What is self-supervised learning?

Alessandro Giusti

Dalle Molle Institute for Artificial Intelligence
Lugano, Switzerland

Contact: alessandrog@idsia.ch https://idsia-robotics.github.io/

Plan of the lecture

- Part **1**: introduction
- Part **2**: warm-up on the CIFAR-10 dataset
- Part **3**: what is self-supervised learning?
- Part **4**: implement&test a simple self-supervised learning method
- Part **5**: some examples of self-supervised learning in robotics

Plan of the lecture

- Part **1**: introduction
- Part **2**: warm-up on the CIFAR-10 dataset
- **Part 3: what is self-supervised learning?**
- Part **4**: implement&test a simple self-supervised learning method
- Part **5**: some examples of self-supervised learning in robotics

Two main meanings for SSL

- Systems that learn to extract meaningful representations from the data itself
- Systems (typically robots) that collect their own training data but then solve a standard supervised learning task

Two main meanings for SSL

- **Systems that learn to extract meaningful representations from the data itself**
- Systems (typically robots) that collect their own training data but then solve a standard supervised learning task

Self-supervised (aka self-taught) deep learning

The data itself is a source of supervision

Shades of supervision: full supervision

To some extent, any visual task can be solved now by:

1. Construct a large-scale dataset labelled for that task
2. Specify a training loss and neural network architecture
3. Train the network and deploy

Classification error on imagenet

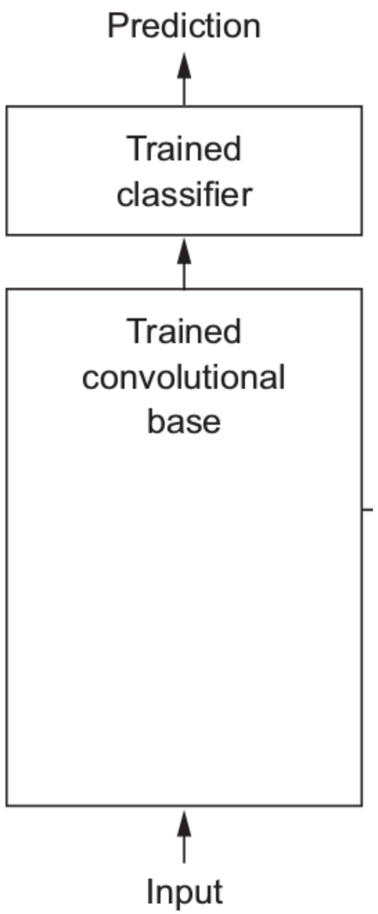
But…

- Labeled data is expensive (eg medical, or whatever problem they are paying you to solve)
- Huge amounts of ***unlabeled*** data
  - Facebook: one billion images uploaded per day
  - 300 hours of video are uploaded to YouTube every minute

→ we want to exploit unlabeled data, at least in part

Using pretrained weights

Step 1



Shades of supervision: self-supervised learning

Can we learn something WITHOUT labels?

How do we (humans) learn?!?

