Applied Artificial Intelligence Research

INNOVIAIR

A Comprehensive Survey of Deep Learning–Based Object Tracking for Augmented Reality in Complex Real-World Scenes

Duoduo Mou[a,*]

[a]*Faculty of Computing, Universiti Teknologi Malaysia, Skudai, Johor 81310, Malaysia*

ARTICLEINFO

Keywords:
Augmented Reality
Object Tracking
Deep Learning
Complex Scenes
Real-Time Systems
Mobile and Wearable AR

ABSTRACT

Augmented Reality (AR) systems critically depend on accurate, temporally stable, and computationally efficient object tracking to maintain geometric alignment and perceptual coherence between virtual content and the physical world. As AR technologies transition from controlled laboratory prototypes to large-scale deployment in industrial, medical, and consumer scenarios, tracking must operate robustly in complex real-world environments characterized by dynamic objects, occlusions, illumination changes, fast motion, and strict computational constraints. Traditional geometry-driven tracking pipelines often degrade under such conditions, motivating increased adoption of deep learning based approaches. This survey provides a comprehensive review of deep learning based object tracking for AR in complex real-world scenes, with particular emphasis on system-level considerations. Object tracking is treated as a central perception primitive that underpins stable AR experiences and interacts tightly with modules such as visual simultaneous localization and mapping, depth and geometry estimation, and semantic scene understanding. In contrast to prior surveys that emphasize algorithmic accuracy in isolation, we explicitly analyze AR-specific constraints including real-time latency, temporal stability, energy efficiency, long-term robustness, and deployment on mobile and wearable platforms. We review major tracking paradigms, representative datasets and benchmarks, AR-centric evaluation criteria, and common failure modes observed in practice, and we outline future research directions toward scalable, reliable, and trustworthy AR tracking systems.

1. Introduction

Augmented Reality (AR) aims to seamlessly integrate digital content into the physical world, enabling spatially aligned perception and interaction across a wide range of application domains such as industrial maintenance, medical navigation, education, cultural heritage, and entertainment[1]. Unlike purely virtual environments, AR systems must continuously sense and interpret the real world while maintaining consistent geometric relationships between virtual and physical entities. Within this pipeline, object tracking serves as a central perception capability. Even small inaccuracies may accumulate into visible drift, jitter, or misalignment, ultimately degrading immersion and user trust. Early AR systems relied on handcrafted features, fiducial markers, and geometric optimization techniques, but these assumptions break down in real-world environments characterized by dynamic objects, occlusions, illumination variation, and high-speed motion, compounded by strict constraints on latency, power, and thermal dissipation on mobile and head-mounted platforms[2]. Deep learning has emerged as a transformative paradigm for improving robustness under such conditions and has reshaped multiple components of the AR perception stack, including detection, tracking, localization, depth estimation, and semantic scene understanding[3]. However, significant gaps remain between academic benchmarks and practical deployment; conventional datasets and evaluation protocols rarely capture AR-specific requirements such as temporal stability, long-term consistency, failure recovery, and resource efficiency. These challenges motivate a system-oriented examination of deep learning based object tracking for AR in realistic conditions.

The remainder of this survey is organized as follows. Section 2 analyzes the complexity of real-world AR scenes and discusses how environmental variability, interaction dynamics, and system constraints fundamentally shape object tracking requirements. Section 3 reviews major deep learning – based object tracking paradigms from a system-

* Corresponding author.
E-mail address: mouduoduo@graduate.utm.my.
https://doi.org/10.65455/ddjkhy91
Received 31 January 2026; Received in revised form 10 March 2026; Accepted 17 March 2026; Available online 24 March 2026

oriented perspective, highlighting their respective strengths, limitations, and suitability for AR deployment. Section 4 examines the coupling between object tracking, visual SLAM, and depth estimation, emphasizing how inter-module interactions influence overall system stability. Section 5 surveys representative datasets, benchmarks, and AR-centric evaluation protocols, with a focus on their relevance to long-term, real-time AR tracking. Section 6 discusses deployment challenges and engineering considerations for practical AR systems, followed by an analysis of common failure modes in complex real-world scenarios in Section 7. Finally, Section 8 outlines future research directions toward scalable, reliable, and trustworthy AR object tracking.

2.Complexity in real-world AR scenes and its implications for object tracking

Object tracking in augmented reality operates under a set of environmental and system conditions that fundamentally differ from those assumed in conventional visual tracking benchmarks. Rather than serving as background context, these complexities directly determine the design requirements, failure modes, and system-level behavior of AR tracking pipelines. Understanding their implications is therefore essential for interpreting the limitations of existing methods and motivating the system-level perspectives discussed in subsequent sections.

2.1.Real-world constraints for practical AR tracking

Real-world AR deployment introduces challenges that extend far beyond those captured in conventional tracking benchmarks. AR systems must operate across heterogeneous environments with rapidly changing illumination, shadows, reflections, and seasonal variations, all of which induce strong appearance non-stationarity and can cause appearance model collapse that directly manifests as misalignment or jitter of virtual content[4]. In addition, AR scenes often contain dynamic and non-rigid objects, articulated humans, and deformable materials, while user interaction introduces unpredictable occlusions and discontinuities that increase the risk of drift and identity switches. These factors require tracking methods that support re-identification, long-term memory, and interaction-aware modeling to prevent persistent misalignment after occlusion. Many AR applications further demand long-term temporal consistency over minutes or hours. Under such conditions, small frame-wise errors may accumulate into perceptible drift, elevating temporal stability from a secondary metric to a primary design objective. Finally, AR platforms such as mobile phones and head-mounted displays impose strict constraints on computation, memory, and energy consumption, requiring tracking to coexist with localization, perception, and rendering under real-time latency and sustained operation[5]. These challenges highlight that practical AR tracking must jointly satisfy accuracy, temporal stability, and resource efficiency to maintain reliable user experiences.

2.2.Summary: complexity as a design driver for AR tracking

Taken together, environmental variability, dynamic interaction, long-term operation, and system constraints fundamentally distinguish AR tracking from conventional visual tracking. These factors motivate a shift from isolated algorithmic optimization toward system-level tracking design, where robustness, stability, and integration play a central role. The following sections build upon this complexity analysis to examine how deep learning–based tracking paradigms address—or fail to address—these AR-specific requirements.

Table 1. AR scene complexity vs. tracking design requirements

AR Scene Complexity	Impact on Tracking	Required Capability
Illumination variability	Appearance collapse	Invariant features
Dynamic objects	Identity switches	Re-identification
Long-term operation	Drift	Temporal stability
Fast motion	Blur	Motion-aware tracking
Occlusion	Target loss	Recovery mechanisms
Resource limits	Latency	Lightweight models

3.Deep learning paradigms for object tracking in augmented reality

Deep learning–based object tracking methods for augmented reality can be broadly categorized according to their modeling assumptions, temporal scope, and degree of integration with the AR perception pipeline. From a system perspective, these paradigms differ not only in tracking accuracy but also in latency, temporal stability, recoverability from failure, and suitability for deployment on resource-constrained devices. Understanding these trade-offs is essential for selecting and designing tracking solutions that meet the stringent requirements of real-world AR systems.

