FIXATION-DRIVEN TIME-AWARE 3D HUMAN MOTION FORECASTING IN INDOOR SCENES

Anonymous authors
Paper under double-blind review

ABSTRACT

Forecasting human motion in indoor scenes is crucial for collaborative robotics and embodied AI. While prior approaches have incorporated gaze implicitly or used it only to rank segmented objects, we argue that gaze, particularly fixations, offers a more intentional and spatially precise signal for predicting human intent. In this work, we introduce a fixation-driven, time-aware framework for 3D human motion forecasting that explicitly supervises a gaze network to distinguish fixations from saccades, and uses fixation-weighted vectors to not only rank candidate objects but to also localize precise interaction points, improving robustness to segmentation errors. Our contribution further includes a duration prediction module that generates variable-length motion sequences, adapting to the spatial and temporal demands of the task. We evaluate our approach on the GIMO and GTA-IM datasets to show more accurate predictions particularly in challenging scenes with small or merged objects, and varying interaction durations through variable-length motion generation. Our code will be made publicly available.

1 INTRODUCTION

Forecasting human motion is critical for collaborative robots, with gaze being among the most reliable predictors of human intent, often preceding motor actions. For instance, wearable eye trackers have been shown to anticipate actions over one second in advance using gaze-centered visual input Li et al. (2018); Liu et al. (2020). Among gaze behaviors, fixations play a crucial role by stabilizing attention on task-relevant targets, providing strong signals for accurate motion prediction Foulsham (2015). In contrast, saccades, rapid gaze shifts between points, are less informative for forecasting motion. Explicitly modeling these distinct patterns is vital for reliable intention forecasting.

Previous research has focused on designing increasingly complex multimodal fusion architectures Zheng et al. (2022); Lou et al. (2024a) or leveraging instance segmentation Qi et al. (2017) to rank candidate objects using either prior motion cues Lou et al. (2024b) or gaze information Yu et al. (2025). While these approaches have reduced overall prediction errors, they typically treat gaze as an auxiliary or implicit signal. However, because gaze is a uniquely intentional and interpretable cue, it enables direct geometric inference of intent, which can be diluted when gaze is entangled with scene and motion features during fusion. Likewise, approaches that use gaze only to rank object segments lose the spatial precision afforded by fixation patterns.

In this work, we demonstrate the importance of explicitly modeling fixations as a core component of our method. We explicitly supervise the gaze network to distinguish fixations from saccades, producing more reliable gaze vectors for downstream tasks such as object prediction and pose forecasting. These fixation-weighted vectors are then used not only to rank objects but to also precisely localize interaction points within the scene, improving spatial precision and robustness to segmentation errors, particularly for small objects that are often merged with background segments. We additionally extend prior work Lou et al. (2024b), which forecasts fixed-length trajectories toward multiple candidate targets, by introducing a duration prediction module that generates variable-length trajectories. This enables the model to adapt motion duration to the spatial distance and context of each target, resulting in more realistic and temporally aligned forecasts. By integrating these novel contributions, our method achieves state-of-the-art performance on indoor motion prediction benchmarks.

Overall, our innovations are as follows: (1) We propose a fixation-based gaze classification module. Instead of treating all gaze points equally, we train the model to explicitly identify fixations, defined in this work as gaze directed toward the object of interest, and use only these fixations to guide intent. (2) We use fixations to pinpoint the exact surface a person intends to interact with, rather than relying on purely the object geometry as was done in prior work. (3) We train a network to predict how many frames are needed to reach the target object, depending on distance and action type. This lets our method be more versatile as it allows forecasting toward multiple objects.

2 Related Works

Gaze-Conditioned Pose Forecasting: Human pose forecasting is typically framed as a sequence-to-sequence learning problem, with methods differing primarily in their encoding and decoding strategies. These include autoregressive Martinez et al. (2017) and non-autoregressive Tevet et al. (2022) approaches, as well as deterministic Lyu et al. (2025) and stochastic Wang et al. (2024) formulations. While early models relied solely on the human pose as input, later advances integrate additional contextual cues, such as text Ahuja & Morency (2019); Xie et al. (2024), semantic action label Guo et al. (2020) audio Li et al. (2021); Han et al. (2024), eye gaze and 3D scene context, often jointly used Razali & Demiris (2022); Zheng et al. (2022); Lou et al. (2024a); Yu et al. (2025).

Most related to our work are Hu et al. (2024); Yan et al. (2023); Zheng et al. (2022); Lou et al. (2024a); Yu et al. (2025); Lou et al. (2024b), which predict 3D human motion using eye gaze. Yan et al. (2023) employs a diffusion model conditioned on gaze and motion to forecast 3D human motion, while Hu et al. (2024) predicts future gaze based solely on past gaze before using it to forecast motion. However, both methods ignore scene information, which makes them quite limited in indoor environments. GIMO Zheng et al. (2022) fuses gaze, scene, and motion features via cross-modal transformers Vaswani et al. (2017) to forecast 3D human motion in indoor scenes, but does not explicitly model object geometry. SIF3D Lou et al. (2024a) supervises point cloud saliency using eye gaze, but similarly does not incorporate object geometry or predict explicit interaction targets. DiMoP3D Lou et al. (2024b) segments the scene into object instances and ranks them using prior motion before generating fixed-length motion sequences toward each object. However, it uses motion cues rather than gaze, which are significantly less informative, especially in cluttered environments where multiple objects lie along similar trajectories. GAP3DS Yu et al. (2025) segments the scene into object instances, uses gaze to generate a distance-based heatmap, retrieves the object instance with the highest score, and then generates motion conditioned on the segmented object geometry. However, this approach relies heavily on segmentation quality. If a small object of interest, such as a cup on a table or a banana on the floor, is incorrectly merged into a larger background segment like the table or floor, the model will completely miss the object and possibly forecast toward a distant, unrelated target. Moreover, both DiMoP3D and GAP3DS generate motion based solely on object geometry without gaze. This makes precise motion generation especially challenging, as the retrieved object may afford interactions at many possible locations, lacking the spatial specificity needed for targeted action.

We show in this paper that explicitly using gaze not only to rank object-level targets, but also to localize the specific region within the object reduces reliance on segmentation and improves spatial precision. Additionally, a similarity shared across prior works is that none explicitly supervises the gaze signal during training. In contrast, we introduce a gaze-aware auxiliary loss that trains the model to distinguish between fixations and saccades, leading to more accurate object inference and localization.

Motion In-Betweening: Generating motion sequences from partially observed frames, commonly referred to as motion in-betweening, has been explored using a variety of modern architectures. Early work explored recurrent models Harvey et al. (2020) before adopting transformer-based architectures Petrovich et al. (2021) and diffusion models Cohan et al. (2024) to improve temporal coherence and sample diversity. These models are typically conditioned either on a motion trajectory or a sparse set of keyframes, often the start and end poses, to synthesize the intermediate motion. Both GAP3DS and DiMoP3D segment the scene to identify multiple candidate targets and generate corresponding secondary forecasts by in-betweening from the observed motion to each target's end pose. While these methods, and diffusion methods in general Guo et al. (2022) can support variable length inputs, they require the number of frames to be fixed in advance, either determined from the ground-truth duration or manually specified, which limits their adaptability in the context of indoor motion forecasting, where the number of frames depend on end point distance. In contrast, we explicitly train the model to predict the number of frames required to reach each target, conditioned on the start pose, end pose, and coarse trajectory. This enables our method to generate motion sequences of variable length that better reflect the spatial and temporal demands of each interaction.

