ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

Tracker-Conditioned Diffusion Proposals for Multi-Object Tracking

Muhammad Mustapha Miko[1*]; Li Dequan[2]

[2]Professor
[1;2]Anhui University of Science and Technology

Corresponding Author: Muhammad Mustapha Miko[1*]

Publication Date: 2026/09/21

Abstract: Diffusion-based detectors begin inference from noisy boxes, whereas tracking-by-detection pipelines usually localize each video frame independently. This study examines whether propagated tracker boxes can guide a frozen DiffusionDet detector without retraining. Confirmed tracks are propagated using their latest observation or a Kalman prediction, mapped to the detector's latent box space, corrupted at a selected diffusion timestep, and mixed with random proposals for new-object discovery. On a KITTI development split, four low-noise proposals around each Kalman prediction improve mean car/pedestrian multiple object tracking accuracy (MOTA) by 1.29 points over a matched short-timestep random control and 0.66 points over full random initialization. Higher order tracking accuracy (HOTA) is effectively tied, while the identity F1 score (IDF1) is lower than with full random initialization. Transferred without retuning to MOT17 half-validation, the configuration averages 53.68 HOTA, 55.88 MOTA, and 63.91 IDF1 across three inference seeds. Mean HOTA improves by 0.99 points over full random initialization, primarily through higher recall at the cost of lower precision. The findings support a conditional detection-recall benefit rather than a consistent improvement in identity association.

Keywords: *Multi-Object Tracking; DiffusionDet; Temporal Proposals; Kalman Prediction; ByteTrack; Tracking-by-Detection.*

How to Cite: Muhammad Mustapha Miko; Li Dequan (2026) Tracker-Conditioned Diffusion Proposals for Multi-Object Tracking. *International Journal of Innovative Science and Research Technology*, 11(9), 616-625. https://doi.org/10.38124/ijisrt/26sep483

I. INTRODUCTION

Tracking-by-detection combines a per-frame object detector with an association module. This modular design is effective, but the detector discards state that the tracker has already estimated. A missed or spatially unstable detection cannot always be repaired after association, especially under occlusion, blur, crowding, and small object scale.

The tracker can only associate the detections available to it. Here, the estimated trajectory is used before localization, when candidate boxes are initialized. This allows continuing objects to influence detection rather than restricting temporal information to the association stage.

DiffusionDet formulates detection as denoising box variables from a random distribution [1]. Its initial proposal distribution is therefore a natural interface through which video state can enter an otherwise image-trained detector. We ask a focused question: can an online tracker's state guide the next frame's proposal distribution without retraining the detector?

This paper makes three contributions. (1) We implement tracker-conditioned DiffusionDet sampling that mixes noisy propagated tracks with random discovery proposals. (2) We isolate temporal initialization using a frozen detector, matched frame seeds, fixed association parameters, and controls for diffusion start timestep. (3) We provide a reproducible evaluation stack with official HOTA, CLEAR, and identity metrics, scenario breakdowns, throughput, and qualitative diagnostics on KITTI and MOT17. We deliberately avoid a state-of-the-art claim: the observed gain is mainly in detection recall, while association quality does not consistently improve.

The central problem is the separation between what the tracking system knows about the recent past and what the detector uses when processing the current frame. The tracker maintains estimates of object locations over time, yet a frame-independent detector starts its localization process without using those estimates. This separation is convenient for modular design, but it leaves temporal information unused at the stage where candidate detections are formed. The present work examines that interface. It asks whether the existing tracker state can guide part of the detector's search while preserving the detector and association components already used by the pipeline.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

The distinction between a candidate box and an accepted detection is important to this question. A propagated tracker box is an estimate derived from earlier observations. It can indicate where an object may continue, but it is not itself evidence that the object is correctly localized in the current frame. In the proposed arrangement, that estimate only affects proposal initialization. The diffusion detector then processes the proposal set and produces the detections consumed by the tracker. The experiment therefore concerns temporal guidance of detection rather than direct substitution of predictions for observations.

This formulation also limits the scope of the contribution. The study does not introduce a new association objective or train a detector specifically to model video trajectories. It changes the distribution from which some initial boxes are sampled at inference time. That narrower intervention makes it possible to examine whether the state already available in an online tracking pipeline has useful information for the detector. It also makes negative or mixed outcomes informative: a recall improvement accompanied by weaker identity quality reveals that temporal localization guidance and successful identity association are distinct requirements.

➢ *Related Work*

Diffusion-based detection. DiffusionDet models object detection as the reverse of a box corruption process and supports varying its proposal count and sampling steps at inference [1]. We retain its training objective and learned denoising head, changing only the inference distribution for a subset of proposals.

DiffusionTrack instead trains a joint detector-associator to denoise paired boxes [2]. SparseVOD propagates learned proposals and aggregates proposal features across video frames, but is based on Sparse R-CNN and is evaluated as video object detection [3]. Our narrower setting reuses the box state of an external online tracker to alter a frozen image-trained DiffusionDet sampler. This distinction motivates an empirical contribution and does not support a broad priority claim.

