This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Joint Hand Motion and Interaction Hotspots Prediction from Egocentric Videos

Shaowei Liu[1]* Subarna Tripathi[2] Somdeb Majumdar[2] Xiaolong Wang[3]
[1]University of Illinois Urbana-Champaign [2] Intel Labs [3]UC San Diego

Abstract

We propose to forecast future hand-object interactions given an egocentric video. Instead of predicting action labels or pixels, we directly predict the hand motion trajectory and the future contact points on the next active object (i.e., interaction hotspots). This relatively low-dimensional representation provides a concrete description of future interactions. To tackle this task, we first provide an automatic way to collect trajectory and hotspots labels on large-scale data. We then use this data to train an Object-Centric Transformer (OCT) model for prediction. Our model performs hand and object interaction reasoning via the self-attention mechanism in Transformers. OCT also provides a probabilistic framework to sample the future trajectory and hotspots to handle uncertainty in prediction. We perform experiments on the Epic-Kitchens-55, Epic-Kitchens-100 and EGTEA Gaze+ datasets, and show that OCT significantly outperforms state-of-the-art approaches by a large margin. Project page is available at https://stevenlsw.github.io/hoi-forecast.

1. Introduction

Achieving the ability to predict a person's intent, preference and future activities is one of the fundamental goals for AI systems. This is particularly useful when it comes to egocentric video data for applications such as augmented reality (AR) and robotics. Imagining with an egocentric view inside the kitchen (e.g., Figure 1), if an AI system can forecast what the human would do next, an AR headset could provide useful and timely guidance, and a robot can react and collaborate with the human more smoothly.

What space should the model predict on? Recent approaches [25,26,28,72] have been proposed to predict the discrete future action category given a sequence of frames as inputs, namely action anticipation. However, predicting a semantic label does not reveal how the human moves and what the human will interact with in the future. On the other hand, predicting pixels for future frames [9, 44, 53, 84] is very challenging due to its high dimension outputs with large uncertainties. Instead of adopting these two representations, our work is inspired by recent work on human motion trajectory prediction [11] which takes images as inputs and outputs the coordinates of future pose joints. Trajectory not only provides a concrete description of motion, but also is a much smaller space to predict compared to pixel prediction. However, unlike previous works, prediction in egocentric videos also involves dense interactions with objects, which cannot be modeled by trajectory alone.

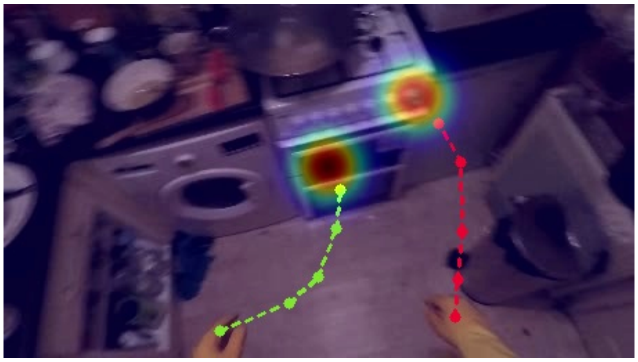

Figure 1. Going beyond predicting a single action label in the future, we propose to jointly predict the future hand motion trajectories (blue and red lines) and interaction hotspots (heatmaps) on the next-active object in egocentric videos.

In this paper, we propose to jointly predict the future hand motion trajectory and the interaction hotspots (affordance) of the next-active object, given a sequence of input frames from an egocentric video. Starting from the final frame of the input video, we will predict the trajectories for both hands by sampling from a probabilistic distribution inferred by the model. Instead of learning a deterministic model, we tackle the uncertainty of future in a probabilistic manner. At the same time, we will predict the contact points on the next-active object interacted by the future hands. These contact points are also represented via probabilistic distributions in the form of interaction hotspots [61] and conditioned on the predicted hand trajectory. To perform joint predictions, we introduce a Transformer-based model and an automatic way to generate a large-scale dataset for training.

Instead of collecting annotations for hand trajectories and interaction hotspots with high-cost human labor, we propose an automatic manner to collect the data in a large-scale. Given a video, we call the input frames to our model the observation frames and the predicted ones are called future frames. We first utilize off-the-shelf hand detectors [73] to locate hands in all the future frames. Since the camera is usually moving in egocentric videos, we leverage homography in nearby frames and project the detected future hands' locations back to the last observation frame. In this way, all the detections are aligned in the same coordinate system. Similarly, we also detect the locations where

*Work partially done during an internship at Intel Labs.

the hand interacts with the object in future frames, and project them back to the last observation frame. This process prepares the data for training our prediction model, and we generate labels for Epic-Kitchens-55, Epic-Kitchens-100 and EGTEA Gaze+ datasets without any human labor.

With the collected data, we propose to learn a novel Object-Centric Transformer (OCT) model which captures the hand-object relations from videos for hand trajectory and interaction hotspots prediction. Given the observation frames as inputs, we first extract their visual representations with a ConvNet. We perform hand and object detection and adopt RoI Align [34] to extract their features. We take both hand and object features as object-centric tokens, and the average-pooled frame feature as image context tokens. We forward all tokens from all input frames to a Transformer encoder, which performs hand, object and environment context interaction reasoning using self-attention. Instead of decoding in a deterministic manner, we adopt the Conditional Variational Autoencoders (C-VAE) as network head in the Transformer decoder to model the uncertainty in prediction. Specifically, we compute cross-attention between the output tokens from the Transformer encoder and predicted future hand locations in the Transformer decoder. The obtained tokens are taken as conditional variables for the C-VAE. The decoder will predict both the hand trajectories and interaction hotspots jointly, and the training is supervised by a reconstruction loss corresponding to the ground-truths.

We perform evaluation on Epic-Kitchens-55 [15], Epic-Kitchens-100 [16] and EGTEA Gaze+ [46] datasets. We manually annotate the validation sets with trajectory and hotspots labels using the Amazon Mechanical Turk platform. Our OCT model significantly outperforms the baselines on both hand trajectory and interaction hotspots prediction tasks. Interestingly, we find that trajectory estimation helps interaction hotspots prediction and with more automatic annotated training data we can get better results. Finally, we experiment with fine-tuning the trained model on the action anticipation task, and find that predicting hand trajectory and interaction hotspots can benefit classifying future actions.

Our contributions are the following:

- We propose to jointly predict hand trajectory and interaction hotspots from egocentric videos, and collect new training and test annotations.
- A novel Object-Centric Transformer which models the hand and object interactions for predicting future trajectory and affordance.
- We not only achieve state-of-the-art performance on both prediction tasks on Epic-Kitchens and EGTEA Gaze+ datasets, but also show our model can help the action anticipation task.

2. Related Work

Video anticipation. Video anticipation aims to forecast future events in videos, including future frames prediction [9, 36, 44, 53, 80, 84, 89], action anticipation [25, 26, 28, 58, 72, 87], and dynamics learning [23, 29, 67]. However, most of these works either relied on anticipating high-dimensional visual representations of the future, which is extremely challenging in dynamic scenes with appearance changes and moving agents, or focused on predicting a semantic label of future actions. The labels could neither tell us where the person intends to move nor the object the person would like to interact with. On the contrary, we predict the future hand motion trajectory and the interaction hotspots, both reflecting human intention and future interactions in low dimensions.

