Labelling unlabelled videos
from scratch with multi-modal self-supervision

Yuki M. Asano[1*]   **Mandela Patrick**[1,2*]   **Christian Rupprecht**[1]   **Andrea Vedaldi**[1,2]

[1] Visual Geometry Group, University of Oxford
yuki@robots.ox.ac.uk
[2] Facebook AI Research
mandelapatrick@fb.com

Abstract

A large part of the current success of deep learning lies in the effectiveness of data – more precisely: labelled data. Yet, labelling a dataset with human annotation continues to carry high costs, especially for videos. While in the image domain, recent methods have allowed to generate meaningful (pseudo-) labels for unlabelled datasets without supervision, this development is missing for the video domain where learning feature representations is the current focus. In this work, we a) show that unsupervised labelling of a video dataset does not come for free from strong feature encoders and b) propose a novel clustering method that allows pseudo-labelling of a video dataset without any human annotations, by leveraging the natural correspondence between the audio and visual modalities. An extensive analysis shows that the resulting clusters have high semantic overlap to ground truth human labels. We further introduce the first benchmarking results on unsupervised labelling of common video datasets Kinetics, Kinetics-Sound, VGG-Sound and AVE[2].

1   Introduction

One of the key tasks in machine learning is to convert continuous perceptual data such as images and videos into a symbolic representation, assigning discrete labels to it. This task is generally formulated as *clustering* [31]. For images, recent contributions such as [6, 13, 37, 72] have obtained good results by combining clustering and representation learning. However, progress has been more limited for videos, which pose unique challenges and opportunities. Compared to images, videos are much more expensive to annotate; at the same time, they contain more information, including a temporal dimension and two modalities, aural and visual, which can be exploited for better clustering. In this paper, we are thus interested in developing methods to *cluster video datasets without manual supervision*, potentially reducing the cost and amount of manual labelling required for video data.

Just as for most tasks in machine learning, clustering can be greatly facilitated by extracting a suitable *representation* of the data. However, representations are usually learned by means of manually supplied labels, which we wish to avoid. Inspired by [79], we note that a solution is to consider one of the recent state-of-the-art self-supervised representation learning methods and apply an off-the-shelf clustering algorithm *post-hoc*. With this, we show that we can obtain very strong baselines for clustering videos.

---

[*]Joint first authors

[2]Code will be made available at `https://github.com/facebookresearch/selavi`

Still, this begs the question of whether even better performance could be obtained by *simultaneously learning to cluster and represent* video data. Our main contribution is to answer this question affirmatively and thus to show that *good clusters do not come for free from good representations.*

In order to do so, we consider the recent method SeLa [6], which learns clusters and representations for still images by solving an optimal transport problem, and substantially improve it to work with multi-modal data. We do this in three ways. First, we relax the assumption made in [6] that clusters are equally probable; this is not the case for semantic video labels, which tend to have a highly-skewed distribution [1, 29, 41], and extend the algorithm accordingly. Second, we account for the multi-modal nature of video data, by formulating the extraction of audio and visual information from a video as a form of data augmentation, thus learning a clustering function which is invariant to such augmentations. For this to work well, we also propose a new initialization scheme that synchronizes the different modalities before clustering begins. This encourages clusters to be more abstract and thus 'semantic' and learns a redundant clustering function which can be computed robustly from either modality (this is useful when a modality is unreliable, because of noise or compression). Third, since clustering is inherently ambiguous, we propose to learn multiple clustering functions in parallel, while keeping them orthogonal, in order to cover a wider space of valid solutions.

With these technical improvements, our method for Self-Labelling Videos (SeLaVi) substantially outperforms the post-hoc approach [79], SeLa [6] applied to video frames, as well as a recent multi-modal clustering-based representation learning method, XDC [2]. We evaluate our method by testing how well the automatically learned clusters match manually annotated labels in four different video datasets: VGG-Sound [17], AVE [68], Kinetics [41] and Kinetics-Sound [3]. We show that our proposed model results in substantially better clustering performance than alternatives. For example, our method can perfectly group 32% of the videos in the VGG-Sound dataset and 55% in the AVE dataset without using any labels during training. Furthermore, we show that, while some clusters do not align with the ground truth classes, they are generally semantically meaningful (e.g. they contain similar background music) and provide an interactive cluster visualization[3].

In a nutshell, our key contributions are: **(i)** establishing video clustering benchmark results on four datasets for which labels need to be obtained in an unsupervised manner; **(ii)** developing and assessing several strong clustering baselines using state-of-the-art methods for video representation learning, and **(iii)** developing a new algorithm tailored to clustering multi-modal data resulting in state-of-the-art highly semantic labels.

2 Related work

Unsupervised labelling for images. Early approaches to clustering images include agglomerative clustering [9] and partially ordered sets of hand-crafted features [10], while more recent methods combine feature learning with clustering. First, there are methods which propose to implicitly learn a clustering function by maximizing mutual information between the image and nuisance transformations [35, 37]. Second, there are methods which use explicit clustering combined with representation learning [6, 13, 14, 16, 48, 77, 80]. Lastly, there are methods which build on strong feature representations and, at a second stage, utilize these to obtain clusters [47, 72, 79].

Representation learning from videos. There is a growing literature on representation learning from videos. Many of these methods are *uni-modal*, leveraging works from the image domain [5, 8, 18, 26, 27, 56, 57, 69, 75, 81], such as predicting rotations [39] and 3D jigsaw puzzles [42]. Other works leverage temporal information explicitly and predict future features [30], the order of frames [46, 53] and clips [78], the direction of time [74] or the framerate [11, 19]. However, videos usually contain multiple modalities, such as audio, speech and optical flow. *Multi-modal* learning, originally proposed by de Sa [22], has seen a resurgence with the goal of learning strong feature representations that can be used for finetuning on downstream tasks. Most works leverage audio-visual semantic correspondence [3, 7, 59, 60] or the synchronized timing of content [43, 58] between the audio and visual streams. Some works use this information to obtain within-clip sound localisation [4, 34, 58, 63, 64, 82] as well as audio-separation [15, 25]. Other methods use a modality distillation framework to learn video encoders from other modalities [59, 61]. In [61], a loss function is meta-learned by computing common self-supervised losses and distilling these and clustering is

---
[3]`https://www.robots.ox.ac.uk/~vgg/research/selavi`

Figure 1: **Our model** views modalities as different *augmentations* and produces a multi-modal clustering of video datasets from scratch that can closely match human annotated labels.

used as an evaluation metric for meta-learning. New methods have started to learn even stronger representations by using ASR generated text from videos as another modality [49, 52, 55, 66, 67].