3 Method

We let $\mathbf{x}_{-T_1:0} = [\mathbf{x}_{-T_1}, \ldots, \mathbf{x}_0] \in \mathbb{R}^{T_1 \times J \times 3}$ denote the observed human pose with J joints across T_1 timesteps, and $\mathbf{g}_{-T_1:0} \in \mathbb{R}^{T_1 \times 2 \times 3}$ the corresponding gaze vectors, represented as 3D origin-direction pairs per frame. We define the scene point cloud as a set of 3D coordinates $\{\mathbf{p}_i\}_{i=1}^{N}$, where $\mathbf{p}_i \in \mathbb{R}^3$ and N is the total number of points. Our goal is to forecast the human motion over T_2 timesteps, which can be expressed as $p(\mathbf{x}_{1:T_2} | \mathbf{x}_{-T_1:0}, \mathbf{g}_{-T_1:0}, \{\mathbf{p}_i\})$.

Human–object interactions in indoor scenes often follow structured, spatially consistent patterns. Objects like chairs, tables, and beds form distinct geometric clusters, and gaze-driven actions typically target a few semantically meaningful regions. However, not all gaze samples are equally informative as fixations indicate task-relevant attention, while saccades reflect transient scanning Foulsham (2015). By leveraging these insights, we can factorize our forecasting objective into the following:

$$\begin{aligned} &\underbrace{p(\{\mathbf{o}_j\} \mid \{\mathbf{p}_i\})}_{\text{Instance Segmentation}} \cdot \underbrace{p(\mathbf{g}_* \mid \mathbf{g}_{-T_1:0}, \{\mathbf{o}_i\})}_{\text{Fixation Classification}} \cdot \underbrace{p(j \mid \mathbf{g}_*, \{\mathbf{o}_j\})}_{\text{Object Ranking}} \\ &\cdot \underbrace{p(\mathbf{x}_{T_2} \mid \mathbf{o}_j, \mathbf{g}_*)}_{\text{Endpose Prediction}} \cdot \underbrace{p(\mathbf{x}_{1:T_2} \mid \mathbf{x}_0, \mathbf{x}_{T_2}, \{\mathbf{p}_i\})}_{\text{Motion In-Betweening}} \end{aligned} \tag{1}$$

where $\{\mathbf{o}_j\}_{j=1}^{J}$ denotes the set of J spatially coherent object candidates inferred from the point cloud $\{\mathbf{p}_i\}$, with each $\mathbf{o}_j \subseteq \{\mathbf{p}_i\}$ corresponding to a physical object such as a chair, table, or bed, including background regions. The vector $\mathbf{g}_* \in \mathbb{R}^{2\times 3}$ represents the fixation-aggregated gaze input. This formulation better reflects the structured nature of indoor human motion forecasting, by first segmenting the scene into object candidates (Instance Segmentation), identifying fixations (Fixation Classification), ranking interaction likelihoods based on fixations (Object Ranking), generating a pose conditioned on object geometry, and additionally on fixation if it intersects the object (Endpose Generation), and finally synthesizing the full motion sequence between the initial and final poses (Motion In-Betweening). Fig. 1 illustrates our method. We now describe each component.

Instance Segmentation: Reliable human-object motion forecasting first requires identifying candidate objects and their geometry. Without a structured scene representation, subsequent gaze localization would lack spatial grounding. We utilize PointNet++ Qi et al. (2017) to perform instance segmentation. PointNet++ is a hierarchical neural network that learns directly from raw point clouds by capturing local geometric structures through set abstraction layers. As shown in Fig 1, the output is a scene-level point cloud where each point is assigned to a spatially coherent cluster representing an object instance $\mathbf{o}_j$ (e.g., chair, table, bed, cup, phone), visualized in unique colors. In practice, small objects like phones, fruits, or utensils are often missed due to limited point cloud resolution. While advances in point cloud segmentation have resulted in improved overall segmentation, fine-grained objects remains challenging. Since end-pose prediction depends on correctly localizing the target object, segmentation alone is insufficient. We address this by incorporating gaze to localize the precise interaction region, as described in the section on Gaze-Aware End-Pose Prediction.

Fixation Classification and Object Ranking: Not all gaze points carry equal intent, fixations reveal purposeful attention while saccades reflect transient scanning. Explicitly distinguishing these behaviors allows the model to isolate meaningful intent cues before ranking object candidates. Existing works often model gaze implicitly. In particular, attention-based methods Zheng et al. (2022); Lou et al. (2024a) dilute gaze information over time, making it difficult to extract reliable fixation cues. To address this, we directly optimize the gaze input by framing it as a temporal classification task that emphasizes fixations and de-emphasizes saccades:

$$\begin{aligned} \hat{\mathbf{f}} &= \sigma(\mathbf{Transformer}(\mathbf{g}_{-T_1:0}, \{\mathbf{o}_j\})) \\ \mathbf{g}_* &= \frac{1}{Z} \sum_{t=0}^{T_1} 1[\hat{\mathbf{f}}_t > 0.5] \cdot \mathbf{g}_t \end{aligned} \tag{2}$$

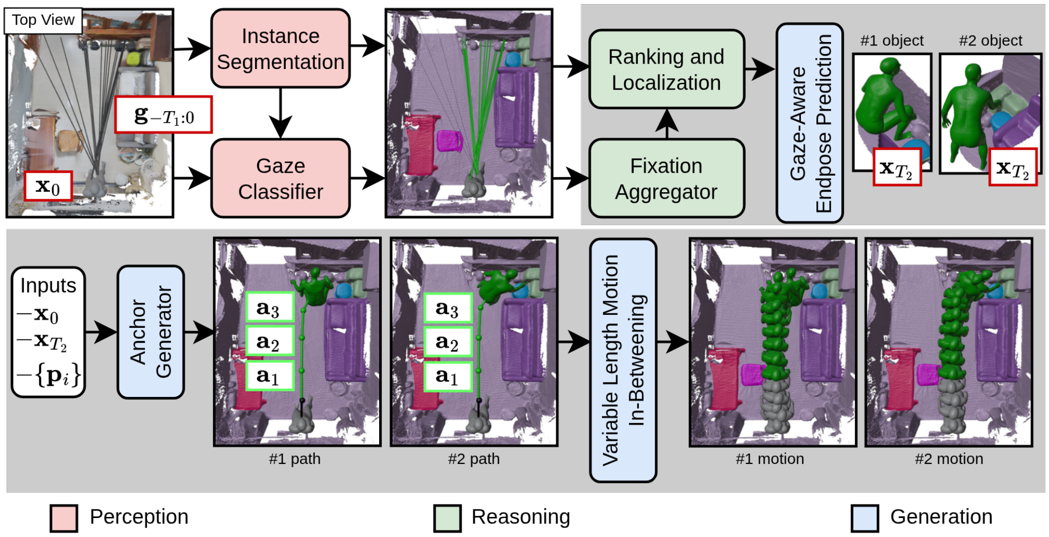


Figure 1: Overview of our method. Components are color-coded into three categories: perception (red), reasoning (green), and generation (blue). The inputs are the observed human pose x_0, raw gaze sequence $g_{-T_1:0}$, and the scene point cloud. The instance segmentation module extracts object instances (colored point clusters), while the gaze classifier identifies fixations (green rays), which are aggregated into a single fixation-weighted vector g^*. This vector is used to rank candidate objects and localizing the intended interaction point. The gaze-aware end-pose prediction module then predicts a final pose x_{T_2} for each ranked target. An A*-based anchor generator produces coarse anchors (a_1, a_2, a_3) between the start and predicted end pose, and a diffusion model performs variable-length in-betweening to synthesize the full trajectory. Predicted motions toward the top-ranked objects (#1 and #2) are shown on the right.

where $\hat{\mathbf{f}} \in [0,1]^{T_1}$ represents the predicted fixation likelihood over the input timesteps, and Z normalizes the sum. We define fixations in the context of this work as gaze points within 0.3m of the target in order to balance precision and noise tolerance. While we adopt this fixed threshold, it is dataset-dependent and could also be personalized for each user, which we leave for future work. We further incorporate scene context to improve fixation detection in cases of gaze drift across large objects such as sofas and beds. At test time, the aggregated gaze vector $\mathbf{g}_* \in \mathbb{R}^{2\times 3}$ is computed by averaging over fixations. Given $\mathbf{g}_*$ and the set of object instances $\mathbf{o}_j$, we rank the candidates by their proximity to the gaze:

$$\mathrm{rank}(j) = \mathrm{sort}_j \left(\min_{\mathbf{p}\in\mathbf{o}_j} \mathrm{dist}(\mathbf{g}_*, \mathbf{p}) \right) \tag{3}$$

Intuitively, this ranking corresponds to measuring the distance between the gaze ray and each object surface. In practice, the top-ranked object is typically the one directly intersected by the aggregated vector. Based on this, we obtain two types of candidates: primary, where the aggregated vector intersects the object instance (even if labeled as background), and secondary, where no intersection occurs. This distinction determines how the end-pose is predicted.

