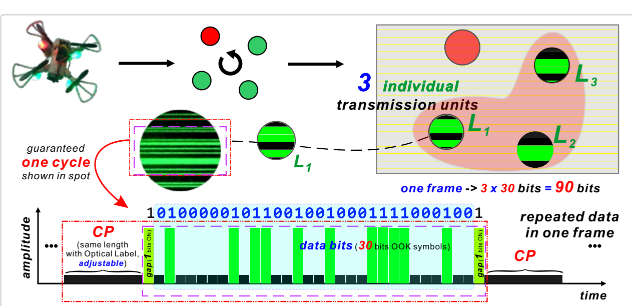


Fig. 10: Quick link modulation design in PoseFly.

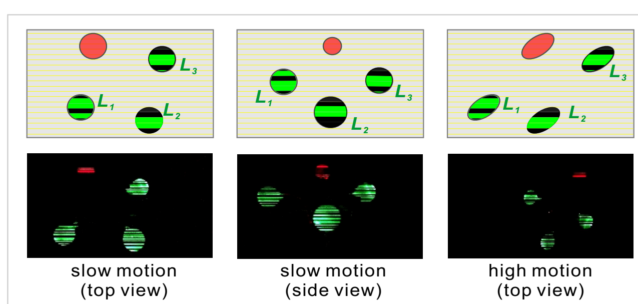


Fig. 11: Robust symbol detection when drones are flying.

we set the CP as 8 continuous Off symbols. The data amount embedded in each spot depends on how many rolling strips are in it and the total data amount in one image frame is the sum of number of strips in all three spots. In each frame, we embed the different data into three green LEDs and choose a proper symbol duration of OOK and CP to guarantee there is over one entire cyclic CP and data sequence in one spot.

To robustly detect the data symbols between CP, PoseFly performs quick link communication within 4 m. As shown in Figure 11, whatever the position of the three spots is in a captured frame with different motion, the strips are clear. So, using the three transmission units that were recorded in each frame, we could collect the data from each green spot and then reconstruct the bit stream. Finally, the data is transferred via the quick link provided by PoseFly, frame by frame.

In our prototype, each image frame embeds 30×3 (the number of spots) = 90 valid OOK data symbols (i.e., 90 bits). The camera frame rate is set as 60 frame per second. The quick link in our proposed PoseFly can achieve the 60×90 = 5400 bits per second data rate, which is 5.4 Kbps, enough for quick link communication among drones to send commands, urgent messages, pose information of drones etc.

VII. System Implementation

A. Transmitter

We implement the PoseFly transmitter prototype for experiments as shown in Figure 12. The main components in one PoseFly prototype are shown in Table I: entry-level drone, 1 Arduino Nano MCU, 1 red and 3 green LEDs wrapped with 1 red and 3 green plastic balls (ϕ = 19mm). The total weight of added components in PoseFly except the drone is **25g** (we use the battery of the drone itself for powering the Arduino Nano) while the total price except the drone is only about **12**$.

B. Receiver

There are numerous commercial smart devices that can be used as the PoseFly reader in our prototype. As shown in Figure 12 (b), these commercial camera devices are widely available and reasonably priced such as VIVO Y71A, and the iPhone 7 we used. To extend the distance for usage of PoseFly, we use a commercial portable lens for smartphone photographing, the price of the lens we used is about 5$. This universal 20x lens can capture the clear images of objects in long distance. In real use scenarios, PoseFly receivers are the mounted cameras similar to cameras in our prototype.

TABLE I: Components in PoseFly.

Component	Price ($)	Details
entry-level drone	40	size: 14cm x 14cm, 125g
Arduino Nano	10	ATmega328P, 5V, 16M
LED	0.1	5mm, green/red, 20000mcd, 20mA
plastic cover	0.3	19mm, green, lightweight
portable lens	$\approx$ 8	Bostionye 20x mobile lens
Total price	$<$ 60	mass produced, cheaper the price

C. Setup

Drone size. The drone used in our prototypes is tiny sized: 14cm$\times$14cm. In the future, we can equip PoseFly on bigger drones (e.g., 1m$\times$1m) to have better performance such as longer distance and higher data rate because of stronger LED power and higher number of strips shown in LED spot.

