This ICCV workshop 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.

Seeing the Unseen: Predicting the First-Person Camera Wearer's Location and Pose in Third-Person Scenes

Yangming Wen[1], Krishna Kumar Singh[3], Markham Anderson[1], Wei-Pang Jan[1], and Yong Jae Lee[1,2]

[1]University of California, Davis
[2]University of Wisconsin-Madison
[3]Adobe Research

Abstract

Our goal is to predict the camera wearer's location and pose in his/her environment based on what's captured by the camera wearer's first-person wearable camera. Toward this goal, we first collect a new dataset in which the camera wearer performs various activities (e.g., opening a fridge, reading a book) in different scenes with time-synchronized first-person and stationary third-person cameras. We then propose a novel deep network architecture, which takes as input the first-person video frames and empty third-person scene image (without the camera wearer) to predict the location and pose of the camera wearer. We explore and compare our approach with several intuitive baselines and show initial promising results on this novel, challenging problem.

1. Introduction

Consider Fig. 1 (left). We are given two sets of images: the first image is an indoor-scene taken from a stationary camera, and the second set of images is one taken from a first-person wearable camera within the same scene. As humans, we can easily imagine that the camera wearer, given the first person view, would be sitting on the chair in front of the laptop; Fig. 1 (right). The goal of this paper is to create a model with this capability; i.e., predict the possible location and pose of the camera-wearer based on the first-person view and corresponding empty (without the camera-wearer) third-person scene image.

While there have been prior work [28, 70] which predict the pose of the camera-wearer using only first-person videos, we argue that it is important to contextualize the pose conditioned on the environment. Specifically, the scene that the camera wearer is operating in, and the objects that make up the scene, constrain the types of actions

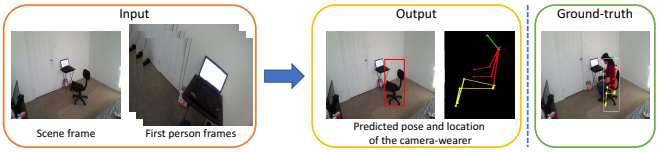

Figure 1. Our model takes first-person frames and a third-person scene frame as input and predicts the pose and location of the camera wearer capturing the first-person video. For example, by looking at the laptop we can predict that the camera wearer is in the sitting position on the chair next to the laptop.

(poses) that the camera wearer can have. This idea of *affordances* [23] itself is not new, but predicting the specific location and pose of the camera wearer in the scene based on the first-person view has not been studied previously. Fan et al. [16] use synchronized first and third-person views to identify the camera wearer in the third-person frame. But in their case, the camera wearer is visible in the third-person frame whereas in our setting the third-person frame does not contain the camera wearer.

We envision several real-world applications of our problem setting, particularly for law enforcement settings where only body-mounted cameras are available. In such cases, being able to infer the pose and location of the camera wearer with respect to a particular scene (whose image can be taken separately) can be critical for surveillance and monitoring purposes. Similarly, augmented and virtual reality (AR/VR) applications can be facilitated by knowing the location and pose of the camera wearer in the scene. For example, we could use the first-person view of the camera on AR glasses and third-person scene image to create a "Smart Room" application, in which the inferred camera wearer's location and pose in the room can be used to determine e.g., which light to be turned on, adjust the temperature, or notification to be shown.

Main idea. Our key idea is to learn *semantic corre-*

spondences between what's captured in the first person sequence of frames (i.e., the movement of the camera-wearer plus what s/he sees) and the objects present in the third-person scene. During training, we have access to synchronized first-person and third-person videos (where the camera wearer is visible in the third person view). The third-person videos provide the ground-truth pose and location of the camera wearer in the scene to train our model. Our model takes first-person frames and a third-person scene image as input, learns visual and contextual similarities between the two sets of images, and predicts the location and pose of the camera wearer in the scene. Our setting is practical for real-world settings as it is convenient to obtain an empty scene image; e.g., for a home AR/VR application, we can take a picture of the room-of-interest once, and using the first-person video, we can predict the pose and location of the person in the room.

Although this problem can be ill-posed in some cases e.g., the third-person scene may only capture the back of a laptop whereas the first-person view may capture the front, the model can still succeed if the correspondences learned between the views are at a semantic level (e.g., that the laptop from both views are the same by taking cues from surrounding objects, as well as by learning what a laptop looks like from any viewpoint). Similarly, if the camera wearer looks at something that is not present in the scene e.g., s/he comes into the scene with a laptop, the model can still succeed by learning that laptops are usually used sitting down on a desk or couch.

Contributions. Our work has three main contributions. 1) We propose the new problem of predicting the location and pose of the camera wearer in a third-person scene simultaneously. To the best of our knowledge, we are the first to tackle this problem. 2) We propose a novel deep network architecture which learns semantic correspondences between the first and third-person views to perform this task. Our quantitative and qualitative results show better performance compared to meaningful baselines.

2. Related Work

First-person (egocentric) vision. Analyzing images captured from a wearable camera has been widely studied for video summarization [35, 41, 62, 68, 43, 6], human action recognition/prediction [18, 19, 11, 37, 2, 56, 42, 53], future motion prediction and planning [55, 57, 46, 65, 3], saliency [58, 4, 5, 49, 71, 22, 8, 52]. Our work goes beyond the first-person view to find semantic correspondences in the third-person scene to predict the camera wearer's location and pose.

Particularly relevant are works that estimate 3D human body pose from first-person cameras [28, 69]. Unlike these approaches, our work incorporates environmental context from the third-person scene, and further requires the localization of the camera wearer in addition to estimating his/her pose.