Gaze-Aware End-pose Prediction: Knowing where a person looks is not enough as the model must also infer how the body aligns with that target. Conditioning end-pose generation on both partial object geometry and the gaze intersection provides stronger spatial grounding, improves robustness to segmentation errors, and benefits the subsequent motion in-betweening stage. We employ a conditional Variational Autoencoder (cVAE) Kingma et al. (2013) to generate a static, whole-body pose for each of the top-ranked object candidates, conditioned on the object's Basis Point Set (BPS) Prokudin et al. (2019) representation, computed over a 1m radius around the object instance to capture not only the object shape, but also nearby obstacles that may influence the final pose. However, small objects like utensils are frequently missed during instance segmentation, which degrades the accuracy of the generated pose. To address this, we first compute a reference point $\mathbf{c}$, defined as:

$$\mathbf{c} = \begin{cases} \mathbf{p}_j & \text{if } \mathbf{p}_j \text{ intersects } \mathbf{g}_* \\ \frac{1}{|\mathbf{o}_j|} \sum\limits_{\mathbf{p} \in \mathbf{o}_j} \mathbf{p} & \text{otherwise} \end{cases} \tag{4}$$

and apply horizontal centering using only the horizontal components i.e. $[c_x, 0, c_z]$. This step intuitively lets the model observe the input, centered at the interaction point. The pose is then generated as:

$$\mathbf{x}_{T_2} = \mathbf{cVAE}(\mathbf{BPS}(\mathbf{o}_j - \mathbf{c})) \tag{5}$$

We use a separate cVAE per action type family (e.g., reach, sit, lie), selected based on the instance label, as it provides a practical compromise between generalization and specialization. For primary candidates, the reference point is derived from the aggregated gaze, yielding more precise localization even when the object is mislabeled as background (e.g., a banana on the floor). For secondary candidates, where no intersection is detected, the model falls back to centering using object geometry at its mean. This selective use of gaze distinguishes our approach from prior methods such as GAP3DS and DiMoP3D, which rely solely on object geometry and are unable to forecast motion toward unsegmented or merged objects. In practice, this distinction is crucial for reaching actions, where gaze helps identify small targets often merged into the background, unlike sitting, where object geometry alone typically suffices.

Variable Length Motion In-Betweening: Given the start and end poses, the model must synthesize realistic motion of appropriate duration. Predicting motion length internally, rather than fixing it manually, ensures temporal realism and adaptability across different interaction distances. Prior work Zheng et al. (2022); Lou et al. (2024a); Yu et al. (2025) requires manually specified lengths, often using the ground-truth number of output frames. While autoregressive models such as LSTMs can in principle produce variable-length trajectories Corona et al. (2020), they still require a stopping criteria and have been largely outperformed by diffusion models in quality. However, a limitation of diffusion models in the context of motion forecasting is that they require the number of frames to fixed in advance, which prevents them from flexibly handling variable-length outputs, which is particularly problematic in indoor motion forecasting where action durations naturally vary depending on end point distance. For example, if the horizon is set too short, the generated motion will reach the target unnaturally quickly. If it is too long, the motion will appear artificially slow. There is thus a need to internally predict motion length rather than controlling it manually from the outside.

To address this, we first follow Lou et al. (2024b) to generate a coarse trajectory from the final observed pose $\mathbf{x}_0$ to the predicted end-pose $\mathbf{x}_{T_2}$ using a graph-based planner such as A* Hart et al. (1968). This produces a sequence of anchor poses $[\mathbf{x}_0, \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{x}_{T_2}]$, where each $\mathbf{a}_i \in \mathbb{R}^{J \times 3}$ only contains the root ground-plane position and zeros elsewhere, in contrast to full poses $\mathbf{x}_t$ with all joints. We then use a 1D CNN to estimate the number of in-between frames τ_i for each consecutive pair of anchors:

$$\tau = \mathbf{CNN}([\mathbf{x}_0, \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{x}_{T_2}]) \tag{6}$$

where each τ_i denotes the number of in-between frames for the segment between each neighbouring anchor pair. Given the anchors and predicted frame counts, we construct a motion template $\mathbf{X}$ with placeholders between anchor poses:

$$\mathbf{X} = [\mathbf{x}_0, \underbrace{\mathbf{x}_0^{(1)}, \dots, \mathbf{x}_0^{(\tau_1)}}_{\tau_1 \text{ elements}}, a_1, \underbrace{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_1^{(\tau_2)}}_{\tau_2 \text{ elements}}, \dots, \mathbf{x}_{T_2}] \tag{7}$$

Each placeholder $\mathbf{x}_i^{(k)}$ is initially filled with Gaussian noise and later denoised by the diffusion model, which learns to generate smooth transitions conditioned on the surrounding anchor frames:

$$\begin{aligned} \mathbf{X}_t &= \sqrt{\bar{\alpha}_t}\, \mathbf{X}_0 + \sqrt{1 - \bar{\alpha}_t}\, \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}) \\ \mathbf{X}_t[i] &\leftarrow \mathbf{X}_0[i] \quad \text{for all } i \in \mathcal{K} \end{aligned} \tag{8}$$

where $\mathcal{K}$ denotes the set of anchor frame indices (e.g., initial pose $\mathbf{x}_o$, endpose $\mathbf{x}_{T_2}$, and A*-generated anchors $\mathbf{a}_i$). In short, the diffusion model iteratively denoises the motion sequence, with known anchor poses overwritten at each timestep to guide generation. Intuitively, it learns to synthesize realistic transitions that satisfy both spatial and temporal constraints defined by the anchors. In summary, the architecture automatically decides the number of frames to output, instead of leaving this choice to the user Guo et al. (2022). We leave training details in the supplementary.

Table 1: Gaze classification and object prediction on the GIMO and GTA-IM datasets. Top-K values denote classification accuracy, and lower MSE is better. A dash (‘–’) indicates that the metric cannot be computed for the corresponding method.

Method	**GIMO**								**GTA-IM**							
	Gaze Classification (%)				**Object Prediction (%)**				**Gaze Classification (%)**				**Object Prediction (%)**			
	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Dist. (m)	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Angle (°)
BiFU	44.5	57.7	0.04	0.07	79.8	87.1	100	0.49	22.0	91.5	0.04	0.08	73.4	80.1	100	29.2
SIF3D	49.3	63.6	0.04	0.08	82.6	89.8	100	0.46	21.6	92.3	0.04	0.09	75.1	82.7	100	28.9
DiMoP3D	–	–	–	–	9.09	33.3	42.4	–	–	–	–	–	17.2	38.5	50.1	–
GAP3DS	–	–	–	–	86.8	94.3	100	–	–	–	–	–	75.5	82.2	100	–
Ours (Gaze)	77.0	86.1	69.3	76.8	83.3	93.3	100	0.37	**93.9**	**94.2**	**94.7**	**93.4**	**76.2**	**82.4**	100	**28.7**
Ours (Gaze + Scene)	**86.7**	**87.8**	**91.0**	**89.4**	**93.2**	**96.2**	100	**0.22**	–	–	–	–	–	–	–	–
GT Gaze	100	100	100	100	100	100	100	0.16	100	100	100	100	76.2	80.9	100	28.9
Median Gaze	65.2	72.4	55.1	62.5	81.0	90.5	100	0.42	88.1	89.5	89.9	88.4	74.0	79.3	100	29.1

