Anticipating Visual Representations from Unlabeled Video

Anticipating Visual Representations from Unlabeled Video

Carl Vondrick Hamed Pirsiavash† Antonio Torralba
Massachusetts Institute of Technology †University of Maryland, Baltimore County
{vondrick,torralba}@mit.edu hpirsiav@umbc.edu

Abstract

Anticipating actions and objects before they start or appear is a difficult problem in computer vision with several real-world applications. This task is challenging partly because it requires leveraging extensive knowledge of the world that is difficult to write down. We believe that a promising resource for efficiently learning this knowledge is through readily available unlabeled video. We present a framework that capitalizes on temporal structure in unlabeled video to learn to anticipate human actions and objects. The key idea behind our approach is that we can train deep networks to predict the visual representation of images in the future. Visual representations are a promising prediction target because they encode images at a higher semantic level than pixels yet are automatic to compute. We then apply recognition algorithms on our predicted representation to anticipate objects and actions. We experimentally validate this idea on two datasets, anticipating actions one second in the future and objects five seconds in the future.

1. Introduction

What action will the man do next in Figure 1 (left)? A key problem in computer vision is to create machines that anticipate actions and objects in the future, before they appear or start. This predictive capability would enable several real-world applications. For example, robots can use predictions of human actions to make better plans and interactions [18]. Recommendation systems can suggest products or services based on what they anticipate a person will do. Predictive models can also find abnormal situations in surveillance videos, and alert emergency responders.

Unfortunately, developing an algorithm to anticipate the future is challenging. Humans can rely on extensive knowledge accumulated over their lifetime to infer that the man will soon shake hands in Figure 1. How do we give machines access to this knowledge?

We believe that a promising resource to train predictive models are abundantly available unlabeled videos. Although lacking ground truth annotations, they are attractive for prediction because they are economical to obtain at massive scales yet still contain rich signals. Videos come with the temporal ordering of frames "for free", which is a valuable asset for forecasting.

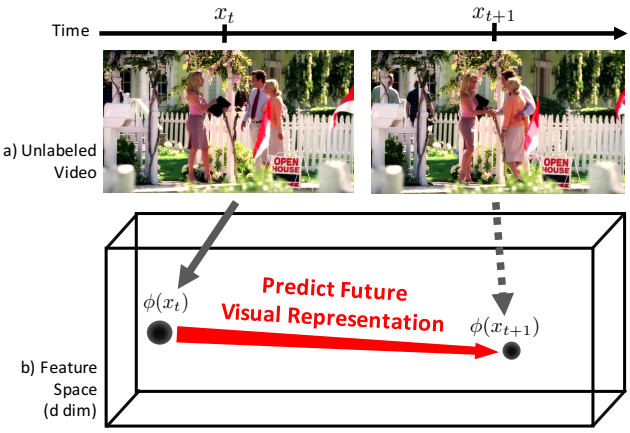

Figure 1: **Predicting Representations:** In this paper, we explore how to anticipate human actions and objects by learning from unlabeled video. We propose to anticipate the visual representation of frames in the future. We can apply recognition algorithms on the predicted representation to forecast actions and objects.

However, how to leverage unlabeled video to anticipate high-level concepts is unclear. Pioneering work in computer vision has capitalized on unlabeled videos before to visualize the future [31, 36, 39] and predict motions [28, 40, 43]. Unfortunately, these self-supervised approaches are not straightforward to apply for anticipating semantics because, unlike pixels or motions, concepts are not readily accessible in unlabeled video. Methods that anticipate concepts have typically required supervision [16, 21, 12], which is expensive to scale.

In this paper, we propose a method to anticipate concepts in the future by learning from unlabeled video. Recent progress in computer vision has built rich visual representations [6, 32, 44]. Rather than predict pixels or depend on supervision, our main idea is to forecast visual representations of future frames. Since these representations contain

signals sufficient to recognize concepts in the present, we then use recognition algorithms on the forecasted representation to anticipate a future concept. Representations have the advantage that they both a) capture the semantic information that we want to forecast and b) scale to unlabeled videos because they are automatic to compute. Moreover, representations may be easier to predict than pixels because distance metrics in this space empirically tend to be more robust [6, 19].

Since we can economically acquire a large amounts of unlabeled video, we create our prediction models with deep networks, which are attractive for this problem because their capacity can grow with the size of data available and are trained efficiently with large-scale optimization algorithms. In our experiments, we downloaded 600 hours of unlabeled video from the web and trained our network to forecast representations 1 to 5 seconds in the future. We then forecast both actions and objects by applying recognition algorithms on top of our predicted representations. We evaluate this idea on two datasets of human actions in television shows [25] and egocentric videos for activities of daily living [29]. Although we are still far from human performance on these tasks, our experiments suggest that learning to forecast representations with unlabeled videos may help machines anticipate some objects and actions.