Visual odometry estimates the location and orientation of an agent based on the images captured by mounted cameras. Most work are feature-based, which extract features from keypoints of the image [32, 30, 7, 59], appearance-based, which compare images directly [50, 24, 9], or a hybrid of the two [20, 48, 40, 51]. Our task is different in that it requires predicting the location and pose of a human camera wearer, which can be much more complex. In addition, our model has the potential to predict the location and pose of the camera wearer in a scene that is related but different from the exact scene in which the camera wearer was in (e.g., predicting the possible location/pose in a different kitchen scene). This is in contrast to the conventional visual odometry/SLAM [13] setting where the localization must be done in the same scene.

Visual affordances and person location prediction. The study of visual affordances [23], i.e., how an object can be used, is also related. Previous work utilize videos/images from the third-person view to learn human affordances for scene layout understanding [26, 21], human action/pose prediction [64, 61, 66, 33, 29, 34, 47], and object understanding [25, 31, 12, 44]. Often these methods require the human to be visible in the input. Also, there has been work [60, 36, 72] to directly predict the location of the person in the scene with the correct pose. However, these predictions are generic for the entire scene as they are made independent of current first-person view. In contrast, in our setting, the model never sees the camera wearer in the third-person view, and focus on the novel task of predicting the location and pose of the camera wearer in the third-person view based on the first-person view.

Joint first person and third person visual learning. This area has been studied for view synthesis [14], learning shared visual representations [54], and video summarization [15]. Others identify the camera wearer location in the third-person view by combining motion and appearance cues from only the first person view [1, 16, 63, 67]. The localization in our task is much more challenging because the camera wearer is never visible in the third person view. To our knowledge, our proposed task of simultaneous localization of the camera wearer in the third-person frame and predicting the human pose has never been done in this setting.

3. Approach

Our goal is to learn a model that can accurately infer the camera wearer's location and pose within a scene, given a first-person view of the camera wearer. We supply two

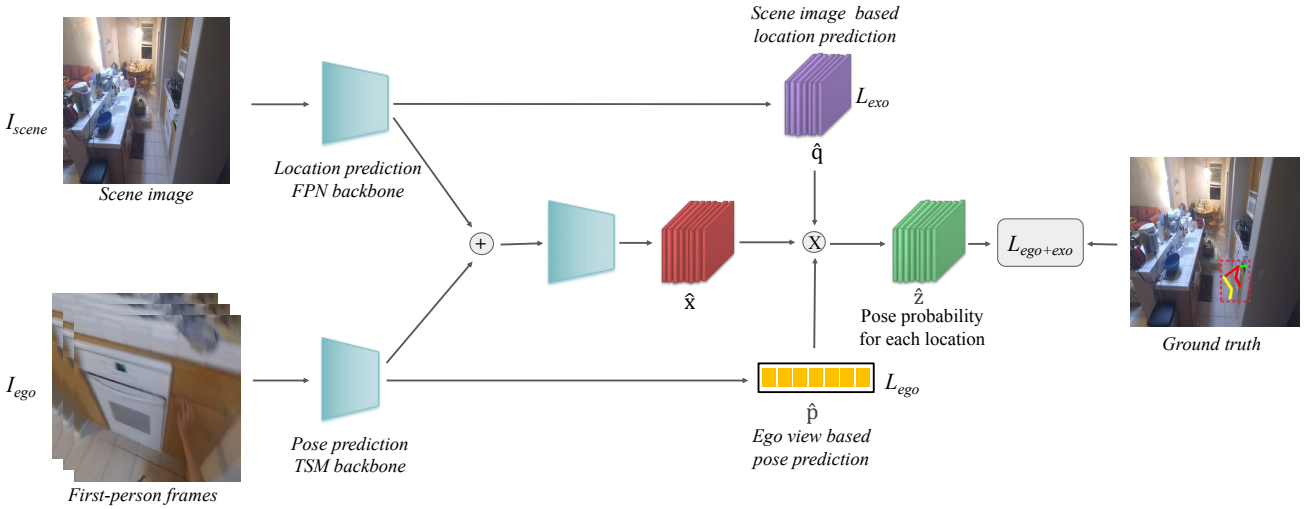

Figure 2. **Model architecture.** For the task of pose and location prediction, our approach takes an input consisting of a sequence of first-person frames (i.e. as seen by the camera wearer) I_{ego}, and a third-person scene image (i.e. an image of the scene where I_{ego} was taken without the camera wearer) I_{scene}. The scene image based location prediction branch learns the semantics of the input scene for predicting the possible locations for each pose. A sequence of I_{ego} frames are fed into the TSM model [38] to perform pose classification. Finally, we concatenate the feature maps from these two branches, and jointly process them to infer the specific pose and location of the camera wearer within I_{scene}.

inputs to the model: (1) the aforementioned first-person frames, hereafter termed I_{ego}, and (2) an image of a scene containing no human, hence named I_{scene}. During training, we have access to the synchronized first-person and third-person videos as well as I_{scene} image where these videos were taken. Since the camera wearer is visible in the third-person video, we can obtain the exact pose and location of the camera wearer in the I_{scene} corresponding to each I_{ego} taken from the first-person video. Using this information as ground-truth, we can train a network which takes semantic features of the I_{scene} and I_{ego} pair as input to predict the camera-wearer's pose and location in I_{scene}. During training, the network will learn correspondences between I_{ego} and I_{scene} to make these predictions.

3.1. Network architecture

Fig. 2 shows our network architecture. The location prediction module (top) takes as input the scene image I_{scene} while the pose prediction module (bottom) takes as input the first-person frames I_{ego}. Their processed features are concatenated and jointly processed (middle) to produce pose probabilities for each location in the scene. Intuitively, the model must see both the scene image and the first-person frame features together in order to learn semantic correspondences to determine the precise location and pose of the camera wearer.