Table 2. Deep learning–based object tracking paradigms for AR under system constraints

Tracking Paradigm	Core Idea	Strengths for AR	Limitations	Typical AR Scenarios
Discriminative (Siamese-based)	Similarity learning	Fast, low latency	Drift under occlusion	Mobile AR
Correlation Filter + DL	Efficient filtering	Real-time, low power	Weak long-term robustness	Lightweight AR
Generative / Model-based	Explicit geometry	Accurate pose	High computation	Industrial / Medical AR
Detection-assisted	Re-detection	Long-term robustness	Latency	Navigation AR
Self-/Unsupervised	Temporal consistency	Adaptive	Stability risk	Personalized AR
Hybrid Geometric–Learning	DL + geometry	System stability	Complexity	Safety-critical AR
Continual and Lifelong Tracking	Online adaptation with memory or regularization	Long-term robustness, personalization	Catastrophic forgetting, drift risk, compute overhead	Persistent and personalized AR

3.1. Discriminative tracking frameworks

Discriminative tracking approaches formulate object tracking as a classification or regression problem. Their primary objective is to distinguish the target object from the background in each frame. Siamese network based trackers have become particularly influential within this paradigm, as they learn a similarity function between a target template and candidate regions, enabling efficient feed-forward inference and strong generalization across object categories[6]. For AR applications, discriminative trackers offer lightweight and highly parallelizable inference pipelines suitable for real-time deployment on mobile GPUs or neural processing units, and the separation between offline training and online inference aligns well with AR system design, where models are trained in advance and repeatedly executed on-device. The template-based formulation further allows rapid initialization when a new object is selected by the user. However, these trackers also exhibit limitations that become pronounced in AR scenarios, since many are optimized for short-term tracking and rely heavily on appearance similarity; in complex scenes with prolonged occlusion, viewpoint change, or dynamic distractors, they may gradually drift from the true target, leading to virtual objects that slide or vibrate relative to the physical world, which is highly perceptible to users. Recent research has explored mechanisms to mitigate these effects, including temporal attention modules that weigh historical observations, memory-augmented architectures that maintain dynamic target representations, and temporal consistency losses that penalize abrupt state changes. Although these techniques improve robustness and stability, they introduce additional computational and memory overhead that must be balanced against the real-time constraints of AR systems.

3.2. Correlation filter-enhanced and lightweight trackers

Correlation filter–based tracking has a long history in real-time vision systems due to its computational efficiency. In recent years, deep learning has been combined with correlation filters to improve feature representation while preserving fast inference. Deep correlation filter trackers leverage learned convolutional features that are more robust to illumination change and deformation than handcrafted descriptors[7]. In AR contexts, correlation filter–enhanced trackers are particularly attractive for low-power devices, such as lightweight head-mounted displays or smartphones operating under thermal constraints. Their frequency-domain optimization enables high frame rates with limited computational resources. However, their reliance on local appearance models makes them vulnerable to long-term occlusion and large appearance variation. Hybrid approaches that integrate deep feature extraction with adaptive correlation filters have been proposed to mitigate these limitations. By periodically refreshing the target model using detection cues or semantic priors, such trackers can maintain efficiency while improving long-term robustness. Nevertheless, careful system-level design is required to prevent overfitting and model corruption during online updates.

3.3. Generative and model-based tracking paradigms

Generative tracking approaches aim to explicitly model the appearance, geometry, or motion of the target object over time. Unlike discriminative methods, which focus on separating the target from the background, generative trackers seek to reconstruct observations by fitting an internal object model[8]. Deep generative models, including variational autoencoders, neural radiance fields, and implicit surface representations, have enabled increasingly expressive object models. In AR applications that require precise spatial alignment, such as industrial assembly guidance or medical navigation, generative tracking offers significant advantages. By modeling object geometry explicitly, these methods support accurate pose estimation and realistic occlusion reasoning. Neural implicit representations, in particular, allow continuous modeling of object shape across viewpoints, reducing sensitivity to partial occlusion and sparse observations[9]. Despite their strengths, generative approaches pose significant challenges for real-time AR deployment. High-fidelity object models are computationally expensive to optimize and often require iterative refinement. Moreover, generative trackers are sensitive to initialization quality; poor initialization can lead to convergence to incorrect local minima. To address these issues, recent work explores hybrid strategies that activate generative modeling selectively, for example during periods of occlusion or large viewpoint change, while relying on faster discriminative tracking during normal operation.

3.4. Detection-assisted and long-term tracking frameworks

Detection-assisted tracking frameworks combine object detection with continuous tracking to improve robustness over long durations. In these systems, a detector periodically re-localizes the target, providing a mechanism for recovery after tracking failure or extended occlusion. This paradigm is particularly relevant for AR applications that involve long-term interaction with the environment, such as navigation, inspection, or collaborative tasks. From a system perspective, detection-assisted tracking introduces trade-offs between robustness and latency. Object detectors are typically more computationally expensive than trackers and may not run at full frame rate. As a result, many AR systems adopt asynchronous designs, where detection operates at a lower frequency while tracking maintains high-rate updates[10]. Coordinating these modules requires careful synchronization to avoid inconsistencies that could destabilize AR alignment. Hybrid approaches further integrate geometric constraints, such as camera motion estimates from SLAM or depth information from sensors, to constrain detection and tracking. By leveraging multi-source information, these systems achieve improved spatial consistency and resilience to appearance ambiguity. However, increased system complexity also raises challenges related to failure diagnosis and recovery.

3.5.Self-supervised and unsupervised tracking methods

The acquisition of large-scale, densely annotated tracking datasets suitable for AR is costly and often impractical. Self-supervised and unsupervised tracking methods address this challenge by exploiting inherent structure in video data. Temporal coherence, motion consistency, and multi-view geometry provide rich supervisory signals without requiring manual annotation[11]. For AR systems that operate in diverse and changing environments, self-supervised learning offers the potential for on-device adaptation. Trackers can refine their representations based on the specific objects and scenes encountered by users, improving robustness to domain shift. However, self-supervised methods also introduce risks of instability, particularly when erroneous predictions are reinforced during online learning. Recent work explores mechanisms for stabilizing self-supervised tracking, including confidence-based update rules and periodic re-initialization using more reliable cues[12]. While promising, these approaches remain an active area of research, especially in the context of safety-critical AR applications.

3.6.Continual and lifelong tracking in AR systems

Augmented reality systems are often expected to operate continuously over extended periods and across diverse environments, tasks, and object instances. Unlike short-term tracking scenarios, AR applications frequently involve repeated exposure to changing illumination conditions, evolving backgrounds, and user-specific interaction patterns. Under such conditions, trackers trained purely in an offline and static manner may gradually suffer from performance degradation when confronted with domain shifts that were not adequately represented during training. Continual and lifelong tracking paradigms aim to address this limitation by enabling tracking models to incrementally adapt to new observations while preserving previously acquired knowledge.

From a system-level perspective, continual tracking is particularly relevant to AR because long-term robustness and personalization are central to user experience. As AR systems are increasingly deployed on personal devices and head-mounted platforms, they must accommodate variations in user behavior, object appearance, and environmental context over time. In principle, lifelong adaptation allows tracking models to refine their internal representations based on deployment-specific data, improving resilience to appearance variation and reducing the need for frequent offline retraining. Such capabilities are especially valuable for persistent AR scenarios, where virtual content must remain stably anchored to physical objects across prolonged usage sessions.

