Diffusion Models for Multi-target Adversarial Tracking

Sean Ye[1], Manisha Natarajan[1], Zixuan Wu[1], and Matthew C. Gombolay[1]

***Abstract*— Target tracking plays a crucial role in real-world scenarios, particularly in drug-trafficking interdiction, where the knowledge of an adversarial target's location is often limited. Improving autonomous tracking systems will enable unmanned aerial, surface, and underwater vehicles to better assist in interdicting smugglers that use manned surface, semi-submersible, and aerial vessels. As unmanned drones proliferate, accurate autonomous target estimation is even more crucial for security and safety. This paper presents Constrained Agent-based Diffusion for ENhanCEd Multi-Agent Tracking (CADENCE), an approach aimed at generating comprehensive predictions of adversary locations by leveraging past sparse state information. To assess the effectiveness of this approach, we evaluate predictions on single-target and multi-target pursuit environments, employing Monte-Carlo sampling of the diffusion model to estimate the probability associated with each generated trajectory. We propose a novel cross-attention based diffusion model that utilizes constraint-based sampling to generate multimodal track hypotheses. Our single-target model surpasses the performance of all baseline methods on Average Displacement Error (ADE) for predictions across all time horizons.**

I. Introduction

Unmanned aerial vehicles (UAVs) are extensively used in military and civilian applications, such as surveillance, search and rescue, anti-smuggling operations, wildlife tracking, and urban traffic monitoring [29]. These missions often involve tracking dynamic targets in large-scale environments, where predicting a target's current and future states is essential and internal states are not fully observable. However, tracking targets in complex environments, especially adversarial ones, presents significant challenges, including sparse observations and multiple possible future states for adversaries. As UAVs continue to advance and showcase their capabilities in diverse fields, refining methods for tracking dynamic targets is increasingly important.

While most works focus on single-target tracking in partially observable environments [22], [35], [36], the challenges becomes significantly more complex in multi-target tracking [2]. One significant challenge is the need to maintain distinct probability distributions for each target while correctly associating detections or observations with the different targets. Furthermore, maintaining multiple probability distributions of various tracks in multi-target settings involves handling track fragmentation (track splitting and merging) when targets interact with one another.

Earlier methods for target tracking encompass model-based approaches like Particle Filters [7], [15], [27] and Kalman Filters [3], [18]. However, these methods fail in sparse environments, where the vast majority of the time we do not receive any information on the target location. Model-free data-driven approaches [19], [24], can often outperform model-based approaches by estimating the behavior of the target with prior behavioral data rather than relying on expert-defined models. One model-free approach has shown promising results on this challenging task [36] by maximizing mutual information to regulate the components of a Gaussian Mixture Model. However, this model was limited to predicting a single time horizon and tracking a single target. Additionally the prior model is limited to a parametric formulation for the multimodal probability distributions by using a mixture of Gaussians.

In this work, we address single and multi-target tracking in large-scale pursuit environments using diffusion probabilistic models. Inspired by the recent success of diffusion models for trajectory generation in robotics [4], [14], [20], we propose a novel approach for target track reconstruction under partial observability. We design a novel approach named **C**onstrained **A**gent-based **D**iffusion for **EN**han**CE**d Multi-Target Tracking (CADENCE) that employs cross-attention to enable information exchange across different agents. A key benefit of diffusion models is their non-parametric formulation for generating multimodal hypotheses as compared to prior work. Additionally, we take inspiration from the computer vision community and adapt the classifier-guided sampling formulation to steer the trajectory generation process to adhere to motion model and environmental constraints.

Contributions: Our key contributions are:

- First, we propose CADENCE to track multiple adversaries, utilizing a cross-attention based diffusion architecture that implicitly conducts target track assignment between the agents.
- Second, we propose a constraint-guided sampling process for our diffusion models to ensure that state transition functions and obstacle constraints are satisfied in the track generation process, reducing collisions with obstacles by 90% compared to models without.
- Finally, we apply our diffusion models to generate track predictions for a single target, surpassing the performance of previous state-of-the-art (SOTA) models by an average of 9.2% in terms of Average Displacement Error. We additionally set a new baseline on the challenging task of multi-target tracking in a large,

*This work was supported in part by the Office of Naval Research (ONR) under grant numbers N00014-19-1-2076, N00014-22-1-2834, and N00173-21-1-G009, the National Science Foundation under grant CNS-2219755, and MIT Lincoln Laboratory grant number 7000437192.

[1]All authors are associated with the Institute of Robotics and Intelligent Machines (IRIM), Georgia Institute of Technology, Atlanta, GA 30308, USA.

partially observable domain.

