Learning the Predictability of the Future

Dídac Surís*, Ruoshi Liu*, Carl Vondrick
Columbia University
hyperfuture.cs.columbia.edu

Abstract

We introduce a framework for learning from unlabeled video what is predictable in the future. Instead of committing up front to features to predict, our approach learns from data which features are predictable. Based on the observation that hyperbolic geometry naturally and compactly encodes hierarchical structure, we propose a predictive model in hyperbolic space. When the model is most confident, it will predict at a concrete level of the hierarchy, but when the model is not confident, it learns to automatically select a higher level of abstraction. Experiments on two established datasets show the key role of hierarchical representations for action prediction. Although our representation is trained with unlabeled video, visualizations show that action hierarchies emerge in the representation.

1. Introduction

The future often has narrow predictability. No matter how much you study Fig. 1, you will not be able to anticipate the exact next action with confidence. Go ahead and study it. Will they shake hands or high five?[1]

For the past decade, predicting the future has been a core computer vision problem [79, 31, 34, 69, 48, 54, 15, 65, 66] with a number of applications in robotics, security, and health. Since large amounts of video are available for learning, the temporal structure in unlabeled video provides excellent incidental supervision for learning rich representations [50, 14, 38, 51, 78, 71, 61, 21, 65, 67, 13, 72, 74]. While visual prediction is challenging because it is an underconstrained problem, a series of results in neuroscience have revealed a biological basis in the brain for how humans anticipate outcomes in the future [35, 3, 60].

However, the central issue in computer vision has been selecting *what* to predict in the future. The field has investigated a spectrum of options, ranging from generating pixels and motions [70, 36, 37] to forecasting activities [1, 65, 68] in the future. However, most representations are not *adaptive* to the fundamental uncertainties in video. While we cannot forecast the concrete actions in Fig. 1 nor generate its motions, all hope is not lost. Instead of forecasting whether the people will high five or shake hands, there is something else predictable here. We can "hedge the bet" and predict the abstraction that they will at least greet each other.

Figure 1: The future is often uncertain. Are they going to shake hands or high five?[1] Instead of answering this question, we should "hedge the bet" and predict the hyperonym that they will *at least* greet each other. In this paper, we introduce a hierarchical predictive model for learning what is predictable from unlabeled video.

This paper introduces a framework for learning from unlabeled video what is predictable. Instead of committing up front to a level of abstraction to predict, our approach learns from data which features are predictable. Motivated by how people organize action hierarchically [4], we propose a hierarchical predictive representation. Our approach jointly learns a hierarchy of actions while also learning to anticipate at the right level of abstraction.

Our method is based on the observation that hyperbolic geometry naturally and compactly encodes hierarchical structure. Unlike Euclidean geometry, hyperbolic space can be viewed as the continuous analog of a tree [53] because tree-like graphs can be embedded in finite-dimension with minimal distortion [19]. We leverage this property and learn predictive models in hyperbolic space. When the model is confident, it will predict at the concrete level of the hierarchy, but when the model is not confident, it learns to automatically select a higher-level of abstraction.

Experiments and visualizations show the pivotal role of hierarchical representations for prediction. On two established video datasets, our results show predictive hyperbolic representations are able to both recognize actions from partial observations as well as forecast them in the future better than baselines. Although our representation is trained with unlabeled video, visualizations also show action hierarchies

*Equal contribution

[1] The answer is in Season 2, Episode 16 of the *The Office*.

automatically emerge in the hyperbolic representation. The model explicitly represents uncertainty by trading off the specificity versus the generality of the prediction.

The primary contribution of this paper is a hierarchical representation for visual prediction. The remainder of this paper describes our approach and experiments in detail. Code and models are publicly available on github.com/cvlab-columbia/hyperfuture.

2. Related Work

Video representation learning aims to learn strong features for a number of visual video tasks. By taking advantage of the temporal structure in video, a line of work uses the future as incidental supervision for learning video dynamics [77, 59, 48]. Since generating pixels is challenging, [51, 14, 75] instead use the natural temporal order of video frames to learn self-supervised video representations. Similar to self-supervised image representations [12], temporal context also provides strong incidental supervision [26, 72].

A series of studies from Oxford has investigated how to *learn* a representation in the future [21, 22] using a contrastive objective. We urge readers to read these papers in detail as they are the most related to our work. While these models learn predictable features, the underlying representation is not adaptive to the varying levels of uncertainty in natural videos. They also focus on action recognition, and not action prediction. By representing action hierarchies in hyperbolic space, our model has robust inductive structure for hedging uncertainty.

Unlike action recognition, **future action prediction** [31] and **early action prediction** [56, 24] are tasks with an intrinsic uncertainty caused by the unpredictability of the future. Future action prediction infers future actions conditioned on the current observations. Approaches for future prediction range from classification [52, 1], through prediction of future features [65], to generation of the future at the skeletal [42] or even pixel [16, 27] levels. Early action prediction aims to recognize actions before they are completely executed. While standard action recognition methods can be used for this task [55, 64], most approaches mimic a sequential data arrival [33, 32, 73]. We evaluate our self-supervised learned representations on these two tasks.

Hyperbolic embeddings have emerged as excellent hierarchical language representations in natural language processing [53, 63]. These works are pioneering. Riemmanian optimization algorithms [6, 5] are used to optimize the models using hyperbolic geometry. Their success is largely attributed to the advantage of hyperbolic space to represent hierarchical structure. Following the Poincare embedding [53], [17] use a hyperbolic entailment cone to represent the hierarchical relation in an acyclic graph. [18] further applies hyperbolic geometry to feedforward neural networks and recurrent neural networks.

Since visual data is naturally hierarchical, hyperbolic space provides a strong inductive bias for images and videos as well. [29, 11, 44] perform several image tasks, demonstrating the advantage of hyperbolic embeddings over Euclidean ones. [45] proposes video and action embeddings in the hyperbolic space and trains a cross-modal model to perform hierarchical action search. We instead use hyperbolic embeddings for prediction and we use the hierarchy to model uncertainty in the future. We also learn the hierarchy from self-supervision, and our experiments show an action hierarchy emerges automatically.

