Gated Temporal Diffusion for Stochastic Long-Term Dense Anticipation

Olga Zatsarynna[1]*, Emad Bahrami[1]*, Yazan Abu Farha[2], Gianpiero Francesca[3], and Juergen Gall[1,4]

[1] University of Bonn, Germany
[2] Birzeit University, Palestine
[3] Toyota Motor Europe, Belgium
[4] Lamarr Institute for Machine Learning and Artificial Intelligence, Germany
*Equal contribution

Abstract. Long-term action anticipation has become an important task for many applications such as autonomous driving and human-robot interaction. Unlike short-term anticipation, predicting more actions into the future imposes a real challenge with the increasing uncertainty in longer horizons. While there has been a significant progress in predicting more actions into the future, most of the proposed methods address the task in a deterministic setup and ignore the underlying uncertainty. In this paper, we propose a novel Gated Temporal Diffusion (GTD) network that models the uncertainty of both the observation and the future predictions. As generator, we introduce a Gated Anticipation Network (GTAN) to model both observed and unobserved frames of a video in a mutual representation. On the one hand, using a mutual representation for past and future allows us to jointly model ambiguities in the observation and future, while on the other hand GTAN can by design treat the observed and unobserved parts differently and steer the information flow between them. Our model achieves state-of-the-art results on the Breakfast, Assembly101 and 50Salads datasets in both stochastic and deterministic settings.

1 Introduction

In this work, we address the task of long-term dense action anticipation. Given a video as observation, the goal is to predict future actions and their durations where the forecasting horizons can span from several seconds to several minutes into the future, which makes it a challenging problem. Yet, solving it is crucial for many real-world applications, such as autonomous driving or human-robot interaction. Over the last few years, the task has gained increased attention and there has been a steady progress [1, 2, 26, 32, 46, 52], but most works address this task deterministically, which means that only one prediction is made for a single observation. The task of forecasting future actions, however, is highly uncertain by nature, especially for longer anticipation horizons, since the same observation can have multiple plausible continuations. Despite its importance, dealing

[2] Corresponding author: zatsarynna@iai.uni-bonn.de
[3] Code: `https://github.com/olga-zats/GTDA`

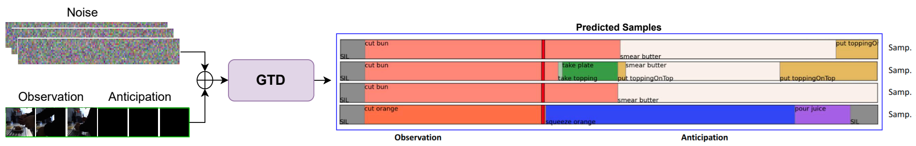

Fig. 1: The proposed Gated Temporal Diffusion (GTD) model generates multiple future long-term predictions of actions from a single partially observed video. In contrast to previous works, it models the uncertainty of both the observation and the future. In this example, the light conditions make it difficult to distinguish if a bun or an orange is cut. This ambiguity is reflected in the predicted samples where the uncertainty of the past impacts the predicted future.

with uncertainty for long-term dense action anticipation has so far received little attention. Farha *et al.* [23] proposed to address this task in a stochastic manner. The approach generates multiple predictions in an autoregressive way by predicting the probabilities of the next action and its duration, and then sampling from the predicted probabilities. Alternatively, a GAN-based probabilistic encoder-decoder network has been proposed to generate multiple predictions [72]. Both approaches, however, assume that the action labels of the observed frames are already given, either pre-estimated [23] or taken from the ground-truth actions [23, 72]. In this way, the uncertainty of the observation is not taken into account. The observation, however, can also be ambiguous due to occlusions or difficult light conditions as shown in Fig. 1. We, therefore, argue that stochastic action anticipation needs to consider not only the ambiguity of the future but also of the observation.

In this work, we propose a diffusion-based model [14,28,49,53,55,56] - **G**ated **T**emporal **D**iffusion (GTD) - that models the uncertainty of both the observation and the future predictions. In particular, we make use of diffusion since it inherently models uncertainty by generating multiple predictions. The key aspect of our GTD, however, is that we explicitly model the uncertainty of observed and future actions jointly by predicting them simultaneously with a single diffusion process. To this end, we use a joint sequence representation illustrated in Fig. 1, which we construct by extending the observed video frames with zero padding in place of future frames. We then cast this sequence as the per-frame conditioning of the past and future action variables, whose conditional joint probabilities will be learned during training. While grouping both observed and unobserved action variables allows us to model them using a single diffusion process, these two parts are nevertheless intrinsically distinct. To take this difference into account, we propose a novel generator - **G**a**T**ed **A**nticipation **N**etwork (GTAN) - that can differentiate between past and future frames in a distinct manner and steer the information flow between them in a data-driven way.

We evaluate our proposed model on the Breakfast [35], Assembly101 [51] and 50Salads [57] datasets. We show that our approach achieves state-of-the-art results for stochastic long-term dense action anticipation. Additionally, despite our focus on stochastic anticipation, we also demonstrate that our proposed

GTAN network achieves state-of-the-art results in the deterministic setting. In summary, we make the following contributions:

– We propose the first diffusion model - Gated Temporal Diffusion - for stochastic long-term action anticipation that jointly models observed and future actions.
– We propose a novel generator backbone - Gated Anticipation Network - which steers the information flow between observed and future frames.
– We show that our model achieves state-of-the-art results in stochastic and deterministic settings on three datasets: Breakfast, Assembly101 and 50Salads.

2 Related Work

Action Anticipation in Videos. The task of action anticipation in videos is to forecast future actions given video observations from the current and past moments of time. Following the introduction of anticipation benchmarks on the recent [18, 27, 51] and less recent [35, 36] datasets, this task has been gaining increasing attention [74]. In general, existing literature distinguishes between short-term and long-term anticipation approaches, based on the length of the anticipation horizon. Short-term anticipation methods [24, 25, 40, 43, 52, 63, 68–70, 73, 75] forecast only a single action that takes place in the near future. Long-term action anticipation methods, which are the focus of our work, have a longer forecasting horizon and predict multiple future actions several minutes into the future. Among long-term anticipation approaches, one can further pin down several established task formulations. In the first line of work [2], Farha *et al.* proposed to anticipate action classes densely for a subset of frames from the predefined future time interval. This formulation, addressed in [1, 2, 26, 32, 46, 52] and referred to here as *deterministic long-term dense anticipation*, implies the necessity to predict not only classes of future actions but also their duration. In their work, Farha *et al.* [2] introduced two models with different modes of predictions based on CNN and RNN networks. Ke *et al.* [32] proposed a TCN-based model with time conditioning to enable direct action prediction at a predefined moment of time. In the later works, several encoder-decoder architectures were proposed by Farha *et al.* [1], Sener *et al.* [52], Gong *et al.* [26] and Nawhal *et al.* [46] based on GRUs [6], Non-Local blocks [62] and Transformer layers [60]. In the second line of work [23], Farha *et al.* extended the previous formulation into the probabilistic domain, so that the uncertainty of future anticipation could be taken into account. In this formulation, called here *stochastic long-term dense anticipation*, actions still need to be anticipated densely, however, several future sequences are allowed to be predicted for a single observation. In their work [23], the authors proposed a probabilistic RNN network that made autoregressive predictions based on the samples drawn from the predicted action class distributions. Zhao *et al.* [72] later proposed a GAN-based probabilistic encoder-decoder network. Lastly, in the anticipation frameworks from [27, 45], which we refer to as *transcript long-term anticipation*, the estimation of duration for future actions is discarded. This setting has been addressed in [3, 19, 27, 42, 45, 46].

