Robust Audio-Visual Instance Discrimination

Pedro Morgado*  Ishan Misra  Nuno Vasconcelos
UC San Diego  Facebook AI Research  UC San Diego

Abstract

We present a self-supervised learning method to learn audio and video representations. Prior work uses the natural correspondence between audio and video to define a standard cross-modal instance discrimination task, where a model is trained to match representations from the two modalities. However, the standard approach introduces two sources of training noise. First, audio-visual correspondences often produce faulty positives since the audio and video signals can be uninformative of each other. To limit the detrimental impact of faulty positives, we optimize a weighted contrastive learning loss, which down-weights their contribution to the overall loss. Second, since self-supervised contrastive learning relies on random sampling of negative instances, instances that are semantically similar to the base instance can be used as faulty negatives. To alleviate the impact of faulty negatives, we propose to optimize an instance discrimination loss with a soft target distribution that estimates relationships between instances. We validate our contributions through extensive experiments on action recognition tasks and show that they address the problems of audio-visual instance discrimination and improve transfer learning performance.

1. Introduction

Self-supervised representation learning aims to learn feature representations that can transfer to downstream tasks without costly human annotations. Many recent self-supervised methods [11, 36, 51, 14, 78, 74] use a variant of the instance discrimination framework [81, 22], which matches features from multiple views/augmentations of the same instance, while distinguishing these features from those of other instances. This often relies on a contrastive loss [31], where different augmentations are considered ‘positives’ and other samples ‘negatives.’

Cross-modal instance discrimination (xID) extends instance discrimination to the realm of multiple modalities, where data modalities, such as video, audio, or text, act as the

*Contacting author.
of learning with noisy targets. This raises the question of how to identify faulty positive and negative samples in the absence of human annotations.

We propose to use cross-modal information during self-supervised training to detect both faulty positive and negative instances. This is done by estimating the quality of the audio-visual correspondence of each instance and optimizing a weighted contrastive learning loss that down-weights the contribution of faulty positive examples. To address faulty negatives, we estimate the similarity across instances to compute a soft target distribution over instances. The model is then tasked to match this distribution. As a result, instances with enough evidence of similarity are no longer used as negatives and may even be used as positives.

The contributions of this work are as follows (Figure 2). We identify two sources of training noise in cross-modal learning: instances with weak cross-modal correspondence, which create faulty positives, and the sampling of semantically similar instances as negatives, which create faulty negatives. We show that removing faulty positives and negatives using an oracle can significantly improve the performance of a state-of-the-art xID method [56]. We then propose a mechanism to replace the oracle and a robust cross-modal instance discrimination loss that limits the impact of faulty correspondences. The effectiveness of the proposed method is demonstrated on several downstream tasks.

2. Related work

Self-supervised representation learning aims to learn representations by solving pretext tasks defined from the data alone, i.e. without human annotations. In computer vision, pretext tasks involve reasoning about spatial context [57, 21, 39, 60, 30, 34, 65], temporal context [52, 45, 80, 39, 53, 33, 23, 77, 9, 34, 35, 65], other visual properties such as hue, brightness and flow [20, 43, 86, 44, 87, 74, 64], or clusters of features [10, 7, 11, 78]. One promising technique is the instance discrimination task proposed in [81, 22] and further explored in [36, 51, 14, 78, 82]. However, contrastive learning from a single modality requires heavy data augmentations to generate distinct views. Instead, we focus on cross-modal instance discrimination, which avoids this issue by generating views from different modalities.

Representation learning from audio-visual correspondences: Since, in video, the audio is naturally paired and synced with the visual component, audio-visual correspondences have been used to draw direct supervision for several tasks, such as visually guided-source separation and localization [26, 28, 91, 90, 25, 68], visually guided audio spatialization [55, 27], audio-visual embodied navigation [13], lip-speech synchronization [18] and audio-visual speech recognition [1, 17].

In the context of contrastive learning, audio-visual correspondences are used to generate alternative views of an instance. While this has been known for a long time [19], self-supervised audio-visual representation learning gained popularity in recent years. For example, [5, 4] propose to learn representations by solving a binary classification problem that identifies audio and video clips belonging to the same instance. [41, 58] predict if audio and video clips are temporally synchronized, and [54] predicts if audio and video clips extracted from a 360 video are spatially aligned. [56, 61] improve upon the audio-visual correspondence problem [5] by posing it as a cross-modal instance discrimination task, where instances are contrasted to a large number of negatives. As a result, [56, 61] achieve impressive performance on downstream tasks such as action recognition.