The primary contribution of this paper is developing a method to leverage unlabeled video for forecasting high-level concepts. In section 2, we first review related work. In section 3, we then present our deep network to predict visual representations in the future. Since the future can be uncertain, we extend our network architecture to produce multiple predictions. In section 4, we show experiments to forecast both actions and objects. We plan to make our trained models and code publicly available.

2. Related Work

The problem of predicting the future in images and videos has received growing interest in the computer vision community, which our work builds upon:

Prediction with Unlabeled Videos: Perhaps the ideas most similar to this paper are the ones that capitalize on the wide availability of big video collections. In early work, Yuen and Torralba [43] propose to predict motion in a single image by transferring motion cues from visually similar videos in a large database. Building on the rich potential of large video collections, Walker et al. [39] demonstrate a compelling data-driven approach that animates the trajectory of objects from a single frame. Ranzato et al. [31] and Srivastava et al. [36] also learn predictive models from large unlabeled video datasets to predict pixels in the future. In this paper, we also use large video collections. However, unlike previous work that predicts low-level pixels or motions, we develop a system to predict high-level concepts such as objects and actions by learning from unlabeled video.

Predicting Actions: There have been some promising works on predicting future action categories. Lan et al. [21] propose a hierarchical representation to predict future actions in the wild. Ryoo [33] and Hoai and De la Torre [11] propose models to predict actions in early stages. Vu et al. in [38] learn scene affordance to predict what actions can happen in a static scene. Pei et al. [26] and Xie et al. [42] infer people's intention in performing actions which is a good clue for predicting future actions. We are different from these approaches because we use large-scale unlabeled data to predict a rich visual representation in the future, and apply it towards anticipating both actions and objects.

Predicting Human Paths: There have been several works that predict the future by reasoning about scene semantics with encouraging success. Kitani et al. [16] use concept detectors to predict the possible trajectories a person may take in surveillance applications. Lezema et al. [23], Gong et al. [8] and Kooij et al. [17] also predict the possible future path for people in the scene. Koppula and Saxena [18] anticipate the action movements a person may take in a human robot interaction scenario using RGB-D sensors. Our approach extends these efforts by predicting human actions and objects.

Predicting Motions: One fundamental component of prediction is predicting short motions, and there have been some investigations towards this. Pickup et al. in [27] implicitly model causality to understand what should happen before what in a video. Fouhey and Zitnick [7] learn from abstract scenes to predict what objects may move together. Lampert [20] predicts the future state of a probability distribution, and applies it towards predicting classifiers adapted to future domains. Pintea et al. [28] predict the optical flow from single images by predicting how pixels are going to move in future. We are hoping that our model learns to extrapolate these motions automatically in the visual representation, which is helpful if we want to perform recognition in the future rather than rendering it in pixel space.

Big Visual Data: We build upon work that leverages a large amount of visual data readily available online. Torralba et al. [37] use millions of Internet images to build object and scene recognition systems. Chen et al. [2] and Divvala et al. [3] build object recognition systems that have access to common sense by mining visual data from the web. Doersch et al. [5] use large repositories of images from the web to tease apart visually distinctive elements of places. Kim and Xing [15] learn to reconstruct story lines in personal photos, and recommend future photos. Zhou et al. [45] train convolutional neural networks on a massive number of scene images to improve scene recognition accuracy. In our work, we also propose to mine information from visual media on the web, however we do it for videos with the

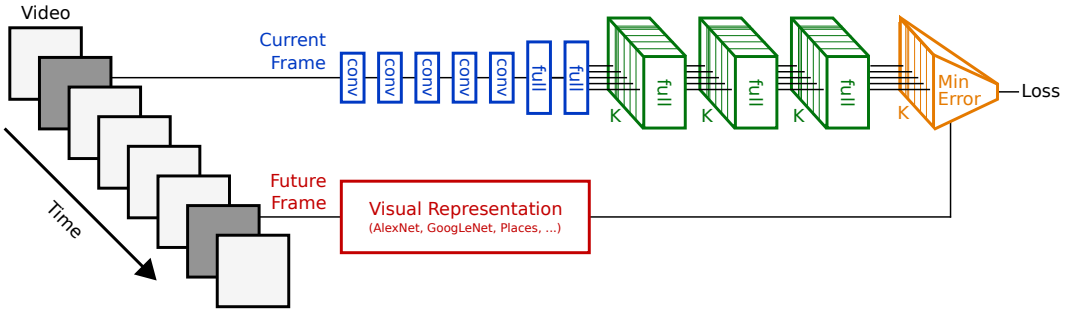

Figure 2: **Network Diagram:** We visualize the network architecture we use in our experiments. During training, the network uses videos to learn to predict the representation of frames in the future. Since predicting the future is a multi-modal problem, our network predicts K future representations. Blue layers are the same for each output while green layers are separate for the K outputs. During inference, we only input the current frame, and the network estimates K representations for the future. Please see section 3 for full details.

goal of learning a model to anticipate semantic concepts.

