Camera-enabled Joint Robotic-Communication Paradigm for UAVs mounted with mmWave Radios

Sara Garcia Sanchez, Rishabh Shukla and Kaushik Chowdhury
Institute for the Wireless Internet of Things, Northeastern University
sgarcia@coe.neu.edu,shukla.ris@northeastern.edu,krc@ece.neu.edu

ABSTRACT

UAVs mounted with millimeter wave base stations will enable last-mile high bandwidth access, as well as help in rapidly deploying point to point aerial backhaul links. Because such transmitters use directional beamforming to increase capacity, UAV deployments require careful selection of the beamwidth. Even under regular hovering conditions, UAVs display minor relative rotations and displacements caused by GPS inaccuracies and environmental factors like wind. To ensure narrow beams are perfectly aligned in such practical conditions, we propose a beamforming framework that (i) fuses out-of-band information obtained from cameras and (ii) leverages antenna beam-patterns characterized online during flight. These inputs provide the UAV pair forming the link with an improved estimate of relative orientation, and furthermore, guide controlled and coordinated movements to ensure the mmWave beams remain aligned. We implement this joint robotics-communication framework within the robot operating system and evaluate the performance for emulated DJI M100 UAVs. results reveal 33% improvement in physical bitrate and 60.4% reduction in latency when compared to RF-only beam sweeping methods.

CCS CONCEPTS

• **Computer systems organization** → **Architectures**; • **Networks** → **Network properties**.

KEYWORDS

UAV, beam tracking, mmWave, optical cameras, robotics

ACM Reference Format:
Sara Garcia Sanchez, Rishabh Shukla and Kaushik Chowdhury. 2022. Camera-enabled Joint Robotic-Communication Paradigm for UAVs mounted with mmWave Radios . In *4th ACM Workshop on Drone-Assisted Wireless Communications for 5G and Beyond (DroneCom'21), January 31-February 4, 2022, New Orleans, LA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3477090.3481052

DroneCom'21, 01/31–02/04, 2022, New Orleans, USA

ACM ISBN 978-1-4503-8705-7/22/01…$15.00
https://doi.org/10.1145/3477090.3481052

1 INTRODUCTION

The explosive growth in IoT deployments and anticipated data volumes that will be generated within future autonomous vehicles requires the wireless network to support high data rates beyond 10 Gbps. Millimeter wave (mmWave) frequencies offer multi Gbps transmission rates with nearly 100x greater channel bandwidths than what is available in the sub-6 GHz legacy band [9]. Moreover, UAVs with mmWave base stations mounted on them also offer great flexibility for rapid deployment [3].

However, the constant displacement experienced during hovering and relative changes in position during intentional flight manoeuvres introduces challenges in maintaining the orientation of highly directional mmWave links, typically undertaken through a technique called as *beamforming* [9]. As an example, 60 GHz COTS Terragraph mmWave hardware can form beams as narrow as 2.8° [2]. The use of such narrow beams require an extensive exploration of the 3-D space to find the optimal beam pair that provides the highest possible signal strength at the receiver. This process, known as *beam sweeping*, results in significant disruption for communication [6]. Furthermore, when these radios are mounted on UAVs, the natural motion during hovering changes the relative location between transmitter (Tx) and receiver (Rx), (see Fig.1). This motion, resulting from continuous course corrections due to inaccuracies in the UAV Inertial Measurement Units (IMU) sensors and GPS localization [4], can cause beams to become misaligned and potentially trigger repeated beam sweeping every time the received power drops below a threshold. Specifically, the standards-defined beam sweeping known as *Sector Level Search* (SLS), can take from hundreds of ms up to a few seconds to complete [10], with complete disruption to the application in this intervening time. Thus, we aim to avoid repeatedly triggering the SLS procedure by ensuring narrow mmWave beams remain aligned during flight.

This paper develops a complete framework combining visual and RF inputs with coordinated motion through the

Robot Operating System (ROS)[7]. Our framework allows the UAVs to independently undertake fine adjustments through (i) corrective physical motions and (ii) online estimation of the mmWave link status during flight, thereby mitigating the effects of sudden link outage.