Since dynamics are often stochastic, **uncertainty representation** underpins predictive visual models. There is extensive work on probabilistic models for visual prediction, and we only have space to briefly review. For example, [23] measures the covariance between outputs generated under different dropout masks, which reflects how close the predicted state is to the training data manifold. A high covariance indicates that the model is confident about its prediction. [2] use ensembling to estimate uncertainty, grounded in the observation that a mixture of neural networks will produce dissimilar predictions if the data point is rare. Another line of work focuses on generating multiple possible future prediction, such as variational auto-encoders (VAE) [30, 68], variational recurrent neural networks (VRNN) [10, 7] and adversarial variations [39]. These models allow sampling from the latent space to capture the multiple outcomes in the output space.

Probabilistic approaches are compatible with our framework. The main novelty of our method is that we represent the future uncertainty hierarchically in a hyperbolic space. The hierarchy naturally emerges during the process of learning to predict the future.

3. Method

We present our approach for learning a hierarchical representation of the future. We first discuss our model, then introduce our hyperbolic representation.

3.1. Predictive Model

Our goal is to learn a video representation that is predictive of the future. Let $x_t \in \mathbb{R}^{T\times W\times H\times 3}$ be a video clip centered at time t. Instead of predicting the pixels in the future, we will predict a representation of the future. We denote the representation of a clip as $z_t = f(x_t)$.

The prediction task aims to forecast the unobserved representation $z_{t+\delta}$ that is δ clips into the future. Given a temporal window of previous clips, our model estimates its prediction of $z_{t+\delta}$ as:

$$\hat{z}_{t+\delta} = \phi(c_t, \delta) \quad \text{for} \quad c_t = g(z_1, z_2, \ldots, z_t) \qquad (1)$$

where $c_t = g(\cdot)$ contextually encodes features of the video from the beginning up to and including frame t.

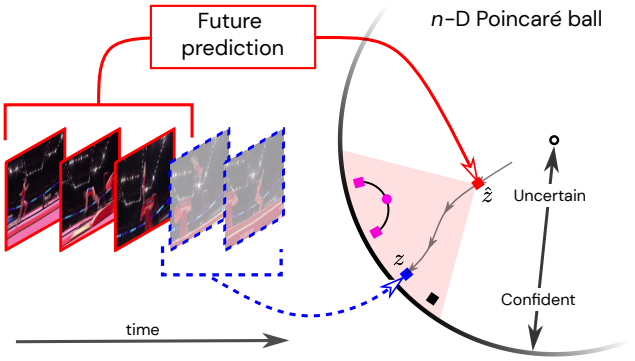

Figure 2: The future is non-deterministic. Given a specific past (first three frames in the figure), different representations (represented by squares in the Poincaré ball) can encode different futures, all of them possible. In case the model is uncertain, it will predict an abstraction of all these possible futures, represented by $\hat{z}$ (red square). The more confident it is, the more specific the prediction can get. Assuming the actual future is represented by z (blue square), the gray arrows represent the trajectory the prediction will follow as more information is available. The pink circle exemplifies the increase in generality when computing the mean of two specific representations (pink squares).

We will model f, g, and ϕ each with a neural network. In order to learn the parameters of these models, we need to define a distance metric between the unobserved future z and the prediction $\hat{z}$. However, the future is often uncertain. Instead of predicting the exact $z_{t+\delta}$ in the future, our goal is to predict an abstraction that encompasses the variation of the possible futures, and no more.

3.2. Hierarchical Representation

The key contribution of this paper is to predict a hierarchical representation of the future. When the future is certain, our model should predict $z_{t+\delta}$ as specifically as possible. However, when the future is uncertain, our model should "hedge the bet" and forecast a hierarchical parent of $z_{t+\delta}$. For example, in Fig. 1 the parent of a hand shake and a high five is a greeting. In order to parameterize this hierarchy, we will learn predictive models in *hyperbolic* space.

Informally, hyperbolic space can be viewed as the continuous analog of a tree [53]. Unlike Euclidean space, hyperbolic space has the unique property that circle areas and lengths grow exponentially with their radius. This density allows hierarchies and trees to be compactly embedded [19]. Since this space is naturally suited for hierarchies, hyperbolic predictive models are able to smoothly interpolate between forecasting abstract representations (in the case of low predictability) to concrete representations (in the case of high predictability).

The hyperbolic n-space, which we denote as $\mathbb{H}^n$, is a Riemannian geometry that has constant negative curvature.[2] While there are several models for hyperbolic space, we will use the the Poincaré model, which is also the most commonly used in gradient-based learning. The Poincaré ball model is formally defined by the manifold $\mathbb{D}^n = \{X \in \mathbb{R}^n : ||x|| < 1\}$ and the Riemmanian metric $g^{\mathbb{D}}$: $g_x^{\mathbb{D}} = \lambda_x^2 g^E$ where $\lambda_x := \frac{2}{1-\|x\|^2}$ such that $g^E = \mathbf{I}_n$ is the Euclidean metric tensor. For more details, see [40, 41].

We use the Poincaré ball model to define the distance metric between a prediction $\hat{z}$ and the observation z:

$$d_{\mathbb{D}}(\hat{z}, z) = \cosh^{-1}\left(1 + 2\frac{\|z-\hat{z}\|^2}{(1-\|z\|^2)\,(1-\|\hat{z}\|^2)}\right) \quad (2)$$

for points on the manifold $\mathbb{D}^n$. Recall that the mean minimizes the sum of squared residuals. The key property is that, in hyperbolic space, the mean between two leaf embeddings is not another leaf embedding, but an embedding that is a parent in the hierarchy. If the model cannot select between two leaf embeddings given the provided information, the expected squared distance will be minimized by instead producing the more abstract one as the prediction.