Human motion forecasting. Predicting future human motions [11, 33, 47, 66, 82] or trajectories [1, 17, 48, 49, 55–57] has been a long-standing research topic. Many of them operate on third-person vision or fixed bird's eye view settings. Given that the first-person vision could better capture people's intention and interactions, as well as its applicability to AR and robotics [39, 43, 68], estimating human motions in egocentric videos [49, 64, 88] worth more attention. As hands are central means for humans to explore and manipulate in egocentric videos, forecasting where human hands move could reveal future activity and understand a person's intention. Liu *et al.* [49] also studied future hand trajectory estimation in egocentric videos, but their method is limited by manual annotation and single hand prediction. In contrast, we design an automatic way to collect the data on a large scale and can learn future trajectories for both hands from the data.

Grounded affordance prediction. Object affordance grounding [19, 21, 42, 60, 69, 74] refers to locating where the interaction occurs on an object. Given the video input, the affordance prediction task is to estimate the future active regions on an object that the human would interact. In general, there are two main categories of prediction, next-active object [5, 18, 24] and interaction hotspots [49, 54, 61]. The former one segments the object that will next come into contact with the hand holistically, neglecting the fine-grained spatial regions on the object's surface. The latter one outputs a heatmap to indicate salient regions on the object. Nagarajan *et al.* [61] proposed a weakly-supervised method to ground interaction hotspots on inactive images. Going one step further, we consider predicting interaction hotspots in egocentric videos. The task is more challenging since it needs to localize the next-active-object in a cluttered scene before grounding the hotspots. In our work, instead of using heatmap representation, we directly predict the contact locations more compactly.

Transformers for video forecasting. Following the immense success of Transformer [79] in natural language processing, recent studies showed its effectiveness in solving vision tasks [12, 14, 20, 50, 78]. The long-range reasoning and sequence modeling capability make Transformers suitable for video understanding [6, 27, 51, 86]. The TimeSformer [6] viewed the video as a sequence of patches and adopted divided space-time attention to capture spatial-temporal relations in videos. Transformers have also been widely used in video forecasting problems like action anticipation [28], trajectory estimation [30, 90] and human motion prediction [45, 66]. Recent works show promising results by incorporating VAE [41] into Transformers for generative modeling [22, 37, 66]. In our work, we propose an Object-Centric Transformer (OCT) that takes the RoIAlign [34] hand, object, and environment feature vectors extracted from a pre-trained ConvNet [83] as input tokens. The Transformer encoder adopts self-attention across all input tokens while the Transformer decoder computes cross-attention between output tokens from the encoder and predicted future hand locations. We also introduce the C-VAE [41] head in

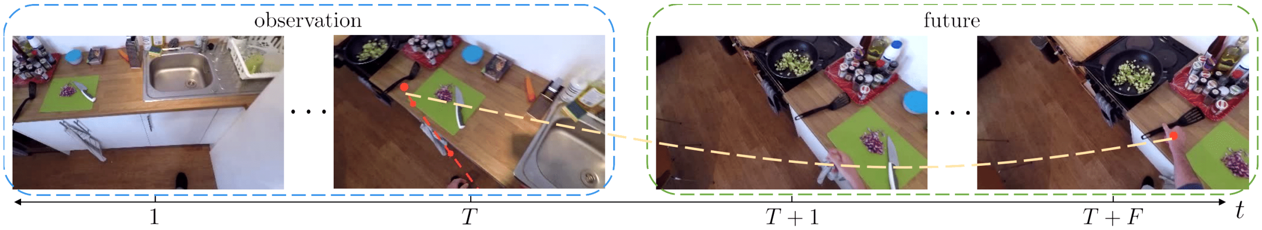

Figure 2. Given T observation frames as input (left), the goal is to forecast F steps future hand trajectory (right) and interaction hotspots. The orange curve shows how we project future hand locations (red dots) to the last observation frame. The future hand trajectory is shown in red dashed line.

the Transformer decoder to express the uncertainty of the future.

3. Problem Setup

3.1. Task Description

Given observation key frames $\mathcal{V}=\{f_1,\cdots,f_T\}$ of length T as input, where f_T is the last observation frame, our goal is to predict future hand trajectories $\mathcal{H}$ and object contact points $\mathcal{O}$ in the future time horizons of F, as shown in Figure 2. $\mathcal{H}=\{h_{T+1},\cdots,h_{T+F}\}$ denotes the future hand trajectories. At each time step t, the future hand location $h_t=(h_l^t,h_r^t)$ consists of the left and right hand 2D pixel locations in the last observation frame. Time step $T+F$ is when the hand-object contact occurs and frame f_{T+F} is the contact frame. $\mathcal{O}=\{o_1,\cdots,o_N\}$ denotes the future object contact points, where N is the maximum number of contact predictions and each element defines the 2D future contact location in the last observation frame.

3.2. Training Data Generation

We describe how to collect training labels of future hand trajectory $\mathcal{H}$ and object hotspots $\mathcal{O}$ from future key frames $\{f_{t+1},\cdots,f_T\}$ automatically without manual labor. We first run an off-the-shelf active hand-object detector [73] to get hand and object bounding boxes per frame, providing the future hand locations in each frame. Then we project them back to the last observation frame to collect a complete future hand trajectory. See Figure 2 for illustration. As shown in [62, 81], the global motion between two consecutive frames is usually small, and they can be related by a homography [77]. Given the homography between every two consecutive frames, we could build a chain and establish the relations of each future frame w.r.t. the last observation frame and project the future hand location back. In order to estimate the homography, we first exclude the moving objects, in particular the detected hands and objects in each frame. We mask out the corresponding location and find the correspondences between two frames outside the masked regions using SURF descriptor [3]. We calculate the homography by sampling 4 points and applying RANSAC to maximize the number of inliers.

Similarly, for collecting the hotspots labels, we perform an additional skin segmentation [70] and fingertip detection [1] within the active intersection region of hand and object bounding boxes to obtain contact points. Then we adopt a similar technique as above to project the sampled contact points from the contact frame to the last observation frame. More detailed discussions are provided in the supplementary.

3.3. Pre-processing

Given the video clip of observation frames $\mathcal{V}=\{f_1,\cdots,f_T\}$, we extract per-frame features, $\{X_1,\cdots,X_T\}$. Each frame consists of three types of input feature tokens $X_t=(X_h^t, X_o^t, X_g^t)$, where$X_i^t$ represents the feature of i-th type in frame t. Subscripts h,o,g refer to the hand, object, and global feature (environment context) vectors respectively. To this end, we first encode each frame f_t using a pre-trained Temporal Segment Network [83] (TSN) and extract hand and object RoIAlign [34] feature $\mathcal{P}_i^t$ given the detected bounding boxes from [73]. The global feature $\mathcal{P}_g^t$ is obtained similarly by average-pooling. Next, for hands and objects, we concatenate the pooled features along with the corresponding center coordinates and forward it to a Multi-Layer Perceptron (MLP), yielding the X_i^t. Take a hand as an example, $X_h^t=W_h[h_t;\mathcal{P}_h^t]$, W_h is the learnable weights of the corresponding hand MLP. When there is no hand or object detected in a certain frame, we set corresponding places with zero vectors. For global features, we directly use an MLP to obtain the output, $X_g^t=W_g\mathcal{P}_g^t$. All the features (global/hand/object) are taken as independent input tokens to the Transformer.

