How and What to Learn: Taxonomizing Self-Supervised Learning for 3D Action Recognition

Amor Ben Tanfous[1,2]* Aimen Zerroug[1,2]* Drew Linsley[2]
Thomas Serre[1,2]

[1] Artificial and Natural Intelligence Toulouse Institute, Toulouse University, France
[2] Carney Institute for Brain Science, Dpt. of Cognitive Linguistic
Psychological Sciences Brown University, Providence, RI 02912

{amor_ben_tanfous, aimen_zerroug, drew_linsley, thomas_serre}@brown.edu

Abstract

There are two competing standards for self-supervised learning in action recognition from 3D skeletons. Su et al., 2020 [31] used an auto-encoder architecture and an image reconstruction objective function to achieve state-of-the-art performance on the NTU60 C-View benchmark. Rao et al., 2020 [23] used Contrastive learning in the latent space to achieve state-of-the-art performance on the NTU60 C-Sub benchmark. Here, we reconcile these disparate approaches by developing a taxonomy of self-supervised learning for action recognition. We observe that leading approaches generally use one of two types of objective functions: those that seek to reconstruct the input from a latent representation ("Attractive" learning) versus those that also try to maximize the representations distinctiveness ("Contrastive" learning). Independently, leading approaches also differ in how they implement these objective functions: there are those that optimize representations in the decoder output space and those which optimize representations in the network's latent space (encoder output). We find that combining these approaches leads to larger gains in performance and tolerance to transformation than is achievable by any individual method, leading to state-of-the-art performance on three standard action recognition datasets. We include links to our code and data.

1. Introduction

Modern deep neural networks require large amounts of labeled data for learning robust visual representations. In recent years, there has been extensive progress towards developing "self-supervised" learning (SSL) methods as a partial solution to this data dependence. SSL involves posing reconstruction tasks on unlabeled datasets, which when appropriately specified, can cause models to learn visual representations that approach standard supervised learning in multiple visual domains, including image categorization [4, 2]. However, far less progress has been made in action recognition, where it remains an open question how best to use SSL.

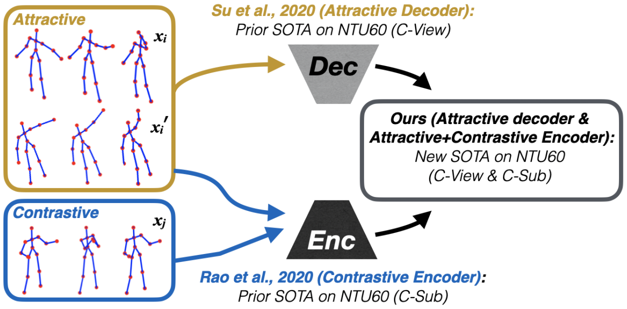
Attractive
x_i
x_i'
Contrastive
x_j
Su et al., 2020 (Attractive Decoder):
Prior SOTA on NTU60 (C-View)
Dec
Ours (Attractive decoder & Attractive+Contrastive Encoder):
New SOTA on NTU60
(C-View & C-Sub)
Enc
Rao et al., 2020 (Contrastive Encoder):
Prior SOTA on NTU60 (C-Sub)

Figure 1. There is little consensus for how to use SSL in action recognition. "Contrastive learning" applied to an encoder leads to top performance on the *C-Sub* evaluation of NTU60 [23]. This approach makes representations of an exemplar x_i more similar to an augmented version x_i' than to different exemplars (e.g., x_j). "Attractive learning" applied to a decoder leads to top performance on the *C-View* evaluation of NTU60 [31]. This approach simply learns to reconstruct exemplar x_i. These two approaches constitute two possible combinations of Contrastive vs. Attractive learning in encoder and decoder spaces out of a much larger number of possible combinations, which have not been studied systematically. We develop a taxonomy of SSL for action recognition and discover a combination that achieves state-of-the-art performance on *both* of these benchmarks.

*These authors contributed equally to this work.

Compared to image categorization, one unique challenge that arises for action recognition is the need to learn robust spatiotemporal representations. A standard approach is to reduce the visual complexity of videos by training and testing models on annotated skeletons of actors. Indeed, skeletonization automatically bypasses several visual challenges associated with raw videos such as figure-ground segmentation. In this domain, self-supervision has led to two notable successes on the NTU60 benchmark. One approach involves optimizing models to solve a raw skeleton reconstruction task from latent space encodings. This approach led to state-of-the-art performance according to the *C-View* evaluation protocol, where models are evaluated on views of actions held out of training [31]. The other approach involves optimizing models to solve a reconstruction task in the latent space, leading to state-of-the-art performance with the *C-Sub* evaluation protocol, which evaluates performance on videos of subjects held out of training [23]. These conflicting approaches and their inconsistent achievements on NTU60 raises the need for a systematic analysis of self-supervision in action recognition (Fig. 1).

Here we develop a novel taxonomy for understanding self-supervision in action recognition (described in Table 1). We decompose leading attempts in self-supervised action recognition into two dimensions. The first dimension describes the type of objective function used: whether it only seeks to cluster similar data samples ("Attractive"), or if it also seeks to dissociate different exemplars ("Contrastive"). The second dimension describes whether representations in the network are being optimized in the input space (Decoder output) or the latent space (Encoder output). Prior work in object recognition has offered empirical evidence that adjusting between Attractive and Contrastive objective functions affects model selectivity and equivariance, which might help performance on downstream tasks [34]. However, it is not known whether this finding extends to action recognition, and how it interacts with where in the network the objective function is posed (i.e., the input or output). Our final taxonomy not only summarizes the state of self-supervision in action recognition, but also reveals a novel approach that outperforms either state-of-the-art approach. Guided by this taxonomy, we contribute the following:

- Models which optimize Encoder and Decoder self-supervised objective functions outperform those which only optimize one or the other.