Fig. 2 visualizes this property on the Poincaré ball. Points near the center of the ball (having a smaller radius) represent abstract embeddings, while points near the edge (having a large radius) represent specific ones. In this example, the mean of two points close to the edge of the ball—illustrated by the two pink squares in Fig. 2—is a node further from the edge, represented with a pink circle. The line connecting the two squares is the minimum distance path between them, or *geodesic*. The midpoint is the mean.

Unlike trees, hyperbolic space is continuous, and there is not a fixed number of hierarchy levels. Instead, there is a continuum from very specific (closer to the border of the Poincaré ball) to very abstract (closer to the center).

3.3. Learning

To learn the parameters of the model, we want to minimize the distance between the predictions $\hat{z}_t$ and the observations z_t. We use the contrastive learning objective function [21] with hyperbolic distance as the similarity measure:

$$\mathcal{L} = -\sum_i \left[\log \frac{\exp\left(-d_{\mathbb{D}}^2(\hat{z}_i, z_i)\right)}{\sum_j \exp\left(-d_{\mathbb{D}}^2(\hat{z}_i, z_j)\right)}\right] \quad (3)$$

where z is the feature representing a spatio-temporal location in a video, and $\hat{z}$ is the prediction of that feature. The contrastive objective pulls positive pairs z and $\hat{z}$ together while also pushing $\hat{z}$ away from a large set of negatives, which avoids the otherwise trivial solution. There are a variety of strategies for selecting negatives [9, 62, 21]. We create negatives from other videos in the mini batch, as well

[2] We assume the curvature to be -1. Hyperbolic is one of three isotropic model spaces. The other two spaces are the Euclidean space $\mathbb{R}^n$ (zero curvature), and the spherical space $\mathbb{S}^n$ (constant positive curvature).

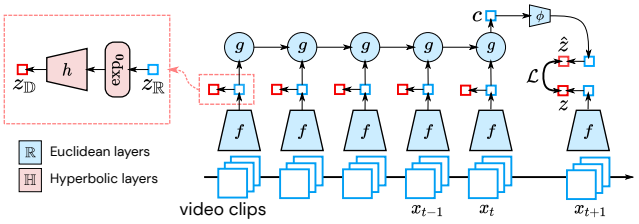

Figure 3: Overview of architecture. Blue and red respectively indicate Euclidean and hyperbolic modules.

as using features that correspond to the same video but in different spatial or temporal locations.

The solution to the hyperbolic contrastive objective minimizes the distance between the positive pair $d^2_{\mathbb{D}}(\hat{z}_i, z_i)$. When there is no uncertainty, the loss is minimized if $\hat{z}_t = z_t$. However, in the face of uncertainty between two possible outcomes a and b, the loss is minimized by predicting the midpoint on the geodesic between a and b. Since hyperbolic space has constant negative curvature, this solution corresponds to the latent parent embedding of a and b.

Our approach treats the hierarchy as latent. Since hyperbolic space is continuous and differentiable, we are able to optimize Eq. 3 with stochastic gradient descent, which jointly learns the predictive models with the hierarchy. The model will learn a hierarchy that is organized around the predictability of the future.

3.4. Classification

After the representation z is trained, we are able to fit any classifier on top of it. We use a linear classifier, and keep the rest of the representation fixed (no fine-tuning). However, since the representation is hyperbolic, we cannot use a standard Euclidean linear classifier. Instead, we use a hyperbolic multiclass logistic regression [18] that assumes the input representations are in the Poincaré ball, and fits a hyperplane in the same space. For the Euclidean baseline we train a standard Euclidean multiclass logistic regression. In both models, we treat each node as an independent category—not requiring a ground truth hierarchy—when training the classifier, and then compute accuracy values independently for each hierarchy level.

3.5. Network Architecture

While we estimate our predictions and loss in hyperbolic space, the entire model does not need to be hyperbolic. This flexibility enables us to take advantage of the extensive legacy of existing neural network architectures and optimization algorithms that have been highly tuned for Euclidean space. We therefore parameterize f, g, and ϕ with neural networks, which will be in Euclidean space. Our approach only instantiates z and $\hat{z}$ in hyperbolic space. Fig. 3 illustrates this architecture.

In order to use this hybrid architecture, we need a projection between the two spaces. The transition from the Euclidean space is based on the process mapping to Riemannian manifolds from their corresponding tangent spaces. A *Riemnnian manifold* is a pair $(\mathcal{M}, g)$, where $\mathcal{M}$ is a smooth manifold and g is a Riemmanian metric. Broadly, smooth manifolds are spaces that locally approximate Euclidean space $\mathbb{R}^n$, and on which one can differentiate [41], and this is precisely the connection between the two spaces. For $x \in \mathcal{M}$, one can define the tangent space $T_x\mathcal{M}$ of $\mathcal{M}$ at x as the first order linear approximation of $\mathcal{M}$ around x.

The *exponential map* $\exp_x : T_x\mathcal{M} \to \mathcal{M}$ at x is a map from the tangent spaces into the manifold. This projection is important for several operations, such as performing gradient updates [6]. The inverse of the exponential map is called *logarithmic map*, denoted $\log_x$.

We use an exponential map centered at $\mathbf{0}$ to project from the Euclidean space to the hyperbolic space [44]. Once the representations are in the hyperbolic space, the mathematical operations and the optimization follow the rules derived by the metric in that space. Essentially, a Riemannian metric defines an inner product g_x that allows us to define a global distance function as the infimum of the lengths of all curves between two points x and y [40]:

$$d(x, y) = \inf_{\gamma} L_g(\gamma), \tag{4}$$

where $\gamma : [0, 1] \to \mathcal{M}$ is a curve, and $L_g(\gamma)$ is the length of the curve, defined as:

$$L_g(\gamma) = \int_0^1 |\dot{\gamma}(t)| dt = \int_0^1 \sqrt{g_{\gamma(t)}(\dot{\gamma}(t), \dot{\gamma}(t))} dt. \tag{5}$$

In the specific case of the Poincaré ball model, this distance is the same as Eq. 2. Based on these concepts, several papers define extensions of the standard (Euclidean) neural network layers to the hyperbolic geometry [18, 58, 8, 43, 20]. We use the hyperbolic feed-forward layer defined in [18] to obtain the representation $z_{\mathbb{D}}$ that we use in Eq. 3, from the Euclidean representation $z_{\mathbb{R}}$. Specifically, we apply this layer after the exponential map, as shown in Fig 3. If the space is not specified, z is assumed to be $z_{\mathbb{D}}$.