However, continual learning also introduces challenges that are more pronounced in AR tracking than in many other vision tasks. Unconstrained online adaptation can lead to catastrophic forgetting, where previously learned representations are overwritten, or to gradual model contamination caused by occlusions, distractors, or inaccurate predictions. In AR systems, these effects are particularly problematic because even small degradations in tracking stability may accumulate into visible drift or jitter of virtual content, directly undermining spatial coherence and user trust. As a result, adaptability alone is insufficient; AR tracking systems must prioritize stability, predictability, and bounded model updates.

To mitigate these risks, recent approaches explore mechanisms such as memory replay, regularization-based constraints, and modular architectures that isolate long-term knowledge from short-term adaptation. Confidence-aware update strategies have also been proposed to restrict online learning to high-reliability observations, thereby reducing the likelihood of reinforcing erroneous states. Despite these efforts, the integration of continual learning into real-time AR tracking pipelines remains limited, largely due to concerns regarding computational overhead, system complexity, and failure diagnosability. Balancing adaptability and stability therefore remains a central open challenge for lifelong AR tracking systems and an important direction for future research[13].

3.7.Hybrid geometric–learning tracking paradigms

Hybrid geometric–learning tracking paradigms combine explicit geometric constraints with data-driven representations to leverage the complementary strengths of model-based and learning-based approaches. In these methods, geometric priors such as camera motion models, multi-view consistency, or depth constraints are integrated with deep feature representations to guide object localization and temporal association. This hybrid design is particularly relevant to AR systems, where geometric consistency and spatial alignment are essential for maintaining stable registration between virtual and physical content.

From a system perspective, hybrid approaches offer several advantages for AR tracking. By incorporating geometric constraints, these methods can improve robustness under partial occlusion, viewpoint changes, and rapid camera motion, while learned representations provide flexibility in handling appearance variation. Such combinations are well suited to AR scenarios that involve close coupling between object tracking, camera pose estimation, and scene geometry, including persistent object anchoring and interaction-aware rendering.

Despite these benefits, hybrid geometric–learning trackers also introduce additional complexity. They often rely on accurate geometric inputs, such as depth estimates or camera pose, making performance sensitive to upstream module errors. Moreover, the integration of geometric reasoning may increase computational overhead and complicate system design, particularly for resource-constrained AR devices. Balancing geometric consistency, learning flexibility, and real-time performance therefore remains a key challenge for hybrid tracking paradigms in practical AR deployments.

3.8.Comparative analysis under AR system constraints

From an AR system perspective, no single tracking paradigm is universally optimal. Discriminative trackers offer speed and simplicity, generative trackers provide geometric precision, detection-assisted frameworks support long-term robustness, and self-supervised methods enable adaptability[14]. Effective AR systems increasingly adopt hybrid designs that

dynamically combine multiple paradigms based on context, available resources, and task requirements. This paradigm-level understanding underscores the importance of evaluation that accounts for cross-module interactions and deployment constraints in AR pipelines. Rather than selecting tracking methods solely based on benchmark accuracy, AR designers must consider latency, stability, recoverability, and integration complexity. These considerations ultimately determine the quality and reliability of AR experiences in complex real-world scenes.

4.System-level coupling between object tracking, SLAM, and depth estimation in AR

In augmented reality systems, object tracking, camera localization, and scene geometry estimation form a tightly coupled perception loop. Unlike modular computer vision pipelines where individual components can be optimized independently, AR systems exhibit strong interdependencies among tracking, visual SLAM, and depth estimation. The overall stability and perceptual quality of an AR experience emerge from the joint behavior of these components rather than from the performance of any single module in isolation. Understanding this system-level coupling is therefore essential for designing robust AR tracking solutions in complex real-world scenes.

4.1.Object tracking as a perception backbone in AR pipelines

In practical AR systems, object tracking serves as a central perception primitive that bridges low-level visual sensing and high-level interaction logic. Tracked objects provide spatial anchors for virtual content, enable persistent interaction, and support semantic reasoning[15]. As a result, tracking errors propagate directly to rendering and interaction layers, making them immediately visible to users. Unlike generic tracking applications, AR tracking must satisfy stringent temporal consistency requirements. Even when frame-wise tracking accuracy remains acceptable, small temporal fluctuations can produce noticeable jitter in rendered virtual objects. This sensitivity places additional constraints on the design of tracking algorithms and necessitates close integration with other perception modules that can provide stabilizing signals.

Table 3. Coupling between tracking, SLAM, and depth

Component	Depends On	Failure Propagation
Object Tracking	SLAM, Depth	Drift
Visual SLAM	Tracking landmarks	Pose error
Depth Estimation	Pose	Occlusion error
Rendering	All modules	Visual instability
Interaction	Stable tracking	Trust loss

4.2.Interaction between object tracking and visual SLAM

Visual simultaneous localization and mapping (SLAM) estimates the camera's pose relative to a global or local map of the environment. In AR systems, SLAM provides the coordinate frame in which virtual objects are rendered and updated over time. Object tracking and SLAM therefore operate in a bidirectional relationship.

4.2.1.How tracking supports SLAM

Deep learning based object trackers can improve SLAM robustness in several ways. First, tracked objects, especially rigid and static ones, can serve as reliable landmarks that complement traditional point features. Second, tracking can identify dynamic objects, enabling SLAM systems to exclude moving regions from map optimization, which is particularly important in complex scenes with pedestrians, vehicles, or articulated machinery. Recent AR systems increasingly incorporate object-level information into SLAM pipelines, allowing for object-aware mapping and localization. By integrating tracking outputs, SLAM systems can achieve improved robustness in environments where traditional feature-based methods struggle, such as texture-poor or repetitive scenes[16].

4.2.2.How SLAM stabilizes tracking

Conversely, SLAM plays a critical role in stabilizing object tracking. By providing a consistent world coordinate frame, SLAM reduces the accumulation of drift that arises when tracking is performed purely in the camera frame. This global context enables tracking systems to maintain long-term spatial consistency, which is essential for persistent AR content[17]. However, when SLAM fails due to motion blur, rapid camera movement, or insufficient visual features, tracking outputs may remain locally accurate but become globally inconsistent. In AR, such failures manifest as sudden jumps or gradual drift of virtual objects relative to the physical environment. These effects highlight the importance of designing tracking systems that can gracefully handle SLAM uncertainty or degradation[18].

4.3.Depth estimation and its role in AR tracking

Depth estimation provides essential geometric information for AR systems, supporting occlusion handling, spatial reasoning, and physical interaction. Modern AR platforms often rely on a combination of monocular depth prediction, stereo vision, and active depth sensors. Deep learning-based depth estimation has significantly improved performance in challenging environments where traditional geometry-based methods fail. Depth information can enhance object tracking in multiple ways[19]. By providing scale and spatial constraints, depth cues help disambiguate appearance-based matches and improve robustness to occlusion. For example, depth-aware trackers can distinguish between overlapping objects at different distances, reducing identity switches in crowded scenes. In AR, depth-assisted tracking is particularly valuable for maintaining correct spatial ordering between virtual and physical objects. Accurate depth enables realistic occlusion effects, which are critical for visual plausibility and user immersion[20]. Despite its benefits, depth estimation introduces its own sources of error. Inaccurate or noisy depth predictions can mislead tracking algorithms, especially those that rely on geometric constraints. Incorrect depth values may result in faulty occlusion boundaries, causing appearance changes that confuse trackers and degrade stability. These interactions illustrate that depth estimation and tracking cannot be treated as independent components. Instead, AR systems must

explicitly account for depth uncertainty and design mechanisms to mitigate error propagation across modules[21].