In order to ensure that the location prediction module focuses on location prediction based on I_{scene}, and the pose prediction module focuses on pose prediction based on I_{ego}, we perform multi-task learning. Specifically, we force the location prediction module to predict all observed (ground-truth) locations of the camera wearer, independent of I_{ego}, while we force the pose prediction module to predict the corresponding ground-truth pose, independent of I_{scene}. In this way, we can prevent the model from taking an undesirable shortcut, e.g., by using only the scene features to predict both location and pose. Ultimately, the I_{scene} features should encode that e.g., the locations near the chair and couch are more likely to contain a person in a sitting pose, independent of what's seen in the first-person frames. Similarly, I_{ego} features should encode that e.g., the camera wearer is in the sitting pose if the I_{ego} frames have downward motion and have a laptop in it, independent of what's seen in the third-person scene image. These features can then be combined to encode the specific location and pose of the camera wearer. In the ensuing sections, we describe each of the modules in more detail.

3.2. Third-person scene image based location prediction

As described earlier, the goal of this stage is to make the features corresponding to I_{scene} capture location information for each pose. To this end, we train the network to predict all locations visited by the camera wearer in I_{scene} by only taking I_{scene} features as input. During training, we have access to the third person video corresponding to I_{scene}. Hence, we know all the ground-truth locations and poses of the camera wearer within I_{scene} to train this module. Once trained, I_{scene} features will contain location specific information for each pose. For example, the I_{scene}

features would encode that the person is more likely to be closer to the chair when she is in a sitting pose whereas for the standing pose there is a higher chance that she is on the floor region.

In order to train this module, we first divide I_{scene} into a $G \times G$ grid. Then, we create a ground-truth label map of size $M \times G \times G$, where M is the total number of poses. The ground-truth label p_i^c for grid position i is set to 1 if the camera wearer visits location i with pose c, otherwise it is set to 0. For location prediction, we provide I_{scene} as input to a fully convolutional network to get M binary predictions of size $G \times G$ for each pose. We use binary predictions since the camera wearer could have been at the same location in multiple poses (at different times), and also because she may not have visited a particular location at all. Specifically, we train using the binary cross-entropy loss, one for each pose, between the true label p_i^c and prediction $\hat{p}_i^c$:

$$\mathcal{L}_{scene} = -\sum_{i=1}^{G\times G}\sum_{c=1}^{M} p_i^c \log(\hat{p}_i^c) + (1-p_i^c)\log(1-\hat{p}_i^c). \tag{1}$$

After training, this module can predict all possible locations for each pose given a scene image. However, it cannot provide the exact location of the camera wearer as it was not conditioned on I_{ego}. For example, it can predict chairs and sofas as possible locations for the sitting pose in the scene, but it needs to see the content of I_{ego} to know the exact chair or sofa on which the camera wearer is sitting. Hence, these scene location features will need to be combined with first-person pose features, which we describe later.

3.3. First-person view based pose recognition

Next, we train the features corresponding to the first-person frames I_{ego} to capture pose information. We use Temporal Shift Module (TSM) [38] as the backbone for this module, which takes a sequence of first-person frames I_{ego} as input and uses both the temporal and visual content of the frames to predict the pose. For example, if there is a fast downward motion and the floor region is visible, then the person is more likely to be in the bending pose. We train this module using the cross-entropy loss between the predicted pose probability $\hat{q}$ and ground-truth pose label q:

$$\mathcal{L}_{ego} = -\sum_{c=1}^{M} q^c \log(\hat{q}^c), \tag{2}$$

where M is the total number of poses. We obtain the ground-truth pose q using the corresponding third-person video in which the camera wearer is visible. We cluster the human poses into M canonical poses (like sitting, bending, and standing) and treat pose prediction as a M-way classification problem. More details are available in the implementation details.

3.4. Joint location and pose prediction

Thus far, I_{scene} and I_{ego} were fed in isolation to predict the location and pose probabilities respectively. However, for the final location and pose prediction of the camera wearer, it is critical that the network see both I_{scene} and I_{ego} features together. By seeing only I_{scene}, the network can learn all the possible locations a person can be present in for a given pose, but it cannot know the exact location unless it also sees I_{ego}. Similarly, the network can infer pose by looking only at I_{ego}, but it can be even more confident in its pose prediction by also seeing I_{scene}. For example, if I_{ego} contains a book on a table, it is very likely that the camera wearer is in a sitting pose. But if I_{scene} shows that there is a chair near the table with the book, then it can be even more confident that the person is in the sitting pose.

To model this, we concatenate the location features computed from the scene image I_{scene} with the pose features computed from the first-person frames I_{ego}, and feed the combined feature into a joint prediction module to predict the location and pose of the camera wearer conditioned on both inputs; see Figure 2. More specifically, the prediction of this joint module is of size $(M+1) \times G \times G$, where $\hat{x}_i^c$ indicates the probability of the camera wearer being in pose c at grid location i. The $M+1$'th pose class indicates the background class (camera wearer not being present). We find that modulating this joint probability with the scene based location probability and first-person pose probability to be helpful, as they can serve as additional constraints and attention to guide the final prediction. Therefore, the final prediction $\hat{z}_i^c$ for each grid location and pose is:[1]

$$\hat{z}_i^c = \hat{p}_i^c * \hat{q}^c * \hat{x}_i^c. \tag{3}$$

We train this joint module by applying the cross-entropy loss between the prediction $\hat{z}_i^c$ and ground-truth mask z_i^c at each grid location:

$$\mathcal{L}_{joint} = -\sum_{i=1}^{G\times G}\sum_{c=1}^{M+1} z_i^c \log(\hat{z}_i^c), \tag{4}$$

where the ground-truth z_i^c is 1 if the camera wearer was present at location i in pose c (which can be determined based on the corresponding third person video), otherwise for all other locations the background class is 1; i.e., there is only one ground-truth pose and location for a given I_{scene} and I_{ego} pair.

[1] As we do not have the location and pose prediction corresponding to the background class for $\hat{p}_i$ and $\hat{q}$, we simply set their background class probabilities to 1.

Our final loss combines all three losses:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{scene} + \lambda_2 \mathcal{L}_{ego} + \lambda_3 \mathcal{L}_{joint}, \quad (5)$$

where $\lambda_1 = 1$, $\lambda_2 = 1$, and $\lambda_3 = 1$.

