Anticipating Visual Representations from Unlabeled Video

CVPR 2016

Carl Vondrick, Hamed Pirsiavash, Antonio Torralba

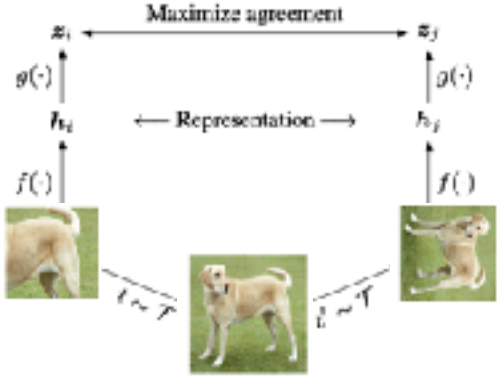
Self-Supervised Learning in Images

Pathak et al. “Context Encoders: Feature Learning by Inpainting”, CVPR 2015

Doersch et al. “Unsupervised Visual Representation Learning by Context Prediction”, ICCV 2015

Real or Fake?

Brock et al. “Large Scale GAN Training For High Fidelity Natural Image Synthesis”, ICLR 2019



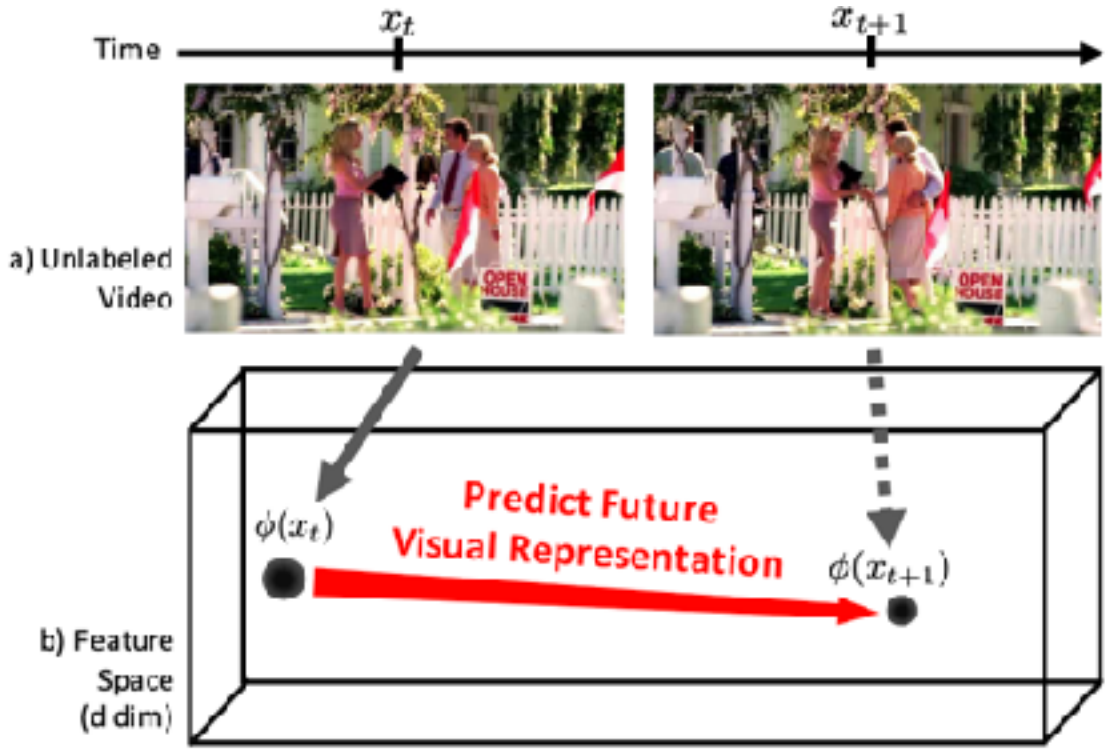
Chen et al. “A Simple Framework for Contrastive Learning of Visual Representations”, ICML 2020

Self-Supervised Learning in Videos

- What can we do with a temporal dimension?