Unsupervised Learning in Vision: To handle large-scale data, there have been some efforts to create unsupervised learning systems for vision. Ramanan et al. [30] uses temporal relationships in videos to build datasets of human faces. Ikizler-Cinbis et al. [13] propose to use images from the web to learn and annotate actions in videos without supervision. Le et al. [22] show that machines can learn to recognize both human and cat faces by watching an enormous amount of YouTube videos. Chen and Grauman [1] propose a method to discover new human actions by only analyzing unlabeled videos, and Mobahi et al. [24] similarly discover objects. This paper also proposes to use unlabeled data, but we use unlabeled video to learn to predict visual representations.

Representation Learning: Recent work has explored how to learn visual representations, for example with images [4] or videos [41]. Our work is different because we do not seek to learn a visual representation. Rather, our goal is to anticipate the visual representation in the future. Moreover, our approach is general, and in principle could predict any representation.

3. Anticipating Visual Representations

Rather than predicting pixels (which may be more difficult) or anticipating labeled categories (which requires supervision), our idea is to use unlabeled video to learn to predict the visual representation in the future. We can then apply recognition algorithms (such as object or action classifiers) on the predicted future representation to anticipate a high-level concept. In this section, we explain our approach.

3.1. Self-supervised Learning

Given a video frame x_t^i at time t from video i, our goal is to predict the visual representation for the future frame $x_{t+\Delta}^i$. Let $\phi(x_{t+\Delta}^i)$ be the representation in the future. Using videos as training data, we wish to estimate a function $g(x_t^i)$ that closely predicts $\phi(x_{t+\Delta}^i)$:

$$\omega^* = \underset{\omega}{\operatorname{argmin}} \sum_{i,t} \|g\left(x_t^i; \omega\right) - \phi\left(x_{t+\Delta}^i\right)\|_2^2 \qquad (1)$$

where our prediction function $g(\cdot)$ is parameterized by ω.

Our method is general to most visual representations, however we focus on predicting the last hidden layer (`fc7`) of AlexNet [19]. We chose this layer because it empirically obtains state-of-the-art performance on several image [32, 6] and video [44] recognition tasks.

3.2. Deep Regression Network

Since we do not require data to be labeled for learning, we can collect large amounts of training data. We propose to use deep regression networks for predicting representations because their model complexity can expand to harness the amount of data available and can be trained with large scale data efficiently with stochastic gradient descent.

Our network architecture is five convolutional layers followed by five fully connected layers. The last layer is the output vector, which makes the prediction for the future representation. In training, we use a Euclidean loss to minimize the distance between our predictions $g(x_t)$ and the representation of the future frame $\phi(x_{t+\Delta})$.

Our choice of architecture is motivated by the successes of the AlexNet architecture for visual recognition [19, 45]. However, our architecture differs by having a regression loss function and three more fully connected layers.

3.3. Multi-Modal Outputs

Given an image, there can be multiple plausible futures, illustrated in Figure 3. We wish to handle the multi-modal nature of this problem for two reasons. Firstly, when there

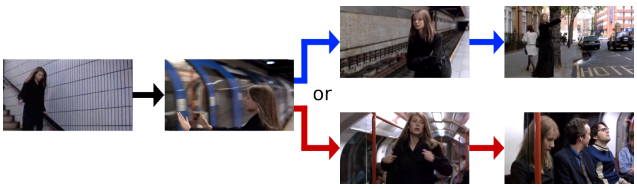

Figure 3: **Multiple Futures:** Since the future can be uncertain, our model anticipates multiple possibilities.

are multi-modal outputs, the optimal least squares solution for regression is to produce the mean of the modes. This is undesirable because the mean may either be unlikely or off the manifold of representations. Secondly, reasoning about uncertain outcomes can be important for some applications of future prediction.

We therefore extend deep regression networks to produce multiple outputs. Suppose that there are K possible output vectors for one input frame. We can support multiple outputs by training a mixture of K networks, where each mixture is trained to predict one of the modes in the future. Given input x_t^i, network k will produce one of the outputs $g_k(x_t^i)$.