During testing, we provide first-person frames (I_{ego}) and scene image (I_{scene}) as input to our model, which predicts the pose and location of the camera-wearer in the scene.

4. Experiments

In this section, we first describe our new dataset, implementation details, and evaluation metrics. We then discuss quantitative and qualitative results.

4.1. Synchronized First and Third Person Video Dataset

We collected a new real-world video dataset by hiring workers through Amazon Mechanic Turk (AMT). Turkers recorded a first-person video and a third-person video simultaneously. In each case, the worker used a stationary camera to record the third-person scene video and wore a head-mounted camera to record first-person video. We asked the workers to clap at the beginning, which we used to synchronize the two videos.

During each pair of recordings, the worker changed their location and pose several times in accordance with one of the scripts from Charades-Ego [54]. Recordings lasted approximately one minute. We selected 162 scripts from Charades-Ego, each set in one of 6 types of indoor scenes: bedroom, kitchen, laundry room, living room, home office, and dining room. We collected a total of 235 videos with 80 different environments (different indoor rooms). For each third-person frame, we use AlphaPose [17] to automatically identify the cameara wearer's location and pose. Subsequently, we exclude frames that contained no human figures or human figures whose pose AlphaPose identified with only poor confidence.

To create pose annotations for our dataset, we cluster the pose skeletons from our training set into 7 clusters. Out of 7 clusters, 4 clusters belong to sitting pose in different orientations, 2 clusters belong to bending in different orientations, and 1 cluster belongs to standing. For the third-person view, we know the orientation of the pose (e.g., left-facing or right-facing). However, for the first-person view, it is hard to know the orientation; we therefore only predict the pose probability for sitting, bending, and standing without considering the orientation, and repeat the probabilities across the different orientations to obtain values for all 7 poses. We calculate the bounding boxes of these 2D skeletons obtained using AlphaPose, and then use its center for the location ground-truth. We divide the scene image I_{scene} into a grid of size 13×13, and the grid location in which a person's center lies is the ground-truth.

Methods	Pose-balanced		Location-balanced	
	1×GT box	2×GT box	1×GT box	2×GT box
Random	5.25%	24.26%	8.46%	25.91%
cGaus (μ:6,σ^2:4)	8.90%	27.94%	11.21%	30.48%
cGaus (μ:6,σ^2:2)	16.61%	43.59%	13.09%	36.29%
Saliency	6.76%	34.76%	15.47%	43.34%
Ours-SceneOnly	18.74%	41.03%	10.86%	34.40%
Ours	**23.67%**	**53.57%**	**19.10%**	**47.08%**

Table 1. Localization accuracy. Our approach outperforms the baselines for camera wearer localization.

4.2. Implementation details

Evaluation split. From our dataset, we use 187 videos for training and the remaining 48 videos for testing. We ensure that the environments of the test images are distinct from those that appear in training images, though the 6 scene categories are common to both.

Training details. We first train the scene image based location prediction module and first-person view based pose prediction module, and then combine them together to train the final location and pose prediction module. We find this leads to more stable training. For the first-person pose prediction branch, we use TSM [38] pre-trained on the Epic-kitchen dataset [10]. For I_{ego}, we use a sequence of 64 frames which is around 2 seconds. For the scene image location prediction branch, we load the backbone of MaskR-CNN [27] pre-trained on MS COCO [39]. We have $M = 7$ poses and $G = 13$ grid resolution. For the $\mathcal{L}_{ego}$ and $\mathcal{L}_{joint}$ losses, we balance the weights of each class according to their frequency. We use SGD with 0.0001 and 0.9 as learning rate and momentum, respectively. The network is trained for 6 epochs with a batch size of 8. We also apply random horizontal flipping and crop as data augmentation.

4.3. Pose prediction results

We first evaluate pose prediction on novel test scenes not seen during training. This is a really challenging task as we need to infer the pose from first-person view and scene image without actually seeing the person. For our prediction, we choose the pose with maximum probability across all 13×13 grid locations of our final branch. We obtain accuracy of 46.52% accuracy for the pose prediction which is significantly better than random prediction over three poses (33.33%).

4.4. Location prediction results

Next, we evaluate the localization task on novel test scenes not seen during training. We consider the localization task to be successful if the model's predicted center location falls within the ground-truth bounding box of the person. Given the difficulty of this task, we also evaluate with the ground-truth box increased by 2x. Also, since our test data is not uniformly distributed across pose and loca-

tion, we average the test accuracies for each pose (Pose-balanced) and 13×13 grid locations (Location-balanced).

For our model's prediction, we choose the location which has the highest pose probability across all the poses in our final branch conditioned on both I_{ego} and I_{scene}. We evaluate our model's performance against the following baselines.

- **Random prediction (*Random*)** We randomly sample a coordinate in the image space (13×13) as our prediction.
- **Predicting center gaussian (*cGaus*)** We sample an image coordinate from a Gaussian distribution centered at the image center with variance 2 and 4 in both x and y coordinates (assuming image size is 13×13).
- **Saliency-based location prediction (*Saliency*)** In order to demonstrate the difference between our task and saliency prediction, we use a popular saliency predictor [45] to perform location prediction. We consider the location with the highest saliency value as the predicted location of the camera wearer. This baseline, unlike our approach, does not rely on I_{ego}, and always predicts the same location for a given I_{scene}.
- **Location-branch-Only:** In this baseline, we only train the third-person scene based location prediction branch and choose the location with the highest probability across all poses as the prediction. This is similar to the *Saliency* baseline, but the key difference is that *Saliency* is a generic saliency detector whereas this baseline is trained on our data for location prediction.

