UVTrack360: UWB-aided Tracking in 360° Videos with Open-Vocabulary Descriptions

Krishna Jaganathan, Ashutosh Dhekne
Georgia Institute of Technology
{kjaganathan7, dhekne}@gatech.edu

***Abstract*—Computer vision based techniques are frequently used to locate a subject of interest in a video, so that video feature enhancements, blurring, and directional microphone-based audio capture can be enabled. This has applications in film-making, security, and privacy. The subject's actual location in the video feed is also important for audio post-processing, say in Dolby Atmos, where dubbed audio is associated with the subject's movements for realistic spatial audio. Detecting where a subject is in a video feed is nontrivial because the lens field of view must be mapped to the 3D space, the subject needs to be identified (potentially using open-vocabulary descriptions of the person), sometimes from several other subjects, and then tracked continuously, even if the subject might be occluded in some frames by other objects. In this paper, we ask: can the subject wear a wireless localization tag to simplify the task? We observe that the accuracy of subject localization in the video feed improves, the subject can be located even in cases where computer vision would fail (low-light conditions), and the compute time required to find the subject in the frame reduces, thereby reducing the detection latency of tracking using video alone.**

***Index Terms*—Person Tracking, UWB, Open-Vocabulary, 360-degree Video, Indoor Localization.**

I. Introduction

Precise subject localization and tracking in video feeds is a cornerstone of modern intelligent systems, with applications spanning far beyond simple surveillance. In film-making and spatial audio, an actor's audio channel is often manipulated to originate from their specific location, requiring exact co-ordinates. In logistics and warehousing, autonomous mobile robots (AMRs) or human workers must be tracked in cluttered environments for fleet management and safety, often where GPS is unavailable. In automated education, lecture capture systems require the camera to autonomously follow a professor pacing across a large stage. Furthermore, in consumer 360° videography and AR/VR, the objective often shifts from simple tracking to "reframing" or aesthetic enhancement, such as automatically keeping an adventure sports enthusiast centered in the frame, or applying computational depth-of-field to focus on the subject and blur out bystanders.

A computer vision algorithm enabling these applications must typically output two distinct data points: a bounding box (or segmentation mask) describing the pixels occupied by the subject for visual effects like cropping or blurring, and the $(x,\ y)$ location of the subject on the physical 2D floor plane. Note that we have substantially more control over the subject of interest in these use-cases than in general surveillance. Since we are concerned about a specific person's movements, we have the liberty to explore other options for locating or tracking the subject, outside of the video feed itself.

In this paper, we ask a simple question: *can we improve locating and tracking a subject's movements within a video if we include knowledge about the subject's movements from another domain?* Of course, we do not know which pixels in the video correspond to the subject; we only have a description of the subject's characteristic in the visual domain (such as "person in blue shirt"). Additionally, we have 2D localization data of the subject's movements with respect to the camera. This location data is obtained from the subject wearing a UWB device that is being tracked from a dual-antenna UWB ranging and angle measurement unit, with its own accuracy limitations. The output of our algorithm should be: (1) a bounding box on the video which moves with the subject, and (2) the $(x,\ y)$ coordinates of the subject, in the 2D floor plane. Fig. 1 depicts these set of inputs and outputs for our *UVTrack360* system.

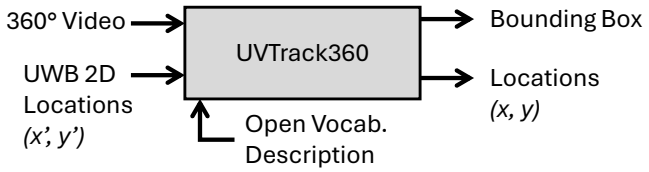

Fig. 1: Input-Process-Output View of UVTrack360.

The core idea of this paper is as follows: In typical filming or monitoring scenarios, a lightweight UWB tag can be attached to the subject of interest. A single dual-antenna UWB anchor collocated with the camera can provide both range and bearing of the subject. This positional cue can then be correlated with the 360° camera's view to assist vision-based localization.

Video processing has advanced significantly over the past decade, and it might seem like a simple problem for vision algorithms to detect the subject and output a bounding box and location. However, low lighting conditions, bright glare, or occlusions prove to be challenging for video-only analysis [1]. Identifying the subject based on open vocabulary descriptions is also quite challenging, but necessary in real-world scenarios.

UWB-only systems also suffer from non-line of sight

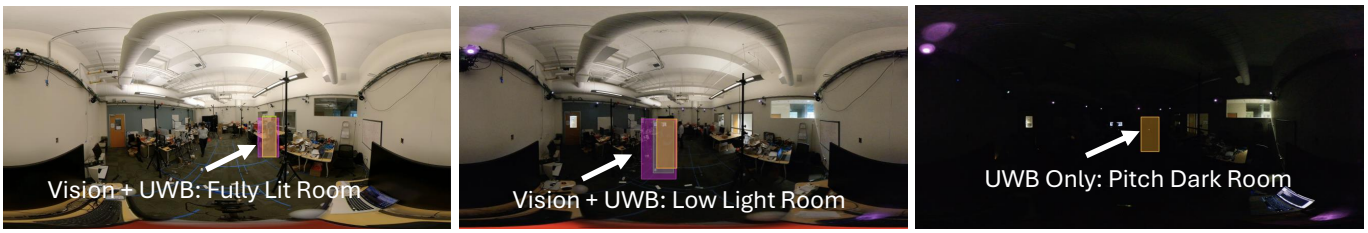

Fig. 2: Lighting conditions significantly affect vision-only detection accuracy. In a dark room, only UWB is being used for determining the subject's location.

(NLOS) and multipath degradation [2], and cannot easily map a bounding box to the video because UWB would not know the size of the subject in the video frame. These problems exacerbate further when a 360° video is used because of lens distortions. Prior work fusing vision and UWB to mitigate single-modality failures exists, but most have relied on monocular narrow field-of-view cameras (which reduces lens distortions) and closed-vocabulary detection [3]–[5]. To the best of our knowledge, a unified framework with the joint integration of 360° vision with UWB, the use of open-vocabulary, and the generation of location output (physical) and panoramic overlay output (virtual), has not yet been developed.