Clustering videos.   Perhaps the simplest way of combining representation learning and clustering in videos is to apply *post-hoc* a clustering algorithm after pretraining a representation. In Cluster-Fit [79], the authors show that running a simple k-means algorithm on the features from a pretrained network on the pretraining dataset yields small but consistent gains for representation learning when these clusters are used as labels and the networks are retrained. While in [79], the authors found the optimal number of clusters to consistently be at least one order of magnitude higher than the number of ground-truth labels, we investigate applying this method on various pretrained methods as baselines for our task of labelling an unlabelled video dataset. Specifically, we apply k-means on state-of-the-art single modality models such as DPC [30] and MIL-NCE [52], as well the multi-modal model XDC [2], which itself uses k-means on the audio and visual streams to learn representations. However, they do this as a pretext task for representation learning and obtain separate clusters for audio and video. In contrast, our goal is multi-modally labelling an unlabelled dataset, and we find that our method works significantly better at this task.

3   Method

Given a dataset $D = \{\boldsymbol{x}_i\}_{i \in \{1,\dots,N\}}$ of multi-modal data $\boldsymbol{x}_i$, our goal is to learn a labelling function $y(\boldsymbol{x}) \in \{1,\dots,K\}$ without access to any ground-truth label annotations. There are two requirements that the labelling function must satisfy. First, the labels should capture, as well as possible, the *semantic content* of the data, in the sense of reproducing the labels that a human annotator would intuitively associate to the videos. As part of this, we wish to account for the fact that semantic classes are not all equally probable, and tend instead to follow a Zipf distribution [1, 41]. We then evaluate the quality of the discovered labels by matching them to the ones provided by human annotators, using datasets where ground-truth labels are known. The second requirement is that the labelling method should not overly rely on a single modality. Instead, we wish to treat each modality *as equally informative* for clustering. In this way, we can learn a more robust clustering function, which can work from either modality. Furthermore, correlating of modalities has been shown to be a proxy to learn better abstractions [4, 43, 58, 60].

While our method can work with any number of data modalities (vision, audio, depth, textual transcripts, . . .), we illustrate it under the assumption of video data $\boldsymbol{x} = (a, v)$, comprising an audio stream a and a visual stream v. The following two sections describe our method in detail and show how it meets our requirements.

3.1   Non-degenerate clustering via optimal transport

In this section, we briefly summarize the formulation of [6] to interpret clustering as an optimal transport problem. SeLa [6] is a method that learns representations via clustering images. The labelling function can be expressed as the composition $y(\Psi(\boldsymbol{x}))$, where $\boldsymbol{z} = \Psi(\boldsymbol{x})$ is a data representation (i.e. a feature extractor implemented by a deep neural network), and $y(\boldsymbol{z}) \in \{1,\dots,K\}$ operates on top of the features rather than the raw data.

Any traditional clustering algorithm, such as k-means or Gaussian mixture models, defines an energy function $E(y)$ that, minimized, gives the best data clustering function y. When the representation is accounted for, the energy $E(y, \Psi)$ is a function of both y and Ψ, and we may be naïvely tempted to

3

optimize over both. However, this is well known to yield unbalanced solutions, which necessitates ad-hoc techniques such as non-uniform sampling or re-initialization of unused clusters [13, 14]. Theoretically, in fact, for most choices of E, the energy is trivially minimized by the representation Ψ that maps all data to a constant.

Asano et al. [6] address this issue by constraining the marginal probability distributions of the clusters to be uniform, and show that this reduces to an optimal transport problem. The algorithm then reduces to alternating the fast Sinkhorn-Knopp algorithm [21] for clustering, and standard neural network training for representation learning. To do this, one introduces the cross-entropy loss $E(q, p)$, between the labels given as one-hot vectors in q (i.e. $q(y(\boldsymbol{x})) = 1 \ \forall \boldsymbol{x}$) and the softmax outputs p of a network Ψ:

$$E(p, q) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{y=1}^{K} q(y|\boldsymbol{x}_i) \log p(y|\boldsymbol{x}_i), \quad p(y|\boldsymbol{x}_i) = \text{softmax } \Psi(\boldsymbol{x}_i), \quad (1)$$

where K is the number of clusters. This energy is optimized under the constraint that the marginal cluster probability $\sum_{i=1}^{N} \frac{1}{N} p(y|\boldsymbol{x}_i) = \frac{1}{K}$ is constant (meaning all clusters are a-priori equally likely). Note that minimizing E with respect to p is the same as training the deep network Ψ using the standard cross-entropy loss.

Next, we show that minimizing $E(p, q)$ w.r.t. the label assignments q results in an optimal transport problem. Let $P_{yi} = p(y|\boldsymbol{x}_i)\frac{1}{N}$ be the $K \times N$ matrix of joint probabilities estimated by the model and $Q_{yi} = q(y|\boldsymbol{x}_i)\frac{1}{N}$ be $K \times N$ matrix of assigned joint probabilities. Matrix Q is relaxed to be an element of a transportation polytope

$$U(r, c) \coloneqq \{Q \in \mathbb{R}_+^{K \times N} \mid Q\mathbb{1} = r, \ Q^\top \mathbb{1} = c\}, \quad r = \mathbb{1}/K, \quad c = \mathbb{1}/N. \quad (2)$$

where $\mathbb{1}$ are vectors of ones, and r and c the marginal projections of matrix Q onto its clusters and data indices, respectively. Finally, optimizing $E(P, Q)$ w.r.t. to $Q \in U(r, c)$ is a linear optimal transport problem, for which [21] provides a fast, matrix-vector multiplication based solution.