II. Related Works

A. Diffusion Models

Deep diffusion models are a new class of generative models which model complex data distributions and have exploded in popularity within the computer vision community [5], [6], [13]. As these models have shown promise on learning within high dimensional data manifolds, other research areas have begun to apply diffusion models as powerful generative and conditional generative models. In robotics, a key work by Janner et al. [14] shows that diffusion models can be used to generate plausible paths for planning. Diffusion policy [4] extends this to work to imitation learning by diffusing the action distributions to accomplish various pushing tasks. Recent contemporary work by Zhu et al. has also used cross-attention within diffusion models [37] to generate multi-agent tracks. However, their work assumes full observability which is not available in the adversarial tracking domain. We also take inspiration from image inpainting (reconstructing missing parts of an image) [1] to condition the diffusion sampling process on detections for producing better target track predictions. *To the best of our knowledge, we are the first to utilize diffusion models for multi-target tracking under partial observability.*

B. Target Tracking

Target tracking involves estimating the positions of one or more targets using sensor data [33] and has various real-world applications such as surveillance [10], sports analysis [8], and traffic management [16]. Traditional approaches, like Particle Filters [7], [15], [27] and Kalman Filters [3], [18], dominate target tracking but require accurate knowledge or estimation of the target's dynamics model. However, recent advancements in model-free object tracking with images have emerged [31]. Our work differs from prior work in computer vision as we rely on sparse observations or detected locations instead of images to predict future target trajectories.

Adversarial tracking involves targeting an intelligent opponent trying to evade trackers [23]. Previous works assume access to target states/observations for training predictive models [9], [11], [26]. However, this assumption is unrealistic in large environments due to non-cooperative adversaries and a limited field of view. Prior work [36] introduced GrAMMI, which predicts dynamic target locations using partial observations from a team of trackers. However, it only focused on single target tracking and was unable to generate predictions for multiple time horizons. Our current work addresses these limitations by utilizing diffusion models to generate trajectories up to any time horizon and extending them to multi-target tracking.

III. Background

A. Partially Observable Markov Game

We define adversarial tracking as a Partially Observable Markov Game (POMG), which consists of a set of states $\mathcal{S}$, a set of private agent observations $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_M$, a set of actions $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_M$, and a transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_M \mapsto \mathcal{S}$ for M-agents. At each time step t, agents receive an observation $O_i^t \in \mathcal{O}_i$, choose an action $a_i^t \in \mathcal{A}_i$, and receive a reward r_i^t based on the reward function $R : \mathcal{S} \times \mathcal{A}_i \mapsto \mathbb{R}$. The initial state is drawn from an initial state distribution ρ.

We simplify the observation formulation of all agents to produce a single array of detections for all targets denoted as $\{d\}_{1\ldots t}$. We denote the trajectory (τ) of adversary states as $\tau_n \ \forall n \in N$, where N is the number of targets being tracked. Thus, the goal of CADENCE is to estimate the joint trajectory of all agents $p_\theta(\tau_{1\ldots n}|\{d\}_{1\ldots t})$.

In this work, we address multi-target and single-target tracking. For the multi-target tracking case, we ablate an assumption, where we either assume we know or *do not* know the origin of a given detection. If we assume we know the detection origin, then we do not have to solve the data association problem. Otherwise, the model must perform target assignment to distinguish the paths.

B. Diffusion Probabilistic Models

Diffusion models are a class of generative models that learn a target distribution through an iterative denoising process $p_\theta(x^{i-1}|x^i)$. The model learns how to reverse the forward noising process $q(x^i|x^{i-1})$, which is commonly parameterized as a Gaussian $\mathcal{N} \sim (0, I)$. Traditionally, x is used to represent images but in our work, we replace this notation with τ as we are generating trajectories.

The training process consists of a noising and denoising process. We utilize the Denoising Diffusion Probablistic Model (DDPM) formulation [12] and create noisy trajectories with Equation 1, where $\bar{\alpha}^i$ is a noise scheduler dependent on the diffusion process timestep i.

$$q(\tau^i|\tau^0) := \mathcal{N}(\tau^i; \sqrt{\bar{\alpha}^i}\tau^0, (1 - \bar{\alpha}^i)\mathbf{I}) \tag{1}$$

We learn a denoising network ϵ_θ to predict the random noise at all denoising iterations i (Equation 2).

$$\mathcal{L} = MSE\left(\epsilon^i, \epsilon_\theta(\sqrt{\bar{\alpha}^i}\tau^0 + \sqrt{1 - \bar{\alpha}^i}\epsilon, i)\right) \tag{2}$$

Finally, with a trained denoising network ϵ_θ, we can iteratively denoise a trajectory τ from pure Gaussian noise using Equation 3, which is equivalent to minimizing the negative log-likelihood of the samples generated by the model distribution under the expectation of the data distribution [30].

$$\tau^{i-1} = \frac{1}{\sqrt{\alpha^i}}\left(\tau^i - \frac{1 - \alpha^i}{\sqrt{1 - \bar{\alpha}^i}}\epsilon_\theta(\tau^i, i)\right) + \mathcal{N}(0, \sigma^2 I) \tag{3}$$

C. Domains and Target Behavior

We test our models in the Prison Escape and Narco Traffic Interdiction (Smuggler) domains first described in [36].