We explore a dual-domain fusion architecture addressing these gaps via three innovations. First, we introduce a confidence-adaptive weighting framework that characterizes specific UWB-vision failure modes. Vision confidence models detection stability and semantic match quality, while UWB confidence integrates distance-dependent accuracy, temporal jitter, and angular geometry, enabling dynamic sensor reliance shifts when one modality degrades, such as UWB in NLOS and vision in low light.

Second, unlike existing solutions [3]–[6], our system processes data in both physical and virtual domains, producing outputs optimized for navigation (metric) and monitoring (overlay) simultaneously.

Third, we employ an efficient propose-then-rank pipeline for open-vocabulary 360° tracking. YOLOv8-nano (YOLOv8n) generates person proposals, Contrastive Language-Image Pretraining (CLIP) ranks them semantically against text queries, and a Discriminative Correlation Filter with Channel and Spatial Reliability (CSRT) maintains frame-to-frame continuity. This constrains expensive CLIP matching to only the YOLOv8n proposals, achieving low latency and omnidirectional coverage. The complete system, using only a single UWB anchor and 360° camera, runs at 30 FPS on a Raspberry Pi 4, demonstrating improved accuracy over single-modality baselines and a median localization error of 0.36 m. A few sample frames in different lighting conditions are shown in Fig. 2. A complete video demo is available at https://uvtrack360.github.io.

II. Related Work

A. Multimodal Person Localization

1) Vision-based Localization: While vision-only localization systems like ORB-SLAM3 are accurate in ideal environments, they are fundamentally unreliable in variable indoor conditions [1], [7]. Monocular SLAM methods suffer tracking failures in low-light, texture-sparse, and dynamic environments [1], [7]. This tracking loss requires reinitialization, limiting vision-only deployment for robust indoor person localization [1], [7].

2) UWB-based Localization: UWB positioning is robust to lighting, but NLOS propagation severely degrades performance, causing errors of over 2 meters in NLOS environments [2]. Recent NLOS mitigation techniques such as anchor link exploitation [2] and adaptive loss functions [8], provide only partial error reduction [2], [8], [9].

3) Vision-RF Fusion Systems: Vision-RF fusion leverages complementary strengths, such as Peng *et al.* combining YOLOv3 and DeepSort with UWB TDOA using Hungarian matching and federated Kalman filtering [3], and Liu *et al.* fusing ORB-SLAM with EKF-processed UWB to achieve 0.2m accuracy and suppress NLOS errors [4]. However, fusion of panoramic vision with UWB localization is unexplored, as prior work uses monocular cameras [3]–[6]. Furthermore, simultaneously generating both metric coordinates for navigation and panoramic overlays for human monitoring from a unified fusion pipeline remains untackled. Additionally, while Peng *et al.* [3] and Ishige *et al.* [5] track multiple people using closed-vocabulary detection, integrating open-vocabulary, text-based person identification with robust UWB localization is also unexplored. Our work addresses all three of these gaps.

B. Open-Vocabulary Person Identification

Open-vocabulary object detection aims to localize and name objects from natural language text input using region-text alignment from vision-language pretraining. However, state-of-the-art models like OWL-ViT [10], and Grounding DINO [11], are computationally intensive. Recent work targets efficiency, such as YOLO-World [12] and Grounding DINO 1.5-Edge [13], which achieve real-time FPS by pre-encoding prompts or using optimization. Our novel three-stage cascade with CSRT, YOLOv8n, and CLIP extends

these strategies, moving beyond prior asynchronous approaches [14] by fully decoupling language encoding into an independent asynchronous stage. Unlike synchronous detectors [10], [11], this allows our high-frequency tracker to maintain continuity without awaiting semantic labels. Furthermore, integrating this on 360° video with UWB fusion for robust spatial localization despite vision failure remains unexplored [7], [12]. We bridge this gap by combining efficient asynchronous open-vocabulary identification with vision-UWB geometric fusion.

C. 360° Vision for Indoor Applications

360° imaging is critical for comprehensive indoor sensing, with recent advances like distortion-aware tokenization [15]. ERP-specific deep learning has matured using customized augmentation, as shown by Ardy *et al.* [16] and distortion-handling methods like EDM from Jung *et al.* [17]. While 360° vision excels, open-vocabulary detection on panoramic video is unexplored. Existing detectors from Yang *et al.* [18] and Pokucinski and Mrozek [19] use closed-vocabulary approaches. Our work fills this gap by employing open-vocabulary detection on equirectangular video aided by UWB for robust indoor person localization.

III. System Design

Fig. 3 illustrates the end-to-end processing flow. The outputs of the two sensing arms undergo dual-domain fusion, simultaneously producing a positional estimate and visual overlay.

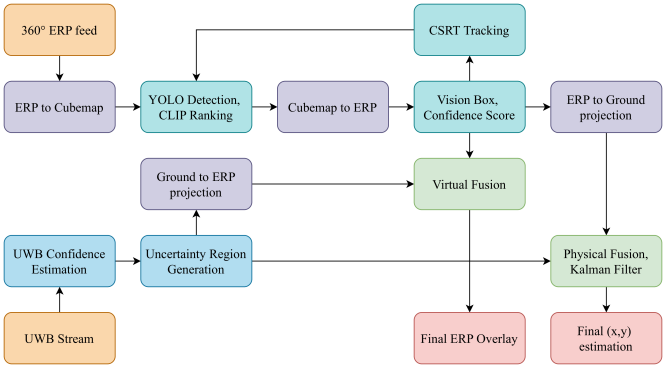

Fig. 3: The end-to-end processing pipeline.

A. Open-Vocabulary Vision Localization

1) Handling 360° Video: The Insta360 ONE X2 captures 360° video in Equirectangular Projection (ERP). Applying CNN-based methods like YOLO directly to ERP fails due to severe geometric distortion, especially at the poles, which violates CNN assumptions [20]. Additionally, the 360° wrap-around seam splits objects at the frame's vertical edges, which is unideal for object detection [21]. To solve this, we project each ERP frame onto a cubemap (six 90° faces), then reproject to ERP after detection and ranking, for tracking and fusion. This method minimizes distortion and eliminates the wrap-around seam [22].