3.2 Clustering with arbitrary prior distributions

A shortcoming of the algorithm just described is the assumption that all clusters are equally probable. This avoids converging to degenerate cases but is too constraining in practice since real datasets follow highly skewed distributions [1, 41], and even in datasets that are collected to be uniform, they are not completely so [17, 41, 68]. Furthermore, knowledge of the data distribution, for example long-tailedness, can be used as additional information (e.g. as in [61] for meta-learning) that can improve the clustering by allocating the right number of data points to each cluster. Next, we describe a mechanism to change this distribution arbitrarily.

In the algorithm above, changing the label prior amounts to choosing a different cluster marginal r in the polytope $U(r, c)$. The difficulty is that r is only known up to an arbitrary permutation of the clusters, as we do not know a-priori which clusters are more frequent and which ones less so. To understand how this issue can be addressed, we need to explicitly write out the energy optimised by the Sinkhorn-Knopp (SK) algorithm [21] to solve the optimal transport problem. This energy is:

$$\min_{Q \in U(r,c)} \langle Q, -\log P \rangle + \frac{1}{\lambda} \text{KL}(Q \| rc^\top), \quad (3)$$

where λ is a fixed parameter. Let $r' = Rr$ where R is a permutation matrix matching clusters to marginals. We then seek to optimize the same quantity w.r.t. R, obtaining the optimal permutation as $R^* = \text{argmin}_R E(R)$ where

$$E(R) = \langle Q, -\log P \rangle + \frac{1}{\lambda} \text{KL}(Q \| Rrc^\top) = \text{const} + \sum_y -q(y) \left[R \log r \right]_y. \quad (4)$$

While there is a combinatorial number of permutation matrices, we show that minimizing Eq. (4) can be done by first sorting classes y in order of increasing $q(y)$, so that $y > y' \Rightarrow q(y) > q(y')$, and

4

then finding the permutation that R that also sorts $[R \log r]_y$ in increasing order.[4] We conclude that R cannot be optimal unless it sorts all pairs. After this step, the SK algorithm can be applied using the optimal permutation R^*, without any significant cost (as solving for R is equivalent to sorting $\mathcal{O}(K \log K)$ with $K \ll N$). The advantage is that it allows to choose any marginal distribution, even highly unbalanced ones which are likely to be a better match for real world image and video classes than a uniform distribution.

3.3 Multi-modal single labelling

Next, we tackle our second requirement of extracting as much information as possible from multi-modal data. In principle, all we require to use the clustering formulation Eq. (1) with multi-modal data $\boldsymbol{x} = (a, v)$ is to design a corresponding multi-modal representation $\Psi(\boldsymbol{x}) = \Psi(a, v)$. However, we argue for *multi-modal single labelling* instead. By this, we mean that we wish to cluster data one modality at a time, but in a way that is modality agnostic. Formally, we introduce *modality splicing transformations* [60] $t_a(\boldsymbol{x}) = a$ and $t_v(\boldsymbol{x}) = v$ and use these as *data augmentations*. Recall that augmentations are random transformations t such as rotating an image or distorting an audio track that one believes should leave the label/cluster invariant. We thus require our activations used for clustering to be an average over augmentations by replacing matrix $\log P$ with

$$[\log P]_{yi} = \mathbb{E}_t[\log \mathrm{softmax}_y \Psi(t\boldsymbol{x}_i)]. \tag{6}$$

If we consider splicing as part of the augmentations, we can learn clusters that are invariant to standard augmentations *as well as* the choice of modality. In practice, to account for modality splicing, we define and learn a pair $\Psi = (\Psi_a, \Psi_v)$ of representations, one per modality, resulting in the same clusters ($\Psi_a(t_a(\boldsymbol{x})) \approx \Psi_v(t_v(\boldsymbol{x}))$). This is illustrated in Figure 1.

Initialization and alignment. Since networks Ψ_a and Ψ_v are randomly initialized, at the beginning of training their output layers are *de-synchronized*. This means that there is no reason to believe that $\Psi_a(t_a(\boldsymbol{x})) \approx \Psi_v(t_v(\boldsymbol{x}))$ simply because the *order* of the labels in the two networks is arbitrary. Nevertheless, in many self-supervised learning formulations, one exploits the fact that even randomly initialized networks capture a useful data prior [71], which is useful to bootstrap learning.

In order to enjoy a similar benefit in our formulation, we propose to *synchronise* the two output layers of Ψ_a and Ψ_b before training the model. Formally, we wish to find the permutation matrix R that, applied to the last layer of one of the two encoders maximizes the agreement with the other (still leaving all the weights to their initial random values). For this, let W_a and W_v be the last layer weight matrices of the two networks,[5] such as $\Psi_a(a) = W_a \bar{\Psi}_a(a)$ and $\Psi_v(v) = W_v \bar{\Psi}_v(v)$. We find the optimal permutation R by solving the optimisation problem:

$$\min_R \sum_{i=1}^{N} \left| \mathrm{softmax}(RW_a\bar{\Psi}_a(t_a(\boldsymbol{x}_i))) - \mathrm{softmax}(W_v\bar{\Psi}_v(t_v(\boldsymbol{x}_i))) \right|, \tag{7}$$

In order to compare softmax distributions, we choose $|\cdot|$ as the 1-norm, similar to [33]. We optimize Eq. (7) with a greedy algorithm: starting with a feasible solution and switching random pairs when they reduce the cost function [20], as these are quick to compute and we do not require the global minimum. Further details are given in Appendix A.3. With this permutation, the weight matrix of the last layer of one network can be resorted to match the other.

Decorrelated clustering heads. Conceptually, there is no single 'correct' way of clustering a dataset: for example, we may cluster videos of animals by their species, or whether they are taken indoor or outdoor. In order to alleviate this potential issue, inspired by [6, 37], we simply learn

---

[4]To see why this is optimal, and ignoring ties for simplicity, let R be any permutation and construct a permutation $\bar{R}$ by applying R and then by further swapping two labels $y > y'$. We can relate the energy of R and $\bar{R}$ as:

$$\begin{aligned} E(R) &= E(\bar{R}) + q(y)[\bar{R} \log r]_y + q(y')[\bar{R} \log r]_{y'} - q(y)[\bar{R} \log r]_{y'} - q(y')[\bar{R} \log r]_y \\ &= E(\bar{R}) + (q(y) - q(y')) \left([\bar{R} \log r]_y - [\bar{R} \log r]_{y'}\right). \end{aligned} \tag{5}$$

Since the first factor is positive by assumption, this equation shows that the modified permutation $\bar{R}$ has a lower energy than R if, and only if, $[\bar{R} \log r]_y > [\bar{R} \log r]_{y'}$, which means that $\bar{R}$ sorts the pair in increasing order.