4.4.Coupled failure modes in tracking-SLAM-depth systems

One of the defining characteristics of AR perception systems is the presence of coupled failure modes. Errors in one module often trigger cascading failures in others, leading to system-level breakdowns that are difficult to diagnose and recover from. For instance, prolonged occlusion of a tracked object may cause the tracker to drift or lose the target. This drift can corrupt object-level landmarks used by SLAM, resulting in degraded camera pose estimates. In turn, inaccurate camera poses affect depth estimation and rendering, further destabilizing tracking[22]. Such cascades highlight the need for holistic system design rather than isolated algorithm optimization.

4.5.Joint optimization and learning-based coupling

To address these challenges, recent research explores joint optimization and learning-based coupling of tracking, SLAM, and depth estimation. End-to-end learning frameworks aim to optimize multiple perception tasks simultaneously, leveraging shared representations and mutual constraints[23]. While joint learning offers potential benefits in terms of consistency and robustness, it also introduces new challenges. End-to-end models are often computationally expensive and difficult to debug. Moreover, failures in one task may be harder to isolate and correct when tasks are tightly coupled within a single network. Hybrid approaches that combine learned components with explicit geometric reasoning have therefore gained popularity[24]. By retaining interpretable geometric structures while incorporating data-driven representations, such systems aim to balance robustness, efficiency, and transparency.

4.6.Implications for AR system design

The tight coupling between object tracking, SLAM, and depth estimation has several important implications for AR system design. First, evaluation of tracking algorithms must consider their interaction with other modules rather than relying solely on standalone benchmarks. Second, system-level metrics such as perceptual stability and failure recovery time are often more informative than frame-wise accuracy. Finally, AR systems must incorporate mechanisms for uncertainty estimation and failure detection across modules[25]. By explicitly modeling confidence and reliability, systems can adapt behavior, trigger recovery strategies, or gracefully degrade functionality when perception quality deteriorates.

5.Datasets, benchmarks, and AR-centric evaluation protocols

The progress of deep learning‐based object tracking for augmented reality is tightly coupled with the availability of representative datasets and appropriate evaluation methodologies. While the broader computer vision community has developed a rich ecosystem of tracking benchmarks, many of these resources fail to reflect the unique requirements and constraints of AR systems. As a result, there exists a growing gap between benchmark performance and real-world AR deployment. This section critically reviews existing datasets and benchmarks from an AR-centric perspective and discusses evaluation protocols that better capture system-level behavior.

5.1.Characteristics of AR-oriented tracking data

AR-oriented tracking data differs fundamentally from generic tracking datasets in several aspects[26]. First, AR scenarios emphasize temporal continuity and long-term consistency. Many AR applications require tracking objects continuously over extended periods, often spanning minutes or hours, whereas most tracking benchmarks focus on short video clips lasting only a few seconds. Second, AR data frequently involves strong coupling between camera motion and object motion. In handheld or head-mounted AR systems, rapid and irregular camera movement is common, introducing motion blur, rolling shutter artifacts, and abrupt viewpoint changes. These factors significantly complicate tracking but are underrepresented in conventional datasets. Third, AR environments are inherently interactive. Users may occlude objects with their hands, manipulate tracked items, or move around them freely. Such interactions introduce non-rigid motion, partial visibility, and complex appearance changes that challenge both appearance-based and geometry-based tracking methods. Finally, AR platforms often rely on multimodal sensing. In addition to RGB images, AR devices may provide depth measurements, inertial data, or spatial mapping outputs[27]. Datasets that capture only monocular RGB data fail to reflect the multimodal nature of real AR systems.

5.2.Limitations of existing tracking benchmarks

Most widely used object tracking benchmarks were developed with general computer vision research in mind rather than AR deployment. These benchmarks typically emphasize frame-wise localization accuracy or overlap-based metrics on short sequences. While such metrics are useful for comparing algorithmic performance, they overlook critical AR-specific factors. One major limitation is the lack of temporal stability evaluation[28]. In AR, small frame-to-frame fluctuations can lead to perceptible jitter in rendered virtual objects, even when average localization accuracy remains high. Standard benchmarks rarely quantify such temporal artifacts. Another limitation is the absence of system-level constraints. Benchmarks usually ignore latency, computational cost, and energy consumption, yet these factors are decisive for AR systems running on mobile or wearable devices. A tracker that achieves high accuracy but introduces excessive delay may be unsuitable for AR despite strong benchmark results. Furthermore, many benchmarks assume a clear separation between foreground objects and background. In real AR scenes, cluttered environments and dynamic distractors are common, increasing the risk of identity switches and long-term drift[29]. These challenges are insufficiently represented in existing datasets.

Table 4. Generic vs. AR-centric evaluation

Aspect	Generic Tracking	AR-Centric
Accuracy	Overlap	Spatial alignment
Temporal behavior	Ignored	Jitter & smoothness
Failure handling	Not measured	Recovery time
Latency	Ignored	Critical
Energy	Ignored	Essential
User perception	Ignored	Key metric

5.3.Emerging AR-specific datasets

Recognizing these gaps, recent efforts have begun to develop datasets tailored to AR applications. These datasets often integrate RGB images with depth measurements, inertial data, and precise camera pose ground truth. Such multimodal annotations enable evaluation of tracking performance in conjunction with localization and mapping accuracy[30]. Some AR-oriented datasets focus on object-level interaction, capturing sequences where users manipulate physical objects while virtual content is overlaid. Others emphasize large-scale environments, such as indoor navigation or industrial settings, where long-term consistency and re-localization are critical. Despite these advances, AR-specific datasets remain limited in scale and diversity compared to mainstream tracking benchmarks. Many datasets are collected in controlled environments or involve a narrow range of object categories[31]. Expanding dataset diversity and realism remains an open challenge.

Table 5. Comparison of representative datasets for AR-oriented object tracking

Dataset	Duration	Modalities	Annotation	AR Relevance	Strengths	Limitations
GOT-10k	Short-term	RGB	BBox	Low	Large-scale, diverse objects	No AR interaction, short sequences
LaSOT	Long-term	RGB	BBox	Medium	Long sequences, drift analysis	Limited AR context
VOT	Short-term	RGB	BBox + Attributes	Medium	Standardized evaluation	Not AR-specific
RGB-D Scenes	Medium-term	RGB-D	BBox + Depth	High	Depth-aware occlusion reasoning	Limited scale
AR Interaction Datasets	Long-term	RGB-D + Pose	Object Pose	High	Supports spatial alignment	Narrow object categories
Localization & Mapping Benchmarks	Long-term	RGB-D + IMU	Pose + Map	High	Persistent AR scenarios	Weak object-level labels

A comparison across existing datasets reveals that no single benchmark fully captures the requirements of real-world AR tracking. Short-term RGB tracking datasets are suitable for evaluating frame-wise accuracy but fail to reflect long-term stability and interaction dynamics. RGB-D and AR-oriented datasets better support occlusion reasoning and spatial alignment, yet they are often limited in scale or object diversity. Benchmarks developed for localization and mapping provide valuable context for persistent AR scenarios, but typically lack fine-grained object-level annotations. These trade-offs highlight the importance of selecting datasets based on application requirements rather than benchmark popularity.