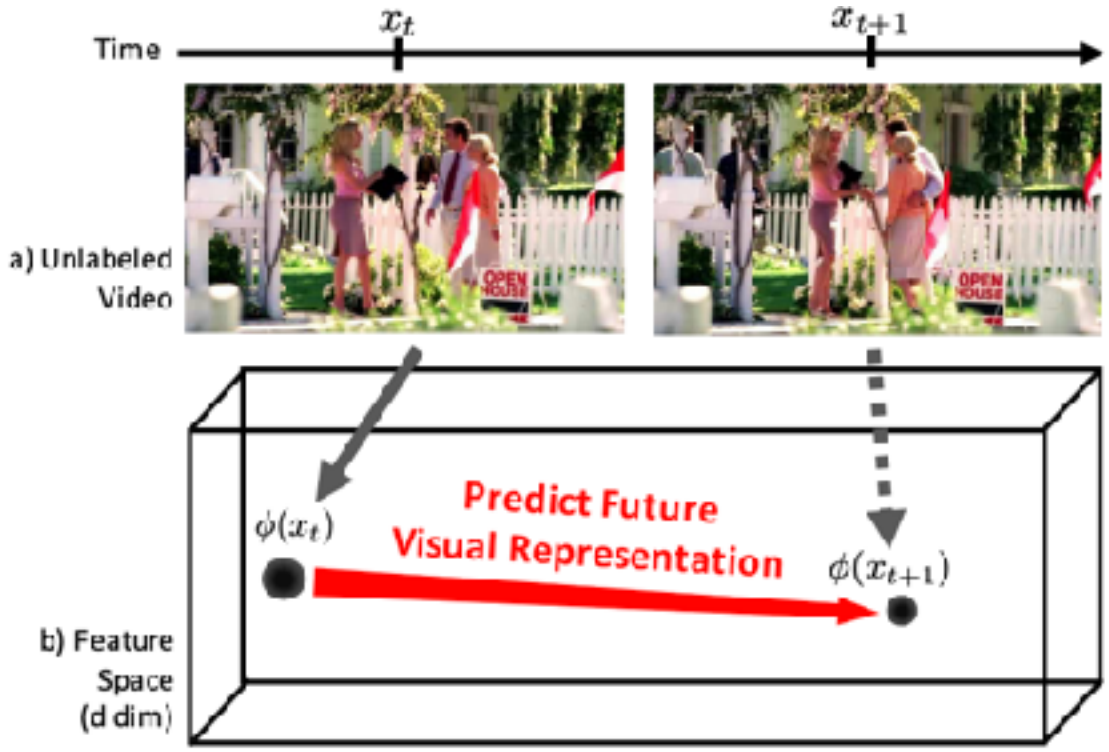
Self-Supervised Learning in Videos

- What can we do with a temporal dimension?