Online multi-object tracking. Simple Online and Realtime Tracking (SORT) combines a Kalman filter with Hungarian assignment [4]. ByteTrack associates both high- and low-confidence detections to recover trajectories that would otherwise fragment [5]. OC-SORT emphasizes observations to reduce motion-estimation error during occlusion [6]. Our association comparison feeds identical serialized DiffusionDet detections to all three backends, separating detector behavior from association behavior. FairMOT provides related context for joint detection and re-identification [7]. MeMOT jointly performs proposal generation and association through a learned spatio-temporal memory [8]. In contrast, our feedback signal is only tracker box state and requires no video-specific detector training.

Evaluation. Multiple object tracking accuracy (MOTA) is sensitive to misses and false positives, the identity F1 score (IDF1) measures identity preservation, and higher order tracking accuracy (HOTA) balances detection and association accuracy [9]. We report their decompositions and counts through TrackEval, rather than selecting a single favorable metric.

The comparison with related work is organized around where temporal information enters the system. A detector can process a frame and leave all temporal reasoning to a later association stage. Alternatively, a learned model can incorporate temporal information within its own features or prediction process. The setting examined here uses an external tracker to alter the initial boxes of a frozen detector. These choices should not be treated as interchangeable merely because all of them use information from more than one frame. They differ in the component being modified, the training requirements, and the role assigned to the previous state.

For this reason, the related systems provide context rather than evidence that the present intervention solves the same problem in the same way. A jointly trained detector-associator and a detector whose proposals are initialized from tracker state can have different sources of improvement. Likewise, reusing learned proposals across video frames involves information beyond simply propagating geometric boxes. The present experiment deliberately stays with tracker box state. Its interpretation should therefore remain tied to the observed effect of this state on a frozen proposal sampler, rather than extending to all forms of temporal detection or joint tracking.

II. MATERIALS AND METHODS

➢ *Tracker-Conditioned Proposal Initialization Let the Detector use N Proposals and Let B Denote t−1*

Confirmed tracker boxes after frame $t - 1$. A motion operator g produces $\hat{B} = g(B)$. We consider the latest observed t t−1 position and the tracker's one-step Kalman prediction.

The propagated boxes are prior locations, not final detections. The latest-observation variant reuses a measured position, while the Kalman variant uses a motion prediction. Both retain the learned denoising process and differ only in where part of the proposal search begins.

Each propagated box is clipped to the resized image, converted from corner coordinates to normalized center-size coordinates, and mapped to DiffusionDet's scaled box domain. At diffusion start timestep s, a proposal is sampled as:

$$x_s = \sqrt{\bar{\alpha}_s} x_0 + \lambda \sqrt{1 - \bar{\alpha}_s} \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \qquad (1)$$

Where x is a propagated box, λ is the prior-noise scale, 0 and $\bar{\alpha}$ is the cumulative diffusion schedule. The resulting s samples replace the first $K \leq N$ random proposals. The remaining $N - K$ proposals remain random to support entering objects and recovery from erroneous track state. A copies parameter draws multiple independent noisy samples around each propagated track, capped by N.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

The unchanged detector head denoises the mixed set. Class-aware ByteTrack consumes the predictions and supplies the next frame's state. This feedback loop changes neither detector weights nor association hyperparameters. At a sequence boundary, tracker state is cleared.

Random proposals retain coverage beyond existing tracks, including entering objects. Resetting state prevents trajectories from one video becoming priors in another. These operations preserve scene coverage and the sequence-specific nature of the feedback loop.

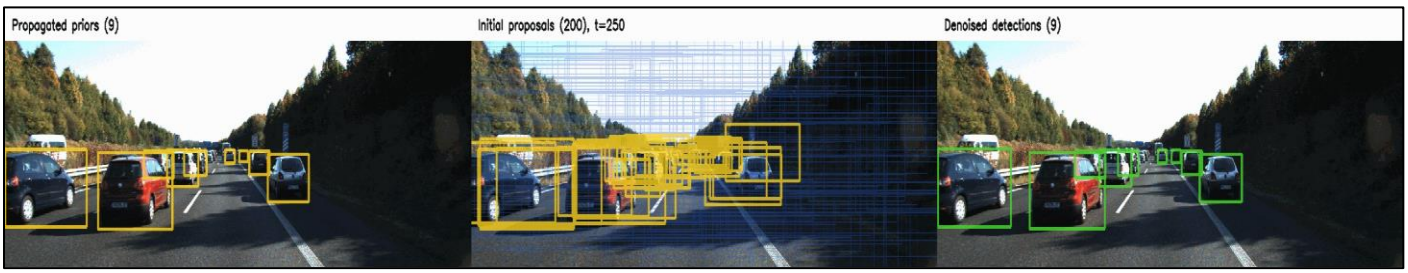

Fig 1 Proposal Diagnostic. Propagated Track Boxes Seed Part of the Noisy Proposal Set; the Remaining Proposals Retain Random Scene Coverage before Denoising.

The proposal budget provides a useful way to understand the intervention. The detector receives a fixed number of candidate boxes, and temporal initialization changes only the origin of a subset of them. The experiment does not increase the total proposal count when tracker state is used. Consequently, a temporal proposal occupies a position that would otherwise be assigned to a random proposal. The comparison is between different uses of the same budget, rather than between a larger temporal search and a smaller random search. This is why the number of proposals is held fixed in the temporal comparisons.