1.1 Summary of Contributions

Our main contributions are as follows:

1. We propose a framework to achieve seamless fusion of multimodal RF hardware and sensors, including optical cameras, IMUs and mmWave radios. We return proactive UAV course-corrections with the objective of keeping mmWave narrow beams aligned at all times.

2. We design a fine-grained method for detecting UAV relative displacements by building in-situ beampatterns of the mounted mmWave antennas. Specifically, our approach can track 2.8° narrow-beams providing 2.6 times faster link recovery compared to the IEEE 802.11ad standard.

3. We show how the video, RF and motor control aspects can be combined within the ROS framework, considering costs of ROS signaling in order to perform high resolution real-time beam-tracking.

4. We release to the community the ROS code for UAV synchronized motion, the C++ image processing code and a dataset of images captured during real flight [5].

2 RELATED WORK

Pure RF-based beam tracking solutions focus on determining the Angle-of-Arrival (AoA) of the incoming signal at the Rx, after which a directive beam is steered towards the Tx [6]. However, AoA techniques offer limited angular resolution in practical scenarios, often compensated by using wider beams for transmission, ranging from 7° to 10° [13], providing lower power density and a consequent SNR drop at the Rx. Robotic control capability offer solutions to enhance UAV stability, which we also aim to leverage in maintaining beam alignment. They do so either by supporting UAV navigation with additional hardware, such as cameras [8], by filtering GPS noise with additional sensors and via predictive methods like Kalman filtering. However, these solutions do not allow an immediate reaction by the wireless system to counter a sudden link performance drop, for e.g., if the performance drop is caused by an external factor, such as wind burst, the link is not re-established until UAV stability is recovered, likely taking up to a few seconds. To enhance the tracking performance, multimodal approaches combine visual data with additional RF or sensor inputs, such as mmWave radar [12]. However, none of these works consider the high resolution provided by narrow mmWave beampatterns.

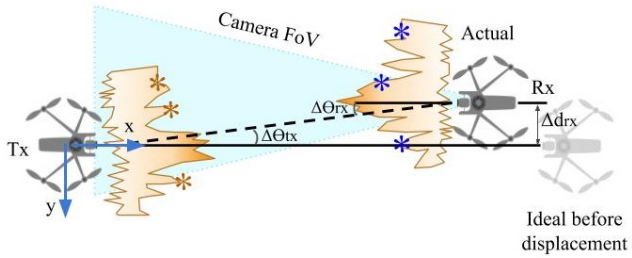


Figure 1: Beam misalignment caused by UAV translational displacements during hovering. The camera at the Rx helps to coarsely estimate the relative UAV locations, while the beam-patterns are used for fine tuning the correctional movement.

3 OVERVIEW OF PROPOSED APPROACH

To ensure continuously oriented beams for UAVs mounted with mmWave radios, we propose the following two-step approach, that uses a combination of camera video and the knowledge of mmWave antenna beampatterns:

Step 1: The Rx processes video frames captured locally, and coarsely recognizes relative displacements of the Tx with respect to the origin, centered at the Rx location. We use ROS in order to ensure the Tx UAV is within the FoV of the Rx camera.

Step 2: Next, we undertake a *fine-grained* adjustment that leverages the shape of narrow mmWave antenna beampatterns, obtained online during flight. The correlation between measured power and UAV's relative displacement then drives the required location correction for UAVs. This is, the direction and amount of rotation for both UAVs that will bring about beam alignment ($\Delta\theta_{tx}$, $\Delta\theta_{rx}$ in Fig.1).

This approach poses several challenges. First, we note that beampatterns are surjective functions, i.e., a given power measurement can be caused by multiple UAV relative displacements (asterisks in Fig.1). Thus, only using the measured power to determine the required rotation leads to ambiguity. Second, it is known that in practice, changing the beam steering direction alters the shape of the side lobes. Since our approach relies on the knowledge of a fixed beampattern, we cannot perform analog/digital beam steering to align the beams. Thus, an alternative technique is needed for tracking. Therefore, we consider Step 1 as a *coarse-grained* estimation of the UAV relative displacement used to alleviate ambiguity. In addition, our approach *freezes* the beam orientation, and thus, the characteristic pattern remains fixed after the initial characterization. Instead of doing analog/digital beam steering, all further alignment in a 2-D beamforming (azimuth steering) case is subsequently undertaken through the UAV's physical rotation through ROS, relying only on the IMU sensor on board the UAVs.

