Contrastive Multiview Coding

Yonglong Tian[1], Dilip Krishnan[2], and Phillip Isola[1]

[1] MIT CSAIL
[2] Google Research

Abstract. Humans view the world through many sensory channels, e.g., the long-wavelength light channel, viewed by the left eye, or the high-frequency vibrations channel, heard by the right ear. Each view is noisy and incomplete, but important factors, such as physics, geometry, and semantics, tend to be shared between all views (e.g., a "dog" can be seen, heard, and felt). We investigate the classic hypothesis that a powerful representation is one that models view-invariant factors. We study this hypothesis under the framework of multiview contrastive learning, where we learn a representation that aims to maximize mutual information between different views of the same scene but is otherwise compact. Our approach scales to any number of views, and is view-agnostic. We analyze key properties of the approach that make it work, finding that the contrastive loss outperforms a popular alternative based on cross-view prediction, and that the more views we learn from, the better the resulting representation captures underlying scene semantics. Code is available at: http://github.com/HobbitLong/CMC/.

1 Introduction

A foundational idea in coding theory is to learn compressed representations that nonetheless can be used to reconstruct the raw data. This idea shows up in contemporary representation learning in the form of autoencoders [64] and generative models [39,23], which try to represent a data point or distribution as losslessly as possible. Yet lossless representation might not be what we really want, and indeed it is trivial to achieve – the raw data itself is a lossless representation. What we might instead prefer is to keep the "good" information (signal) and throw away the rest (noise). How can we identify what information is signal and what is noise?

To an autoencoder, or a max likelihood generative model, a bit is a bit. No one bit is better than any other. Our conjecture in this paper is that some bits *are* in fact better than others. Some bits code important properties like semantics, physics, and geometry, while others code attributes that we might consider less important, like incidental lighting conditions or thermal noise in a camera's sensor.

We revisit the classic hypothesis that the good bits are the ones that are shared between multiple *views* of the world, for example between multiple sensory modalities like vision, sound, and touch [69]. Under this perspective "presence of

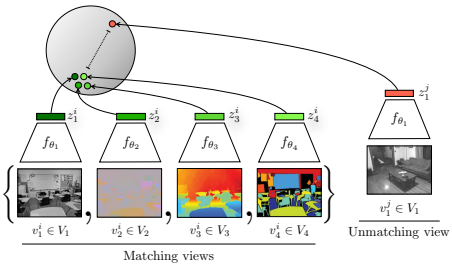

Fig. 1: Given a set of sensory views, a deep representation is learnt by bringing views of the *same* scene together in embedding space, while pushing views of *different* scenes apart. Here we show and example of a 4-view dataset (NYU RGBD [52]) and its learned representation. The encodings for each view may be concatenated to form the full representation of a scene.

dog" is good information, since dogs can be seen, heard, and felt, but "camera pose" is bad information, since a camera's pose has little or no effect on the acoustic and tactile properties of the imaged scene. This hypothesis corresponds to the inductive bias that the way you view a scene should not affect its semantics. There is significant evidence in the cognitive science and neuroscience literature that such view-invariant representations are encoded by the brain (e.g., [69,14,31]). In this paper, we specifically study the setting where the different views are different image channels, such as luminance, chrominance, depth, and optical flow. The fundamental supervisory signal we exploit is the *co-occurrence*, in natural data, of multiple views of the same scene. For example, we consider an image in Lab color space to be a paired example of the co-occurrence of two views of the scene, the L view and the ab view: $\{L, ab\}$.

Our goal is therefore to learn representations that capture information shared between multiple sensory channels but that are otherwise compact (i.e. discard channel-specific nuisance factors). To do so, we employ contrastive learning, where we learn a feature embedding such that views of the same scene map to nearby points (measured with Euclidean distance in representation space) while views of different scenes map to far apart points. In particular, we adapt the recently proposed method of Contrastive Predictive Coding (CPC) [56], except we simplify it – removing the recurrent network – and generalize it – showing how to apply it to arbitrary collections of image channels, rather than just to temporal or spatial predictions. In reference to CPC, we term our method *Contrastive Multiview Coding* (CMC), although we note that our formulation is arguably equally related to Instance Discrimination [79]. The contrastive objective in our formulation, as in CPC and Instance Discrimination, can be understood as attempting to maximize the mutual information between the representations of multiple views of the data.

We intentionally leave "good bits" only loosely defined and treat its definition as an empirical question. Ultimately, the proof is in the pudding: we consider a representation to be good if it makes subsequent problem solving easy, on tasks of human interest. For example, a useful representation of images might be a feature space in which it is easy to learn to recognize objects. We therefore evaluate our method by testing if the learned representations transfer well to standard semantic recognition tasks. On several benchmark tasks, our method achieves

results competitive with the state of the art, compared to other methods for self-supervised representation learning. We additionally find that the quality of the representation improves as a function of the number of views used for training. Finally, we compare the contrastive formulation of multiview learning to the recently popular approach of cross-view prediction, and find that in head-to-head comparisons, the contrastive approach learns stronger representations.

The core ideas that we build on: contrastive learning, mutual information maximization, and deep representation learning, are not new and have been explored in the literature on representation and multiview learning for decades [63,44,80,3]. Our main contribution is to set up a framework to extend these ideas to *any number of views*, and to empirically study the factors that lead to success in this framework. A review of the related literature is given in Section 2; and Fig. 1 gives a pictorial overview of our framework. Our main contributions are:

- We apply contrastive learning to the multiview setting, attempting to maximize mutual information between representations of different views of the same scene (in particular, between different image channels).
- We extend the framework to learn from *more than two* views, and show that the quality of the learned representation improves as number of views increase. Ours is the first work to explicitly show the benefits of multiple views on representation quality.
- We conduct controlled experiments to measure the effect of mutual information estimates on representation quality. Our experiments show that the relationship between mutual information and views is a subtle one.
- Our representations rival state of the art on popular benchmarks.
- We demonstrate that the contrastive objective is superior to cross-view prediction.