- We find that Contrastive and Attractive objective functions are better suited to maximizing model tolerance to different classes of transformations. Applications of these objective functions in the Encoder space are more successful at building tolerance for different classes of transformations than applications in the Decoder space. When these approaches are combined together, models achieve better tolerance than if they utilize any single one.

- We achieve state-of-the-art performance on three benchmarks with a model using a combination of "Attractive Decoder" and "Attractive + Contrastive Encoder" objective functions.

2. Self-supervised learning

Self-supervised learning (SSL) describes methods for label-free representation learning models. SSL objective functions involve defining a reconstruction task that forces models to become selective for features that might be generally useful for downstream tasks, for which there is little labeled data. By tracing the development of SSL, we find that these methods can be decomposed along two orthogonal dimensions (Fig. 2). The first dimension describes the normative goal of the SSL objective function. The second dimension describes where in the network SSL objective functions are applied. We begin by describing the first dimension of our taxonomy, and what we see as the principle difference between SSL objective functions: those that seek to pull visually similar representations together (Attractive), versus those that also seek to push visually dissimilar representations apart (Contrastive).

2.1. Objective functions

Attractive SSL The simple insight that the pose of objects tends to vary slowly from frame to frame was the inspiration behind an early form of SSL, called Slow Feature Analysis (SFA) [35]. By training a model to minimize the difference between representations of sequential frames, subject to constraints which avoid trivial solutions, SFA drives models towards learning a spectral decomposition of video sequences. More generally, SFA can be seen as analagous to laplacian eigenmaps [28], thus casting spectral methods as a whole as a specific example of Attractive SSL. A closely related approach is Predictive Coding, which was initially introduced for probablistic models [24], and then later extended to neural networks [18].

The approach of SFA – comparing the current frame to a future frame – has been generalized in SSL. A common approach is now to reconstruct an image from a perturbed version of that image. Such reconstruction tasks include "filling-in" [22], the "Jigsaw" [21], and "colorization" [42], each of which have represented the state of the art for image classification when they were introduced. Like SFA, these methods optimize network representations with an Attractive objective function to maximize the similarity between the true input and an augmented version of that input:

$$\mathcal{L}_{A\mathcal{E}} = -\mathcal{S}(\mathcal{E}(x), \mathcal{E}(x')) \tag{1}$$

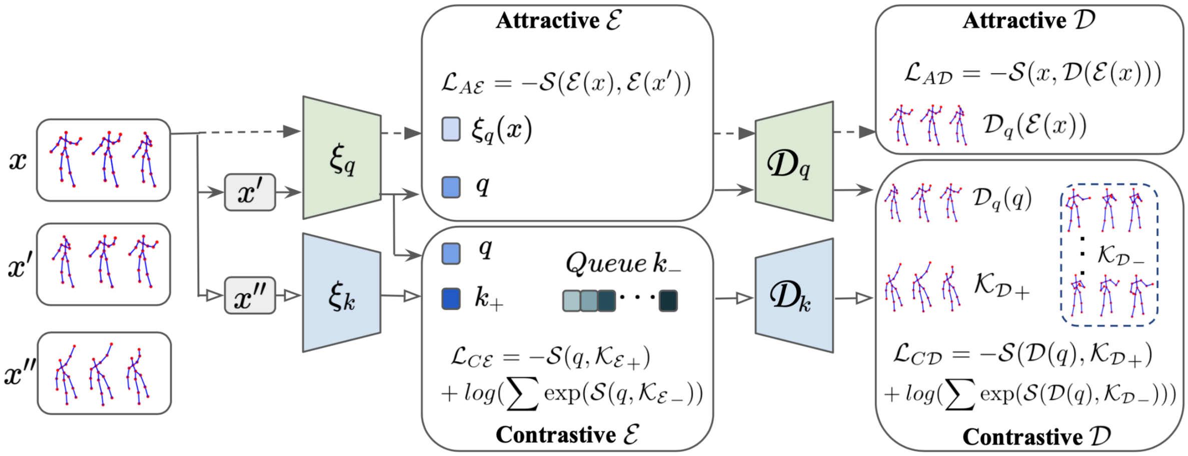
Attractive $\mathcal{E}$
$\mathcal{L}_{A\mathcal{E}} = -\mathcal{S}(\mathcal{E}(x), \mathcal{E}(x'))$
$\xi_q(x)$
q
Attractive $\mathcal{D}$
$\mathcal{L}_{A\mathcal{D}} = -\mathcal{S}(x, \mathcal{D}(\mathcal{E}(x)))$
$\mathcal{D}_q(\mathcal{E}(x))$
x
x'
x''
x'
x''
ξ_q
ξ_k
$\mathcal{D}_q$
$\mathcal{D}_k$
q
k_+
$Queue\ k_-$
$\mathcal{L}_{C\mathcal{E}} = -\mathcal{S}(q, \mathcal{K}_{\mathcal{E}+})$
$+\,log(\sum \exp(\mathcal{S}(q, \mathcal{K}_{\mathcal{E}-}))$
Contrastive $\mathcal{E}$
$\mathcal{D}_q(q)$
$\mathcal{K}_{\mathcal{D}-}$
$\mathcal{K}_{\mathcal{D}+}$
$\mathcal{L}_{C\mathcal{D}} = -\mathcal{S}(\mathcal{D}(q), \mathcal{K}_{\mathcal{D}+})$
$+\,log(\sum \exp(\mathcal{S}(\mathcal{D}(q), \mathcal{K}_{\mathcal{D}-})))$
Contrastive $\mathcal{D}$