Diffusion. Diffusion models [28, 49, 53, 55, 56] are a class of deep probabilistic generative models that recover the data sample from Gaussian noise via a gradual denoising process. In a following work, denoising diffusion implicit models (DDIMs) [54] were introduced to speed up the diffusion model sampling. Later, [12,17] extended the continuous diffusion models to generate discrete data. Diffusion models have shown outstanding results in various generation tasks such as image synthesis [21,50], video generation [10,29], speech processing [48,58,65], natural language processing [5, 30] and motion generation [59, 71]. Furthermore, diffusion models have achieved great success in computer vision perception tasks such as segmentation [8, 11, 16], object detection [15], temporal action segmentation [38] and detection [44]. The stochastic nature of diffusion models has been leveraged for motion anticipation [9, 59, 64] and procedure planning [61].

3 Stochastic Long-Term Dense Anticipation

Previously proposed stochastic long-term anticipation approaches [23,72] assume that the observed video segments share the same format as the future predictions, namely action labels. This assumption, however, overlooks the ambiguity inherent in certain video observations, such as insufficient context due to limited observation duration, challenging lighting conditions, occlusions, and other factors (see Fig. 1 and Fig. 4). In such cases, the estimated or pre-defined action labels fail to convey the uncertainty associated with observed frames.

To overcome this limitation, we aim to jointly model the uncertainty of both future and observed actions. To this end, we propose an approach based on diffusion, known for its suitability for modelling uncertainty [14]. Standard diffusion models, however, cannot be directly applied to the task of long-term action anticipation. Thus, we introduce a novel model for stochastic long-term anticipation, termed Gated Temporal Diffusion for Anticipation (GTD), which we describe in detail in Sec. 4. Although we focus on stochastic anticipation, we also show how our method can be used in the deterministic case in Sec. 4.4. Before discussing our proposed approach, we formally define the task of stochastic long-term anticipation.

Following the popular protocol introduced by [2], the observed part and the duration of the future prediction are defined by percentages α and β of the entire video. More precisely, given $\alpha|V|$ observed frames of a video with $|V|$ frames, the goal is to predict action labels for the future $\beta|V|$ frames. Accordingly, $N_o = \alpha|V|$ is the number of observed frames, $N_a = \beta|V|$ is the number of frames whose action classes we want to anticipate, and $N = N_o + N_a$ is the total number of frames that are considered. Since β can go up to 0.5, the prediction duration can be very long. While deterministic approaches make only a single most likely future prediction, stochastic approaches consider the uncertainty of the future modelling and generate multiple ($M > 1$) future samples.

Formally, the task of stochastic long-term anticipation can be formulated as learning to draw samples from the underlying joint probability of per-frame future actions conditioned on the observed video frames:

$$\hat{Y}^{N_o+1:N} \sim p_\theta(Y^{N_o+1}, \dots, Y^N | F) \tag{1}$$

$$F = (\phi(x^1), \dots, \phi(x^{N_o})) \in \mathbb{R}^{N_o \times D}, \tag{2}$$

where x^k is the k^{th} input frame, ϕ is the feature extractor network, Y^i is the action variable corresponding to i^{th} frame and $Y^{i:j}$ denotes a sequence of variables for frame i to j.

Since in our work, we aim to additionally model the uncertainty present in the video observations, we instead learn to sample $\hat{Y} = \hat{Y}^{1:N}$ from the conditional joint distribution of both future and observed actions.

4 Gated Temporal Diffusion for Anticipation

To model uncertainty and perform stochastic action anticipation, we formulate our network as a diffusion model as illustrated in Fig. 1. While the standard diffusion model, which we describe in Sec. 4.1, serves as a foundation, it cannot be directly applied to the task of long-term action anticipation. Hence, we introduce a novel model, called Gated Temporal Diffusion (GTD), which jointly models the uncertainty of both observed and unobserved events while preserving the inherent difference between the two. We discuss our proposed approach in Sec. 4.2.

4.1 Diffusion Model

Diffusion models learn to map noise samples $Y_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to the samples from the data distribution $Y_0 \sim q(Y)$ in an iterative manner using a reverse Markov chain process $p_\theta(Y_{0:T})$ with learnable transition parameters θ:

$$p_\theta(Y_{0:T}) = p_\theta(Y_T) \prod_{t=1}^{T} p_\theta(Y_{t-1}|Y_t), \quad p_\theta(Y_{t-1}|Y_t) = \mathcal{N}(Y_{t-1}; \mu_\theta(Y_t, t), \mathbf{\Sigma}(t)). \tag{3}$$

To learn these parameters, a forward Markov process is defined. It specifies the transitions in the inverse direction by adding Gaussian noise to the data according to a fixed variance schedule $\beta_1, \dots, \beta_T$:

$$q(Y_{1:T}|Y_0) = \prod_{t=1}^{T} q(Y_t|Y_{t-1}), \quad q(Y_t|Y_{t-1}) = \mathcal{N}(Y_t; \sqrt{1-\beta_t} Y_{t-1}, \beta_t \mathbf{I}). \tag{4}$$

Training. Optimization of diffusion models is performed using the variational lower bound on the negative log-likelihood of the data samples where some properties of the forward process are harnessed. Given the Gaussian nature of the forward transition probabilities, one can use the reparametrization trick [33] to draw samples directly from Y_0 by corrupting it following the schedule γ_t:

$$Y_t = \sqrt{\gamma_t} Y_0 + \sqrt{(1-\gamma_t)} \epsilon_t, \tag{5}$$

where $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\gamma_t = \prod_{k=1}^{t}(1-\beta_k)$. For training, Y_t is obtained using the forward process (5) at step $t \sim \mathcal{U}(1, T)$ and a denoising generator network

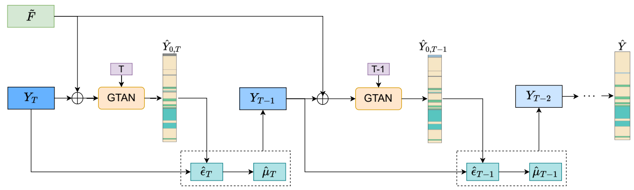

Fig. 2: We formulate stochastic action anticipation as a diffusion process where the initial input consists of Gaussian noise, Y_T, and zero-padded features, $\tilde{F}$. Given the inputs, the GTAN generator predicts the denoised action labels, $\hat{Y}_{0,T}$. From step T-1 to 0, the GTAN generator uses self-conditioning by taking the previous denoised action labels as additional input. The noise $\hat{\epsilon}_t$ and mean $\hat{\mu}_t$ terms for steps T-1 to 0 are computed using equations (7) and (8). $\oplus$ indicates channel-wise concatenation.