In this work, we address two issues inherent to cross-modal instance discrimination, namely the detrimental impact of faulty positives and negatives. Recently, [3, 8] proposed to learn representations by iteratively clustering the audio and visual representations and seeking to predict cluster assignments from the opposite modality. While clustering can also discourage faulty negatives from acting as repelling forces, our method accomplishes this by optimizing a simple instance discrimination loss with soft targets, thus avoiding the significant computational overhead of clustering.
Supervised learning from noisy labels. Our work is closely related to supervised learning from noisy labels [66, 88, 62, 32, 47]. Since label collection is expensive and time-consuming, scaling human annotation to large datasets often requires the use of non-experts or non-curated labels such as user tags, which are prone to noise. Since deep neural networks can easily overfit to noisy labels [84], this results in poor generalization. Several techniques have been developed to increase the robustness of learning algorithms to label noise, including losses that reduce the impact of outliers [29, 88, 79], loss correction approaches that model the sources of label noise [62, 37, 12, 66, 6, 49, 70], meta-learning procedures that learn how to correct the sources of label noise [47, 67, 46, 69, 89] and regularization procedures tailored to lower the impact of noise [85, 63]. We present procedures to mitigate the detrimental empirical effects of faulty samples.

3. Analysis: Instance Discrimination

We analyze the cross-modal instance discrimination method [56, 74, 61] and show that faulty positives and negatives have a disproportionately large contribution to the training updates. Additionally, in Table 1, we document the detrimental empirical effects of faulty samples.

Cross-Modal Instance Discrimination Consider a dataset \( D = \{ (v_i, a_i) \}_{i=1}^N \) containing \( N \) samples (or instances) of video \( v_i \) and audio \( a_i \). Cross-modal instance discrimination uses a contrastive loss [31] to learn video and audio encoders, \( f_v(\cdot) \) and \( f_a(\cdot) \), so as to align the two modalities belonging to the same instance [74, 56, 61] by minimizing

\[
L_{\text{ID}} (v_i, a_i) = - \log P(\bar{a}_i | v_i; \tau) - \log P(\bar{v}_i | a_i; \tau) \tag{1}
\]

where

\[
P(\bar{t}_j | s_i; \tau) = \frac{\exp(s_i^T t_j / \tau)}{\sum_k \exp(s_i^T t_k / \tau)},
\]

where \( v_i = f_v(v_i) \) and \( a_i = f_a(a_i) \) are visual and audio features normalized to the unit sphere, \( \bar{v}_i \) and \( \bar{a}_i \) are target representations, and \( \tau \) is a temperature hyper-parameter. Prior works differ by the type of target representations employed. For example, \( \bar{v}_i \) and \( \bar{a}_i \) can be entries of a memory bank as in [56, 81], the network representations themselves \( \bar{v}_i = f_v(v_i) \) and \( \bar{a}_i = f_a(a_i) \) as in SimCLR [14], the outputs of momentum encoders as in MoCo [36], or the centroids of an online clustering procedure as in SwAV or CLD [11, 78].

In this work, we build on the Audio-Visual Instance Discrimination (AVID) method of [56], focusing on target representations sampled from a memory bank. However, the principles introduced below can also be applied to SimCLR, MoCo or SwAV style targets.

Faulty positives and negatives in practice. The contrastive loss of Equation 1 is minimized when audio and visual representations from the same instance are aligned (dot-product similarities \( \bar{v}_i^T \bar{a}_i \) and \( \bar{a}_i^T \bar{v}_i \) as close to 1 as possible), and representations from different instances are far apart. In practice, however, the two modalities are not informative of each other for a significant number of instances (see Figure 1). We refer to these as corresponding to faulty positives.\(^1\)

On the other hand, a significant number of contrastive learning negatives are semantically similar to the base instance. We term these semantically similar negatives as faulty negatives since they should ideally be used as positives.

Figure 3 shows the histogram of similarities \( \bar{v}_i^T \bar{a}_i \) after training an audio-visual model with the loss of Equation 1. As can be seen, instances with higher scores tend to have stronger correspondences (i.e. the audio and video signals are informative of each other). Instances where the two modalities are uninformative of each other tend to have lower scores and are generally faulty positives. On the other hand, Figure 4 shows the histograms of similarities between a video \( i \) and negatives \( j \). As can be seen, faulty negatives tend to occur for negatives \( j \) with high similarity \( \bar{v}_j^T \bar{a}_i \).

How do faulty positives and negatives affect learning? Faulty positives and negatives have a disproportionately large contribution to the training updates. To see this, examine the gradients that are computed when optimizing Equation 1. The partial derivatives are given as

\[
- \frac{\partial L_{\text{ID}}}{\partial v_i} = \frac{\bar{a}_i}{\tau} (1 - P(\bar{a}_i | v_i)) - \sum_{n \neq i} \frac{\bar{a}_n}{\tau} P(\bar{a}_n | v_i) \tag{3}
\]

\[
- \frac{\partial L_{\text{ID}}}{\partial a_i} = \frac{\bar{v}_i}{\tau} (1 - P(\bar{v}_i | a_i)) - \sum_{n \neq i} \frac{\bar{v}_n}{\tau} P(\bar{v}_n | a_i). \tag{4}
\]

Intuitively, the target representations \( \bar{v}_i \) and \( \bar{a}_i \) of the instance itself act as ‘attraction points’ for the encoder of the opposing modality, while the target representations of other (negative) instances, \( \bar{v}_n \) and \( \bar{a}_n \), act as ‘repelling points’. For example, in Equation 3, the negative gradient pushes \( \bar{v}_i \) toward \( \bar{a}_i \) and away from \( \bar{a}_n, n \neq i \) in the opposite direction. The attraction forces are weighed by the complement of the prediction confidence, i.e., \( 1 - P(\bar{v}_i | a_i) \) or \( 1 - P(\bar{a}_i | v_i) \). When positive samples are faulty, these gradients lead to noisy training signals. As shown in Figure 3, faulty positives tend to have lower similarities and thus less confident predictions. As a result, the cross-modal loss of Equation 1 assigns stronger gradients to faulty positive samples. On the other hand, the repelling forces of negative instances are also weighted by the likelihood of matching the base sample, i.e., \( P(\bar{v}_n | a_i) \).