We implement f with a 3D-ResNet18, g as a one-layer Convolutional Gated Recurrent Unit (ConvGRU) with kernel size $(1, 1)$, and ϕ using a two-layer perceptron. The dimensionality of the ResNet output is 256. When training with smaller dimensionality, we add an extra linear layer to project the representations. For more implementation details, we refer the reader to the supplementary materials.

4. Experiments

The basic objective of our experiments is to analyze how hyperbolic representations encode varying levels of uncertainty in the future. We quantitatively evaluate on two different tasks and two different datasets. We also show several visualizations and diagnostic analysis.

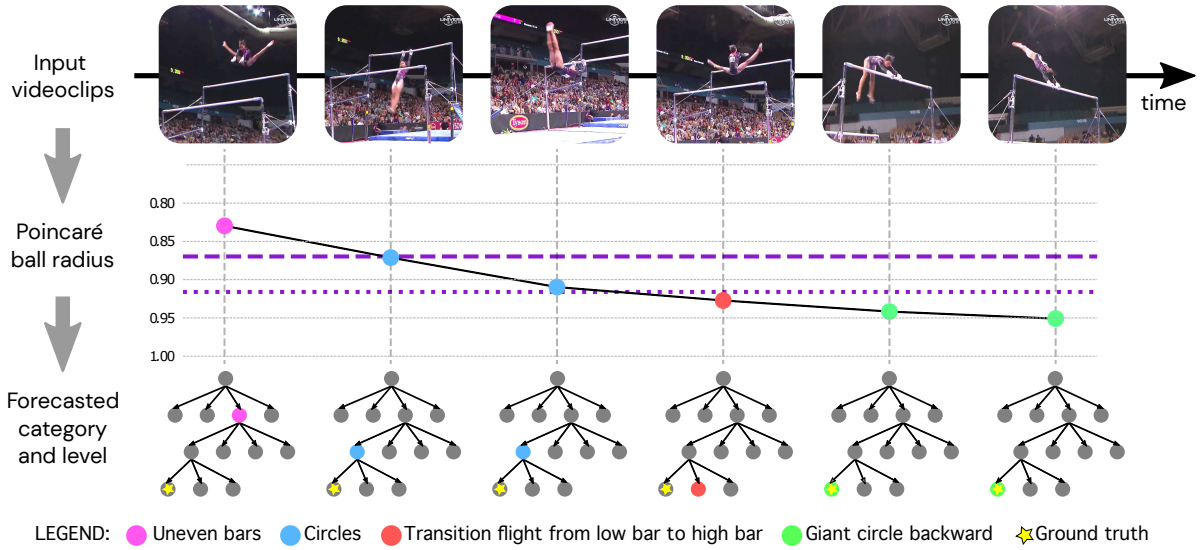

Figure 4: We show an example of future action prediction, where the model has to determine both a hierarchy level and a class within that level. At each time step, the model has to predict the class of the last action in the video. At the top of the figure we show the input video, where each image represents a video clip. From the input video, the model computes a representation and a level in the hierarchy based on the Euclidean norm of this representation. The thresholds shown in purple are the 33% percentile (dashed line) and the 66% percentile (dotted line), ranked by the radius of predicted hyperbolic embeddings of all videos in FineGym test set. Once a level in the tree has been determined, the model predicts the class within that level. We can see how the closer we get to the actual action to be predicted, the more confident the model is. Also, we can see how being overly confident (by setting a low threshold and choosing a level that is too specific) may lead the model to predict the wrong class. In this specific case, the fourth prediction (red dot) would have been more accurate had the model predicted a class in the parent level (i.e. had it predicted the blue dot). Note that the y-axis is inverted in the graph. Also, the nodes represented in the tree are not all the classes in the FineGym dataset, and each node has more children not shown.

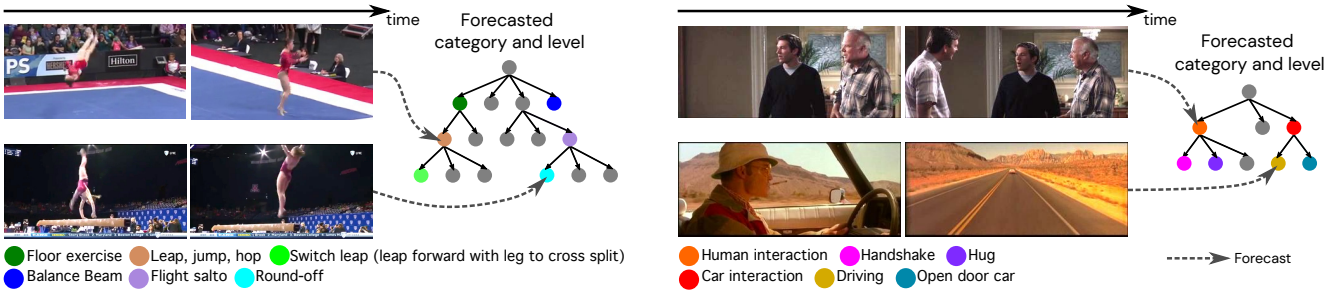

Figure 5: We show four more examples where the model correctly predicts the class at different hierarchical levels. The FineGym examples (left) are evaluated on future action prediction, and the Hollywood2 ones (right) on early action prediction, where we show the model half of the clips in the video. The shown trees are just partial representations of the complete hierarchies. Click here for video visualizations.

4.1. Datasets and Common Setup

We use a common evaluation setup throughout our experiments. We first learn a self-supervised representation from a large collection of unlabeled videos by optimizing Eq. 3. After learning the representation, we transfer these representations to the target domain using a smaller, labeled dataset. On the target domain, we fine-tune on the same objective before fitting a supervised linear classifier on $\hat{z}$ using a small number of labeled examples.