$G_\theta(Y_t, t) = \hat{Y}_{0,t}$ is trained to reverse the noise and predict the reconstruction $\hat{Y}_{0,t}$ of Y_0 by minimizing the l_2 reconstruction error between them:

$$L_{diff} = \mathbb{E}_{t\sim\mathcal{U}(1,T),\epsilon_t\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\|G_\theta(Y_t,t) - Y_0\|^2. \tag{6}$$

The key contribution of a diffusion model is the design of the generator network $G_\theta(Y_t, t)$.

Inference. Once trained, sampling from the diffusion model requires following a sequence of denoising steps. In the DDPM [28] sampling procedure, inference follows T denoising steps. Starting at step $t = T$, a random sample is drawn from $Y_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and fed to the generator G_θ that predicts $\hat{Y}_{0,T}$. Assuming that we have already sampled Y_t and reconstructed $\hat{Y}_{0,t}$ at step t, we can generate the next sample Y_{t-1} by first approximating ϵ_t by:

$$\hat{\epsilon}_t = \frac{1}{\sqrt{1-\gamma_t}}(Y_t - \sqrt{\gamma_t}\hat{Y}_{0,t}). \tag{7}$$

Since the reverse transition probabilities $q(Y_{t-1}|Y_t, Y_0)$ become tractable when conditioned on Y_0 and can be expressed as Gaussians $\mathcal{N}(Y_{t-1}; \tilde{\mu}_t, \tilde{\beta}_t\mathbf{I})$, we can estimate the parameters of this Gaussian by:

$$\hat{\mu}_t = \sqrt{\gamma_{t-1}}\hat{Y}_{0,t} + \sqrt{1-\gamma_{t-1}-\tilde{\beta}_t}\hat{\epsilon}_t, \quad \tilde{\beta}_t = \frac{1-\gamma_{t-1}}{1-\gamma_t}\beta_t \tag{8}$$

and sample $Y_{t-1} \sim \mathcal{N}(Y_{t-1}; \hat{\mu}_t, \tilde{\beta}_t\mathbf{I})$. The steps continue until $t = 1$, after which $\hat{Y}_{0,1}$ is taken as the final generated sample. For an alternative DDIM [54] sampling, variances of the transition probabilities are set to zero during inference, *i.e.*, $\tilde{\beta}_t = 0$, making the denoising process deterministic for a particular noise sample Y_T. This way, Y_{t-1} is equal to the mean value $Y_{t-1} = \hat{\mu}_t$. The DDIM sampling scheme performs better than DDPM if less denoising steps are used during inference compared to training [54].

4.2 Gated Temporal Diffusion for Anticipation

In contrast to previous works, we aim to model uncertainty both in the past observation and the future. We thus extend the formulation described in Sec. 3

and generate multiple samples not only for the future, but also multiple interpretations of the past as shown in Fig. 4. Since the uncertainty of the future depends on the uncertainty in the observation, we treat them as a unified sequence $\hat{Y} = \hat{Y}^{1:N}$ and model them with a shared diffusion model.

While the diffusion model described in Sec. 4.1, generates multiple data samples by sampling $Y_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ repeatedly, it is not directly applicable to our problem for several reasons. Specifically, it operates in the continuous domain and it does not incorporate any conditioning information during the generation process. Moreover, the generator network G_θ treats all variables uniformly, disregarding the distinction between observed and future action variables in the input sequence. To address these limitations, we propose a Gated Temporal Diffusion model (GTD). As our key contribution, we introduce a GTAN generator network, described in Sec. 4.3. This network employs gated temporal convolutions and, on one hand, models observed and unobserved data jointly, while, on the other hand, steers the information flow between past and future entries with the learnable gates, thereby controlling their fusion. Apart from that, we also propose discretization and conditioning schemes, elaborated below.

To model discrete action categories using continuous state diffusion, we represent action labels as one-hot encoded vectors $Y_0 \in \mathbb{R}^{N\times C}$ and regard them as "analog bits" [17] that can be directly modelled by continuous state diffusion models. In this way, training remains unchanged and the inference process also remains the same, except that we map generated samples back to the discrete domain by applying the argmax operation over the class dimension.

To condition the generation on the observed frames, we make use of frame-wise feature vectors F (2) and adapt them to act as the per-frame condition for both observed and future action variables. More specifically, we expand it by incorporating zero-padding to compensate for the absent features of the future unobserved frames:

$$\hat{Y} \sim p_\theta(Y|\tilde{F}), \quad \tilde{F} = (\phi(x^1), \dots, \phi(x^{N_o}), \mathbf{0}, \dots, \mathbf{0}) \in \mathbb{R}^{N\times D}. \tag{9}$$

In this way, the resulting vector $\tilde{F}$ can be used to condition the per-frame diffusion generation process. To this end, we channel-wise concatenate the padded observed frame features $\tilde{F} \in \mathbb{R}^{N\times D}$ to the current sample $Y_t \in \mathbb{R}^{N\times C}$ at each step t. Furthermore, we employ self-conditioning [17] by using the previous estimate $\hat{Y}_{0,t+1} \in \mathbb{R}^{N\times C}$ as additional input for the generator G_θ:

$$G_\theta(Y_t, \hat{Y}_{0,t+1}, \tilde{F}, t) = \hat{Y}_{0,t} \in \mathbb{R}^{N\times C}, \tag{10}$$

where during training $\hat{Y}_{0,t+1}$ is randomly set to zero with probability p, which is then equivalent to training G_θ without self-conditioning. While Y_t, $\hat{Y}_{0,t+1}$, and $\tilde{F}$ are concatenated and used as input to the generator G_θ, shown in Fig. 2, at each diffusion step t, we encode the step t as sinusoidal positional embedding. In the next section, we describe our proposed GTAN generator network in detail.

4.3 Gated Anticipation Generator

To model temporal dependencies in the input, we use temporal convolutional layers for the generator network G_θ. As the input sequence to our network con-

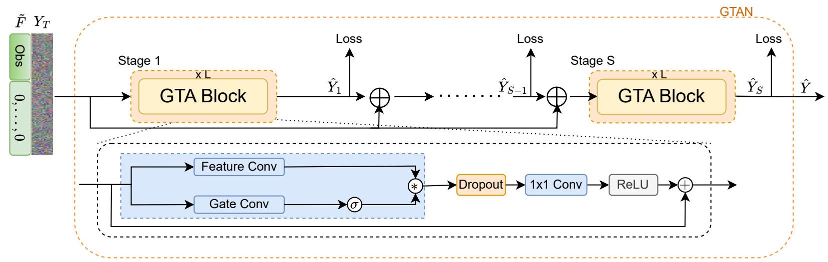

Fig. 3: The GTAN takes as input a joint representation sequence for observed and future frames. Each stage consists of GTA blocks. The dilated gated convolutions deactivate features at certain frames.

sists of two distinct parts (observed and unobserved), directly applying the network with classical temporal convolution layers as in other tasks [7, 22, 66] is sub-optimal. This is because the distinction between observed and future parts would not be possible, leading to all values being treated equally. While gated convolutions [4, 20, 47, 67] have been proposed for different purposes, we adopt this concept such that the generator network adaptively decides how much mixing between the past and future occurs at different levels of the network.