2) Two-Stage Detection Pipeline: Our open-vocabulary localization employs a propose-then-rank architecture balancing efficiency and semantic discrimination. First, YOLOv8n independently processes each cubemap face, detecting all 'person' class instances after resizing and normalization. This stage generates a candidate set of bounding boxes with coordinates and YOLO confidence scores, effectively serving as a class-specific region proposal network.

Second, CLIP refines these generic detections into query-specific identifications. We crop each candidate region from its cubemap face. The user's text query is encoded once by CLIP's text encoder into a 512-dimensional embedding and cached, while each cropped region is processed by CLIP's ViT-B/32 vision encoder into a corresponding 512-dimensional image embedding. Semantic relevance is quantified via cosine similarity between the text and image embeddings, yielding scores in $[-1, 1]$. Candidates are ranked by these CLIP scores, enabling zero-shot identification from natural language descriptions. This two-stage design constrains CLIP's expensive vision encoder to only the YOLO-detected candidates, achieving a significant reduction in vision-language matching cost while maintaining discrimination as seen in Fig. 4, which shows the output on one ERP frame after running with the prompt "woman with red backpack and blue clothes". CSRT is used to track the target on every frame [23], allowing YOLO and CLIP to run only at specified intervals, further conserving resources. If the CSRT track is lost, YOLO and/or CLIP are forced to rerun. To ensure zero frame interruptions, YOLO and CLIP operate as background processes in a multithreaded environment.

Fig. 4: The vision detection pipeline. YOLO detections are in green, and the top-1 CLIP ranking is in yellow.

3) Confidence Estimation: We compute a vision confidence metric as a weighted combination of YOLO detection probability and CLIP text-image similarity as $q_v = 0.3 \cdot c_{\text{YOLO}} + 0.7 \cdot \sigma(s_{\text{CLIP}})$. Here, c_{YOLO} and s_{CLIP} are the confidence scores for the YOLOv8n detection and CLIP ranking respectively, and σ is the sigmoid function, used for normalization. While these metrics may have limitations, they provide useful relative quality indicators [24], [25].

B. Single-Anchor UWB/PDoA Integration

1) Single-Anchor Deployment Advantages: We employ a single-anchor UWB configuration to dramatically reduce deployment complexity compared to conventional multi-anchor systems. Traditional UWB localization requires surveying and installing three or more synchronized anchors for trilateration, imposing substantial infrastructure costs and installation burden that hinder rapid deployment [26], [27]. In contrast, single-anchor systems eliminate the need for multi-anchor baseline calibration and synchronization, enabling portable deployments particularly suited for mobile robotics, temporary monitoring scenarios, and field operations where fixed infrastructure is impractical [27]. While multi-anchor triangulation provides higher absolute accuracy through geometric redundancy, our single-anchor approach using Phase Difference of Arrival (PDoA) achieves competitive 2D positioning by extracting both range and bearing from a single measurement point [26], [28].

2) Phase Difference of Arrival (PDoA) Principle: PDoA enables angle-of-arrival (AoA) estimation by measuring the phase difference between UWB signals arriving at two spatially separated antennas on the anchor node. The Qorvo DWM1002 anchor employs two DW1000 transceivers clocked from a common crystal oscillator to maintain phase coherence [28]. When a signal from the body-worn tag arrives at incident angle θ relative to the antenna baseline, it travels an additional path length $p = d\sin(\theta)$ to reach the second antenna, where d is the antenna separation (2.08 cm for DW1000-based systems) [28], [29]. This path difference manifests as a phase difference that maps to an AoA. Crucially, PDoA requires only a single anchor-tag pair to determine bearing, unlike Time Difference of Arrival (TDoA) systems that necessitate synchronized multi-anchor networks. Combined with Time-of-Flight (ToF) ranging for distance estimation, the anchor produces 2D position estimates without requiring additional infrastructure.

3) UWB Confidence Estimation: To account for UWB measurement degradation from multipath, NLOS, and distance [30], [31], we quantify UWB uncertainty for accurate weighting in fusion. We model UWB confidence q_u based on distance-dependent errors [30], temporal instability from multipath, which we quantify by position variance σ_w over a 10-frame window [31], and angular dependency, which is optimal at broadside 90° and degrades towards endfire with $\sigma_\theta = 30°$ [30]. We combine these factors as follows.

$$q_u = w_d \cdot e^{-\alpha d} + w_s \cdot e^{-\beta \sigma_w} + w_a \cdot \exp\left(-\frac{(\theta - 90°)^2}{2\sigma_\theta^2}\right) \quad (1)$$

Here, d is distance (m), σ_w is position standard deviation, θ is AoA, $\alpha = 0.2\ \mathrm{m}^{-1}$, $\beta = 0.5\ \mathrm{m}^{-1}$, and empirically tuned weights $w_d = 0.5, w_s = 0.3, w_a = 0.2$ sum to unity. This yields $q_u \in [0, 1]$ to modulate the UWB fusion weight inversely with measurement degradation.

4) Uncertainty Region Representation: Our UWB output, the 2D position (x, y) and confidence q_u, is converted into a probabilistic circular region on the floor plane, centered at (x, y) with radius r_u, to enable geometric fusion. The radius is determined by an inverse mapping of the confidence as $r_u = \frac{r_0}{r_0 + q_u}$, where $r_0 = 0.5$ is a calibrated reference distance. This conservative uncertainty model means a high q_u shrinks the radius, reflecting higher confidence, while a low q_u expands it, reflecting greater positional ambiguity. This probabilistic representation enables downstream geometric operations for the fusion pipeline, maintaining mathematical consistency by modeling confidence as spatial uncertainty.

C. Dual-Domain Sensor Fusion

1) Projection Architecture: To enable fusion across coordinate systems, we use bidirectional projection between the ERP image and the floor plane. The UWB uncertainty region on the floor projects to a bounding box in the ERP image by sampling varying points across the disk boundary. Conversely, a vision bounding box in the ERP image back-projects to a circular region on the floor by ray-casting. Confidence estimates propagate through these transforms, enabling weighted fusion.

2) Physical Fusion: Physical domain fusion produces metric floor coordinates for navigation applications. Unlike traditional Kalman filters that continuously assimilate all sensor inputs, our architecture introduces a pre-fusion geometric gating stage to ensure topological consistency. We posit that fusing two contradictory sensor readings, such as a visual detection on one side of a room and a multipath-corrupted UWB signal on the other, results in a mathematically valid but physically impossible location in the center of the room.