Figure 2. Overview of Attractive and Contrastive objective functions in an auto-encoder architecture. Attractive $\mathcal{D}$ seeks to reconstruct an input x from its latent representation, whereas Attractive $\mathcal{E}$ learn to push representations of x and its transformed instance x' close to each other. Contrastive learning aims to attract transformations x' and x'' while also discriminating x' from other exemplars within a dictionary (Queue $\mathcal{K}$). This could be performed in encoders latent space or decoders output space.

Here, x is an input video and x' is the same video with an augmentation applied (e.g., a random spatial symmetry or a rotation). The "encoder" network being optimized is denoted as $\mathcal{E}$, and S is an arbitrary similarity function, although a common choice is cross-entropy. Note that this formulation implies weight sharing between the two instances of $\mathcal{E}(.)$ such that weight updates will force the model to learn an embedding where x is close to x'.

Contrastive SSL In recent years, SSL has begun to approach the performance of supervised learning due to a renewed interest in Contrastive learning. Contrastive learning goes beyond Attractive learning by including an additional "repulsion" term that pushes representations of different instances apart. Contrastive learning has recently achieved performance rivaling supervised learning in image classification [3, 4, 36]. Of particular interest is the "Momentum Contrast" (MoCo) approach of [4], which is one of the current standards in object recognition. Building off of Eq. 1, we rearrange terms in the original formulation of *MoCo* (see SI for details) to define Contrastive learning as:

$$\mathcal{L}_{C\mathcal{E}} = \underbrace{-\mathcal{S}(\mathcal{E}(x), \mathcal{K}_{\mathcal{E}+})}_{attraction} + \underbrace{log(\sum \exp(\mathcal{S}(\mathcal{E}(x), \mathcal{K}_{\mathcal{E}-})))}_{repulsion} \tag{2}$$

The repulsion term in MoCo is introduced by means of a momentum network and a dictionary $\mathcal{K}_{\mathcal{E}}$ containing representations of augmented versions of x as well as other data exemplars, updated over the course of training. This approach forces the encoder $\mathcal{E}(.)$ to maximize the probability that the representation of x is more similar to $\mathcal{K}_{\mathcal{E}+}$ than the remaining entries $\mathcal{K}_{\mathcal{E}-}$ in $\mathcal{K}_{\mathcal{E}}$. An additional free parameter that we omit from our formulation controls the concentration of representations (see SI for the full treatment).

2.2. Encoder and Decoder supervision

Self-supervised Encoding Many of the aforementioned methods, including SFA and Contrastive learning, involve posing an objective function on the latent space, or output of a model. We refer to these as Encoder objective functions. A notable quality of this class of methods is their need for constraints on the objective function to avoid trivial solutions. For Attractive-learning approaches like SFA, these are explicitly added (zero mean, unit variance, and sparse connectivity), whereas the repulsion term of Contrastive learning accomplishes a similar goal.

Self-supervised Decoding Another approach for SSL is to use a decoder network, $\mathcal{D}$, to reconstruct the input from the latent space. The classic example of this is the autoencoder [25, 39], which uses an Attractive objective function similar to the one in Eq. 1 but avoids trivial solutions by comparing the reconstructed latent representation to the input:

$$\mathcal{L}_{A\mathcal{D}} = -\mathcal{S}(x, \mathcal{D}(\mathcal{E}(x))) \tag{3}$$

While autoencoders have been called unsupervised in the past [9], in our taxonomy they are classified as self-supervised models optimizing an attractive objective function. A Contrastive objective function can be also used within a decoder by replacing $\mathcal{E}(x)$ with $\mathcal{D}(\mathcal{E}(x))$ in Eq. 2:

$$\mathcal{L}_{C\mathcal{D}} = -\mathcal{S}(\mathcal{D}(\mathcal{E}(x)), \mathcal{K}_{\mathcal{D}+})$$

$$+log(\sum \exp(\mathcal{S}(\mathcal{D}(\mathcal{E}(x)), \mathcal{K}_{\mathcal{D}-}))) \tag{4}$$

Attractive $\mathcal{E}$	**Contrastive $\mathcal{E}$**	**Attractive $\mathcal{D}$**	**Contrastive $\mathcal{D}$**	**Acc UCLA**
✓				59.98
✓			✓	76.81
			✓	77.73
	✓		✓	79.56
✓		✓		80.15
	✓			83.73
		✓		83.45
	✓	✓		85.70
✓	✓	✓		**86.08**

Table 1. A taxonomy of self-supervised learning objective functions for action recognition. Each row denotes the performance of a different model trained with a particular combination of Attractive/Contrastive objective functions, on the model's encoder ($\mathcal{E}$) or decoder ($\mathcal{D}$). The presence of an objective function is denoted by ✓. Evaluations performed on the UCLA dataset.

where $\mathcal{K}_{\mathcal{D}}$ is a dictionary of reconstructed augmentations of training samples.