[5]We assume that the linear layer biases are incorporated in the weight matrices.

multiple labelling functions y, using multiple classification heads for the network. We improve this scheme as follows. In each round of clustering, we generate two random augmentations of the data. Then, the applications of SK to half of the heads (at random) see the first version, and the other half the second version, thus increasing the variance of the resulting clusters. This increases the cost of the algorithm by only a small amount — as more time is used for training instead of clustering.

4   Experiments

The experiments are divided into three parts. First, in Section 4.1, we analyze the need for using both modalities when clustering and investigate the effect of our individual technical contributions via ablations and comparison to other approaches. Second, in Section 4.2, we demonstrate how our method achieves its stated goal of labelling a video dataset without human supervision. Third, in Section 4.3, we show that a side effect of our method is to learn effective audio-visual representations that can be used for downstream tasks *e.g.* video action retrieval, establishing a new state of the art.

Datasets.   While the goal and target application of this work is to group unlabelled video datasets, for analysis purposes only, we use datasets that contain human annotated labels. The datasets range from small- to large-scale: The first is the recently released **VGG-Sound** [17], which contains around 200k videos obtained in the wild from YouTube with low labelling noise and covering 309 categories of general classes. The second dataset is **Kinetics-400** [41], which contains around 230k videos covering 400 human action categories. Third, we test our results on **Kinetics-Sound** proposed in [3], formed by filtering the Kinetics dataset to 34 classes that are potentially manifested visually and audibly, leading to 22k videos. Lastly, we use the small-scale **AVE Dataset** [68], originally proposed for audio-visual event localization and containing only around 4k videos. Among these, only VGG-Sound and Kinetics-400 are large enough for learning strong representations from scratch. We therefore train on these datasets and unsupervisedly finetune the VGG-Sound model on Kinetics-Sound and AVE.

Training details.   Our visual encoder is a R(2+1)D-18 [70] network and our audio encoder is a ResNet [32] with 9 layers. For optimization, we use SGD for 200 epochs with weight decay of 10^{-5} and momentum of 0.9, further implementation details are provided in Appendix A.2.

Table 1: **Architectures and pretraining datasets.** We use state-of-the-art representation learning methods and combine pretrained representations with k-means as baselines in the Tables 5a to 5d.

Method	Input shape	Architecture	Pretrain dataset
Supervised	$32 \times 3 \times 112 \times 112$	R(2+1)D-18	Kinetics-400
DPC [30]	$40 \times 3 \times 224 \times 224$	R3D-34	Kinetics-400
MIL-NCE [52]	$32 \times 3 \times 224 \times 224$	S3D	HowTo100M
XDC [2]	$32 \times 3 \times 224 \times 224$	R(2+1)D-18	Kinetics-400

Baselines.   To compare our method on this novel task of clustering these datasets, we obtained various pretrained video representations (DPC [30], MIL-NCE [52] and XDC [2]), both supervised[6] and self-supervised (see Table 1 for details). For comparison, following [79], we run k-means on the global-average-pooled features, setting k to the same number of clusters as our method to ensure a fair comparison. For the k-means algorithm, we use the GPU accelerated version from the Faiss library [40].

Evaluation.   We adopt standard metrics from the self-supervised and unsupervised learning literature: the *normalized mutual information* (NMI), the *adjusted rand index* (ARI) and *accuracy* (Acc) after matching of the self-supervised labels to the ground truth ones (for this we use the Kuhn–Munkres/Hungarian algorithm [45]). We also report the *mean entropy* and the *mean maximal purity* per cluster, defined in Appendix A.4, to analyze the qualities of the clusters. For comparability and interpretability, we evaluate the results using the ground truth number of clusters – which usually is unknown – but we find our results to be stable w.r.t. other number of clusters (see Appendix).

---

[6]The R(2+1)D-18 model from PyTorch trained on Kinetics-400 [41] from `https://github.com/pytorch/vision/blob/master/torchvision/models/video/resnet.py`.

Table 2: **The value of multi-modal understanding** is observed for obtaining a strong set of labels for VGG-Sound. Our method combines both modalities effectively to yield accurracies beyond a single modality and other methods.

Method	🔊	🎥	NMI	ARI	Acc.	$\langle H \rangle$	$\langle p_{max} \rangle$
Random	✗	✓	10.2	4.0	2.2	4.9	3.5
Supervised	✗	✓	46.5	15.6	24.3	2.9	30.8
DPC	✗	✓	15.4	0.7	3.2	4.7	4.9
MIL-NCE	✗	✓	48.5	12.5	22.0	2.6	32.9
XDC	✗	✓	16.7	1.0	3.9	4.5	6.4
	✓	✗	14.0	0.8	2.9	4.6	4.4
	✓	✓	18.1	1.2	4.5	4.41	7.4
SeLaVi	✗	✓	52.8	19.7	30.1	2.6	35.6
	✓	✗	47.5	15.2	26.5	2.8	32.9
	✓	✓	**56.7**	**22.5**	**32.3**	**2.4**	**38.0**

Figure 2: **Effective use of multi-modality** is found for our method when the visual input is compressed and decompressed.

Table 3: **Ablation** of multi-modality, Modality Alignment and Gaussian marginals. Decorrelated Heads. Models are evaluated at 100 epochs on the VGG-Sound dataset.

Method	🖼	(🔊 🎥)	MA?	G.?	DH?	NMI	ft.-Acc
(a) SeLa	✓	✗	–	–	–	20.1	5.9
(b) Concat	✗	✓	–	✗	✗	24.7	7.6
(c) SeLaVi	✗	✓	✗	✗	✗	51.9	27.5
(d) SeLaVi	✗	✓	✗	✓	✓	52.6	27.8
(e) SeLaVi	✗	✓	✓	✗	✓	52.7	28.0
(f) SeLaVi	✗	✓	✓	✓	✗	52.4	27.3
(g) **SeLaVi**	✗	✓	✓	✓	✓	53.2	28.7

Table 4: **Retrieval** via various number of nearest neighbors.