To prevent this corruption, the system first evaluates spatial compatibility. We compute the Euclidean distance between the vision and UWB estimates and the Intersection-over-Union (IoU) of their projected uncertainty disks. As detailed in Algorithm 1, fusion occurs only if the estimates overlap or fall within a proximity threshold defined by their combined uncertainty radii. If the inputs are geometrically incompatible, the system rejects the fusion step entirely and executes a "winner-takes-all" selection, isolating the modality with the higher instantaneous confidence. This hard-switching mechanism allows the system to instantaneously reject catastrophic outliers, such as Non-Line-of-Sight (NLOS) UWB reflections or visual tracking hallucinations, rather than averaging them into the trajectory.

We use constant thresholds ($\delta = 1.5$, $\tau = 0.1$) requiring minimal overlap for fusion. If estimates are incompatible, we select the single most confident source. If compatible, a weighted fusion is applied, naturally prioritizing the more confident sensor. The fused position is then temporally smoothed via a Kalman filter, which bridges gaps during sensor dropouts and filters high-frequency jitter, yielding a trajectory that is both continuous and robust to single-modality failures.

3) Virtual Fusion: Virtual domain fusion produces a bounding box overlay in the ERP image space for visualization applications. Unlike physical fusion which outputs a

Algorithm 1 Physical Domain Fusion

Input: Vision region (P_v, r_v, q_v), UWB region (P_u, r_u, q_u)
Output: Fused position P_f, radius r_f, confidence q_f

$d \leftarrow \|P_u - P_v\|$ ▷ Distance between centers
$\text{IoU} \leftarrow \text{CircleIoU}(P_v, r_v, P_u, r_u)$ ▷ Region overlap
if $d > \delta \cdot (r_v + r_u)$ **or** $\text{IoU} < \tau$ **then** ▷ Incompatible
    return region with higher confidence
end if
$w_u \leftarrow \dfrac{q_u}{q_u + q_v}, \quad w_v \leftarrow \dfrac{q_v}{q_u + q_v}$ ▷ Normalize weights
$P_f \leftarrow w_v P_v + w_u P_u$
$q_f \leftarrow \max(q_v, q_u)$
$r_f \leftarrow w_v r_v + w_u r_u$
return (P_f, r_f, q_f)

single metric position, virtual fusion preserves spatial extent by computing the geometric intersection of vision and projected UWB regions. The vision detector provides a bounding box $\mathbf{B}_v$ with confidence q_v, while UWB provides a floor-plane region that we project to the spherical image as a bounding box $\mathbf{B}_u$ with confidence q_u. The fused virtual overlay combines both sources weighted by their confidence as follows.

$$\mathbf{B}_f = \begin{cases} \alpha\mathbf{I} + (1-\alpha)\mathbf{B}_{\text{strong}}, & \text{if area}(\mathbf{I}) > 0 \\ \mathbf{B}_{\text{strong}}, & \text{otherwise} \end{cases} \tag{4}$$

Here, $\mathbf{I} = \text{intersect}(\mathbf{B}_u, \mathbf{B}_v)$ is the overlapping region, $\alpha = \frac{\min(q_v, q_u)}{(q_v + q_u)}$ weights the intersection, and $\mathbf{B}_{\text{strong}}$ is the bounding box from the sensor with $\max(q_v, q_u)$.

When sensor estimates overlap $(\text{area}(\mathbf{I}) > 0)$, the fused box blends the intersection with the high-confidence source's extent, producing a stable overlay that reflects agreement between modalities. When regions do not intersect, we default to the stronger bounding box to maintain a meaningful visual overlay for the operator. The weighting factor α naturally reduces the influence of low-confidence measurements, with the stronger sensor dominating the final overlay boundary while the weaker sensor modulates the extent through intersection geometry. This approach ensures the virtual overlay remains visually coherent even under single-sensor failure conditions.

D. Asynchronous Thread Scheduling

To ensure the system maintains real-time frame rates despite the computational demands of the vision models, we implement a decoupled, hierarchical threading architecture. A single-threaded sequential execution of panoramic projection, object detection, and semantic ranking would exceed the 33 ms per-frame budget required for 30 FPS operation. Consequently, the system separates low-latency tracking operations from high-latency semantic inferences using a non-blocking scheduling strategy.

The primary application thread executes synchronous tasks that require strict temporal alignment with the video feed. This includes frame acquisition, high-frequency Kalman filtering, and lightweight visual tracking via the CSRT tracker. The CSRT tracker provides rapid frame-to-frame position updates for visual continuity while deeper inference models operate in the background. Simultaneously, computationally intensive tasks are offloaded to dedicated daemon worker threads that communicate with the main thread via thread-safe queues. A detection worker handles the transformation of equirectangular frames to cubemaps and the execution of the quantized YOLOv8n model, while a subsequent identification worker processes candidate crops through the CLIP encoder for semantic ranking.

The scheduling logic employs a capacity-constrained dispatch policy to manage computational load. On each frame cycle, the main thread verifies the status of the worker threads. If the workers are idle and a scheduling interval has elapsed, a new inference job is enqueued. Conversely, if the workers are occupied, the main thread bypasses detection for that specific frame, relying solely on the CSRT tracker and UWB data to maintain the output frame rate. Upon completion of a background inference task, the main thread retrieves the results via non-blocking polling and performs a correction step. This involves re-initializing the lightweight CSRT tracker at the semantically verified coordinates provided by the deep learning pipeline, effectively correcting tracker drift and confirming target identity without interrupting the real-time display.

IV. Implementation

To ensure deployability with low-cost, portable hardware requiring minimal installation, our system uses a Raspberry Pi 4 (8GB RAM) for its low power usage (5W) and 64-bit edge processor [32]. An Insta360 ONE X2 provides 5.7K 360° ERP video [33], selected to ensure full room coverage while avoiding multi-camera calibration. For UWB, a Qorvo PDoA evaluation kit consisting of the DWM1002 and DWM1003 modules enables single-anchor PDoA for 2D localization [29]. Our hardware setup is shown in Fig. 5.