2.3. Self-Supervised learning in Action Recognition

To date, most work in action recognition has focused on fully supervised methods using large annotated action datasets [40]. Because it is expensive and time consuming to generate annotations for video, there has been extensive work developing SSL for action recognition. Initial efforts focused on action recognition in RGB videos [1, 12, 19, 29, 30], but there is now a growing body of work focusing on action recognition with 3D skeleton data [20, 31, 43, 14, 7]. There are two distinct approaches that have been explored in action classification: methods that rely on reconstruction in the input space versus reconstruction in the output space.

Reconstruction in the input space, also referred to as an "auto-encoder" approach, has been shown by multiple groups to be effective for SSL in action recognition. Examples of reconstruction for SSL include a GAN-like training routine for SSL [43], and an approach for matching latent representations of augmented and non-augmented videos [20]. Another notable of this approach is by Su et al. [31], who showed that by forcing models to reconstructing input sequences leads to state-of-the-art performance on the C-View evaluation of NTU60. Reconstruction in the latent space has also been effective, especially with "Contrastive" objective functions. Rao et al. [23] achieved state-of-the-art performance on the C-Sub evaluation of NTU60 by borrowing from the MoCo approach for image classification [4]. Others have found success with Contrastive learning for predicting future frames [14].

3. Taxonomizing self-supervised learning

The conflicting success of Attractive decoding versus Contrastive encoding for action recognition raises the question: What is the optimal approach for SSL in action recognition? We address this by turning to the North-Western UCLA (NW-UCLA) action recognition dataset, and building a taxonomy of SSL methods.

Dataset The UCLA [33] dataset consists of 1,494 videos of 10 actions. Each action was performed by 10 actors and repeated between one to six times. Videos were skeletonized using Kinect V1. There are three views of each action and 20 joints in 3D for each subject. Following the standard in the field, the first two views (V1, V2) are used for training, and the last view is held-out for testing (V3) [31]. We also utilize the pre-processed data of Su et al. [31] in these experiments.

Reconstruction tasks A central strategy for SSL is to train models to solve reconstruction tasks, where augmented versions of the input are compared to the original input. In Contrastive learning for image classification, it has been shown that performance is highly dependent on the number of distinct augmentations applied to images [3]. Augmentation strategies for this domain have mostly fallen into four main groups [10]: color transformations, geometric transformation, context-based tasks (e.g., temporal augmentations), and cross-modal based tasks. In contrast, when dealing with 3D skeletons, work has mostly focused on spatial affine and temporal transformations of sequences [23]. Guided by the current state-of-the-art [23], here we pose reconstruction tasks using a combination of spatial affine and temporal augmentations (Fig. 3).

Spatial affine transformations: Unlike 2D images, the 3D parameterization of skeleton joints can be used to explicitly model various transformations while preserving action information. For our action classification experiments, we follow [23] and apply the shearing transformation. To analyse representations obtained with different models, we also use out-of-plane rotations along the azimuth.

Temporal transformations: Action recognition models are

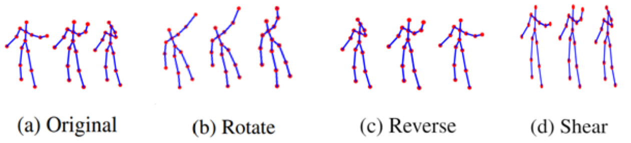

Figure 3. SSL routines for action recognition involve training models on reconstruction tasks. To make these tasks difficult, reconstruction often involves comparing a clean input with a version that has been transformed with spatial affine and temporal augmentations. Spatial affine augmentations include rotations and shearing, whereas temporal augmentations involve reversing the direction of the action sequence.

naturally sensitive to the temporal direction of actions. This means that reversing the direction of an action leads to significant drops in model performance, even though the action category remains the same. For instance, walking forward versus backwards leads to slight differences in pose despite belonging to the same broad action class. To encourage tolerance to such transformations, we followed [23] and introduced an augmentation for randomly reversing the sequence ordering with a 50% probability.

Model A central goal for our taxonomy was to compare different combinations of self-supervised objective functions (within $\mathcal{E}$ and $\mathcal{D}$) in action recognition. To do this we performed our experiments on an architecture capable of implementing any of the aforementioned approaches to SSL. We began with the state-of-the-art network of Su et al. [31], which consists of a 3-layer Encoder $\mathcal{E}$ and a 1-layer Decoder $\mathcal{D}$, both constructed from gated recurrent units (GRU [5]). To improve training speed, convergence, and the interpretability of our taxonomy, we simplified this architecture into a 1-layer Encoder $\mathcal{E}$ and a 1-layer Decoder $\mathcal{D}$. Each layer consisted of 512 units (compared to 1,024 units in [31]). Comparison with architectures with different inductive biases, such as Graph Convolutional Networks (GCNs) are left for future work. Models using Contrastive losses also included an MLP with 128 units, for implementing the MoCo objective. Models were trained with the Adam optimizer [11] for 150 epochs, and early stopping was used to select weights which performed best on a held-out validation set of videos. Additional hyper-parameters scaled the magnitude of the objective functions explored in the taxonomy. After training, a linear classifier was fit to the model's latent space representations on the training set in order to render classification decisions. The implementation will soon be available at `https://github.com/serre-lab/ssl_actionrec`.

Objective functions In our taxonomy we compared Attractive and Contrastive objective functions for SSL. For Attractive SSL, we follow the standards set in action recognition [31] and use different metrics for the similarity function $\mathcal{S}$. Specifically, we used $\mathcal{L}_1$ in the Decoder loss in Eq. 3, whereas in Eq. 1 we used a cosine similarity distance in the projection space (output space of MLP that follows the encoder in our implementation). For Contrastive SSL, we use MoCo [8], which introduces a separate momentum model for computing the Contrastive loss. This momentum model contains the dictionary of examples $\mathcal{K}$, which are compared to encodings (respectively reconstructions) of the current inputs to $\mathcal{E}$ (respectively $\mathcal{D}$). The similarity function $\mathcal{S}$ in MoCo is the dot-product between representations.