Table 1 shows the results. Our approach yields the best localization performance, outperforming the best baseline by 4.93% in the pose-balanced evaluation metric for 1x ground-truth box. The center Gaussian baseline (*cGaus*) outperforms random prediction (*Random*) as there is a center bias; i.e., the camera wearer tends to be near the center of the scene for multiple different activities. The saliency baseline (*Saliency*) can only identify generic salient regions in the scene like a chair, couch, and bed without considering where humans can be (i.e., affordances), and thus performs poorly. Finally, the *Location-branch-Only* baseline performs worse than our full approach. This baseline cannot specify the exact location of the camera wearer as it only relies on the scene image and does not use the first-person view I_{ego}, and hence does not know where the camera wearer is looking in the scene.

4.5. Joint pose and location prediction results

Finally, we evaluate joint pose and location prediction. Since this is a new problem with no existing methods, we compare to different variants of our model:

Methods	Pose-balanced		Location-balanced	
	1×GT box	2×GT box	1×GT box	2×GT box
Ours-SingleBranch	10.64%	20.24%	6.88%	14.81%
Ours-NoModulation	11.16%	19.26%	8.02%	13.27%
Ours	**12.87%**	**26.89%**	**11.34%**	**22.99%**

Table 2. Joint pose and location prediction. Our full approach produces the best results, which demonstrates the importance and complementarity of each component.

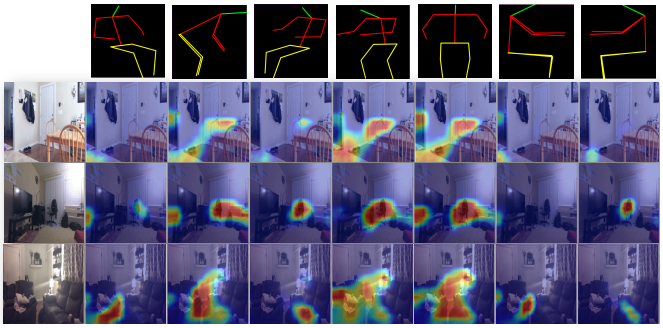

Figure 3. Location prediction for each pose based only on the scene image I_{scene}. In each column, we show the predicted locations corresponding to each pose. We can see that for poses associated with sitting, regions like chair and sofa get highlighted, whereas for standing, the floor region gets highlighted.

- **Ours-SingleBranch:** We only apply $\mathcal{L}_{joint}$, and remove $\mathcal{L}_{scene}$ and $\mathcal{L}_{ego}$. In other words, we train a single branch which takes I_{scene} and I_{ego} as input.
- **Ours-NoModulation:** We also train a version in which we do not modulate the final location and pose probabilities with the probabilities of the location and pose branches; i.e use $\hat{x}$ instead $\hat{z}$ for $\mathcal{L}_{joint}$. We still have separate modules for scene based location prediction and first-person frames based pose prediction, but we do not use their output to constraint our final location and pose prediction.

Table 2 shows the results. For a prediction to be correct, it has to match both pose and location according to our previous location and pose evaluation criteria. The *Ours-SingleBranch* baseline performs worse than our full model, which shows the importance of multi-task learning and training location prediction module only conditioned on I_{scene} and pose prediction module conditioned only on I_{ego}, as described in Section 3.1. The *Ours-NoModulation* baseline also performs worse, which shows the importance of using the output of the other two branches as to guide the final prediction. Our full model gives the best performance compared to the baselines showing the importance and complementarity of each component of our approach.

In addition, we analyze the variation in location prediction. We find the average standard deviation of the x and y coordinates for each test video to be 0.94 and 0.64 for *Ours-SingleBranch* and *Ours-NoModulation*, respectively.