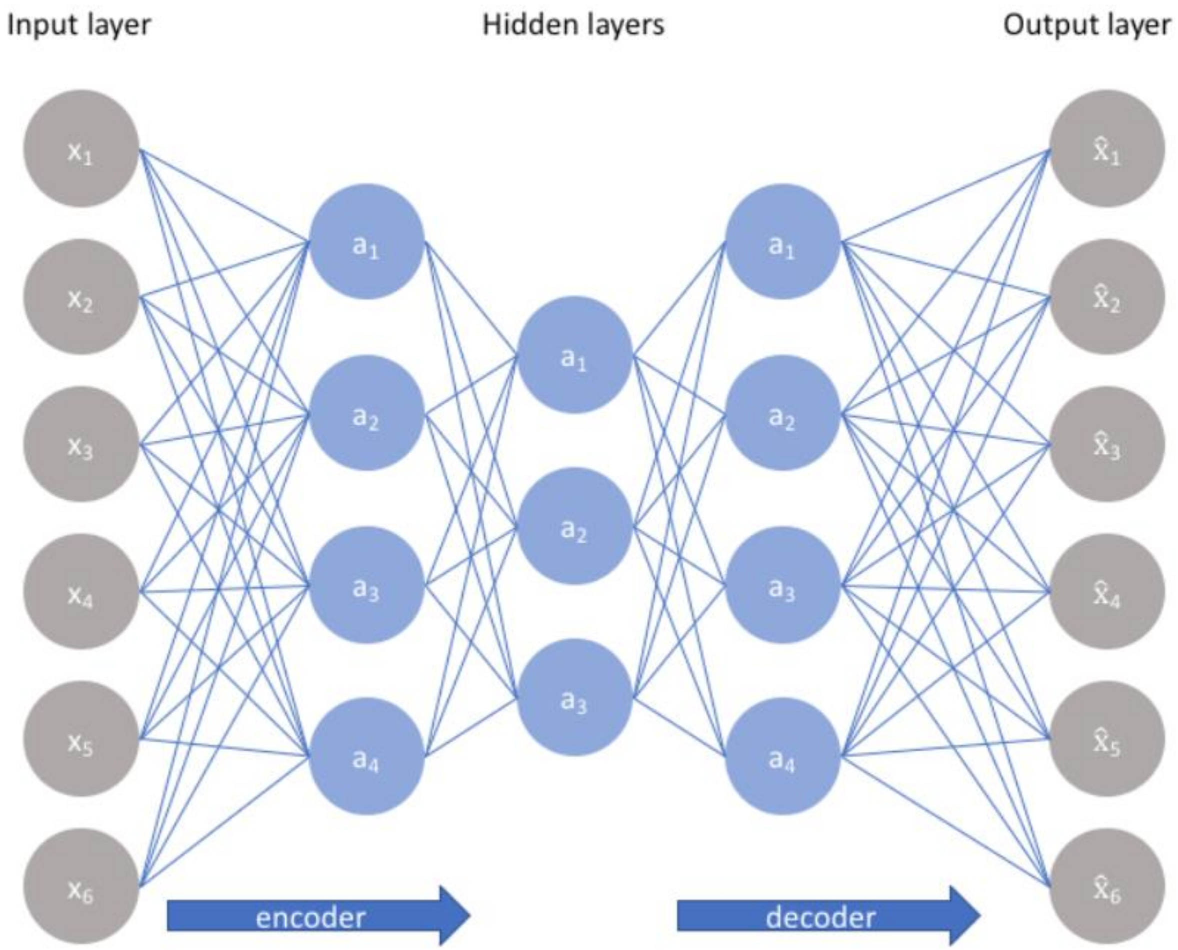
The Scientist in the Crib: What Early Learning Tells Us About the Mind
by Alison Gopnik, Andrew N. Meltzoff and Patricia K. Kuhl
The Development of Embodied Cognition: Six Lessons from Babies
by Linda Smith and Michael Gasser

Definition

- You are interested in solving problem A
- Take a lot of data similar to the one you'll use, without labels
  (of course: you are lazy)
- Invent a problem B (*pretext task*) on the data for which
  - you can get a ground truth for free from the data itself
  - you need to "**understand**" the data in order to solve it
- Train a network for B