2 Related work

Unsupervised representation learning is about learning transformations of the data that make subsequent problem solving easier [7]. This field has a long history, starting with classical methods with well established algorithms, such as principal components analysis (PCA [36]) and independent components analysis (ICA [32]). These methods tend to learn representations that focus on low-level variations in the data, which are not very useful from the perspective of downstream tasks such as object recognition.

Representations better suited to such tasks have been learnt using deep neural networks, starting with seminal techniques such as Boltzmann machines [70,64], autoencoders [29], variational autoencoders [39], generative adversarial networks [23] and autoregressive models [55]. Numerous other works exist, for a review see [7]. A powerful family of models for unsupervised representations are collected under the umbrella of "self-supervised" learning [63,34,85,84,78,59,83]. In these models, an input X to the model is transformed into an output $\hat{X}$, which is supposed to be close to another signal Y (usually in Euclidean space), which

itself is related to X in some meaningful way. Examples of such X/Y pairs are: luminance and chrominance color channels of an image [85], patches from a single image [56], modalities such as vision and sound [57] or the frames of a video [78]. Clearly, such examples are numerous in the world, and provides us with nearly infinite amounts of training data: this is one of the appeals of this paradigm. Time contrastive networks [67] use a triplet loss framework to learn representations from aligned video sequences of the same scene, taken by different video cameras. Closely related to self-supervised learning is the idea of multi-view learning, which is a general term involving many different approaches such as co-training [8], multi-kernel learning [12] and metric learning [6,87]; for comprehensive surveys please see [80,44]. Nearly all existing works have dealt with one or two views such as video or image/sound. However, in many situations, many more views are available to provide training signals for any representation.

The objective functions used to train deep learning based representations in many of the above methods are either reconstruction-based loss functions such as Euclidean losses in different norms e.g. [33], adversarial loss functions [23] that learn the loss in addition to the representation, or contrastive losses e.g. [25,24,35] that take advantage of the co-occurence of multiple views.

Some of the prior works most similar to our own (and inspirational to us) are Contrastive Predictive Coding (CPC) [56], Deep InfoMax [30], and Instance Discrimination [79]. These methods, like ours, learn representations by contrasting between congruent and incongruent representations of a scene. CPC learns from two views – the past and future – and is applicable to sequential data, either in space or in time. Deep Infomax [30] considers the two views to be the input to a neural network and its output. Instance Discrimination learns to match two sub-crops of the same image. CPC and Deep InfoMax have recently been extended in [28] and [4] respectively. These methods all share similar mathematical objectives, but differ in the definition of the views. Our method differs from these works in the following ways: we extend the objective to the case of *more than two* views, and we explore a different set of view definitions, architectures, and application settings. In addition, we contribute a unique empirical investigation of this paradigm of representation learning. The idea of contrastive learning has started to spread over many other tasks in various other domains [73,86,60,76,47,37,74,81].

3 Method

Our goal is to learn representations that capture information shared between multiple sensory views without human supervision. We start by reviewing previous predictive learning (or reconstruction-based learning) methods, and then elaborate on contrastive learning within two views. We show connections to mutual information maximization and extend it to scenarios including more than two views. We consider a collection of M views of the data, denoted as $V_1, \ldots, V_M$. For each view V_i, we denote v_i as a random variable representing samples following $v_i \sim \mathcal{P}(V_i)$.

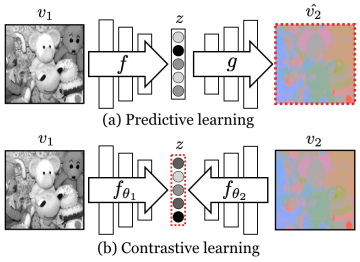

Fig. 2: Predictive Learning vs Contrastive Learning. Cross-view prediction (**Top**) learns latent representations that predict one view from another, with loss measured in the *output* space. Common prediction losses, such as the $\mathcal{L}_1$ and $\mathcal{L}_2$ norms, are *unstructured*, in the sense that they penalize each output dimension independently, perhaps leading to representations that do not capture all the shared information between the views. In contrastive learning (**Bottom**), representations are learnt by contrasting congruent and incongruent views, with loss measured in *representation* space. The red dotted outlines show where the loss function is applied.

3.1 Predictive Learning

Let V_1 and V_2 represent two views of a dataset. For instance, V_1 might be the luminance of a particular image and V_2 the chrominance. We define the *predictive learning* setup as a deep nonlinear transformation from v_1 to v_2 through latent variables z, as shown in Fig. 2. Formally, $z = f(v_1)$ and $\hat{v_2} = g(z)$, where f and g represent the encoder and decoder respectively and $\hat{v_2}$ is the prediction of v_2 given v_1. The parameters of the encoder and decoder models are then trained using an objective function that tries to bring $\hat{v_2}$ "close to" v_2. Simple examples of such an objective include the $\mathcal{L}_1$ or $\mathcal{L}_2$ loss functions. Note that these objectives assume independence between each pixel or element of v_2 given v_1, i.e., $p(v_2|v_1) = \Pi_i p(v_{2i}|v_1)$, thereby reducing their ability to model correlations or complex structure. The predictive approach has been extensively used in representation learning, for example, colorization [84,85] and predicting sound from vision [57].