Motivated by this, we propose a Gated Temporal Anticipation Network (GTAN). It comprises S stages that produce output vectors of action probabilities over C classes. Each stage consists of L residual GTA blocks as shown in Fig. 3. Each GTA block includes two dilated temporal convolutional layers: one for feature processing *(feature convolution)* and another for gating the features *(gate convolution)*. Formally, given a feature vector as input, both layers are applied to it separately with the same dilation rate, kernel size, and number of channels. The layers are then combined by the element-wise product between the output of the feature convolution and the sigmoid-activated output of the gate convolution. Given the input feature $\tilde{H}_{s,l-1}$, the output $\bar{H}_{s,l}$ of the gated convolution at layer l and stage s is computed as follows:

$$\bar{H}_{s,l} = \sigma(W_g * \tilde{H}_{s,l-1} + b_g) \odot (W_f * \tilde{H}_{s,l-1} + b_f), \tag{11}$$

where σ is the sigmoid function, $*$ denotes the convolution operator, and $\odot$ denotes the element-wise product. W_g and W_f are weights of the convolutional filters and b_g and b_f are biases for gate and feature convolutions, respectively. The output $\bar{H}_{s,l}$ is then passed through a dropout layer and then to a 1×1 convolutional layer followed by the ReLU non-linearity. Finally, the output of the layer l is computed by applying a residual connection after the ReLU. At each layer l, the dilation factor for the gate and feature convolution is set to 2^l, *i.e.*, it increases by a factor of two with each layer.

In our ablations studies, we show the benefit of gating and demonstrate the importance of learning temporal gates in a data-driven way, as opposed to using manual masks [39] or using channel-wise gating [31], *i.e.* without the temporal component. We also show that our proposed gated generator is superior to the previously proposed gated temporal convolutional network [4].

4.4 Training and Inference

For training our proposed Gated Temporal Diffusion model, we sample step $t \sim \mathcal{U}(1,T)$ and Y_t (5) for a given observation $\tilde{F}$ (9) with ground-truth Y_0. We then apply our proposed GTAN generator $G_\theta(Y_t, \hat{Y}_{S,0,t+1}, \tilde{F}, t)$ that produces a set of stage-wise predictions $\{\hat{Y}_{s,0,t}\}_{s=1}^{S}$. The self-conditioning with prediction $\hat{Y}_{S,0,t+1}$ at step $t+1$ is only included if $t < T$. We train our network with the l_2 reconstruction loss accumulated over all S stages:

$$L_{stoch} = \mathbb{E}_{t\sim\mathcal{U}(1,T),\epsilon_t\sim\mathcal{N}(\mathbf{0},\mathbf{I})} \sum_{s=1}^{S} \|\hat{Y}_{s,0,t} - Y_0\|^2. \tag{12}$$

During inference, we apply the DDIM [54] sampling scheme and use a subset of D denoising steps to get our final predictions. As the final output of our model, we take the reconstruction made at the denoising step $t = 1$ of the last GTAN stage S, *i.e.*, $\hat{Y} = \hat{Y}_{S,0,1}$. To generate multiple action sequences for a given observation, we run the denoising process starting from M different noise samples $\{Y_{T,m}\}_{m=1}^{M}$, which are then reconstructed into distinct predictions $\{\hat{Y}_m\}_{m=1}^{M}$, where $\{\hat{Y}_m^{N_o+1:N}\}_{m=1}^{M}$ are the future action predictions.

Deterministic Anticipation. Although we focus on stochastic long-term anticipation, we also report results for the deterministic anticipation. In this case, we use the proposed GTAN without the diffusion process. Given $\tilde{F}$ as the input sequence, our model produces intermediate $\{\hat{Y}_s\}_{s=1}^{S-1}$ and final $\hat{Y} = \hat{Y}_S$ predictions. Using these predictions, we train our network using the standard cross-entropy loss applied to all stages and frames:

$$L_{determ} = -\sum_{s=1}^{S}\sum_{n=1}^{N} Y^n \log \hat{Y}_s^n, \tag{13}$$

where $Y^n \in \mathbb{R}^C$ is the one-hot encoded ground-truth action at frame n. During inference, we only consider predictions for the future frames.

5 Experiments

5.1 Datasets and Evaluation Metrics

We evaluate our proposed network on three challenging datasets: Breakfast, Assembly101 and 50Salads.

Breakfast [35] contains breakfast preparation videos. It contains 1712 videos of actors preparing 10 breakfast-related dishes. Each video is densely annotated with actions from 48 classes. On average, videos are 2.3 minutes long and contain 6 action segments. The longest video is 10.8 minutes long, so the duration of anticipation is up to 5.4 minutes. For evaluation, we use the standard 4 splits for cross-validation and report the average result.

Assembly101 [51] is a large-scale dataset of toy vehicle assembly and disassembly. It contains 4321 videos, which are densely annotated with 100K coarse action segments from 202 coarse action classes. The duration of anticipation is

Table 1: Comparison to the state of the art for stochastic anticipation on Breakfast, Assembly101 and 50Salads. * Indicates that we trained UAAA on Assembly101.

Dataset	Metric	Method	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
			0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
Breakfast	Mean MoC	Tri-gram [23]	15.4	13.7	12.9	11.9	19.3	16.6	15.8	13.9
		UAAA [23]	15.7	14.0	13.3	13.0	19.1	17.2	17.4	15.0
		Ours	**24.0**	**22.0**	**21.4**	**20.6**	**29.1**	**26.8**	**25.3**	**24.2**
	Top-1 MoC	UAAA [23]	28.9	28.4	27.6	28.0	32.4	31.6	32.8	30.8
		Ours	**51.2**	**47.3**	**45.6**	**45.0**	**54.0**	**50.4**	**49.6**	**47.8**
Assembly101	Mean MoC	Tri-gram	2.8	2.2	1.9	1.5	3.5	2.7	2.3	1.8
		UAAA* [23]	2.7	2.1	1.9	1.7	2.4	2.1	1.9	1.7
		Ours	**6.4**	**4.4**	**3.5**	**2.8**	**5.9**	**4.2**	**3.5**	**2.9**
	Top-1 MoC	UAAA* [23]	6.9	5.9	5.6	5.1	5.9	5.5	5.2	4.9
		Ours	**18.0**	**12.8**	**9.9**	**7.7**	**16.0**	**11.9**	**10.2**	**7.7**
50Salads	Mean MoC	Tri-gram [23]	21.4	16.4	13.3	9.4	24.6	15.6	11.7	8.6
		UAAA [23]	23.6	19.5	**18.0**	**12.8**	28.0	18.0	**14.8**	**12.1**
		Ours	**28.3**	**22.1**	17.8	11.7	**29.9**	**18.5**	14.2	10.6
	Top-1 MoC	UAAA [23]	53.5	43.0	40.5	**33.7**	56.4	42.8	35.8	30.2
		Ours	**69.6**	**55.8**	**45.2**	28.1	**66.2**	**44.9**	**39.2**	**31.0**

up to 12.5 minutes in videos lasting 25 minutes. The dataset is divided into train, validation, and test splits. Since the test split is not publicly available, we train our model on the train split and report our results on the validation set.