➢ *Propagation and Denoising Perform Different Functions*

Propagation transfers the tracker's estimate from the preceding frame into a location hypothesis for the current frame. Denoising then operates within the detector using the current inference process. The latest-observation and Kalman alternatives change the first of these operations. Neither alternative changes the learned denoising head. Keeping these roles separate helps avoid interpreting a propagated location as a detection that has already been confirmed. A prior may be useful because it directs attention toward a continuing object, but its usefulness is still assessed through the final detector and tracker outputs.

The coordinate conversion places the propagated location in the representation expected by the detector. The tracker's box coordinates cannot be inserted unchanged if the proposal sampler operates in a normalized center-size space. Clipping, normalization, and mapping to the scaled box domain make the temporal proposal compatible with that sampler. These transformations concern representation rather than object identity: they describe the same location hypothesis in the coordinates used by the detection process. Their role in the method is therefore to connect the tracker interface to the existing detector interface without modifying the detector's learned parameters.

Equation (1) combines a location-dependent term with a random term. The propagated box supplies the location-dependent component, while the sampled noise prevents the temporal initialization from being a single deterministic copy of that box. The start timestep determines the schedule quantities used in the expression, and the prior-noise scale controls the additional noise multiplier. Both are part of the proposal construction. The experimental comparison must therefore distinguish the effect of a temporal location from the effect of changing the diffusion start itself. This distinction motivates the matched short-timestep random control used in the results.

The copies setting changes how many independent samples are drawn around a propagated track. Multiple samples do not represent additional tracked objects; they are several initial hypotheses derived from the same propagated location. They still count toward the fixed detector budget. Thus, the setting controls how much of the proposal set is allocated around continuing tracks, subject to the cap on the total number of proposals. The reported four-copy configuration should be interpreted in this limited sense. The manuscript does not infer that allocating still more copies would necessarily improve detection or association.

The random remainder is essential to the interpretation of a mixed proposal set. Existing tracker state describes trajectories already maintained by the system, whereas an entering object may have no such trajectory. If all search were concentrated around existing state, the initialization would have no independent mechanism for covering locations outside that state. Retaining random proposals preserves the original source of scene coverage alongside the temporal hypotheses. The method therefore combines two purposes within one proposal budget: continuity for tracked objects and coverage that is not conditioned on those tracks.

Temporal guidance also introduces a possible feedback path for errors. A poor tracker estimate can influence the next frame's proposal locations, and those locations can influence the detections from which the following tracker state is formed. The presence of this feedback path does not mean that every error persists, but it explains why a temporal prior cannot be assumed to help in every frame. The mixed random component and the unchanged detector remain important parts of the evaluation. The result must be judged from the complete sequence, where useful guidance and misleading guidance can both occur.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

Sequence boundaries define the lifetime of this feedback. Clearing the tracker state ensures that a location hypothesis derived from one video is not reused when a different video begins. Within a sequence, the state belongs to the ongoing online process; at a boundary, that process starts again. This operational distinction matters for evaluation because the same detector settings can be applied to many videos while their temporal histories remain separate. The manuscript's sequence-specific execution and seed schedule support this separation without changing the detector checkpoint between the compared initialization strategies.

Another useful distinction is between the source of a proposal and the source of the final tracking decision. A temporal proposal is identifiable by how its initial coordinates were formed, but it still participates in the detector's common denoising process. Association occurs after the detector has produced its predictions. The experiment therefore does not assign a final trajectory identity simply because a proposal began near a particular track. This separation is consistent with the reported identity limitations: spatial guidance can influence detection coverage without making the subsequent association correct by construction.

The proposal cap also means that the copies setting and the random remainder must be interpreted together. Samples allocated to propagated tracks use part of the same finite set available to unconditioned proposals. Increasing emphasis around known locations consequently changes the composition of the search, not just its temporal component in isolation. The reported configuration describes one such composition. The paper's evidence concerns that configuration under the documented controls, and does not establish a general rule that a more concentrated proposal distribution is always better.

- *Experimental Protocol*

KITTI. KITTI Tracking provides 21 labeled training videos and 29 server-evaluated test videos [10]. We fit DiffusionDet on training sequences 0000–0016 and use 0017–0020 as a development split for temporal ablation and configuration selection. These local results are not an untouched test estimate or KITTI test-server result. The frozen 18,000-iteration R50 checkpoint uses 200 proposals. Car and pedestrian are the official headline classes; cyclist, occlusion, and size analyses are supplemental.

MOT17. We follow the common first-half training, second-half validation protocol across all seven MOT17 training scenes [11]. A one-class DiffusionDet-R50 initialized from public COCO weights is trained for 30,000 iterations. The validation half contains 2,659 frames. Reported values are not hidden-test-server results.

Metrics and controls. TrackEval computes HOTA, DetA, AssA, MOTA, MOTP, detection recall and precision, IDF1, FP, FN, identity switches, and fragmentation. COCO evaluation is applied to serialized per-frame detector output. All temporal comparisons hold the checkpoint, 0.1 serialization threshold, 200 proposals, ByteTrack settings, image order, and per-frame seed schedule fixed. Three inference seeds are used for the principal MOT17 comparison. Ninety-five percent intervals use a Student-t interval with two degrees of freedom and are descriptive at this small sample size.

Changing the start timestep also changes sampling. The matched random control therefore separates the effect of temporal initialization from the effect of a shorter diffusion start. Matching frame seeds reduces variation unrelated to the proposal strategy.