4 Experiments

Datasets: The **GIMO** dataset Zheng et al. (2022) contains $\approx$ 129K frames of body pose sequences captured using IMUs, eye gaze data recorded via AR headsets, and 3D LiDAR scans of indoor environments. Participants perform a variety of everyday tasks such as walking to grasp objects such as door handles and dumbbells, sitting on chairs, or lying on beds. We sample sequences at 15Hz to obtain more fine-grained gaze data and adopt the official train-val split. The **GTA-IM** dataset Cao et al. (2020) is a synthetic dataset with $\approx$ 1M frames of 3D human poses and semantic scene information, where the person walks around the scene but only interacts by sitting on chairs. The human motions are automatically generated using in-game animations, and gaze is approximated using head orientation in the ground plane. As the person does not directly look at the object of interest, we modify eq. 2 to exclude scene information. Consequently, the primary object in eqs. 3, 4, 5 does not use the gaze intersection at all. We follow the same 15Hz sampling as in GIMO. As interactions in GTA-IM are sparse, we follow prior work Yu et al. (2025) to retrieve only up to 6 seconds preceding the moment of interaction.

Baselines: We compare our approach to the most closely related existing methods: (1) BiFU Zheng et al. (2022) employs a bidirectional fusion module that enables mutual conditioning between gaze and motion features, while incorporating global scene context via a motion-scene transformer. (2) SIF3D Lou et al. (2024a) improves on this by attending over relevant regions of the 3D scene. (3) DiMoP3D Lou et al. (2024b) segments the scene into object instances ranked by past motion, then uses A* search to generate fixed-length motion. (4) GAP3DS Yu et al. (2025) ranks objects using gaze and predicts motion toward the top candidate, using affordance rather than precise gaze to infer interaction points.

Gaze Classification and Object Ranking: We first evaluate the accuracy of gaze estimation through two tasks: (1) gaze classification, where we distinguish between fixations and saccades, and (2) object ranking, where the goal is to identify the object of interest based on the aggregated gaze. For the GIMO dataset, we empirically define a fixation as a gaze point within 0.3 m of the ground-truth object, and label all other cases as saccades. For GTA-IM, we define fixations as gaze velocities below 5 °.

We then aggregate classification or attention scores differently across methods. For our method, which outputs binary classification scores (0 for saccade, 1 for fixation), we compute a weighted sum over gaze vectors using these scores as weights following eq. 2. For attention-based methods such as BiFU and SIF3D, which produce continuous attention weights that sum to 1, we find that selecting the gaze vector corresponding to the maximum attention weight performs better than using a weighted average. DiMoP3D does not utilize gaze, whereas GAP3DS uses raw gaze vectors to compute a point-cloud distance matrix, and thus does not output any interpretable scores for the gaze sequence.

The resulting single gaze vector is then used to evaluate object ranking. For this, we define a simple protocol based on proximity. For GIMO, we use spatial proximity by computing the intersection of the aggregated gaze with the scene and measuring its Euclidean distance to each object's surface. For GTA-IM, we use angular proximity, measuring the angle between the predicted gaze direction and the vector from the head to the object center. Objects are ranked based on these distances or angles, and we report top-N accuracy, where a hit is recorded if the ground-truth object appears within the top-N. In addition, we report the gaze-to-target distance using the appropriate metric: Euclidean distance in metres for GIMO and angular distance in degrees for GTA-IM. This metric captures both model performance and scene clutter. For example, low accuracy but low distance suggests scene clutter, whereas high distance indicates either poor model performance or unreliable gaze data.

Table 1 summarizes the results, with metrics grouped according to the two tasks described above: gaze classification, and object prediction for the Top-1, Top-2, and Top-3 objects, as well as the gaze-to-target Euclidean or angular distances in meters and degrees respectively. The bottom row shows the performance using ground-truth gaze vectors, serving as an upper bound under perfect gaze input. For GIMO, this corresponds to an aggregated gaze point approximately 0.16 m from the target object when using a 0.3 m fixation threshold. In addition, we include a simple Median Gaze baseline, which aggregates the gaze sequence by taking its median vector.

Our method, which explicitly models fixations, outperforms prior approaches. This is because gaze, especially when recorded using wearable eye trackers, is highly informative. During interaction, people often fixate directly on the object they intend to engage with rather than on the surrounding context. This predictable fixation behavior enables our model to more reliably infer the target object. In contrast, prior methods that rely on implicit cross-modal attention mechanisms, such as BiFU and SIF3D, may struggle due to variability in fixation and saccade patterns, and the added challenge of disentangling object geometry. Furthermore, our classification-based approach produces independent probability scores for each gaze, whereas attention-based methods must assign weights that sum to one, which can dilute focus when handling longer input sequences, or lose important cues when sampling at very low fps. This is illustrated in Fig. 2, where our method consistently classifies fixations correctly, while SIF3D exhibits inconsistent responses due to distributed attention, with fixation scores never exceeding 0.2.

Ablation with and without scene information: We compare our model with and without scene information during gaze classification. Both the table and figure highlight its impact: as shown in Fig. 2 on the right, our method reliably detects fixations even when they are spatially dispersed across a large object. Our method continues to show improvements over the SOTA even on the more challenging GTA-IM dataset where gaze merely sweeps along the ground xz plane without any changes in its y component. We also evaluate alternative architectures for our gaze classifier, specifically Bi-LSTM and CNN variants, with results provided in the supplementary material.

Ablation at varying fixation thresholds: We further ablate the fixation-distance threshold used to generate binary fixation labels on the GIMO dataset in Table 2. For each threshold, we train a separate fixation classifier, evaluating values from 0.1 m to 0.5 m. We observe that performance remains stable from 0.2 m onward, which provides the best balance between noise tolerance (e.g., gaze jitter and LiDAR sparsity) and spatial precision. In contrast, the 0.1 m threshold is overly strict: minor gaze deviations, such as the ray slightly missing small objects, or transient attention shifts to nearby distractors such as items on a table, cause many true fixations to be mislabeled as saccades, degrading overall performance. In the same table, the velocity-threshold ablation on GTA-IM exhibits a similar trend: performance is stable across moderate thresholds but degrades at very low and very high angular thresholds. This occurs because overly strict thresholds label almost no fixations, while overly permissive ones label nearly the entire sequence as a fixation, reducing the supervisory signal to noise.

Ablation using only velocity thresholds on the GIMO dataset: We additionally examine the effect of replacing distance-based thresholds on GIMO with velocity-based ones, as done in GTA-IM (Table 3). Gaze-classification accuracy remains relatively stable because the model is learning to reproduce the velocity thresholded-generated labels in the presence of noise, effectively a weak-supervision setup. However, object-prediction performance drops substantially. This is because object prediction depends on how spatially correct the positive fixation labels are. Velocity thresholds do not guarantee that slow gaze corresponds to fixations towards the target object, and short or drifting fixations introduce noise into the aggregated gaze vector. In contrast, using distance-based thresholds to generate the ground truth helps model to learn when meaningful fixations occur within a temporal sequence and produce far more reliable spatial cues for downstream object and pose inference.