1) Narco Traffic Interdiction: This simulation involves illegal maritime drug trafficking along the Central American Pacific Coastline. A team of tracker agents, including airplanes and marine vessels, pursue a drug smuggler. Airplanes have a greater search radius and speed, while vessels can

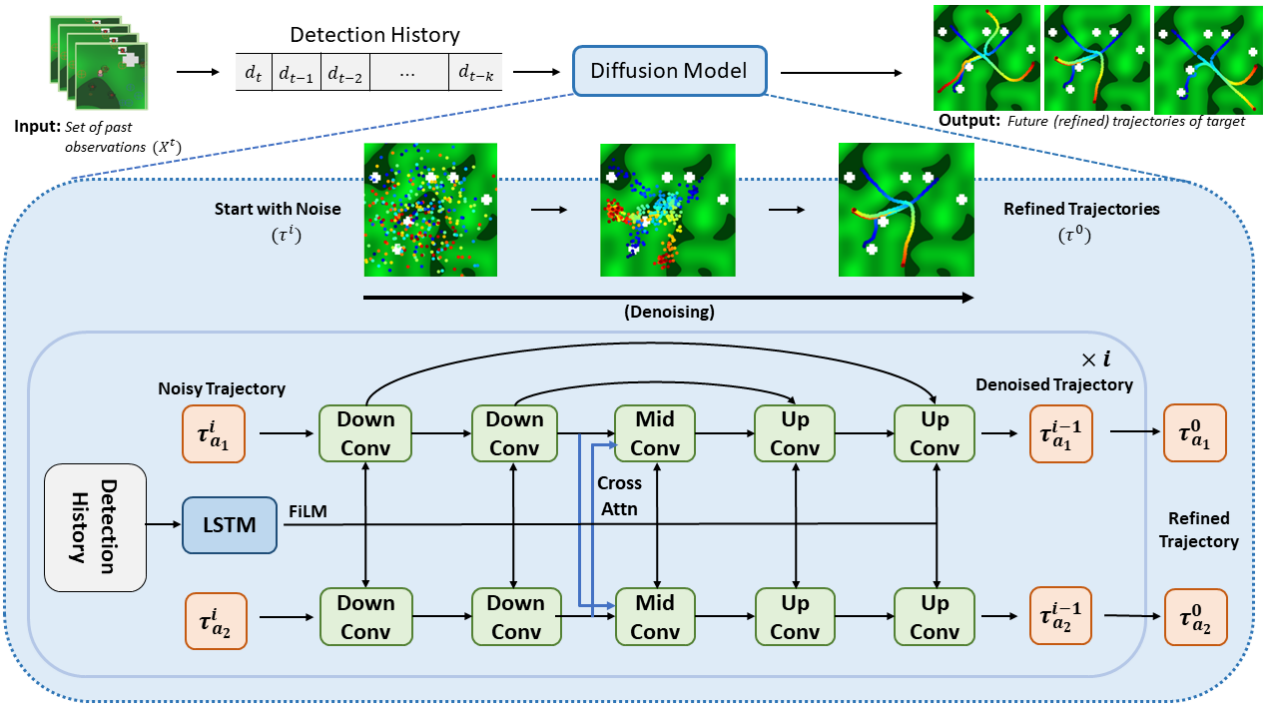

Fig. 1: Our proposed architecture (CADENCE) is a diffusion probabilistic model that iteratively refines trajectories for hypothesizing the future states of an adversarial target from a given history of past detections. Here we show cross attention across two parallel tracks but the model can be used for any number of n agents. The color gradient shows the time horizon for each trajectory (blue to red).

capture the smuggler. The smugglers must reach rendezvous points before heading to hideouts. The tracking team is unaware of hideout and rendezvous point locations. Episodes end when the smuggler reaches a hideout or is captured.

2) Prisoner Escape: In this scenario, a team consisting of cameras, search parties, and helicopters work together to track down an escaped prisoner, which is heading to one of several goal locations (hideouts). The game takes place on a map measuring 2428×2428 units, with various mountains representing obstacles. The motivation behind this domain stems from situations encountered in military surveillance and border patrol, where the objective is to track and intercept adversaries.