The two datasets serve different roles in the study. The KITTI development split is used to examine temporal settings and select a configuration. MOT17 then evaluates the transferred configuration without another round of temporal retuning. This ordering gives a more specific meaning to the cross-dataset result: it tests the chosen setting in a second local evaluation protocol. It does not convert either evaluation into a hidden-test result. The distinction between development, validation, and benchmark test data remains necessary even when all of the reported measures are computed with official evaluation tools.

Within KITTI, the small development partition places an important limit on generalization. The reported means summarize the selected car and pedestrian evaluations on that partition, not all possible videos or operating conditions. Because the same partition is used for choosing temporal settings, the measurements are useful for understanding the intervention and identifying a candidate configuration, but they cannot independently establish how that selection will perform on unseen benchmark test data. This is the reason the manuscript consistently describes the split as a development split rather than presenting it as an untouched test estimate.

For MOT17, the first-half training and second-half validation protocol defines both what the detector is trained on and which frames are scored locally. The reported values belong to that protocol. They should be read with the detector checkpoint, validation frames, and association settings in mind. A published test-server number can use a different data partition and a different detection system even when its column headings use the same metric names. Matching the names of HOTA, MOTA, and IDF1 is therefore insufficient to make two experiments directly comparable.

The controls reduce several alternative explanations for a difference in the final tracking metrics. Holding the checkpoint fixed rules out a change in learned detector weights within a temporal comparison. Holding the proposal count fixed rules out a larger search budget. Holding association settings fixed keeps the downstream matching policy from changing at the same time as initialization. Keeping frame order and serialization settings fixed preserves the input sequence and the detections made available to the tracker. The remaining interpretation is still empirical, but these controls make it more specifically about proposal initialization.

The evaluation measures describe related but different aspects of the output. Detection recall and false negatives concern objects that the system misses, whereas precision and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

false positives concern detections that should not have been retained. Identity-oriented measures address whether detections are connected consistently over time. A change can improve one of these aspects while worsening another. Reporting several measures is therefore not redundant: it makes visible whether a gain comes from recovering objects, reducing spurious detections, preserving identity, or some combination of these effects.

The three-seed experiment estimates variation associated with the sampled inference runs under the stated configuration. Matching seeds makes the comparison paired: each temporal run has a corresponding random-control run under the matched seed schedule. The paired differences are more directly relevant to the initialization comparison than unrelated averages from different runs. Nevertheless, only three seed pairs are available. The Student-t intervals reported for their differences are descriptive summaries of this small set, and the manuscript does not treat them as a substitute for broader evaluation across data partitions or experimental conditions.

The distinction between detector output and tracker output remains relevant when checking the evaluation artifacts. Serialized detections document the image-level predictions provided to the association backends, whereas tracking metrics describe the trajectories produced after association and preprocessing. Inspecting both prevents a change caused downstream from being attributed automatically to a change in the detector. The static comparison fixes the detections to isolate this downstream behavior. The temporal comparison then fixes the association configuration while allowing the proposal initialization to affect the detections that enter it.

No single control removes every limitation of an empirical study. A fixed checkpoint and seed schedule make the initialization comparison more specific, but they do not enlarge the development split or supply additional independent training runs. Similarly, an official evaluator improves consistency of metric computation without making a local validation partition equivalent to a benchmark test set. The manuscript combines these controls with explicit qualifications because they address different sources of ambiguity. The controlled comparison is useful precisely when its conclusion remains limited to the conditions actually evaluated.

- *Reproducibility Details*

The frozen KITTI checkpoint is the 17,999-iteration model configured for 200 proposals. The MOT17 checkpoint is the 29,999-iteration model with SHA-256 ffe40ff3...b20646fc; its complete digest is recorded in the experiment manifest. Temporal experiments use seeds 40244023, 40244024, and 40244025. The principal transferred configuration uses Kalman propagation, start timestep 250, noise scale 0.25, and four samples per propagated track. Controls use random proposals at timesteps 999 and 250. ByteTrack uses a 0.5 track threshold, 30-frame buffer, 0.8 association threshold, and nominal 30 frames/s on MOT17. Every sequence has a separate runtime checkpoint and can be restarted without changing its seed schedule. Measurements use an NVIDIA GeForce RTX 3070 Laptop GPU (8,192 MiB), PyTorch 2.0.1 with CUDA 11.8 and cuDNN 8.7, OpenCV 4.8.1, and NumPy 1.24.4.

The checkpoint identifiers and software environment specify which trained detector and execution setting produced the reported measurements. They should be distinguished from the temporal parameters, which govern how the already trained model is initialized during inference. This separation is central to reproducing the comparison: changing the checkpoint would change the detector, whereas changing the propagation, timestep, noise scale, or copies would change the temporal proposal configuration. Reproducing a reported difference requires keeping the former fixed while applying the documented alternatives for the latter.

The sequence-specific runtime organization is also part of the reproducibility context. A restarted sequence should retain its intended seed schedule rather than receiving an unrelated sequence of random initializations solely because execution was interrupted. Separate runtime checkpoints support this purpose. This does not add a new evaluation result, but explains why the execution details accompany the model and parameter settings. They support consistency between the serialized outputs, the paired seed comparisons, and the sequence-level metrics subsequently produced by the evaluation stack.

- *Evaluation Artifacts*

Machine-readable artifacts include combined and per-sequence TrackEval results, paired within-seed differences, scenario tables, and compute and wall-clock runtimes. The released scripts regenerate TrackEval's MOTChallenge directory structure from original frame numbers, renumber validation frames locally, and retain official preprocessing.