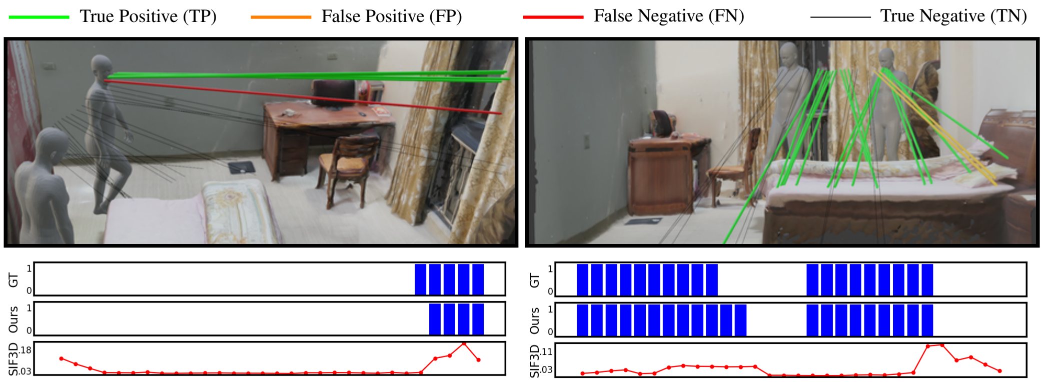


Figure 2: Gaze classification on GIMO. (Green: TP, Orange: FP, Red: FN, Gray: TN). The middle row presents the ground-truth and predicted binary fixation labels, where blue bars denote fixation frames. The bottom row shows SIF3D's probabilistic fixation outputs (red curve). Our method closely matches the binary ground-truth sequence, in contrast to the inconsistent fixation scores produced by SIF3D.

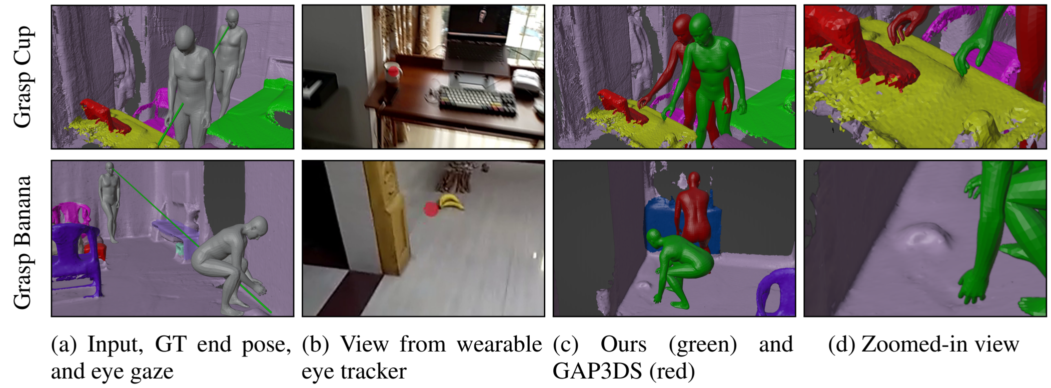
Table 2: Ablation on the fixation- and velocity-distance thresholds. Top-K values denote classification accuracy, and lower MSE is better. We use the best model for the respective datasets i.e., Gaze + Scene for GIMO and Gaze for GTA-IM. The table shows relatively stable performance except for values at the lower and higher ends.

Dist. Thr.	GIMO								Vel. Thr. (°/s)	GTA-IM							
	Gaze Classification (%)				Object Prediction (%)					Gaze Classification (%)				Object Prediction (%)			
	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Dist. (m)		Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Angle (°)
0.10	86.0	80.8	84.2	82.0	86.5	90.4	100	0.24	1	61.0	70.5	73.0	69.8	60.0	65.0	75.3	35.4
0.20	93.0	86.1	88.4	90.1	94.0	96.0	100	0.23	5	93.9	94.2	94.7	93.4	76.2	82.4	100	28.7
0.30	93.2	87.8	91.0	89.4	93.2	96.2	100	0.22	10	95.5	92.0	96.5	92.1	74.5	84.0	100	29.9
0.40	94.0	86.3	92.7	88.1	93.5	95.0	100	0.22	15	92.0	95.0	92.5	94.0	78.0	80.0	100	27.3
0.50	89.8	85.1	88.0	85.9	90.5	94.1	100	0.23	30	96.2	91.3	97.2	91.0	66.0	70.0	100	33.0

Table 3: Ablation using velocity-based thresholds on GIMO. Top-K values denote classification accuracy, and lower MSE is better. Performance is consistently lower than using distance based thresholds in Figure 2 as gaze speeds do not always guarantee the person is looking at the object of interest.

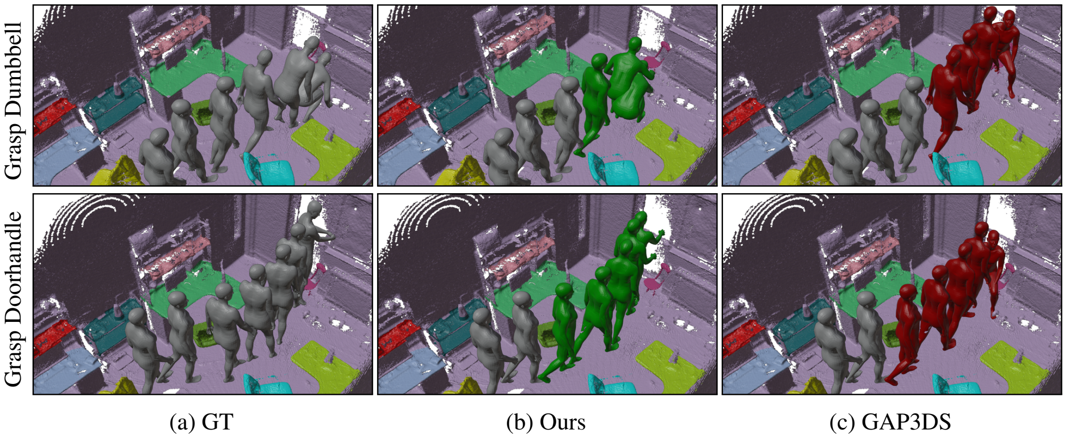
Velocity Threshold	Gaze Classification (%)				Object Prediction			
	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Dist. (m)
1	85.9	86.3	92.1	86.8	77.4	72.7	76.0	0.41
5	94.4	95.1	98.8	94.7	85.5	80.5	100	0.28
10	93.2	96.0	99.5	95.0	86.0	81.0	100	0.27
15	94.1	94.6	98.2	94.0	85.0	80.0	100	0.29
30	92.8	95.3	97.5	93.5	77.0	73.5	83.5	0.35

End-pose Prediction: Fig. 3 illustrates the importance of conditioning end-pose prediction on the aggregated gaze. The left column shows the input pose and ground-truth end pose, along with the aggregated gaze in green. The middle column shows the corresponding RGB view from the wearable eye tracker with the red dot indicating the gaze. The right column compares our predicted end pose (green) against GAP3DS's (red). Our method correctly forecasts interactions with small or merged objects, such as a cup on a table or a banana on the floor, even when these objects are incorrectly merged with their surroundings by the segmentation model. GAP3DS fails in these cases because it uses gaze solely to rank segmented object instances and does not leverage gaze during pose prediction. In contrast, our method integrates gaze as a direct conditioning signal for end-pose prediction. This allows gaze to serve a dual purpose: identifying the target object and precisely localizing the intended interaction region. As a result, our approach is more robust to segmentation errors and scene ambiguity. A limitation (in the supplementary) is a case where noisy gaze, or an



(a) Input, GT end pose, and eye gaze (b) View from wearable eye tracker (c) Ours (green) and GAP3DS (red) (d) Zoomed-in view

Figure 3: End-pose prediction on GIMO. Each row shows a different human–object interaction. Grey poses show input and ground truth; green vectors indicate fixations. Predictions are in green (Ours) and red (GAP3DS), with RGB views for reference. The rightmost column provides zoomed-in views highlighting improvements on small-object interactions.



(a) GT (b) Ours (c) GAP3DS

Figure 4: Motion forecasting on GIMO. GT (gray), ours (green), and GAP3DS (red) are shown for grasping a dumbbell (top) and doorhandle (bottom). Ours succeeds on small objects whereas GAP3DS fails due to segmentation errors.