3.4. Learning

To train multiple regression networks, we must address two challenges. Firstly, videos only show one of the possible futures (videos like Figure 3 are rare). Secondly, we do not know to *which* of the K mixtures each frame belongs. We overcome both problems by treating the mixture assignment for a frame as latent.

Let $z_t^i \in \{1, \ldots, K\}$ be a latent variable indicating this assignment for frame t in video i. We first initialize z uniformly at random. Then, we alternate between two steps. First, we solve for the network weights w end-to-end using backpropagation assuming z is fixed:

$$\omega^* = \underset{\omega}{\operatorname{argmin}} \sum_{i,t} \left|\left| g_{z_t^i}\left(x_t^i; \omega\right) - \phi\left(x_{t+\Delta}^i\right) \right|\right|_2^2 \quad (2)$$

Then, we re-estimate z using the new network weights:

$$z_t^i = \underset{k \in \{1,\ldots,K\}}{\operatorname{argmin}} \left|\left| g_k\left(x_t^i; \omega\right) - \phi\left(x_{t+\Delta}^i\right) \right|\right|_2^2 \quad (3)$$

We alternate between these two steps several times, a process that typically takes two days. We learn w with warm starting, and let it train for a fixed number of iterations before updating z. We illustrate this network in Figure 2.

Although we train our network offline in our experiments, we note our network can be also be trained online with streaming videos. Online learning is attractive because the network can continuously learn how to anticipate the future without storing frames. Additionally, the model can adapt in real time to the environment, which may be useful in some applications.

3.5. Predicting Categories

Since our network uses unlabeled videos to predict a representation in the future, we need a way to attach semantic category labels to it. To do this, we use a relatively small set of labeled examples from the target task to indicate the category of interest. As the representation that we predict is the same that is used by state-of-the-art recognition systems, we can apply standard recognition algorithms to the predicted representation in order to forecast a category.

We explore two strategies for using recognition algorithms on the predicted representations. The first strategy uses a visual classifier trained on the standard features (we use `fc7`) from frames containing the category of interest, but applies it on a predicted representation. The second strategy trains the visual classifier on the predicted representations as well. The second strategy has the advantage that it can adapt to structured errors in the regression.

During inference, our model will predict multiple representations of the future. By applying category classifiers to each predicted representation, we will obtain a distribution for how likely categories are to happen in each future representation. We marginalize over these distributions to obtain the most likely category in the future.

3.6. Implementation

Our network architecture consists of 5 convolutional layers followed by 5 fully connected layers. We use ReLU nonlinear activations throughout the network. The convolutional part follows the AlexNet architecture, and we refer readers to [19] for complete details. After the convolutional layers, we have 5 fully connected layers each with 4096 hidden units.

The K networks (for each output) can either be disjoint or share parameters between them. In our experiments, we opted to use the following sharing strategy in order to reduce the number of free parameters. For the five convolutional layers and first two hidden layers, we tie them across each mixture. For the last three fully connected layers, we interleave hidden units: we randomly commit each hidden unit to a network with probability $p = \frac{1}{2}$, which controls the amount of sharing between networks. We do this assignment once, and do not change it during learning.

We trained the networks jointly with stochastic gradient descent. We used a Tesla K40 GPU and implemented the network in Caffe [14]. We modified the learning procedure to handle latent variables. We initialized the first seven layers of the network with the Places-CNN network weights [45], and the remaining layers with Gaussian white noise and the biases to a constant. During learning, we also used dropout [35] with a dropout ratio of $\frac{1}{2}$ on every fully con-

Figure 4: **Unlabeled Videos:** We collected more than 600 hours of unlabeled video from YouTube. We show a sample of the frames above. We use this data to train deep networks that predict visual representations in the future.

nected layer. We used a fixed learning rate of .001 and momentum term of 0.9.

4. Experiments

In this section, we experiment with how well actions and objects can be forecasted using the predicted representations. We show results for forecasting basic human actions one second before they start, and anticipating household objects five seconds before they appear.

4.1. Unlabeled Repository

In order to train our network to predict features, we leverage a large amount of unlabeled video. We experimented with two sources of unlabeled videos:

Television Shows: We downloaded over 600 hours of publicly available television shows from YouTube. To pick the set of television shows, we used the top shows according to Google. The videos we downloaded generally consist of people performing a large variety of everyday actions, such as eating or driving, as well as interactions with objects and other people. We show a few example frames of these videos in Figure 4. We use this repository in most of our experiments. Since we test on different datasets, one concern is that there may be videos in the repository that also appear in a testing set. To check this, we queried for nearest neighbors between this repository and all testing sets, and found no overlap.