	HMDB			UCF		
Recall @	1	5	20	1	5	20
3D-Puzzle [42]	–	–	–	19.7	28.5	40.0
OPN [46]	–	–	–	19.9	28.7	40.6
ST Order [12]	–	–	–	25.7	36.2	49.2
ClipOrder [78]	7.6	22.9	48.8	14.1	30.3	51.1
SpeedNet [11]	–	–	–	13.0	28.1	49.5
VCP [51]	7.6	24.4	53.6	18.6	33.6	53.5
VSP [19]	10.3	26.6	54.6	24.6	41.9	76.9
SeLaVi	**24.8**	**47.6**	**75.5**	**52.0**	**68.6**	**84.5**

4.1 Technical Analysis

Multi-modality. In order to shed light on the nature of labelling a multi-modal dataset, we provide a detailed study of the use and combination of modalities in Table 2. While visual-only methods such as the Kinetics-400 supervisedly pretrained model, MIL-NCE or DPC cannot work when only audio-stream (🔊) is provided, we show the results for XDC and our method when a single or both modalities are provided. In particular, we find that even when only the visual-stream (🎥) is present at test-time, our method (57% NMI) already outperforms methods solely developed for representation learning, even surpassing the 100M videos with transcripts trained MIL-NCE (49% NMI). When only the audio-stream is used, our method's performance drops only slightly, congruent with the balanced informativeness of both modalities in our method. Finally, when both modalities are used, we find that our method profits from both, such that the combined performance is significantly higher than the maximal performance of each single modality alone.

Degraded modality. In Fig. 2, we analyze how well our method fares when the quality of one modality is reduced. For this, we compress the video-stream by down and upsampling of the video resolution by factors 1, 4, 8 and 16 (details are provided in Appendix A.5). Even though our method has not been trained with compressed videos, we find that its performance degrades more gracefully than the baselines indicating it has learned to rely on both modalities.

Ablation. Next, we ablate the key parameters of our method and show how they each contribute to the overall clustering quality in Table 3. First, we show a baseline model in Table 3(a), when naïvely applying the publically available source-code for SeLa [6] on video frames (🖼), this yields a NMI of 20%. Compared to this frame-only method, in row (b), we find the results for concatenating the features of both modalities (followed by a single clustering head) to only lead to a small improvement upon the frame-only method with a NMI of 25%. Our method is shown in row (c), where we
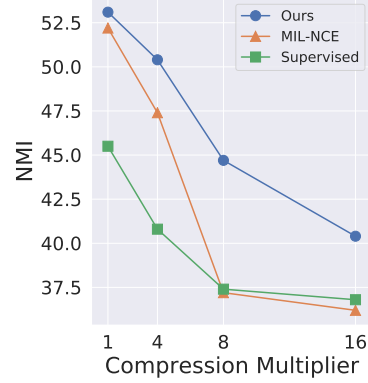
Table 5: **Unsupervised labelling of datasets.** We compare labels from our method to labels that are obtained with k-means on the representations from a supervised and various unsupervised methods on four datasets.

(a) **VGG-Sound.**

Method	NMI	ARI	Acc.	$\langle \mathbf{H} \rangle$	$\langle \mathbf{p}_{max} \rangle$
Random	10.2	4.0	2.2	4.9	3.5
Supervised	46.5	15.6	24.3	2.9	30.8
DPC	15.4	0.7	3.2	4.7	4.9
XDC	18.1	1.2	4.5	4.41	7.4
MIL-NCE	48.5	12.5	22.0	2.6	32.9
SeLaVi	**56.7**	**22.5**	**32.3**	**2.4**	**38.0**

(b) **AVE.**

Method	NMI	ARI	Acc.	$\langle \mathbf{H} \rangle$	$\langle \mathbf{p}_{max} \rangle$
Random	9.2	1.3	9.3	2.9	12.6
Supervised	58.4	34.8	50.5	1.1	60.6
DPC	18.4	5.0	15.1	2.7	17.5
XDC	17.1	6.0	16.4	2.6	19.1
MIL-NCE	56.3	30.3	42.6	**1.2**	57.1
SeLaVi	**64.4**	**43.0**	**54.8**	**1.2**	58.6

(c) **Kinetics.**

Method	NMI	ARI	Acc.	$\langle \mathbf{H} \rangle$	$\langle \mathbf{p}_{max} \rangle$
Random	11.1	0.2	1.8	5.1	3.3
Supervised	70.5	43.4	54.9	1.6	62.2
DPC	16.1	0.6	2.7	4.9	3.9
XDC	17.2	0.8	3.4	4.7	6.2
MIL-NCE	**48.9**	**12.5**	**23.5**	**2.7**	**33.7**
SeLaVi	24.9	2.5	6.6	4.4	8.7

(d) **Kinetics-Sound.**

Method	NMI	ARI	Acc.	$\langle \mathbf{H} \rangle$	$\langle \mathbf{p}_{max} \rangle$
Random	2.8	0.5	5.9	3.3	8.3
Supervised	81.7	66.3	75.0	0.5	85.4
DPC	8.8	2.2	9.6	3.1	13.6
XDC	7.5	1.9	9.4	3.1	13.6
MIL-NCE	47.5	24.0	37.8	**1.5**	**51.0**
SeLaVi	**50.2**	**31.4**	**43.2**	1.7	48.1

find a substantial improvement with a NMI of 52%, i.e. a relative gain of more than 100%. While part of the gain comes from multi-modality, especially compared to row (b), the largest gain comes from the ability of our method in exploiting the natural correspondence provided in the multi-modal data. Finally, by ablating the technical improvements in rows (d)-(f) we find the strongest gain to be coming decorrelated heads, followed by the audio-visual modality alignment (MA) procedure, and that each improvement indeed benefits the model. To analyze the gains obtained by using multiple heads, we have also computed the average NMI between all pairs of heads as $(77.8 \pm 4\%)$. This means that the different heads do learn fairly different clusterings (as NMI takes permutations into account) whilst being at a similar distance to the 'ground-truth' $(53.1 \pm 0.1\%)$.