We evaluate on two different video datasets, which we selected for their realistic temporal structure:

Sports Videos: In this setting, we learn the self-supervised representation on Kinetics-600 [28] and fine-tune and evaluate on FineGym [57]. Kinetics has 600 human action classes and $500,000$ of videos which contain rich and diverse human actions. We discard its labels. FineGym is a dataset of gymnastic videos where clips are annotated with three-level hierarchical action labels, ranging from specific exercise names in the lowest level to generic gymnastic routines (e.g. *balance-beam*) in the highest one. The highest level of the hierarchy is consistent for all the clips in a video, so we use it as a label for the whole video.

Movies: In our second setting, we learn the self-supervised representation on MovieNet [25], then fine-tune and evaluate the Hollywood2 dataset [47]. MovieNet contains $1,100$ movies and $758,000$ key frames. In order to obtain a hierarchy of actions from Hollywood2 at the video level, we grouped action classes into more general ones to

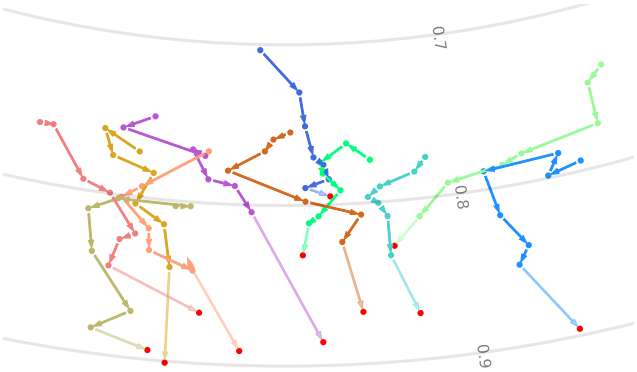

Figure 6: Trajectories showing the evolution with time of the predictions in Kinetics, where the task consists in predicting features of the last video clip. We show the two dimensions with larger values across all the predictions. Each line represents a prediction for a specific video.

form a 2-level hierarchy. The hierarchy is shown in the supplementary materials. Since Hollywood2 does not have fine-grained clip-level action labels like FineGym, we do not evaluate future action prediction on this dataset.

4.2. Evaluation metrics

In our quantitative evaluation, our goal is twofold. First, we want to evaluate that the obtained representations are better at modeling scenarios with high uncertainty. Second, we want to analyze the hierarchical structure of the learned space. We use three different metrics:

Accuracy: Standard classification accuracy. Compares only classes at the lowest (most specific) hierarchical level.

Bottom-up hierarchical accuracy: A prediction is considered partially correct if it predicts the wrong node at the leaf level but the correct nodes at upper levels. We weigh each level with a reward that decays by 50% as we go up in the hierarchy tree.

Top-down hierarchical accuracy: In the hierarchical metrics literature it is sometimes argued that predicting the root node correctly is more important than predicting the exact leaf node [76]. Therefore we also report the accuracy value that gives the root node a weight of 1, and decreases it the closer we get to the leaf node, also by a factor of 1/2.

For clarity, in both hierarchical evaluations we normalize the accuracy to be always within the $[0, 1]$ range. In all cases, higher is better.

4.3. Baselines

The main goal of the paper is to compare **hyperbolic** representations to **Euclidean** ones [21]. We therefore present our experiments on comparisons between these two spaces, keeping the rest of the method the same. It is worth noting that [21] has state-of-the-art results for several video tasks among self-supervised video approaches. Additionally, we report **chance** accuracy, resulting from randomly selecting a class, and the **most common** strategy, which always selects the most common class in the training set. We also experimented with several state-of-the-art video VAEs, and report the best one, which is **VRNN** [7].

Trees are compact in hyperbolic space. We show results for feature spaces with 256 dimensions, as in [21], as well as 64 dimensions.

Representation	Dim.	Accuracy (%)
Hyperbolic (ours)	256	**82.54**
Euclidean [21]	256	68.16
Hyperbolic (ours)	64	73.35
Euclidean [21]	64	66.04
VRNN [7]	512	44.28
Most common		35.40
Chance		25.00

Table 1: Early action prediction on FineGym. We do not report hierarchical accuracy because FineGym only annotates the hierarchy at the clip level, not video level. See Section 4.4 for discussion.

Representation	Dim.	Accuracy	Top-down hier. acc.	Bottom-up hier. acc.
Hyperbolic (ours)	256	**23.10**	**33.99**	**28.55**
Euclidean [21]	256	21.77	33.00	27.38
Hyperbolic (ours)	64	22.25	31.47	26.86
Euclidean [21]	64	15.47	24.09	19.78
VRNN [7]	512	18.75	20.33	18.20
Most common		17.08	25.19	21.13
Chance		8.33	16.11	12.22

Table 2: Early action prediction on Hollywood2. See Section 4.4.

4.4. Early Action Recognition

We first evaluate on the early action recognition task, which aims to classify actions that have started, but not finished yet. In other words, the observed video is only partially completed, producing uncertainty. We use video-level action labels to train the classification layer on $\hat{z}_N(c_t)$, for all time steps t. Tab. 1 and Tab. 2 show that, with everything else fixed, hyperbolic models learn significantly better representations than the Euclidean counterparts (up to 14% gain). The hyperbolic representation enjoys substantial compression efficiency, indicated by the 64 dimensional hyperbolic embedding outperforming the larger 256 dimensional Euclidean embedding (up to 5%). As indicated by both hierarchical accuracy metrics, when there is uncertainty, the hyperbolic representation will predict a more appropriate parent than Euclidean representations.

4.5. Future Action Prediction

We next evaluate the representation on future action prediction, which aims to predict actions before they start given the past context. There is uncertainty because the next actions are not deterministic.