Different optical environment. Figure 12 (c) shows the scenarios of our implemented PoseFly transmitter flying in two environments (day and night). Figure 12 (c) also shows the experiment scenarios in day and night with different distance.

Simulate the drone flying. In our experiments, we hold the drone in hand or hang it on a hanger and simulate flight with different distances, angles, and speeds to the PoseFly receiver (smartphone) in day and night.

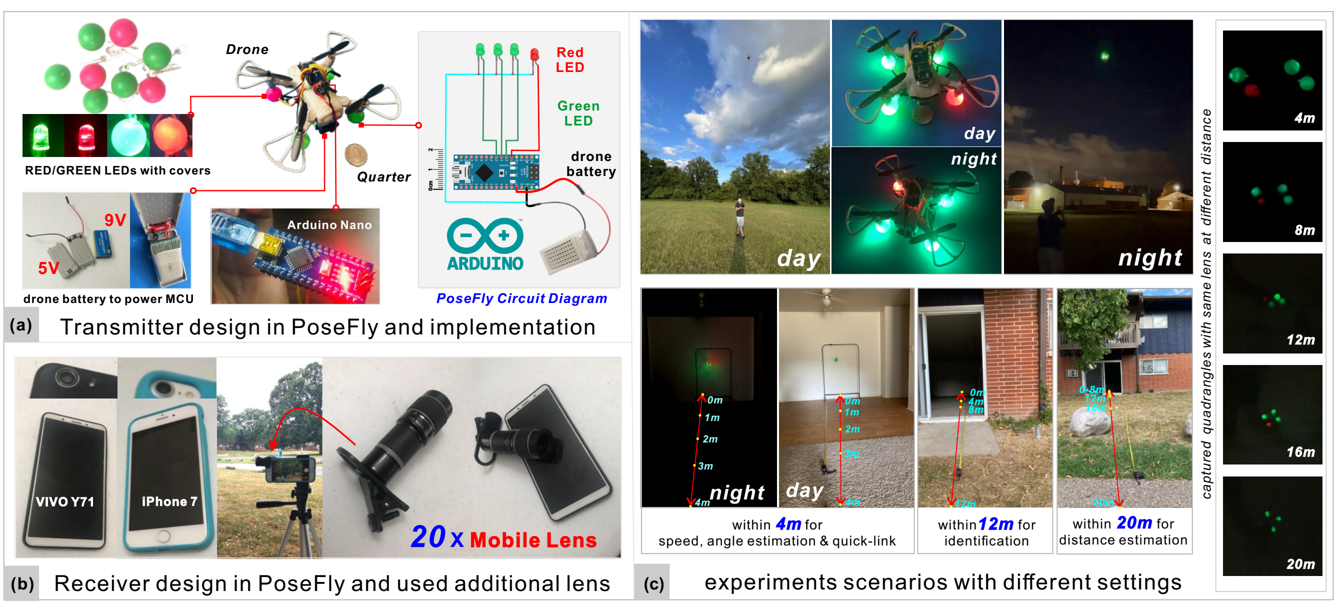
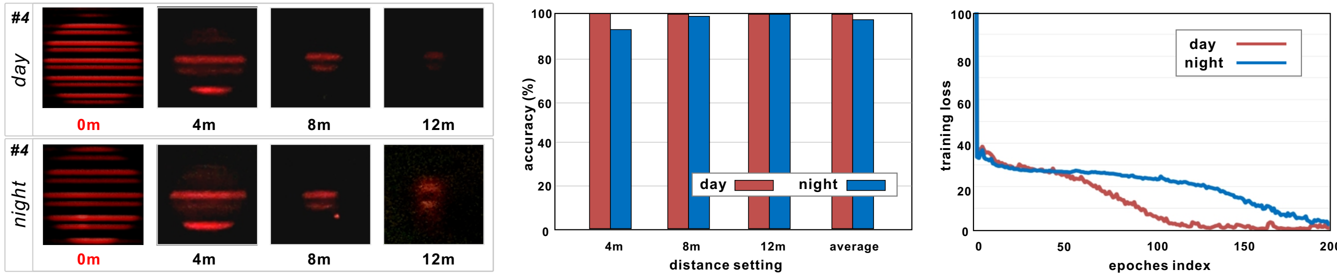


Fig. 12: PoseFly implementation including transmitter (a), receiver (b), and the experimental scenarios and setup (c).



