Ego-Exo: Transferring Visual Representations from Third-person to First-person Videos

Yanghao Li\textsuperscript{1} Tushar Nagarajan\textsuperscript{1,2} Bo Xiong\textsuperscript{1} Kristen Grauman\textsuperscript{1,2} \\
\textsuperscript{1}Facebook AI Research \textsuperscript{2}UT Austin \\
lyttonhao@fb.com, tushar@cs.utexas.edu, bxiong@fb.com, grauman@fb.com

Abstract

We introduce an approach for pre-training egocentric video models using large-scale third-person video datasets. Learning from purely egocentric data is limited by low dataset scale and diversity, while using purely exocentric (third-person) data introduces a large domain mismatch. Our idea is to discover latent signals in third-person video that are predictive of key egocentric-specific properties. Incorporating these signals as knowledge distillation losses during pre-training results in models that benefit from both the scale and diversity of third-person video data, as well as representations that capture salient egocentric properties. Our experiments show that our “Ego-Exo” framework can be seamlessly integrated into standard video models; it outperforms all baselines when fine-tuned for egocentric activity recognition, achieving state-of-the-art results on Charades-Ego and EPIC-Kitchens-100.

1. Introduction

Egocentric video captured by wearable cameras offers a unique perspective into human behavior. It is the subject of a recent surge in research interest in first-person activity recognition [35, 77], anticipation [21, 1], and video summarization [36, 74, 13] with many valuable future applications in augmented reality and robotics. Compared to third-person videos, egocentric videos show the world through a distinct viewpoint, encode characteristic egocentric motion patterns due to body and head movements, and have a unique focus on hands, objects, and faces, driven by the camera wearer’s attention and interaction with their surroundings.

However, these unique properties also present a fundamental challenge for video understanding. On the one hand, learning models purely from egocentric data are limited by dataset scale. Current egocentric video datasets are small (e.g., 90k clips in EPIC-Kitchens-100 [11] vs. 650k in Kinetics-700 [34]) and lack diversity (e.g., videos only in kitchen scenes). On the other hand, a purely exocentric approach that uses more readily available third-person video—the status-quo for pre-training video models [19, 67, 69]—ignores the unique properties of egocentric video and faces a major domain mismatch. Prior work has shown that this latter strategy, though popular, is insufficient: pre-training egocentric action recognition models with third-person data alone produces significantly worse results than pre-training with first-person data [62]. In an attempt to bridge the domain gap, prior work explores traditional embedding learning [61, 75] or domain adaptation approaches [10], but they require paired egocentric and third-person videos that are either concurrently recorded or annotated for the same set of activities, which are difficult to collect and hence severely limit their scope.

Despite their differences, we hypothesize that the exocentric view of activity should in fact inform the egocentric view. First, humans are able to watch videos of other people performing activities and map actions into their own (egocentric) perspective; babies in part learn new skills in just this manner [48, 55]. Second, exocentric video is not devoid of person-centered cues. For example, a close-up instructional video captured from the third-person view may...
nonetheless highlight substantial hand-object interactions; or video captured with a hand-held phone may follow an event (e.g., a parade) as it unfolds with attentional cues related to a head-mounted camera.

Building on this premise, in this work we ask: “How can we best utilize current video datasets to pre-train egocentric video models?” Our key idea is to discover latent signals in third-person video that approximate egocentric-specific properties. To that end, we introduce a feature learning approach in which ego-video features are guided by both exo-video activity labels and (unlabeled) ego-video cues, to better align traditional third-person video pre-training with downstream egocentric video tasks.

Specifically, we introduce a series of ego-inspired tasks that require the video model to be predictive of manipulated objects, spatiotemporal hand-object interaction regions, and general egocentricity. Then we incorporate these tasks into training as knowledge-distillation losses to supplement an action classification pre-training objective on third-person video. See Fig. 1.

By design, our video models can continue to enjoy large amounts of labeled third-person training data, while simultaneously embedding egocentric signals into the learned features, making them a suitable drop-in replacement for traditional video encoders for egocentric video tasks. Finally, our approach does not assume any paired or activity-labeled egocentric videos during pre-training; the egocentric signals are directly inferred from third-person video.