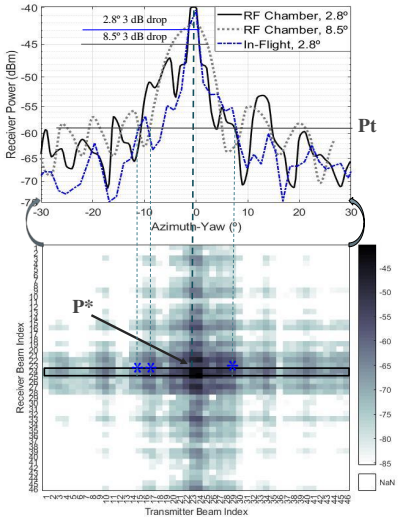


Figure 2: Extraction of a 2.8° beampattern from an measured heatmap and comparison with 2.8° and 8.5° beampatterns measured on an RF anechoic chamber.

For an extension to 3-D beamforming, and given that UAVs cannot be rotated in the elevation dimension, our approach bounds the maximum rotational angles by performing an online characterization of the spatial volume where a UAV is expected to hover with high probability. This allows for the mmWave beamwidth adjustment in elevation, tailored to specific in-flight hovering conditions. Instead of relying on GPS measurements, we derive this region of space from the measured fluctuations in the received power, providing an order of magnitude improvement in localization accuracy. We note that our approach provides several advantages over existing solutions. First, the closed-loop visual-RF control helps to make proactive decisions, *before* the adverse impacts of the wireless link degradation totally disrupts communication. Second, narrower beamwidths than state-of-the-art solutions can be used, providing higher channel capacity. Third, the online UAV hovering estimation supports optimum selection of beamwidth in elevation, resulting in higher gain.

4 JOINT ROBOTIC-COMMUNICATIONS FRAMEWORK

Next, we detail our approach and introduce a decision framework to establish and track mmWave links under mobility.

4.1 Fine-grained Correction via Antenna Beampattern

We first note that the beam misalignment loss is an outcome of the unique signature shape of the antenna beampatterns (see Fig.1). This is primarily related to the shape of the main and side lobes, corresponding to the power distribution in azimuth (yaw) and elevation (pitch). Thus, from the antenna beampattern, we aim to accurately map the measured power with an estimate of the UAV's relative displacement. This solution provides an accurate estimation, particularly for narrow beams, whose shapes tend to be highly irregular. In order to show the trade-off between directionality through narrow beams and the resulting irregularity on the beampattern side lobes, we synthesize a narrow (2.8°) and wide (8.5°) beams in an RF anechoic chamber, as shown in Fig.2. To do so, we use COTS Terragraph mmWave channel sounders designed for channel modeling in 60 GHz [2]. The characteristic shapes of the side lobes shape exhibit unique gain variations when measurements are taken at different angles within the beampattern. Thus, we leverage the rich angular diversity present in narrow beampatterns to determine the relative displacements between the Tx and Rx UAV pair.

4.2 Online beampattern characterization

While an accurate characterization of the antenna beampattern in an RF anechoic chamber is helpful, this may not always be feasible from a resource availability viewpoint. Furthermore, the UAV airframe introduces signal reflections at mmWave frequencies, which cannot be characterised unless that particular model of the UAV is also part of the measurement setup. To overcome these limitations, we propose an *online* method to characterize the beampattern during the standard SLS process. We illustrate this approach in Fig. 2, where we show a heatmap containing received power values collected during SLS for a beamwidth of 2.8°during an actual flight. The experimental setup consists of a pair of Terragraph channel sounders mounted on M600 DJI Pro UAVs to create an aerial 60 GHz link. The x-axis and y-axis of the heatmap represent the Tx and Rx beam indices evaluated during SLS, respectively. Thus, each point of the heatmap contains the received power value for a given Tx-Rx beam pair. The grid point with highest received power P^* represents the best beam pair combination after SLS, and so, the row and column containing P^*, serve as an initial approximation for the Tx and Rx antenna beampatterns. The approximated beampattern for the Tx is illustrated in Fig. 2 and contrasted with its accurate characterization obtained within the RF anechoic chamber. We note from Fig.2 that several grid locations have similar received power values, introducing ambiguity in determining the correction angle, as discussed in Fig.1. This implies that the UAV can potentially be rotated in completely divergent directions, possibly further worsening the orientation. Thus, we augment the fine-grained adjustment by the correction described next.