(a) captured optical label #4 at different distance (b) identification accuracy of optical labels (c) training loss curves for day and night

Fig. 13: Drone identification: (a) captured optical labels of #4 in different distances, (b) optical label identification accuracy in both day and night, (c) training loss curves in epochs from [0, 200] in both day and night.

VIII. Performance

A. *Identification Accuracy*

In our experiment, we evaluate the identification accuracy of 15 active optical labels with index number in range of [1, 15]. We capture the optical labels shown in the red LED spot at 3 distance settings: 4 m, 8 m, and 12 m in both day and night time with random postures of the drone.

We capture 10 images for each setting (a specific optical label, a specific distance, day/night setting), thus in total we have $10\times15\times3\times2 = 900$ images as training dataset. The sampled images of label #4 are shown in Figure 13 (a). We evaluate the label identification accuracy performance during day and night, and their training loss in [0, 200] epochs.

Although the number of strips displayed on the cover become less with the increased distance from the drone to the camera and hard for recognizing by human eyes as shown in Figure 13 (a), the cyclic rolling pattern is still good enough for CNN to be classified which is demonstrated by Figure 13 (b). The identification accuracy of 15 optical labels achieves average 100% in day time and more than 97% at night. The training loss curve for data set of day time drops faster and earlier than the night as shown in Figure 13 (c). The reason is that it is harder to distinguish amplitudes between CP and On symbols at the night due to the fusion of optical signals.

B. *Localization Accuracy*

1) *Distance Estimation:*

We evaluate the distance estimation accuracy of 5 settings in [4 m, 8 m, 12 m, 16 m, 20 m]. We capture the spot shape of the drone with random postures and speed in both day and night time. We capture 10 images for each setting (a specific distance, day/night setting), thus in total we have $10\times5\times2 = 100$ images as the training dataset.

As shown in Figure 14 (a), the distance estimation accuracy during day time among all distance settings achieves 100%, which demonstrates our PoseFly can provide distance ranging within 20m among drones during day time. Similarly, PoseFly also works well for distance estimation at night with 100% accuracy within 20m.

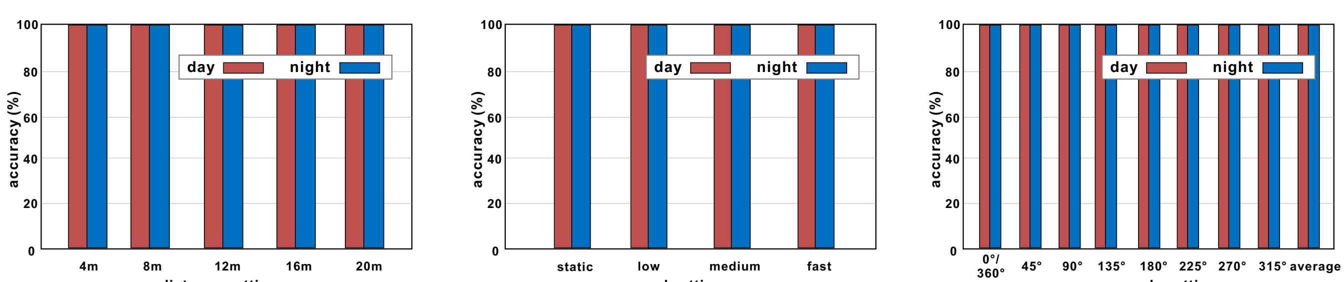
2) *Relative Speed Estimation:*

We evaluate the speed estimation accuracy of 4 settings in [static, low, medium, fast]. We capture the spot shape of drone with random postures at 4 m during both day and night time. We capture 10 images for each setting (a specific speed, day/night setting), thus in total we have $10\times4\times2$=80 images as the training dataset.

As shown in Figure 14 (b), the speed estimation accuracy during the day time among all four speed settings achieves 100% for both day and night, which demonstrates our PoseFly can provide robust relative speed estimation among drones.