Our experiments on three challenging egocentric video datasets show that our “Ego-Exo” framework learns strong egocentric feature representations from third-person video. On Charades-Ego [62], our model improves over models pre-trained on Kinetics—the standard pre-training and fine-tuning paradigm—by +3.26 mAP, and outperforms methods that specifically aim to bridge the domain gap between viewpoints. Finally, our pre-trained model achieves state-of-the-art results on EPIC-Kitchens-100 [11], the largest available first-person dataset.

2. Related Work

Egocentric video understanding The unique viewpoint in egocentric video presents interesting research challenges including action recognition and anticipation [77, 21, 59, 1], daily life summary generation [36, 74], inferring body pose [32, 51], and estimating gaze [37, 31]. Several egocentric video datasets have been created to support these challenges [12, 38, 54, 62]. Model architectures proposed for these tasks include multi-stream networks [46, 38, 35, 68], recurrent networks [22, 21, 65], 3D conv nets [53, 44] and spatially grounded topological graph models [50].

These architectures vary significantly, but all use video encoders that are similarly pre-trained with third-person video datasets, despite being applied to egocentric video tasks. In contrast, we introduce key egocentric losses during exocentric video pre-training that bridge the domain gap when applied to downstream egocentric video tasks.

Joint first/third person video understanding Several strategies have been proposed to address the domain gap between first and third person video. Prior work learns viewpoint-invariant representations using embedding learning methods, and applies them to action recognition [63, 61, 3], video summarization [30], image retrieval [17], person segmentation [72], and attention-driven gaze prediction [75]. Image generation methods [16, 56, 57, 41] use generative adversarial frameworks to synthesize one viewpoint from the other. Viewpoint invariance has also been treated as a domain adaptation task in prior work, adapting third-person video models for overhead drone-footage [10]. Other methods use egocentric video as a modality to supplement top-view footage to improve identification and tracking models [2, 4, 5, 73].

The above methods require paired datasets that are either simultaneously recorded or that share the same labels for instances across viewpoints. In contrast, our method leverages only third-person video datasets, but is augmented with pseudo-labels (derived from first-person models) to learn egocentric video representations, thus circumventing both the need for first-person video during pre-training and the need for paired labeled data.

Knowledge distillation for video In knowledge distillation (KD), one network is trained to reproduce the outputs of another [29]. Distillation serves to compress models [29, 49, 8] or incorporate privileged information from alternate tasks [43]. In videos, KD can incorporate information from alternate modalities like audio [24, 6], depth and flow [28, 64], or a combination [45], and object-level information [52]. In the context of self-supervised learning, prior work assigns weak image labels to video instances as supervision for video-level models [25]. In contrast, we use inferred weak labels that are relevant to the egocentric domain, and we use them alongside third-person video labels, rather than in place of them during pre-training.

Egocentric cues in video understanding models Egocentric video offers several unique cues that have been leveraged to improve video understanding models [39, 46]. These include attention mechanisms from gaze and motor attention [47, 38, 42], active object detection [20, 7, 15, 70], and hands in contact [66, 60, 33]. We are also interested in such important egocentric cues, but unlike prior work we do not train models on labeled egocentric video datasets to detect them. Instead, we embed labels predicted for these cues as auxiliary losses in third-person video models to steer feature learning towards egocentric relevant features. Unlike any of these prior methods, our goal is to leverage third-person video to pre-train first-person video models.
3. Ego-Exo Approach

Our goal is to learn egocentric video representations from third-person video datasets, by discovering and distilling important cues about hands, objects, and interactions (albeit in a different viewpoint) that are relevant to egocentric activity during pre-training. To do this, we automatically assign third-person video instances with various egocentric pseudo-labels that span simple similarity scores to complex spatiotemporal attention maps, and then we introduce auxiliary losses that force our features to be predictive of these pseudo-labels. On the one hand, our approach retains the benefits of large-scale third-person video and the original action classification task to guide general video feature learning. On the other hand, we steer feature learning towards better egocentric features using automatically generated egocentric labels, as opposed to collecting manually labeled instances.