Combined scores and per-sequence results answer different questions. The combined values provide a compact summary of the protocol, while the sequence-level outputs allow variation within that protocol to be inspected. Scenario tables similarly provide diagnostic views without replacing the official headline measures. Keeping these outputs distinct helps prevent a selected scenario or a favorable sequence from being presented as if it represented the complete evaluation. The paper's conclusions are based on the complete-sequence comparisons, with the more detailed artifacts serving to explain where those aggregate outcomes come from.

Frame numbering is an interface between saved detections and the evaluator. The described conversion from original frame identifiers to local validation numbering ensures that the serialized predictions are presented in the structure expected by the evaluation tools. The goal is consistency with the chosen validation protocol, not a change to which observations count as predictions or ground truth. Retaining official preprocessing is important for the same reason: a difference in conversion or preprocessing could otherwise be confused with a difference in the detector or tracker.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

III. RESULTS AND DISCUSSION

➢ *Static Association Baselines*

Table 1 compares trackers using identical detections. On KITTI, ByteTrack gives the strongest aggregate results of the three backends. On MOT17 half-validation, ByteTrack gains recall at a cost in precision: its CLEAR recall is 65.09% versus 56.57% for OC-SORT, while precision is 87.90% versus 91.98%. This yields 4,601 fewer false negatives but 2,175 more false positives and 71 more identity switches.

Table 1 Static DiffusionDet Detections with Three Association Backends. KITTI Entries are Mean Car/Pedestrian Values on the Development Split; MOT17 Entries use the Combined Validation-Half Evaluation.

Dataset	Tracker	HOTA	MOTA	IDF1
KITTI	SORT	50.75	57.66	66.70
KITTI	OC-SORT	52.53	58.91	68.66
KITTI	ByteTrack	54.48	62.14	71.96
MOT17	SORT	45.33	50.98	52.70
MOT17	OC-SORT	49.98	51.28	60.34
MOT17	ByteTrack	52.63	55.63	63.11

The static baseline comparison establishes the behavior of the association backends before temporal proposal feedback is considered. Because all three trackers receive identical serialized detections, the comparison focuses on what the association stage does with those detections. It should not be interpreted as a comparison between three separately optimized detector-tracker systems. This common-input design is useful for choosing the downstream association setting used in the subsequent experiments and for showing that changes in tracking metrics can occur even when detector output is held unchanged.

The MOT17 recall and precision comparison illustrates why one aggregate score cannot describe the whole change. Recovering more detections reduces misses, but retaining additional detections can also increase false positives and create more opportunities for identity errors. The reported counts show both sides of this trade-off in the static comparison. ByteTrack's stronger aggregate results in this setting therefore do not imply that every error measure improves. The temporal experiments use the same broader interpretation: an improvement must be located in its detection and association components rather than inferred from a single favorable number.

The KITTI and MOT17 rows should also be read within their own protocols. The KITTI entries average the two headline classes on the development split, whereas the MOT17 entries summarize the local validation-half evaluation. Their numerical scales are not evidence that one dataset is inherently easier or that a tracker will preserve the same ranking in every setting. The table documents these particular comparisons with the stated detector outputs. Its role is to establish a controlled baseline, not to provide a universal ranking of tracking algorithms.

➢ *KITTI Temporal Ablation*

Full random initialization at t = 999 obtains 54.48 mean HOTA, 62.14 MOTA, and 71.96 IDF1. A random t = 250 control obtains 54.02, 61.52, and 71.14, showing that a shorter start alone does not produce the gain. Four Kalman-centered proposals per track at t = 250 and $\lambda = 0.25$ obtain 54.50 HOTA, 62.81 MOTA, and 71.29 IDF1. Relative to the matched t = 250 control, the differences are +0.48 HOTA, +1.29 MOTA, and +0.15 IDF1. Relative to full random initialization, they are +0.02, +0.66, and -0.67. The temporal variant reduces false negatives for both official classes but does not improve identity quality relative to the original initialization.

Recovering missed detections does not automatically preserve identities. The lower false-negative count and the IDF1 comparison therefore support different conclusions: better detection coverage, but no uniform improvement in identity consistency.

The two random initializations answer an important preliminary question: whether shortening the diffusion start is itself sufficient to explain the observed improvement. The short-timestep random control performs differently from full random initialization, even though neither uses a propagated tracker location. It follows that comparing the temporal configuration only against the full-start baseline would combine the temporal change with the start-timestep change. The matched control makes the interpretation narrower by comparing random and temporal initialization at the same shorter start.

Against that matched control, the reported differences indicate that the selected temporal configuration recovers some of the performance lost by changing the start and improves the measured MOTA further. Against full random initialization, however, the pattern is more mixed: HOTA is effectively tied and IDF1 is lower. These are not contradictory descriptions. They use different reference conditions. Stating both comparisons prevents the stronger matched-control gain from being mistaken for a uniform gain over the original detector initialization.