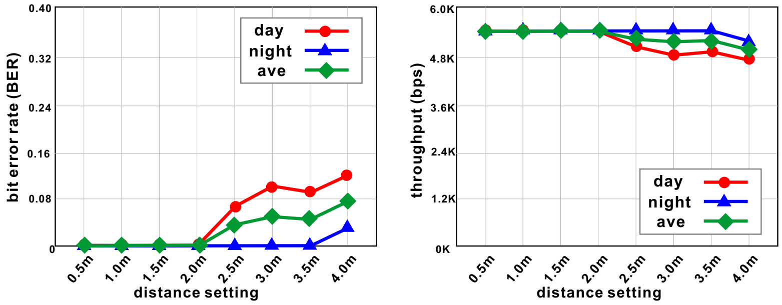
3) *Relative Angle Parsing:*

We evaluate the relative angle estimation accuracy of 8 settings in [0° or 360°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°]. We capture the spot shape of the drone with random speed at 4 m both day and night time. We capture 10 images for each setting (a specific relative angle, day/night setting),thus in total we have $10\times8\times2 = 160$ images as the training dataset. As shown in Figure 14 (c), the CNN model saved at the 200^{th} epoch can classify the drones with different relative angles of 8 options accurately for both day and night



(a) distance estimation performance (b) relative speed estimation performance (c) relative angle estimation performance

Fig. 14: Drone localization performance: (a) distance estimation accuracy, (b) speed estimation accuracy, (c) angle estimation accuracy in both day and night with models saved at 200^{th} epoch.



(a) quick-link bit error rate (b) quick-link throughput

Fig. 15: Quick link performance: (a) BER, and (b) throughput.

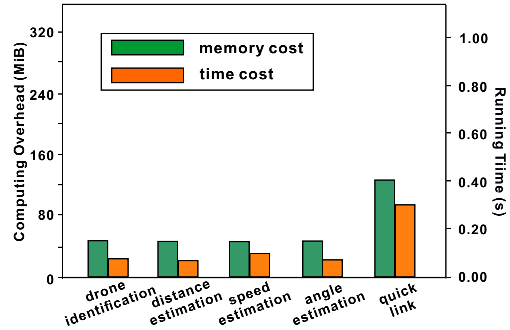


Fig. 16: Computation cost and latency evaluation.

with estimation accuracy of 100% within 4m sensing distance.

To sum up, our AI-assisted drone pose parsing/localization works well for all three aspects during day and the night in different distances for the flying drones.

C. Quick-link Evaluation

We evaluate the Quick-link performance within 4m (0.5m, 1m, 1.5m, 2m, 2.5m, 3m, 3.5m, 4m) during both day and night. We set the shutter speed properly (12Khz) with transmission frequency to capture clear rolling strips shown on the three green spots in each frame and set the frame rate as 60FPS. For each setting (a specific distance, day/night setting), we capture 10 images, thus totally $8\times2\times10 = 160$ images to measure its BER and achieved throughput.

BER performance. We decode OOK data sequence inside of two CPs. As shown in Figure 15 (a), the bit error rate in each frame is **0** within **2m** for both day and night. With the increased distance, the BER increased as well due to the weaker optical signals at longer distances. Nevertheless, our prototype still achieves the average BER less than **0.08** at **4m**. The reason the BER is higher during the day than at night is because of the lower amplitude gap of the captured On symbols and Off symbol during day due to the strong ambient light than at night for the same distance.

Throughput performance. The valid data bits in each frame is the sum of valid data in three green spots, which is calculated by 30 bits (32-2)$\times$3$\times$frame rate (60 FPS)$\times$BER. As shown in Figure 15 (b), our PoseFly achieves **5.4 Kbps** within **2m** for both day and night. Although the throughput drops with increased transmission distance, the dropped data amount is limited. Even at **4m**, our PoseFly still achieves the average throughput over **5 Kbps**. Although the captured spot size is decreased by the increased distance, we can still capture the complete and differentiable strips at 4m with lens.

D. Overhead

Computation overhead. LEDs provide lighting function and are energy efficient. Thus, we only consider the computation overhead at reader side. The reader should not conduct complex computations and consume energy too fast. The training processes are offline, the drone identification, distance, speed, and angle estimations are real-time tasks conducted with low computation cost for each step when the drone is flying. As shown in Figure 16, the quick link requires the most memory resources due to more narrow strips in decoding compared with CNN based tasks mentioned above. For all these tasks, they require a combined 313 MiB of memory and is not a computational burden for a commercial smart device.