4.2 Unsupervised labelling audio-visual data

Table 5 shows the quality of the labels obtained automatically by our algorithm. We find that for the datasets VGG-Sound, Kinetics-Sound, and AVE, our method achieves state-of-the-art clustering performance with high accuracies of $32\%, 43\%, 55\%$, even surpassing the one of the strongest video feature encoder at present, the manually-supervised R(2+1)D-18 network. This result echoes the findings in the image domain [72] where plain k-means on representations is found to be less effective compared to learning clusters. For Kinetics-400, we find that the clusters obtained from our method are not well aligned to the human annotated labels. This difference can explained by the fact that Kinetics is strongly focused on visual (human) actions and thus the audio is given almost no weighting in the annotation. We encourage exploring our interactive material, where our method finds clusters grouped by similar background music, wind or screaming crowds. We stress that such a grouping is *ipso facto* not wrong, only not aligned to this set of ground truth labels.

4.3 Labelling helps representation learning

Finally, we show how the visual feature representations unsupervisedly obtained from our method perform on downstream tasks. While not the goal of this paper, we test our representation on a standardized video action retrieval task in Table 4 and also provide results on video action classification in Table A.3, and refer to the Appendix for implementation details. We find that in obtaining strong labels, our method simultaneously learns robust, visual representations that can be used for other tasks without any finetuning and significantly improve the state of the art by over 100% for Recall @1 on UCF-101 and HMDB-51.

5   Conclusion

In this work, we have established strong baselines for the problem of unsupervised labelling of several popular video datasets; introduced a simultaneous clustering and representation learning approach for multi-modal data that outperforms all other methods on these benchmarks; and analysed the importance of multi-modality for this task in detail. We have further found that strong representations are not a sufficient criterion for obtaining good clustering results, yet, the strongest feature representations remain those obtained by supervised, *i.e.* well-clustered training. We thus expect the field of multi-modal clustering to be rapidly adopted by the research community who can build upon the presented method and baselines.

Broader Impact

We propose a method for clustering videos automatically. As such, we see two main areas of potential broader impact on the community and society as a whole.

Few-label harmful content detection.   Our method clusters a video dataset into multiple sets of similar videos, as evidenced by the audio- and visual-stream and produces consistent, homogenous groupings. In practice, unsupervised clustering is especially useful for reducing the amount of data that human annotators have to label, since for highly consistent clusters only a single label needs to be manually obtained which can be propagated to the rest of the videos in the cluster. Using such an approach for the purpose of detecting harmful online content is especially promising. In addition, label-propagation might further lead to a beneficial reduction of type I errors (saying a video is safe when it is not). Furthermore, the multi-modality of our method allows it to potentially detect harmful content that is only manifested in one modality such as static background videos of harmful audio. Multi-modal harmful content detection has also been a subject of a recent data challenge that emphasizes insufficiency of using a single modality[7]. Lastly, the generality of our method allows it to also scale beyond these two modalities and in the future also include textual transcripts. Given the importance of this topic, it is also important to acknowledge, while less of a direct consequence, potential biases that can be carried by the dataset. Indeed models trained using our method will inherit the biases present in the dataset, which could be known but also unknown, potentially leading to propagation of biases without a clear way to analyze them, such as via labels. However, given the numerous pitfalls and failures when deploying computer vision systems to the real world, we believe that the positive impact of foundational research on public datasets, such as is presented in this paper, far outweighs these risks lying further downstream.

Overestimating clustering quality.   The main benefit of our approach is to reduce the cost of grouping large collections of video data in a 'meaningful' way. It is difficult to think of an application where such a capability would lead directly to misuse. In part, this is due to the fact that better clustering results can generally be obtained by using some manual labels, so even where clustering videos could be misused, this probably would not be the method of choice. Perhaps the most direct risk is that a user of the algorithm might overestimate its capabilities. Clustering images is sometimes done in critical applications (e.g. medical science [36, 38, 50]). Our method clusters data based on basic statistical properties and the inductive prior of convolutional neural networks, without being able to tap into the deep understanding that a human expert would have of such domain expertise. Hence, the clusters determined by our method may not necessarily match the clusters an expert would make in a particular domain. Further, as the method is unsupervised, it may learn to exploit biases present in the data that might not be desired by the users. While we believe it has potential to be broadly applied after being finetuned to a specific domain, at present, our method is a better fit for applications such as indexing personal video collections where clustering 'errors' can be tolerated.

---

[7]Hateful memes challenge: `https://www.drivendata.org/competitions/64/hateful-memes/`

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

[2] Humam Alwassel, Dhruv Mahajan, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *arXiv preprint arXiv:1911.12667*, 2019.

[3] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017.

[4] Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.

[5] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *ICLR*, 2020.

[6] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.

[7] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.

[8] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views, 2019.

[9] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Bjorn Ommer. Cliquecnn: Deep unsupervised exemplar learning. In *NeurIPS*, pages 3846–3854, 2016.

[10] Miguel A Bautista, Artsiom Sanakoyeu, and Bjorn Ommer. Deep unsupervised similarity learning using partially ordered sets. In *CVPR*, pages 7130–7139, 2017.

[11] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the speediness in videos, 2020.

[12] Uta Buchler, Biagio Brattoli, and Bjorn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 770–786, 2018.

[13] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[14] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, 2019.

[15] Anna Llagostera Casanovas, Gianluca Monaci, Pierre Vandergheynst, and Rémi Gribonval. Blind audiovisual source separation based on sparse redundant representations. *IEEE Transactions on Multimedia*, 12(5):358–371, 2010.

[16] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Deep adaptive image clustering. In *ICCV*, pages 5879–5887, 2017.

[17] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. *ICASSP)*, May 2020.

[18] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.

[19] Hyeon Cho, Taehoon Kim, Hyung Jin Chang, and Wonjun Hwang. Self-supervised spatio-temporal representation learning using variable playback speed prediction. *arXiv preprint arXiv:2003.02692*, 2020.

[20] Nicos Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group, 1976.

[21] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, pages 2292–2300, 2013.

[22] Virginia R. de Sa. Learning classification with unlabeled data. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *NeurIPS*, pages 112–119. Morgan-Kaufmann, 1994.

[23] Thomas Feo and Mauricio Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6:109–133, 03 1995. doi: 10.1007/BF01096763.

[24] Paola Festa and Mauricio GC Resende. An annotated bibliography of grasp. *Operations Research Letters*, 8:67–71, 2004.