4. Object-Centric Transformer

The proposed Object-Centric Transformer (OCT) has an encoder-decoder architecture as shown in Figure 3. Both the encoder and the decoder stack multiple basic encoding and decoding blocks $\mathcal{B}$. Each block has an attention module named Att and a feed-forward module that consists of a two-layer MLP followed by a layer normalization [2] (LN). The only difference between the two blocks is the attention module, where we perform self-attention across input tokens in the encoding block and cross-attention between the encoder output and prediction in the decoding block. Assume $Q^{\ell-1}$ is the output query from block $\ell-1$, and the key, value, mask denoted by $K^{\ell-1}$,$V^{\ell-1}$,M respectively, are three additional inputs to block ℓ. Then the output $Q^\ell=\mathcal{B}(Q^{\ell-1};K^{\ell-1};V^{\ell-1},M)$, of the block $\mathcal{B}$ could be written as follows:

$$[Q;K;V]=W[Q^{\ell-1};K^{\ell-1};V^{\ell-1}]$$
$$Q'=Q^{\ell-1}+\text{Att}(Q,K,V,M)$$
$$Q^\ell=Q'+\text{MLP}(\text{LN}(Q'))$$

[1] https://www.computervision.zone/courses/finger-counter/

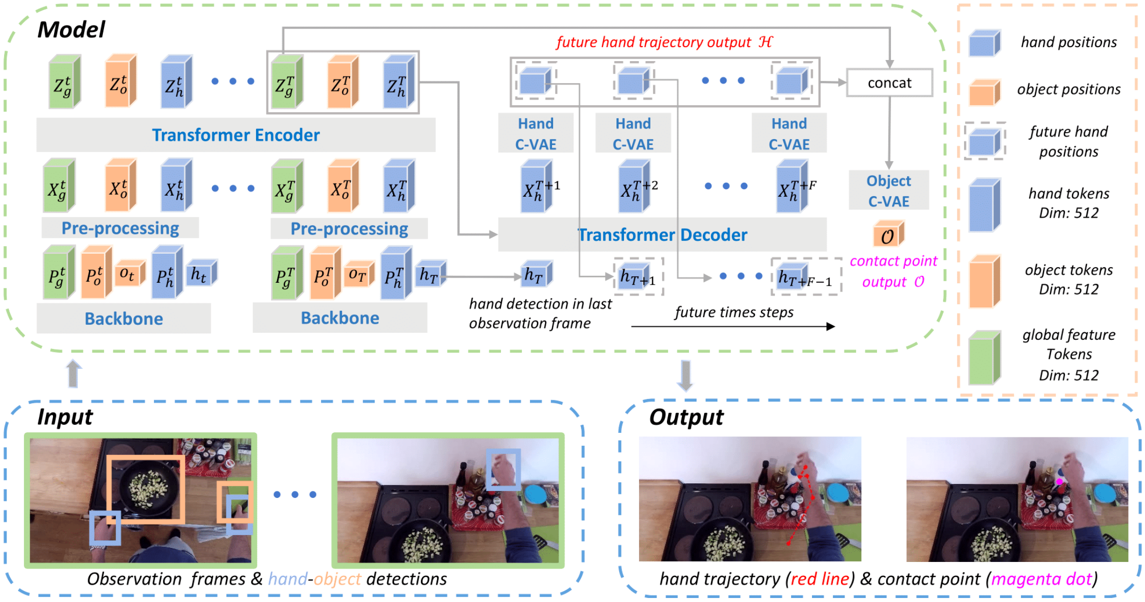
Model
Z_g^t Z_o^t Z_h^t Z_g^T Z_o^T Z_h^T
Transformer Encoder
X_g^t X_o^t X_h^t X_g^T X_o^T X_h^T
Pre-processing
Pre-processing
P_g^t P_o^t o_t P_h^t h_t
P_g^T P_o^T o_T P_h^T h_T
Backbone
Backbone
future hand trajectory output $\mathcal{H}$
concat
Hand C-VAE
Hand C-VAE
Hand C-VAE
X_h^{T+1} X_h^{T+2} X_h^{T+F}
Object C-VAE
$\mathcal{O}$
Transformer Decoder
h_T h_{T+1} h_{T+F-1}
contact point output $\mathcal{O}$
hand detection in last observation frame
future times steps
hand positions
object positions
future hand positions
hand tokens Dim: 512
object tokens Dim: 512
global feature Tokens Dim: 512
Input
Observation frames & hand-object detections
Output
hand trajectory (red line) & contact point (magenta dot)

Figure 3. The OCT has an encoder-decoder architecture. The input is observation frames and corresponding hand-object detections. The output is the future hand trajectory and contact point prediction. Inside the model, we use ConvNet to extract hand, object and global features of each frame as input tokens to the Transformer encoder. All tokens (global/hand/object) are passed through the Transformer independently. We take the output from the encoder and previously predicted hand locations as input to the decoder. The decoder output is sent to hand C-VAE and object C-VAE to obtain the final results.

where the input tokens $Q^{\ell-1}$, $K^{\ell-1}$, $V^{\ell-1}$ are first passed through a linear transformation layer parameterized by W to produce embeddings Q, K, V; then they are forwarded to the attention module Att. The attention output is sent to the feed-forward module with residual connection [35] to obtain the final output Q^{ℓ}. The attention operator is defined as follows:

$$\mathrm{Att}(Q,K,V,M) = \mathrm{softmax}(\frac{QK^T + M}{\sqrt{D}})V$$

D is the dimension of the attention module. The attention operator computes a weighted sum of value V where the weight is computed by the taking dot-product between query Q and key K and adding up the mask M followed by scaling and Softmax normalization. M masks out the padding values in key K by setting corresponding positions to *-inf* before the Softmax calculation.

4.1. Encoder

The Encoder $\mathcal{E}$ stacks multiple encoding blocks $\mathcal{B}$ and generate outputs $\{Z_1,\cdots,Z_T\}$ from inputs $\{X_1,\cdots,X_T\}$ (Sec. 3.3):

$$Z_1,\cdots,Z_T = \mathcal{E}(X_1,\cdots,X_T) \tag{1}$$

Each $X_t = (X_h^t, X_o^t, X_g^t)$ and $Z_t = (Z_h^t, Z_o^t, Z_g^t)$ consists of a triplet of tokens, hand tokens h, object tokens o and global tokens g. The input tokens are encoded by two kinds of embeddings, a learnable spatial embedding [20] to represent category-specific (global/hand/object) information of different features, and the sinusoidal positional embedding [79] to incorporate the temporal position information. All tokens passed through the encoding blocks independently. In each encoding block, we compute self-attention over all input tokens across space and time. Given that there could be padded tokens when there is no hand/object detected in certain frames, we use mask M to mask out such tokens. Thus we have $Q^{\ell} = \mathcal{B}(Q^{\ell-1}, Q^{\ell-1}, Q^{\ell-1}, M)$ in each of the ℓ-th encoding blocks, where the query, key, and value come from the same output $Q^{\ell-1}$.