For efficient edge deployment on the Raspberry Pi 4, we use a quantized YOLOv8n with INT8 weights for fast person detection, which reduces memory and latency while balancing accuracy on ARM architecture [34]. For zero-shot identification, we use a distilled CLIP (ViT-B/32) variant with reduced hidden dimensions, cutting inference time while maintaining its language query capability [35], [36].

In order to ensure accurate evaluations, we synchronize vision and UWB streams via acoustic calibration, aligning audio spike events recorded by both systems to establish a common temporal reference. While our architecture supports real-time operation, evaluation uses file-based replay since the Insta360 ONE X2 lacks a real-time ERP output interface. Per-frame processing completes in 25 ms on average, as seen in

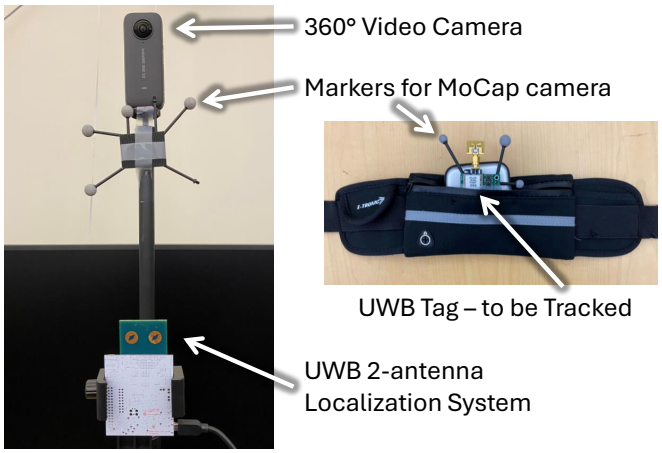

Fig. 5: Our hardware setup consists of a 360° camera, a UWB anchor, and a wearable UWB tag, tracked using motion capture markers.

the evaluation section, well within the 33 ms budget required for real-time operation at 30 FPS.

For the calibration of constants/parameters, all hyperparameters were fixed once using a small calibration set of prior measurements, and then held constant across all scenarios to ensure stability and robustness.

V. Evaluation

A. Experimental Setup

We collected footage in an environment with multiple clutter configurations to create occlusions. Trials were performed under 3 lighting conditions, namely lights on (ideal conditions), lights dimmed (low light conditions), and lights off (pitch dark conditions). A total of 40 videos were collected, each of approximate length 20 seconds, spanning various combinations of lighting, occlusions, and clothing. The UWB anchor and camera are mounted on the same tripod fixture near one of the room's walls. The camera is placed 2 m above the ground plane, and the UWB anchor is 15cm directly below the camera.

Ground truth positions were obtained via the OptiTrack motion capture system (MoCap) with markers at chest height. Twelve motion capture cameras evenly spaced along the room's walls captured the full motion of the MoCap markers and internally calculated the 3D position of the target, and the $(x,\ y)$ coordinates parallel to the ground plane from that position were used as the ground truth. The MoCap setup was calibrated with an OptiTrack calibration wand and ground plane before each trial. We synchronize the MoCap ground truth data with the sensor measured data by logging timestamps of UWB/camera activation and trimming appropriately from the MoCap data. Ground truth overlays were obtained by evenly sampling 30 frames per video and annotating bounding boxes around the target in ERP.

Each trial included the target along with one distractor, to evaluate the open-vocabulary efficacy of the vision pipeline. The target wore different types and colors of clothing across trials. Both the target and distractor walked in diverse trajectories throughout the trials, such as straight lines, loops, jagged/zig-zag paths, and a multitude of other walking styles. Walking speed and relative proximity of the target and distractor varied across and within the trials as well.

For baselines, we compare Vision-only, UWB-only, and our confidence-based fusion solution combining both modalities. The same projection algorithms and architectures explained in the System Design section were used for all three, to facilitate a fair comparison between the baselines and our fused solution. We also run an ablation study on our fusion algorithm, comparing two simpler forms of fusion, detailed below, to our algorithm.

While our architecture supports real-time operation, evaluation uses file-based replay since the Insta360 ONE X2 lacks a real-time ERP output interface. To ensure that this mimics a real-time setup, we process frames serially and carefully measure the per-frame latency, as seen below in the Evaluation section.

B. Physical Domain Localization Accuracy

We compute 2D position error as

$$e_t = \sqrt{(\hat{x}_t - x_t)^2 + (\hat{y}_t - y_t)^2}$$

per frame on the floor plane, where $(\hat{x}_t, \hat{y}_t)$ is the estimated position and (x_t, y_t) is ground truth. Fig. 6 (a) presents the empirical cumulative distribution function (CDF) across tested frames.

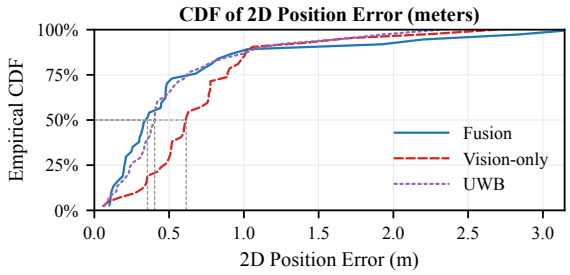

Fig. 6: A CDF of our 2D position error, comparing fusion, UWB, and vision.

Fusion consistently outperforms vision alone across the majority error distribution, and is comparable to UWB alone in terms of localization accuracy. The median error of 0.36 m demonstrates sub-half-meter accuracy suitable for indoor navigation and zone-based analytics applications.

C. Failure Mode Analysis

We define a failure frame for localization as any frame where 2D error exceeds 2.0 m, to reflect drastic errors in localization. For the overlay, we define failure as having an IoU with the ground truth overlay of less than 0.2, or not producing an overlay at all. Table I reports percentage of frames failed for both localization and overlay per scenario.