4.3 Coarse-grained Correction via Video

In this step, we leverage visual information from a camera mounted at the Rx UAV. The location of the Tx UAV within

the FoV of the camera, captured via video, provides a coarse-estimation of the Tx-Rx relative location. In a 2-D beamforming scenario, we utilize this information to eliminate the left vs right ambiguity described in the above section.

Specifically, the UAV Rx obtains the camera image frame I_t at instant t (when the link power drops) and compares it with the image I_0 taken at t_0, when beams were last aligned. Then, it computes the UAV Tx relative direction of movement within the time window $[t_0, t]$ that caused the detected power drop, and allocates one of the following two labels to I_t: {Right (R_t), Left (L_t)}. Note from Fig.1 that, for the given translational displacement, the labels for both UAVs are the same (L_t, L_t). This applies to any combination of UAV Tx-Rx translational displacements, and therefore, it is sufficient to perform the analysis at either Tx or Rx. We chose to do it at the Rx, as it directly collects the SLS heatmap, thus avoiding feedback to the Tx. The left/right label so identified helps to bring focus on the region of the beampattern in which we apply the fine-grained RF-domain analysis. In the case of Fig.1, we only consider the left half of the antenna beampattern, as Tx and Rx are located to the right of each other (opposite to the label). Thus, the image analysis reveals that the UAVs communicate through the left region of their respective beampatterns.

To determine the location of the UAVs within the images, we use a known method from computer vision which performs object detection in a camera feed [11]. We convert every image to grayscale and binarize each pixel to a value of either 0 (black) or 1 (white). We then perform object contouring to distinguish the multiple objects present in the frame. After running a set of morphological operations on the contours, including object boundary reduction and noise removal, we distinctively separate the UAV from the surroundings and generate a bounding box to localize it within the image. We do so based on the UAV aspect ratio, i.e, the width of the UAV is 1.5 to 2.5 times its height, as this is distance-independent. This approach, however, is not sufficient to detect the UAVs when the background behind has similar color tones. Thus, we additionally run in parallel a CSRT (Channel and Spatial Reliability Tracking) object tracker, that takes as initial input the UAV location from the contour object detector and continuously tracks the UAV through successive video frames. If the contour object detector fails to detect the UAV, we take the location estimated by the CSRT tracker. Moreover, if a mismatch between the contour object location and CSRT location occurs, we re-initialize the CSRT to compensate for the error. We store the central location of the UAV in each frame for optical tracking. The difference between the horizontal index of the central pixel for images collected at instants t and t_0 determines the labels R_t, L_t and resolves positional ambiguity.

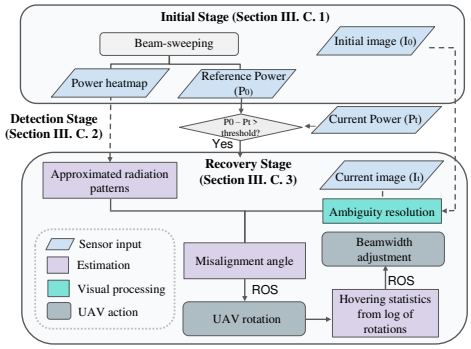


Figure 3: Three-stage decision framework.

4.4 Decision Framework

Next, we present a three-stage decision framework (Fig.3) to establish and maintain robust links by using coarse- and fine-grained orientation corrections and beamwidth adjustment.

1. *Initialization Stage.* Our approach detects link performance degradation when the current received power P_t differs from a reference value P_0. Ideally, P_0 only depends on the antenna gains, operation frequency and distance between Tx and Rx. However, propagation effects like multipath do alter the received power. Thus, in order to accurately determine an initial value for P_0 during deployment, we first trigger the standard SLS to establish the link. The received power measured right after establishing the link P^*, corresponds to the maximum achievable reference value P_0. From the SLS, we also obtain the power heatmap in Fig. 2. We use this heatmap to build online the approximated Tx and Rx antenna beampatterns. To complete the initialization stage, the Rx takes a reference image I_0 right after SLS, when beams are perfectly aligned, later used for coarse-grained correction.