In the following sections, we first describe the traditional video pretraining framework (Sec 3.1) and how we incorporate our auxiliary loss terms into it (Sec 3.2). Next we describe the three egocentric tasks we use, namely Ego-Score, Object-Score, and Interaction-Map (Sec 3.3). Finally, we present our full training and evaluation pipeline in Sec 3.4.

3.1. Video model pre-training

Video models benefit greatly from strong initializations. The standard procedure for training egocentric video models is thus to first pre-train models using large-scale third-person video datasets, and then fine-tune for a specific downstream task.

More formally, we are provided with a large-scale third-person (exocentric) video dataset $V_{exo}$. In pre-training, each video instance $v \in V_{exo}$ consists of $T$ frames $\{f_1, ..., f_T\}$ and an associated action label $y^{act}$. These frames are encoded into a series of $N$ spatiotemporal clip features $\{x_1, ..., x_N\}$, where $x_i \in \mathbb{R}^{c \times h \times w}$, using a video encoder backbone (e.g., a 3D CNN model). These features are then passed to a classifier head, which spatiotemporally pools the feature and uses a linear classifier to generate the predicted action class $\hat{y}^{act}$. Predictions are typically generated for each clip and then averaged to generate video-level predictions.

The network is trained to minimize the cross entropy loss $\mathbb{L}_{act}(y^{act}, \hat{y}^{act})$. See Fig 2 (left panel).

Once pre-trained, the backbone weights are retained, the head is replaced with a task-specific classifier, and the new network is trained with instances from a target egocentric dataset $V_{ego}$ to predict egocentric video labels.

3.2. Ego-Exo pre-training

Third-person pre-training alone results in strong, general-purpose video features. However, it ignores important egocentric signals and introduces a domain gap that limits its utility for downstream egocentric tasks. We introduce auxiliary egocentric task losses to overcome this gap.

Specifically, along with datasets $V_{exo}$ and $V_{ego}$, we assume access to off-the-shelf video models that address a set of egocentric video understanding tasks. For each task $\tau$, the model $M^\tau$ takes as input a video (as either frames or clips) and generates predicted labels $y^\tau$. We use these pre-trained models to associate egocentric pseudo-labels to the third-person video instances in $V_{exo}$. We stress that the videos in $V_{exo}$ are not manually labeled for any task $\tau$.

We introduce a task-specific head $H^\tau$ for each task that is trained to approximate these pseudo-labels for each video.

Figure 2: Ego-Exo framework. To enhance traditional pre-training (left panel), we generate soft-labels for third-person videos from a set of pre-trained egocentric models (top-right) that capture a variety of key egocentric signals (Sec 3.3), and we train distillation modules to approximate the responses of these models (bottom-right). Once pre-trained, the video backbone can be directly fine-tuned for a downstream egocentric task.
instance, leading to an auxiliary loss term $\mathbb{L}_{\tau}(H^\tau(v), y^\tau)$. Each head is trained to approximate the response of an egocentric video model when applied to a third-person video instance, and thus can be seen as a knowledge-distillation mechanism that distills information from the egocentric tasks into the video encoder model. The final pre-training objective is the combination of the action classification loss $\mathbb{L}_{act}$ and each of the auxiliary loss terms $\mathbb{L}_{\tau}$. See Fig 2 for our full framework.

Note that these pseudo-labels vary in structure and semantics, ranging from scalar scores (e.g., to characterize how egocentric-like a third-person video is), categorical labels (e.g., to identify the manipulated objects in video) and spatiotemporal attention maps (e.g., to characterize hand-object interaction regions). Moreover, these labels are egocentric-specific, but they are automatically generated for third-person video instances. This diverse combination leads to robust feature learning for egocentric video, as our experiments will show. Once pre-trained, we can retain our enhanced backbone weights to fine-tune on an egocentric video task using data from $V_{ego}$.

### 3.3. Auxiliary egocentric tasks

Next, we describe each task we use, how we source $M^*$ and pseudo-labels $y^*$, the loss terms $\mathbb{L}_{\tau}$, and their relevance to egocentric feature learning. Note that no egocentric activity labels are used for the task models, and each task model is applied to third-person video instances in $V_{exo}$.