Taxonomy We surveyed a set of combinations of Attractive and Contrastive objective functions within Encoders and Decoders. By definition (Eq. 2), Contrastive functions include an Attractive term. Thus, in Table 1, we consider only combinations of either Attractive or Contrastive encoders $\mathcal{E}$ and decoders $\mathcal{D}$. An exception is given by the last row in Table 1 where an Attractive term is added to the encoder objective in case a model combines Contrastive $\mathcal{E}$ and Attractive $\mathcal{D}$. Here, the Attractive $\mathcal{D}$ objective is trained to reconstruct a given input x, while Contrastive $\mathcal{E}$ is trained to pull transformations of x in the latent space. For regularization purpose, we add an Attractive $\mathcal{E}$ objective that is trained to attract the latent of x to that of its first transformation. We refer the reader to Algo. 1 in SI for more details. Also note that we provide additional experiments on the full $\sum_{k=1}^{K} \frac{4!}{k!(4-k)!} = 15$ combinations in SI. Through the taxonomy in Table 1, we make the following observations:

Decoder learning outperforms Encoder learning. On average, models relying on Decoder learning were more accurate ($N = 3, 82.23\%$) than models relying on Encoder learning ($N = 3, 74.82\%$). Indeed, the three best performing models utilize a Decoder objective.

Contrastive learning and Attractive learning lead to similar performance. In stark contrast to the image classification, where Contrastive methods like MoCo dominate, here we find that Attractive and Contrastive objective functions yield nearly identical performance ($N = 3$ for both, 81.92% for Attractive versus 82.23% for Contrastive).

Combining Encoder with Decoder learning improves performance. The two best performing models utilize a combination of $\mathcal{E}$ and $\mathcal{D}$ learning. This finding is emblematic of a trend in our taxonomy, where models that combine $\mathcal{E}$ and $\mathcal{D}$ ($N = 9$, 79.86%) learning outperform those that do not ($N = 6$, 78.52%).

Understanding the taxonomy Including the "Contrastive and Attractive Encoder" objective functions with an "Attractive Decoder" objective function yield the best performance in our taxonomy. While including a "Contrastive

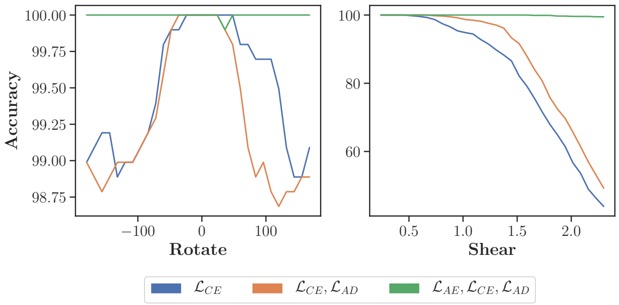

Figure 4. Analysis of the tolerance that different SSL methods build to transformations. We report the nearest neighbor based accuracy, which judges the discriminability of a transformed set of exemplars from their original sample, for the top-3 SSL methods from our taxonomy: "Contrastive $\mathcal{E}$", "Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$", and "Contrastive + Attractive $\mathcal{E}$ + Attractive $\mathcal{D}$". While "Contrastive $\mathcal{E}$" yields better tolerance to rotation, "Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$" achieves better tolerance to shearing. Our best performing model, a combination of the other two approaches depicted here, is far more tolerant to either type of transformation.

Decoder" objective function decreases the model's performance. See supplementary information for additional statistics and an extended discussion. For the remainder of the paper, we investigate the best performing model and compare to the current state-of-the-art approaches [23, 31].

4. Tolerance to transformations

In principle, SSL should build selectivity to the broadest set of features possible in a given dataset, while also maximizing tolerance to transformations of those features. That is, transformations will either be discarded ("invariance") or decodable ("equivariance") from the representations of actions. While tolerance must correlate with performance on downstream tasks, it is unclear the extent to which the models in our taxonomy build tolerance to different transformations.

To investigate this question, we focused on the top-performing models in our taxonomy: *(i)* Contrastive $\mathcal{E}$, *(ii)* Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$, and *(iii)* our top-performing Attractive and Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$. We computed the latent space representations of each model for a set of input samples along with their transformed versions, using fixed transformation parameters (as opposed to randomly sampled parameters, as is normally done in training). Then, for each sample, we calculated the average $\mathcal{L}_2$ distance in latent space to all other samples in the set $||\mathcal{E}(x) - \mathcal{E}(y)||_2$ and to augmented versions of the sample $||\mathcal{E}(x) - \mathcal{E}((x'))||_2$. We report an accuracy metric based on whether an augmented exemplar is the nearest neighbor to the originak sample compared to all other samples of the set. We repeated these measurements for a range of transformation parameter values to derive curves that capture model tolerance to transformations. Using this analysis, we measured model tolerance to shearing and rotation, which are the parametric transformations included in our SSL taxonomy.

We found that different SSL approaches dramatically affect model tolerance to transformations (Fig. 4). The Contrastive $\mathcal{E}$ model was more tolerant to rotations than the Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$, but less tolerant to shearing than the Contrastive $\mathcal{E}$ + Attractive $\mathcal{D}$. Surprisingly, our top-performing model that featured a combination of these approaches was far more tolerant to rotation and shearing than *either* other model. In other words, the combination of these approaches imparted greater tolerance than the sum of the individual parts.