2. *Detection Stage.* During operation, we monitor the received power P_t over time, and compare it to P_0. Fluctuations in P_t can be caused by slow UAV displacements during hovering or sudden changes due to environmental factors like wind. We set a design *threshold* beyond which any power drop triggers the recovery stage.

3. *Recovery Stage.* When the power drop difference between P_0 and P_t exceeds the threshold, the camera at the Rx takes an image I_t. Using I_t and I_0, we determine the labels R_t or L_t, according to Sec.4.3. This eliminates ambiguity and focuses the beam tracking algorithm in either left or right sides of the beampattern. Notice that for any type of translational displacement among the UAVs, Tx and Rx need to rotate equal angles for re-establishing beam alignment (see Fig.1, where $\Delta\theta_{tx}$ equals to $\Delta\theta_{rx}$). Thus, we only consider equally rotated angles for Tx and Rx, which we list from lower to higher magnitude. Then, both UAVs start synchronized rotation though ROS commands towards the determined common direction, an amount equal to the first (smallest) angle in the list. If rotation does not result in reducing the difference

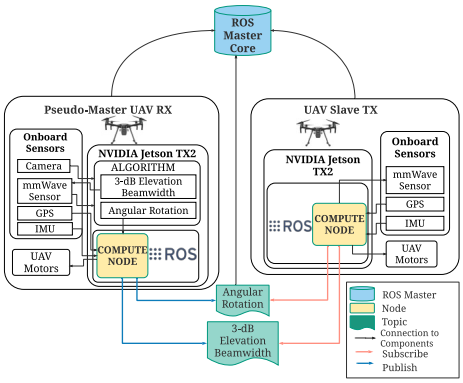


Figure 4: ROS framework: computational, communication and control elements. ROS runs on a NVIDIA JETSON TX2 mounted on the UAVs. It issues directives to the UAV flight control using IMU and GPS data.

between P_0 and P_t below the threshold, both UAVs are further rotated according to the next angle in the list, until link performance is restored and P_t approaches P_0. Once the link is reestablished, we take a new reference image to update I_0.

4.5 Systems Implemenation via ROS

Once the Rx estimates the angular rotation, it needs to be shared with the Tx. Then, both UAVs perform synchronized and coordinated motion to re-establish beam alignment. To do so, we use ROS, as it is a convenient and light software framework that helps establish communication at a high level between UAVs, and at a low level between on-board sensors and the computing module (NVIDIA Jetson TX2). We illustrate the proposed ROS framework in Fig.4. On the Rx UAV side, the TX2 compute board has two functional blocks: the *algorithm block* and the *ROS block*. The former takes inputs from the camera and mmWave sensor and calculates the angular rotation needed for both UAVs, and, if it applies, the updated 3-dB elevation beamwidth. These data are then fed into the compute node of the ROS block. This node performs two main tasks: First, using feedback from the IMU sensor data, it commands the UAV motors to achieve the desired rotation. Second, it publishes the rotation commands on a ROS *topic* at an update rate of 4 samples per second and the 3-dB elevation beamwidth under severe detected hovering conditions. The ROS compute node on the slave TX subscribes to both topics to read and implement the commands.