The false-negative reduction is consistent with the purpose of allocating proposals around continuing tracks. A location estimate from the preceding tracker state can make those locations better represented in the proposal set. However, the identity measure records a different requirement: detections must also remain connected to the correct trajectories. The observed IDF1 comparison shows why a useful geometric prior cannot by itself establish better identity tracking. The paper therefore retains the narrower interpretation even though the MOTA change is favorable.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

The selected configuration is a result of the development-stage analysis. Its value for the study lies in providing a documented setting that can then be transferred to a second dataset. The KITTI comparison should not be stretched into a claim that the same timestep, noise scale, and copy count are optimal more generally. The later MOT17 experiment addresses transfer of this particular setting. It does not remove the need to distinguish configuration selection from independent evaluation when interpreting the KITTI values.

The use of mean car/pedestrian values should be kept in view when interpreting small aggregate differences. A mean condenses the selected class results into one headline value and does not describe every class or scenario separately. The paper therefore identifies the headline classes and treats cyclist, size, and occlusion analyses as supplemental diagnostics. This reporting choice prevents a favorable aggregate change from being interpreted as evidence of a uniform improvement for every object category or every visual condition in the development videos.

➢ *MOT17 Matched-Seed Temporal Evaluation*

The transferred Kalman configuration changes HOTA/MOTA/IDF1 by +1.67/+0.42/+1.41 points relative to the matched t = 250 random control, and by +0.99/+0.17/+0.49 relative to full random initialization. The paired 95% interval for the HOTA change is [0.94, 2.39] against the short-timestep control and [0.36, 1.63] against full random initialization; corresponding MOTA and IDF1 intervals include zero. Its mean CLEAR detection recall is 67.72% and precision is 85.75%. Because the configuration was selected on KITTI and then frozen, this experiment tests cross-dataset transfer rather than retuning on MOT17. The three-seed estimate remains too small for a strong significance claim.

The positive HOTA interval describes these matched runs; three seeds give limited information about broader variability. MOTA and IDF1 intervals containing zero do not establish reliable improvements. Keeping the selected configuration fixed avoids another tuning step on MOT17.

Table 2 MOT17 Half-Validation Mean ± Sample Standard Deviation Across Three Matched Inference Seeds.

Initialization	HOTA	MOTA	IDF1
Random, t = 999	52.69 ± 0.34	55.71 ± 0.08	63.42 ± 0.42
Random, t = 250	52.01 ± 0.21	55.46 ± 0.16	62.50 ± 0.48
Kalman, t = 250	53.68 ± 0.09	55.88 ± 0.11	63.91 ± 0.30

The MOT17 comparison extends the experiment by applying the chosen temporal setting to a separately trained detector in the stated half-validation protocol. The temporal setting is not retuned on those validation results. This makes the result evidence about transfer of the selected initialization rule, rather than evidence about a second search for the best temporal parameters. The distinction is especially relevant when interpreting small differences: a transferred setting answers a different question from a setting selected because it already performed well on the evaluation partition.

The average HOTA gain is reported relative to both random controls, and the paired intervals for that gain remain positive in the three observed seed pairs. This is a more consistent direction than is shown for MOTA and IDF1, whose intervals include zero. The manuscript therefore separates a positive observed HOTA pattern from a stronger claim that all tracking measures improve reliably. The latter would require evidence that the reported small-sample intervals do not provide. Retaining this distinction keeps the interpretation aligned with both the means and their uncertainty summaries.

The recall and precision values further qualify the average improvement. Increasing recall means fewer relevant detections are missed under the evaluation protocol, while a lower precision indicates a larger share of retained detections is incorrect. A temporal prior can therefore change the balance of detection errors rather than simply remove errors. This trade-off is compatible with the paper's overall conclusion that the principal benefit is related to detection recall. It does not establish that the temporal configuration is preferable under every application's relative cost of missed and spurious detections.

The standard deviations in Table 2 describe variation among the three inference seeds for each configuration. The paired intervals describe variation in the differences between matched configurations. These summaries should not be conflated: a small spread within one configuration does not by itself quantify the uncertainty of every comparison with another configuration. Both are reported to make the evidence inspectable. With only three runs, neither summary should be used to imply that the effects have been characterized comprehensively across all possible stochastic executions.

The comparison with full random initialization and the comparison with the matched short-timestep control have complementary roles here as well. The former locates the transferred method relative to the original initialization setting. The latter asks whether tracker-conditioned locations help within the selected shorter sampling start. Reporting both avoids choosing a reference condition solely because it produces the larger apparent difference. The positive average HOTA changes can be described alongside the smaller MOTA and IDF1 effects without implying that all three measures provide equally strong evidence.

➢ *Runtime Analysis*

The static detector runs at 6.51 frames/s on an RTX 3070 Laptop GPU; tracker-inclusive compute throughput is 6.35–6.42 frames/s. Temporal variants include detector and feedback time directly. Fig. 2 reports the KITTI speed-accuracy tradeoff.

The runtime measurements accompany the accuracy results because temporal initialization changes the inference procedure even when the learned detector is frozen. The static

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

detector rate and the tracker-inclusive rates refer to different scopes of computation. The temporal variants include the detector and feedback time directly, as specified in the experiment. These scopes matter when reading the throughput comparison: a detector-only figure should not be treated as identical to a rate that includes the association or feedback work required by the complete tracking loop.

Figure 2 places the measured compute throughput beside mean car/pedestrian MOTA for the KITTI development setting. Its purpose is to show the observed relationship between these quantities within the experiment. It does not establish that a configuration dominates under another device, another image workload, or another evaluation protocol. The hardware and software details supplied with the experiment identify the setting of the measurements. The accuracy interpretation remains subject to the same development-split qualification as the other KITTI results.