	Method	**CV**	**CS**
Sup	HBRNN-L [6]	64.0	59.1
	2L P-LSTM [15]	70.3	62.9
	ST-LSTM [16]	77.7	69.2
	VA-RNN [41]	87.6	79.4
	3s-CrosSCLR (finetuned) [13]	92.5	86.2
	Colorization [37]	94.9	88.0
	DGCN [38]	**96.0**	**91.5**
SSL	3s-CrosSCLR *(GCN)* [13]	83.4	**77.8**
	Colorization *(PointCloud)* [37]	83.1	75.2
	Thoker et al. *(IMG+SEQ+STG)* [32]	**85.2**	76.3
	LongT GAN [43]	48.1	39.1
	MS^{2}L [14]	–	52.5
	Predict and cluster [31]	76.3	50.7
	AS-CAL [23]	64.8	58.5
	3s-CrosSCLR [13]	69.2	62.8
	Thoker et al. *(SEQ)* [32]	82.5	–
	Ours	76.3	67.0

Table 2. Action classification performance (%) with supervised (first row) and self-supervised (second row) state-of-the-art approaches on the NTU-60 dataset.

5. Experiments

We next turn to standard benchmarks in action recognition to compare the performance of the top-performing model in our taxonomy to the current state of the art. All models are trained according to the methods described in Section 3, and evaluations are dataset dependent, as detailed below.

5.1. Datasets

NTU RGB+D (NTU-60) This dataset consists of 60 different human action classes divided into three major groups: daily actions, mutual actions, and health-related actions [26]. There are $56,880$ action samples in total which are performed by 40 distinct actors. The 3D skeleton data that we focus on consist of the 3D positions of 25 body

	Method	CS	CE
Sup	Part-Aware LSTM * [15]	26.3	25.5
	Soft RNN * [15]	36.3	44.9
	ST-LSTM * [16]	55.7	57.9
	GCA-LSTM * [17]	58.3	59.2
	Two-Stream Attention LSTM * [27]	61.2	63.3
	3s-CrosSCLR (finetuned) [13]	80.5	80.4
	DGCN [38]	**87.3**	88.6
SSL	3s-CrosSCLR *(GCN)* [13]	**67.9**	66.7
	Thoker et al. *(IMG+SEQ+STG)* [32]	67.1	**67.9**
	AS-CAL [23]	48.6	49.2
	3s-CrosSCLR [13]	53.9	53.2
	Ours	59.1	61.5

Table 3. Action classification performance (%) with supervised (first row) and self-supervised (second row) state-of-the-art approaches on the NTU-120 dataset. Results with (*) are reported from [15].

	Method	UCLA
Sup	HBRNN-L [6]	78.5
	VA-RNN [41]	90.7
	AGC-LSTM [27]	**93.3**
SSL	Colorization *(PointCloud)* [37]	**91.1**
	LongT GAN [43]	74.3
	MS^{2}L [14]	76.8
	Predict and cluster [31]	84.9
	Ours	86.08

Table 4. Action classification accuracy (%) with supervised (first row) and self-supervised (second row) state-of-the-art on the UCLA dataset.

joints per skeleton. Each frame consists of two skeletons for mutual actions and one skeleton for The remaining actions. Two standard evaluation protocols are used for this dataset: cross-subject (CS) and cross-view (CV). Under the cross-subject protocol, actions performed by 20 subjects constitute the training set and the rest of actions performed by the other 20 subjects are used for testing. For cross-view evaluation, samples captured by the first two cameras are used for training and the third one is used for testing.

NTU RGB+D (NTU-120) This dataset extends NTU RGB+D 60 with an additional 57,367 skeleton sequences over 60 extra action classes, totalling 113,945 samples over 120 classes captured from 106 distinct subjects and 32 different camera setups [15]. The authors now recommend replacing the Cross-View setting with a Cross-Setup (CE) setting, where 54,468 action sequences collected from half of the camera setups are used for training and the remaining 59,477 samples are used for testing. In the Cross-Subject (CS) setting, 63,026 samples from a selected group of 53 subjects are used for training, and the remaining 50,919 samples for testing.

5.2. Comparison with the state-of-the-art

We report action classification results in Tables 2, 3 and 4 on the NTU-60, NTU-120 and UCLA datasets, respectively. In each case, we compare our results to supervised methods in the first row. In the second row, SSL methods that use RNN based architectures are listed in the second group and methods that use other inductive biases are listed in the first one.

Overall, we observe that our approach achieves state-of-the-art accuracy on most datasets and settings amongst SSL methods based on RNN architectures, with an improvement of more than 10% on NTU-60 (CS), NTU-120 (CS) and NTU-120 (CE). Here, the method in [23] uses Contrastive $\mathcal{E}$ learning. The performance of our approach demonstrates the gain achieved by combining this model with Attractive $\mathcal{D}$ learning. For the NTU-60 (CV) setting, our model is on par with [31] which uses an Attractive $\mathcal{D}$ objective function. Further, the method of [14], which combines Contrastive $\mathcal{E}$ and Attractive $\mathcal{D}$ objective functions performs remarkably less well than our approach with a difference exceeding 8% on the NTU-60 (CS) dataset. Similar observations can be seen on the UCLA dataset where we improve state-of-the-art performance w.r.t Su et al. [31] and outperform the closely related method of [14] by more than 10%.