#### Ego-Score: Discriminating ego videos

A good egocentric video representation should be able to capture the underlying differences between first- and third-person videos, to discriminate between the two viewpoints. Based on this motivation, we design an Ego-Score task $\tau^{ego}$ to characterize the egocentricity likelihood of the video.

For this, we train a binary ego-classifier $M^{ego}$ on the Charades-Ego dataset [62], which has both egocentric and third-person videos of indoor activities involving object interactions. While the dataset offers paired instances showing the same activity from two views, our method does not use this pairing information or egocentric activity labels. It uses only the binary labels indicating if a sample is egocentric or exocentric. Please see Supp. for more training details and an ablation study about the pairing information.

We use this trained classifier to estimate the real-valued pseudo task-labels $y^{ego}$ for each video in our pre-training framework described in Sec 3.2. We sample multiple clips from the same video and average their score to generate a video-level label. Formally, for a video $v$ with $N$ clips $\{x_1, ..., x_N\}$ we generate scores:

$$y^{ego}_i(v) = \frac{\exp\left(\frac{1}{\beta} \sum_n z^{ego}_i(x_n)\right)}{\sum_j \exp\left(\frac{1}{\beta} \sum_n z^{ego}_j(x_n)\right)},$$

where $\beta$ is a scalar temperature parameter, $z^{ego}_i(x_n)$ is the predicted logits from the ego-classifier $M^{ego}$, and $i \in \{0, 1\}$ is the class label.

Third-person videos display various egocentric cues, resulting in a broad distribution of values for Ego-Score, despite sharing the same viewpoint (details in Supp). This score is used as the soft target in the auxiliary task loss, which we predict using a video classification head $H^{ego}$:

$$\mathbb{L}_{ego}(x) = -\sum_i y^{ego}_i(v) \log(H^{ego}_i(x)).$$

#### Object-Score: Finding interactive objects

In egocentric videos, interactions with objects are often central, as evident in popular egocentric video datasets [62, 12, 38]. Motivated by this, we designate an Object-Score task $\tau^{obj}$ for each video that encourages video representations to be predictive of manipulated objects.

Rather than require ground-truth object labels for third-person videos, we propose a simple solution that directly uses an off-the-shelf object recognition model $M^{obj}$ trained on ImageNet [14] to describe objects in the video. Formally, for a video $v$ with frames $\{f_1, ..., f_T\}$ we average the predicted logits from $M^{obj}$ across frames to generate the video-level Object-Score $y^{obj}_i(v)$:

$$y^{obj}_i(v) = \frac{\exp\left(\frac{1}{\beta} \sum_j z^{obj}_i(f_j)\right)}{\sum_j \exp\left(\frac{1}{\beta} \sum_j z^{obj}_j(f_j)\right)},$$

where $z^{obj}_i(f_j)$ is predicted logits for the $i^{th}$ class from the recognition model, and $\beta$ is the temperature parameter.

Similar to the Ego-Score, we introduce a knowledge-distillation loss during pre-training to make the video model predictive of the Object-Score using a module $H^{obj}$:

$$\mathbb{L}_{obj}(x) = -\sum_i y^{obj}_i(v) \log H^{obj}_i(x).$$

#### Interaction-Map: Discovering hand interaction regions

The Object-Score attempts to describe the interactive objects. Here we explicitly focus on the spatiotemporal regions of interactions in videos. Prior work shows it is possible to recognize a camera wearer’s actions by attending to only a small region around the gaze point [38], as gaze often focuses on hand-object manipulation. Motivated by this, we introduce an Interaction-Map task $\tau^{int}$ to learn features that are predictive of these important spatiotemporal hand-object interaction regions in videos.