5 PERFORMANCE EVALUATION

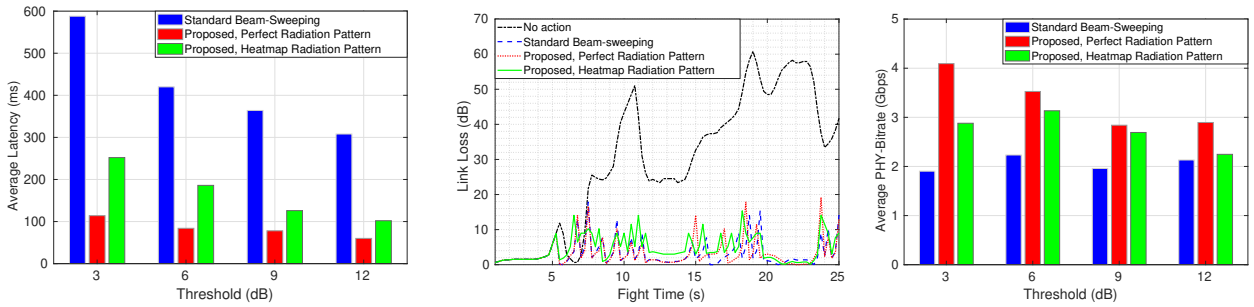
5.1 Emulation Setup

To validate our approach, we emulate a UAV-to-UAV link within the ROS environment [7]. For the emulation, both UAVs rely on noisy GPS for navigation, and thus, they suffer from hovering-related translational displacement. In addition, we equip the UAVs with noise-free (ideal) GPS sensors, which provide their true locations and relative displacements. From this, and the realistic beampattern presented in Fig.2, we determine the experienced link loss and the value of P_t. Using the Gazebo simulator [1], we collect a video from the Rx UAV, as it a realistic demonstrator of outcomes compared to numerical simulations. Taking the video frames and mmWave sensor as inputs, the UAV Rx calculates the angular rotation needed for both UAVs and send it to the Tx through ROS. Then, using feedback from the IMU sensors, we command the UAVs motors to achieve the desired synchronized rotation. Although the end-to-end performance described next uses video from Gazebo, we separately validate our vision detection method with video collected during live UAV flights. We record video at 28 frames/s, using a DJI Zenmuse x3 camera on the UAV Rx, that provides H.264 encoding and 1280x720 resolution. Including the CSRT, we achieve UAV detection under complex background in 100% of the frames, compared to a 29.5% using contour object detector only.

5.2 End to End Performance

We take as baseline the standard IEEE 802.11ad and use the narrow beams of 2.8° beamwidth shown in Fig.2. We consider that Tx and Rx evaluate a total of 30 beam pairs over 360° during the SLS process, introducing an estimated latency of 700 ms [10]. Once the best beam pair is found, the beam steering time, in the order of μs, is negligible in comparison. From our emulations, ROS introduces a latency below 1 ms, making the computing loop itself the biggest contributor to the latency, of approximately 5 ms. The image processing is kept below 20 ms using C++. The mechanical steering for UAV rotation takes 100-200 ms for angular rotation below 10° in our emulations.

In Fig.5(a) we show the latency using the three different approaches for link recovery: (i) the standard beam sweeping, (ii) our proposed approach assuming perfect knowledge of the antenna beampattern, i.e., with the pattern characterized in an RF anechoic chamber, and (iii) our proposed approach in which we obtain the antenna beampatterns online, from the power heatmap empirically measured during an actual UAV flight (shown in Fig.2). For each method, we estimate the link disruption rate during the flight time, i.e., the number of times for which the link loss goes above a given design threshold, and the recovery stage is triggered. The latency is derived from the disruption rate, averaged, and expressed in milliseconds. In Fig.5(b), we show the link loss during 25 seconds of emulated flight time, using all three recovery approaches as well as not using link recovery. In this case, the recovery is triggered every time the power drops above 6 dB. For Fig.5(c), we first estimate the average link loss during the flight time for all three approaches. Then, we map the link loss and physical (PHY)-bitrate based on experimental data



(a) (b) (c)

Figure 5: (a) Average latency in resuming beam-alignment per 1s of emulation time, (b) link loss during emulated flight, threshold for triggering link recovery set to 6 dB (c) average PHY-bitrate including the effect of latency.

collected with the Terragraph mmWave channel sounders. We include the effect of latency on the total bitrate in Fig.5(a).

The standard solution achieves perfect beam alignment (misalignment loss drops to 0 dB) after beam sweeping, as it always finds the best pair beam with power P^*, at an expense of high latency. In contrast, our approach accuracy is bounded to a maximum $\pm 1.2^\circ$ and an average of 1°, caused by the error between the actual and the measured (online) beampatterns. The near-constant PHY-bitrate achieved by the standard solution for different thresholds shown in Fig.5(c) is the result of the compensation between higher average misalignment loss and lower average latency as the threshold increases and the beam sweeping is called fewer times. Our proposed approach outperforms the standard solution due its near-perfect beam alignment and reduced latency.