3.2 Contrastive Learning with Two Views

The idea behind contrastive learning is to learn an embedding that separates (contrasts) samples from two different distributions. Given a dataset of V_1 and V_2 that consists of a collection of samples $\{v_1^i, v_2^i\}_{i=1}^N$, we consider contrasting congruent and incongruent pairs, i.e. samples from the joint distribution $x \sim p(v_1, v_2)$ or $x = \{v_1^i, v_2^i\}$, which we call *positives*, versus samples from the product of marginals, $y \sim p(v_1)p(v_2)$ or $y = \{v_1^i, v_2^j\}$, which we call *negatives*.

We learn a "critic" (a discriminating function) $h_\theta(\cdot)$ which is trained to achieve a high value for positive pairs and low for negative pairs. Similar to recent setups for contrastive learning [56,24,50], we train this function to correctly select a single positive sample x out of a set $S = \{x, y_1, y_2, ..., y_k\}$ that contains k negative samples:

$$\mathcal{L}_{contrast} = -\mathop{\mathbb{E}}_{S}\left[\log \frac{h_\theta(x)}{h_\theta(x) + \sum_{i=1}^{k} h_\theta(y_i)}\right] \tag{1}$$

To construct S, we simply fix one view and enumerate positives and negatives from the other view, allowing us to rewrite the objective as:

$$\mathcal{L}_{contrast}^{V_1,V_2} = - \underset{\{v_1^1, v_2^1, \ldots, v_2^{k+1}\}}{\mathbb{E}} \left[\log \frac{h_\theta(\{v_1^1, v_2^1\})}{\sum_{j=1}^{k+1} h_\theta(\{v_1^1, v_2^j\})} \right] \tag{2}$$

where k is the number of negative samples v_2^j for a given sample v_1^1. In practice, k can be extremely large (e.g., 1.2 million in ImageNet), and so directly minimizing Eq. 2 is infeasible. In Section 3.4, we show two approximations that allow for tractable computation.

Implementing the critic We implement the critic $h_\theta(\cdot)$ as a neural network. To extract compact latent representations of v_1 and v_2, we employ two encoders $f_{\theta_1}(\cdot)$ and $f_{\theta_2}(\cdot)$ with parameters θ_1 and θ_2 respectively. The latent represention are extracted as $z_1 = f_{\theta_1}(v_1)$, $z_2 = f_{\theta_2}(v_2)$. We compute their cosine similarity as score and adjust its dynamic range by a hyper-parameter τ:

$$h_\theta(\{v_1, v_2\}) = \exp\left(\frac{f_{\theta_1}(v_1) \cdot f_{\theta_2}(v_2)}{\|f_{\theta_1}(v_1)\| \cdot \|f_{\theta_2}(v_2)\|} \cdot \frac{1}{\tau}\right) \tag{3}$$

Loss $\mathcal{L}_{contrast}^{V_1,V_2}$ in Eq. 2 treats view V_1 as anchor and enumerates over V_2. Symmetrically, we can get $\mathcal{L}_{contrast}^{V_2,V_1}$ by anchoring at V_2. We add them up as our two-view loss:

$$\mathcal{L}(V_1, V_2) = \mathcal{L}_{contrast}^{V_1,V_2} + \mathcal{L}_{contrast}^{V_2,V_1} \tag{4}$$

After the contrastive learning phase, we use the representation z_1, z_2, or the concatenation of both, $[z_1, z_2]$, depending on our paradigm. This process is visualized in Fig. 1.

Connecting to mutual information The optimal critic h_θ^* is proportional to the density ratio between the joint distribution $p(z_1, z_2)$ and the product of marginals $p(z_1)p(z_2)$ (proof provided in supplementary material):

$$h_\theta^*(\{v_1, v_2\}) \propto \frac{p(z_1, z_2)}{p(z_1)p(z_2)} \propto \frac{p(z_1|z_2)}{p(z_1)} \tag{5}$$

This quantity is the pointwise mutual information, and its expectation, in Eq. 2, yields an estimator related to mutual information. A formal proof is given by [56,61], which we recapitulate in supplement, showing that:

$$I(z_i; z_j) \geq \log(k) - \mathcal{L}_{contrast} \tag{6}$$

where, as above, k is the number of negative pairs in sample set S. Hence minimizing the objective $\mathcal{L}$ maximizes the lower bound on the mutual information $I(z_i; z_j)$, which is bounded above by $I(v_i; v_j)$ by the data processing inequality. The dependency on k also suggests that using more negative samples can lead to an improved representation; we show that this is indeed the case (see supplement). We note that recent work [46] shows that the bound in Eq. 6 can be very weak; and finding better estimators of mutual information is an important open problem.

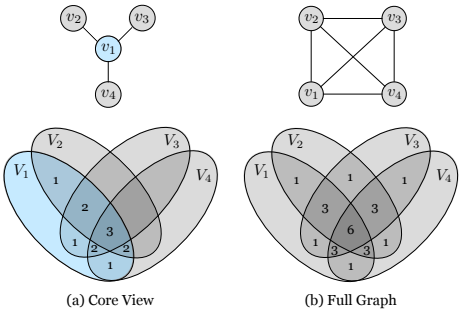

Fig. 3: Graphical models and information diagrams [1] associated with the core view and full graph paradigms, for the case of 4 views, which gives a total of 6 learning objectives. The numbers within the regions show how much "weight" the total loss places on each partition of information (i.e. how many of the 6 objectives that partition contributes to). A region with no number corresponds to 0 weight. For example, in the full graph case, the mutual information between all 4 views is considered in all 6 objectives, and hence is marked with the number 6.