Self-Supervised Learning in Videos

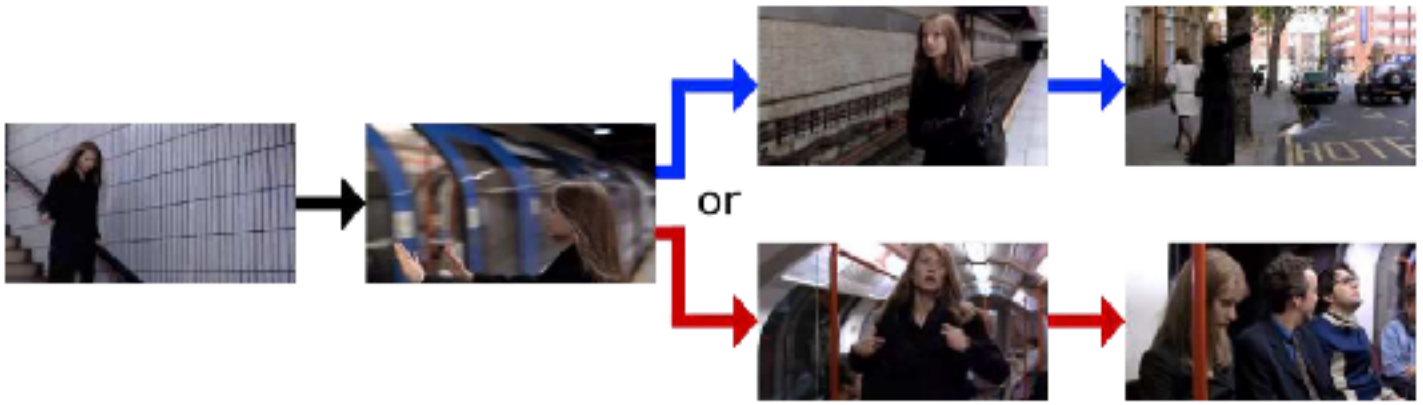
- What can we do with a temporal dimension?



Enables the model to anticipate future.

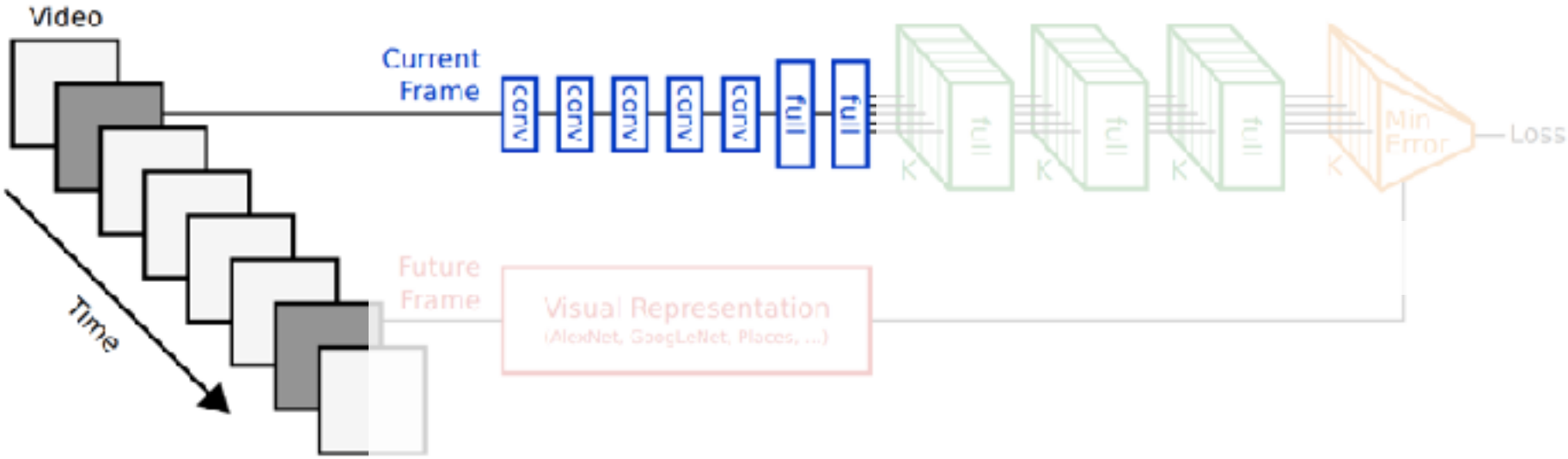
Challenges

- Future is inherently uncertain.
- How do we incorporate that into our model?