Goal locations for the fugitive are randomly selected without replacement from a predetermined set for each episode. The Prison Escape scenario incorporates evasive behaviors for the prisoner and introduces a fog-of-war element that limits the detection range of all agents. Notably, the tracking agents are only capable of tracking the prisoner and not capturing them, allowing for analysis of long-term predictions. The episode concludes either when the prisoner reaches a goal location or after a maximum number of timesteps.

IV. Methodology

Given sparse observations of the target $\{d\}_{1\ldots t}$, our diffusion model generates possible paths of the various target(s) τ_n. We describe CADENCE's design architecture for the multi-target and single-target domain.

A. Model Architecture

The multi-target diffusion model consists of two key components, 1) the temporal U-net and 2) the cross-attention mechanism between parallel tracks.

1) Temporal U-net: We utilize a 1D temporal CNN-based architecture from [4] as our noise prediction network ϵ_θ for each target agent in the environment. This architecture predicts an entire trajectory non-autoregressively and uses temporal convolutional blocks to encode the trajectory. Within the diffusion framework, the temporal U-net takes as input a noisy trajectory and outputs a refined (less noisy) trajectory. We repeat this i times to produce a noiseless trajectory from the noisy one. For a agents in the environment, we have use a parallel denoising networks, where each pathway denoises the trajectory for a single agent.

We utilize Feature-wise Linear Modulation (FiLM) at each convolutional layer [25] to condition the generative process on past detections $\{d\}_{1\ldots t}$. The history of past detections $\{d\}_{1\ldots t}$ is encoded in an LSTM, where each detection consists of the $\delta t, x, y$ denoting the time since detection and location of the detection. Figure 1 shows the full multi-agent denoising architecture.

2) Cross Attention: A key assumption in CADENCE is that the track generation is permutation equivariant— the ordering of the track inputs does not impact the results. This is achieved by sharing parameters between the track generators and using cross-attention to communicate information from one track to the other. The cross-attention formulation is a

variant of the scaled dot product attention [34]:

$$x^{m\prime} = \sum_m \text{softmax}\left(\frac{Q^m {K^n}^T}{\sqrt{dim_k}}\right) V^n \tag{4}$$

where $Q \in \mathbb{R}^{N\times dim}, K \in \mathbb{R}^{N\times dim}, V \in \mathbb{R}^{N\times dim}$ are vectors of query, key, and value. N is the number of query, key, and value vectors, dim is the dimension of the vector, and m, n are the indices for each target agent A. The attention module can be interpreted as a combination of both self-attention ($m = n$) and cross-attention across other agents ($m \neq n$). Crucially, the computation can be batched as the key, queries, and values k, q, v for each agent track only needs to be computed once for each agent. Finally, we adopt the multi-headed attention formalism and concatenate multiple heads to produce $x^{m\prime}$. We use the cross attention embeddings by interspersing them between the convolutional blocks in the U-net.

3) Single-Target Architecture: In the special case where we are only tracking a single agent, the cross-attention module is not used. In this formulation, a single history of detections is passed through the LSTM, and the model generates a single trajectory.

B. Constraint-Guided Sampling

Within our domain, two dominant constraints exist: 1) the motion model of target agents and 2) obstacle (mountains) constraints. In this work, we adapt classifier-based guidance [6], which was first used in image diffusion models to steer models towards certain classes. Classifier-based guidance uses a trained discriminative model to estimate $p(y|x)$, which denotes a class of an image based upon its input. The guidance augments the diffusion sampling procedure by changing the predicted means using the gradients of the classifier $\nabla_x \log p(y|x)$.

We adapt *classifier*-based sampling to *constraint*-based sampling by substituting the classifier with an objective function $J(\mu^i)$. We denote the mean of the trajectory we learn as μ and the sampled path as τ. Then, the guidance process can be written as $\tau^{i-1} \sim \mathcal{N}(\tau^i; \mu_\theta(\tau^i) + s\Sigma g, \Sigma)$, where the mean of the new distribution is perturbed by $g = \nabla_\tau J(\mu)$ and s is a gradient scale. In our implementation, we use an additional Adam optimizer to perform this gradient update (**Algorithm 1**). Using the Adam optimizer [17] relieves the need for hyper-parameter tuning s and allows us to combine multiple constraints together.

In our modified sampling algorithm (**Algorithm 1**), we begin with a completely noisy trajectory (line 2). Then, the denoising process occurs for T timesteps (line 3), where the denoised trajectory means are sampled from the model (line 4). We then use our constraint functions to move the means (line 6) and sample from the new distribution (line 8). Finally, we condition the model on detected locations at each diffusion timestep (line 9).