THUMOS: We also experimented with using videos from the THUMOS challenge [9], which consists of 400 hours of video from the web. These videos tend to be tutorials and sports, which has a different distribution from television shows. We only use THUMOS as a diagnostic dataset to quantify the performance of our method when the training distribution is very different from the testing set.

4.2. Baselines

Our goal in this paper is to learn from unlabeled video to anticipate high-level concepts (specifically actions and objects) in the future. Since our method uses minimal supervision to attach semantic meaning to the predicted representation, we compare our model against baselines that use a similar level of supervision. See Table 1 for an overview of the methods we compare.

SVM: One reasonable approach is to train a classifier on the frames before the action starts to anticipate the category label in the future. This baseline is able to adapt to contextual signals that may suggest the onset of an action. However, since this method requires annotated videos, it does not capitalize on unlabeled video.

MMED: We can also extend the SVM to handle sequential data in order to make early predictions. We use the code out-of-the-box provided by [11] for this baseline.

Nearest Neighbor: Since we have a large unlabeled repository, one reasonable approach is to search for the nearest neighbor, and use the neighbor's future frame as the predicted representation, similar to [43].

Linear: Rather than training a deep network, we can also train a linear regression on our unlabeled repository to predict `fc7` in the future.

Adaptation: We also examine two strategies for training the final classifier. One way is to train the classifier on the ground truth regression targets, and test it on the inferred output of the regression. The second way is to adapt to the predictions by also training the classifier on the inferred output of the regression. The latter can adapt to the errors in the regression.

4.3. Forecasting Actions

Dataset: In order to evaluate our method for action forecasting, we require a labeled testing set where a) actions are temporally annotated, b) we have access to frames before the actions begin, and c) consist of everyday human actions (not sports). We use the TV Human Interactions dataset [25] because it satisfies these requirements. The dataset consists of people performing four different actions (hand shake, high five, hug, and kissing), with a total of 300 videos.

Setup: We run our predictor on the frames before the annotated action begins. We use the provided train-test splits with 25-fold cross validation. We evaluate classification accuracy (averaged across cross validation folds) on making predictions one second before the action has started. To attach semantic meaning to our predicted representation, we use the labeled examples from the training set in [25]. As we make multiple predictions, for evaluation purposes we consider a prediction to be correct only if the ground truth action is the most likely prediction under our model.

Results: Table 2 shows the classification accuracy of different models for predicting the future action one second into the future given only a single frame. Our results suggest that training deep models to predict future representations with unlabeled videos may help machines forecast actions, obtaining a **relative gain of 19%** over baselines.

Method	Feature	Regression				Classifier		
		Train Data	Method	Output	K	Frame	Data	Method
SVM Static	fc7	-	-	-	1	During	RO	SVM
SVM	fc7	-	-	-	1	Before	RO	SVM
MMED	fc7	-	-	-	1	Before	RO	MMED
Nearest Neighbor	fc7	UV	1-NN	fc7	1	Before	RI	SVM
Nearest Neighbor Adapted	fc7	UV	1-NN	fc7	1	Before	RO	SVM
Linear	fc7	UV	Linear	fc7	1	Before	RI	SVM
Linear Adapted	fc7	UV	Linear	fc7	1	Before	RO	SVM
Deep K=1	RGB	UV	CNN	fc7	1	Before	fc7 of RI	SVM
Deep K=1 Adapted	RGB	UV	CNN	fc7	1	Before	RO	SVM
Deep K=3	RGB	UV	CNN	fc7	3	Before	fc7 of RI	SVM
Deep K=3 Adapted	RGB	UV	CNN	fc7	3	Before	RO	SVM
Deep K=3 THUMOS	RGB	THUMOS	CNN	fc7	3	Before	fc7 of RI	SVM
Deep K=3 THUMOS Adapted	RGB	THUMOS	CNN	fc7	3	Before	RO	SVM
Deep K=1 ActionBank Adapted	RGB	UV	CNN	ActionBank	1	Before	RO	SVM
Deep K=3 ActionBank Adapted	RGB	UV	CNN	ActionBank	3	Before	RO	SVM

Table 1: **Overview of Models:** We compare several different ways of training models, and this table shows their different configurations. To train the regression (if any), we specify which source of unlabeled videos we use (UV for our repository, or THUMOS), the method, the regression target output, and the number of outputs K. This is then fed into the classifier, which uses labeled data. To train the classifier, we specify which frame to train the classifier on (during action, or before action), the regression input (RI) or output (RO), and the classifier. During testing, the procedure is the same for all models.