We adopt an off-the-shelf hand-object detector [60] $M^{int}$ to detect hands and interacting objects. For each frame $f_i$ in a video, the hand detector predicts a set of bounding-box coordinates and associated confidence scores $B_i = \{(b^h, s^h)\}$ for detected hands. These bounding boxes are scaled to $h \times w$—the spatial dimensions of the video clip feature. We then generate a $t \times h \times w$ spatiotemporal
3.4. Ego-Exo training and evaluation

The three proposed ego-specific auxiliary tasks are combined together during the pre-training procedure to construct the final training loss:

\[ L(x) = L_{act}(x) + w_{ego} * L_{ego}(x) + w_{obj} * L_{obj}(x) + w_{int} * L_{int}(x), \]

where \( L_{act} \) is the standard cross-entropy loss for third-person action recognition, and \( w_{ego}, w_{obj} \) and \( w_{int} \) are the corresponding loss weights for the three auxiliary tasks, selected via cross-validation on downstream tasks.

Note that third-person video instances without hand-object interactions or salient interactive objects still contribute to the auxiliary loss terms, and are not ignored. Our distillation models approximate the responses of the pretrained egocentric models as soft-targets instead of hard labels, offering valuable information about perceived egocentric cues, whether positive or negative for the actual label.

Training with our auxiliary losses results in features that are more suitable for downstream egocentric tasks, but it does not modify the network architecture itself. Consequently, after pre-training, our model can be directly used as a drop-in replacement for traditional video encoders, and it can be applied to various egocentric video tasks.

4. Experiments

Datasets. Our experiments use the following datasets.

- **Kinetics-400** [34] is a popular third-person video dataset containing \( \sim 300k \) videos and spanning 400 human action classes. We use this dataset to pre-train all our models.
- **Charades-Ego** [62] has \( \sim 68k \) instances spanning 157 activity classes. Each instance is a pair of videos corresponding to the same activity, recorded in the first and third-person perspective. Our method does not require this pairing, and succeeds even if no pairs exist (Supp.).
- **EPIC-Kitchens** [12] is an egocentric video dataset with videos of non-scripted daily activities in kitchens. It contains 55 hours of videos consisting of 39k action segments, annotated for interactions spanning 352 objects and 125 verbs. **EPIC-Kitchens-100** [11] extends this to 100 hours and 90k action segments, and is currently the largest annotated egocentric video dataset.

Due to its large scale and diverse coverage of actions, Kinetics has widely been adopted as the standard dataset for pre-training both first- and third-person video models [27, 19, 35, 11]. EPIC-Kitchens and Charades-Ego are two large and challenging egocentric video datasets that are the subject of recent benchmarks and challenges.

Evaluation metrics. We pre-train all models on Kinetics, and fine-tune on Charades-Ego (first-person only) and EPIC-Kitchens for activity recognition. Following standard practice, we report mean average precision (mAP) for Charades-Ego and top-1 and top-5 accuracy for EPIC.

Implementation details. We build our Ego-Exo framework on top of PySlowFast [18] and use SlowFast [19] video models as backbones with 8 input frames and stride 8.
We use a Slow-only ResNet50 architecture for all ablation experiments, and a SlowFast ResNet50/101 architecture for final results.

For our distillation heads $H^{ego}$ and $H^{obj}$ (Sec 3.3), we use a spatiotemporal pooling layer, followed by a linear classifier. We implement our Interaction-Map heads $H^b$ and $H^0$ as two 3D conv layers with kernel sizes $1\times3\times3$ and $1\times1\times1$, and ReLU activation.

For our combined loss function (Eqn 7), we set the loss weights $w_{ego}$, $w_{obj}$ and $w_{int}$ to 0.1, 0.5, 1.0 respectively through cross-validation on the EPIC-Kitchens validation set (cross-validation on Charades-Ego suggested similar weights). The temperature parameter $\beta$ in Eqn 1 and Eqn 3 is set to 1. Training schedule and optimization details can be found in Supp.

### 4.1. Ego-Exo pre-training

We compare our pre-training strategy to these methods:

- **Scratch** does not benefit from any pre-training. It is randomly initialized and directly fine-tuned on the target egocentric dataset.

- **Third-only** is pre-trained for activity labels on Kinetics 400 [34]. This represents the status-quo pre-training strategy for current video models.

- **First-only** is pre-trained for verb/noun labels on EPIC-Kitchens-100 [11], the largest publicly available egocentric dataset.