50Salads [57] contains videos of salad preparations. It comprises 50 videos annotated with dense segments from 17 fine-grained actions. The mean length of videos is 6.4 minutes, while the longest video is 10.1 minutes long, so the duration of anticipation is up to 5.1 minutes. For evaluation, we use the predefined 5 splits for cross-validation and report the average result.

For a fair comparison with existing methods on these datasets, we used previously extracted visual features. On the Breakfast and 50Salads datasets, we use the I3D [13] features provided by [2], while for Assembly101 we utilized TSM [37] features provided by [51]. Further implementation details of our model for these datasets are provided in the supp. material.

Evaluation Protocol and Metrics. We evaluate our method using the α and β from the protocol defined in [2]. Specifically, we test our network for $\alpha \in \{0.2, 0.3\}$ and $\beta \in \{0.1, 0.2, 0.3, 0.5\}$, where α and β denote the percentages of frames of a video that are used as observation and future prediction, respectively. We evaluate our approach in the stochastic and deterministic settings. For the deterministic setting, we report mean over classes (MoC) accuracy as in [2,26,32]. In the stochastic setting, we generate multiple predictions for the same observed frames. As proposed in [23], we report two metrics: mean and top-1 MoC across $M = 25$ predictions, called Mean MoC and Top-1 MoC, respectively.

5.2 Stochastic Anticipation

Comparison with State of the Art. We first compare our proposed stochastic approach with the state of the art for stochastic anticipation on the Breakfast, Assembly101 and 50Salads datasets. In Sec. 5.4, we compare our approach to deterministic approaches. The comparison of our diffusion-based model with UAAA [23] and tri-gram baseline [23] is presented in Tab. 1. UAAA [23] and the tri-gram baseline are the only available probabilistic models with a comparable evaluation protocol since [72] only uses ground-truth action labels as their observations. [23] is a two-step approach that first predicts action labels for the observed frames and then forecasts the future actions. To compare our method to [23] on the Assembly101 dataset, we used MS-TCN [22] for the first step. We trained a MS-TCN network for each value of α using full supervision, but only the observed frames as training data, *i.e.*, the first 20% or 30%, respectively. We then trained UAAA [23]. Note that two-step approaches [23, 72] do not model any uncertainty in the observation. As shown in Tab. 1, our method outperforms [23] with a large margin on Breakfast and Assembly101 at both Mean and Top-1 MoC accuracy with improvements across all observation and anticipation ratios. On the 50Salads dataset, our approach outperforms [23] at Top-1 MoC, while the Mean MoC accuracy is on par.

In Fig. 4, we present predicted samples from our model. In the first example, the action sequence involves the high-level activity 'Scrambled Eggs'. However, the observed part of the sequence contains only the action 'Butter Pan', shared with another high-level class, 'Fry Eggs'. Thus, based solely on this segment, distinguishing the underlying activity is challenging. Our network predicts sequences belonging to either 'Scrambled Eggs' or 'Fry Eggs', demonstrating ambiguity in order, length, and presence/absence of actions within categories. In the second example, the ground-truth activity is 'Cereals'. Poor lighting conditions make recognizing actions in the observed segment difficult and ambiguous. Consequently, our model classifies observed actions differently across samples, leading to consistent yet different predictions. By addressing uncertainty in observations, our model produces correct predictions despite the poor quality of the observed segment. We provide more qualitative results in the supp. material.

5.3 Ablation Study

Number of Inference Steps and Stages. In Tab. 2, we explored the impact of varying the number of denoising steps D in our diffusion model while keeping the number of GTAN stages fixed. Increasing the number of steps from 10 to 50 led to improved accuracy, but further increasing it to 100 did not yield additional benefits. Fig. 6 illustrates how predictions evolve with different denoising steps: noise decreases significantly after 10 steps, with further refinement observed

Table 2: Ablation on the num. of diffusion inference steps and stages in GTAN on Breakfast. Numbers show Mean MoC accuracy.

Num. stages	Num. steps	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
		0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
5	1	19.6	17.7	17.3	15.9	24.9	22.6	22.5	20.0
5	10	23.0	20.7	20.3	19.5	28.0	25.7	24.7	23.8
5	50	<u>24.0</u>	**22.0**	**21.4**	**20.6**	<u>29.1</u>	**26.8**	**25.3**	**24.2**
5	100	**24.2**	21.8	21.3	20.5	29.1	26.7	25.0	24.0
1	50	23.7	21.9	21.1	20.2	29.2	26.7	25.3	23.9
3	50	23.4	21.6	21.0	20.4	**29.6**	26.8	25.2	**24.4**
5	50	**24.0**	**22.0**	**21.4**	**20.6**	<u>29.1</u>	**26.8**	**25.5**	<u>24.2</u>
5	50	**24.0**	**22.0**	**21.4**	**20.6**	29.1	26.8	25.3	**24.2**
1	250	23.9	21.8	21.0	20.0	**29.6**	**27.0**	**25.5**	23.9

Table 3: Ablation on GTAN architecture on Breakfast. Numbers show Mean MoC accuracy.

Method	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
	0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
Ours	**24.0**	**22.0**	**21.4**	**20.6**	**29.1**	**26.8**	**25.3**	**24.2**
Ours w/o GC	23.0	21.1	20.6	19.8	27.6	25.7	24.3	23.5
Ours w/o Dil. GC	22.9	21.1	20.6	20.2	28.6	26.0	24.7	23.9
Aslan *et al.* [4]	18.8	17.3	16.7	15.7	20.7	18.9	18.8	16.7
Part. Conv. [39]	23.2	21.5	20.9	20.1	27.9	25.6	24.4	23.6
SE [31]	22.4	21.0	20.4	19.9	28.0	25.7	24.5	23.6

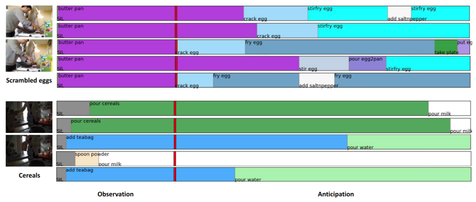

Fig. 4: Qualitative results of our proposed GTD for stochastic long-term action anticipation on Breakfast. Best viewed zoomed in.