5.4.Evaluation metrics beyond accuracy

For AR applications, evaluating object tracking requires metrics that extend beyond traditional accuracy measures. Temporal stability is a primary concern. Metrics that quantify jitter, smoothness, and trajectory consistency are more indicative of user-perceived quality than frame-wise localization error alone. Failure recovery is another critical aspect. In AR, transient tracking failures may be tolerable if recovery is fast and visually smooth. Conversely, prolonged failures or abrupt re-initialization can severely disrupt the user experience. Evaluation protocols should therefore measure recovery time and post-failure stability. Latency and responsiveness are equally important[32]. Even accurate tracking may be unusable if delays exceed perceptual thresholds. AR-centric evaluation should include end-to-end latency measurements that account for sensing, inference, and rendering delays. Finally, energy efficiency is increasingly relevant as AR devices become more compact and wearable. Benchmarks that incorporate power consumption or thermal behavior provide valuable insights for real-world deployment.

5.5.System-level evaluation in AR pipelines

Evaluating tracking algorithms in isolation provides limited insight into their behavior within full AR pipelines. System-level evaluation considers how tracking interacts with SLAM, depth estimation, and rendering. For example, a tracker that produces smooth but slightly biased estimates may be preferable to one that is more accurate but unstable, depending on how errors propagate through the system. User studies also play an important role in AR evaluation. Objective metrics do not always correlate perfectly with subjective perception[33]. Controlled user experiments can reveal subtle issues related to comfort, immersion, and trust that are difficult to capture quantitatively.

5.6.Toward standardized AR evaluation protocols

The lack of standardized AR-centric evaluation protocols remains a major obstacle for progress in the field. Developing shared benchmarks that capture long-term operation, multimodal sensing, and system-level constraints would facilitate more meaningful comparisons between methods. Future benchmarks should consider continuous evaluation over extended durations, realistic interaction scenarios, and deployment on representative hardware platforms[34]. By aligning evaluation more closely with real-world AR requirements, the community can better assess the practical impact of tracking innovations.

6.Deployment challenges and engineering considerations for AR object tracking

While deep learning–based object tracking methods have demonstrated impressive performance in controlled benchmarks, deploying these models in real-world AR systems introduces a distinct set of engineering challenges. AR platforms operate under strict constraints on latency, power consumption, memory, and reliability, particularly on mobile phones and head-mounted displays. This section discusses the key deployment considerations that shape the design and practical adoption of deep tracking solutions in AR.

6.1.Real-time constraints and end-to-end latency

Real-time performance is a non-negotiable requirement for AR object tracking[35]. To maintain perceptual coherence, AR systems typically require update rates of 30–60 frames per second, with end-to-end latency kept below perceptual thresholds. Tracking algorithms must therefore process incoming sensor data, produce stable object estimates, and deliver results to the rendering pipeline with minimal delay. Deep learning models introduce latency at multiple stages, including feature extraction, network inference, and post-processing. While many state-of-the-art trackers achieve high accuracy, their computational complexity can be prohibitive for real-time deployment on embedded hardware. As a result, AR systems often favor models with simpler architectures or reduced resolution inputs, even at the cost of some accuracy[36]. Pipeline-level optimization is equally important. Asynchronous processing, pipelining, and parallel execution across CPU, GPU, and neural accelerators can significantly reduce perceived latency. However, these optimizations increase system complexity and require careful coordination to avoid race conditions and inconsistent state updates.

6.2.Model Compression and Hardware-Aware Design

Model compression techniques play a central role in adapting deep trackers for AR deployment. Pruning removes redundant parameters, quantization reduces numerical precision, and knowledge distillation transfers knowledge from larger models to compact ones. These techniques can substantially reduce memory footprint and inference time while preserving acceptable performance. Beyond generic compression, hardware-aware neural architecture design has gained increasing attention. By tailoring network structures to the characteristics of mobile GPUs, neural processing units, or specialized accelerators, designers can achieve favorable trade-offs between accuracy and efficiency[37]. In AR systems, where thermal constraints may throttle sustained performance, hardware-aware optimization is often critical for maintaining long-term stability.

6.3.Energy consumption and thermal constraints

Energy efficiency is particularly important for wearable AR devices, which rely on limited battery capacity and must manage heat dissipation to ensure user comfort. Continuous object tracking places sustained load on processing units, potentially leading to thermal throttling that degrades performance over time. To address these challenges, AR systems often adopt adaptive strategies that adjust tracking fidelity based on context[38]. For example, high-resolution tracking may be activated only when precise alignment is required, while lower-cost tracking suffices during less critical periods. Such adaptive approaches require mechanisms for monitoring system state and dynamically balancing performance and energy consumption.

6.4.Robustness, reliability, and failure detection

In many AR applications, particularly in industrial or medical contexts, tracking failures can have serious consequences. Ensuring reliability therefore extends beyond average accuracy to include robust handling of edge cases and unexpected conditions. Failure detection mechanisms aim to identify when tracking confidence degrades below acceptable levels. Confidence-aware tracking approaches estimate uncertainty alongside object state, enabling the system to trigger recovery strategies or warn users when alignment quality deteriorates[39]. Despite their importance, uncertainty estimation and failure detection remain underexplored in the tracking literature. Redundancy is another strategy for improving reliability. Combining multiple tracking cues, such as appearance, motion, depth, and semantic information, can reduce the likelihood of catastrophic failure. However, redundancy increases computational load and system complexity, highlighting the need for careful engineering trade-offs.

6.5.Integration with AR software frameworks

Deploying tracking algorithms in AR systems requires integration with broader software frameworks that manage sensing, rendering, interaction, and networking. Popular AR platforms provide abstractions for camera access, spatial mapping, and rendering, but integrating custom deep learning models often requires bridging between different execution environments[40]. Efficient integration involves managing data transfer between sensors and inference engines, synchronizing tracking outputs with rendering updates, and handling variability in sensor quality and availability. Poor integration can negate algorithmic advances by introducing additional latency or instability.

6.6.Scalability and maintainability in deployed systems

As AR applications scale to larger user bases and more diverse environments, maintainability becomes a critical concern. Tracking models may require periodic updates to address new object categories, environments, or failure modes. Deploying updates to large numbers of devices raises challenges related to version compatibility, testing, and rollback. On-device learning and federated approaches offer potential solutions by enabling models to adapt locally while sharing improvements across devices without centralizing raw data. However, these approaches introduce additional complexity and require robust safeguards to prevent model degradation[41].