[25] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *ECCV*, pages 35–53, 2018.

[26] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *ICLR*, 2018.

[27] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words, 2020.

[28] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[29] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *CVPR*, pages 6047–6056, 2018.

[30] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCV*, 2019.

[31] John A Hartigan. Direct clustering of a data matrix. *Journal of the american statistical association*, 67(337):123–129, 1972.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[33] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[34] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *CVPR*, pages 9248–9257, 2019.

[35] Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. Learning discrete representations via information maximizing self-augmented training. In *ICML*, pages 1558–1567, 2017.

[36] Dimitris K Iakovidis, Spiros V Georgakopoulos, Michael Vasilakakis, Anastasios Koulaouzidis, and Vassilis P Plagianakos. Detecting and locating gastrointestinal anomalies using deep learning and iterative cluster unification. *IEEE transactions on medical imaging*, 37(10):2196–2210, 2018.

[37] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.

[38] Yizhang Jiang, Kaifa Zhao, Kaijian Xia, Jing Xue, Leyuan Zhou, Yang Ding, and Pengjiang Qian. A novel distributed multitask fuzzy clustering algorithm for automatic mr brain image segmentation. *Journal of medical systems*, 43(5):118, 2019.

[39] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018.

[40] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.

[41] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.

[42] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.

[43] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.

[44] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *ICCV*, 2011.

[45] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955.

[46] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.

[47] Juho Lee, Yoonho Lee, and Yee Whye Teh. Deep amortized clustering. *arXiv preprint arXiv:1909.13433*, 2019.

[48] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

[49] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination, 2020.

[50] Zhuoling Li, Minghui Dong, Shiping Wen, Xiang Hu, Pan Zhou, and Zhigang Zeng. Clu-cnns: Object detection for medical images. *Neurocomputing*, 350:53–59, 2019.

[51] Dezhao Luo, Chang Liu, Yu Zhou, Dongbao Yang, Can Ma, Qixiang Ye, and Weiping Wang. Video cloze procedure for self-supervised spatio-temporal learning. In *AAAI*, 2020.

[52] 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, 2019.

[53] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.

[54] Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with cross-modal agreement, 2020.

[55] Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *CVPR*, 2020.

[56] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[57] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, 2017.

[58] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multi-sensory features. In *ECCV*, 2018.

[59] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.

[60] Mandela Patrick, Yuki M. Asano, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations, 2020.

[61] AJ Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning, 2020.

[62] Mauricio GC Resende and Celso C Ribeiro. Greedy randomized adaptive search procedures: Advances and extensions. In *Handbook of metaheuristics*, pages 169–220. Springer, 2019.

[63] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *ICASSP*, 2019.

[64] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. *CVPR*, Jun 2018.

[65] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human action classes from videos in the wild. In *CRCV-TR-12-01*, 2012.

[66] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019.

[67] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7464–7473, 2019.

[68] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018.

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

[70] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[71] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *arXiv preprint arXiv:1711.10925*, 2017.

[72] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision (ECCV)*, 2020.

[73] 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 *CVPR*, 2019.

[74] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *CVPR*, 2018.

[75] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, June 2018.

[76] Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.

[77] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016.

[78] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.

[79] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Cluster-Fit: Improving Generalization of Visual Representations. In *CVPR*, 2020.

[80] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *CVPR*, pages 5147–5156, 2016.

[81] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[82] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, pages 1735–1744, 2019.

A Appendix

A.1 Pretrained model details

Here we provide additional information about the pretrained models we have used in this work.

Table A.1: **Details for audio encoder.** Architectural and pretraining details for XDC's audio encoder used for benchmarking.

Method	Input shape	Architecture	Pretrain dataset
XDC	$40 \times 1 \times 100$	Resnet-18	Kinetics-400

A.2 Implementation details

We train our method using the Sinkhorn-Knopp parameter $\lambda = 20$, an inverse quadratic clustering schedule with 100 clustering operations and 10 heads which we adopt from [6]. For evaluation, we report results for head 0 to compare against the ground-truth, as we found no significant difference in performance between heads. For the Gaussian distribution, we take the marginals to be from $\mathcal{N}(1, 0.1) * N/K$. For the clustering-heads, we use two-layer MLP-heads as in [8, 18]. The video inputs are 30 frame long clips sampled consecutively from 30fps videos and are resized such that the shorter side is 128 and during training a random crop of size 112 is extracted, no color-jittering is applied. Random horizontal flipping is applied to the video frames with probability 0.5, and then the channels of the video frames are Z-normalized using mean and standard deviation statistics computed across the dataset. The audio is processed as a $1 \times 257 \times 199$ image, by taking the log-mel bank features with 257 filters and 199 time-frames and for training, random volume jittering between 90% and 110% is applied to raw waveform, similar to [54]. For evaluation, a center-crop is taken instead for the video inputs and audio volume is not jittered. We use a mini-batch size of 16 on each of our 64 GPUs giving an effective batch size of 1024 for distributed training for 200 epochs. The initial learning rate is set to 0.01 which we linearly scale with the number of GPUs, after following a gradual warm-up schedule for the first 10 epochs [28]. For training on Kinetics-Sound and AVE, we initialize our model with a VGG-Sound pretrained backbone due to the small training set sizes ($N = 22$k and $N = 3328$). The clustering heads are re-initialized randomly. This ensures a more fair comparison as XDC, DPC and the supervised model are pretrained on Kinetics-400 with $N = 230$k and MIL-NCE on HowTo100M with $N = 100$M videos. We train on VGG-Sound for 200 epochs, which takes around 2 days.

A.3 Pair-based optimization for AV-Alignment

For aligning the visual and audio encoder, we use a greedy switching algorithm that starts from a feasible initial solution [23, 24, 62]. In particular, we consider 50000 potential pair switches with 5 randomized restarts and take the final permutation that yields the lowest cost.

A.4 Evaluation metrics details

The **normalized mutual information** (NMI) is calculated by the formula

$$\text{NMI} = \frac{\text{MI}(U, V)}{0.5H(U) + 0.5H(V)}, \tag{8}$$

where the Mutual information MI is given by $\text{MI}(U, V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i, j) \log \left(\frac{P(i,j)}{P(i)P'(j)} \right)$, and H is the standard entropy, with $H(U) = -\sum_{i=1}^{|U|} P(i) \log(P(i))$. The NMI ranges from 0 (no mutual information) to 100%, which implies perfect correlation.