- **Domain-adapt** introduces a domain adaptation loss derived from gradients of a classifier trained to distinguish between first- and third-person video instances [23]. This strategy has been used in recent work to learn domain invariant features for third-person vs. drone footage [10].

- **Joint-embed** uses paired first- and third-person video data from Charades-Ego to learn viewpoint-invariant video models via standard triplet embedding losses [61]. We first pre-train this model with Kinetics to ensure that the model benefits from large-scale pre-training.

- **Ego-Exo** is pre-trained on Kinetics-400, but additionally incorporates the three auxiliary egocentric tasks (Sec 3.3) together with the original action classification loss, to learn egocentric-specific features during pre-training.

For this experiment, all models share the same backbone architecture (Slow-only, ResNet-50) and only the pre-training strategy is varied to ensure fair comparisons. Domain Adapt uses additional unlabeled egocentric data during pre-training, but from the same target dataset that the model will have access to during fine-tuning. Joint-embed uses paired egocentric and third-person data, an advantage that the other methods do not have, but offers insight into performance in this setting. Only First-only has access to ego-videos labeled for actions during pre-training.

### 4.2. Ablation studies

#### Impact of auxiliary ego-tasks.

We next analyze the impact of each auxiliary egocentric task in our Ego-Exo framework. As shown in Table 2, adding the Ego-Score task improves performance on both EPIC-Kitchens tasks, while...
Figure 4: **Class-level performance on Charades-Ego.** Our method significantly improves on several classes that focus on active object manipulations (green lines), and performs only marginally worse across most under-performing classes (red lines). 35 most/least improved classes (out of all 157 classes) are shown.

### Table 3: Effect of Ego-Exo losses during fine-tuning.

Adding distillation losses during fine-tuning improves performance for both models, and results in a larger performance gain for our Ego-Exo pre-trained models. Values are averaged over 3 runs.

<table>
<thead>
<tr>
<th>Methods</th>
<th>C-Ego mAP</th>
<th>EPIC verbs top-1</th>
<th>EPIC verbs top-5</th>
<th>EPIC nouns top-1</th>
<th>EPIC nouns top-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Third-only</td>
<td>24.69</td>
<td>61.19</td>
<td>87.49</td>
<td>46.18</td>
<td>69.72</td>
</tr>
<tr>
<td>Third-only +aux</td>
<td>25.00</td>
<td>62.36</td>
<td>87.72</td>
<td>46.59</td>
<td>68.33</td>
</tr>
<tr>
<td>Δ</td>
<td>+0.31</td>
<td>+1.17</td>
<td>+0.23</td>
<td>+0.42</td>
<td>-1.39</td>
</tr>
<tr>
<td>Ego-Exo</td>
<td>26.23</td>
<td>62.83</td>
<td>87.63</td>
<td>48.15</td>
<td>70.28</td>
</tr>
<tr>
<td>Ego-Exo +aux</td>
<td>27.47</td>
<td>64.26</td>
<td>88.45</td>
<td>48.39</td>
<td>70.68</td>
</tr>
<tr>
<td>Δ</td>
<td>+1.24</td>
<td>+1.43</td>
<td>+0.82</td>
<td>+0.24</td>
<td>+0.40</td>
</tr>
</tbody>
</table>

Table 4: **Comparison to prior work on Charades-Ego.** Despite having no access to paired egocentric data, our model outperforms specialized joint-embedding and domain adaptation based methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActorObserverNet [61]</td>
<td>20.0</td>
</tr>
<tr>
<td>SSDA [10]</td>
<td>23.1</td>
</tr>
<tr>
<td>I3D [10]</td>
<td>25.8</td>
</tr>
<tr>
<td>SlowFast [19]</td>
<td>25.93</td>
</tr>
<tr>
<td>Ego-Exo</td>
<td>28.04</td>
</tr>
<tr>
<td>Ego-Exo*</td>
<td>29.19</td>
</tr>
<tr>
<td>Ego-Exo*-R101</td>
<td>30.13</td>
</tr>
</tbody>
</table>

Adding **Object-Score** and **Interaction-Map** consistently improves all results. This reveals that despite varying structure and semantics, these scores capture important underlying egocentric information to complement third-person pre-training, and further boost performance when used together.