Method	Accuracy
Random	25.0
SVM Static	36.2 ± 4.9
SVM	35.8 ± 4.3
MMED	34.0 ± 7.0
Nearest Neighbor	29.9 ± 4.6
Nearest Neighbor [43], Adapted	34.9 ± 4.7
Linear	32.8 ± 6.1
Linear, Adapted	34.1 ± 4.8
Deep K=1, ActionBank [34]	34.0 ± 6.1
Deep K=3, ActionBank [34]	35.7 ± 6.2
Deep K=1	36.1 ± 6.4
Deep K=1, Adapted	40.0 ± 4.9
Deep K=3	35.4 ± 5.2
Deep K=3, Adapted	$\mathbf{43.3 \pm 4.7}$
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	$\mathbf{43.6 \pm 4.8}$
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Table 2: **Action Prediction:** Classification accuracy for predicting actions one second before they begin given only a single frame. The standard deviation across cross-validation splits is next to the accuracy.

We conjecture our network may obtain the stronger performance partly because it can better predict the future `fc7`. The mean Euclidean distance between our model's regressions and the actual future is about 1789, while regressing the identity transformation is about 1907 and a linear regression is worse, around 2328.

Human Performance: To establish an upper expectation for the performance on this task, we also had 12 human volunteers study the training sets and make predictions on our testing set. Human accuracy is good (an average human correctly predicts 71% of the time), but not perfect due to the uncertain nature of the task. We believe humans are not perfect because the future has inherent uncertainty, which motivates the need for models to make multiple predictions. Interestingly, we can use the "wisdom of the crowds" to ensemble the human predictions and evaluate the majority vote, which obtains accuracy (85%).

We also performed several experiments to breakdown the performance our method. **Different Representations:** We also tried to train a deep network to forecast ActionBank [34] in the future instead of `fc7`, which performed worse. Representations are richer than action labels, which may provide more constraints during learning that can help build more robust models [10]. **Different Training Sets:** We also evaluated our network trained on videos that are not television shows, such as sports and tutorials. When we train our network with videos from THUMOS [9] instead of our repository, we still obtain competitive performance, suggesting our method may be robust to some dataset biases. However, adaptation becomes more important for THUMOS, likely because the classifier must adapt to the dataset bias. **Different Intervals:** We also evaluated our

Figure 5: **Example Action Forecasts:** We show some examples of our forecasts of actions one second before they begin. The left most column shows the frame before the action begins, and our forecast is below it. The right columns show the ground truth action. Note that our model does not observe the action frames during inference.

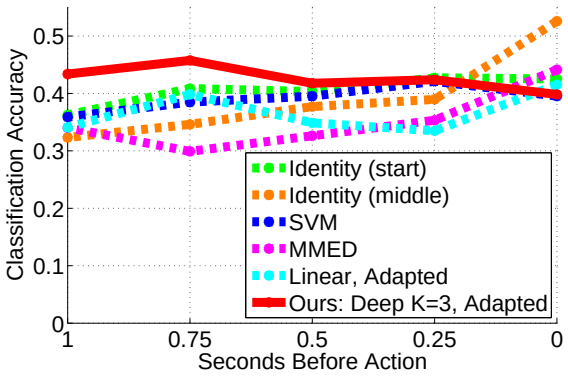

Figure 6: **Performance vs Δ:** We plot performance on forecasting actions versus number of frames before the action starts. Our model (red) performs better when the time range is longer (left of plot). Note that, since our model takes days to train, we evaluate our model trained for one second, but evaluate on different time intervals. The baselines are trained for each time interval.

model varying the time before the action starts in Figure 6. The relative gain of our method is often better as the prediction time frame increases.

Figure 7: **Multiple Predictions:** Given an input frame (left), our model predicts multiple representations in the future that can each be classified into actions (middle). When the future is uncertain, each network can predict a different representation, allowing for multiple action forecasts. To obtain the most likely future action, we can marginalize the distributions from each network (right).

Figure 8: **Example Object Forecasts:** We show examples of high scoring forecasts for objects. The left most frame is five seconds before the object appears.

Multiple Predictions: Since we learn a mixture of networks, our model can make diverse predictions when the future is uncertain. To analyze this, Figure 7 shows a scene and a distribution of possible future actions. For example, consider the first row where the man and woman are about to embrace, however whether they will kiss or hug is ambiguous. In our model, two of the networks predict a representation where kissing is the most likely future action, but one network predicts a representation where the most