→ The network has learned something valuable for A, i.e. to understand the data

You already know at least one method to achieve this: **autoencoders**

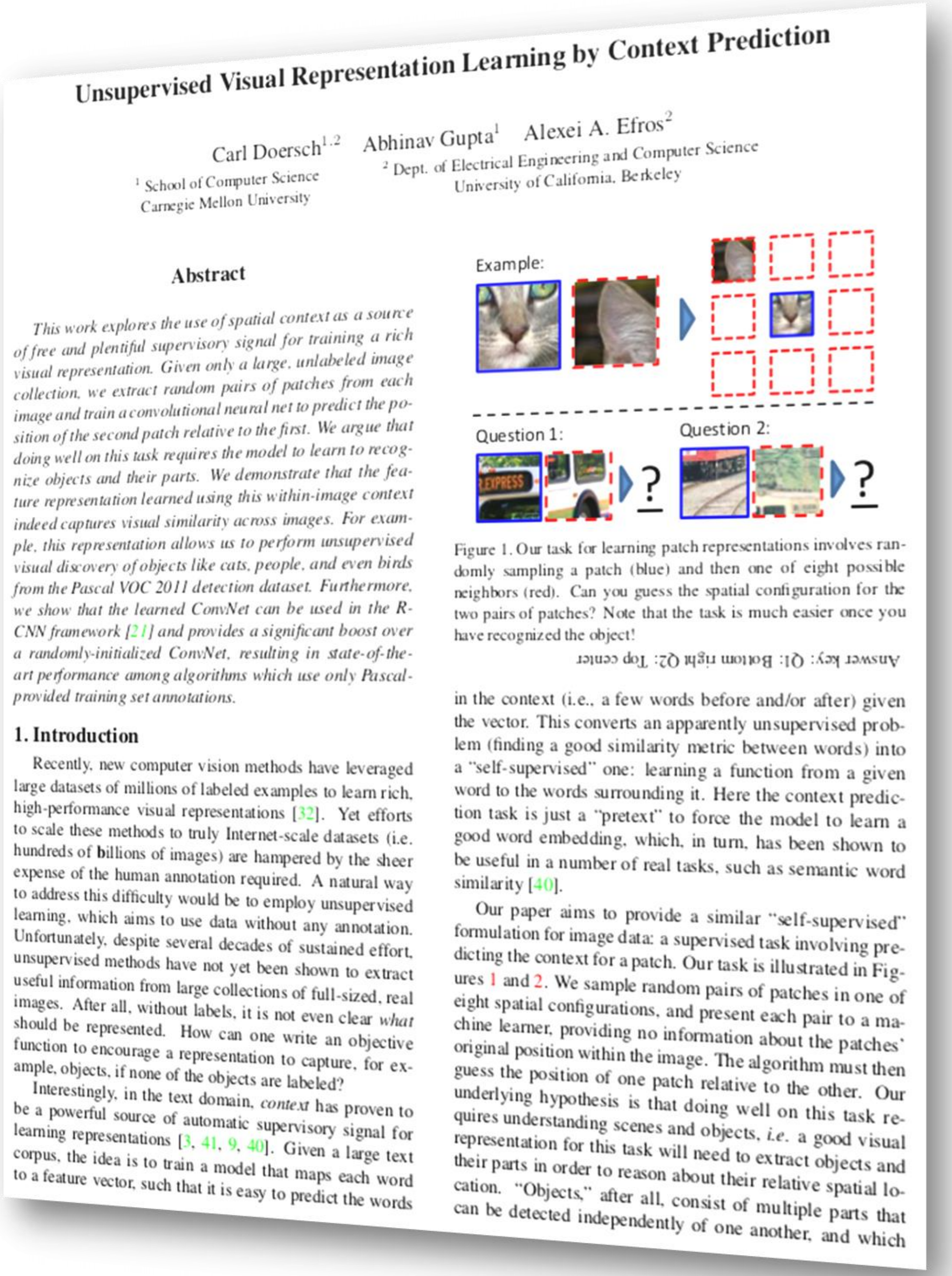


Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to "**understand**" the data in order to solve it

https://www.jeremyjordan.me/autoencoders/

Unsupervised Visual Representation Learning by Context Prediction

https://arxiv.org/abs/1505.05192, 2015



Unsupervised Visual Representation Learning by Context Prediction

Carl Doersch[1,2] Abhinav Gupta[1] Alexei A. Efros[2]
[1] School of Computer Science
Carnegie Mellon University
[2] Dept. of Electrical Engineering and Computer Science
University of California, Berkeley

Abstract

This work explores the use of spatial context as a source of free and plentiful supervisory signal for training a rich visual representation. Given only a large, unlabeled image collection, we extract random pairs of patches from each image and train a convolutional neural net to predict the position of the second patch relative to the first. We argue that doing well on this task requires the model to learn to recognize objects and their parts. We demonstrate that the feature representation learned using this within-image context indeed captures visual similarity across images. For example, this representation allows us to perform unsupervised visual discovery of objects like cats, people, and even birds from the Pascal VOC 2011 detection dataset. Furthermore, we show that the learned ConvNet can be used in the R-CNN framework [21] and provides a significant boost over a randomly-initialized ConvNet, resulting in state-of-the-art performance among algorithms which use only Pascal-provided training set annotations.