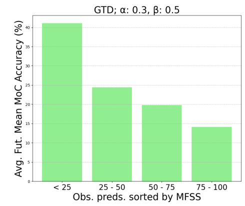

Fig. 5: Mean MoC of GTD for sequences sorted by MFSS diversity for the obs. part on Breakfast.

with 50 steps. Next, with D fixed at 50, we evaluated the influence of different numbers of stages. Smaller networks (1 and 3 stages) exhibited lower accuracy with 50 denoising steps, but with 250 steps, the single-stage GTAN's performance approached that of 5 stages. For all other experiments, we use 5 stages and 50 denoising steps.

Gated Convolution. In Tab. 3, we assess the impact of gated convolution on our GTAN generator. Removing the gate convolution branch (see Fig. 3) from the GTA block and leaving only the feature convolution (Ours w/o GC) leads to lower performance, highlighting the necessity of the gating mechanism. Similarly, removing the dilation factor from the gate convolution branch (Ours w/o Dil. GC) also decreases performance, as expected. Substituting our GTAN architecture with the gated temporal convolutional network proposed by Aslan *et al*. [4] results in a significant decline in performance, indicating its unsuitability for dense anticipation.

We also explore alternatives to gated convolutions, including manual mask adaptation and channel-wise feature reweighing, by replacing gated convolutions with partial convolution [39] and squeeze-and-excitation [31] blocks, respectively. Although manual mask adaptation outperforms the model without gating, it falls short compared to our proposed approach, highlighting the advantage of learning to perform gating in a data-driven manner. Employing the squeeze-and-excitation block instead of gated convolutions yields better performance compared to previous alternatives, but it remains inferior to our proposed method. This emphasizes the importance of the temporal component of gated convolutions. We provide further ablations of the GTAN design and a qualitative example of learned gates in the supp. material.

Modelling Observation Uncertainty. We conduct an experiment to investigate the impact of modelling observation ambiguity on the performance of

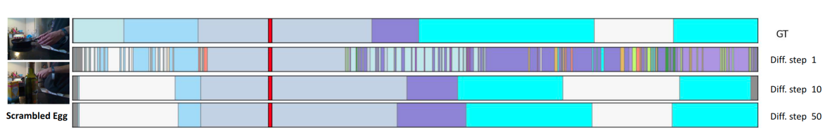

Fig. 6: Qualitative results of our proposed GTD with different numbers of inference diffusion steps. Best viewed zoomed in.

Table 4: Ablation on modelling observation uncertainty (o.u.) on Breakfast.

Metric	Method	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
		0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
Mean MoC	Ours	**24.0**	**22.0**	**21.4**	**20.6**	29.1	26.8	**25.3**	**24.2**
	Ours w/o o.u.	23.7	21.4	20.5	19.9	**29.8**	**27.1**	**25.3**	24.1
Top-1 MoC	Ours	**51.2**	**47.3**	**45.6**	**45.0**	**54.0**	**50.4**	**49.6**	**47.8**
	Ours w/o o.u.	42.8	40.7	38.8	38.4	48.4	44.2	44.9	43.5
MFSS	Ours	**41.5**	**44.3**	**45.6**	**48.4**	**33.7**	**36.6**	**38.0**	**41.6**
	Ours w/o o.u.	31.2	34.7	36.3	39.4	26.1	29.4	30.8	34.3

Table 5: Ablation on the diffusion loss type on Breakfast.

Metric	Loss	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
		0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
Mean MoC	MSE	24.0	22.0	21.4	20.6	29.1	26.8	25.3	24.2
	CE	**26.2**	**23.3**	**23.9**	**22.6**	**32.1**	**29.2**	**27.5**	**26.3**
Top-1 MoC	MSE	**51.2**	**47.3**	**45.6**	**45.0**	**54.0**	**50.4**	**49.6**	**47.8**
	CE	46.9	42.2	41.7	40.7	48.8	46.5	45.3	44.0
MFSS	MSE	**41.5**	**44.3**	**45.6**	**48.4**	**33.7**	**36.6**	**38.0**	**41.6**
	CE	29.0	31.8	33.5	35.0	24.0	26.9	28.2	30.0

our model. To this end, we utilize an MS-TCN [22] to predict labels of the observed part, which serves as the condition vector for GTD instead of the frame features F. Also, we compute the loss function L_{stoch} only for the future frames. The results of this experiment are shown in Tab. 4. While the Mean MoC of both methods remains comparable, the Top-1 MoC of the model without uncertainty modelling drops significantly, indicating its limited ability to generate diverse predictions. To directly measure the prediction diversity, we additionally introduce a new metric - Mean Framewise Sample Similarity (MFSS). MFSS calculates diversity as the mean normalized pairwise sample distance averaged over the videos in the split:

$$MFSS = \frac{1}{Z} \sum_{z=1}^{Z} \left(\frac{2}{M(M-1)} \sum_{1 \leq i < j \leq M} 100 \left(1 - \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}(\hat{Y}^n_{z,i} = \hat{Y}^n_{z,j}) \right) \right), \tag{14}$$

where Z is the total number of videos. As evident from the results, omitting the observation uncertainty modelling leads to a reduction in the diversity of predictions. In Fig. 5, we demonstrate the correlation between Mean MoC anticipation accuracy and uncertainty of the GTD model for the observation. To this end, we sorted the videos into four groups based on the MFSS of the predictions on the observed parts of the videos and calculated the average Mean MoC for anticipation for each group. For sequences with high MFSS, GTD is uncertain about the observed actions. The plot reveals an inverse correlation between future Mean MoC and observed MFSS since future actions are harder to predict for videos with higher observation uncertainty. This analysis underscores the significance of modelling ambiguity in observed frames for future action prediction.

Diffusion Loss Type. We employ Mean Squared Error (MSE) as the loss function for our diffusion approach, following prior work [17, 28]. Additionally, we evaluate the impact of using the Cross Entropy (CE) loss and present the results in Tab. 5. Notably, the model trained with MSE loss achieves the highest Top-1 MoC accuracy and MFSS, while the model trained with CE loss attains the best Mean MoC. These results of the CE-trained network can be attributed to additional Softmax normalization required by this loss. The Softmax restricts the handling of uncertainty in early diffusion steps, emphasizing the most likely action class prematurely. Directly measuring diversity using the MFSS metric confirms these observations, with the MSE-trained model exhibiting higher diversity than the CE-trained model. We used the MSE loss in our experiments.

Table 6: Comparison with state-of-the-art methods for deterministic anticipation on the Breakfast dataset. * Indicates retrained results.