On the other hand, it is worth mentioning that despite using no explicit supervision and a simple 1-layer GRU encoder/decoder architecture, our method is competitive with deep LSTM-based supervised methods, like [16], and outperforms many other supervised approaches [6, 15] on the NTU-60 and NTU-120 datasets.

5.3. Ablation studies

Our method stands out from prior approaches to SSL by combining Attractive and Contrastive $\mathcal{E}$ objective functions with Attractive $\mathcal{D}$ learning. While our taxonomy demonstrates the relative importance of these components on the UCLA dataset, it is unclear how they contribute to our state-of-the-art results on the complete set of standard benchmarks in action recognition.

To explore this question, we evaluate the effectiveness of each of our method's SSL components (i.e., Attractive $\mathcal{D}$, Contrastive $\mathcal{E}$, and combination of the two) on the NTU60 (CS/CV), NTU 120 (CS/CE), and UCLA datasets and report the obtained results in Table 5. Contrastive $\mathcal{E}$ learning performs better than Attractive $\mathcal{D}$ learning on all benchmarks, and combining these two improves performance beyond either individual approach. Adding an Attractive $\mathcal{E}$ objective improved performance even further.

Model	**NTU60 (CS)**	**NTU60 (CV)**	**UCLA**	**NTU120 (CS)**	**NTU120 (CE)**
Attractive $\mathcal{D}$	59.19	66.67	83.45	48.79	51.92
Contrastive $\mathcal{E}$	66.54	71.86	83.73	56.96	59.72
Attractive $\mathcal{D}$ + Contrastive $\mathcal{E}$	66.88	71.90	85.70	57.78	59.94
Contrastive/Attractive $\mathcal{E}$ + Attractive $\mathcal{D}$	**67.07**	**76.30**	**86.08**	**59.1**	**61.5**

Table 5. Ablating components of our top model demonstrates the relative importance of combining Encoder and Decoder learning for achieving state-of-the-art performance in SSL on the UCLA, NTU60 and NTU120 datasets.

5.3.1 Generalization of the proposed taxonomy

To further validate our proposed taxonomy (Table 1), we performed similar experiments on a larger dataset: NTU60 (CS). In addition, we studied the effect of increasing depth and width of the proposed architecture. Recall that for simplicity, our baseline model consists on 1-layer Encoders $\mathcal{E}$ and 1-layer Decoders $\mathcal{D}$, both constructed from GRU with 512 units. Here, we test two additional architectures. The first is a deeper 3-layer ($\mathcal{E}$ and $\mathcal{D}$) version while the second is a wider (1-layer) model with 1024 units in its layers. Obtained results are reported in Table 6, from which we can observe the following: 1) Previous conclusions that we made from Table 1 on UCLA can be generalized to NTU60 (see ”Base” column in Table 6 vs. Table 1). 2) Increasing the number of layers (see ”Deeper” column in Table 6) hurts the performance of all contrastive models (including our best model) while it improves performance of purely attractive ones. 3) Increasing the number of units (”Wider” column) achieves similar or better performance in all cases with the exception of the purely Contrastive $\mathcal{D}$ model.

$\mathcal{L}_{AE}$	$\mathcal{L}_{CE}$	$\mathcal{L}_{AD}$	$\mathcal{L}_{CD}$	**Base**	**Deeper**	**Wider**
✓				34.63	12.35	36.08
✓			✓	37.93	10.07	43.31
			✓	42.84	38.88	48.31
		✓		59.19	**61.90**	62.58
✓		✓		59.31	61.51	62.67
	✓			66.54	55.37	67.21
	✓	✓		66.88	56.61	66.59
	✓		✓	66.93	55.66	66.88
✓	✓	✓		**67.06**	59.99	**68.64**

Table 6. A comparison of our baseline architecture with deeper (3-layers) and wider (1024 units) architectures on NTU60 (CS). Each row denotes the performance of a different model trained with a particular combination of the objective functions: $\mathcal{L}_{AE}$, $\mathcal{L}_{CE}$, $\mathcal{L}_{AD}$ and $\mathcal{L}_{CD}$.

6. Conclusion

A main drawback of neural networks is their dependence on extremely large labeled datasets to learn robust visual representations. One natural approach to this problem is to induce biases on the model architecture that will support more efficient learning of visual datasets. An orthogonal approach is to develop self-supervised learning routines which can train models in the absence of any labels whatsoever. While SSL has led to breakthroughs for image categorization, the returns are less clear for spatiotemporal tasks like action classification.

Through a systematic survey of approaches in the field, we observe that one possible explanation for the lagging performance in action recognition is that the space of methods has not been adequately searched. Our taxonomy of SSL in action recognition confirms as much: there are large gains in performance to be had by *combining* existing approaches to SSL. Indeed, we demonstrate that the combination of methods non-linearly increases model tolerance to transformations beyond individual components. Most importantly, we find that our taxonomy, which was evaluated on the relatively small UCLA dataset, generalizes to the large body of action classification benchmarks. Our approach achieves a new state of the art in self-supervised approaches to action classification on the NTU-60, NTU-120, and UCLA datasets, while also rivaling the leading RNN-based supervised approaches on these benchmarks.

In summary, by taxonomizing SSL in action classification, we resolve the inconsistent performance of existing methods on different benchmarks, and establish a new standard for SSL in action recognition. We expect that future work which pairs our approach with more elaborate architectures – replete with appropriate inductive biases – will close the divide between self-supervised and supervised learning for action recognition for good.