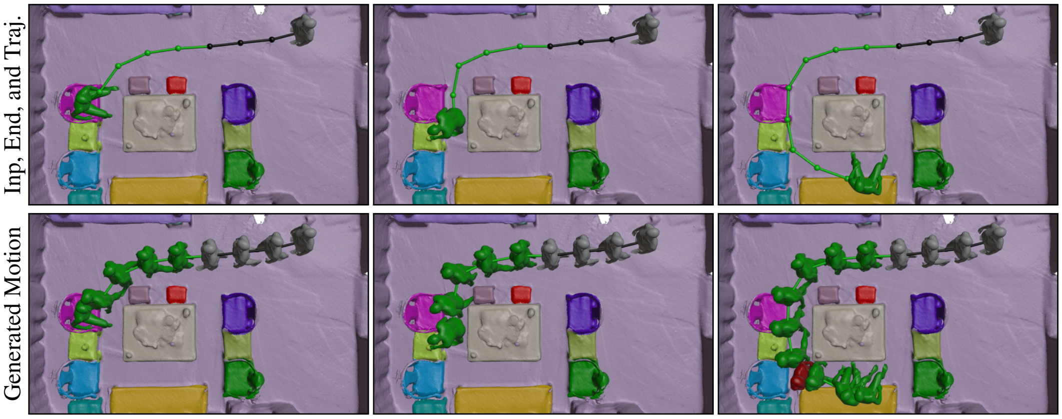
improperly selected input sequence leads to floor interaction instead of the chair. We argue that this reflects plausible stochasticity in human motion, rather than random error.

Motion Forecasting: The quality of end-pose prediction directly impacts downstream motion forecasting, as shown in Fig. 4. We display intermediate poses every 0.5 m, along with the end pose. In the dumbbell example, our method correctly predicts the person walking toward the dumbbell and bending down to pick it up, despite the dumbbell being mislabeled as background. In contrast, GAP3DS predicts a sitting motion toward a nearby chair, since its retrieval module only considers segmented objects, and the dumbbell is excluded due to segmentation failure. A similar issue occurs in the doorbell example, where GAP3DS entirely misses the interaction. Although BiFU and SIF3D are more robust in such cases as they do not rely on segmentation, they underperform compared to ours because they do not explicitly predict end-poses, an important geometric constraint that guides the motion more accurately toward the intended target.

Table 4 reports results using Mean Per Joint Position Error (MPJPE) across time horizons, end-pose error, and Average Cumulated Penetration Depth (ACPD) Xu et al. (2023) all in millimeters (mm). With ground-truth gaze, our method shows greater improvements in end-pose errors, translating to consistent overall gains. Short-horizon (e.g. 0.5s) MPJPE remains difficult to improve since very little movement occurs, making errors naturally small. ACPD remains comparable to prior work,

Table 4: MPJPE from 0.5s to 5.0s, End Pose Error, and ACPD, all reported in millimeters (mm). Lower values indicate better performance. Our method consistently outperforms prior work for longer durations.

	GIMO						GTA IM					
	0.5s	1.0s	2.0s	5.0s	End	ACPD	0.5s	1.0s	2.0s	5.0s	End	ACPD
BiFU	78	99	118	133	131	3.45	101.2	109.1	121.8	141.5	150.9	2.07
SIF3D	78	97	114	127	130	3.23	100.4	107.3	119.5	139.1	141.2	2.21
DiMoP3D	**75**	93	103	125	104	1.04	**98.5**	104.7	109.8	131.1	94.9	0.79
GAP3DS	76	**92**	105	123	101	2.16	98.7	104.5	108.5	130.1	96.2	1.47
Ours (Gaze)	77	94	109	125	112	**1.00**	99.6	105.6	112.0	134.2	99.8	**0.73**
Ours (Gaze + Scene)	76	93	**100**	**120**	**80**	1.02	99.2	**104.3**	**106.9**	**127.5**	**92.2**	0.75
Ours (GT Gaze)	76	93	100	118	75	—	99.2	103.9	106.0	125.6	91.2	—



(a) Rank 1 (chair) (b) Rank 2 (table) (c) Rank j (sofa).

Figure 5: Multi-target forecasting on GIMO. Top: input and trajectory (gray), A* anchors, and predicted end pose (green). Bottom: GT (gray), ours (green), and DiMoP3D (red, third column), where fixed-length forecasting fails.

as obstacle avoidance is only addressed during path planning. Improvements on GTA-IM are lower due to the lack of fine objects and the focus on sitting interactions.

Multi-Target Motion Forecasting: To showcase our method's flexibility, we demonstrate its ability to predict motion lengths and forecast toward multiple secondary targets, each with varying distances, as shown in Fig. 5. The figure displays the coarse trajectory together with the generated poses, enabling accurate modeling of motion sequences with variable lengths. In contrast, DiMoP3D generates fixed-length sequences that do not adapt to the actual distance between the human and the target. As shown in the third column, its final frame (in red) falls short of the intended sitting pose, which our method successfully reaches. Furthermore, when DiMoP3D's fixed sequence exceeds the required length, it often results in frozen or repetitive poses after reaching the target, further highlighting the limitations of fixed-length generation. These advantages also translate to improved quantitative performance across generative-relevant metrics such as accuracy, FID, human-object vertex distance and time prediction errors. We further provide additional results in the supplementary, which highlight the importance of our time prediction module: methods without it often produce motions that appear unnaturally fast or slow when the number of frames is not accurately set in advance. All corresponding quantitative results for cases without ground-truth frame counts or poses are also reported in the supplementary.

5 Conclusion

We present a fixation-driven, time-aware framework for 3D human motion forecasting in indoor scenes. By explicitly training the model to distinguish fixations from saccades, we improve gaze reliability, which in turn enhances object localization, end-pose prediction, and motion forecasting. Additionally, we introduce a duration prediction module that enables our diffusion-based model to generate variable-length motion sequences tailored to each interaction. These contributions result in more accurate forecasts across diverse targets and object configurations. Future work may explore the use of head orientation as a proxy for gaze in real-world scenarios where eye tracking is unreliable, as well as methods for more personalized fixation modelling.

REFERENCES

Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pp. 719–728. IEEE, 2019.

Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pp. 387–404. Springer, 2020.

Setareh Cohan, Guy Tevet, Daniele Reda, Xue Bin Peng, and Michiel van de Panne. Flexible motion in-betweening with diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–9, 2024.

Enric Corona, Albert Pumarola, Guillem Alenya, and Francesc Moreno-Noguer. Context-aware human motion prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6992–7001, 2020.

Tom Foulsham. Eye movements and their functions in everyday tasks. *Eye*, 29(2):196–199, 2015.

Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 2021–2029, 2020.

Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *European Conference on Computer Vision*, pp. 580–597. Springer, 2022.

Bo Han, Hao Peng, Minjing Dong, Yi Ren, Yixuan Shen, and Chang Xu. Amd: Autoregressive motion diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 2022–2030, 2024.

Peter Hart, Nils Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi: 10.1109/tssc.1968.300136. URL `https://doi.org/10.1109/tssc.1968.300136`.

Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion inbetweening. *ACM Transactions on Graphics (TOG)*, 39(4):60–1, 2020.

Zhiming Hu, Syn Schmitt, Daniel Häufle, and Andreas Bulling. Gazemotion: Gaze-guided human motion forecasting. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13017–13022. IEEE, 2024.

Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13401–13412, 2021.

Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 619–635, 2018.

Miao Liu, Siyu Tang, Yin Li, and James M Rehg. Forecasting human-object interaction: joint prediction of motor attention and actions in first person video. In *European Conference on Computer Vision*, pp. 704–721. Springer, 2020.

Zhenyu Lou, Qiongjie Cui, Haofan Wang, Xu Tang, and Hong Zhou. Multimodal sense-informed forecasting of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2144–2154, 2024a.

Zhenyu Lou, Qiongjie Cui, Tuo Wang, Zhenbo Song, Luoming Zhang, Cheng Cheng, Haofan Wang, Xu Tang, Huaxia Li, and Hong Zhou. Harmonizing stochasticity and determinism: Scene-responsive diverse human motion prediction. *Advances in Neural Information Processing Systems*, 37:39784–39811, 2024b.