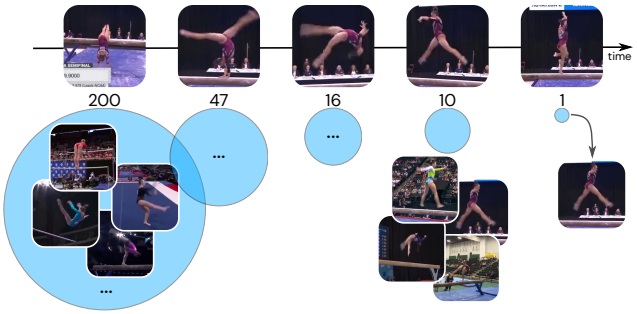

Figure 7: The area of the circle is proportional to the number of videos retrieved within a certain threshold distance. The more specific the prediction gets, the further most of the clips are (only a few ones get closer). The threshold is computed as the mean of all the distances from predictions to clip features. Note that the last circle does not necessarily have to contain exactly one video clip. The total number of clips for this retrieval experiment was 300.

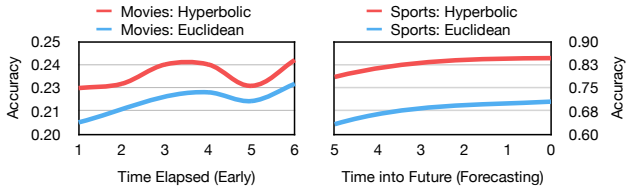

Figure 8: Classification accuracy for early action prediction (left) and future action prediction (right). Performance increases as the model receives more observations or predicts closer to the target. For all horizons, the hyperbolic representation is more accurate.

A key advantage of hyperbolic representations is that the model will automatically decide to select the level of abstraction based on its estimate of the uncertainty. If the prediction is closer to the center of the Poincaré ball, the model lacks confidence and it predicts a parental node close to the root in the hierarchy. If the prediction is closer to the border of the Poincaré ball, the model is more confident and consequently predicts a more specific outcome.

We fine-tune our model to learn to predict the class of the last clip of a video at each time step, for each of the three hierarchy levels in the FineGym dataset. We use clip-level labels to train the classification layer on the model's prediction $\hat{z}_N(c_t)$. We select a threshold between hierarchy levels by giving each level the same probability of being selected: the predictions that have a radius in the smaller than the 33% percentile will select the more general level, the ones above the 66% percentile will select the more specific level, and the rest will select the middle level.[3] Once the thresholds are set, we obtain both the predicted hierarchy level as well as the predicted class within that level.

Table 3 compares our predictive models versus the baseline in Euclidean space. We report values for $t = N - 1$. For all three metrics, predicting a hierarchical representation substantially outperforms baselines by up to 14 points. The gains in both top-down and bottom-up hierarchical accuracy show that our model selects a better level of abstraction than the baselines in the presence of uncertainty. We visualize the hyperbolic representation and the resulting hierarchical predictions in Fig. 4. We show more examples of forecasted levels and classes in Fig 5.

[3]These thresholds can be modified according to the risk tolerance of the application.

Representation	Dim.	Accuracy	Top-down hier. acc.	Bottom-up hier. acc.
Hyperbolic (ours)	256	**13.37**	**66.64**	**33.04**
Euclidean [21]	256	10.08	52.00	24.75
Hyperbolic (ours)	64	10.29	56.67	27.49
Euclidean [21]	64	9.26	52.41	26.22
VRNN [7]	512	6.72	43.54	18.34
Most common		3.64	27.90	12.75
Chance		0.00	16.24	5.67

Table 3: Future action prediction on FineGym. See Section 4.5.

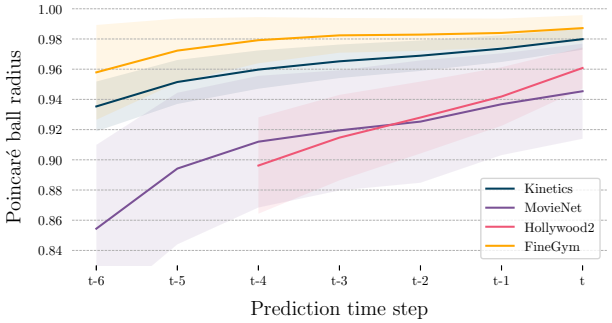

Figure 9: We visualize how the radius of predicted features evolve as more data is observed. The larger the radius, the more confident and more specific the prediction becomes.

The hyperbolic model also obtains better performance than the Euclidean model at the standard classification accuracy, which only evaluates the leaf node prediction. Since classification accuracy does not account for the hierarchy, this gain suggests hyperbolic representations help even when the future is certain. We hypothesize this is because the model is explicitly representing uncertainty, which stabilizes the training compared to the Euclidean baseline.

Since our model represents its prediction of the uncertainty, we are able to visualize which videos are predictable. Fig. 10 visualizes several examples.

4.6. Analysis of the Representation

We next analyze the emergent properties of the learned representations, and how they change as more information is given to the model. We conduct our analysis on the self-supervised representation, before supervised fine-tuning.

Fig. 6 visualizes the trajectory that representations follow as frames are continuously observed. We visualize the representation from Kinetics. In order to plot a 2D graph, we select the two dimensions with the highest mean value,