3.3 Contrastive Learning with More than Two Views

We present more general formulations of Eq. 2 that can handle any number of views. We call them the "core view" and "full graph" paradigms, which offer different tradeoffs between efficiency and effectiveness. These formulations are visualized in Fig. 3.

Suppose we have a collection of M views $V_1, \ldots, V_M$. The "core view" formulation sets apart one view that we want to optimize over, say V_1, and builds pair-wise representations between V_1 and each other view $V_j, j > 1$, by optimizing the sum of a set of pair-wise objectives:

$$\mathcal{L}_C = \sum_{j=2}^{M} \mathcal{L}(V_1, V_j) \tag{7}$$

A second, more general formulation is the "full grap" where we consider all pairs $(i, j), i \neq j$, and build $\binom{n}{2}$ relationships in all. By involving all pairs, the objective function that we optimize is:

$$\mathcal{L}_F = \sum_{1 \leq i < j \leq M} \mathcal{L}(V_i, V_j) \tag{8}$$

Both these formulations have the effect that information is prioritized in proportion to the number of views that share that information. This can be seen in the information diagrams visualized in Fig. 3. The number in each partition of the diagram indicates how many of the pairwise objectives, $\mathcal{L}(V_i, V_j)$, that partition contributes to. Under both the core view and full graph objectives, a factor, like "presence of dog", that is common to all views will be preferred over a factor that affects fewer views, such as "depth sensor noise".

The computational cost of the bivariate score function in the full graph formulation is combinatorial in the number of views. However, it is clear from Fig. 3 that this enables the full graph formulation to capture more information

between different views, which may prove useful for downstream tasks. For example, the mutual information between V_2 and V_3 or V_2 and V_4 is completely ignored in the core view paradigm (as shown by a 0 count in the information diagram). Another benefit of the full graph formulation is that it can handle missing information (e.g. missing views) in a natural manner.

3.4 Implementing the Contrastive Loss

Better representations using $\mathcal{L}_{contrast}^{V_1,V_2}$ in Eqn. 2 are learnt by using many negative samples. In the extreme case, we include every data sample in the denominator for a given dataset. However, computing the full softmax loss is prohibitively expensive for large dataset such as ImageNet. One way to approximate this full softmax distribution, as well as alleviate the computational load, is to use Noise-Contrastive Estimation [24,79] (see supplement). Another solution, which we also adopt here, is to randomly sample m negatives and do a simple $(m+1)$-way softmax classification. This strategy is also used in concurrent work [4,28] and dates back to [71].

Memory bank. Following [79], we maintain a memory bank to store latent features for each training sample. Therefore, we can efficiently retrieve m negative samples from the memory buffer to pair with each positive sample without recomputing their features. The memory bank is dynamically updated with features computed on the fly. The benefit of a memory bank is to allow contrasting against more negative pairs, at the cost of slightly stale features.

4 Experiments

We extensively evaluate Contrastive Multiview Coding (CMC) on a number of datasets and tasks. We evaluate on two established image representation learning benchmarks: ImageNet [15] and STL-10 [11] (see supplement). We further validate our framework on video representation learning tasks, where we use image and optical flow modalities, as the two views that are jointly learned. The last set of experiments extends our CMC framework to more than two views and provides empirical evidence of its effectiveness.

4.1 Benchmarking CMC on ImageNet

Following [84], we evaluate task generalization of the learned representation by training 1000-way *linear* classifiers on top of different layers. This is a standard benchmark that has been adopted by many papers in the literature.

Setup. Given a dataset of RGB images, we convert them to the *Lab* image color space, and split each image into L and ab channels, as originally proposed in SplitBrain autoencoders [85]. During contrastive learning, L and ab from the same image are treated as the positive pair, and ab channels from other randomly selected images are treated as a negative pair (for a given L). Each split represents a view of the orginal image and is passed through a separate encoder.

Setting	ResNet-50 x0.5	ResNet-50 x1	ResNet-50 x2
$\{L, ab\}$	57.5 / 80.3	64.0 / 85.5	68.3 / 88.2
$\{Y, DbDr\}$	58.4 / 81.2	64.8 / 86.1	69.0 / 88.9
$\{Y, DbDr\}$ + RA	**60.0 / 82.3**	**66.2 / 87.0**	**70.6 / 89.7**

Table 1: Top-1/5 *Single* crop classification accuracy (%) on ImageNet with a supervised logistic regression classifier. We evaluate CMC using ResNet50 with different width as encoder for *each* of the two views (e.g., L and ab). "RA" stands for RandAugment [13].

As in SplitBrain, we design these two encoders by evenly splitting a given deep network, such as AlexNet [42], into sub-networks across the channel dimension. By concatenating representations layer-wise from these two encoders, we achieve the final representation of an input image. As proposed by previous literature [56,30,3,87,79], the quality of such a representation is evaluated by freezing the weights of encoder and training linear classifier on top of each layer.

Implementation. Unless otherwise specified, we use PyTorch [58] default data augmentation. Following [79], we set the temperature τ as 0.07 and use a momentum 0.5 for memory update. We use 16384 negatives. The supplementary material provides more details on our hyperparameter settings.

CMC with ResNets. We verify the effectiveness of CMC with larger networks such as ResNets [27]. We experiment on learning from luminance and chrominance views in two colorspaces, $\{L, ab\}$ and $\{Y, DbDr\}$ (see Section 4.6 for validation of this choice), and we vary the width of the ResNet encoder for each view. We use the feature after the global pooling layer to train the linear classifier, and the results are shown in Table 1. $\{L, ab\}$ achieves 68.3% top-1 single crop accuracy with ResNet50x2 for each view, and switching to $\{Y, DbDr\}$ further brings about 0.7% improvement. On top of it, strengthening data augmentation with RandAugment [13] yields better or comparable results to other state-of-the-art methods [40,79,87,26,48,18,28,4].