5.3 Computational Overhead

The computational overhead of our approach is mainly determined by the object detection algorithm. As soon as the UAV is first detected, the CSRT tracker initializes and continuously runs in the background, updating the UAV location at a rate of 20 frames per second. We only perform any further image processing when the received power drops below the specified threshold. For a threshold of 6 dB, our algorithm is triggered every 1.78 and 0.92 seconds in average, for perfect and heatmap-based beampatterns, respectively. At these instances, a video frame is taken and processed at the Rx, where most of the computation occurs. Since the heatmap does not need to be shared with the Tx, only two float numbers (as the algorithm output) are sent through ROS on a wireless WiFi channel, which along with headers (timestamp, sequence number and frame), require less than 50 bps.

6 CONCLUSIONS

In this paper, we present a multimodal framework which leverages mmWave, visual and IMU sensor data in order to track mmWave narrow beams in UAV-to-UAV links under typical mobility conditions. Using computer vision and the online characterization of the antenna beampatterns during flight, we perform coarse- and fine-grained estimations of the UAV pair relative displacements. Our results based on ROS emulations demonstrate beam alignment with typical errors lower than 1° and 33% improvement on the link PHY-bitrate compared to the standard approach.

ACKNOWLEDGEMENT

This work is supported by the Office of Naval Research under grant N000141612651.

REFERENCES

[1] Gazebo Simulator. http://gazebosim.org/. Accessed: 2020-10-14.
[2] Terragraph Radios. https://terragraph.com. Accessed: 2020-06-30.
[3] M. Gapeyenko, V. Petrov, D. Moltchanov, S. Andreev, N. Himayat, and Y. Koucheryavy. 2018. Flexible and Reliable UAV-Assisted Backhaul Operation in 5G mmWave Cellular Networks. *IEEE JSAC* 36, 11 (2018), 2486–2496.
[4] S. Huang, J. Huang, D. Tang, and F. Chen. 2018. Research on UAV Flight Performance Test Method Based on Dual Antenna GPS/INS Integrated System. In *IEEE ICCIS*. 106–116.
[5] GENESYS Lab. [n.d.]. ROS and Beam Tracker for UAV at mmWave. https://www.genesys-lab.org/source-codes. Accessed: 2021-07-30.
[6] T. Nitsche, A. B Flores, E. W Knightly, and J. Widmer. 2015. Steering with Eyes Closed: mmWave Beam Steering without In-Band Measurement. In *IEEE INFOCOM*. 2416–2424.
[7] M. A Olivares-Mendez, S. Kannan, and H. Voos. 2014. Setting Up a Testbed for UAV Vision Based Control Using V-REP & ROS: a Case Study on Aerial Visual Inspection. In *IEEE ICUAS*. IEEE, 447–458.
[8] B. Patel, M. Warren, and A. Schoellig. 2019. Point Me In The Right Direction: Improving Visual Localization on UAVs With Active Gimballed Camera Pointing. In *IEEE CRV*. 105–112.
[9] A. N. Uwaechia and N. M. Mahyuddin. 2020. A Comprehensive Survey on Millimeter Wave Communications for Fifth-Generation Wireless Networks: Feasibility and Challenges. *IEEE Access* 8 (2020), 62367–62414.
[10] Y. Yaman and P. Spasojevic. 2016. Reducing the LOS ray beamforming setup time for IEEE 802.11ad and IEEE 802.15.3c. In *IEEE MILCOM*. 448–453.
[11] Jamileh Yousefi. 2011. Image binarization using Otsu thresholding algorithm. *University of Guelph, Ontario, Canada* (2011).
[12] R. Zhang and S. Cao. 2019. Extending Reliability of mmwave Radar Tracking and Detection via Fusion With Camera. *IEEE Access* 7 (2019), 137065–137079.
[13] W. Zhang and W. Zhang. 2018. Beam Training and Tracking Efficiency Analysis for UAV mmWave Communication. In *IEEE ICCS*. 115–119.