➢ *Published MOT17 Context*

Published private-detector MOT17 test results are substantially higher: ByteTrack reports 63.1 HOTA, 80.3 MOTA, and 77.3 IDF1 [5]; OC-SORT reports 63.2, 78.0, and 77.5 [6]; FairMOT reports 59.3, 73.7, and 72.3 [7]. These systems use different detectors, training data, and the hidden test set. They are protocol-qualified context, not direct baselines against our half-validation experiment.

Table 3 Protocol-Qualified MOT17 Context. Published Hidden-Test Values are Not Directly Comparable to Our Local Half-Validation Result.

System	Protocol	HOTA	MOTA	IDF1
Ours + ByteTrack	val-half	52.63	55.63	63.11
ByteTrack [5]	test	63.10	80.30	77.30
OC-SORT [6]	test	63.20	78.00	77.50
FairMOT [7]	test	59.30	73.70	72.30

The published test results provide context for the scale of established systems, but the protocol column is essential to their interpretation. A hidden-test evaluation and a local half-validation evaluation do not use the same observations, and the systems can differ in detection, training, and implementation. Their scores can therefore be placed together for orientation only when those differences remain explicit. The manuscript does not use the table to rank the proposed initialization rule directly against the published systems or to infer a test-set performance for the local model.

This restriction is compatible with a meaningful local ablation. A controlled comparison can show how one intervention affects a fixed pipeline even when that pipeline is not competitive with the strongest published benchmark systems. The contribution then concerns understanding the intervention under stated conditions. Here, the temporal comparison asks whether tracker-conditioned proposals change a frozen detector's tracking output. The published context prevents that narrower empirical question from being confused with a claim of state-of-the-art benchmark performance.

IV. LIMITATIONS

The KITTI development split is small, class-imbalanced, and used for temporal setting selection; its effect therefore needs independent confirmation. Three stochastic seeds provide only a coarse uncertainty estimate. The method uses box state without an appearance representation, so prolonged full occlusion can still produce identity errors. Temporal feedback can reinforce false positives or stale trajectories, and mixed random proposals remain necessary for new objects. Scenario metrics are diagnostic rather than official benchmark results. Finally, neither local validation split establishes state of the art, and published test-server values cannot be compared without matching data and submission protocols.

The limitations arise at several levels and should not be collapsed into a single uncertainty statement. The development split limits how broadly the selected KITTI effect can be generalized. The small number of inference seeds limits the characterization of stochastic variation. The use of geometric tracker state limits what information is available to maintain identities through difficult situations. Finally, the use of local validation protocols limits comparison with hidden-test benchmark reports. A favorable result at one level does not automatically resolve the limitations at the others.

Geometric continuity is informative but incomplete. A propagated box describes a location hypothesis derived from a trajectory, whereas identity consistency requires that the trajectory remain associated with the correct object. Long occlusion, stale tracks, and erroneous detections can weaken that connection. The reported method does not add an appearance representation to resolve such cases. This helps explain why the detection-recall benefit should be distinguished from identity improvement and why the mixed outcomes across HOTA, MOTA, and IDF1 remain central to the conclusion.

The feedback mechanism also means that the prior is only as useful as the state from which it is formed. When that state is misleading, repeatedly conditioning proposals around it may be unhelpful. Retaining random proposals provides coverage outside the current trajectory hypotheses, but the experiments do not show that this completely eliminates error reinforcement. The paper therefore presents random coverage as part of the method's design and a necessary safeguard, while leaving its sufficiency to the empirical results rather than assuming it guarantees recovery.

The stated limitations also constrain how the runtime observations should be used. The measured throughput characterizes the reported implementation and hardware, while the accuracy values characterize the selected data

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

protocol. Neither measurement by itself establishes an operating guarantee for another deployment. This does not diminish their value as experimental observations, but it clarifies their role: they document the cost and behavior of the evaluated initialization alternatives. The paper does not extend those measurements into claims about untested devices, workloads, or application-level reliability.

V. CONCLUSION

Tracker state can be inserted directly into a diffusion detector through its proposal distribution without retraining. On KITTI, low-noise Kalman proposals primarily improve MOTA and recall, while HOTA is tied and IDF1 does not improve over full random initialization. On MOT17, the frozen configuration improves mean HOTA by 0.99 points over full random initialization across three matched seeds, with a paired 95% interval of [0.36, 1.63]; the smaller MOTA and IDF1 intervals include zero. The evidence supports a narrow conclusion: temporal proposals can steer detection toward continuing objects, but association quality, error feedback, and new-object discovery remain limiting factors.

Taken together, the experiments support the use of tracker state as a practical source of proposal guidance within the evaluated frozen-detector pipeline. The evidence is strongest when expressed as a conditional change in the balance of detection and tracking errors. The favorable recall-related results coexist with identity limitations and protocol-dependent uncertainty. This interpretation preserves the distinction between showing that temporal initialization can help under the reported settings and claiming that it produces a general improvement in online multi-object tracking.