Figure 1. Our task for learning patch representations involves randomly sampling a patch (blue) and then one of eight possible neighbors (red). Can you guess the spatial configuration for the two pairs of patches? Note that the task is much easier once you have recognized the object!

Answer key: Q1: Bottom right Q2: Top center

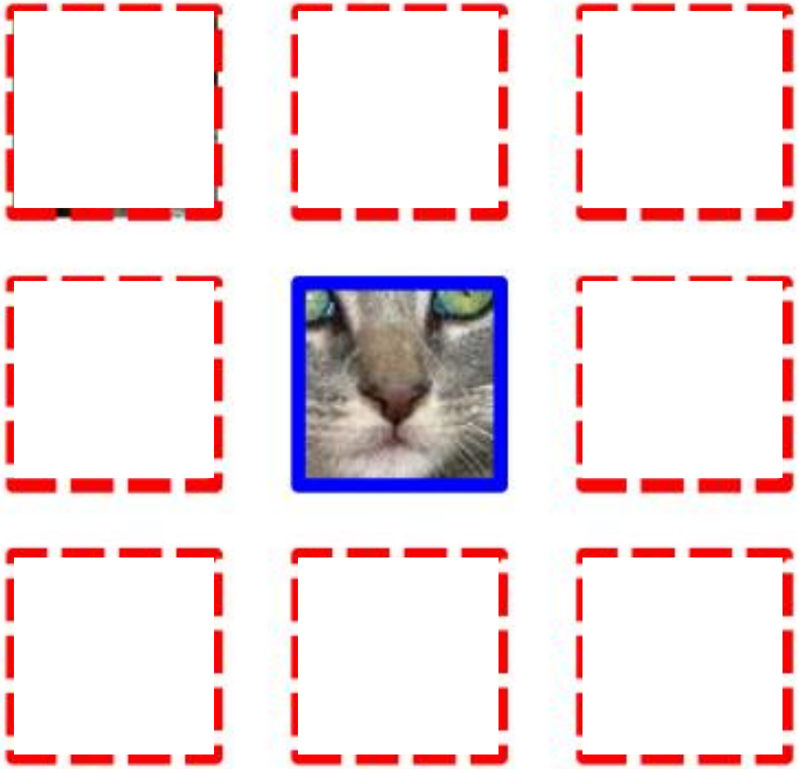
1. Introduction

Recently, new computer vision methods have leveraged large datasets of millions of labeled examples to learn rich, high-performance visual representations [32]. Yet efforts to scale these methods to truly Internet-scale datasets (i.e. hundreds of billions of images) are hampered by the sheer expense of the human annotation required. A natural way to address this difficulty would be to employ unsupervised learning, which aims to use data without any annotation. Unfortunately, despite several decades of sustained effort, unsupervised methods have not yet been shown to extract useful information from large collections of full-sized, real images. After all, without labels, it is not even clear *what* should be represented. How can one write an objective function to encourage a representation to capture, for example, objects, if none of the objects are labeled?

Interestingly, in the text domain, *context* has proven to be a powerful source of automatic supervisory signal for learning representations [3, 41, 9, 40]. Given a large text corpus, the idea is to train a model that maps each word to a feature vector, such that it is easy to predict the words in the context (i.e., a few words before and/or after) given the vector. This converts an apparently unsupervised problem (finding a good similarity metric between words) into a "self-supervised" one: learning a function from a given word to the words surrounding it. Here the context prediction task is just a "pretext" to force the model to learn a good word embedding, which, in turn, has been shown to be useful in a number of real tasks, such as semantic word similarity [40].