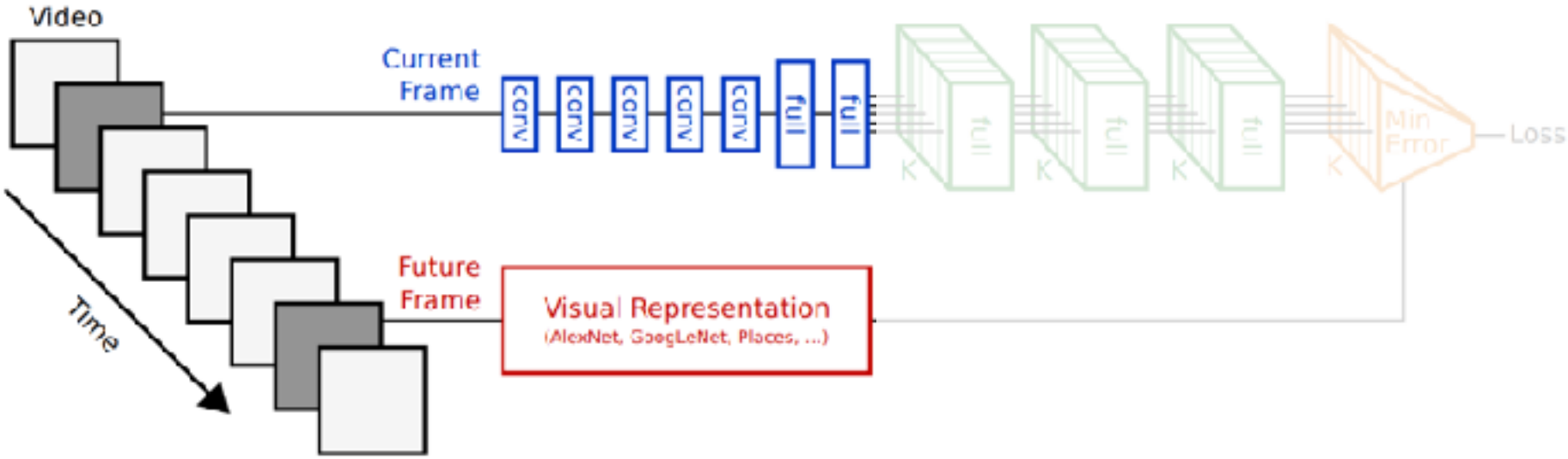
Technical Approach

- During training, the network uses videos to learn to predict the representation of frames in the future.
- To account for uncertainty in future prediction, the network predicts K future representations.