4.2. Decoder

The decoder $\mathcal{D}$ predicts future hand feature X_{T+t} one at a time, where $t \in (T+1, T+F)$ is the future time step. The predicted features X_{T+t} are then sent to the trajectory head network to predict the future hand location h_{T+t}. At each step, the decoder is auto-regressive [32], consuming the previously generated future hand locations $(h_T,\cdots,(h_{T+t-1})$ as additional input when generating X_{T+t}. The 0-th input to the decoder is the hand location h_T in the last observation frame. The prediction at future time step t of the decoder can be written as follows:

$$X_{T+t} = \mathcal{D}(h_T,\cdots,h_{T+t-1}) \tag{2}$$

The decoder consists of several decoding blocks $\mathcal{B}$. Each block works like an encoding block, except it performs cross-attention that takes the output $Q^{\ell-1}$ from block $\ell-1$ as query, output token Z_T (Sec. 4.1) of the encoder in the last observation frame as key and value. To constrain the decoder block to only attend to earlier input positions of $Q^{\ell-1}$, we create a mask M' that masks out subsequent positions. Thus we have $Q^{\ell} = \mathcal{B}(Q^{\ell-1}, Z_T, Z_T, M')$ in each decoding block ℓ, where the three inputs to block $\mathcal{B}$ correspond to query, key, and value. Before forwarding the input to the 1st decoding block, we encode it with sinusoidal positional embedding [79] to incorporate the temporal information.

4.3. Head Networks

We employ two C-VAE as two heads; one for hand trajectory estimation and another for object contact points prediction. A

C-VAE contains two functions: an encoding function $\mathcal{F}_{enc}$ which encodes the input x and condition c into a latent z-space parameterized by mean μ and co-variance σ, and a decoding function $\mathcal{F}_{dec}$ which decodes sampled z from the latent space and condition c to reconstruct input x. Formally, we have $\mu,\sigma = \mathcal{F}_{enc}(x;c)$ and $\hat{x} = \mathcal{F}_{dec}(z;c)$ where $z \sim \mathcal{N}(\mu,\sigma^2)$. The $\mathcal{F}_{enc}$ and $\mathcal{F}_{dec}$ are implemented as a MLP. At training time, we minimize the objective of reconstruction error $\mathcal{L}_{recon}(x,\hat{x}) = ||x-\hat{x}||^2$ between ground-truth x and predicted $\hat{x}$, as well as a KL-Divergence term $\mathcal{L}_{kl}(\mu,\sigma) = -KL[\mathcal{N}(\mu,\sigma^2)||\mathcal{N}(0,1)]$ that regularizes the latent z-space close to normal distribution $\mathcal{N}(0,1)$. During inference, we sample z from the latent space and concatenate with condition c to predict output $\hat{x}$.

Hand C-VAE. At future time step t, the hand C-VAE takes hand locations h_{T+t} as input, and conditioned on the hand feature tokens X_{T+t} (Sec. 4.2) from the decoder output. The encoding function $\mathcal{F}^h_{enc}$ outputs distribution parameters μ_h and σ_h of the latent space. The decoding function $\mathcal{F}^h_{dec}$ predicts future hand locations $\hat{h}_{T+t}$. Thus, the loss function of hand C-VAE $\mathcal{L}_{\mathcal{H}}$ is the reconstruction loss over all future time steps t and KL-Divergence regularization:

$$\mathcal{L}_{\mathcal{H}} = \sum_{t=1}^{F} \mathcal{L}_{recon}(h_{T+t},\hat{h}_{T+t}) + \mathcal{L}_{kl}(\mu_h,\sigma_h) \tag{3}$$

Object C-VAE. The object C-VAE takes the future contact points o sampled from the generated ground-truth set of future contact points $\mathcal{O}$ (Sec 3.1) as input, and is conditioned on global feature token Z^T_g (Sec. 4.1) in the last observation frame from the Transformer encoder output and future hand locations $(h_T, \cdots, h_{T+F})$. The hand trajectory is forwarded to a fully-connected layer and concatenated with Z^T_g as the conditional input. We found that the object's future contact points could be predicted more accurately with the future hand trajectory as conditional input. During training, we use teacher forcing [85] by taking ground-truth future hand trajectory as input. During inference, we use the predicted future hand trajectory as input for the object C-VAE. Similar to hand C-VAE, the encoding function $\mathcal{F}^o_{enc}$ outputs μ_o and σ_o, while the decoding function $\mathcal{F}^o_{dec}$ predicts future object contact points $\hat{h}_o$. The loss function, $\mathcal{L}_{\mathcal{O}}$, of the object C-VAE is the following:

$$\mathcal{L}_{\mathcal{O}} = \mathcal{L}_{recon}(o,\hat{o}) + \mathcal{L}_{kl}(\mu_o,\sigma_o) \tag{4}$$

4.4. Training and Inference

Training. We train the Object-Centric Transformer with both the hand trajectory loss $\mathcal{L}_{\mathcal{H}}$ and object contact point loss $\mathcal{L}_{\mathcal{O}}$. We observe that the object contact points labels are noisier than the hand trajectory labels in our generated training set. The total loss is $\mathcal{L} = \mathcal{L}_{\mathcal{H}} + \lambda \mathcal{L}_{\mathcal{O}}$, where $\lambda = 1e^{-1}$, is a constant coefficient to balance the training loss.

Inference. During inference, we sample 20 times for both the trajectories and contact points from the C-VAE for each input video. Following the evaluation protocol in previous work [57, 59, 67, 89] that involves stochastic unit in trajectory estimation, we report the minimum of among 20 samples for trajectory evaluation. We collect all predicted contact points and convert them into a heatmap by centering a Gaussian distribution over each point for affordance evaluation.

5. Experiments

5.1. Implementation Details

We sample $T = 10$ frames at 4 FPS (frames per second) as input observations and forecast 1s in the future on Epic-Kitchens, where the future time horizon $F = 4$. We sample $T = 9$ frames at 6 FPS on EGTEA Gaze+, forecasting 0.5s with $F = 3$. We use the pre-trained TSN [83] from [25] as the backbone to extract RGB features from the input video clip. We use the detector proposed in [73] to detect active hand and object bounding boxes in each input frame. Then we use RoIAlign [34] and average pooling to produce a 1024-D vector for hand $\mathcal{P}^t_h$, object $\mathcal{P}^t_o$ and global features $\mathcal{P}^t_g$ (Sec. 3.3) at input time step t. We set the embedding dimension of the OCT to 512. We set the number of blocks in encoder and decoder to be 6 and 4 on Epic-Kitchens, 2 and 1 on EGTEA Gaze+. Each block has 8 attention heads. For encoding and decoding function $\mathcal{F}_{enc}$ and $\mathcal{F}_{dec}$ in C-VAE, we use a single-layer MLP for both the hand and object. The OCT is trained using Adam optimizer [40] with a learning rate of $1e-4$ and a batch size of 128. Training takes 35 epochs on Epic-Kitchens, 25 epochs on EGTEA Gaze+, including 5 epochs warm-up [31] and rest epochs with cosine decay [52]. During inference, we sample 20 times from the C-VAE for both hand trajectory and object contact points. Please see supplementary for detailed network structures.