We use two constraint functions, one for the motion model and the second for the obstacles.

1) **Motion Model Constraint**: $\sum_t \|\tau_t - \tau_{t+1}\|$
   For each consecutive point in our trajectory, we create a simple smoothness loss such that consecutive states in the trajectory are close by.
2) **Obstacle Constraint**: $\|\tau_t - c\| < \epsilon \; \forall t \in T, c \in C$
   For each state (τ_t) in the trajectory and each obstacle C on the map, we provide a loss that pushes states away from obstacles in the environment.

Algorithm 1: Optimizer Based Constraint-Guided Sampling

1: **Input:** constraint function $J(\mu_\theta)$
2: $\tau^T \leftarrow$ sample from $\mathcal{N}(0, I)$
3: **for** all i from T to 1 **do**
4: $\quad(\mu^i, \Sigma^i) \leftarrow \mu_\theta(\tau^i), \Sigma_\theta(\tau^i)$
5: $\quad$**for** each gradient step **do**
6: $\quad\quad\mu^i \leftarrow \mu^i + \lambda \nabla_{\mu^i} J(\mu^i)$
7: $\quad$**end for**
8: $\quad\tau^{i-1} \sim \mathcal{N}(\mu^i, \Sigma^i)$
9: $\quad\tau_0 \leftarrow s_0$ if s_0 is known
10: **end for**
11: **Return** τ^0

C. Conditioning Detected Observations

While almost all the detected state information about the adversary is in the past, we provide a way to implement detected locations at the current time horizon $t = 0$ directly into the diffusion model sampling process. We alter the sampling process of the diffusion model, where if the detected location at the current timestep is known, we replace the sampled value with the known location after each diffusion timestep i (Algo 1, line 9).

Planning with diffusion models have used a similar process to goal-condition trajectories based on the starting and ending location [14]. This solution is inspired by inpainting [21], [30] in computer vision, where parts of an image are known and the diffusion model must generate the rest of the image.

V. Evaluation

We evaluate our single-target models for target tracking in the Prison Escape and Smuggler scenario introduced in [36] and use Monte-Carlo Sampling from the trained diffusion model to estimate the distribution of trajectories by generating 30 paths per sample. We use three datasets in the Prison Escape scenario (Prisoner-Low, Prisoner-Medium, Prisoner-High) that contain opponent detection rates of 12.9%, 44.0%, and 63.1%, respectively and two Smuggler datasets with opponent detection rates, 13.8% and 31.5%.

We create new multi-agent datasets within the same domain. However, we do not include target adversaries in this domain for simplicity. Instead, we randomly sample 10-12% of the timesteps and assume that these are the detected locations. These detected location samples are not resampled during training such that there is only one set of detections per trajectory rollout. We create two types of behavior 1) where the all the agents meet together before traveling to the goal location and 2) where the agents directly go to

the goal location. This assumption produces a multimodal distribution where the behaviors of the agents in the domain are dependent on each other. All agents use A^* to traverse through the landscape, where terrain with lower coverage visibility is preferred over higher coverage visibility. In the maps shown in Figure 2, the low visibility areas correspond to the darker regions of the map. In our environment, we choose three target agents to track but our model can be used to track any n number of targets.

In our analysis, we examine two scenarios regarding detections. The first scenario assumes that we have knowledge about the origin of each detection. The second scenario assumes that we *do not* have any information about the origin of the detections. In real-world target tracking situations, it is common for us to lack knowledge about the origin of the detections and, therefore, implicitly conduct target assignments.

A. Metrics

We evaluate CADENCE on two measures used in prior work [37] — Average Displacement Error (ADE, minADE) to compare against previous models. In the case of multi-agent tracking, we average the metrics across all agents. Previous work that fit a probability distribution included log-likelihood $\log(p(s_t|\theta))$, as a measure, Where θ is the model parameters. Computing the exact log-likelihood through the diffusion process is still an open research topic [32], where deterministic samplers based on probablistic flow ODEs have been used to compute exact likelihoods.

1) **Average Displacement Error (ADE)**: Given a ground truth trajectory τ, we compute the average l_2 distance between each sampled trajectory and the ground truth trajectory over all timesteps.
2) **Minimum Average Displacement Error (minADE)**: minADE measures the distance of the closest sampled path to the ground truth trajectory.

VI. Results & Discussion

We evaluate our multi-target models within the multi-target Prison Escape domain, showing ablations for knowing the origin of the detections and without knowing the origins of the detections.

The performance of our single-target tracking model was evaluated on the three Prison Escape datasets introduced in [36]. Our models show better ADE than the previous best Gaussian Mixture-based model (GMM) on every prediction horizon.