CMC with AlexNet. As many previous unsupervised methods are evaluated with AlexNet [42] on ImageNet [15,41,16,84,53,17,85,54,21,10,83], we also include the the results of CMC using this network in supplementary material.

4.2 CMC on videos

We apply CMC on videos by drawing insight from the two-streams hypothesis [66,22], which posits that human visual cortex consists of two distinct processing streams: the ventral stream, which performs object recognition, and the dorsal stream, which processes motion. In our formulation, given an image i_t that is a frame centered at time t, the ventral stream associates it with a neighbouring frame i_{t+k}, while the dorsal stream connects it to optical flow f_t centered at t. Therefore, we extract i_t, i_{t+k} and f_t from two modalities as three views of a video; for optical flow we use the TV-L1 algorithm [82]. Two separate contrastive learning objectives are built within the ventral stream (i_t, i_{t+k}) and within the dorsal stream (i_t, f_t). For the ventral stream, the negative sample for i_t is chosen

Method	# of Views	UCF-101	HMDB-51
Random	-	48.2	19.5
ImageNet	-	67.7	28.0
VGAN* [77]	2	52.1	-
LT-Motion* [45]	2	53.0	-
TempCoh [51]	1	45.4	15.9
Shuffle and Learn [49]	1	50.2	18.1
Geometry [20]	2	55.1	23.3
OPN [43]	1	56.3	22.1
ST Order [9]	1	58.6	25.0
Cross and Learn [65]	2	58.7	**27.2**
CMC (V)	2	55.3	-
CMC (D)	2	57.1	-
CMC (V+D)	3	**59.1**	26.7

Table 2: Test accuracy (%) on UCF-101 which evaluates *task* transferability and on HMDB-51 which evaluates *task* and *dataset* transferability. Most methods either use single RGB view or additional optical flow view, while VGAN explores sound as the second view. * indicates different network architecture.

as a random frame from another randomly chosen video; for the dorsal stream, the negative sample for i_t is chosen as the flow corresponding to a random frame in another randomly chosen video.

Pre-training. We train CMC on UCF101 [72] and use two CaffeNets [42] for extracting features from images and optical flows, respectively. In our implementation, f_t represents 10 continuous flow frames centered at t. We use batch size of 128 and contrast each positive pair with 127 negative pairs.

Action recognition. We apply the learnt representation to the task of action recognition. The spatial network from [68] is a well-established paradigm for evaluating pre-trained RGB network on action recognition task. We follow the same spirit and evaluate the transferability of our RGB CaffeNet on UCF101 and HMDB51 datasets. We initialize the action recognition CaffeNet up to conv5 using the weights from the pre-trained RGB CaffeNet. The averaged accuracy over three splits is present in Table 2. Unifying both ventral and dorsal streams during pre-training produces higher accuracy for downstream recognition than using only single stream. Increasing the number of views of the data from 2 to 3 (using both streams instead of one) provides a boost for UCF-101.

4.3 Does representation quality improve as number of views increases?

We further extend our CMC learning framework to multiview scenarios. We experiment on the NYU-Depth-V2 [52] dataset which consists of 1449 labeled images. We focus on a deeper understanding of the behavior and effectiveness of CMC. The views we consider are: luminance (L channel), chrominance (ab channel), depth, surface normal [19], and semantic labels.

Setup. To extract features from each view, we use a neural network with 5 convolutional layers, and 2 fully connected layers. As the size of the dataset is relatively small, we adopt the sub-patch based contrastive objective (see supplement) to increase the number of negative pairs. Patches with a size of 128×128 are randomly cropped from the original images for contrastive learning

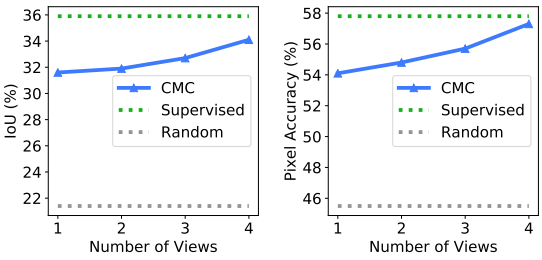

Fig. 4: We show the Intersection over Union (IoU) (left) and Pixel Accuracy (right) for the NYU-Depth-V2 dataset, as CMC is trained with increasingly more views from 1 to 4. As more views are added, both these metrics steadily increase. The views are (in order of inclusion): L, ab, depth and surface normals.

	Pixel Accuracy (%)	mIoU (%)
Random	45.5	21.4
CMC (core-view)	57.1	34.1
CMC (full-graph)	57.0	34.4
Supervised	**57.8**	**35.9**

Table 3: Results on the task of predicting semantic labels from **L channel** representation which is learnt using the patch-based contrastive loss and all 4 views. We compare CMC with *Random* and *Supervised* baselines, which serve as lower and upper bounds respectively. Th core-view paradigm refers to Fig. 3(a), and full-view Fig. 3(b).

(from images of size 480×640). For downstream tasks, we discard the fully connected layers and evaluate using the convolutional layers as a representation.