5.2. Datasets

We use Epic-Kitchens-55 (EK55) [15], Epic-Kitchens-100 (EK100) [16] and EGTEA Gaze+ (EG) [46] datasets for experiments. The EK100 dataset is an extended version of the EK55 dataset. All datasets capture daily activities in the kitchen. Following the standard partition protocol in [16, 25], we split the training set of both datasets into training and validation splits. Given the test set are only used for action anticipation, we don't incorporate them in our experiments. We used the method in Sec. 3.2 to generate training labels automatically. The evaluation is performed on the validation split of all datasets. We manually filtered out badly generated hand trajectories and collected interaction hotspot annotations on a challenging subset via the Amazon Mechanical Turk platform (see supplementary for details). Given the last observation frame and contact frame in the future, we ask workers to place 1-5 future contact points in the last observation frame. Following [21, 61], we convert these annotations into an affordance heatmap as our ground-truth. On the EK55 dataset, we collect 8523 training samples, 1894 evaluation hand trajectories, and 241 interaction evaluation hotspots. On the EK100 dataset, we collect 24148 training samples, 3513 evaluation hand trajectories, and 401 evaluation interaction hotspots. On the EG dataset, we collect 1880 training samples, 442 evaluation hand trajectories, and 69 evaluation interaction hotspots.

Table 1. **Future hand trajectory estimation performance** on three datasets. (↑/↓ indicates higher/lower is better.) Our method outperforms previous approaches by a large margin and achieves comparable performance with the more elaborate divided space-time attention design.

	EK55		EK100		EG	
Methods	ADE ↓	FDE ↓	ADE ↓	FDE ↓	ADE ↓	FDE ↓
KF [7]	0.34	0.33	0.33	0.32	0.49	0.48
Seq2Seq [75]	0.18	0.14	0.18	0.14	0.18	0.14
FHOI [49]	0.36	0.35	0.35	0.35	0.34	0.34
Divided	**0.11**	**0.11**	**0.12**	**0.11**	0.15	0.15
Ours	0.12	0.12	**0.12**	**0.11**	**0.14**	**0.14**

Table 2. **Future object interaction hotspots prediction performance** on three datasets. (↑/↓ indicates higher/lower is better.) Our method outperforms prior work as well as Divided Attention significantly.

	EK55			EK100			EG		
Methods	SIM ↑	AUC-J ↑	NSS ↑	SIM ↑	AUC-J↑	NSS ↑	SIM ↑	AUC-J↑	NSS ↑
Center	0.09	0.61	0.33	0.09	0.62	0.31	0.09	0.63	0.27
Hotspots [61]	0.15	0.66	0.53	0.14	0.66	0.47	0.15	0.71	0.69
FHOI [49]	0.13	0.57	0.21	0.12	0.56	0.18	0.15	0.66	0.51
Divided	0.19	0.67	0.67	0.16	0.66	0.50	0.19	0.70	0.69
Ours	**0.22**	**0.70**	**0.87**	**0.19**	**0.69**	**0.72**	**0.23**	**0.75**	**1.01**

Table 3. **Cross-dataset hand trajectory estimation generalization performance**. All models are trained on Epic-Kitchens and tested on EGTEA Gaze+.

Methods	ADE ↓	FDE ↓
Seq2Seq [75]	0.24	0.19
FHOI [49]	0.31	0.32
Divided	**0.15**	**0.13**
Ours	0.16	**0.13**

Table 4. **Cross-dataset interaction hotspots prediction generalization performance**. All models are trained on Epic-Kitchens and tested on EGTEA Gaze+.

Methods	SIM ↑	AUC-J↑	NCC ↑
Hotspots [61]	0.15	0.71	0.69
FHOI [49]	0.12	0.54	0.10
Divided	0.21	0.74	0.80
Ours	**0.23**	**0.78**	**1.02**

5.3. Evaluation Metrics

Trajectory evaluation. We use normalized predicted 2D hand locations for evaluation using the following metrics.

- **Average Displacement Error (ADE)**. ADE is calculated as the ℓ_2 distance between the predicted future and the ground-truth averaged over the entire trajectory and two hands.
- **Final Displacement Error (FDE)**. FDE measures the ℓ_2 distance between the predicted future and ground truth at the last time step and averaged over two hands.

Interaction hotspots evaluation. We downsample and normalize the affordance heatmap with a resolution of $32x$ and ensure it sums up to 1. We don't use KLD (Kullback-Leibler Divergence) metric [10] as it is known to be sensitive to the tail of the distributions [4,63,91]. A small difference in the low-density regions may induce a huge KLD, especially severe for forecasting problems.

- **Similarity Metric (SIM)**: SIM [76] measures the similarity between the predicted affordance map distribution and the ground-truth one. It is computed as the sum of the minimum values at each pixel location between the predicted map and the ground-truth map.
- **AUC-Judd (AUC-J)**: AUC-J [38] is a variant of AUC proposed by Judd *et al.* [38]. The AUC evaluates the ratio of ground-truth captured by the predicted affordance map under different thresholds [10].
- **Normalized Scanpath Saliency (NSS)**: NSS [65] measures the correspondence between the predicted affordance map and the ground truth. It is computed by normalizing the predicted affordance map to have zero mean and unit standard deviation and averaging over ground truth locations.

5.4. Comparison to the state-of-the-art

Trajectory estimation. We evaluate our method against several baselines and state-of-the-art approaches. Kalman Filter (**KF**) [7] tracks the center of the hand in observation frames and predicts future hands locations. **Seq2Seq** [75] used LSTM to encode temporal information in the observation sequence and decode the target locations. Forecasting HOI (**FHOI**) [49] used I3D [13] (CNN) with motor attention to forecast future hand motion. Note that **FHOI** only used observation frames as input without accessing to hand-object detections. Besides, we also compare against Divided Attention (**Divided**) [6] Transformer design by applying temporal attention and spatial attention separately in the encoder of the OCT instead of doing them jointly (Sec. 4.1). We compute temporal attention only across hand tokens in different frames and spatial attention only within each frame. The results are shown in Table 1. Experimental results show that our method outperforms previous approaches by a large margin, improving the ADE by 50%, and FDE by 27.3% on the EK100 dataset against the second-best method of each metric, and achieves similar performance with the Divided Attention Transformer encoder design. This demonstrates the superiority of using Transformer to capture hand, object, and environment context interactions in egocentric videos.

Interaction hotspots prediction. We compare our results with the following methods. **Center** [49, 54, 61] generated the heatmap by placing a fixed Gaussian at the center of the image. **Hotspots** [61] anticipated spatial interaction regions using Grad-Cam [71], given the future action label as additional input. **FHOI** [49] and **Divided** [6] are the same method and baseline introduced in trajectory estimation, where they used I3D (CNN) and divided space-time Transformer encoder respectively. Table 2 summarizes the results of interaction hotspots prediction. Our method achieves the best performance across datasets and all metrics, improves SIM by +5%, AUC-J by +3%, and NSS by +25% on the EK100 dataset against the second-best method of each metric. Compared to Divided Attention, jointly modeling all hand-object tokens in observation frames is more beneficial for the prediction. These results also highlight that the Transformer architecture is more suitable for visual forecasting problems.