A. Multi-Target Tracking

We present findings regarding the performance of models with known detection origins compared to those without, as well as the qualitative analysis of the generated tracks in the multi-target tracking domain.

First, we are able to qualitatively show the multimodal behavior and dependencies between tracks with the cross-attention mechanism (Figure 2). Given the same inputs, the model produces tracks that do not intersect (left image) along with tracks that do intersect (right image), showing the models have learned the relationship between the tracks that is inherent within the dataset.

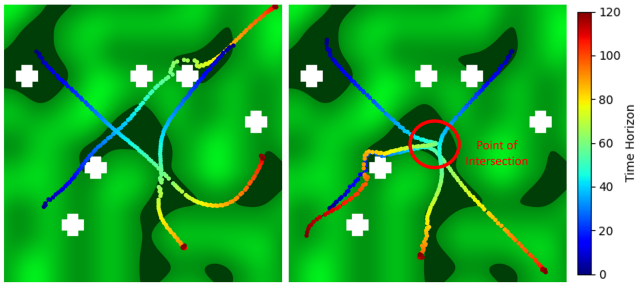

Fig. 2: Multi-Hypothesis Trajectory Samples: In one sample (left) the agents take separate paths to the final location while the other sample (right) shows all agents meeting at an intermediary point.

Detection Origin Ablation: We compare the ADE and minADE for models with detection origin information and those without (Figure 3). For models with origin information, we pass each target's detections through its own LSTM encoder to retrieve an embedding per target. This embedding is fed uniquely into each of the track generators. For models without detection origin information, we feed all detections through a single LSTM encoder whose embedding is shared amongst the diffusion tracks.

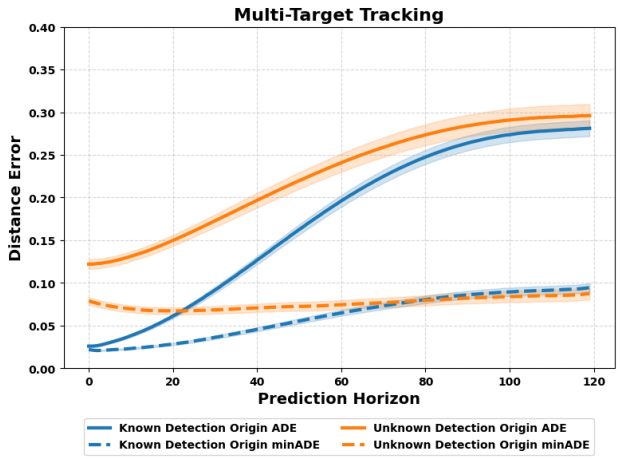

Fig. 3: Detection Origin Ablation: ADE and minADE over Prediction Horizon. Shaded regions represent standard error of the mean.

As expected, the models that possess knowledge of detection origins exhibit superior performance over all time horizons in terms of ADE and minADE compared to the models lacking such knowledge. Notably, there is a significant difference between ADE and minADE, indicating that while the distribution of paths is large, we can also generate hypothetical tracks that align well with the actual trajectories.

Furthermore, we observed that models without detection origins exhibit significantly higher error rates comparatively for shorter prediction horizons compared to longer horizons.

As the prediction horizon increases, trajectories are drawn to the discrete set of total goal locations, and the diffusion model directs trajectories towards these points. Therefore, in our domain, the challenge of assigning targets to the detections becomes more evident in shorter prediction horizons.

CADENCE is the first method capable of performing multi-target tracking under partial observability and implicitly conducts target assignment to generate consistent interactions between agents.

B. Constraint-Guided Samples

We compare how effective the constraint-guided sampling process prevents collisions with obstacles in the environment. A visualization of the difference is shown in Figure 4. We report our findings where samples generated without the constraint-sampling method resulted in an average of 5% of the produced states colliding with obstacles. On the other hand, states generated using the constraint-sampling method only encountered mountain collisions approximately 0.641% of the time. Our findings demonstrate that the new sampling procedure led to a significant 90% decrease in the number of collisions in the generated trajectories and produces more consistent hypothetical trajectories with actual trajectories.

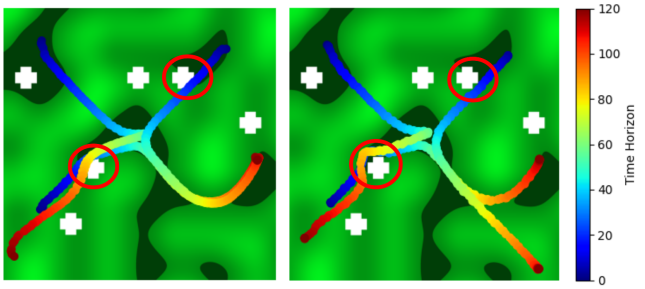

Fig. 4: Original sampling (left) vs Constraint-Based sampling (right): We show key improvements to avoid the mountains in the sample on the right (circled in red).