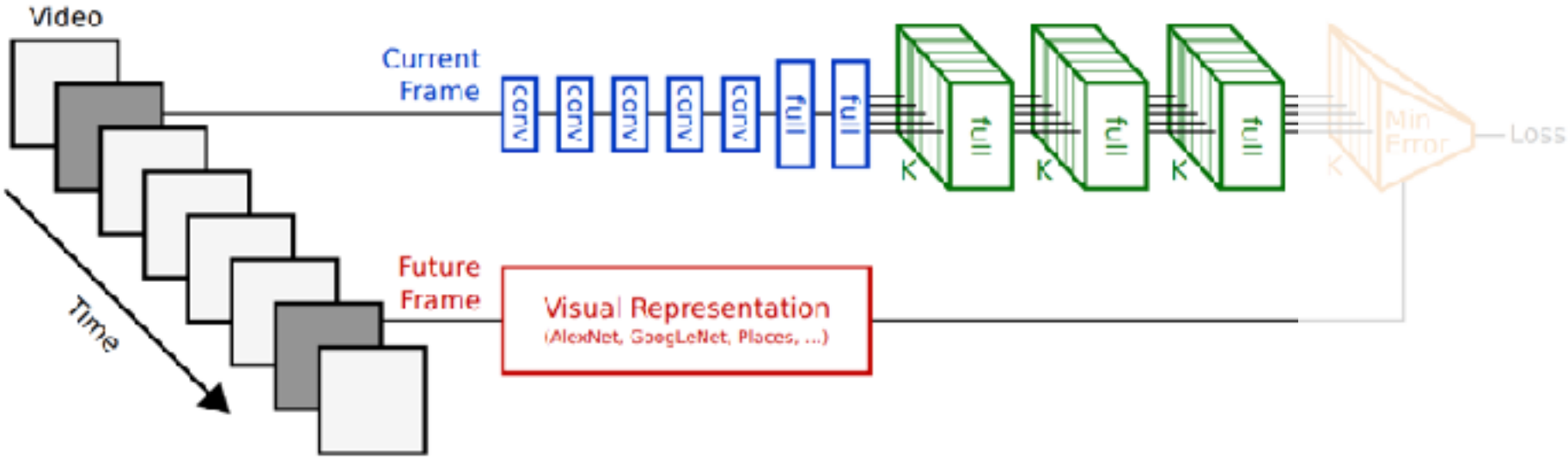
Technical Approach

- During training, the network uses videos to learn to predict the representation of frames in the future.
- To account for uncertainty in future prediction, the network predicts K future representations.



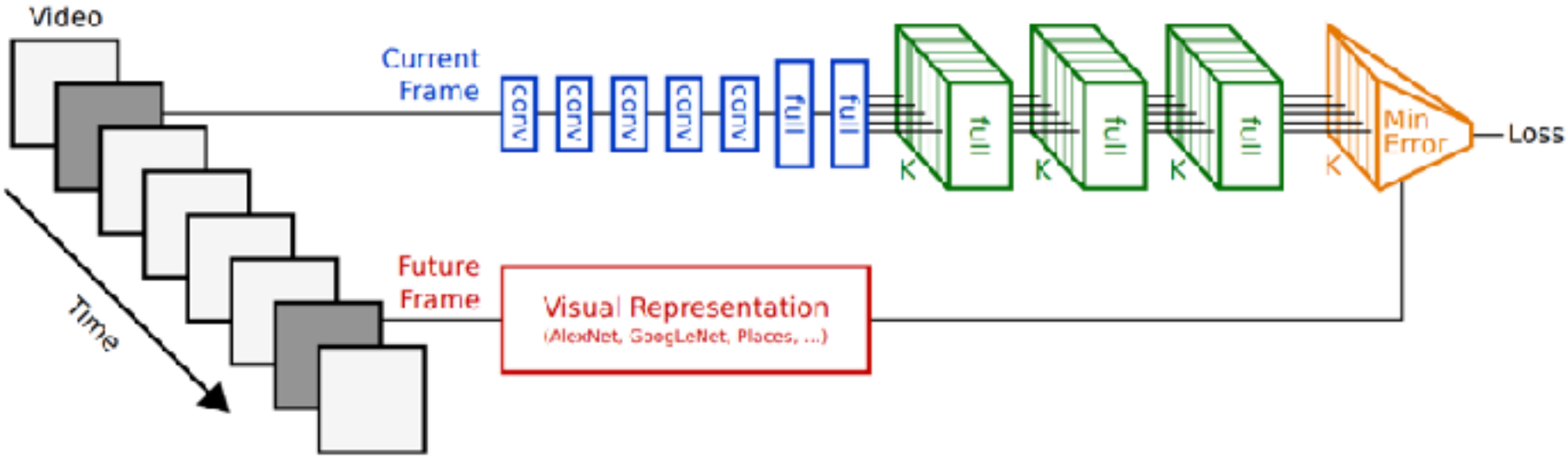
Technical Approach

- During training, the network uses videos to learn to predict the representation of frames in the future.
- To account for uncertainty in future prediction, the network predicts K future representations.



Technical Approach

- During training, the network uses videos to learn to predict the representation of frames in the future.
- To account for uncertainty in future prediction, the network predicts K future representations.



Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

Frame t

Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

Network g

Network g parameters

Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

Frame t+\delta

Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

Pretrained model used to extract features from that frame

Loss Function

- During the forward pass, we feed frame t through all K branches.
- During the backward pass, we only backpropagate gradients through the branch associated with the minimum loss.