Dataset	Method	β ($\alpha = 0.2$)				β ($\alpha = 0.3$)			
		0.1	*0.2*	*0.3*	*0.5*	*0.1*	*0.2*	*0.3*	*0.5*
Breakfast	RNN [2]	18.1	17.2	15.9	15.8	21.6	20.0	19.7	19.2
	CNN [2]	17.9	16.3	15.4	14.5	22.4	20.1	19.7	18.8
	UAAA (mode) [23]	16.7	15.4	14.5	14.2	20.7	18.3	18.4	16.9
	Time Cond. [32]	18.4	17.2	16.4	15.8	22.8	20.4	19.6	19.8
	TempAgg [52]	24.2	21.1	20.0	18.1	30.4	26.3	23.8	21.2
	Cycle Cons. [1]	25.9	23.4	22.4	21.5	29.7	27.4	25.6	25.2
	FUTR [26]	27.7	24.6	22.8	22.1	32.3	29.9	27.5	25.9
	Ours	**28.8**	**26.3**	**25.8**	**26.0**	**35.5**	**32.9**	**30.5**	**29.6**
Assembly101	UAAA (mode)* [23]	2.7	2.1	1.8	1.6	2.4	2.1	1.9	1.7
	FUTR* [26]	7.5	5.5	4.7	4.1	7.8	6.0	5.2	4.0
	Ours	**9.0**	**6.8**	**6.6**	**5.5**	**8.4**	**6.8**	**6.0**	**5.0**

5.4 Deterministic Anticipation

Comparison with State of the Art. The majority of dense long-term anticipation methods [1,2,26,32,52] has been trained and evaluated in the deterministic setting and we compare with these methods in Tab. 6. We do not compare with Anticipatr [46] since it uses a different protocol [74]. We also compare with UAAA [23] in the deterministic setting (mode). On Assembly 101, we compare with UAAA (mode) and FUTR [26] using the publicly available code for training and evaluation. We provide results for the 50Salads dataset, qualitative comparisons and ablation studies for the deterministic GTAN in the supp. material. On both Breakfast and Assembly101, our method outperforms all methods that use the same evaluation protocols. Compared to the previously best-performing method FUTR [26], our method shows a substantial improvement in particular on Breakfast for long-term prediction ($\beta = 0.5$) where MoC is increased by $+3.9$ and $+3.7$ for $\alpha = 0.2$ and $\alpha = 0.3$, respectively.

6 Conclusion

We have proposed a Gated Temporal Diffusion network to address the task of stochastic long-term dense action anticipation. As the backbone for our diffusion model, we introduced a Gated Anticipation Network (GTAN) that allows for mutual modelling of the actions in the observed and future frames. In this way, the uncertainty is not only modelled for the future but also for the observed part. We demonstrated that the approach generates different predictions for the observed frames in case of ambiguities due to poor light conditions and that these ambiguities impact the future predictions. Furthermore, we demonstrated that GTAN can be applied to deterministic long-term dense action anticipation as well. In our experiments, we showed that our model achieves state-of-the-art results on three datasets in deterministic and stochastic settings. A limitation of our proposed model is its current efficiency. For example, the average time of generating a single prediction on the Breakfast dataset, which has an average anticipation horizon of 1.15 minutes, is 3.8 seconds. While this is sufficient for mid-term action planning, i.e. range of minutes, a further reduction of inference time is needed. This can be achieved by using techniques to accelerate inference of diffusion models like distillation [34] or DeepCache [41].

Acknowledgements

The work has been supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) GA 1927/4-2 (FOR 2535 Anticipating Human Behavior), by the Federal Ministry of Education and Research (BMBF) under grant no. 01IS22094E WEST-AI, the project iBehave (receiving funding from the programme "Netzwerke 2021", an initiative of the Ministry of Culture and Science of the State of Northrhine Westphalia), and the ERC Consolidator Grant FORHUE (101044724). The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time through the John von Neumann Institute for Computing (NIC) on the GCS Supercomputer JUWELS at Jülich Supercomputing Centre (JSC). The sole responsibility for the content of this publication lies with the authors.

References

1. Abu Farha, Y., Ke, Q., Schiele, B., Gall, J.: Long-term anticipation of activities with cycle consistency. In: DAGM German Conference on Pattern Recognition (GCPR) (2020)
2. Abu Farha, Y., Richard, A., Gall, J.: When will you do what?-Anticipating temporal occurrences of activities. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
3. Ashutosh, K., Girdhar, R., Torresani, L., Grauman, K.: Hiervl: Learning hierarchical video-language embeddings. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
4. Aslan, F., Kozat, S.: Handling irregularly sampled signals with gated temporal convolutional networks. Signal, Image and Video Processing (2022)
5. Austin, J., Johnson, D.D., Ho, J., Tarlow, D., van den Berg, R.: Structured denoising diffusion models in discrete state-spaces. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
6. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. ArXiv (2014)
7. Bahrami, E., Francesca, G., Gall, J.: How much temporal long-term context is needed for action segmentation? In: IEEE International Conference on Computer Vision (ICCV) (2023)
8. Baranchuk, D., Voynov, A., Rubachev, I., Khrulkov, V., Babenko, A.: Label-efficient semantic segmentation with diffusion models. In: International Conference on Learning Representations (ICLR) (2022)
9. Barquero, G., Escalera, S., Palmero, C.: Belfusion: Latent diffusion for behavior-driven human motion prediction. In: IEEE International Conference on Computer Vision (ICCV) (2023)
10. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
11. Brempong, E.A., Kornblith, S., Chen, T., Parmar, N., Minderer, M., Norouzi, M.: Denoising pretraining for semantic segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

12. Campbell, A., Benton, J., Bortoli, V.D., Rainforth, T., Deligiannidis, G., Doucet, A.: A continuous time framework for discrete denoising models. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)
13. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
14. Chan, M.A., Molina, M.J., Metzler, C.A.: Hyper-diffusion: Estimating epistemic and aleatoric uncertainty with a single model. arXiv preprint arXiv:2402.03478 (2024)
15. Chen, S., Sun, P., Song, Y., Luo, P.: Diffusiondet: Diffusion model for object detection. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
16. Chen, T., Li, L., Saxena, S., Hinton, G., Fleet, D.J.: A generalist framework for panoptic segmentation of images and videos. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
17. Chen, T., Zhang, R., Hinton, G.: Analog bits: Generating discrete data using diffusion models with self-conditioning. In: International Conference on Learning Representations (ICLR) (2023)
18. Damen, D., Doughty, H., Farinella, G.M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., Wray, M.: The epic-kitchens dataset: Collection, challenges and baselines. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2021)
19. Das, S., Ryoo, M.S.: Video + clip baseline for ego4d long-term action anticipation. ArXiv (2022)
20. Dauphin, Y., Fan, F., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. In: International Conference on Machine Learning (ICML) (2016)
21. Dhariwal, P., Nichol, A.: Diffusion models beat GANs on image synthesis. Advances in Neural Information Processing Systems (NeurIPS) (2021)
22. Farha, Y., Gall, J.: Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
23. Farha, Y., Gall, J.: Uncertainty-aware anticipation of activities. In: IEEE International Conference on Computer Vision Workshop (ICCVW) (2019)
24. Furnari, A., Farinella, G.M.: Rolling-unrolling lstms for action anticipation from first-person video. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020)
25. Girdhar, R., Grauman, K.: Anticipative Video Transformer. In: IEEE International Conference on Computer Vision (ICCV) (2021)
26. Gong, D., Lee, J., Kim, M., Ha, S., Cho, M.: Future transformer for long-term action anticipation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
27. Grauman, K., Westbury, A., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
28. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems (NeurIPS) (2020)
29. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)

30. Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., Welling, M.: Argmax flows and multinomial diffusion: Learning categorical distributions. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
31. Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
32. Ke, Q., Fritz, M., Schiele, B.: Time-conditioned action anticipation in one shot. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
33. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. International Conference on Learning Representations (ICLR) (2014)
34. Kohler, J., Pumarola, A., Schönfeld, E., Sanakoyeu, A., Sumbaly, R., Vajda, P., Thabet, A.: Imagine flash: Accelerating emu diffusion models with backward distillation. arXiv preprint arXiv:2405.05224 (2024)
35. Kuehne, H., Arslan, A., Serre, T.: The language of actions: Recovering the syntax and semantics of goal-directed human activities. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
36. Li, Y., Liu, M., Rehg, J.M.: In the eye of the beholder: Gaze and actions in first person video. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020)
37. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: IEEE International Conference on Computer Vision (ICCV) (2019)
38. Liu, D., Li, Q., Dinh, A.D., Jiang, T., Shah, M., Xu, C.: Diffusion action segmentation. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
39. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: European Conference on Computer Vision (ECCV) (2018)
40. Liu, M., Tang, S., Li, Y., Rehg, J.M.: Forecasting human-object interaction: joint prediction of motor attention and actions in first person video. In: European Conference on Computer Vision (ECCV) (2020)
41. Ma, X., Fang, G., Wang, X.: Deepcache: Accelerating diffusion models for free. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
42. Mascaró, E., Ahn, H., Lee, D.: Intention-conditioned long-term human egocentric action anticipation. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2023)
43. Miech, A., Laptev, I., Sivic, J., Wang, H., Torresani, L., Tran, D.: Leveraging the present to anticipate the future in videos. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW) (2019)
44. Nag, S., Zhu, X., Deng, J., Song, Y.Z., Xiang, T.: Difftad: Temporal action detection with proposal denoising diffusion. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
45. Nagarajan, T., Li, Y., Feichtenhofer, C., Grauman, K.: Ego-topo: Environment affordances from egocentric video. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
46. Nawhal, M., Jyothi, A.A., Mori, G.: Rethinking learning approaches for long-term action anticipation. In: European Conference on Computer Vision (ECCV) (2022)
47. Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders. Advances in Neural Information Processing Systems (NeurIPS) (2016)
48. Popov, V., Vovk, I., Gogoryan, V., Sadekova, T., Kudinov, M.: Grad-tts: A diffusion probabilistic model for text-to-speech. In: International Conference on Machine Learning (ICML) (2021)

49. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. ArXiv preprint (2022)
50. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
51. Sener, F., Chatterjee, D., Shelepov, D., He, K., Singhania, D., Wang, R., Yao, A.: Assembly101: A large-scale multi-view video dataset for understanding procedural activities. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
52. Sener, F., Singhania, D., Yao, A.: Temporal aggregate representations for long-range video understanding. In: European Conference on Computer Vision (ECCV) (2020)
53. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International Conference on Machine Learning (ICML) (2015)
54. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: International Conference on Learning Representations (ICLR) (2021)
55. Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. Advances in Neural Information Processing Systems (NeurIPS) (2019)
56. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: International Conference on Learning Representations (ICLR) (2021)
57. Stein, S., McKenna, S.J.: Combining embedded accelerometers with computer vision for recognizing food preparation activities. ACM international joint conference on Pervasive and ubiquitous computing (2013)
58. Tae, J., Kim, H., Kim, T.: EdiTTS: Score-based Editing for Controllable Text-to-Speech. In: Proc. Interspeech 2022 (2022)
59. Tanke, J., Zhang, L., Zhao, A., Tang, C., Cai, Y., Wang, L., Wu, P.C., Gall, J., Keskin, C.: Social diffusion: Long-term multiple human motion anticipation. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
60. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NeurIPS) (2017)
61. Wang, H., Wu, Y., Guo, S., Wang, L.: Pdpp: Projected diffusion for procedure planning in instructional videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
62. Wang, X., Girshick, R.B., Gupta, A.K., He, K.: Non-local neural networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
63. Wu, Y., Zhu, L., Wang, X., Yang, Y., Wu, F.: Learning to anticipate egocentric actions by imagination. In: IEEE Transactions on Image Processing (TIP) (2020)
64. Xu, S., Wang, Y.X., Gui, L.Y.: Stochastic multi-person 3d motion forecasting. In: International Conference on Learning Representations (ICLR) (2023)
65. Yang, D., Yu, J., Wang, H., Wang, W., Weng, C., Zou, Y., Yu, D.: Diffsound: Discrete diffusion model for text-to-sound generation. IEEE/ACM Transactions on Audio, Speech, and Language Processing (2023)
66. Yi, F., Wen, H., Jiang, T.: Asformer: Transformer for action segmentation. In: The British Machine Vision Conference (BMVC) (2021)
67. Yu, J., Lin, Z.L., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. IEEE/CVF International Conference on Computer Vision (ICCV) (2018)

68. Zatsarynna, O., Farha, Y., Gall, J.: Multi-modal temporal convolutional network for anticipating actions in egocentric videos. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW) (2021)
69. Zatsarynna, O., Gall, J.: Action anticipation with goal consistency. In: IEEE International Conference on Image Processing (ICIP) (2023)
70. Zatsarynna, O., Farha, Y.A., Gall, J.: Self-supervised learning for unintentional action prediction. In: DAGM German Conference on Pattern Recognition (GCPR) (2022)
71. Zhang, M., Cai, Z., Pan, L., Hong, F., Guo, X., Yang, L., Liu, Z.: Motiondiffuse: Text-driven human motion generation with diffusion model. arXiv preprint arXiv:2208.15001 (2022)
72. Zhao, H., Wildes, R.P.: On diverse asynchronous activity anticipation. In: European Conference on Computer Vision (ECCV) (2020)
73. Zhao, Y., Krähenbühl, P.: Real-time online video detection with temporal smoothing transformers. In: European Conference on Computer Vision (ECCV) (2022)
74. Zhong, Z., Martin, M., Voit, M., Gall, J., Beyerer, J.: A survey on deep learning techniques for action anticipation. In: Arxiv (2023)
75. Zhong, Z., Schneider, D., Voit, M., Stiefelhagen, R., Beyerer, J.: Anticipative feature fusion transformer for multi-modal action anticipation. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2023)