Latency. For collaboration tasks among drones, time can be important to improve the efficacy and efficiency. Compared with state-of-art drone localization systems, including audio-based systems, PoseFly has nearly no time delay in signal propagation due to the fast propagation of light. Thus we only consider the computational latency. As shown in Figure 16, the drone identification, drone on-site localization (distance, speed, angle estimation) have a low running time of about 0.07s-0.09s for each. These functions can be run in a pipeline manner (i.e., 0.07s-0.09s in total) and thus achieve the real-time on-site pose parsing. For example, given 2 drones with 20m/s relative speed, after drone A completes its pose parsing function for drone B, the parsed distance may only have 20m/s$\times$0.09 = 1.8 m distance estimation error. The distance estimation in PoseFly is designed for discrete distance ranges [4 m, 8 m, 12 m, 16 m, 20 m], and 1.8 m distance estimation error is acceptable and practical. In contrast to real-time on-site drone pose parsing, the quick link function is designed for information sharing (e.g., roughly which drones are nearby, some broadcast commands) if needed which does not strictly require real-time communication. Thus, 0.31 s latency is acceptable, which is similar to the collaborations among geese.

IX. Discussion

Comparison with Existing Work. *(1) Passive optical label.* Compared with passive optical labels such as bar codes and QR codes with similar size (2 cm x 2 cm) as the red cover in our prototype, we measured that these passive optical labels are only workable within 50cm. *(2) RF-based localization.* RF-based localization can provide distance estimation error within about several meters with a localization time of more than 70 seconds while not providing other aspects of drone pose parsing in our PoseFly such as angle and speed estimation [24]. *(3) RF/OCC communication.* RF techniques can provide long communication distance, however, they face the severe interference when there are massive drones. Existing OCC approaches can achieve similar several Kbps throughput ability, however, they do not provide optical labeling, and on-site localization functions [25].

Other Concerns. *(1) discrete value.* Current PoseFly provides discrete relative localization instead of continuous relative distance/angle/speed value. However, PoseFly is designed for swarming drones' collaboration which does not require the exact value of relative positioning, similar to geese flying. *(2) modulated ambient light.* Although there are modulated light such as LiFi (>100KHz) transmitters, our PoseFly can filter them out via spatial diversity of millions of camera pixels and a different frequency (about 10 KHz). *(3) frame gap loss.* The transmitted data in quick-link channel are repeated for broadcast and thus the frame gaps caused data loss will not impact the final decoded data. *(4) foggy weather.* The foggy weather can impact light propagation. However, our PoseFly is designed for swarming drones. Provided two nearby drones can sense and communicate with each other in foggy weather (i.e., less than 4m), the network of swarming drones can still work. *(5) future work.* We will investigate vision geometry-based algorithms and models for localization instead of discrete CNN classification to reduce overall dependency on CNN. We will also explore the upper bound of the PoseFly by extended experiments and prototype upgrade as well as security concerns in the future.

X. Conclusion

In this paper, we propose PoseFly for simple and robust on-site drone pose parsing via 4-in-1 optical camera communication. We design a color-arc scheme and investigate spatial embedding ability of rolling shutter cameras and first exploit it for drone localization including relative distance, speed, and angle estimations. Besides, we design active optical labels with cyclic pilot and data sequences in frame-level for high-capacity drone indication and quick-link communication for real-time and smooth collaboration among drones. Finally, we conduct experiments on implemented prototypes in various scenarios. The solid experiments show that our PoseFly can achieve near 100% accuracy for drone identification at up to 12m, 100% drone localization as well as 5 Kbps average data rate with average BER lower than 0.08 at up to 4m for both day and night. These results demonstrate our PoseFly works well.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. CNS-2226888 and Grant No. CCF-2007159. Great thanks for helps from my friends and lab mates especially Eakachai and Kanishka.

References

[1] B. Li, Z. Fei, and Y. Zhang, "Uav communications for 5g and beyond: Recent advances and future trends," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2241–2263, 2018.

[2] A. Sharma, P. Vanjani, N. Paliwal, C. M. W. Basnayaka, D. N. K. Jayakody, H.-C. Wang, and P. Muthuchidambaranathan, "Communication and networking technologies for uavs: A survey," *Journal of Network and Computer Applications*, vol. 168, p. 102739, 2020.

[3] V. M. Research. (2022) Drones market size and forecast. [Online]. Available: https://www.verifiedmarketresearch.com/product/drones-market/

[4] L. Bertizzolo, S. D'Oro, L. Ferranti, L. Bonati, E. Demirors, Z. Guan, T. Melodia, and S. Pudlewski, "Swarmcontrol: An automated distributed control framework for self-optimizing drone networks," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020.