\(^1\)We prefer ‘faulty positives’ over ‘false positives’ to distinguish from supervised learning where one has access to labels.
and $P(\tilde{a}_n | v_i)$. However, as shown in Figure 4, faulty negatives tend to have high similarity scores, leading to high posteriors $P(\tilde{v}_n | a_i)$ and $P(\tilde{a}_n | v_i)$. Thus, the targets $\tilde{v}_n$ and $\tilde{a}_n$ of faulty negatives act as strong repelling forces for $a$ and $v$ (see Equation 3–4), even though they should ideally be close in feature space.

4. Robust audio-visual representation learning

We have seen that contrastive learning places too much emphasis on the impossible goals of bringing together the audio-visual components of faulty positives and repelling the feature representations from faulty negatives. We next propose solutions to these two problems.

4.1. Weighted xID: Tackling Faulty Positives

To reduce the impact of faulty positives, we propose to optimize a weighted loss. Let $w_i \in [0, 1]$ be a set of sample weights that identify faulty positives. Robustness is achieved by re-weighting the xID loss of Equation 1

$$L_{RxID} = \frac{\sum_i w_i L_{xID}(v_i, a_i)}{\sum_i w_i}.$$

To estimate sample weights $w_i$, we leverage observations from Figure 3. Since low similarities $\tilde{v}^T a_i$ are indicative of faulty positives, we define the weights $w_i$ to be proportional to the cumulative distribution of these scores. We assume the scores to be normally distributed and define $w_i$ as

$$w_i = t_{w_{\text{min}}} \left( C_N' (\tilde{a}_i^T \tilde{v}_i; \mu + \delta \sigma, \kappa \sigma^2) \right),$$

where $\mu$ and $\sigma^2$ are the sample mean and variance of the scores, $C_N'$ is the cumulative distribution of a transformed normal distribution $N(\mu + \delta \sigma, \kappa \sigma^2)$, and $t_{w_{\text{min}}}(x) = x \cdot (1 - w_{\text{min}}) + w_{\text{min}}$ is a soft truncation function used to assign a non-zero weight $w_{\text{min}}$ to low score instances. $\delta$, $\kappa$ and $w_{\text{min}}$ are shape hyper-parameters that provide flexibility to the weight function, adjusting the location and rate of decay of the weights. Figure 5 shows how the weighting function varies with the shape hyper-parameters $\delta$, $\kappa$ and $w_{\text{min}}$. 

As observed in §3, faulty negatives are overemphasized during training. The underlying reason is that the xID loss of Equation 1 has too strict a definition of negatives: every negative instance $j \neq i$ is considered ‘equally negative.’ To limit the impact of faulty negatives, we introduce a ‘softer’ definition by introducing soft targets $T(j|i)$, based on the similarity between instance $i$ and negative $j$. We then minimize a soft-xID loss

$$L_{\text{Soft-xID}}(v_i, a_i) = - \sum_j T(v_j|i) \log P(a_i|v_j; \tau) - \sum_j T(a_j|i) \log P(v_i|a_j; \tau)$$

where $1_{i=j}$ is the one-hot targets of vanilla xID, $S_a$ and $S_v$ are softening scores (described next) used to adjust the one-hot targets, and $\lambda \in [0, 1]$ is a mixing coefficient that weights the two terms. Equations 1 and 7 are identical when $\lambda = 0$. Since $T(j|i)$ is no longer strictly zero for similar instances, minimizing Equation 7 reduces the force to repel faulty negatives and thus their impact.

Estimating softening scores $S$. Since our approach focuses on self-supervised learning, we must estimate the softening

![Figure 3: Faulty positives in a pretrained cross-modal model. Histogram of similarity scores $\tilde{v}_j^T a_i$ between video and audio representations, and examples obtained at various points of the distribution. We describe both the sound and video content in the plot. Examples with lower similarity scores contain audio that is less predictive of the video content, which creates faulty positives for training.](image)

![Figure 4: Faulty negatives in a pretrained cross-modal model.](image)

![Figure 5: Weights as function of similarity scores $\tilde{v}_j^T a_i$ for different values of shape parameters $\delta$, $\kappa$ and $w_{\text{min}}$. Parameters $\mu$, $\sigma$ are automatically determined from the histogram of similarity scores $\tilde{v}_j^T a_i$ (shown in red).](image)
scores $S$ automatically, *i.e.*, without class labels. We describe multiple strategies for estimating these values and illustrate them in Figure 6.