Method	Mean	dish	door	utensil	cup	oven	person	soap	tap	tbrush	tpaste	towel	trashc	tv	remote
Random	1.2	1.2	2.8	1.1	2.4	1.6	0.8	1.5	2.1	0.2	0.3	0.6	1.1	0.5	0.3
SVM Static	6.4	2.6	15.4	2.9	5.0	9.4	6.9	11.5	17.6	1.6	1.0	1.5	6.0	2.0	5.9
SVM	5.3	3.0	8.2	5.2	3.6	8.3	12.0	6.7	11.7	3.5	1.5	4.9	1.3	0.9	4.1
Scene	8.2	3.3	18.5	5.6	3.6	18.2	10.8	9.2	6.8	8.0	8.1	5.1	5.7	2.0	10.3
Scene, Adapted	7.5	4.6	9.1	6.1	5.7	15.4	13.9	5.0	15.7	13.6	3.7	6.5	2.4	1.8	1.7
Linear	6.3	7.5	9.3	7.2	5.9	2.8	1.6	13.6	15.2	3.9	5.6	2.2	2.9	2.3	7.8
Linear, Adapted	5.3	2.8	13.5	3.8	3.6	11.5	11.2	5.8	4.9	5.4	3.3	3.4	1.6	2.1	1.0
Deep K=1	9.1	4.4	17.9	3.0	14.8	11.9	9.6	17.7	15.1	6.3	6.9	5.0	5.0	1.3	8.8
Deep K=1, Adapted	8.7	3.5	11.0	9.0	6.5	16.7	16.4	8.4	22.2	12.4	7.4	5.0	1.9	1.6	0.5
Deep K=3	**10.7**	4.1	22.2	5.7	16.4	17.5	8.4	19.5	20.6	9.2	5.3	5.6	4.2	8.0	2.6
Deep K=3, Adapted	**10.1**	3.5	14.7	14.2	6.7	14.9	15.8	8.6	29.7	12.6	4.6	10.9	1.8	1.4	1.9

Table 3: **Object Prediction:** We show average precision for forecasting objects five seconds before they appear in egocentric videos. For most categories, our method improves prediction performance. The last column is the mean across all categories.

likely action is hugging. The other rows show similar scenarios. Since performance drops when $K = 1$, modeling multiple outputs may be important both during learning and inference.

Qualitative Results: We qualitatively show some of our predictions in Figure 5. For example, in some cases our model correctly predicts that a man and woman are about to kiss or hug or that men in a bar will high five. The second to last row shows a comic scene where one man is about to handshake and the other is about to high five, which our model confuses. In the last row of Figure 5, our model incorrectly forecasts a hug because a third person unexpectedly enters the scene.

4.4. Forecasting Objects

Dataset: Since our method predicts a visual representation in the future, we wish to understand how well we can anticipate concepts other than actions. We experimented with forecasting objects in egocentric videos five seconds before the object appears. We use the videos from Activities of the Daily Living dataset [29], which is one of the largest datasets of egocentric videos from multiple people. Anticipating objects in this dataset is challenging because even recognizing objects in these videos is difficult [29].

Setup: In order to train our deep network on egocentric videos, we reserved three fourths of the dataset as our repository for self-supervised learning. We evaluate on the remaining one fourth videos, performing leave-one-out to learn future object category labels. Since multiple objects can appear in a frame, we evaluate the average precision for forecasting the occurrence of objects five seconds before they appear, averaged over leave-one-out splits.

Baselines: We compare against baselines that are similar to our action forecasting experiments. However, we add an additional baseline that uses scene features [45] to anticipate objects. One hypothesis is that, since most objects are correlated with their scene, recognizing the scene may be a good cue for predicting the onset of objects. We use an SVM trained on state-of-the-art scene features [45].

Results: Table 3 shows average precision for our method versus the baselines on forecasting objects five seconds into the future. For the many of the object categories, our model outperforms the baselines at anticipating objects, with **a mean relative gain of 30%** over baselines. Moreover, our model with multiple outputs improves over a single output network, suggesting that handling uncertainty in learning is helpful for objects too. The adapted and off-the-shelf networks perform similarly to each other in the average. Finally, we also qualitatively show some high scoring object predictions in Figure 8.

5. Conclusion

The capability for machines to anticipate future concepts before they begin is a key problem in computer vision that will enable many real-world applications. We believe abundantly available unlabeled videos are an effective resource we can use to acquire knowledge about the world, which we can use to learn to anticipate future.

Acknowledgements: We thank members of the MIT vision group for predicting the future on our test set. We thank TIG for managing our computer cluster, especially Garrett Wollman for troubleshooting many data storage issues. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research. This work was supported by NSF grant IIS-1524817, and by a Google faculty research award to AT, and a Google PhD fellowship to CV.

References

[1] C.-Y. Chen and K. Grauman. Watching unlabeled video helps learn new human actions from very few labeled snapshots. *CVPR*, 2013.

[2] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. *ICCV*, 2013.