TABLE I: Failure analysis by light conditions

Conditions	Vision-only		UWB-only		Fusion	
	Loc.	*Overlay*	*Loc.*	*Overlay*	*Loc.*	*Overlay*
Ideal	17.8%	8.3%	20.9%	35.9%	17.4%	5.1%
Low Lighting	21.9%	12.2%	17.7%	34.0%	17.0%	5.0%
Pitch Dark	56.1%	95.6%	18.6%	32.2%	16.4%	26.2%

Fusion demonstrates a clear advantage, especially under challenging light conditions. The benefit is most pronounced in pitch dark environments. In these conditions, vision-only localization fails 56.1% of the time, but the fusion method maintains a much lower 16.4% failure rate, due to the reliance on UWB. Similarly, in pitch dark conditions, the vision-only overlay fails 95.6% of the time, while fusion cuts this failure rate to 26.2%, effectively leveraging the UWB-only performance (32.2%) to compensate for the visual failure. Additionally, in ideal conditions, fusion delivers the best results (5.1% overlay failure) by intelligently weighting the stronger vision-only modality (8.3%). Across all scenarios, fusion consistently provides close to the lowest failure rates, proving its robustness.

D. System Performance and Latency

We measure per-stage processing latency on the Raspberry Pi 4 platform running Ubuntu 22.04 (64-bit). Latencies represent the 50th percentile across all frames of steady-state operation. The results are shown in Table II.

TABLE II: Processing latency by light conditions

	Ideal	Low Lighting	Pitch Dark
p50 (ms)	25.04	26.69	1.46 (no vision)

The system achieves 25.04 ms end-to-end latency under ideal conditions, comfortably within the 33.3ms budget for real-time 30 FPS operation. Across all conditions, latency remains below the frame deadline, demonstrating robust real-time performance under varying computational load. The latency for pitch dark conditions is around 1.46 ms as there are no vision detections, so computation time is purely due to UWB.

We also evaluate the computational latency of our end-to-end pipeline against the OWL-ViT (ViT-B/32) Open-Vocabulary Object Detector [10]. Fig. 6 (b) shows a cumulative distribution function of the latency per frame of OWL-ViT and our algorithm, across all video frames. The tests ran on an Apple MacBook Air M4, as OWL-ViT failed to run on the Raspberry Pi 4.

Our method achieves a significantly lower latency, with over 90% of frames processed within around 30 ms, while OWL-ViT requires approximately 90 ms to reach the same percentile. This indicates that our fusion pipeline is substantially faster than standard open vocabulary detectors.

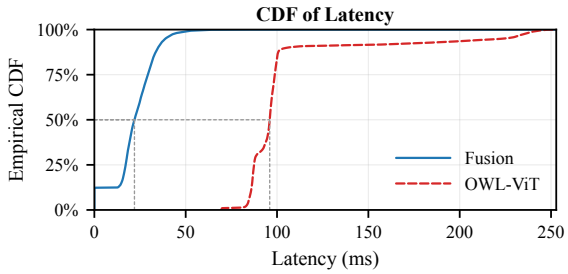

Fig. 7: CDF of computational latency of our fusion algorithm and OWL-ViT in milliseconds.

E. Virtual Domain Overlay Quality

We evaluate the fused ERP bounding box against ground truth boxes annotated on equirectangular frames. A detection is considered correct if its ERP bounding box has IoU ≥ 0.5 with the ground truth box. Our evaluation metrics are percent accuracy (Acc@0.5) and median IoU over correct frames. We compare vision-only, UWB-only, and fused overlays across scenarios, in Table III.

TABLE III: Virtual overlay accuracy

Conditions	Vision-only		UWB-only		Fusion	
	Acc.	Med. IoU	Acc.	Med. IoU	Acc.	Med. IoU
Ideal	59.3%	0.75	32.6%	0.63	88.2%	0.84
Low lighting	61.0%	0.72	28.7%	0.61	87.1%	0.87
Pitch-dark	0.7%	0.76	27.4%	0.68	61.3%	0.74

Fused overlays achieve 88.2% ideal accuracy, significantly outperforming vision-only (60.1%) and UWB-only (32.3%), with a high 0.84 median IoU indicating well-aligned overlays. In low lighting, fusion remains strong at 87.1% accuracy and 0.87 median IoU, exceeding vision-only (61.0%) and UWB-only (28.7%). In pitch-darkness, fusion sustains 61.3% accuracy even as vision collapses (0.7%), effectively leveraging UWB (27.4%). Across conditions, fusion's median IoU is consistently higher (0.84 overall) than both vision-only (0.74) and UWB-only (0.63), confirming tighter and more stable bounding boxes.

F. 2D Localization Accuracy

We evaluate the estimated 2D position coordinates (x, y) against the ground truth trajectory. Accuracy is measured using the Euclidean distance between the estimated position and the ground truth. Our primary metrics are Median Error in meters, representing the median positional deviation, and Root Mean Square Error (RMSE), which highlights stability by penalizing large outliers. We compare vision-only, UWB-only, and fused positioning across scenarios, in Table IV.

The fused localization achieves an overall median error of 0.36 m, outperforming vision-only (0.61 m) and UWB-only (0.40 m), with an overall RMSE of 0.96 m indicating robust tracking without major drift. Under ideal conditions, fusion minimizes error to 0.22 m (vs. 0.40 m vision-only and 0.79 m UWB-only) with an RMSE of 0.67 m. In low

TABLE IV: 2D Localization Accuracy (in meters)

Conditions	Vision-only		UWB-only		Fusion	
	Median	RMSE	Median	RMSE	Median	RMSE
Ideal	0.40	0.76	0.79	1.09	0.22	0.67
Low lighting	0.48	0.79	0.90	1.15	0.34	1.17
Pitch-dark	2.27	2.37	0.89	1.07	0.44	1.09
Overall	0.61	0.90	0.40	1.12	0.36	0.96

lighting, fusion maintains improved accuracy with a 0.34 m median error compared to 0.48 m (vision-only) and 0.90 m (UWB-only). Its RMSE is 1.17 m (vs. 0.79 m vision-only and 1.15 m UWB-only), indicating higher variability despite lower average error. In pitch-darkness, fusion compensates for visual failure, reducing the vision-only median error of 2.27 m by leveraging UWB, and achieves a median error of 0.44 m (UWB-only: 0.89 m) with an RMSE of 1.09 m. Across all conditions, fusion provides the lowest median error overall and a comparable RMSE, supporting a more reliable trajectory than either single-sensor approach.