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

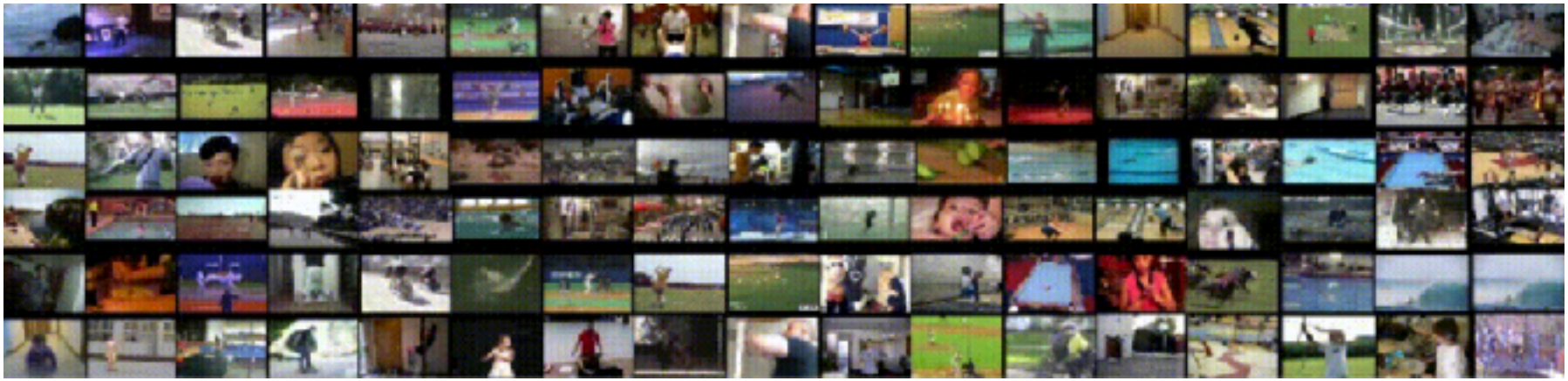
Only backpropagating through the branch with minimum loss

Training Dataset #1

- ~600 hours of publicly available television shows from YouTube
- The authors used the top shows according to Google

Training Dataset #2

- The authors also experimented with using videos from the THUMOS dataset.
- 400 hours of video from the web (mostly tutorials and sports).



Evaluation

- The evaluation is done on the action forecasting task.
- To assess action forecasting performance, the authors use the TV Human Interactions dataset.
- The dataset consists of people performing four different actions (hand shake, high five, hug, and kissing).
- 300 videos in total.

Inference

- During inference, the proposed model will predict multiple representations of the future.
- Applying category classifiers to each predicted representation will lead to a distribution for how likely categories are to happen in each future representations.
- The final prediction can then be obtained via averaging.

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Random guessing of the action category produces accuracy of 25%

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

The proposed approach outperforms simple baselines by a convincing margin.

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Using fc7 representation as a supervisory signal is more beneficial than using predicted action labels.

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

The method is quite robust to different training datasets.

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Using multiple branches to model future uncertainty is beneficial

Results on TV Human Interactions

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	**43.3 ± 4.7**
Deep K=3, THUMOS [9], Off-the-shelf	29.1 ± 3.9
Deep K=3, THUMOS [9], Adapted	**43.6 ± 4.8**
Human (single)	71.7 ± 4.2
Human (majority vote)	85.8 ± 1.6

Humans can obtain ~70-80% accuracy on this task.

Qualitative Results

Qualitative Results

Contributions

- An elegant method for the action forecasting task.
- Instead of using manually annotated data, the proposed approach uses unlabeled video data, which is easy and cheap to obtain.
- The proposed approach models uncertainty in future prediction.

Discussion Questions

- Is this a self-supervised approach?

Discussion Questions

- Is this a self-supervised approach?
- What's the disadvantage of using unlabeled vs labeled data for representation learning?

Discussion Questions

- Is this a self-supervised approach?
- What's the disadvantage of using unlabeled vs labeled data for representation learning?
- Questionable details in the approach?