[3] S. K. Divvala et al. Learning everything about anything: Webly-supervised visual concept learning. *CVPR*, 2014.

[4] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. *ICCV*, 2015.

[5] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros. What makes paris look like paris? *ACM Trans. Graph.*, 2012.

[6] J. Donahue et al. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv*, 2013.

[7] D. F. Fouhey and C. L. Zitnick. Predicting object dynamics in scenes. *CVPR*, 2014.

[8] H. Gong, J. Sim, M. Likhachev, and J. Shi. Multi-hypothesis motion planning for visual object tracking. *CVPR*, 2011.

[9] A. Gorban et al. THUMOS challenge: Action recognition with a large number of classes, 2015.

[10] G. E. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *NIPS*, 2014.

[11] M. Hoai and F. De la Torre. Max-margin early event detectors. *IJCV*, 2014.

[12] D.-A. Huang and K. M. Kitani. Action-reaction: Forecasting the dynamics of human interaction. *ECCV*, 2014.

[13] N. Ikizler-Cinbis, R. G. Cinbis, and S. Sclaroff. Learning actions from the web. *ICCV*, 2009.

[14] Y. Jia, E. Shelhamer, et al. Caffe: Convolutional architecture for fast feature embedding. *arXiv*, 2014.

[15] G. Kim and E. P. Xing. Reconstructing storyline graphs for image recommendation from web community photos. In *CVPR*, 2014.

[16] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert. Activity forecasting. *ECCV*, 2012.

[17] J. F. P. Kooij, N. Schneider, F. Flohr, and D. M. Gavrila. Context-based pedestrian path prediction. *ECCV*, 2014.

[18] H. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. *RSS*.

[19] A. Krizhevsky et al. Imagenet classification with deep convolutional neural networks. *NIPS*, 2012.

[20] C. H. Lampert. Predicting the future behavior of a time-varying probability distribution. In *CVPR*, 2015.

[21] T. Lan, T.-C. Chen, and S. Savarese. A hierarchical representation for future action prediction. *ECCV*, 2014.

[22] Q. V. Le et al. Building high-level features using large scale unsupervised learning. *ICML*, 2013.

[23] J. Lezama et al. Track to the future: Spatio-temporal video segmentation with long-range motion cues. *CVPR*, 2011.

[24] H. Mobahi, R. Collobert, and J. Weston. Deep learning from temporal coherence in video. *ICML*, 2009.

[25] A. Patron-Perez et al. High five: Recognising human interactions in tv shows. *BMVC*, 2010.

[26] M. Pei, Y. Jia, and S.-C. Zhu. Parsing video events with goal inference and intent prediction. *ICCV*, 2011.

[27] L. C. Pickup et al. Seeing the arrow of time. *CVPR*, 2014.

[28] S. L. Pintea et al. Déjà vu. *ECCV*, 2014.

[29] H. Pirsiavash and D. Ramanan. Detecting activities of daily living in first-person camera views. *CVPR*, 2012.

[30] D. Ramanan, S. Baker, and S. Kakade. Leveraging archival video for building face datasets. *ICCV*, 2007.

[31] M. Ranzato et al. Video (language) modeling: a baseline for generative models of natural videos. *arXiv*, 2014.

[32] A. S. Razavian et al. Cnn features off-the-shelf: an astounding baseline for recognition. *arXiv*, 2014.

[33] M. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. *ICCV*, 2011.

[34] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. *CVPR*, 2012.

[35] N. Srivastava et al. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[36] N. Srivastava et al. Unsupervised learning of video representations using lstm. *arXiv*, 2015.

[37] A. Torralba et al. 80 million tiny images: A large data set for nonparametric object and scene recognition. *PAMI*, 2008.

[38] T.-H. Vu, C. Olsson, I. Laptev, A. Oliva, and J. Sivic. Predicting actions from static scenes. *ECCV*, 2014.

[39] J. Walker, A. Gupta, and M. Hebert. Patch to the future: Unsupervised visual prediction. *CVPR*, 2014.

[40] J. Walker, A. Gupta, and M. Hebert. Dense optical flow prediction from a static image. *arXiv*, 2015.

[41] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. *arXiv*, 2015.

[42] D. Xie, S. Todorovic, and S.-C. Zhu. Inferring" dark matter" and" dark energy" from videos. *ICCV*, 2013.

[43] J. Yuen and A. Torralba. A data-driven approach for event prediction. *ECCV*, 2010.

[44] S. Zha et al. Exploiting image-trained cnn architectures for unconstrained video classification. *arXiv*, 2015.

[45] B. Zhou et al. Learning deep features for scene recognition using places database. *NIPS*, 2014.