- **Bootstrapping** [66] is a well established procedure to create soft targets. It uses the model’s own predictions (posteriors) as the softening scores, *i.e.*,

$$S_v(j|i) = P(\hat{a}_j|\hat{v}_i; \tau_s)$$

where $\tau_s$ controls the peakiness of the distribution. However, bootstrapping computes the target distribution without aggregating information from any other source other than each model’s own posterior.

- **Swapped prediction** improves upon bootstrapping by using the posteriors of the opposite modality, *i.e.*, the softening scores $S_v$ for the video modality are computed using the posterior of the audio encoder and vice-versa,

$$S_v(j|i) = P(\hat{v}_j|\hat{a}_i; \tau_v)$$

As a result, in addition to the instance itself, the model is asked to predict which other instances are deemed similar in the opposite modality.

- **Neighbor prediction** relies on within-modal relationships to estimate the similarity between instances, thus avoiding potential mismatched audio and visual modalities when computing the soft targets. Specifically, we define

$$S_v(j|i) = \rho(\bar{v}_j^T \bar{v}_j/\tau_v)$$

where $\rho$ is the softmax operator.

- **Cycle consistent prediction** improves upon ‘swapped prediction’ by focusing on negatives that are good correspondences themselves, *i.e.*, negatives with high similarity scores $\bar{v}_j^T \bar{a}_j$. In this case, we define

$$S_v(j|i) = \rho(\bar{v}_j^T \bar{a}_i/\tau_i + \bar{a}_j^T \bar{v}_j/\tau_v + \bar{v}_j^T \bar{a}_j/\tau_v)$$

$$S_a(j|i) = \rho(\bar{a}_j^T \bar{v}_i/\tau_i + \bar{v}_j^T \bar{a}_j/\tau_v + \bar{a}_j^T \bar{v}_j/\tau_v)$$

where $\tau_v$ and $\tau_i$ control the relative importance of swapped prediction target and avoiding negatives with weak correspondences. As shown in Figure 6, the terms $\bar{v}_j^T \bar{a}_i$ and $\bar{v}_j^T \bar{a}_j$ complete a cycle over instances $i$ and $j$.

**How do soft targets mitigate faulty negatives?** The soft xID loss of Equation 7 prevents overemphasizing faulty negatives by relying on soft targets $T(j|i)$ that encode similarities between instances. To better understand the mechanism, we examine the partial derivatives of the soft-xID loss:

$$-\frac{\partial L_{\text{Soft-xID}}}{\partial \bar{v}_i} = \sum_j \bar{a}_j (T_v(j|i) - P(\hat{a}_i|\hat{v}_i))$$

$$-\frac{\partial L_{\text{Soft-xID}}}{\partial \bar{a}_i} = \sum_j \bar{v}_j (T_a(j|i) - P(\hat{v}_i|\hat{a}_i)).$$

Since faulty negatives $j$ tend to be similar to the base instance $i$, the soft targets $T(j|i)$ are higher. Thus, the target representations $\bar{v}_j$ and $\bar{a}_j$ of faulty negatives act as weaker negatives, or even as positives when $T(j|i)$ is larger than the model posteriors.

4.3. Training

We introduced two procedures to deal with noisy training signals inherent to cross-modal instance discrimination. §4.1 presents a weighting mechanism that limits the effect of faulty positives, while §4.2 proposes a soft instance discrimination loss that predicts relations between instances, thus preventing the training algorithm from overemphasizing faulty negatives. Since both procedures rely on the alignment between audio and visual target representations to find weak correspondences, we start by training the model for cross-modal instance discrimination alone (Equation 1). After the initial warmup stage, the two procedures can be combined by minimizing

$$L = \frac{1}{\sum_k w_k} \sum_i w_i L_{\text{Soft-xID}}(\bar{v}_i, \bar{a}_i)$$

where $w_i$ are the sample weights of Equation 6 and $L_{\text{Soft-xID}}$ is the xID loss with soft targets of Equation 7.

5. Experiments

We perform experiments to better understand cross-modal learning and validate the proposed improvements. We pre-train models on a subset of the Kinetics-400 [76] dataset containing 50K videos and evaluate the pretrained models by transfer learning.

5.1. Experimental Setup

**Video and audio preprocessing.** During training, we extract video clips of length $T = 8$ frames and resolution $80 \times 80$ at 16 fps. Video clips are augmented using temporal jittering, multi-scale cropping, horizontal flipping, color jittering, gray-scaling, and Gaussian blur [14]. All data augmentations are applied consistently over all frames. For the audio, we extract mono clips of length 2s at a sample rate of 11025Hz, and compute log spectrograms on 50ms windows with a hop size of 25ms. The spectrogram is then converted...
to a mel scale with 80 bands, yielding an audio input of size $80 \times 80$. Audio data is augmented by randomly changing the volume by at most 20%.