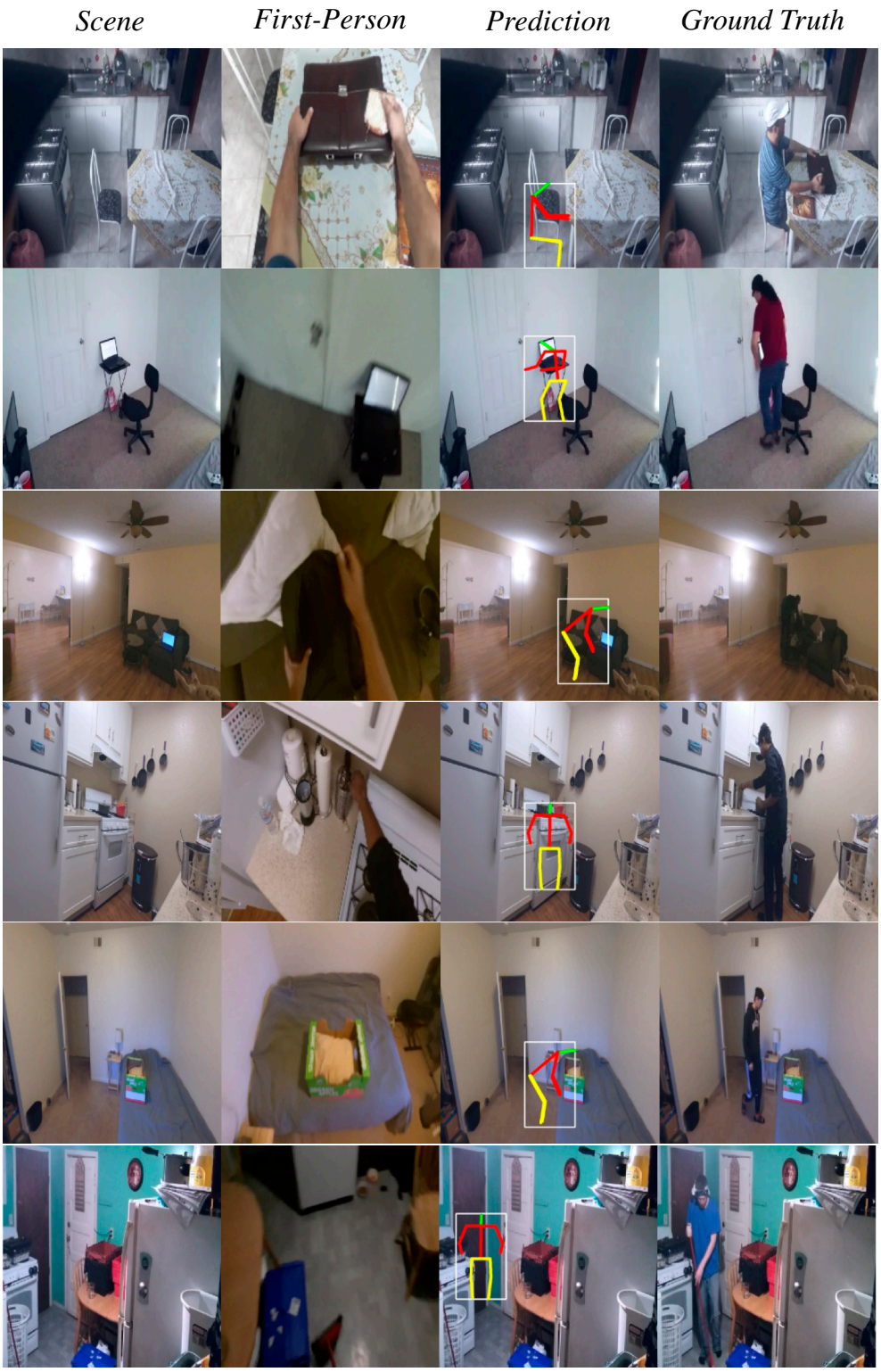

Figure 4. Successful Predictions. In each row, we show the third person scene image (I_{scene}), center frame of the first-person frame sequence (I_{ego}), our model's location and pose prediction, and ground-truth showing actual camera wearer in the scene.

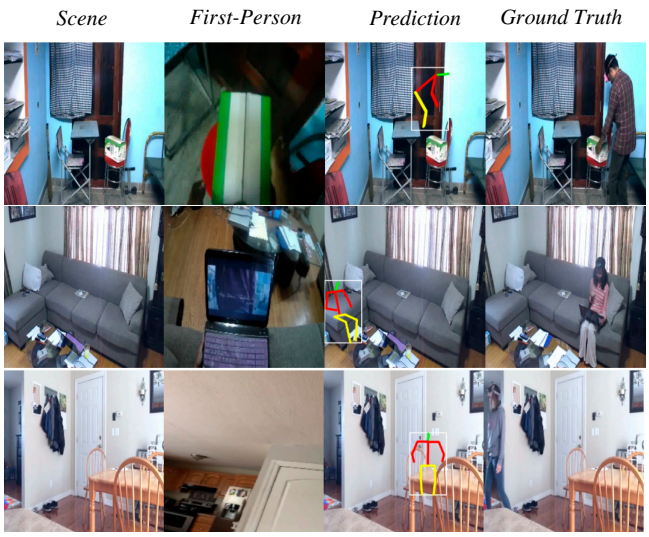

Figure 5. Failure cases. We show cases where our model fails to predict correct location and pose for the camera wearer.

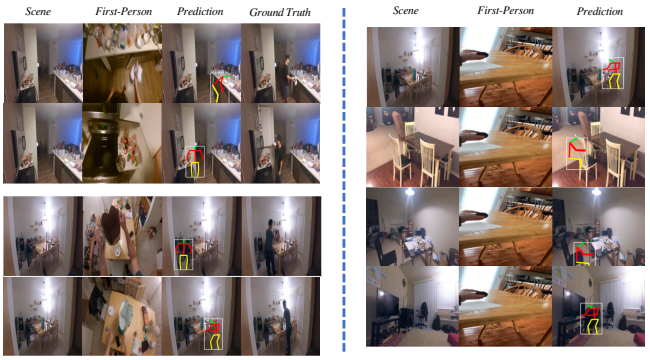

Figure 6. **Left:** Prediction results when the third-person scene image remains the same but corresponding first-person frames change. Given the same third-person scene image, our model's predictions change when the first-person frames change. **Right:** Prediction results when the first-person image remains the same but corresponding third-person scene image changes. Our approach can sometimes predict reasonable pose and location even when the first-person image is not taken in the scene image.

Our approach has a higher standard deviation of 1.11, which indicates that it is less biased towards a single location and has more variation in its prediction.

4.6. Qualitative results

First, in Fig. 3, we show the location predictions corresponding to each pose using our model when conditioned only on I_{scene} (we draw a bounding box of fixed size 2×3 around the predicted center in the 13×13 image grid). We can see our model makes meaningful predictions. For example, in the second row, the chair region gets highlighted for the sitting pose with the correct orientation which matches the chair's orientation. Whereas for the standing pose, the floor region gets highlighted in all three rows.

Next, in Fig. 4, we show successful pose + location predictions of our model. The first two columns show the two inputs to the network: the scene image I_{scene} and its corresponding first-person I_{ego} frame (the center frame from the frame sequence is shown). In the third column, we show our predicted location and pose of the camera wearer. In the last column, we show the ground-truth with the camera wearer in the scene image. In the first row, our model predicts that the person would be in a sitting pose on the chair, likely by using the table in I_{ego} as a cue. In the fifth row, interestingly, the model correctly predicts that the person is bending near the bed, likely because there is a box in I_{ego} on the bed and the model is able to find the corresponding box in the scene image. In Fig. 5, we show common failure cases of our approach. In the last row, the direction of the camera wearer's gaze puts the content of I_{ego} completely outside of I_{scene}, therefore our model gets confused about the pose and location. In the second last row, our model gets confused about the sitting location as there are multiple possible locations for the sitting.