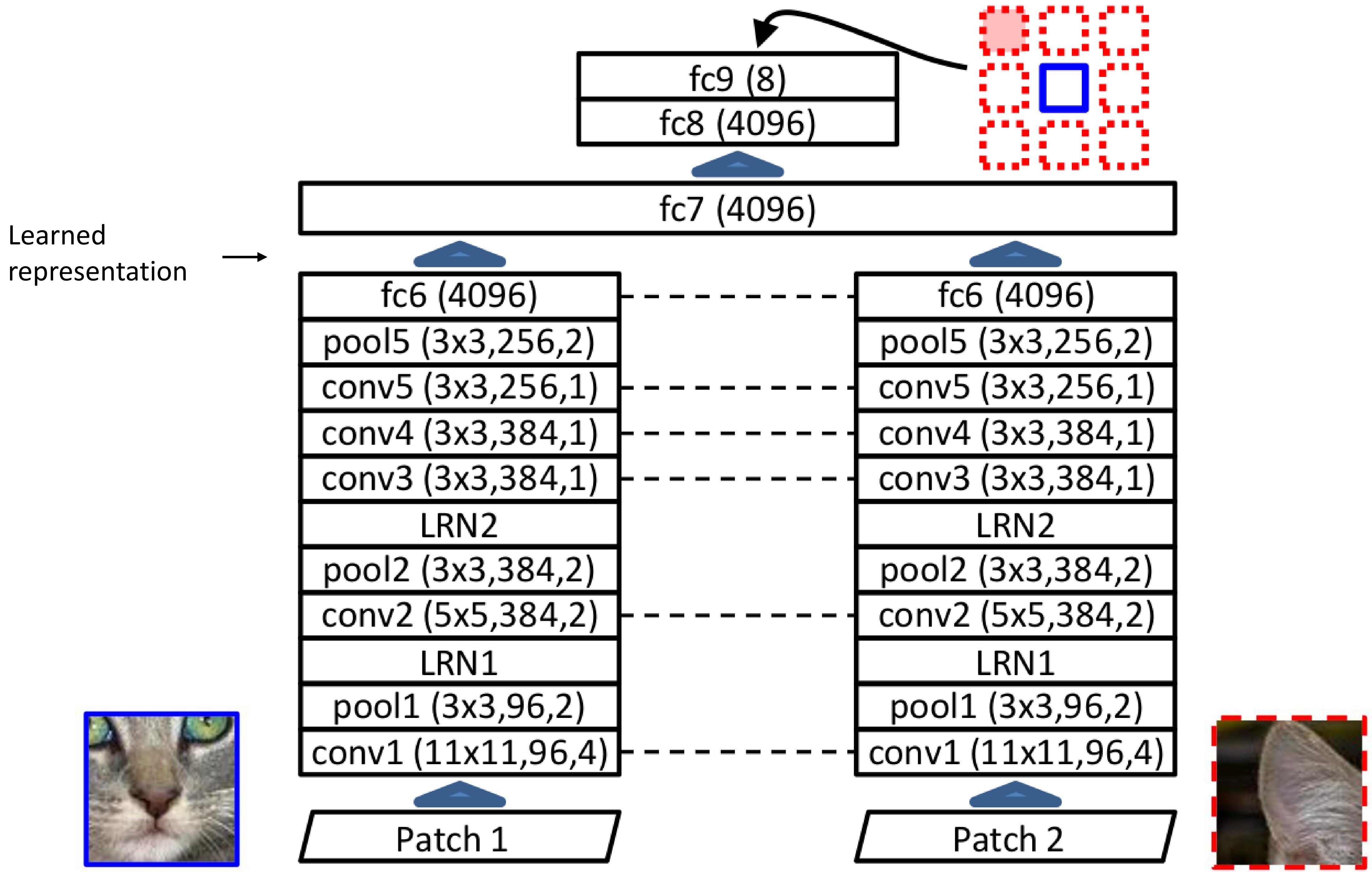
Our paper aims to provide a similar "self-supervised" formulation for image data: a supervised task involving predicting the context for a patch. Our task is illustrated in Figures 1 and 2. We sample random pairs of patches in one of eight spatial configurations, and present each pair to a machine learner, providing no information about the patches' original position within the image. The algorithm must then guess the position of one patch relative to the other. Our underlying hypothesis is that doing well on this task requires understanding scenes and objects, *i.e.* a good visual representation for this task will need to extract objects and their parts in order to reason about their relative spatial location. "Objects," after all, consist of multiple parts that can be detected independently of one another, and which

Think!