**Video and audio models.** The video encoder is a 9-layer version of the R(2+1)D model of [75]. Following [5, 56], we replaced global average pooling with max pooling. The audio encoder is a 9-layer 2D ConvNet with batch normalization and global max pooling. Both encoders yield 512-dimensional features, which are mapped into a 128-dimensional sphere using a non-linear projection head (as in [14]) followed by L2 normalization.

**Pretraining.** In the warm-up stage, the video and audio models are trained to optimize the loss of Equation 1 using the Adam optimizer [40] with default hyper-parameters ($\beta_1 = 0.9$ and $\beta_2 = 0.999$) for $400$ epochs with a learning rate of $1e-4$ and a batch size of $224$ split over $2$ 12Gb GPUs. In order to reduce the memory footprint of our models, we employ mixed-precision training [50] using PyTorch AMP [59]. Following [56, 81], the audio and video target representations, $\hat{a}$ and $\hat{v}$, are generated using memory banks updated by exponential moving average with an update constant of $0.5$. The contrastive loss of Equation 1 is defined by opposing the target representation of the opposite modality to $1024$ negatives randomly drawn from the memory bank. The temperature hyper-parameter is set to $\tau = 0.07$.

After the initial warm-up stage, models are trained for an additional $200$ epochs to optimize the loss of Equation 17 using the Adam optimizer and a cosine learning rate schedule starting at $1e-4$ and ending at $1e-5$. The hyper-parameters for the weighting function (Equation 6) and the soft xID loss (Equation 7) are discussed below. To provide a fair comparison to the AVID baseline [56], we control for the number of epochs by training the baseline model for an additional $200$ epochs as well.

**Downstream tasks.** We evaluate audio and video features using transfer learning. Video features are evaluated on the UCF [72] and HMDB [42] datasets. Models are fine-tuned using 8-frame clips for $200$ epochs using the Adam optimizer with a batch size of $192$ on a single GPU and a cosine learning rate schedule starting at $1e-4$ and ending at $1e-5$. To prevent overfitting, we use dropout after the global max-pooling layer, weight decay of $1e-3$, and reduced the learning rate for backbone weights by a factor of $10$. At test time, top-1 accuracy is measured on video level predictions computed by averaging the predictions of $10$ clips uniformly sampled over the entire video.

Following [8, 83], we also evaluate the quality of video representations by conducting retrieval experiments without fine-tuning. Feature maps of size $4 \times 4 \times 512$ are extracted from $10$ clips per video and averaged. We then use videos in the test set to query the training set. As in [8, 83], a correct retrieval occurs when the class of one of the top-$k$ retrieved videos matches the query, and performance is measured by the average top-$k$ retrieval performance ($R@K$).

**5.2. Weighted cross-modal learning**

We analyze the impact of faulty positives on the representations learned by cross-modal instance discrimination.

**Faulty positives are detrimental to representation learning.** We artificially control the number of faulty positives to assess their impact on representation learning. The pretraining dataset already contains an unknown (but significant) number of faulty positives. We increase this number by injecting more faulty positives. A faulty positive is injected by replacing the audio of an instance with a randomly selected audio that is not part of the training set. After pretraining, the learned visual representation is evaluated on the UCF and HMDB datasets using both classification and retrieval protocols. Figure 7 shows that as the fraction of faulty positives increases, the transfer performance of cross-modal instance discrimination (xID) decreases significantly.

**Weighted xID reduces the impact of faulty positives.** We evaluate the effectiveness of the weighted xID loss (Equation 5) as a function of the number of faulty positives. We compare the representations learned by Weighted-xID to its unweighted counterpart (xID), as well as an oracle weight function (Oracle-xID) which assigns $w_i = 0$ to artificially altered instances and $w_i = 1$ otherwise. The weight function of Equation 5 is defined with $\kappa = 0.5$ and $w_{\text{min}} = 0.25$. For simplicity, we assume that the noise level is known and set $\delta$ in Weighted-xID so that the midpoint of the weighting function coincides with the known fraction of altered samples. In practice, the noise level would need to be estimated either by cross-validation or by manual inspection. Weighted-xID is not very sensitive to these parameters (see appendix).

Figure 7 shows the performance of the three approaches. Oracle-xID consistently outperforms xID when the fraction...
of injected faulty positives is high. This shows that the detrimental impact of noisy correspondences can be mitigated with a weighting strategy. Weighted-xID also outperforms the unweighted version (xID) in nearly all cases, with larger margins for larger fractions of noisy correspondences. In fact, Weighted-xID even outperforms the oracle weight function, especially at lower noise levels. This is because the original Kinetics dataset already contains a significant amount of weak correspondences, which the oracle weight function treats as clean $w_i = 1$, while the weighting function of Equation 6 can suppress them.

### 5.3. Instance discrimination with soft targets

To limit the impact of faulty negatives, we proposed to match a soft target distribution that encodes instance similarity. We analyze different design decisions for creating the soft targets and their effect on transfer performance.