We also show our prediction results for multiple instances of I_{ego} corresponding to the same I_{scene} in Fig. 6 (left). In the first example, when the camera wearer in I_{ego} is looking at the right side of the kitchen at an angle, our approach predicted that the person was in a bending pose on the right side. When I_{ego} changes to the other side of the kitchen, our prediction changes accordingly. These results show that our approach takes I_{ego}'s first-person information into consideration in its predictions. Finally, in Fig. 6 (right), we show predictions on semantically-related third-person scene images (I_{scene}) which do not correspond to I_{ego}; i.e., the I_{ego} was not taken in the I_{scene}. Even without having direct correspondence between I_{ego} and I_{scene}, our model can sometimes make reasonable predictions. In these four cases, it predicts the person to be close to the table as the table is visible in the I_{ego} image. In the second and third rows, our model predicts that the person is sitting on the chair close to the table, which is reasonable given the I_{ego} image.

5. Conclusion

In this paper, we presented the novel task of predicting the camera wearer's location in the third-person scene image by looking at the first-person frames. We collected a new dataset for this task, and proposed an intuitive network architecture. We obtained initial promising results, but admittedly, failures are common as this is a very challenging task. Nonetheless, we hope that this work will motivate other researchers to pursue research in this direction.

Acknowledgement This work was supported in part by NSF IIS-1812850.

References

[1] Shervin Ardeshir and Ali Borji. Ego2top: Matching viewers in egocentric and top-view videos. In *ECCV*, 2016.

[2] Sven Bambach, Stefan Lee, David J. Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. *ICCV*, 2015.

[3] Gedas Bertasius, Aaron Chan, and Jianbo Shi. Egocentric basketball motion planning from a single first-person image. *CVPR*, 2018.

[4] Gedas Bertasius, Hyun Soo Park, Stella X. Yu, and Jianbo Shi. First-person action-object detection with egonet. *ArXiv*, 2017.

[5] Gedas Bertasius, Hyun Soo Park, Stella X. Yu, and Jianbo Shi. Unsupervised learning of important objects from first-person videos. *ICCV*, 2017.

[6] Xiaojun Chang, Yaoliang Yu, Yi Yang, and Eric P. Xing. Semantic pooling for complex event analysis in untrimmed videos. *TPAMI*, 2017.

[7] Hsiang-Jen Chien, Chen-Chi Chuang, Chia-Yen Chen, and Reinhard Klette. When to use what feature? sift, surf, orb, or a-kaze features for monocular visual odometry. *IVCNZ*, 2016.

[8] Eunji Chong, Nataniel Ruiz, Yongxin Wang, Yun Zhang, Agata Rozga, and James M. Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency. In *ECCV*, 2018.

[9] Gabriele Costante and Thomas Alessandro Ciarfuglia. Ls-vo: Learning dense optical subspace for robust visual odometry estimation. *IEEE Robotics and Automation Letters*, 2018.

[10] Dima Damen and et al. Scaling egocentric vision: The epic-kitchens dataset. *ECCV*, 2018.

[11] Dima Damen, Teesid Leelasawassuk, Osian Haines, Andrew Calway, and Walterio W. Mayol-Cuevas. You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video. In *BMVC*, 2014.

[12] Vincent Delaitre, David F. Fouhey, Ivan Laptev, Josef Sivic, Abhinav Gupta, and Alexei A. Efros. Scene semantics from long-term observation of people. In *ECCV*, 2012.

[13] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 2006.

[14] Mohamed Elfeki, Krishna Regmi, Shervin Ardeshir, and Ali Borji. From third person to first person: Dataset and baselines for synthesis and retrieval. *ArXiv*, 2018.

[15] Mohamed Elfeki, Aidean Sharghi, Srikrishna Karanam, Ziyan Wu, and Ali Borji. Multi-view egocentric video summarization. *ArXiv*, 2018.

[16] Chenyou Fan, Jangwon Lee, Mingze Xu, Krishna Kumar Singh, Yong Jae Lee, David J. Crandall, and Michael S. Ryoo. Identifying first-person camera wearers in third-person videos. *CVPR*, 2017.

[17] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *ICCV*, 2017.

[18] Alireza Fathi, Ali Farhadi, and James M. Rehg. Understanding egocentric activities. *ICCV*, 2011.

[19] Alireza Fathi, Yin Li, and James M. Rehg. Learning to recognize daily actions using gaze. In *ECCV*, 2012.

[20] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. Svo: Fast semi-direct monocular visual odometry. *ICRA*, 2014.

[21] David F. Fouhey, Vincent Delaitre, Abhinav Gupta, Alexei A. Efros, Ivan Laptev, and Josef Sivic. People watching: Human actions as a cue for single view geometry. *IJCV*, 2014.

[22] David F. Fouhey, Weicheng Kuo, Alexei A. Efros, and Jitendra Malik. From lifestyle vlogs to everyday interactions. *CVPR*, 2018.

[23] James J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, 1979.

[24] Ramon Gonzalez, Francisco Rodriguez, Jose Luis Guzman, Cedric Pradalier, and Roland Siegwart. Combined visual odometry and visual compass for off-road mobile robots localization. *Robotica*, 2012.

[25] Helmut Grabner, Juergen Gall, and Luc Van Gool. What makes a chair a chair? *CVPR 2011*, 2011.

[26] Abhinav Gupta, Scott Satkin, Alexei A. Efros, and Martial Hebert. From 3d scene geometry to human workspace. *CVPR*, 2011.

[27] Kaiming He, Gkioxari Georgia, Dollár Piotr, and Girshick Ross. Mask r-cnn. *ICCV*, 2017.

[28] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. *CVPR*, 2017.

[29] Ying Jiang, Hema Swetha Koppula, and Ashutosh Saxena. Hallucinated humans as the hidden context for labeling 3d scenes. *CVPR*, 2013.

[30] Alex Kendall, Matthew Koichi Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. *ICCV*, 2015.

[31] Hedvig Kjellstr, Javier Romero, and Danica Kragic. Visual object-action recognition: Inferring object affordances from human demonstration. *CVIU*, 2011.