Some more...

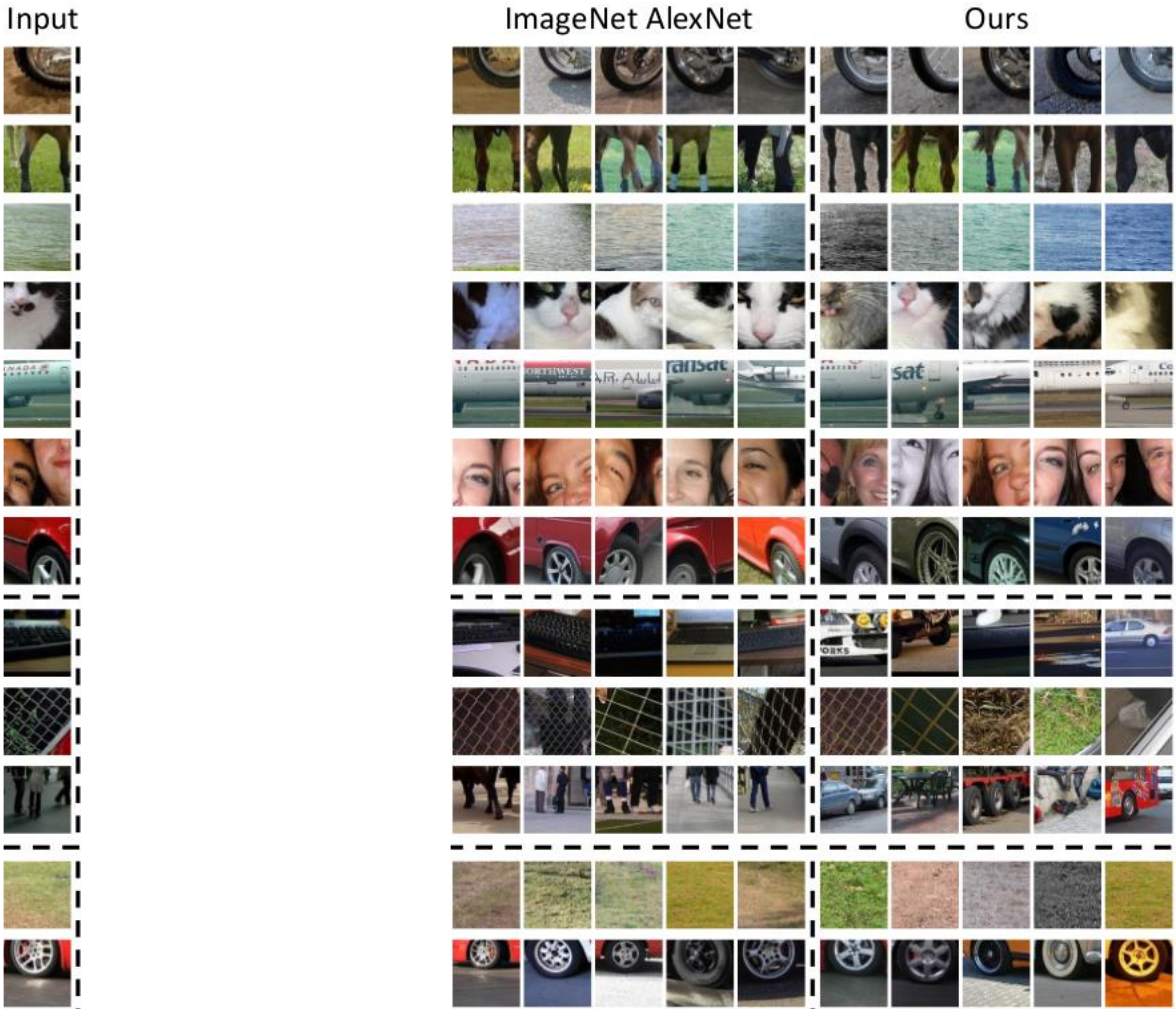
Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to “**understand**” the data in order to solve it