Kedi Lyu, Haipeng Chen, Zhenguang Liu, Yifang Yin, Yukang Lin, and Yingying Jiao. Hvis: A human-like vision and inference system for human motion prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5928–5936, 2025.

Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *CVPR*, 2017.

Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10985–10995, 2021.

Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4332–4341, 2019.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

Haziq Razali and Yiannis Demiris. Using eye gaze to forecast human pose in everyday pick and place actions. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 8497–8503, 2022. doi: 10.1109/ICRA46639.2022.9812079.

Guy Tevet et al. Human motion diffusion model. In *CVPR*, 2022.

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.

Zhihao Wang, Yulin Zhou, Ningyu Zhang, Xiaosong Yang, Jun Xiao, and Zhao Wang. Existence is chaos: Enhancing 3d human motion prediction with uncertainty consideration. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 5841–5849, 2024.

Zhenyu Xie, Yang Wu, Xuehao Gao, Zhongqian Sun, Wei Yang, and Xiaodan Liang. Towards detailed text-to-motion synthesis via basic-to-advanced hierarchical diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 6252–6260, 2024.

Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14928–14940, 2023.

Haodong Yan, Zhiming Hu, Syn Schmitt, and Andreas Bulling. Gazemodiff: Gaze-guided diffusion model for stochastic human motion prediction. *arXiv preprint arXiv:2312.12090*, 2023.

Ting Yu, Yi Lin, Jun Yu, Zhenyu Lou, and Qiongjie Cui. Vision-guided action: Enhancing 3d human motion prediction with gaze-informed affordance in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2025. Poster.

Yang Zheng, Yanchao Yang, Kaichun Mo, Jiaman Li, Tao Yu, Yebin Liu, C Karen Liu, and Leonidas J Guibas. Gimo: Gaze-informed human motion prediction in context. In *European Conference on Computer Vision*, pp. 676–694. Springer, 2022.

A APPENDIX

Training: We train our architecture with the following objective:

$$\sum_{s,i,t} \underbrace{\lambda_1 f_t \log \hat{f}_t + (1-f_t)\log(1-\hat{f}_t)}_{\text{fixation}} + \underbrace{\lambda_2 \|\hat{\tau}_i - \tau_i\|_1}_{\text{duration}}$$
$$+ \underbrace{\lambda_3 \|\hat{x}_{T_2} - x_{T_2}\|_2^2 + \lambda_4 \text{KL}}_{\text{end pose}} + \underbrace{\lambda_5 \|\epsilon_\theta(X_t, s) - \epsilon\|_2^2}_{\text{diffusion}}$$

where s denotes the diffusion timestep. The PointNet++ instance segmentation module is trained using the loss function from Qi et al. (2017) and is omitted here to keep the formulation focused on our novel objectives. Explicitly, we trained every neural based architecture in Figure 1 independently for ease of tuning and modularity. We train the model for 1000 epochs using the ADAM optimizer, with a batch size of 32 on an RTX A5000 GPU, requiring approximately 12 hours in total. This modular strategy also makes training feasible on a single 24 GB GPU. While no individual component requires this memory, training all components jointly in an end-to-end manner would exceed the available VRAM. Inference does not face this limitation, and training jointly end-to-end may offer additional performance gains.

B ADDITIONAL QUANTITATIVE RESULTS

We compare our approach to DiMoP3D and GAP3DS for multi-target forecasting as both BiFU and SIF3D are limited to producing a single motion sequence. DiMoP3D Lou et al. (2024b) segments the scene into object instances ranked by past motion, then uses A* search to generate fixed-length motion. GAP3DS Yu et al. (2025) similarly segments the scene and ranks objects using gaze cues, predicting motion toward the top-ranked instance using object affordances to infer interaction points. However, both methods do not explicitly predict the number of frames required to reach the target. We modify both architectures to output the frame count using an LSTM conditioned on the prior motion and the predicted keyframe. We use the following metrics for evaluating generative human motion:

- **Final Pose Accuracy (Acc, %, ↑)**: We evaluate the predicted final pose using a pre-trained MLP action classifier trained to distinguish between reaching, sitting, and lying actions. This measures whether the generated motion ends in a plausible and semantically correct posture. Higher is better.
- **Fréchet Inception Distance (FID, unitless, ↓)** Guo et al. (2020): We extract features from real and generated motion sequences using a pre-trained transformer-based motion classifier. The FID is computed between the feature distributions, quantifying the realism and diversity of the generated motions compared to the ground truth. Lower is better.
- **Human-Object Vertex Distance (HO Dist., mm, ↓)**: We measure the proximity of the final human pose to the intended target object. For reach actions, we compute the minimum distance from the left/right hand vertices to the object. For sit/lie actions, we use the full body. Lower values indicate more accurate spatial grounding.
- **Time Error (Time Err., s, ↓)**: Measures the discrepancy between the predicted and ground-truth number of frames required to reach the object, indicating whether the model correctly anticipates motion duration. Lower is better.
- **Average Cumulative Penetration Distance (ACPD, mm, ↓)**: Measures the average distance that the predicted human mesh penetrates into the scene geometry across all frames. Lower values indicate fewer and smaller collisions.

From the table, GAP3DS and our method achieve similar final pose accuracy, as both directly predict the target pose and the metric reflects semantic correctness over spatial precision. DiMoP3D performs slightly worse, likely due to its fixed time constraint (e.g., 5 seconds in their paper), which may prevent reaching the target. For FID, DiMoP3D remains competitive thanks to its A* planning, which yields plausible, temporally consistent trajectories even if incomplete, unlike GAP3DS which does not truncate outputs. As a result, when the predicted motion duration is underestimated, the

	GIMO					GTA IM				
	Acc (↑)	FID (↓)	HO Dist (↓)	Time Err. (↓)	ACPD (↓)	Acc (↑)	FID (↓)	HO Dist (↓)	Time Err. (↓)	ACPD (↓)
DiMoP3D	**99.7**	3.55	212.1	1.52	1.13	100	**2.11**	181.4	1.35	1.03
GAP3DS	95.1	5.12	91.4	1.47	2.32	100	4.41	17.3	1.29	2.01
Ours	99.6	**3.51**	91.1	**0.87**	**1.11**	100	**2.11**	17.5	**0.78**	**0.96**
Ours (w/o Time)	**99.7**	5.21	**90.9**	1.42	1.14	100	4.31	**17.1**	1.31	1.01
Avg Vel	—	—	—	2.01	—	—	—	—	1.82	—
Real	100	1.41	79.5	—	—	100	1.01	11.9	—	—

Table 5: Final pose classification accuracy (Acc), Frechet Inception Distance (FID), human-object distance (HO Dist), time prediction error (Time Err.), and Average Cumulative Penetration Distance (ACPD). Units: Acc (%), FID (unitless), HO Dist (mm), Time Err. (s), ACPD (mm). ↑ indicates higher is better, ↓ lower is better. Our method performs visibly better overall.

Method	**GIMO**								**GTA-IM (without scene information)**							
	Gaze Classification (%)				**Object Prediction (%)**				**Gaze Classification (%)**				**Object Prediction (%)**			
	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Dist. (m)	Acc	Prec	Rec	F1	Top-1	Top-2	Top-3	Angle (°)
LSTM	85.5	85.0	86.0	85.8	89.0	93.5	100	0.30	87.8	85.2	86.4	85.9	74.0	79.5	100	29.1
Bi-LSTM	**87.2**	86.5	88.2	87.3	90.2	94.5	100	0.27	89.6	86.8	88.0	87.2	74.8	80.5	100	28.9
CNN	84.0	86.2	88.0	87.0	90.0	94.2	100	0.28	89.4	86.6	87.8	87.0	75.0	80.8	100	28.8
Transformer	86.7	**87.8**	**91.0**	**89.4**	**93.2**	**96.2**	100	**0.22**	**93.9**	**94.2**	**94.7**	**93.4**	**76.2**	**82.4**	100	**28.7**

Table 6: Gaze classification and object prediction on the GIMO and GTA-IM datasets. Top-K values denote classification accuracy, and lower distance/angle is better. All architectures are evaluated using scene information for GIMO, while for GTA-IM the evaluation is performed without scene information.

steps between frames may become large and unnatural. This reduces motion quality and leads to a higher FID compared to DiMoP3D or ours.