6.7.Ethical, privacy, and safety considerations

Continuous visual sensing inherent to AR raises ethical and privacy concerns. Object tracking systems may capture sensitive information about users and their surroundings. Designing privacy-aware tracking solutions involves minimizing data retention, performing inference on-device, and providing transparency and control to users. Safety considerations also extend to how tracking errors are communicated. In safety-critical applications, AR systems must avoid overconfidence and provide clear indications of uncertainty[42]. Addressing these concerns is essential for building trustworthy AR technologies.

7.Failure modes in deep learning − based object tracking for complex AR scenes

Despite substantial progress in deep learning-based object tracking, failures remain inevitable in complex real-world AR scenarios. Unlike offline vision tasks, tracking failures in AR have immediate and visible consequences, directly affecting spatial alignment, interaction fidelity, and user trust. Importantly, many failures observed in deployed AR systems do not stem from catastrophic algorithmic breakdowns but from subtle degradations that accumulate over time or propagate across system components. This section analyzes common failure modes of deep tracking in AR from a system-level perspective, emphasizing their causes, manifestations, and implications.

Table 6. Failure modes and system-level consequences

Failure Mode	Root Cause	Tracking Effect	AR Impact
Illumination shock	Lighting change	Tracking loss	Virtual jump
Long-term occlusion	Invisibility	ID error	Wrong binding
Drift accumulation	Model contamination	Misalignment	Persistent instability
Dynamic distractors	Similar objects	ID switch	Interaction failure
Fast motion	Camera shake	Feature loss	SLAM cascade
Depth inconsistency	Noisy geometry	Occlusion error	Visual artifacts

7.1.Illumination shock and appearance collapse

Sudden illumination changes represent one of the most frequent failure triggers in AR tracking[43]. Transitions between indoor and outdoor environments, changes in artificial lighting, or strong shadows can drastically alter object appearance within a few frames. Although deep trackers are generally more robust than handcrafted-feature-based methods, they remain sensitive to appearance distributions encountered during training. In AR, illumination-induced failures often manifest as abrupt tracking loss or rapid drift. Because virtual content is anchored to tracked objects, such failures produce noticeable jumps or sliding artifacts that break immersion. Unlike short-term tracking benchmarks, where temporary failure may be acceptable, AR systems require rapid recovery without perceptible disruption. This requirement highlights the need for illumination-aware representations and adaptive mechanisms that detect and compensate for appearance collapse.

7.2.Long-term occlusion and re-identification errors

Occlusion is an inherent aspect of interactive AR environments. Users frequently occlude objects with their hands, other objects, or their own bodies. While short-term occlusions can often be handled by temporal smoothing or motion extrapolation, long-term occlusions pose significant challenges[44]. When a tracked object disappears for extended periods, trackers must rely on re-identification mechanisms upon reappearance. Failures in re-identification can lead to identity switches or incorrect association with similar objects in the scene. In AR, such errors are particularly disruptive, as virtual content may reattach to the wrong physical object, undermining user trust and potentially causing safety issues in industrial contexts. Detection-assisted and memory-based trackers partially address this problem, but they introduce trade-offs between robustness and latency. Moreover, re-identification remains challenging in cluttered scenes with multiple visually similar objects, underscoring the need for stronger semantic and geometric priors.

7.3.Drift accumulation and temporal instability

Drift accumulation is a subtle yet pervasive failure mode in AR tracking. Even when frame-wise accuracy remains high, small systematic errors can accumulate over time, leading to gradual misalignment. This phenomenon is often exacerbated by appearance model updates that inadvertently incorporate background features or occluders[45]. In AR applications, drift manifests as virtual objects slowly sliding away from their intended positions or exhibiting low-frequency jitter. Such artifacts may not be captured by conventional accuracy metrics but are immediately noticeable to users. Temporal instability is therefore a critical consideration that motivates the use of explicit temporal consistency constraints and stabilization mechanisms.

7.4.Dynamic distractors and identity confusion

Complex AR scenes often contain multiple objects with similar appearance, size, or motion patterns. Discriminative trackers that rely primarily on appearance similarity may confuse targets with distractors, leading to identity switches. These errors are especially problematic in crowded environments or collaborative AR scenarios. Identity confusion not only disrupts interaction but can also propagate to other system components[46]. For example, incorrect tracking may corrupt object-level landmarks used by SLAM, further destabilizing camera localization and rendering. Addressing this failure mode requires integrating additional cues, such as depth, motion history, and semantic context, to disambiguate targets.

7.5.Fast motion, motion blur, and camera dynamics

Handheld and head-mounted AR systems frequently experience rapid and irregular camera motion. Fast motion

introduces motion blur, rolling shutter artifacts, and large inter-frame displacement, reducing feature reliability and challenging both discriminative and generative trackers. Tracking failures under fast motion often coincide with SLAM degradation, creating coupled failure cascades. In such scenarios, recovery is particularly difficult because both object-centric and camera-centric references are compromised[47]. Designing trackers that explicitly model motion uncertainty and leverage inertial cues remains an important research direction.

7.6.Depth and geometry-induced failures

Depth estimation errors represent another source of tracking instability in AR systems. Inaccurate depth predictions can distort occlusion boundaries, causing sudden appearance changes that confuse trackers. Errors in geometry estimation may also lead to incorrect scale or pose estimates, further degrading alignment. Because depth and tracking are tightly coupled in AR pipelines, failures in one module can amplify errors in the other[48]. Robust AR systems must therefore account for depth uncertainty and avoid over-reliance on potentially noisy geometric cues.

7.7.System-level failure propagation

A defining characteristic of AR tracking failures is their tendency to propagate across system components. A local tracking error can corrupt SLAM landmarks, degrade camera pose estimates, and destabilize rendering. Conversely, SLAM failures can invalidate the spatial context required for stable tracking. This interdependence highlights the limitations of evaluating tracking algorithms in isolation. System-level resilience requires coordinated failure detection, uncertainty estimation, and recovery strategies across modules[49]. Designing such mechanisms remains an open challenge.

7.8.Mitigation strategies and design implications

Mitigating tracking failures in AR requires a combination of algorithmic and system-level approaches. Confidence-aware tracking enables systems to detect degradation and trigger recovery mechanisms before failures become perceptible. Redundant sensing and multi-cue integration improve robustness but increase complexity and resource consumption. From a design perspective, AR systems must balance aggressiveness and conservatism[50]. Overly aggressive adaptation may destabilize tracking, while excessive smoothing can introduce lag and reduce responsiveness. Achieving this balance is central to reliable AR deployment.

8.Future research directions toward trustworthy and scalable AR object tracking

While deep learning has significantly advanced object tracking for augmented reality, current solutions remain far from meeting the robustness, scalability, and trustworthiness required for large-scale real-world deployment. Future research must therefore move beyond incremental accuracy improvements and address fundamental system-level challenges. This section outlines key research directions that are likely to shape the next generation of AR object tracking systems, with an emphasis on practicality, reliability, and long-term operation.