To measure the quality of the learned representation, we consider the task of predicting semantic labels from the representation of L. We follow the *core view paradigm* and use L as the core view, thus learning a set of representations by contrasting different views with L. A UNet style architecture [62] is utilized to perform the segmentation task. Contrastive training is performed on the above architecture that is equivalent of the UNet's encoder. After contrastive training is completed, we initialize the encoder weights of the UNet from the L encoder (which are equivalent architectures) and keep them frozen. Only the decoder is trained during this finetuning stage.

Since we use the patch-based contrastive loss, in the 1 view setting case, CMC coincides with DIM [30]. The 2-4 view cases contrast L with ab, and then sequentially add depth and surface normals. The semantic labeling results are measured by mean IoU over all classes and pixel accuracy, shown in Fig. 4. We see that the performance steadily improves as new views are added. We have tested different orders of adding the views, and they all follow a similar pattern.

We also compare CMC with two baselines. First, we randomly initialize and freeze the encoder, and we call this the *Random* baseline; it serves as a lower bound on the quality since the representation is just a random projection. Rather than freezing the randomly initialized encoder, we could train it jointly with the decoder. This end-to-end *Supervised* baseline serves as an upper bound. The results are presented in Table 3, which shows our CMC produces high quality feature maps even though it's unaware of the downstream task.

	Metric (%)	L	ab	Depth	Normal
Rand.	mIoU	21.4	15.6	30.1	29.5
	pix. acc.	45.5	37.7	51.1	50.5
CMC	mIoU	34.4	26.1	39.2	37.8
	pix. acc.	57.0	49.6	**59.4**	57.8
Sup.	mIoU	**35.9**	**29.6**	**41.0**	**41.5**
	pix. acc.	**57.8**	**52.6**	59.1	**59.6**

Table 4: Performance on the task of using single view v (L, ab, depth or surface normal) to predict the semantic labels. Our CMC framework improves the quality of unsupervised representations towards that of supervised ones, for all of views investigated. This uses the full-graph paradigm Fig. 3(b).

4.4 Is CMC improving all views?

A desirable unsupervised representation learning algorithm operating on multiple views or modalities should improve the quality of representations for all views. We therefore investigate our CMC framwork beyond L channel. To treat all views fairly, we train these encoders following the *full graph paradigm*, where each view is contrasted with all other views.

We evaluate the representation of each view v by predicting the semantic labels from only the representation of v, where v is L, ab, depth or surface normals. This uses the full-graph paradigm. As in the previous section, we compare CMC with *Random* and *Supervised* baselines. As shown in Table 4, the performance of the representations learned by CMC using full-graph significantly outperforms that of randomly projected representations, and approaches the performance of the fully supervised representations. Furthermore, the full-graph representation provides a good representation learnt for all views, showing the importance of capturing different types of mutual information across views.

4.5 Predictive Learning vs. Contrastive Learning

While experiments in section 4.1 show that contrastive learning outperforms predictive learning [85] in the context of Lab color space, it's unclear whether such an advantage is due to the natural inductive bias of the task itself. To further understand this, we go beyond chrominance (ab), and try to answer this question when geometry or semantic labels are present.

We consider three view pairs on the NYU-Depth dataset: (1) L and depth, (2) L and surface normals, and (3) L and segmentation map. For each of them, we train two identical encoders for L, one using contrastive learning and the other with predictive learning. We then evaluate the representation quality by training a linear classifier on top of these encoders on the STL-10 dataset.

The comparison results are shown in Table 5, which shows that contrastive learning consistently outperforms predictive learning in this scenario where both the task and the dataset are unknown. We also include "random" and "supervised" baselines similar to that in previous sections. Though in the unsupervised stage we only use 1.3K images from a dataset much different from the target dataset STL-10, the object recognition accuracy is close to the supervised method, which uses an end-to-end deep network directly trained on STL-10.

	Accuracy on STL-10 (%)	
Views	Predictive	Contrastive
L, Depth	55.5	**58.3**
L, Normal	58.4	**60.1**
L, Seg. Map	57.7	**59.2**
Random	25.2	
Supervised	65.1	

Table 5: We compare predictive learning with contrastive learning by evaluating the learned encoder on unseen dataset and task. The contrastive learning framework consistently outperforms predictive learning.

Given two views V_1 and V_2 of the data, the predictive learning approach approximately models $p(v_2|v_1)$. Furthermore, losses used typically for predictive learning, such as pixel-wise reconstruction losses usually impose an independence assumption on the modeling: $p(v_2|v_1) \approx \Pi_i p(v_{2i}|v_1)$. On the other hand, the contrastive learning approach by construction does not assume conditional independence across dimensions of v_2. In addition, the use of random jittering and cropping between views allows the contrastive learning approach to benefit from spatial co-occurrence (contrasting in space) in addition to contrasting across views. We conjecture that these are two reasons for the superior performance of contrastive learning approaches over predictive learning.

4.6 How does mutual information affect representation quality?

Given a fixed set of views, CMC aims to maximize the mutual information between representations of these views. We have found that maximizing information in this way indeed results in strong representations, but it would be incorrect to infer that information maximization (infomax) is the key to good representation learning. In fact, this paper argues for precisely the opposite idea: that cross-view representation learning is effective because it results in a kind of information minimization, *discarding* nuisance factors that are not shared between the views.

The resolution to this apparent dilemma is that we want to maximize the "good" information – the *signal* – in our representations, while minimizing the "bad" information – the *noise*. The idea behind CMC is that this can be achieved by doing infomax learning on two views that share signal but have independent noise. This suggests a "Goldilocks principle" [38]: a good collection of views is one that shares some information but not too much. Here we test this hypothesis on two domains: learning representations on images with different colorspaces forming the two views; and learning representations on pairs of patches extracted from an image, separated by varying spatial distance.