REFERENCES

[1]. Chen S, Sun P, Song Y, Luo P. DiffusionDet: diffusion model for objectdetection. In: CVPR; 2023.
[2]. Luo R, Song Z, Ma L, Wei J, Yang W, Yang M. DiffusionTrack: diffusion model for multi-object tracking. arXiv:2308.09905. 2024.
[3]. Hashmi KA, Stricker D, Afzal MZ. Spatio-temporal learnable proposalsfor end-to-end video object detection. In: BMVC; 2022.
[4]. Bewley A, Ge Z, Ott L, Ramos F, Upcroft B. Simple online and real-timetracking. In: ICIP; 2016.
[5]. Zhang Y, Sun P, Jiang Y, et al. ByteTrack: multi-object tracking byassociating every detection box. In: ECCV; 2022.
[6]. Cao J, Pang J, Weng X, Khirodkar R, Kitani K. Observation-centricSORT: rethinking SORT for robust multi-object tracking. In: CVPR; 2023.
[7]. Zhang Y, Wang C, Wang X, Zeng W, Liu W. FairMOT: on the fairnessof detection and re-identification in multiple object tracking. International Journal of Computer Vision. 2021.
[8]. Cai J, Xu M, Li W, Xiong Y, Xia W, Tu Z, et al. MeMOT: multi-objecttracking with memory. In: CVPR; 2022.
[9]. Luiten J, Osep A, Dendorfer P, Torr P, Geiger A, Leal-Taixe L, et al.HOTA: a higher order metric for evaluating multi-object tracking. International Journal of Computer Vision. 2021.
[10]. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving?The KITTI vision benchmark suite. In: CVPR; 2012.
[11]. Dendorfer P, Osep A, Milan A, et al. MOTChallenge: a benchmark forsingle-camera multiple object tracking. International Journal of Computer Vision. 2021.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep483

RUNTIME ANALYSIS

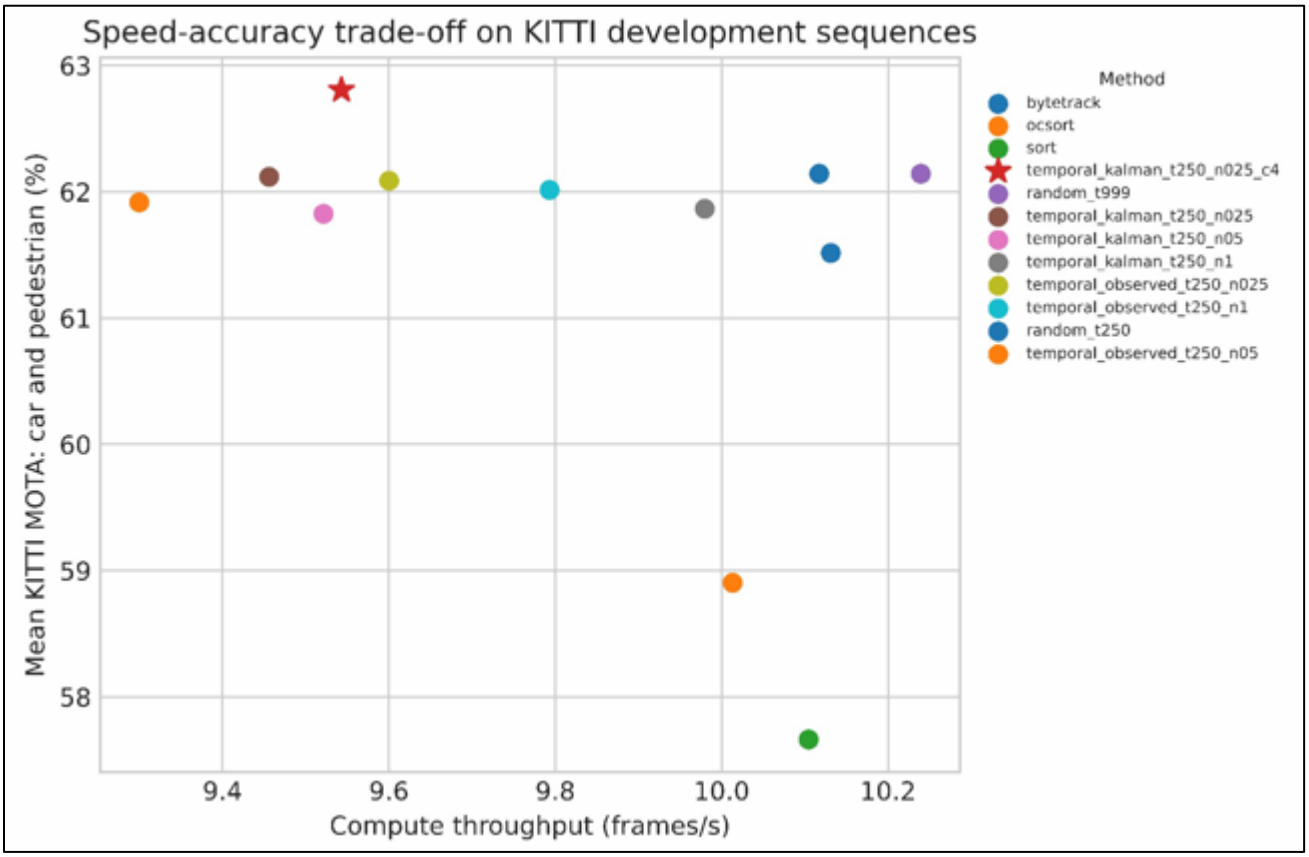

Fig 2 Measured KITTI Compute Throughput Versus Mean Car/Pedestrian MOTA.