8.1.Foundation models adapted for AR tracking

Large-scale foundation models have demonstrated remarkable generalization capabilities across a wide range of vision tasks. Their success raises natural questions about their applicability to AR object tracking. In principle, foundation models trained on diverse visual data could offer improved robustness to appearance variation, occlusion, and domain shift. However, directly deploying such models in AR systems remains challenging. Foundation models are typically computationally intensive and optimized for offline or server-side inference, conflicting with the real-time and energy constraints of mobile and wearable AR platforms. Moreover, most foundation models lack explicit mechanisms for enforcing temporal stability, which is critical for AR alignment. Future research should explore task-adapted foundation models for AR tracking. This includes model distillation techniques that transfer knowledge from large models to lightweight, AR-specific architectures, as well as hybrid designs that combine foundation-level representations with task-specific temporal and geometric constraints. Achieving this balance is essential for leveraging the strengths of foundation models without violating system constraints.

8.2.Lifelong and continual learning under stability constraints

AR systems are expected to operate continuously across diverse environments, objects, and usage scenarios. Lifelong learning offers the potential for trackers to adapt over time, improving robustness and personalization. However, unconstrained continual adaptation poses significant risks in AR contexts, where instability can lead to unpredictable behavior and degraded user experience. A central challenge lies in balancing plasticity and stability. Trackers must adapt to new conditions without forgetting previously learned knowledge or destabilizing established tracking behavior. Research into regularization-based continual learning, modular architectures, and memory replay strategies offers promising directions, but their integration into real-time AR systems remains limited. Future work should focus on developing continual learning mechanisms explicitly designed for AR, incorporating confidence estimation, bounded updates, and rollback strategies. Such mechanisms would allow systems to benefit from adaptation while maintaining predictable and trustworthy behavior.

8.3.Physics- and interaction-aware tracking

Most current deep trackers rely primarily on appearance cues, with limited incorporation of physical constraints. In AR, however, objects often interact with users and the environment in physically meaningful ways. Incorporating physics-based reasoning into tracking can improve robustness

under occlusion, contact, and deformation. Physics-aware tracking models can leverage constraints such as rigidity, kinematics, and contact dynamics to reduce ambiguity and prevent implausible motion estimates. Learning such constraints from data, or integrating them explicitly into tracking pipelines, represents an important research direction. Interaction-aware tracking further enables more natural AR experiences by anticipating and responding to user actions.

8.4.Multimodal and cross-sensor fusion at scale

Future AR systems will increasingly rely on heterogeneous sensors, including RGB cameras, depth sensors, inertial measurement units, and potentially audio or haptic inputs. Effectively fusing these modalities can significantly enhance tracking robustness, particularly in challenging conditions where individual sensors degrade. While multimodal fusion has been explored in controlled settings, scaling such approaches to real-world AR deployment remains challenging. Sensor synchronization, calibration drift, and variable data quality complicate fusion. Research into adaptive, confidence-aware fusion strategies that dynamically weight sensor contributions based on reliability is a promising direction.

8.5.Uncertainty-aware and failure-resilient tracking

Trustworthy AR systems must not only perform well on average but also handle failures gracefully. Uncertainty-aware tracking explicitly models confidence alongside object state estimates, enabling systems to reason about reliability and trigger appropriate responses when uncertainty increases. Future research should integrate uncertainty estimation more deeply into tracking pipelines and system-level decision making. This includes designing trackers that output calibrated uncertainty measures and developing policies that adapt rendering, interaction, or user feedback based on confidence levels. Such approaches are essential for safety-critical AR applications.

8.6.Human-in-the-loop optimization and feedback

Humans are an integral part of AR systems, yet most tracking research treats users as passive observers. Incorporating lightweight human-in-the-loop mechanisms offers a practical pathway to improving tracking reliability in deployed systems. User feedback, whether explicit or implicit, can provide valuable signals for correcting errors, re-initializing trackers, or refining models. Designing effective human-in-the-loop strategies requires careful consideration of usability and cognitive load. Feedback mechanisms must be unobtrusive and intuitive, enhancing rather than disrupting the AR experience. Exploring this design space represents an important interdisciplinary research direction.

8.7.Evaluation, benchmarking, and reproducibility

Finally, advancing AR object tracking requires improved evaluation practices. Standardized AR-centric benchmarks, system-level metrics, and reproducible evaluation pipelines are essential for meaningful comparison and progress. Future benchmarks should emphasize long-term operation, interaction, and deployment on representative hardware platforms. Open-source systems and shared evaluation tools can further accelerate progress by lowering barriers to entry and enabling rigorous validation. Strengthening the connection between academic research and deployed AR systems will be critical for translating innovations into practice.

9.Conclusion

This survey has provided a comprehensive review of deep learning based object tracking for augmented reality (AR) in complex real-world scenes, emphasizing system-level considerations often overlooked in conventional tracking research. By framing tracking as a core perception capability tightly integrated with visual SLAM, depth estimation, and rendering, we showed that algorithmic accuracy alone is insufficient for reliable AR deployment, since even small instabilities can propagate through the system and manifest as perceptible artifacts that affect user trust. Our analysis of tracking paradigms, datasets, evaluation protocols, and failure modes from an AR-centric perspective highlighted the importance of temporal stability, failure recovery, and resource efficiency, which are frequently more critical than peak accuracy for practical applications. Looking ahead, progress toward scalable and trustworthy AR tracking will require closer coupling between learning-based methods and system design, with promising directions including uncertainty-aware tracking, task-adapted foundation models, lifelong learning under stability constraints, and interaction- or physics-aware representations. Ultimately, advancing AR tracking will depend not only on algorithmic innovation but also on deployment-aware research practices and rigorous system-level validation that ensure reliable performance beyond controlled benchmarks.

References

[1] WU Y, LIM J, YANG M H. Online object tracking: A benchmark//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Portland, USA: IEEE, 2013: 2411 – 2418.

[2] DANELLJAN M, BHAT G, KHAN F S, et al. ECO: Efficient convolution operators for tracking//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, USA: IEEE, 2017: 6931 – 6939.

[3] LI B, YAN J, WU W, et al. High performance visual tracking with Siamese region proposal network//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, USA: IEEE, 2018: 8971 – 8980.

[4] WANG Q, ZHANG L, BERTINETTO L, et al. Fast online object tracking and segmentation: A unifying approach//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, USA: IEEE, 2019: 1328 – 1338.

[5] HU P, WANG Q, ZHANG L, et al. Learning Siamese representation for real-time visual tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(3): 3072 – 3089.

[6] HUANG L, ZHAO X, HUANG K. GOT-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(5): 1562 – 1577.