G. Ablation Study: Fusion Strategies

To validate the necessity of our confidence-adaptive weighting mechanism, we compare the physical domain localization accuracy of *UVTrack360* against two alternative fusion strategies:

1) *Naive Fusion:* A static averaging approach where weights are fixed at $w_v = w_u = 0.5$, ignoring sensor quality.
2) *Selection-Only:* A "winner-takes-all" approach that exclusively selects the position estimate of the modality with the highest instantaneous confidence without blending.

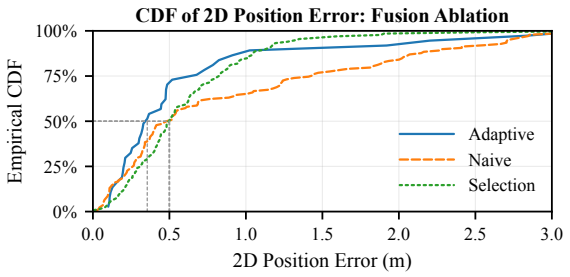

Fig. 8: CDF of position error comparing Naive Fusion (50/50), Selection-Only (Winner-takes-all), and our Adaptive Fusion algorithm.

Fig. 8 presents the Cumulative Distribution Function (CDF) of the 2D position error for these three strategies. The *Naive Fusion* approach performs poorly, particularly in the tail of the distribution (errors > 1.0 m), where the curve flattens significantly compared to the other methods. This confirms that blindly averaging a robust sensor with a failing one (e.g., vision in pitch dark or UWB in severe NLOS) corrupts the final estimate, effectively dragging the tracked position away from the ground truth.

The *Selection-Only* strategy offers improved robustness over naive averaging in the long tail by rejecting low-confidence inputs, approaching 100% precision faster than the naive approach. However, it results in a higher median error (≈ 0.5 m, indicated by the rightmost vertical dashed line) compared to our method. This is attributed to "switching jitter"; when q_v and q_u are comparable, the system oscillates between sensors, introducing discontinuities in the trajectory.

Our *Adaptive Fusion* achieves the best performance, maintaining the highest probability of lower error throughout the distribution (median error ≈ 0.36 m, indicated by the leftmost vertical dashed line). It effectively combines the benefits of both baselines: it behaves like the selection method during single-modality failure (by assigning near-zero weight to the failed sensor) but exploits the variance reduction properties of weighted averaging when both sensors are providing valid data.

VI. Concluding Remarks

In this work, we introduced *UVTrack360*, a dual-domain fusion architecture that integrates single-anchor UWB localization with open-vocabulary 360° tracking. By employing a confidence-adaptive weighting framework, our system effectively mitigates the individual failure modes of each modality—compensating for visual occlusions with RF ranging and correcting UWB multipath jitter with semantic visual confirmation. The result is a unified pipeline that robustly outputs both metric coordinates for navigation and semantic overlays for monitoring, achieving sub-meter accuracy even in challenging lighting conditions.

Despite these advances, several concrete limitations remain. First, while our single-anchor formulation reduces infrastructure costs, it relies on Phase Difference of Arrival (PDoA) for bearing estimation. As noted in our confidence model, PDoA accuracy is geometrically dependent, degrading at endfire angles (0° and 180°) compared to the optimal range around 90°. Second, while our processing pipeline achieves real-time latency on edge hardware, the current commercial 360° camera hardware (Insta360 ONE X2) lacks a native real-time ERP streaming interface, necessitating file-based replay for evaluation. Finally, our fusion relies on static extrinsic calibration between the camera and anchor, and in long-term deployments, mechanical shifts could induce drift that degrades the fusion geometry.

Future work will address these challenges through three key avenues. First, we plan to implement UWB-guided visual attention, using the RF angle-of-arrival to dynamically crop the high-resolution 360° feed. This would allow the computationally expensive vision pipeline to process only relevant sectors of the panorama, significantly reducing latency and energy consumption, allowing for more powerful model use and higher accuracy. Second, we aim to extend the system to multi-object tracking by incorporating Time Division Multiple Access (TDMA) protocols for multiple tags, utilizing the UWB identity to disambiguate visually similar subjects.

Finally, we intend to explore cross-modal self-supervision, where the robust UWB signal serves as a weak teacher to automatically generate pseudo-labels for the vision model. This would allow the system to iteratively fine-tune its open-vocabulary detector to the specific environment and subject without manual annotation.

REFERENCES

[1] P. Chen, X. Zhao, L. Zeng *et al.*, "A review of research on slam technology based on the fusion of lidar and vision," *Sensors*, vol. 25, no. 5, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/5/1447

[2] Y. Chen, J. Wang, and J. Yang, "Exploiting anchor links for nlos combating in uwb localization," *ACM Trans. Sen. Netw.*, vol. 20, no. 3, May 2024. [Online]. Available: https://doi.org/10.1145/3657639

[3] P. Peng, C. Yu, Q. Xia, Z. Zheng, K. Zhao, and W. Chen, "An indoor positioning method based on uwb and visual fusion," *Sensors*, vol. 22, no. 4, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/4/1394

[4] F. Liu, J. Zhang, J. Wang, H. Han, and D. Yang, "An uwb/vision fusion scheme for determining pedestrians' indoor location," *Sensors*, vol. 20, no. 4, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/4/1139

[5] M. Ishige, Y. Yoshimura, and R. Yonetani, "Opt-in camera: Person identification in video via uwb localization and its application to opt-in systems," 2025. [Online]. Available: https://arxiv.org/abs/2409.19891

[6] C. Wang, H. Zhang, T.-M. Nguyen, and L. Xie, "Ultra-wideband aided fast localization and mapping system," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Sep. 2017, p. 1602–1609. [Online]. Available: http://dx.doi.org/10.1109/IROS.2017.8205968

[7] A. Tourani, H. Bavle, J. L. Sanchez-Lopez, and H. Voos, "Visual slam: What are the current trends and what to expect?" *Sensors*, vol. 22, no. 23, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/23/9297

[8] Y. Liu, E. Hu, Y. Chen, and C. Guo, "Neurodynamic robust adaptive UWB localization algorithm with NLOS mitigation," *Scientific Reports*, vol. 15, no. 1, p. 14271, Apr. 2025.