In patch experiments we randomly crop two RGB patches of size 64x64 from the same image, and use these patches as the two views. Their relative position is fixed. Namely, the two patches always starts at position (x, y) and $(x + d, y + d)$ with (x, y) being randomly sampled. While varying the distance d, we start from 64 to avoid overlapping. There is a possible bias that with an image of relatively small size (e.g., 512x512), a large d (e.g., 384) will always push these two patches around boundary. To minimize this bias, we use high resolution images (e.g. $2k$) from DIV2K [2] dataset.

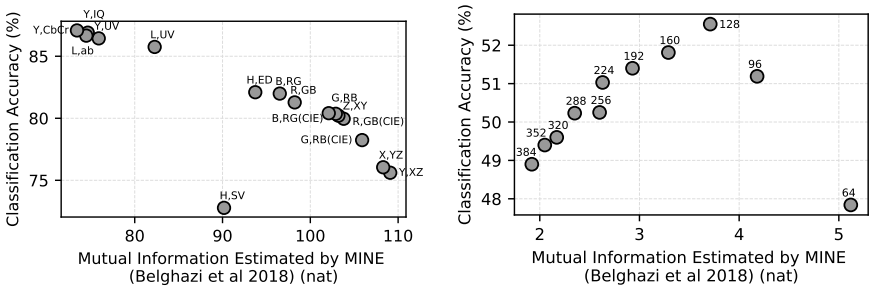

Fig. 5: How does mutual information between views relate to representation quality? (Left) Classification accuracy against estimated MI between channels of different color spaces; (Right) Classification accuracy vs estimated MI between patches at different distances in pixels. MI estimated using MINE [5].

Fig. 5 shows the results of these experiments. The left plot shows the result of learning representations on different colorspaces (splitting each colorspace into two views, such as (L, ab), (R, GB) etc). We then use the MINE estimator [5] to estimate the mutual information between the views. We measure representation quality by training a linear classifier on the learned representations on the STL-10 dataset [11]. The plots clearly show that using colorspaces with minimal mutual information give the best downstream accuracy (For the outlier HSV in this plot, we conjecture the representation quality is harmed by the periodicity of H. Note that the H in HED is not periodic.). On the other hand, the story is more nuanced for representations learned between patches at different offsets from each other (Fig. 5, right). Here we see that views with too little or too much MI perform worse; a sweet spot in the middle exists which gives the best representation. That there exists such a sweet spot should be expected. If two views share *no* information, then, in principle, there is no incentive for CMC to learn anything. If two views share all their information, no nuisances are discarded and we arrive back at something akin to an autoencoder or generative model, that simply tries to represent all the bits in the multiview data.

These experiments demonstrate that the relationship between mutual information and representation quality is meaningful but not direct. Selecting optimal views, which just share relevant signal, has been further discussed in a follow-up work [75] of CMC, and may be a fruitful direction for future research.

5 Conclusion

We have presented a contrastive learning framework which enables the learning of unsupervised representations from multiple views or modalities of a dataset. The principle of maximization of mutual information enables the learning of powerful representations. A number of empirical results show that our framework performs well compared to predictive learning and scales with the number of views.

References

1. Information Diagram - Wikipedia. https://en.wikipedia.org/wiki/Information_diagram
2. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: CVPR (2017)
3. Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., Saunshi, N.: A theoretical analysis of contrastive unsupervised representation learning. In: ICML (2019)
4. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. arXiv preprint arXiv:1906.00910 (2019)
5. Belghazi, M.I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., Hjelm, R.D.: Mine: mutual information neural estimation. arXiv preprint arXiv:1801.04062 (2018)
6. Bellet, A., Habrard, A., Sebban, M.: Similarity learning for provably accurate sparse linear classification. arXiv preprint arXiv:1206.6476 (2012)
7. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. TPAMI (2013)
8. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT. ACM (1998)
9. Buchler, U., Brattoli, B., Ommer, B.: Improving spatiotemporal self-supervision by deep reinforcement learning. In: ECCV (2018)
10. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: ECCV (2018)
11. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: AISTATS (2011)
12. Cortes, C., Mohri, M., Rostamizadeh, A.: Learning non-linear combinations of kernels. In: NIPS (2009)
13. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical data augmentation with no separate search. arXiv preprint arXiv:1909.13719 (2019)
14. Den Ouden, H.E., Kok, P., De Lange, F.P.: How prediction errors shape perception, attention, and motivation. Frontiers in psychology (2012)
15. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
16. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: CVPR (2015)
17. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. In: ICLR (2017)
18. Donahue, J., Simonyan, K.: Large scale adversarial representation learning. In: NIPS (2019)
19. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICCV (2015)
20. Gan, C., Gong, B., Liu, K., Su, H., Guibas, L.J.: Geometry guided convolutional neural networks for self-supervised video representation learning. In: CVPR (2018)
21. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. In: ICLR (2018)
22. Goodale, M.A., Milner, A.D.: Separate visual pathways for perception and action. Trends in neurosciences (1992)
23. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014)

24. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: AISTATS (2010)
25. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: CVPR (2006)
26. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. arXiv preprint arXiv:1911.05722 (2019)
27. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
28. Hénaff, O.J., Razavi, A., Doersch, C., Eslami, S., Oord, A.v.d.: Data-efficient image recognition with contrastive predictive coding. arXiv preprint arXiv:1905.09272 (2019)
29. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. science (2006)
30. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. In: ICLR (2019)
31. Hohwy, J.: The predictive mind. Oxford University Press (2013)
32. Hyvärinen, A., Karhunen, J., Oja, E.: Independent component analysis, vol. 46. John Wiley & Sons (2004)
33. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)
34. Isola, P., Zoran, D., Krishnan, D., Adelson, E.H.: Learning visual groups from co-occurrences in space and time. arXiv preprint arXiv:1511.06811 (2015)
35. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: ICCV (2019)
36. Jolliffe, I.: Principal component analysis. Springer (2011)
37. Kawakami, K., Wang, L., Dyer, C., Blunsom, P., Oord, A.v.d.: Learning robust and multilingual speech representations. arXiv preprint arXiv:2001.11128 (2020)
38. Kidd, C., Piantadosi, S.T., Aslin, R.N.: The goldilocks effect: Human infants allocate attention to visual sequences that are neither too simple nor too complex. PloS one (2012)
39. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
40. Kolesnikov, A., Zhai, X., Beyer, L.: Revisiting self-supervised visual representation learning. In: CVPR (2019)
41. Krähenbühl, P., Doersch, C., Donahue, J., Darrell, T.: Data-dependent initializations of convolutional neural networks. arXiv preprint arXiv:1511.06856 (2015)
42. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS (2012)
43. Lee, H.Y., Huang, J.B., Singh, M., Yang, M.H.: Unsupervised representation learning by sorting sequences. In: ICCV (2017)
44. Li, Y., Yang, M., Zhang, Z.M.: A survey of multi-view representation learning. TKDE (2018)
45. Luo, Z., Peng, B., Huang, D.A., Alahi, A., Fei-Fei, L.: Unsupervised learning of long-term motion dynamics for videos. In: CVPR (2017)
46. McAllester, D., Statos, K.: Formal limitations on the measurement of mutual information. arXiv preprint arXiv:1811.04251 (2018)
47. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: CVPR (2020)

48. Misra, I., van der Maaten, L.: Self-supervised learning of pretext-invariant representations. arXiv preprint arXiv:1912.01991 (2019)
49. Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and learn: unsupervised learning using temporal order verification. In: ECCV (2016)
50. Mnih, A., Kavukcuoglu, K.: Learning word embeddings efficiently with noise-contrastive estimation. In: NIPS (2013)
51. Mobahi, H., Collobert, R., Weston, J.: Deep learning from temporal coherence in video. In: ICML (2009)
52. Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV (2012)
53. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: ECCV. Springer (2016)
54. Noroozi, M., Pirsiavash, H., Favaro, P.: Representation learning by learning to count. In: ICCV (2017)
55. Oord, A.v.d., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. arXiv preprint arXiv:1601.06759 (2016)
56. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
57. Owens, A., Isola, P., McDermott, J., Torralba, A., Adelson, E.H., Freeman, W.T.: Visually indicated sounds. In: CVPR (2016)
58. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: NIPS (2019)
59. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR (2016)
60. Piergiovanni, A., Angelova, A., Ryoo, M.S.: Evolving losses for unlabeled video representation learning. In: CVPR (2020)
61. Poole, B., Ozair, S., Oord, A.v.d., Alemi, A.A., Tucker, G.: On variational bounds of mutual information. In: ICML (2019)
62. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention (2015)
63. Sa, V.: Sensory modality segregation. In: NIPS (2004)
64. Salakhutdinov, R., Hinton, G.: Deep boltzmann machines. In: AISTATS (2009)
65. Sayed, N., Brattoli, B., Ommer, B.: Cross and learn: Cross-modal self-supervision. arXiv preprint arXiv:1811.03879 (2018)
66. Schneider, G.E.: Two visual systems. Science (1969)
67. Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S., Brain, G.: Time-contrastive networks: Self-supervised learning from video. In: ICRA (2018)
68. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NIPS (2014)
69. Smith, L., Gasser, M.: The development of embodied cognition: Six lessons from babies. Artificial life (2005)
70. Smolensky, P.: Information processing in dynamical systems: Foundations of harmony theory. Tech. rep., Colorado Univ at Boulder Dept of Computer Science (1986)
71. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: NIPS (2016)
72. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)

73. Sun, C., Baradel, F., Murphy, K., Schmid, C.: Contrastive bidirectional transformer for temporal representation learning. arXiv preprint arXiv:1906.05743 (2019)
74. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. In: ICLR (2020)
75. Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., Isola, P.: What makes for good views for contrastive learning. arXiv preprint arXiv:2005.10243 (2020)
76. Tschannen, M., Djolonga, J., Ritter, M., Mahendran, A., Houlsby, N., Gelly, S., Lucic, M.: Self-supervised learning of video-induced visual invariances. arXiv preprint arXiv:1912.02783 (2019)
77. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: NIPS (2016)
78. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: ICCV (2015)
79. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018)
80. Xu, C., Tao, D., Xu, C.: A survey on multi-view learning. arXiv preprint arXiv:1304.5634 (2013)
81. Ye, M., Zhang, X., Yuen, P.C., Chang, S.F.: Unsupervised embedding learning via invariant and spreading instance feature. In: CVPR (2019)
82. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime tv-l 1 optical flow. In: Joint pattern recognition symposium (2007)
83. Zhang, L., Qi, G.J., Wang, L., Luo, J.: Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In: CVPR (2019)
84. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV. Springer (2016)
85. Zhang, R., Isola, P., Efros, A.A.: Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In: CVPR (2017)
86. Zhuang, C., Andonian, A., Yamins, D.: Unsupervised learning from video with deep neural embeddings. arXiv preprint arXiv:1905.11954 (2019)
87. Zhuang, C., Zhai, A.L., Yamins, D.: Local aggregation for unsupervised learning of visual embeddings. arXiv preprint arXiv:1903.12355 (2019)