**Comparison of strategies for computing targets** As summarized in Figure 6, the soft target distributions can be computed by aggregating evidence from all modalities. Four different strategies were proposed, bootstrapping, swapped or cycle consistent assignments. Models were trained to minimize the loss of Equation 7 with $\lambda = 0.5$. We empirically found that peaker target distributions work better, and set the temperature parameter $\tau$ to 0.02. For cycle consistent assignments, the terms $\bar{v}_j a_j$ are used so as to focus on negatives that are good correspondences themselves. A temperature hyper-parameter of $\tau = 0.07$ was sufficient to impose such constraint. Beyond the baseline xID, we also compare to an oracle target distribution that has access to class labels to determine the similarity between instances. Specifically, the oracle considers two instances $i$ and $j$ to be similar if they share the same class label, and computes $T_v(j|i)$ and $T_a(j|i)$ by assigning a uniform distribution over similar instances, and 0 to non-similar ones.

Table 1 shows the performance of different target distributions. We observe a large gap between vanilla xID and xID with an oracle soft target, which demonstrates the detrimental effect of faulty negatives. In the self-supervised case, however, labels are not available for determining the target distribution. Nevertheless, the estimated target distributions (bottom four rows) still significantly improve over the xID loss. Regarding the various types of target distributions, bootstrapping is the least effective. This is expected since, in this case, the target distribution is a peakier version of the model posterior, i.e., it is obtained without aggregating information from any other sources. Cycle consistent prediction is the most effective most often. This is because cycle consistent prediction not only leverages the opposite modality to create the target distribution, but it also avoids targets that are not good correspondences themselves, i.e., avoids samples with low cross-modal similarities.

### 5.4. Robust instance discrimination with soft targets

Sample weighting and soft targets are designed to address two different sources of noisy training signals inherent to cross-modal contrastive learning: faulty positives and faulty negatives. Table 2 shows that the two proposed improvements (Weighted-xID and Soft-xID) not only improve upon the representations of vanilla xID, they are also complementary to each other. By combining the two approaches using the loss of Equation 17, Robust-xID improved upon Weighted and Soft-xID.

### 6. Comparison to prior work

We compare Robust-xID to prior work in self-supervised learning. We train our models on the Kinetics dataset, using an 18-layer R(2+1)D model [75] for the video, and a 9-layer 2D ConvNet with batch normalization for the audio. Video clips of length 8-frames and $112 \times 112$ resolution are extracted at 16fps, and the same data augmentations from §5 are used. We extract audio clips of length 2s at 24KHz and compute log mel spectrograms with 128 time steps and 128 frequency bands. All models are trained with the Adam optimizer with a batch size of 512 distributed across 8 12Gb GPUs. We warm-up the models for 200 epochs by training on the xID loss alone with a learning rate of $5e^{-4}$. The models are then trained with sample weights and cycle consistent soft targets for an additional 200 epochs using a cosine learning rate schedule from $5e^{-4}$ to $5e^{-5}$.

After pre-training, models are evaluated on UCF and HMDB. We fine-tune the models using either 8 or 32 frame clips for action recognition and report the top-1 accuracy of video level predictions (with 10 clips per video) in Ta-
Table 3: Comparison to prior work (finetuning). Performance on the downstream UCF and HMDB datasets by full network fine-tuning after pre-training on Kinetics. We report top-1 accuracy of video level predictions (10 clips per video). We also list the video encoder, amount of compute used for pre-training and the fine-tuning resolution.

<table>
<thead>
<tr>
<th>Method</th>
<th>Model</th>
<th>Compute # GPUs</th>
<th>Finetuning Resolution</th>
<th>UCF</th>
<th>HMDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPC [33]</td>
<td>S3D</td>
<td>4</td>
<td>25 x 128^2</td>
<td>75.7</td>
<td>35.7</td>
</tr>
<tr>
<td>CBT [73]</td>
<td>S3D</td>
<td>16</td>
<td>11 x 12^2</td>
<td>79.5</td>
<td>44.6</td>
</tr>
<tr>
<td>Multisensory [58]</td>
<td>3D-ResNet18</td>
<td>3</td>
<td>32 x 221^2</td>
<td>82.1</td>
<td>–</td>
</tr>
<tr>
<td>AVTS [41]</td>
<td>MC3-18</td>
<td>4</td>
<td>25 x 221^2</td>
<td>84.1</td>
<td>52.5</td>
</tr>
<tr>
<td>SeLaVi [8]</td>
<td>R(2+1)D-18</td>
<td>64</td>
<td>32 x 112^2</td>
<td>83.1^*</td>
<td>47.1^*</td>
</tr>
<tr>
<td>XDC [3]</td>
<td>R(2+1)D-18</td>
<td>64</td>
<td>8 x 221^2</td>
<td>74.2^*</td>
<td>39.0^*</td>
</tr>
<tr>
<td>AVID-CMA [56]</td>
<td>R(2+1)D-18</td>
<td>64</td>
<td>32 x 221^2</td>
<td>86.8^*</td>
<td>52.6^*</td>
</tr>
<tr>
<td>GDT [61]</td>
<td>R(2+1)D-18</td>
<td>64</td>
<td>8 x 221^2</td>
<td>83.7^*</td>
<td>49.5^*</td>
</tr>
<tr>
<td>xID+CMA [56]</td>
<td>R(2+1)D-18</td>
<td>64</td>
<td>32 x 221^2</td>
<td>87.5^*</td>
<td>60.8^*</td>
</tr>
<tr>
<td>Robust-xID</td>
<td>R(2+1)D-18</td>
<td>8</td>
<td>8 x 112^2</td>
<td>80.6</td>
<td>48.6</td>
</tr>
<tr>
<td></td>
<td>Robust-xID</td>
<td>8</td>
<td>32 x 112^2</td>
<td>84.9</td>
<td>54.7</td>
</tr>
</tbody>
</table>