C. Single-Target Tracking

We present our results on single-target tracking using a unique version of the model without cross-attention, as there is only one agent. Our findings for single target tracking are presented based on the analysis of three Prisoner Escape and Smuggler datasets (Table I). The results demonstrate that diffusion models outperform the previous best Gaussian Mixture models on average by 9.2%. At higher prediction horizons of 60, 90 and 120 timesteps into the future, the diffusion models yield improvements of 12.2%. Additionally, our models are able to generate complete trajectories compared to the previous models which could only predict states at a fixed horizon length.

The non-parametric formulation of the diffusion model lends itself better for trajectory generation in our sparse detection environments than fitting a mixture of Gaussians. Fitting a mixture of Gaussians requires identifying the appropriate number of Gaussians — a hyperparameter for the target tracks. The diffusion models overcome this requirement and are able to represent a more diverse set of tracks.

		Prediction Horizon				
		0 min	30 min	60 min	90 min	120 min
P-Low	VRNN	0.106	0.093	0.119	0.146	0.177
	GrAMMI w/o MI	0.060	0.083	0.109	0.144	0.165
	GrAMMI	0.060	0.080	0.110	0.154	0.163
	Ours	**0.057**	**0.077**	**0.100**	**0.127**	**0.154**
P-Med	VRNN	0.172	0.086	0.110	0.144	0.167
	GrAMMI w/o MI	0.047	0.078	0.110	0.142	0.168
	GrAMMI	0.049	0.077	0.110	0.146	0.167
	Ours	**0.046**	**0.076**	**0.103**	**0.129**	**0.153**
P-High	VRNN	0.105	0.059	0.100	0.117	0.145
	GrAMMI w/o MI	0.016	0.057	0.095	0.132	0.167
	GrAMMI	**0.015**	0.056	0.092	0.122	0.162
	Ours	0.017	**0.054**	**0.078**	**0.099**	**0.118**
S-Low	VRNN	0.147	0.156	0.186	0.187	0.203
	GrAMMI w/o MI	0.122	0.142	0.159	0.169	0.182
	GrAMMI	0.121	0.144	0.181	0.183	0.193
	Ours	**0.112**	**0.123**	**0.135**	**0.148**	**0.160**
S-High	VRNN	0.138	0.153	0.183	0.179	0.185
	GrAMMI w/o MI	0.125	0.144	0.161	0.169	0.178
	GrAMMI	0.131	0.163	0.174	0.175	0.184
	Ours	**0.113**	**0.124**	**0.138**	**0.152**	**0.163**

TABLE I: ADE Results for three Prisoner Escape (P-low, P-med, P-high) and two Smuggler (S-low, S-high) Datasets. Bolded values represent the best-performing model.

By incorporating the complete trajectories generated by our diffusion models, we can enhance the capabilities of searching agents and improve target containment strategies. The availability of full track predictions allows us to anticipate the target's movements, identify potential escape routes, and strategically position agents to cut off those routes effectively. This enables us to employ more advanced policies for cornering the target and maximizing the chances of capture.

VII. Limitations

While our diffusion models show great improvements over previous models, a main limitation is the sampling time for generating tracks. Improving sampling speeds for diffusion models is currently an active area of research [28].

Additionally, our model can perform track prediction for future timesteps but implicitly learns the target track assignment from past detections. We, therefore, did not consider the task of track *reconstruction* in this work, where all trajectories generated are of future timesteps and not of the past. Finaly, we currently assume all agents are homogeneous in this work but could extend this to heterogeneous agents by removing the shared weights between track generators.

VIII. Conclusion

We proposed a novel approach using diffusion probabilistic models for single and multi-target tracking in large-scale environments. Our model incorporates cross-attention, constraint-guided sampling, and conditioning techniques to improve track prediction accuracy and adhere to motion model and environmental constraints. The experimental results demonstrate the effectiveness of the approach, surpassing the performance of previous state-of-the-art models in single-target tracking and achieving successful multi-target tracking with improved target track assignment.

REFERENCES

[1] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, 2000.

[2] Biswajit Bose, Xiaogang Wang, and Eric Grimson. Multi-class object tracking algorithm that handles fragmentation and grouping. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[3] Rong Chen and Jun S Liu. Mixture kalman filters. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(3):493–508, 2000.