The rand index (RI) is given by $\text{RI} = \frac{a+b}{C}$, where a, b are the number of pairs of elements that are in the same/different set in the ground truth labelling and in the same/different set in the predicted

clustering and C is the total number of such pairs. The **adjusted Rand index** (ARI) corrects for random assignments and is given by

$$\text{ARI} = \frac{\text{RI} - E[\text{RI}]}{\max(\text{RI}) - E[\text{RI}]}, \tag{9}$$

where the expected RI of a random label assignment is subtracted in both nominator and denominator. Due to the subtraction, the ARI varies from -1 to 1 with a value close to 0 implying random correlation and a value of 1 implying identical agreement.

The **mean entropy per cluster** is given by

$$\langle H \rangle = \frac{1}{K} \sum_{k \in K} H(p(y|\hat{y}_k = k)), \tag{10}$$

where $\hat{y}$ are unsupervisedly obtained clusters and $p(y|\hat{y}_k = k)$ is the distribution of ground-truth clusters for cluster k. Hence, the optimal number of this metric is 0 and a chance assignment yields $\langle H \rangle = -\log 1/K$.

Further, as we wish to understand the semantic purity compared to the ground truth labels of each cluster, so we additionally report the the **mean maximal purity per cluster**,

$$\langle p_{\max} \rangle = \frac{1}{K} \sum_{k \in K} \max(p(y|\hat{y}_k = k)), \tag{11}$$

which ranges from $\langle p_{\max} \rangle = 1/K$ (chance level) to perfect matching at $\langle p_{\max} \rangle = 100\%$.

A.5 Single modality degradation experiment details

We use the default input-sizes for each model, i.e. 112 for ours and the supervised model, 224 for MIL-NCE. Compression is implemented by nearest-neighbor downsampling and subsequently nearest-neighbor upsamling for speed. For this experiment only, we evaluate the performance on the smaller validation sets.

A.6 Further ablations

In Table A.2, we provide the results for varying the number of clusters K in our algorithm. We find that even when moving from the ground-truth number of classes ($K = 309$), to lower numbers ($K = 256$) or higher estimates ($K = 619, 1024$) our results remain stable with the NMI staying almost constant. While the ARI does drop for larger K, we also observe an increase in the purity of the clusters for a larger number of clusters from $\langle \mathbf{p}_{\max} \rangle = 38.0$ for $K = 309$ to $\langle \mathbf{p}_{\max} \rangle = 42.7$ for $K = 619$, which can be particularly useful when dividing the dataset into clusters and subsequently only obtaining human annotations for few examples per cluster.

Table A.2: **Varying K** in our method degrades performances only slightly, showing that our method is robust to various estimations of the ground-truth number of classes. Results on VGG-Sound.

Method	K	**NMI**	**ARI**	**Acc.**	$\langle \mathbf{H} \rangle$	$\langle \mathbf{p}_{\max} \rangle$
SeLaVi	309	56.7	22.5	32.3	2.4	38.0
SeLaVi	256	56.8	24.3	34.2	2.4	36.9
SeLaVi	619	56.9	16.8	23.0	2.2	42.7
SeLaVi	1024	55.1	16.3	9.6	2.1	42.2

A.7 Retrieval downstream task implementation details

We follow [78] in our evaluation protocol and use split 1 of UCF101 and HMDB-51. We uniformly sample 10 clips per video, and average the max-pooled features after the last residual block for each clip per video. We then utilize the averaged features from the validation set to query the videos in the training set. The cosine distance of representations between the query clip and all clips in the

training set are computed and when the class of a test clip appears in the classes of k nearest training clips, it is considered to be correctly retrieved. R@k refers to the retrieval performance using k nearest neighbors.

A.8   Visual classification downstream task

Table A.3: **Representation learning downstream evaluation.**   Self-supervised and fully-supervised trained methods on UCF101 and HMDB51 benchmarks. We follow the standard protocol and report the average top-1 accuracy over the official splits and show results for finetuning the whole network. Methods with † indicate the additional use of video titles and ASR generated text as supervision. Methods with * use ASR generated text.

Method	Architecture	Pretrain Dataset	Top-1 Acc%	
			UCF	HMDB
Full supervision [2]	R(2+1)D-18	ImageNet	82.8	46.7
Full supervision [2]	R(2+1)D-18	Kinetics-400	**93.1**	**63.6**
Weak supervision, CPD [49]†	3D-Resnet50	Kinetics-400	88.7	57.7
MotionPred [73]	C3D	Kinetics-400	61.2	33.4
RotNet3D [39]	3D-ResNet18	Kinetics-600	62.9	33.7
ST-Puzzle [42]	3D-ResNet18	Kinetics-400	65.8	33.7
ClipOrder [78]	R(2+1)D-18	Kinetics-400	72.4	30.9
DPC [30]	3D-ResNet34	Kinetics-400	75.7	35.7
CBT [66]	S3D	Kinetics-600	79.5	44.6
Multisensory [58]	3D-ResNet18	Kinetics-400	82.1	-
XDC [2]	R(2+1)D-18	Kinetics-400	84.2	47.1
AVTS [43]	MC3-18	Kinetics-400	85.8	56.9
AV Sync+RotNet [76]	AVSlowFast	Kinetics-400	87.0	54.6
GDT [60]	R(2+1)D-18	Kinetics-400	<u>88.7</u>	<u>57.8</u>
SeLaVi	R(2+1)D-18	Kinetics-400	83.1	47.1
SeLaVi	R(2+1)D-18	VGG-Sound	87.7	53.1

In Table A.3 we show the performance of our method on two common visual-only video feature representation benchmarks, UCF-101 [65] and HMDB-51 [44]. Note that, as is the standard in this evaluation, we use our visual encoder as initialization and fine-tune the whole network on the target down-stream task. In particular, we follow the finetuning schedule of the one of the current state-of-the-art methods [60]. We find that we achieve competitive performance when trained on VGG-Sound, even surpassing XDC, despite our method using only a spatial resolution of 112×112 and not 224×224.