[9] J. Yang, B. Dong, and J. Wang, "Vuloc: Accurate uwb localization for countless targets without synchronization," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 6, no. 3, Sep. 2022. [Online]. Available: https://doi.org/10.1145/3550286

[10] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn *et al.*, "Simple open-vocabulary object detection," in *Computer Vision – ECCV 2022*. Berlin, Heidelberg: Springer-Verlag, 2022, p. 728–755. [Online]. Available: https://doi.org/10.1007/978-3-031-20080-9_42

[11] S. Liu *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," in *Computer Vision – ECCV 2024*. Cham: Springer Nature Switzerland, 2025, pp. 38–55.

[12] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *CVPR*, June 2024, pp. 16 901–16 911.

[13] T. Ren, Q. Jiang *et al.*, "Grounding dino 1.5: Advance the "edge" of open-set object detection," 2024. [Online]. Available: https://arxiv.org/abs/2405.10300

[14] S. Cho, S. Lee, M. Lee, J. Lee, and S. Lee, "Find first, track next: Decoupling identification and propagation in referring video object segmentation," 2025. [Online]. Available: https://arxiv.org/abs/2503.03492

[15] N. Wakai, S. Sato, Y. Ishii, and T. Yamashita, "Panoramic distortion-aware tokenization for person detection and localization using transformers in overhead fisheye images," 2025. [Online]. Available: https://arxiv.org/abs/2503.14228

[16] R. D. Ardy, A. Yuniarti, and C. A. Sari, "Enhancing face detection performance in 360-degree video using yolov8 with equirectangular augmentation techniques," *JUTI: Jurnal Ilmiah Teknologi Informasi*, vol. 23, no. 1, p. 81–94, Feb. 2025. [Online]. Available: https://juti.if.its.ac.id/index.php/juti/article/view/1255

[17] D. Jung, J. Choi, Y. Lee, S. Jeong, T. Lee, D. Manocha, and S. Yeon, "Edm: Equirectangular projection-oriented dense kernelized feature matching," in *CVPR*, June 2025, pp. 6337–6347.

[18] W. Yang, Y. Qian, J.-K. Kämäräinen, F. Cricri, and L. Fan, "Object detection in equirectangular panorama," in *2018 ICPR*, 2018, pp. 2190–2195.

[19] S. Pokuciński and D. Mrozek, "Yolo-based object detection in panoramic images of smart buildings," in *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*, 2023, pp. 1–9.

[20] H. Chen, J. Liu, M. Li, K. Jiang, Z. Xu, R. Sun, and Y. Sui, "Equirectangular image construction method for standard cnns for semantic segmentation," 2023. [Online]. Available: https://arxiv.org/abs/2310.09122

[21] A. Christensen, N. Mojab, K. Patel, K. Ahuja, Z. Akata, O. Winther, M. Gonzalez-Franco, and A. Colaco, "Geometry fidelity for spherical images," 2024. [Online]. Available: https://arxiv.org/abs/2407.18207

[22] D. Pio and E. Kuzyakov. Under the hood: Building 360 video. Engineering at Meta. [Online]. Available: https://engineering.fb.com/2015/10/15/video-engineering/under-the-hood-building-360-video/

[23] A. Lukežič, T. Vojíř, L. Čehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter tracker with channel and spatial reliability," *International Journal of Computer Vision*, vol. 126, no. 7, p. 671–688, Jan. 2018. [Online]. Available: http://dx.doi.org/10.1007/s11263-017-1061-3

[24] Ultralytics, "Performance metrics deep dive," *Ultralytics YOLO Docs*, accessed: 2025-10-25. [Online]. Available: https://docs.ultralytics.com/guides/yolo-performance-metrics/#object-detection-metrics

[25] T. Team, "Clip score," *TorchMetrics Documentation*, Lightning AI, version 1.8.2. Accessed: 2025-10-25. [Online]. Available: https://lightning.ai/docs/torchmetrics/stable/multimodal/clip_score.html

[26] B. Van Herbruggen, S. Luchie, R. Wilssens, and E. De Poorter, "Single anchor localization by combining uwb angle-of-arrival and two-way-ranging: an experimental evaluation of the dw3000," in *2024 International Conference on Localization and GNSS (ICL-GNSS)*, 2024, pp. 1–7.

[27] S. O. Schmidt, M. Cimdins, F. John, and H. Hellbrück, "Salos—a uwb single-anchor indoor localization system based on a statistical multipath propagation model," *Sensors*, vol. 24, no. 8, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/8/2428

[28] M. Heydariaan, H. Dabirian, and O. Gnawali, "Anguloc: Concurrent angle of arrival estimation for indoor localization with uwb radios," in *2020 DCOSS*, 2020, pp. 112–119.

[29] N. Smaoui, M. Heydariaan, and O. Gnawail, "Single-antenna aoa estimation with uwb radios," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, 2021, pp. 1–7.

[30] R. Juran, P. Mlynek *et al.*, "Hands-on experience with uwb: Angle of arrival accuracy evaluation in channel 9," in *2022 ICUMT*, 2022, pp. 45–49.

[31] A. Arun, S. Saruwatari, S. Shah, and D. Bharadia, "Xrloc: Accurate uwb localization to realize xr deployments," in *SenSys '23'*. New York, NY, USA: Association for Computing Machinery, 2024, p. 459–473. [Online]. Available: https://doi.org/10.1145/3625687.3625810

[32] Raspberry Pi (Trading) Ltd., "Raspberry pi 4 model b," Raspberry Pi (Trading) Ltd., Datasheet Release 1.1, Mar. 2024. [Online]. Available: https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf

[33] Insta360 one x2 — waterproof 360 action camera. Insta360. [Online]. Available: https://www.insta360.com/product/insta360-onex2

[34] M. T. Okano, W. A. C. Lopes *et al.*, "Edge ai for industrial visual inspection: Yolov8-based visual conformity detection using raspberry pi," *Algorithms*, vol. 18, no. 8, 2025. [Online]. Available: https://www.mdpi.com/1999-4893/18/8/510

[35] OpenAI, "Clip: Connecting text and images," https://openai.com/index/clip/, Jan. 2021, accessed: 2025-10-25.

[36] L. Zhong, A. Ghazal, J.-J. Wan, F. Zilly, P. Mackens, J. Vollrath, and B. Coseriu, "Clip4retrofit: Enabling real-time image labeling on edge devices via cross-architecture clip distillation," in *CVPR Workshops*, June 2025, pp. 3868–3876.