[32] Kishore Reddy Konda and Roland Memisevic. Learning visual odometry with a convolutional network. In *VISAPP*, 2015.

[33] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from rgb-d videos. *I. J. Robotics Res.*, 2013.

[34] Hema Swetha Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *TPAMI*, 2016.

[35] Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012.

[36] Xueting Li, Sifei Liu, Kihwan Kim, Xiaolong Wang, Ming-Hsuan Yang, and Jan Kautz. Putting humans in a scene: Learning affordance in 3d indoor environments. In *CVPR*, 2019.

[37] Yin Li, Zhefan Ye, and James M. Rehg. Delving into egocentric actions. 2015.

[38] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. *CVPR*, 2019.

[39] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Lawrence C. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[40] Yuzhe Lin, Zhaoxiang Liu, Jianfeng Huang, Chaopeng Wang, Guoguang Du, Jinqiang Bai, Shiguo Lian, and Bill Huang. Deep global-relative networks for end-to-end 6-dof visual localization and odometry. *ArXiv*, 2018.

[41] Zheng Lu and Kristen Grauman. Story-driven summarization for egocentric video. *CVPR*, 2013.

[42] Minghuang Ma, Haoqi Fan, and Kris M. Kitani. Going deeper into first-person activity recognition. *CVPR*, 2016.

[43] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. *CVPR*, 2017.

[44] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. *ArXiv*, 2018.

[45] Junting Pan, Cristian Canton Ferrer, McGuinness Kevin, Noel E. O'Connor, Jordi Torres, Elisa Sayrol, and Xavier Giro i Nieto. Salgan: Visual saliency prediction with generative adversarial networks. In *CVPR*, 2017.

[46] Hyun Soo Park, Jyh-Jing Hwang, Yedong Niu, and Jianbo Shi. Egocentric future localization. *CVPR*, 2016.

[47] Siyuan Qi, Siyuan Huang, Ping Wei, and Song-Chun Zhu. Predicting human activities using stochastic grammar. *ICCV*, 2017.

[48] Noha Radwan, Abhinav Valada, and Wolfram Burgard. Vlocnet++: Deep multitask learning for semantic visual localization and odometry. *CoRR*, 2018.

[49] Nicholas Rhinehart and Kris M. Kitani. First-person activity forecasting with online inverse reinforcement learning. *ICCV*, 2017.

[50] Richard Roberts, , Hai Nguyen, Niyant Krishnamurthi, and Tucker Balch. Memory-based learning for visual odometry. In *ICRA*, 2008.

[51] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelovic, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *NeurIPS*, 2018.

[52] Jayant Sharma, Zhongru Wang, Alberto Speranzon, Vijay Venkataraman, and Hyun Soo Park. Eco: Egocentric cognitive mapping. *ArXiv*, 2018.

[53] Yang Shen, Bingbing Ni, Zefan Li, and Ning Zhuang. Egocentric activity prediction via event modulated attention. In *ECCV*, 2018.

[54] Gunnar A. Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Actor and observer: Joint modeling of first and third-person videos. *CVPR*, 2018.

[55] Krishna Kumar Singh, Kayvon Fatahalian, and Alexei A. Efros. Krishnacam: Using a longitudinal, single-person, egocentric dataset for scene understanding tasks. *WACV*, 2016.

[56] Suriya Singh, Chetan Arora, and C. V. Jawahar. First person action recognition using deep learned descriptors. *CVPR*, 2016.

[57] Shan Su, Jung Pyo Hong, Jianbo Shi, and Hyun Soo Park. Social behavior prediction from first person videos. *ArXiv*, 2016.

[58] Yu-Chuan Su and Kristen Grauman. Detecting engagement in egocentric video. In *ECCV*, 2016.

[59] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. *CVPR*, 2018.

[60] Xiaolong Wang, Rohit Girdhar, and Abhinav Gupta. Binge watching: Scaling affordance learning from sitcoms. In *CVPR*, 2017.

[61] Rohit Girdhar Xiaolong Wang and Abhinav Gupta. Binge watching: Scaling affordance learning from sitcoms. *CVPR*, 2017.

[62] Jia Xu, Lopamudra Mukherjee, Yin Li, Jamieson Warner, James M. Rehg, and Vikas Singh. Gaze-enabled egocentric video summarization via constrained submodular maximization. In *CVPR*, 2015.

[63] Mingze Xu, Chenyou Fan, Yuchen Wang, Michael S. Ryoo, and David J. Crandall. Joint person segmentation and identification in synchronized first- and third-person videos. In *ECCV*, 2018.

[64] Kihwan Kim Xiaolong Wang Ming-Hsuan Yang Xueting Li, Sifei Liu and Jan Kautz. Putting humans in a scene: Learning affordance in 3d indoor environments. *CVPR*, 2019.

[65] Takuma Yagi, Karttikeya Mangalam, Ryo Yonetani, and Yoichi Sato. Future person localization in first-person videos. *CVPR*, 2018.

[66] Heiko Neumann Michael J. Black Yan Zhang, Mohamed Hassan and Siyu Tang. Generating 3d people in scenes without people. *CVPR*, 2020.

[67] Liang Yang, Hao Jiang, Jizhong Xiao, and Zhouyuan Huo. Ego-downward and ambient video based person location association. *ArXiv*, 2018.

[68] Ting Yao, Tao Mei, and Yong Rui. Highlight detection with pairwise deep ranking for first-person video summarization. *CVPR*, 2016.

[69] Ye Yuan and Kris Kitani. 3d ego-pose estimation via imitation learning. In *ECCV*, 2018.

[70] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time pd control. *ICCV*, 2019.

[71] Mengmi Zhang, Keng Teck Ma, Joo-Hwee Lim, Qi Zhao, and Jiashi Feng. Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks. *CVPR*, 2017.

[72] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J Black, and Siyu Tang. Generating 3d people in scenes without people. In *CVPR*, 2020.