For human-object vertex distance, our method and GAP3DS perform similarly, as both explicitly predict the interaction point. DiMoP3D performs worse on this metric because motion truncation may leave the final pose farther from the intended object. Our time prediction approach, which estimates motion duration by dividing the trajectory into anchor pose segments, results in lower errors.

For ACPD, our method performs comparably to DiMoP3D since both use path planning to avoid obstacles. However, collisions can still occur because the planned trajectory only considers the root path and not the full body geometry. We include Real as a reference computed using ground-truth final poses and durations, representing the upper bound for all metrics. Avg Vel serves as a non-learned baseline for time estimation, where the number of frames is inferred from the average velocity across the dataset.

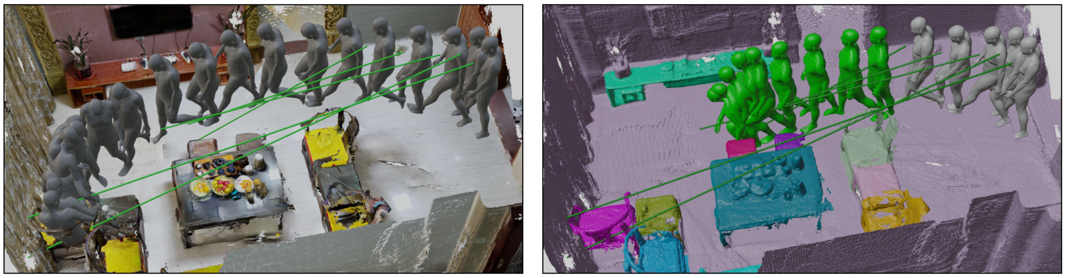
Our method achieves strong overall performance, consistently ranking among the top across all metrics. Importantly, it does not underperform on any evaluation criterion. For comparison, GAP3DS yields the highest FID scores, suggesting limited realism in its predicted motion, while DiMoP3D suffers from large human-object distances. Additionally, all baselines exhibit substantially higher time prediction errors, with differences exceeding 50% in some cases.

Gaze Classification Architectures: We compare different architectures for gaze classification, including an LSTM, Bi-LSTM, CNN, and Transformer, as shown in Fig. 6. The Bi-LSTM and CNN outperform the vanilla LSTM, likely because they can aggregate temporal information more effectively, with the Bi-LSTM capturing dependencies in both directions. The Transformer achieves the best performance overall, benefiting from its optimized architecture and ability to attend to information across the entire sequence simultaneously. All architectures are evaluated using scene information for GIMO, while for GTA-IM the evaluation is performed without scene information (see Table 1).

C Additional Qualitative Results

We present additional visualizations in Figures 6 to 9, as well as in the supplemental video. Figure 6 illustrates a failure case where an improperly selected input sequence causes the predicted motion to target the floor instead of the chair. This occurs because our method selects the region where the aggregated gaze vectors intersect the point cloud, which in this case lies on the floor. However, we argue that this reflects plausible stochasticity in human motion, rather than random error since both outcomes correspond to valid behavioral interpretations. For example, the user may intend to pick up an object on the ground, and the earlier gaze toward another object may simply reflect transient distraction. We believe that addressing such ambiguity would require incorporating additional contextual cues, such as verbal or textual instructions (e.g., "clean the table"), to more clearly indicate the intended target.

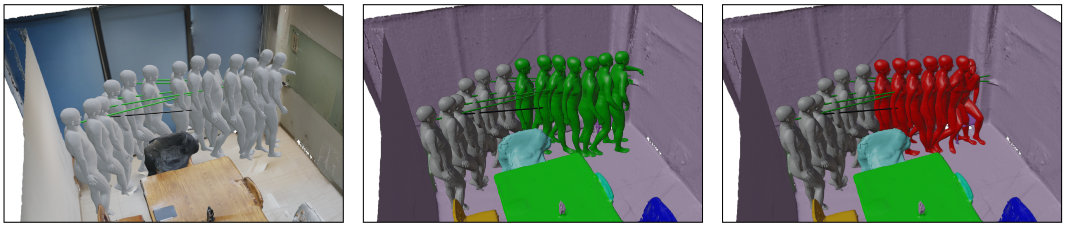
Figure 7 demonstrates the robustness of our method to segmentation failures: while GAP3DS incorrectly predicts the person interacting with a chair, our method correctly predicts interaction near the whiteboard. Figures 8 and 9 highlight the importance of accurate time prediction. In both cases, the generated motions from GAP3DS and DiMoP3D appear less natural compared to ours, as shown in the figures and corresponding videos.



(a) Ground Truth Motion and Eye Gaze

(b) Ours

Figure 6: Fail Case. Our method predicts the human motion interacting with the floor instead of the chair.



(a) GT Motion and Eye Gaze

(b) Ours

(c) GAP3DS

Figure 7: Our method can predict the human motion interacting with the whiteboard even in scenes with poor segmentation.

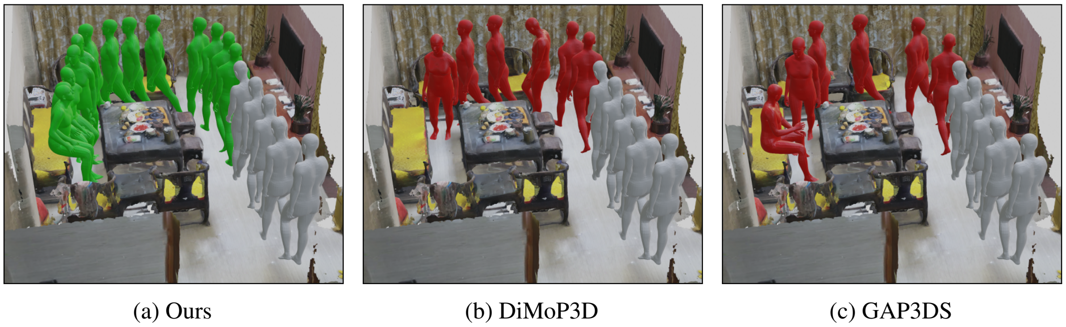

(a) Ours (b) DiMoP3D (c) GAP3DS

Figure 8: Our method continues to predict the human motion moving at an appropriate speed compared to GAP3DS, while completing the sequence compared to DiMoP3D.

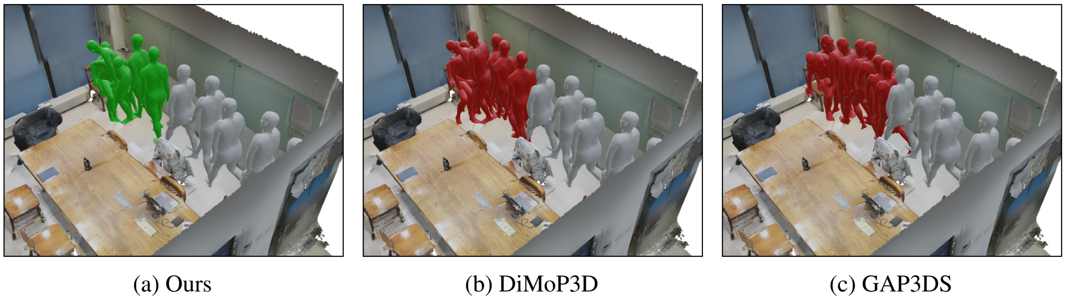

(a) Ours (b) DiMoP3D (c) GAP3DS

Figure 9: Our method continues to predict the human motion moving at an appropriate speed compared to GAP3DS, while completing the sequence compared to DiMoP3D.