Acknowledgment

This work was funded by ANITI (Artificial and Natural Intelligence Toulouse Institute, ANR-19-PI3A-0004). Additional support was provided by ONR (N00014-19-1-2029), NSF (IIS-1912280 and EAR-1925481), DARPA (D19AC00015), and NIH/NINDS (R21 NS 112743). We acknowledge the Cloud TPU hardware resources that Google made available via the TensorFlow Research Cloud (TFRC) program as well as computing hardware supported by NIH Office of the Director grant S10OD025181.

References

[1] Unaiza Ahsan, Rishi Madhok, and Irfan Essa. Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 179–189, 2019.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. Feb. 2020.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. Feb. 2020.

[4] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. Mar. 2020.

[5] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. Dec. 2014.

[6] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[7] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[8] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[9] Geoffrey E Hinton and Richard S Zemel. Autoencoders, minimum description length, and helmholtz free energy. *Adv. Neural Inf. Process. Syst.*, 6:3–10, 1994.

[10] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. Oct. 2020.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. Dec. 2014.

[12] Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Unsupervised learning of view-invariant action representations. In *Advances in Neural Information Processing Systems*, pages 1254–1264, 2018.

[13] Linguo Li, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, and Wenjun Zhang. 3D human action representation learning via Cross-View consistency pursuit. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4741–4750, 2021.

[14] Lilang Lin, Sijie Song, Wenhan Yang, and Jiaying Liu. MS2L: Multi-Task Self-Supervised learning for skeleton based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, pages 2490–2498, New York, NY, USA, Oct. 2020. Association for Computing Machinery.

[15] Jun Liu, Amir Shahroudy, Mauricio Lisboa Perez, Gang Wang, Ling-Yu Duan, and Alex Kot Chichung. Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[16] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-Temporal LSTM with trust gates for 3D human action recognition. In *Computer Vision – ECCV 2016*, pages 816–833. Springer International Publishing, 2016.

[17] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1647–1656, 2017.

[18] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. May 2016.

[19] Zelun Luo, Boya Peng, De-An Huang, Alexandre Alahi, and Li Fei-Fei. Unsupervised learning of long-term motion dynamics for videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2203–2212, 2017.

[20] Qiang Nie, Ziwei Liu, and Yunhui Liu. Unsupervised 3D human pose representation with viewpoint and pose disentanglement. In *European Conference on Computer Vision (ECCV)*, 2020.

[21] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. Mar. 2016.

[22] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. Apr. 2016.

[23] Haocong Rao, Shihao Xu, Xiping Hu, Jun Cheng, and Bin Hu. Augmented skeleton based contrastive action learning with momentum LSTM for unsupervised action recognition. Aug. 2020.

[24] Rajesh P N Rao and Dana H Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat. Neurosci.*, 2(1):79–87, Jan. 1999.

[25] D E Rumelhart and J L McClelland. Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, pages 318–362. MIT Press, 1987.

[26] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[27] Chenyang Si, Wentao Chen, Wei Wang, Liang Wang, and Tieniu Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1227–1236, 2019.

[28] Henning Sprekeler. On the relation of slow feature analysis and laplacian eigenmaps. *Neural Comput.*, 23(12):3287–3302, Dec. 2011.

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

[30] Bing Su, Jiahuan Zhou, Xiaoqing Ding, and Ying Wu. Unsupervised hierarchical dynamic parsing and encoding for action recognition. *IEEE Trans. Image Process.*, 26(12):5784–5799, 2017.

[31] Kun Su, Xiulong Liu, and Eli Shlizerman. Predict & cluster: Unsupervised skeleton based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9631–9640, 2020.

[32] Fida Mohammad Thoker, Hazel Doughty, and Cees G M Snoek. Skeleton-Contrastive 3D action representation learning. Aug. 2021.

[33] Jiang Wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2649–2656, 2014.

[34] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. May 2020.

[35] Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: unsupervised learning of invariances. *Neural Comput.*, 14(4):715–770, Apr. 2002.

[36] Z Wu, Y Xiong, S X Yu, and D Lin. Unsupervised feature learning via non-parametric instance discrimination. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, June 2018.

[37] Siyuan Yang, Jun Liu, Shijian Lu, Meng Hwa Er, and Alex C Kot. Skeleton cloud colorization for unsupervised 3D action representation learning. Aug. 2021.

[38] Fanfan Ye, Shiliang Pu, Qiaoyong Zhong, Chao Li, Di Xie, and Huiming Tang. Dynamic GCN: Context-enriched topology learning for skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, pages 55–63, New York, NY, USA, Oct. 2020. Association for Computing Machinery.

[39] Yann Lecun Yoshua Bengio. Scaling learning algorithms toward AI. In *Large-Scale Kernel Machines*, pages 321–359. MIT Press, 2007.

[40] Hong-Bo Zhang, Yi-Xiang Zhang, Bineng Zhong, Qing Lei, Lijie Yang, Ji-Xiang Du, and Duan-Sheng Chen. A comprehensive survey of vision-based human action recognition methods. *Sensors*, 19(5):1005, 2019.

[41] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive neural networks for high performance Skeleton-Based human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(8):1963–1978, Aug. 2019.

[42] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. Mar. 2016.

[43] Nenggan Zheng, Jun Wen, Risheng Liu, Liangqu Long, Jianhua Dai, and Zhefeng Gong. Unsupervised representation learning with long-term dynamics for skeleton based action recognition. In *Thirty-Second AAAI conference on artificial intelligence*, 2018.