Cross dataset generalization. We evaluate learned models' cross-dataset generalization ability on both tasks. All models are trained on Epic-Kitchens and tested on EGTEA Gaze+. The hand trajectory estimation and interaction hotspots prediction performances are shown in Table 3 and Table 4 respectively. In addition to superior in-domain performance, our method

Table 5. Ablation study of **trajectory estimation by using different head network**. Stochastic models are in **bold**.

Heads	ADE ↓	FDE ↓
MLP	0.21	0.16
Bivariate	0.19	0.14
C-VAE	**0.12**	**0.11**

Table 6. Ablation study of **hotspots prediction by using different head network**. Stochastic models are in **bold**.

Heads	SIM ↑	AUC-J↑	NCC ↑
MLP	0.14	0.59	0.43
MDN	0.16	0.64	0.53
C-VAE	**0.19**	**0.69**	**0.72**

Table 7. Ablation study of **different C-VAE conditions**. $\mathcal{H}$ and $\mathcal{O}$ are future hand trajectory and contact point. $\mathcal{O}|\mathcal{H}$ stands for object C-VAE is conditioned on hand trajectory, similar for $\mathcal{H}|\mathcal{O}$. None means no conditions. Predicting contact points conditioned on hand trajectory gives the best performance for both tasks.

	Trajectory		Interaction Hotspots		
Condition	ADE ↓	FDE ↓	SIM ↑	AUC-J↑	NCC ↑
None	0.14	0.12	0.16	0.64	0.53
$\mathcal{H}\|\mathcal{O}$	0.13	0.12	0.16	0.64	0.54
$\mathcal{O}\|\mathcal{H}$	**0.12**	**0.11**	**0.19**	**0.69**	**0.72**

demonstrates strong cross-domain generalization by significantly outperforming other approaches across all metrics on both tasks.

5.5. Ablations and Analysis

We do ablation studies of our method on the EK100 dataset.

Head ablation. First, we evaluate the performance of using different stochastic/deterministic head networks for trajectory estimation and contact points prediction. For trajectory estimation, we compare proposed **C-VAE** with **MLP** and **Bivariate**. MLP deterministically outputs the future hand locations, while the Bivariate [1] assumes the future hand location follows a bivariate Gaussian distribution at each time step and explicitly samples from the predicted distribution during inference. For future contact points prediction, we compare **C-VAE** with **MLP** and **MDN**. MDN [8] adopts the Mixture Density Model (MDN) and models the distribution of future contact points as a mixture of Gaussians, where we set the number of Gaussian components to be 3. As shown in Table 5 and Table 6, stochastic models outperform the deterministic one on both tasks, thanks to their ability to deal with uncertainty. Adopting C-VAE against MLP improves the trajectory estimation performance by 75.0% on ADE and 45.5% on FDE, also obtains +5%, +10%, and +29% gain on SIM, AUC-J, and NCC of hotspots prediction. Besides, we also observe that C-VAE achieves better results compared to Bivariate and MDN. It demonstrates modeling stochastic in latent space works better than output space.

C-VAE condition. Besides modeling uncertainty in C-VAE, we analyze the effect of condition dependency in C-VAE. In Table 7, we evaluate the performance of using different C-VAE conditions for both the hand and the object. We compare three cases: no condition between the hand trajectory and object contact point, denoted as **None**; hand trajectory is conditioned on object contact point, denoted as $\boldsymbol{\mathcal{H}|\mathcal{O}}$; object contact points is conditioned on hand trajectory, denoted as $\boldsymbol{\mathcal{O}|\mathcal{H}}$. We find that explicitly incorporating the conditional dependency in C-VAE improves the overall performance. Predicting interaction hotspots conditioned on future hand trajectory leads to the best result on both tasks, obtaining +3%, +5%, and +18% performance gain on SIM, AUC-J, and NCC against conditioned on the inverse order. It suggests that the two tasks are intertwined and modeling their relation explicitly benefits the performance.

Table 8. Ablation study of **leveraging more automatically annotated training data**. We compare two models trained on EK55 and KE100 training split under the same setting, and evaluate the performance on EK100 validation split. Training with more automatically annotated training data (EK100) gives better performance on both tasks.

		Trajectory		Interaction Hotspots		
Train	Evaluation	ADE ↓	FDE ↓	SIM ↑	AUC-J↑	NCC ↑
EK55	EK100	0.13	0.12	0.18	0.68	0.60
EK100	EK100	**0.12**	**0.11**	**0.19**	**0.69**	**0.72**

Table 9. Ablation study of **different input features** for trajectory estimation. Global features that encode environmental context and hand features are most important.

hand	object	global	ADE ↓	FDE ↓
✗	✓	✓	0.13	0.16
✓	✗	✓	0.13	**0.11**
✓	✓	✗	0.15	0.13
✓	✓	✓	**0.12**	**0.11**

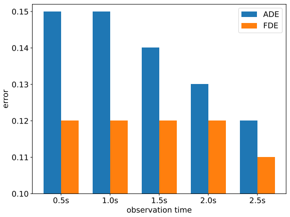

Figure 4. Ablation study of **observation time** contributes to trajectory estimation. Longer temporal context are helpful.

More training data. As we generate our training data automatically without manual labeling, we are interested in understanding whether leveraging more automatically annotated training data can help boost performance. We trained two models under the same setting on EK55 and EK100 training split respectively. We evaluate their performances on the manually-collected EK100 validation split having no overlap with both EK55 and EK100 training splits. As shown in Table 8, we observe that a model trained with larger data (EK100) outperforms a model trained on EK55 on both tasks. This demonstrates the effectiveness of our method. Even though there is inevitable noise introduced during training data generation, our method could still learn useful representations for forecasting and it benefits from utilizing more training data. It also indicates a great potential for deploying our method on larger-scale egocentric videos.

Input ablation. We evaluate the contributions of different input settings to the performance of trajectory estimation. Contact points prediction performance is conditioned on the trajectory, thus the input relation is not as straightforward as trajectory estimation. We evaluate the contribution of different features by

Table 10. **Action anticipation performance** on EK55 and EK100 validation split. We report top-5 accuracy/recall in terms of verb, noun, and action on EK55/EK100 respectively, following [25, 26, 28]. We add a single MLP on top of the OCT encoder to classify future actions. We compare the model trained from scratch (**Scratch**) and the model pre-trained using trajectory and hotspots estimation task (**Fine-tune**). The fine-tuned model greatly outperforms the model trained from scratch across all anticipation metrics on both datasets.