* Models pre-trained with more than one compute node (8 GPUs).

Table 4: Retrieval performance on UCF and HMDB datasets after pre-training on Kinetics for different numbers of retrieved neighbors.

<table>
<thead>
<tr>
<th>Method</th>
<th>UCF</th>
<th>HMDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>SpeedNet [9]</td>
<td>R@1: 13.0 R@5: 28.1 R@20: 49.5</td>
<td>R@1: - R@5: - R@20: -</td>
</tr>
<tr>
<td>VCP [40]</td>
<td>18.6 33.6 53.5</td>
<td>7.6 24.4 53.6</td>
</tr>
<tr>
<td>VSP [16]</td>
<td>24.6 41.9 76.9</td>
<td>10.3 26.6 54.6</td>
</tr>
<tr>
<td>CoCLR [35]</td>
<td>55.9 70.8 82.5</td>
<td>26.1 45.8 87.9</td>
</tr>
<tr>
<td>SeLaVi [8]</td>
<td>52.0 68.6 84.5</td>
<td>24.8 47.6 75.5</td>
</tr>
<tr>
<td>GDT [61]</td>
<td>57.4 73.4 88.1</td>
<td>25.4 51.4 75.0</td>
</tr>
<tr>
<td>xID+CMA [56]</td>
<td>60.1 76.6 90.1</td>
<td>29.7 54.4 77.1</td>
</tr>
<tr>
<td>Robust-xID</td>
<td>60.9 79.4 90.8</td>
<td>30.8 55.8 79.7</td>
</tr>
</tbody>
</table>

Table 5: Few-shot learning on UCF and HMDB after pre-training on Kinetics. Classification is conducted using a one-vs-all SVM trained on the pretrained features of n images per class. Top-1 accuracy is reported for n ∈ {1, 5, 20}.

<table>
<thead>
<tr>
<th>Method</th>
<th>UCF</th>
<th>HMDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>D3-ResNet [38]</td>
<td>15.0 31.5 47.1</td>
<td>-  -  -</td>
</tr>
<tr>
<td>GDT [61]</td>
<td>26.3 42.4 49.4</td>
<td>13.4 15.6 20.8</td>
</tr>
<tr>
<td>xID+CMA [56]</td>
<td>30.8 53.1 66.9</td>
<td>13.5 25.0 33.6</td>
</tr>
<tr>
<td>Robust-xID</td>
<td>32.8 54.6 67.8</td>
<td>14.1 25.9 34.9</td>
</tr>
</tbody>
</table>

R@K for K = 1, K = 5 and K = 20 neighbors in Table 4. The retrieval protocol was described in §5. Following [38, 61], we also assessed the few-shot learning performance of Robust-xID models on UCF and HMDB. For the few-shot evaluation, we average the pretrained max-pooling features of 10 clips per video. The features from n videos per class are then used to learn a one-vs-all linear SVM classifier with C = 1. We report the top-1 accuracy averaged over 50 trials in Table 5. On both the retrieval and few-shot learning tasks, Robust-xID improves significantly over all prior work, reaffirming the importance of mitigating the training noise introduced by faulty positives and faulty negatives.

7. Discussion and future work

We identified and tackled two significant sources of noisy training signals in audio-visual instance discrimination, namely instances with weak audio-visual correspondence (or faulty positives) and semantically similar negatives (or faulty negatives). We showed the impact of faulty correspondences on representation learning by removing them using an oracle with access to ground-truth annotations. We then proposed a method that mitigates the impact of faulty correspondences without relying on ground-truth annotations. Extensive analysis and experimental evaluations show that the proposed procedure enhances representation learning and improves transfer performance significantly.

Our findings show that cross-modal learning should be seen as a problem of learning with noisy targets. While we propose two specific methods to address faulty positives and faulty negatives (i.e. weighting and soft targets), there is a rich literature regarding supervised learning with noisy labels. Developing methods that tackle noisy correspondences are a promising avenue for future research. Furthermore, we focused on audio-visual learning, but other pairs of modalities such as RGB and flow or text from instructional videos also present similar problems. We believe that our method will also benefit cross-modal learning from other modalities.