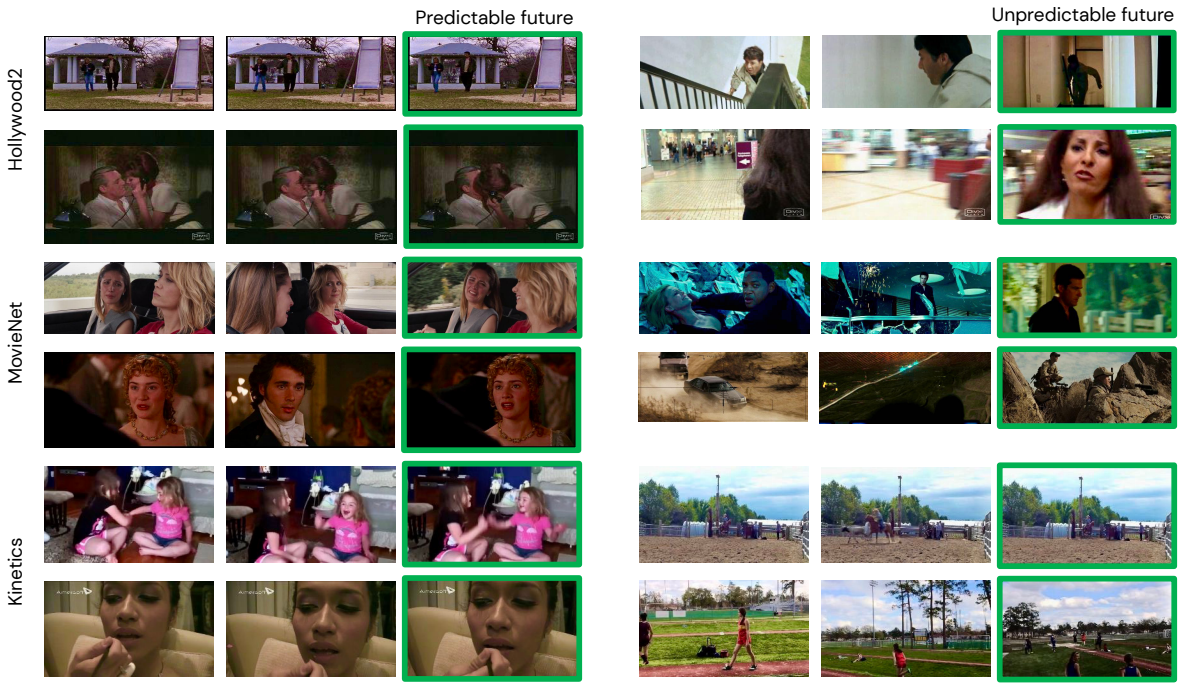

Figure 10: We show examples with high predictability (left), and low predictability (right). The first two frames represent the content the model sees, and the frame in green represents the action the model has to predict. The high predictability examples are selected from above the 99 percentile and the low predictability examples are selected from below the 1 percentile, measured by the radius of the prediction. In both Hollywood2 and MovieNet, the model is more certain about static actions, where the future is close to the past. Specifically in MovieNet we found that most of the highly predictable clips involved close-up conversations of people. The unpredictable ones correspond mostly to action scenes where the future possibilities are very diverse. In the Kinetics dataset, we also noticed that most of the predictable futures corresponded to videos with static people, and the unpredictable ones were associated generally to sports.

and plot their trajectories with the time.[4] The red dots show the actual features that have to be predicted. As the observations get closer to its prediction target, predictions get closer to the edge of the Poincaré ball, indicating they are becoming more specific in the hierarchy and increasingly confident about the prediction.

The distance to the center of the Poincaré ball gives us intuition about the underlying geometry, and Fig. 9 quantifies this behavior. We show the average radius of the predictions at each time step, together with the standard deviation. As more frames become available, the prediction gets consistently more confident for all datasets.

Abstract predictions will encompass a large number of specific features that can be predicted, while specific predictions will restrict the options to just a few. We visualize these predictions using nearest neighbors. Given a series of video clips belonging to the same event and gymnastics instrument in FineGym, we compute features for each one of the clips in these videos, that we use as targets to retrieve. For each video, we then predict the last representation at each time step (i.e., for every clip in the video), and use these as queries. We show the results for one such videos in Fig. 7. The retrieved number corresponds to the number of clips that are in a distance within a threshold, that we compute as the mean of all the distances from predictions to features. As the time horizon to the target action shrinks, the more specific the representation becomes, and thus fewer options are recalled. Fig. 8 quantifies performance versus time horizon, showing the hyperbolic representation is more accurate than the Euclidean representation for all time periods.

[4]Projecting using TSNE [46] or uMAP [49] does not respect the local structure well enough to visualize the radius of the prediction.

5. Conclusion

While there is uncertainty in the future, parts of it are predictable. We have introduced a hyperbolic model for video prediction that represents uncertainty hierarchically. After learning from unlabeled video, experiments and visualizations show that a hierarchy automatically emerges in the representation, encoding the predictability of the future.

Acknowledgments: We thank Basile Van Hoorick, Will Price, Mia Chiquier, Dave Epstein, Sarah Gu, and Ishaan Chandratreya for helpful feedback. This research is based on work partially supported by NSF NRI Award #1925157, the DARPA MCS program under Federal Agreement No. N660011924032, the DARPA KAIROS program under PTE Federal Award No. FA8750-19-2-1004, and an Amazon Research Gift. We thank NVidia for GPU donations. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Government.

References

[1] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what?-anticipating temporal occurrences of activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5343–5352, 2018.

[2] Charles Blundell Balaji Lakshminarayanan, Alexander Pritzel. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. *NIPS*, 2017.

[3] Christopher Baldassano, Janice Chen, Asieh Zadbood, Jonathan W Pillow, Uri Hasson, and Kenneth A Norman. Discovering event structure in continuous narrative perception and memory. *Neuron*, 95(3):709–721, 2017.

[4] Roger G Barker and Herbert F Wright. Midwest and its children: The psychological ecology of an american town. 1955.

[5] Gary Becigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. In *International Conference on Learning Representations*, 2019.

[6] Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013.

[7] Lluis Castrejon, Nicolas Ballas, and Aaron Courville. Improved conditional VRNNs for video prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7608–7617, 2019.

[8] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. In *Advances in neural information processing systems*, pages 4868–4879, 2019.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[10] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *Advances in Neural Information Processing Systems*, 2015-Janua:2980–2988, 2015.

[11] Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. Hierarchical image classification using entailment cone embeddings. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2020-June:3649–3658, 2020.

[12] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015.

[13] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1801–1810, 2019.

[14] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3636–3645, 2017.

[15] Katerina Fragkiadaki, Pulkit Agrawal, Sergey Levine, and Jitendra Malik. Learning visual predictive models of physics for playing billiards. *arXiv preprint arXiv:1511.07404*, 2015.

[16] Harshala Gammulle, Simon Denman, Sridha Sridharan, and Clinton Fookes. Predicting the future: A jointly learnt model for action anticipation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5562–5571, 2019.