	EK55			EK100		
Train	Verb	Noun	Action	Verb	Noun	Action
Scratch	68.7	36.1	18.9	18.9	24.0	10.0
Fine-tune	**73.9**	**45.9**	**24.4**	**21.9**	**27.6**	**12.4**

removing them from the input and seeing the performance drop. As shown in Table 9, global features that encode environmental context (Sec. 3.3) and hand features are more crucial to the performance. By removing global features from input, ADE metric drops 25.0%. Without hand features as input, FDE metric drops 45.5%. This demonstrates that the global features are as imperative as hand features to the trajectory estimation. Besides, we also analyze the effect of different observation lengths in Figure 4. We observe that the performance improves as we incorporate more observation frames as input, which also proves our model is capable of capturing useful temporal information.

Action anticipation. So far we have shown the capability of our method on the trajectory estimation and interaction hotspots prediction. We further investigate the potential of our trained model for action anticipation task. We only use the OCT encoder and add a single MLP on top of it that takes the output global feature token in the last observation frame Z_g^T (Sec. 4.1) from the encoder as an input and predicts future action labels. Following previous work [25, 26, 28], we report top-5 accuracy on EK55 dataset, and top-5 recall on EK100 dataset for verb/noun/action predictions. Each action label consists of *(verb, noun)*. We trained our model on the same training split as we used for trajectory estimation and interaction hotspots prediction and evaluated the performance on corresponding validation splits. We only used cross-entropy loss between the prediction and ground-truth action labels during training. We obtained verb/noun prediction scores by marginalizing over the action scores. We compared two training strategies: training the model from scratch, denoted as **Scratch**, and fine-tuning the model pre-trained on trajectory and hotspots prediction task, denoted as **Fine-tune**. In the **Fine-tune** version, we freeze the OCT encoder and only trained the added MLP. The action anticipation performance of the two methods is shown in Table 10. The **Fine-tune** model outperforms **Scratch** model by a large margin across all datasets and all metrics. Specifically, **Fine-tune** obtains +4.3%, +8.5%, +4.4% performance gain on EK55 dataset, and +2.9%, +3.8%, +2.6% performance gain on EK100 dataset. Note that the performance of our model is not fully comparable with state-of-the-art action anticipation models as we only use a subset of samples for training and evaluation. Neither do we adopt any fancy tricks, network structures, or other loss functions for action anticipation.

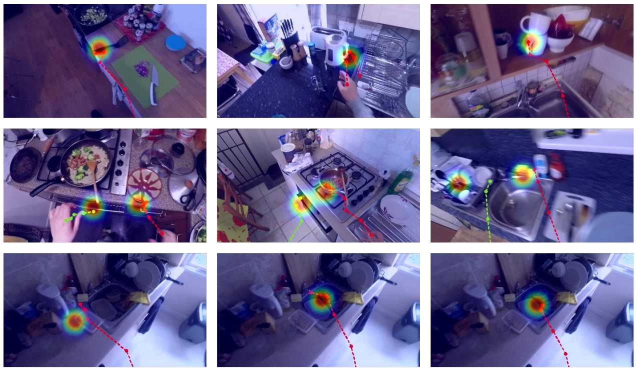

Figure 5. Qualitative visualization of future hand trajectory and interaction hotspots. The right and left hand trajectory are shown in red and green. The first two row shows single-hand and two-hand scenarios. The third row shows diversity in future trajectory and interaction hotspots prediction.

The experimental results show that the representation learned on trajectory estimation and interaction hotspots prediction could benefit the action anticipation task. This also proves the usefulness of the two tasks and generalization to other forecasting tasks.

Qualitative visualization. We visualize predicted future hand trajectory and interaction hotspots in Figure 5. Our method could deal with single and two hands scenarios (when hands are visible in the last observation frame) in the first two rows. Our method can also generate diverse predictions of the future in the third row. This demonstrates that our method is able to forecast the future hand-object interaction considering the future uncertainty. Please see supplementary for more visualizations.

6. Discussion

Conclusion. We propose to forecast future hand-object interactions in egocentric videos. We solve this task by proposing an automatic way to collect training data, and a novel Object-Centric Transformer (OCT) model that jointly predicts future hand trajectory and interaction hotspots given a sequence of observation frames as input. Through extensive experiments and ablations, we show that OCT significantly outperforms state-of-the-art approaches, and could benefit from stochastic modeling of the future and conditional dependency of trajectory and interaction hotspots into account. Furthermore, we show that our proposed method could leverage more training data to achieve better performance and easily adapt to action anticipation task. In the future, we hope to apply our method to solve more visual forecasting problems in egocentric videos with less human supervision.

Limitation and future Work. Our training dataset generation process relies on widely used off-the-shelf tools such as active hand-object detectors and skin segmentation. Thus the ground-truth annotations for training might be affected by the bias and errors from the off-the-shelf tools. In future work, we plan to incorporate self-supervision signals during training to make our model more robust to label noise.

Acknowledgements. Prof. Wang's lab is supported, in part, by grants from NSF CCF-2112665 (TILOS).

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 2, 7

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3

[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006. 3

[4] Assaf Ben-David, Hao Liu, and Andrew D Jackson. The kullback-leibler divergence as an estimator of the statistical properties of cmb maps. *Journal of Cosmology and Astroparticle Physics*, 2015(06):051, 2015. 6

[5] Gedas Bertasius, Hyun Soo Park, Stella X Yu, and Jianbo Shi. First person action-object detection with egonet. *arXiv preprint arXiv:1603.04908*, 2016. 2

[6] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021. 2, 6

[7] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 6

[8] Christopher M Bishop. Mixture density networks. Technical report, Aston University, 1994. 7

[9] Wonmin Byeon, Qin Wang, Rupesh Kumar Srivastava, and Petros Koumoutsakos. Contextvp: Fully context-aware video prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 753–769, 2018. 1, 2

[10] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Frédo Durand. What do different evaluation metrics tell us about saliency models? *IEEE transactions on pattern analysis and machine intelligence*, 41(3):740–757, 2018. 6

[11] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *European Conference on Computer Vision*, pages 387–404. Springer, 2020. 1, 2

[12] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 2

[13] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 6

[14] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 2

[15] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 2, 5

[16] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision*, pages 1–23, 2021. 2, 5

[17] Patrick Dendorfer, Aljosa Osep, and Laura Leal-Taixé. Goal-gan: Multimodal trajectory prediction based on goal position estimation. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 2

[18] Eadom Dessalene, Chinmaya Devaraj, Michael Maynord, Cornelia Fermuller, and Yiannis Aloimonos. Forecasting action through contact representations from first person video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[19] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 5882–5889. IEEE, 2018. 2

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 4

[21] Kuan Fang, Te-Lin Wu, Daniel Yang, Silvio Savarese, and Joseph J. Lim. Demo2vec: Reasoning object affordances from online videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 5

[22] Le Fang, Tao Zeng, Chaochun Liu, Liefeng Bo, Wen Dong, and Changyou Chen. Transformer-based conditional variational autoencoder for controllable story generation. *arXiv preprint arXiv:2101.00828*, 2021. 2

[23] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29:64–72, 2016. 2

[24] Antonino Furnari, Sebastiano Battiato, Kristen Grauman, and Giovanni Maria Farinella. Next-active-object prediction from egocentric videos. *Journal of Visual Communication and Image Representation*, 49:401–411, 2017. 2

[25] Antonino Furnari and Giovanni Farinella. Rolling-unrolling lstms for action anticipation from first-person video. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1, 2, 5, 8