Fig 5 shows instances from Kinetics based on our auxiliary pseudo-label scores combined with the weights in Eqn 7. Our score is highest for object-interaction heavy activities (e.g., top row: knitting, changing a tire), while it is low for videos of broader scene-level activities (e.g., bottom row: sporting events). Note that these videos are not in the egocentric viewpoint—they are largely third-person videos from static cameras, but are *ego-like* in that they prominently highlight important features of egocentric activity (e.g. hands, object interactions).

**Adding auxiliary ego-tasks during fine-tuning.** Our auxiliary losses may also be added after pre-training, for fine-tuning downstream egocentric models similar to prior semi-supervised learning work [9]. We re-introduce our Interaction-Map loss $L_{int}$ (Eqn 6) for downstream egocentric training. We do not include Ego-Score (which is trivially high for all videos) and Object-Score (which is subsumed in the interaction label for this setting) as their impact after pre-training was minimal.

Table 3 shows that while both the baseline and our method further improve by adding the auxiliary task during fine-tuning, our improvements (Ego-Exo + aux) are larger, especially on Charades-Ego. This is likely because our distillation heads benefit from training to detect hands and objects in large-scale third-person video prior to fine-tuning for the same task on downstream egocentric datasets.

### 4.3. Comparison with state-of-the-art

Finally, we compare our method with state-of-the-art models, many of which use additional modalities (flow, audio) compared to our RGB-only models. We include three competitive variants of our model using SlowFast [19] backbones: (1) **Ego-Exo** uses a ResNet50 backbone; (2) **Ego-Exo** additionally incorporates our auxiliary distillation loss during fine-tuning; (3) **Ego-Exo*-R101 further uses a ResNet-101 backbone.

---

1 Same as Ego-Exo + aux in Table 3, but here with a SlowFast backbone
semble schemes (e.g. the top ranking method ensembles 8 models).

Other reported results on the competition show large performance gains over prior work, including the competing methods, our method does not require any egocentric data which is paired or shared category labels with third-person data during pre-training.

**Charades-Ego.** Table 4 compares our Ego-Exo method with existing methods. Our Ego-Exo and Ego-Exo* yield state of the art accuracy, improving performance over the strongest baseline by +2.11% and +3.26% mAP. We observe large performance gains over prior work, including ActorObserverNet [61] and SSDA [10], which use joint-embedding or domain adaptation approaches to transfer third-person video features to the first-person domain. In addition, unlike the competing methods, our method does not require any egocentric data which is paired or shared category labels with third-person data during pre-training.

**EPIC-Kitchens.** Table 6 compares our method to state-of-the-art models on the EPIC-Kitchens test set. Ego-Exo and Ego-Exo* consistently improve over SlowFast (which shares the same backbone architecture) for all categories on both seen and unseen test sets. Epic-Fusion [35] uses additional optical flow and audio modalities together with RGB, yet Ego-Exo outperforms it on the top-1 metric for all categories. AVSlepFast [71] also utilizes audio, but is outperformed by our model with the same backbone (Ego-Exo*-R101) on the S1 test set.² On **EPIC-Kitchen-100** [11], as shown in Table 5, Ego-Exo consistently improves over the

---

"Table 5: Comparison on EPIC-Kitchens-100 action recognition test set. Our method is best in all categories.