[17] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. *35th International Conference on Machine Learning, ICML 2018*, 4:2661–2673, 2018.

[18] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *Advances in Neural Information Processing Systems*, 2018-December:5345–5355, 2018.

[19] Michael Gromov. Hyperbolic groups. In *Essays in group theory*, 1987.

[20] Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, and Nando de Freitas. Hyperbolic attention networks. *7th International Conference on Learning Representations, ICLR 2019*, i:1–15, 2019.

[21] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[22] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. *arXiv preprint arXiv:2008.01065*, 2020.

[23] Mikael Henaff, Yann LeCun, and Alfredo Canziani. Model-predictive policy learning with uncertainty regularization for driving in dense traffic. *7th International Conference on Learning Representations, ICLR 2019*, pages 1–20, 2019.

[24] Minh Hoai and Fernando De la Torre. Max-margin early event detectors. *International Journal of Computer Vision*, 107(2):191–202, 2014.

[25] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *European Conference on Computer Vision (ECCV)*, 2020.

[26] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *Workshop track - ICLR 2016*, 2015.

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

[28] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[29] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6418–6428, 2020.

[30] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, pages 1–14, 2014.

[31] Kris M Kitani, Brian D Ziebart, James Andrew Bagnell, and Martial Hebert. Activity forecasting. In *European Conference on Computer Vision*, pages 201–214. Springer, 2012.

[32] Yu Kong, Dmitry Kit, and Yun Fu. A discriminative model with multiple temporal scales for action prediction. In *European conference on computer vision*, pages 596–611. Springer, 2014.

[33] Yu Kong, Zhiqiang Tao, and Yun Fu. Deep sequential context networks for action prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1473–1481, 2017.

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

[35] Zoe Kourtzi and Nancy Kanwisher. Activation in human mt/mst by static images with implied motion. *Journal of cognitive neuroscience*, 12(1):48–55, 2000.

[36] Manoj Kumar, Mohammad Babaeizadeh, Dumitru Erhan, Chelsea Finn, Sergey Levine, Laurent Dinh, and Durk Kingma. Videoflow: A conditional flow-based model for stochastic video generation. In *International Conference on Learning Representations*, 2019.

[37] Yong-Hoon Kwon and Min-Gyu Park. Predicting future frames using retrospective cycle gan. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1811–1820, 2019.

[38] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 667–676, 2017.

[39] Hsin Ying Lee, Hung Yu Tseng, Jia Bin Huang, Maneesh Singh, and Ming Hsuan Yang. Diverse Image-to-Image Translation via Disentangled Representations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11205 LNCS:36–52, 2018.

[40] John M Lee. *Riemannian manifolds: an introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

[41] John M Lee. Smooth manifolds. In *Introduction to Smooth Manifolds*, pages 1–31. Springer, 2013.

[42] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5226–5234, 2018.

[43] Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. In *Advances in Neural Information Processing Systems*, pages 8230–8241, 2019.

[44] Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic Visual Embedding Learning for Zero-Shot Recognition. In *Computer Vision and Pattern Recognition*, pages 9273—-9281, 2020.

[45] Teng Long, Pascal Mettes, Heng Tao Shen, and Cees Snoek. Searching for Actions on the Hyperbole. *Cvpr*, pages 1138–1147, 2020.

[46] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[47] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2009.

[48] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[49] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[50] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-End Learning of Visual Representations from Uncurated Instructional Videos. In *CVPR*, 2020.

[51] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, pages 527–544. Springer, 2016.

[52] Yan Bin Ng and Basura Fernando. Forecasting future sequence of actions to complete an activity. *arXiv preprint arXiv:1912.04608*, 2019.

[53] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in Neural Information Processing Systems*, 2017-December(Nips):6339–6348, 2017.

[54] MarcAurelio Ranzato, Arthur Szlam, Joan Bruna, Michael Mathieu, Ronan Collobert, and Sumit Chopra. Video (language) modeling: a baseline for generative models of natural videos. *arXiv preprint arXiv:1412.6604*, 2014.

[55] Michalis Raptis and Leonid Sigal. Poselet key-framing: A model for human activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2650–2657, 2013.

[56] Michael S Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *2011 International Conference on Computer Vision*, pages 1036–1043. IEEE, 2011.

[57] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[58] Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. Hyperbolic neural networks++, 2020.

[59] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015.

[60] Kimberly L Stachenfeld, Matthew M Botvinick, and Samuel J Gershman. The hippocampus as a predictive map. *Nature neuroscience*, 20(11):1643, 2017.

[61] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. In *arXiv*, June 2019.

[62] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[63] Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincaré Glove: Hyperbolic word embeddings. *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[64] Arash Vahdat, Bo Gao, Mani Ranjbar, and Greg Mori. A discriminative key pose sequence model for recognizing human interactions. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1729–1736. IEEE, 2011.

[65] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 98–106, 2016.

[66] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances in neural information processing systems*, pages 613–621, 2016.

[67] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 391–408, 2018.

[68] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision*, pages 835–851. Springer, 2016.

[69] Jacob Walker, Abhinav Gupta, and Martial Hebert. Patch to the future: Unsupervised visual prediction. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3302–3309, 2014.

[70] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *Proceedings of the IEEE international conference on computer vision*, pages 3332–3341, 2017.

[71] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4006–4015, 2019.

[72] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2015.

[73] Xionghui Wang, Jian-Fang Hu, Jian-Huang Lai, Jianguo Zhang, and Wei-Shi Zheng. Progressive teacher-student learning for early action prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3556–3565, 2019.

[74] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019.

[75] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8052–8060, 2018.

[76] Cinna Wu, Mark Tygert, and Yann LeCun. A hierarchical loss and its problems when classifying non-hierarchically. *Plos one*, 14(12):e0226222, 2019.

[77] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[78] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10334–10343, 2019.

[79] Jenny Yuen and Antonio Torralba. A data-driven approach for event prediction. In *European Conference on Computer Vision*, pages 707–720. Springer, 2010.