[26] Antonino Furnari and Giovanni Maria Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6252–6261, 2019. 1, 2, 8

[27] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019. 2

[28] Rohit Girdhar and Kristen Grauman. Anticipative video transformer. *arXiv preprint arXiv:2106.02036*, 2021. 1, 2, 8

[29] Rohit Girdhar, Laura Gustafson, Aaron Adcock, and Laurens van der Maaten. Forward prediction for physical reasoning. *arXiv preprint arXiv:2006.10734*, 2020. 2

[30] Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10335–10342. IEEE, 2021. 2

[31] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 5

[32] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013. 4

[33] Ikhsanul Habibie, Daniel Holden, Jonathan Schwarz, Joe Yearsley, and Taku Komura. A recurrent variational autoencoder for human motion synthesis. In *28th British Machine Vision Conference*, 2017. 2

[34] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2, 3, 5

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[36] Dinesh Jayaraman, Frederik Ebert, Alexei A Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. *arXiv preprint arXiv:1808.07784*, 2018. 2

[37] Junyan Jiang, Gus G Xia, Dave B Carlton, Chris N Anderson, and Ryan H Miyakawa. Transformer vae: A hierarchical model for structure-aware and interpretable music representation learning. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 516–520. IEEE, 2020. 2

[38] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *2009 IEEE 12th international conference on computer vision*, pages 2106–2113. IEEE, 2009. 6

[39] Takeo Kanade and Martial Hebert. First-person vision. *Proceedings of the IEEE*, 100(8):2442–2453, 2012. 2

[40] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[41] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014. 2

[42] Hedvig Kjellström, Javier Romero, and Danica Kragić. Visual object-action recognition: Inferring object affordances from human demonstration. *Computer Vision and Image Understanding*, 115(1):81–90, 2011. 2

[43] Hema S Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):14–29, 2015. 2

[44] Sangmin Lee, Hak Gu Kim, Dae Hwi Choi, Hyung-Il Kim, and Yong Man Ro. Video prediction recalling long-term motion context via memory alignment learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3054–3063, 2021. 1, 2

[45] 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*, pages 13401–13412, 2021. 2

[46] 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)*, pages 619–635, 2018. 2, 5

[47] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. *arXiv preprint arXiv:1707.05363*, 2017. 2

[48] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019. 2

[49] 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*, pages 704–721. Springer, 2020. 2, 6

[50] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 2

[51] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 2

[52] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[53] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016. 1, 2

[54] Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. Learning visual affordance grounding from demonstration videos. *arXiv preprint arXiv:2108.05675*, 2021. 2, 6

[55] Wei-Chiu Ma, De-An Huang, Namhoon Lee, and Kris M Kitani. Forecasting interactive dynamics of pedestrians with fictitious play. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 774–782, 2017. 2

[56] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From goals, waypoints & paths to long term human trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15233–15242, 2021. 2

[57] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, pages 759–776. Springer, 2020. 2, 5

[58] Antoine Miech, Ivan Laptev, Josef Sivic, Heng Wang, Lorenzo Torresani, and Du Tran. Leveraging the present to anticipate the future in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[59] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14424–14432, 2020. 5

[60] Austin Myers, Ching L Teo, Cornelia Fermüller, and Yiannis Aloimonos. Affordance detection of tool parts from geometric features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1374–1381. IEEE, 2015. 2

[61] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8688–8697, 2019. 1, 2, 5, 6

[62] Tushar Nagarajan, Yanghao Li, Christoph Feichtenhofer, and Kristen Grauman. Ego-topo: Environment affordances from egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 163–172, 2020. 3

[63] Sherjil Ozair, Corey Lynch, Yoshua Bengio, Aaron van den Oord, Sergey Levine, and Pierre Sermanet. Wasserstein dependency measure for representation learning. *arXiv preprint arXiv:1903.11780*, 2019. 6

[64] Hyun Soo Park, Jyh-Jing Hwang, Yedong Niu, and Jianbo Shi. Egocentric future localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4697–4705, 2016. 2

[65] Robert J Peters, Asha Iyer, Laurent Itti, and Christof Koch. Components of bottom-up gaze allocation in natural images. *Vision research*, 45(18):2397–2416, 2005. 6

[66] Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. *arXiv preprint arXiv:2104.05670*, 2021. 2

[67] Haozhi Qi, Xiaolong Wang, Deepak Pathak, Yi Ma, and Jitendra Malik. Learning long-term visual dynamics with region proposal interaction networks. *arXiv preprint arXiv:2008.02265*, 2020. 2, 5

[68] Nicholas Rhinehart and Kris M Kitani. First-person activity forecasting from video with online inverse reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):304–317, 2018. 2

[69] Johann Sawatzky, Abhilash Srikantha, and Juergen Gall. Weakly supervised affordance detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2795–2804, 2017. 2

[70] Frerk Saxen and Ayoub Al-Hamadi. Color-based skin segmentation: An evaluation of the state of the art. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4467–4471. IEEE, 2014. 3

[71] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 6

[72] Fadime Sener, Dipika Singhania, and Angela Yao. Temporal aggregate representations for long-range video understanding. In *European Conference on Computer Vision*, pages 154–171. Springer, 2020. 1, 2

[73] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9869–9878, 2020. 1, 3, 5

[74] Dan Song, Nikolaos Kyriazis, Iason Oikonomidis, Chavdar Papazov, Antonis Argyros, Darius Burschka, and Danica Kragic. Predicting human intention in visual observations of hand/object interactions. In *2013 IEEE International Conference on Robotics and Automation*, pages 1608–1615. IEEE, 2013. 2

[75] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014. 6

[76] Michael J Swain and Dana H Ballard. Color indexing. *International journal of computer vision*, 7(1):11–32, 1991. 6

[77] Richard Szeliski. Image alignment and stitching: a tutorial, foundations and trends in computer graphics and computer vision. *Now Publishers*, 2(1):120, 2006. 3

[78] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 2

[79] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2, 4

[80] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 98–106, 2016. 2

[81] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013. 3

[82] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9401–9411, 2021. 2

[83] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 2, 3, 5

[84] Yunbo Wang, Lu Jiang, Ming-Hsuan Yang, Li-Jia Li, Mingsheng Long, and Li Fei-Fei. Eidetic 3d lstm: A model for video prediction and beyond. In *International conference on learning representations*, 2018. 1, 2

[85] Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989. 5

[86] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021. 2

[87] Yu Wu, Linchao Zhu, Xiaohan Wang, Yi Yang, and Fei Wu. Learning to anticipate egocentric actions by imagination. *IEEE Transactions on Image Processing*, 30:1143–1152, 2020. 2

[88] Takuma Yagi, Karttikeya Mangalam, Ryo Yonetani, and Yoichi Sato. Future person localization in first-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7593–7602, 2018. 2

[89] Yufei Ye, Maneesh Singh, Abhinav Gupta, and Shubham Tulsiani. Compositional video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10353–10362, 2019. 2, 5

[90] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 2

[91] Kai Zhang and James T Kwok. Simplifying mixture models through function approximation. *IEEE Transactions on Neural Networks*, 21(4):644–658, 2010. 6