[7] FAN H, LIN L, YANG F, et al. LaSOT: A high-quality benchmark for large-scale single object tracking. International Journal of Computer Vision, 2021, 129(2): 439 – 461.
[8] KRISTAN M, LEONARDIS A, MATAS J, et al. The visual object tracking VOT challenge: A retrospective. International Journal of Computer Vision, 2016, 124(4): 527 – 559.
[9] JIAO L, WANG D, BAI Y, et al. Deep learning in visual tracking: A review. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34(9): 5497 – 5516.
[10] MARVASTI-ZADEH S M, LI J, ZOU J, et al. Deep learning for visual tracking: A comprehensive survey. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(5): 3943 – 3968.
[11] ZHANG L, FAN H, XIANG T, et al. Visual object tracking: Progress, challenges, and future directions. The Innovation, 2023, 4(4): 100395.
[12] YE J, CAO Z, LI B. Transformer-based visual object tracking: A survey. Pattern Recognition, 2024, 145: 109823.
[13] CHEN Z, PENG C, LIU S, et al. Spatial-temporal transformer networks for visual object tracking//Proceedings of the European Conference on Computer Vision (ECCV). Cham: Springer, 2022: 238 – 255.
[14] REZATOFIGHI H, MILAN A, SHI J, et al. A survey on multiple object tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(1): 1 – 20.
[15] AMOSA T I, SEBASTIAN P, IZHAR L I, et al. Multi-camera multi-object tracking: A review of current trends and future advances. Neurocomputing, 2023, 533: 158 – 184.
[16] FU C, LU K, ZHENG G, et al. Siamese object tracking for unmanned aerial vehicles: A review and comprehensive analysis. Artificial Intelligence Review, 2023, 56(6): 5805 – 5851.
[17] GAO S, XIAO Z, JIANG Z. RGB-D object tracking: A survey. Sensors, 2023, 23(4): 1828.
[18] SYED T A, SIDDIQUI M S, ABDULLAH H B, et al. In-depth review of augmented reality: Tracking technologies, development tools, AR displays, collaborative AR, and security concerns. Sensors, 2023, 23(1): 146.
[19] VAN KREVELEN D W F, POELMAN R. A survey of augmented reality technologies, applications and limitations. International Journal of Virtual Reality, 2010, 9(2): 1 – 20.
[20] CARMIGNIANI J, FURHT B, ANISETTI M, et al. Augmented reality technologies, systems and applications. Multimedia Tools and Applications, 2011, 51(1): 341 – 377.
[21] AZUMA R T, BAILLOT Y, BEHRINGER R, et al. Recent advances in augmented reality. IEEE Computer Graphics and Applications, 2001, 21(6): 34 – 47.
[22] MANURI F, SANNA A. A survey on applications of augmented reality. ACSIJ Advances in Computer Science: an International Journal, 2016, 5(1): 18 – 27.
[23] MALTA A, FARINHA T, MENDES M. Augmented reality in maintenance—history and perspectives. Journal of Imaging, 2023, 9(7): 142.
[24] GHASEMI Y, JEONG H, CHOI S H, et al. Deep learning-based object detection in augmented reality: A systematic review. Computers in Industry, 2022, 139: 103661.
[25] SARLIN P E, DUSMANU M, SCHÖNBERGER J L, et al. LaMAR: Benchmarking localization and mapping for augmented reality//Computer Vision – ECCV 2022. Cham: Springer, 2022: 686 – 704.
[26] XIANG Y, SCHMIDT T, NARAYANAN V, et al. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. International Journal of Computer Vision, 2018, 126(7): 749 – 766.
[27] PARK K B, CHOI S H, KIM M, et al. Deep learning-based mobile augmented reality for task assistance using 3D spatial mapping and snapshot-based RGB-D data. Computers & Industrial Engineering, 2020, 146: 106585.
[28] KONSTANTINIDIS F K, KANSIZOGLOU I, SANTAVAS N, et al. MARMA: A mobile augmented reality maintenance assistant for fast-track repair procedures in the context of Industry 4.0. Machines, 2020, 8(4): 88.
[29] MOURTZIS D, SIATRAS V, ANGELOPOULOS J. Real-time remote maintenance support based on Augmented Reality (AR). Applied Sciences, 2020, 10(5): 1855.
[30] WANG S, ZARGAR S A, YUAN F G. Augmented reality for enhanced visual inspection through knowledge-based deep learning. Structural Health Monitoring, 2021, 20(2): 426 – 442.
[31] ALVES J B, MARQUES B, FERREIRA C, et al. Comparing augmented reality visualization methods for assembly procedures. Virtual Reality, 2021, 26(2): 235 – 248.
[32] TANG A, OWEN C, BIOCCA F, et al. Comparative effectiveness of augmented reality in object assembly//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI). New York, USA: ACM, 2003: 73 – 80.
[33] HENDERSON S J, FEINER S K. Augmented reality in the psychomotor phase of a procedural task//Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR). Basel, Switzerland: IEEE, 2011: 191 – 200.
[34] LAI Z H, TAO W, LEU M C, et al. Smart augmented reality instructional system for mechanical assembly towards worker-centered intelligent manufacturing. Journal of Manufacturing Systems, 2020, 55: 69 – 81.
[35] ZHENG L, LIU X, AN Z, et al. A smart assistance system for cable assembly by combining wearable augmented reality with portable visual inspection. Virtual Reality & Intelligent Hardware, 2020, 2(1): 12 – 27.
[36] SUN Y, KANTAREDDY S N R, SIEGEL J, et al. Towards industrial IoT – AR systems using deep learning-based object pose estimation//Proceedings of the 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC). London, UK: IEEE, 2019: 1 – 8.
[37] BASTES J B, RIBEIRO S, PINTO A, et al. Augmented reality for training and maintenance of reclosers: A case study of a wearable application//2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC). Madrid, Spain: IEEE, 2021: 426 – 442.
[38] DINI G, DALLE MURA M. Application of augmented reality techniques in through-life engineering services. Procedia CIRP, 2015, 38: 14 – 23.
[39] BOBOC R G, GÎRBACIA F, BUTILĂ E V. The application of augmented reality in the automotive industry: A systematic literature review. Applied Sciences, 2020, 10(12): 4259.
[40] ZHOU F, DUH H B L, BILLINGHURST M. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR//Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR). Cambridge, UK: IEEE, 2008: 193 – 202.
[41] SARHAN A, LERAY M, CREUSIER T, et al. Augmented reality knowledge management for industrial transformation and innovation. Procedia CIRP, 2024, 128: 19 – 24.
[42] MAO W, SCHEFFER S, MAJUMDAR A. Augmented reality-enabled knowledge management in industrial maintenance: The DILEAF framework. Computers & Industrial Engineering, 2025, 187: 111363.
[43] BILLINGHURST M, CLARK A, LEE G. A survey of augmented reality. Foundations and Trends in Human – Computer Interaction, 2015, 8(2 – 3): 73 – 272.
[44] SARLIN P E, LARSSON V, DUSMANU M, et al. Benchmarking localization and mapping for AR with LaMAR. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024, 46(2): 912 – 928.
[45] WANG T, QIN H, BAI X. Evaluation metrics for deep learning based object tracking. Pattern Recognition Letters, 2024, 177: 64 – 72.
[46] CHEN Y, SONG R, HU T. Tracking datasets for AR and VR: A review. Multimedia Tools and Applications, 2023, 82(23): 34501 – 34530.
[47] SUALEH M, KIM G. A review on robustness and challenges of simultaneous localization and mapping for AR/VR applications. Robotics and Autonomous Systems, 2023, 158: 104255.
[48] ZHANG J, YANG L, CHEN Y, et al. Simultaneous localization and mapping toward augmented reality: A survey. Sensors, 2023, 23(9): 4171.
[49] RAUF A, ELMASRY M, KIM S. Tracking technologies in augmented reality: A review. Multimedia Tools and Applications, 2024, 83(9): 25411 – 25442.
[50] RICCI S, et al. Viewpoint: Virtual and augmented reality in basic and clinical science. Journal of NeuroEngineering and Rehabilitation, 2022, 19(1): 54.