[5] J. Hu, A. Bruno, D. Zagieboylo, M. Zhao, B. Ritchken, B. Jackson, J. Y. Chae, F. Mertil, M. Espinosa, and C. Delimitrou, "To centralize or not to centralize: A tale of swarm coordination," *arXiv preprint arXiv:1805.01786*, 2018.

[6] X. Zhang, G. Klevering, X. Lei, Y. Hu, L. Xiao, and T. Guanhua, "The security in optical wireless communication: A survey," *ACM Computing Surveys (CSUR)*, 2023.

[7] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Towards scalable and ubiquitous millimeter-wave wireless networks," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 257–271.

[8] A. Galisteo, Q. Wang, A. Deshpande, M. Zuniga, and D. Giustiniano, "Follow that light: Leveraging leds for relative two-dimensional localization," in *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*, 2017, pp. 187–198.

[9] S. Zhu, C. Zhang, and X. Zhang, "Automating visual privacy protection using a smart led," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, 2017, pp. 329–342.

[10] X. Zhang and L. Xiao, "Lighting extra data via owc dimming," in *Proceedings of the Student Workshop*, 2020, pp. 29–30.

[11] ——, "RainbowRow: Fast Optical Camera Communication," in *2020 IEEE 28th International Conference on Network Protocols (ICNP)*. IEEE, 2020, pp. 1–6.

[12] X. Zhang, J. Mariani, L. Xiao, and M. W. Mutka, "Lifod: Lighting extra data via fine-grained owc dimming," in *2022 19th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2022, pp. 73–81.

[13] X. Zhang, H. Guo, J. Mariani, and L. Xiao, "U-star: An underwater navigation system based on passive 3d optical identification tags," in *The 28th Annual International Conference on Mobile Computing and Networking*, 2022.

[14] X. Zhang, G. Klevering, and L. Xiao, "Exploring rolling shutter effect for motion tracking with objective identification," in *Proceedings of the Twentieth ACM Conference on Embedded Networked Sensor Systems*, 2022, pp. 816–817.

[15] X. Zhang and L. Xiao, "Effective subcarrier pairing for hybrid delivery in relay networks," in *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 2020, pp. 238–246.

[16] E. Unlu, E. Zenou, and N. Riviere, "Using shape descriptors for uav detection," *Electronic Imaging*, vol. 2018, no. 9, pp. 128–1, 2018.

[17] P. Nguyen, M. Ravindranatha, A. Nguyen, R. Han, and T. Vu, "Investigating cost-effective rf-based detection of drones," in *Proceedings of the 2nd workshop on micro aerial vehicle networks, systems, and applications for civilian use*, 2016, pp. 17–22.

[18] P. Nguyen, T. Kim, J. Miao, D. Hesselius, E. Kenneally, D. Massey, E. Frew, R. Han, and T. Vu, "Towards rf-based localization of a drone and its controller," in *Proceedings of the 5th workshop on micro aerial vehicle networks, systems, and applications*, 2019, pp. 21–26.

[19] K. R. Sapkota, S. Roelofsen, A. Rozantsev, V. Lepetit, D. Gillet, P. Fua, and A. Martinoli, "Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Ieee, 2016.

[20] L. Tao, T. Hong, Y. Guo, H. Chen, and J. Zhang, "Drone identification based on centernet-tensorrt," in *2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. IEEE, 2020, pp. 1–5.

[21] D. Li, J. Hu, R. Ma, X. Wang, Y. Liu, and Z. Zhu, "Spad-based lidar with real-time accuracy calibration and laser power regulation," *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2022.

[22] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2015.

[23] "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)*, pp. 1–407, April 2019.

[24] Z. Shaikhanov, A. Boubrima, and E. W. Knightly, "Autonomous drone networks for sensing, localizing and approaching rf targets," in *2020 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2020, pp. 1–8.

[25] P. Hu, P. H. Pathak, H. Zhang, Z. Yang, and P. Mohapatra, "High speed led-to-camera communication using color shift keying with flicker mitigation," *IEEE Transactions on Mobile Computing*, vol. 19, no. 7, pp. 1603–1617, 2020.