Acknowledgements This work was partially funded by NSF awards IIS-1924937, IIS-2041009, and gifts from Amazon and Qualcomm. We also acknowledge and thank the use of the Nautilus platform for some of the experiments above.
References


[47] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In Int. Conf. Computer Vision (ICCV), 2017. 3


[49] Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In Int. Conf. on Machine Learning (ICML), 2018. 3


[53] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In Int. Conf. on Machine Learning (ICML), 2009. 2


[56] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement. In IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2021. 1, 2, 3, 6, 7, 8


[61] Mandela Patrick, Yuki M Asano, Ruth Fong, João F Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal


[80] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In Int. Conf. on Machine Learning (ICML), 2017. 3


A. Parametric studies

We provide a parametric study of key Robust-xID hyper-parameters.

**Weight function shape parameter** \( \delta \) One critical parameter of Weighted-xID is the shape parameter \( \delta \), which specifies the mid-point location of the weight function. For example, when \( \delta = -2 \), the midpoint is located at \( \mu - 2\sigma \) where \( \mu \) and \( \sigma \) are the sample mean and standard deviation of the scores \( v_i^T a_i \). This means that for \( \delta = -2 \), the majority of samples will have a weight of 1, and only a small fraction will have a weight close to \( w_{\text{min}} \). As \( \delta \) increases, the proportion of samples that are down-weighted also increases. To study the impact of \( \delta \), we trained several models using Weighted-xID with different values of \( \delta \) and for different amounts of injected faulty positives \( n_0 \). Other hyper-parameters were kept at their default values \( w_{\text{min}} = 0.25 \) and \( \kappa = 0.5 \). The transfer performance is shown in Figure 8. As can be seen, the proposed robust xID procedure is not very sensitive to this hyper-parameter. This suggests that Robust-xID can help representation learning as long as clear faulty positives are suppressed.

**Soft-xID: Mixing coefficient** The mixing coefficient \( \lambda \) specifies the degree to which the one-hot targets of instance discrimination are softened in Soft-xID. The one-hot instance discrimination targets are used when \( \lambda = 0 \). As \( \lambda \) increases, the softening scores \( S(j|i) \) are increasingly used to adjust the one-hot targets. To study the impact of the mixing coefficient \( \lambda \), we trained several models using Soft-xID with various values of \( \lambda \). Cycle consistent targets were used as the softening strategy. Figure 9 shows the transfer performance of the learned models on UCF and HMDB under the fine-tuning and retrieval protocols. The trend is consistent across the two datasets and two evaluation protocols. Softening the instance discrimination targets enhances representation learning, with the optimal performance achieved with a mixing coefficient between 0.25 and 0.5. However, as the mixing coefficient increases substantially \( \lambda > 0.65 \), the targets are derived from the model prediction alone and disregard instance labels. In this case of large \( \lambda \), the pre-training fails completely, *i.e.*, the learned representations have very low transfer performance.

B. Additional analysis

The proposed approach learns high-quality feature representations that can be used to discriminate several action classes. This was shown in the main paper by reporting transfer learning results. We now provide additional qualitative evidence and analysis.

**Retrieval** For each video, we extracted \( 4 \times 4 \times 512 \) feature maps from the video encoder learned using Robust-xID on the full Kinetics dataset. Figure 11 depicts a list of Kinetics classes from a video encoder learned using Robust-xID on the full Kinetics dataset. Figure 10 depicts the top 4 closest videos for several query samples. As can be seen, Robust-xID produces highly semantic features, enabling correct retrievals for a large number of videos spanning a large number of classes. Furthermore, even when a video of a different class is retrieved, the errors are intuitive (for example, the confusion between ‘American football’ and ‘Hurling’ in the third row). Failure cases also seem to be correlated with classes that are hard to distinguish from the audio alone (e.g., different types of kicking sports or swimming strokes).

**Class-based analysis** To better understand which classes are better modeled by the Robust-xID framework, we measured the top-1 retrieval performance (\( R@1 \)) averaged across all images of each class. Similar to the analysis above, each video is represented by a \( 4 \times 4 \times 512 \) feature map extracted from a video encoder learned using Robust-xID on the full Kinetics dataset. Figure 10 depicts a list of Kinetics classes sorted by their average \( R@1 \) score. As can be seen, action classes which are often accompanied by long and distinctive sounds (*e.g.*, squash, harp, drums, accordion, or scuba...
Faulty positive detection performance To obtain a rough estimate of performance of the faulty positive detection procedure, we randomly sampled 100 videos from the 10000 most likely faulty positives, as identified by Robust-xID trained on the full Kinetics dataset. We then manually labeled them according to how related their audio and visual signals are. From those, 67 were clear faulty pairs; 24 contained narrative voice-overs (i.e., required natural language understanding to link the two modalities); and 9 samples were clearly misidentified.
Figure 11: Retrievals. In each row, the first image depicts the query video, and the following four images depict the top 4 retrievals. The corresponding Kinetics class description is provided above each frame. Each video is represented by a $4 \times 4 \times 512$ feature map produced by the video encoder learned using Robust-xID on the full Kinetics dataset. Euclidean distance is used to determine video similarity.