[4] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[5] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[7] Petar M Djuric, Mahesh Vemula, and Mónica F Bugallo. Target tracking by particle filtering in binary sensor networks. *IEEE Transactions on signal processing*, 56(6):2229–2238, 2008.

[8] Rikke Gade and Thomas B Moeslund. Constrained multi-target tracking for team sports activities. *IPSJ Transactions on Computer Vision and Applications*, 10:1–11, 2018.

[9] Aditya Grover, Maruan Al-Shedivat, Jayesh Gupta, Yuri Burda, and Harrison Edwards. Learning policy representations in multiagent systems. In *International conference on machine learning*, pages 1802–1811. PMLR, 2018.

[10] Chao Gui and Prasant Mohapatra. Power conservation and quality of surveillance in target tracking sensor networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 129–143, 2004.

[11] He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *International conference on machine learning*, pages 1804–1813. PMLR, 2016.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[13] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022.

[14] Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, 2022.

[15] Li Jia-qiang, Zhao Rong-hua, Chen Jin-li, Zhao Chun-yan, and Zhu Yan-ping. Target tracking algorithm based on adaptive strong tracking particle filter. *IET Science, Measurement & Technology*, 10(7):704–710, 2016.

[16] Konstantinos Kanistras, Goncalo Martins, Matthew J Rutherford, and Kimon P Valavanis. A survey of unmanned aerial vehicles (uavs) for traffic monitoring. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 221–234. IEEE, 2013.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[18] William F Leven and Aaron D Lanterman. Unscented kalman filters for multiple target tracking with symmetric measurement equations. *IEEE Transactions on Automatic Control*, 54(2):370–375, 2009.

[19] Jingxian Liu, Zulin Wang, and Mai Xu. Deepmtt: A deep learning maneuvering target-tracking algorithm based on bidirectional lstm network. *Information Fusion*, 53:289–304, 2020.

[20] Weiyu Liu, Tucker Hermans, Sonia Chernova, and Chris Paxton. Structdiffusion: Object-centric diffusion for semantic rearrangement of novel objects. *arXiv preprint arXiv:2211.04604*, 2022.

[21] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022.

[22] Darko Musicki, Barbara F. La Scala, and Robin J. Evans. Integrated track splitting filter - efficient multi-scan single target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1409–1425, 2007.

[23] Samer Nashed and Shlomo Zilberstein. A survey of opponent modeling in adversarial domains. *Journal of Artificial Intelligence Research*, 73:277–327, 2022.

[24] Shuchao Pang, Juan José del Coz, Zhezhou Yu, Oscar Luaces, and Jorge Díez. Deep learning to frame objects for visual target tracking. *Engineering Applications of Artificial Intelligence*, 65:406–420, 2017.

[25] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[26] Roberta Raileanu, Emily Denton, Arthur Szlam, and Rob Fergus. Modeling others using oneself in multi-agent reinforcement learning. In *International conference on machine learning*, pages 4257–4266. PMLR, 2018.

[27] G Mallikarjuna Rao and Ch Satyanarayana. Visual object target tracking using particle filter: a survey. *International Journal of Image, Graphics and Signal Processing*, 5(6):1250, 2013.

[28] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

[29] Hazim Shakhatreh, Ahmad H. Sawalmeh, Ala Al-Fuqaha, Zuochao Dou, Eyad Almaita, Issa Khalil, Noor Shamsiah Othman, Abdallah Khreishah, and Mohsen Guizani. Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges. *IEEE Access*, 7:48572–48634, 2019.

[30] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR.

[31] Zahra Soleimanitaleb, Mohammad Ali Keyvanrad, and Ali Jafari. Object tracking methods: a review. In *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 282–288. IEEE, 2019.

[32] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[33] Éfren L Souza, Eduardo F Nakamura, and Richard W Pazzi. Target tracking for sensor networks: A survey. *ACM Computing Surveys (CSUR)*, 49(2):1–31, 2016.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[35] Jingjing Xiao, Rustam Stolkin, and Ales Leonardis. Single target tracking using adaptive clustered decision trees and dynamic multi-level appearance models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[36] Sean Ye, Manisha Natarajan, Zixuan Wu, Rohan Paleja, Letian Chen, and Matthew C. Gombolay. Learning models of adversarial agent behavior under partial observability. *Proceedings of the International Conference on Intelligent Robots and Systems*, 2023. To appear.

[37] Zhengbang Zhu, Minghuan Liu, Liyuan Mao, Bingyi Kang, Minkai Xu, Yong Yu, Stefano Ermon, and Weinan Zhang. Madiff: Offline multi-agent learning with diffusion models. *arXiv preprint arXiv:2305.17330*, 2023.