<table>
<thead>
<tr>
<th>S1 (seen)</th>
<th>Methods</th>
<th>Overall top-1</th>
<th>Overall top-5</th>
<th>Unseen Participants top-1</th>
<th>Tail Classes top-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/ audio</td>
<td>Epic-Fusion [35]</td>
<td>62.40</td>
<td>46.50</td>
<td>35.11</td>
<td>88.74</td>
</tr>
<tr>
<td></td>
<td>TSN fusion [12]</td>
<td>58.43</td>
<td>46.54</td>
<td>32.79</td>
<td>87.27</td>
</tr>
<tr>
<td></td>
<td>TRN [22]</td>
<td>62.56</td>
<td>45.70</td>
<td>34.41</td>
<td>88.24</td>
</tr>
<tr>
<td></td>
<td>TSM [40]</td>
<td>65.51</td>
<td>48.48</td>
<td>37.58</td>
<td>89.39</td>
</tr>
<tr>
<td>w/o audio</td>
<td>SlowFast [19]</td>
<td>63.89</td>
<td>49.66</td>
<td>37.42</td>
<td>88.71</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo</td>
<td>64.08</td>
<td>50.59</td>
<td>38.38</td>
<td>88.57</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo*</td>
<td>65.02</td>
<td>51.74</td>
<td>39.52</td>
<td>89.26</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo*-R101</td>
<td>66.07</td>
<td>51.51</td>
<td>39.98</td>
<td>89.39</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S2 (unseen)</th>
<th>Methods</th>
<th>Overall top-1</th>
<th>Overall top-5</th>
<th>Unseen Participants top-1</th>
<th>Tail Classes top-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/ audio</td>
<td>Epic-Fusion [35]</td>
<td>64.75</td>
<td>49.70</td>
<td>39.70</td>
<td>87.27</td>
</tr>
<tr>
<td></td>
<td>AVSF-R101 [71]</td>
<td>65.70</td>
<td>51.50</td>
<td>39.51</td>
<td>89.39</td>
</tr>
<tr>
<td>w/o audio</td>
<td>SlowFast [19]</td>
<td>63.47</td>
<td>49.60</td>
<td>38.38</td>
<td>88.57</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo</td>
<td>65.84</td>
<td>49.89</td>
<td>39.62</td>
<td>89.64</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo*</td>
<td>66.19</td>
<td>50.11</td>
<td>39.98</td>
<td>89.74</td>
</tr>
<tr>
<td></td>
<td>Ego-Exo*-R101</td>
<td>65.97</td>
<td>50.32</td>
<td>40.12</td>
<td>89.74</td>
</tr>
</tbody>
</table>

**Table 6: Comparison to prior work on EPIC-Kitchens (test set).** Methods in gray use additional audio modality information. Our method outperforms all methods that use consistent modalities in both settings, and is competitive with models that benefit from audio stream inputs.

---

2Table 6 compares existing methods under a controlled setting: using a single model with RGB or RGB+audio as input, and only Kinetics/ImageNet for pre-training. Other reported results on the competition page may use extra modalities, larger pre-training datasets, or model ensemble schemes (e.g. the top ranking method ensembles 8 models).

---

"Figure 5: Kinetics instances sorted by Ego-Exo scores. Our task scores are maximum for videos that prominently feature hands/objects in view (top row), and minimum for scenes devoid of human-centered activity (bottom row)."

---

"5. Conclusion"

We proposed a novel method to embed key egocentric signals into the traditional third-person video pre-training pipeline, so that models could benefit from both the scale and diversity of third-person video datasets, and create strong video representations for downstream egocentric understanding tasks. Our experiments show the viability of our approach as a drop-in replacement for the standard Kinetics-pretrained video model, achieving state-of-the-art results on egocentric action recognition on Charades-Ego and EPIC-Kitchens-100. Future work could explore alternate distillation tasks and instance-specific distillation losses to maximize the impact of third-person data for training egocentric video models.

---

"Acknowledgments: UT Austin is supported in part by the NSF AI Institute and FB Cognitive Science Consortium."
References

[31] Yifei Huang, Minjie Cai, Zhengqiang Li, and Yoichi Sato. Predicting gaze in egocentric video by learning task-dependent attention transition. In ECCV, 2018. 2


Yong Jae Lee and Kristen Grauman. Predicting important objects for egocentric video summarization. IJCV, 2015.


Yin Li, Miao Liu, and James M Rehg. In the eye of the beholder: Joint learning of gaze and actions in first person video. In ECCV, 2018.


[75] Huangyue Yu, Minjie Cai, Yunfei Liu, and Feng Lu. What i see is what you see: Joint attention learning for first and third person video co-analysis. In ACM MM, 2019. 1, 2