fc9 (8)
fc8 (4096)
fc7 (4096)
Learned representation
fc6 (4096)
pool5 (3x3,256,2)
conv5 (3x3,256,1)
conv4 (3x3,384,1)
conv3 (3x3,384,1)
LRN2
pool2 (3x3,384,2)
conv2 (5x5,384,2)
LRN1
pool1 (3x3,96,2)
conv1 (11x11,96,4)
Patch 1
fc6 (4096)
pool5 (3x3,256,2)
conv5 (3x3,256,1)
conv4 (3x3,384,1)
conv3 (3x3,384,1)
LRN2
pool2 (3x3,384,2)
conv2 (5x5,384,2)
LRN1
pool1 (3x3,96,2)
conv1 (11x11,96,4)
Patch 2

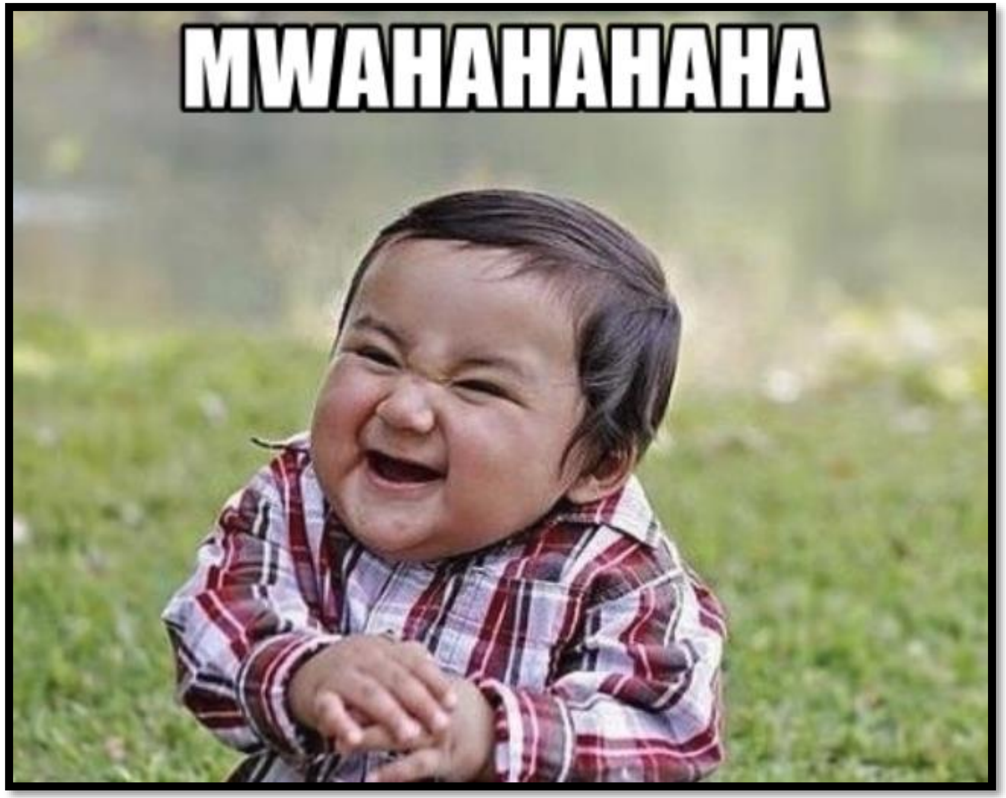
How can we evaluate whether the representation makes sense?

- Given a query patch, we can look for nearest neighbors in the dataset
  - Are these semantically similar?
- It turns out that... Yes, they are
- Surprisingly they also are somewhat similar if the network is randomly initialized (!)



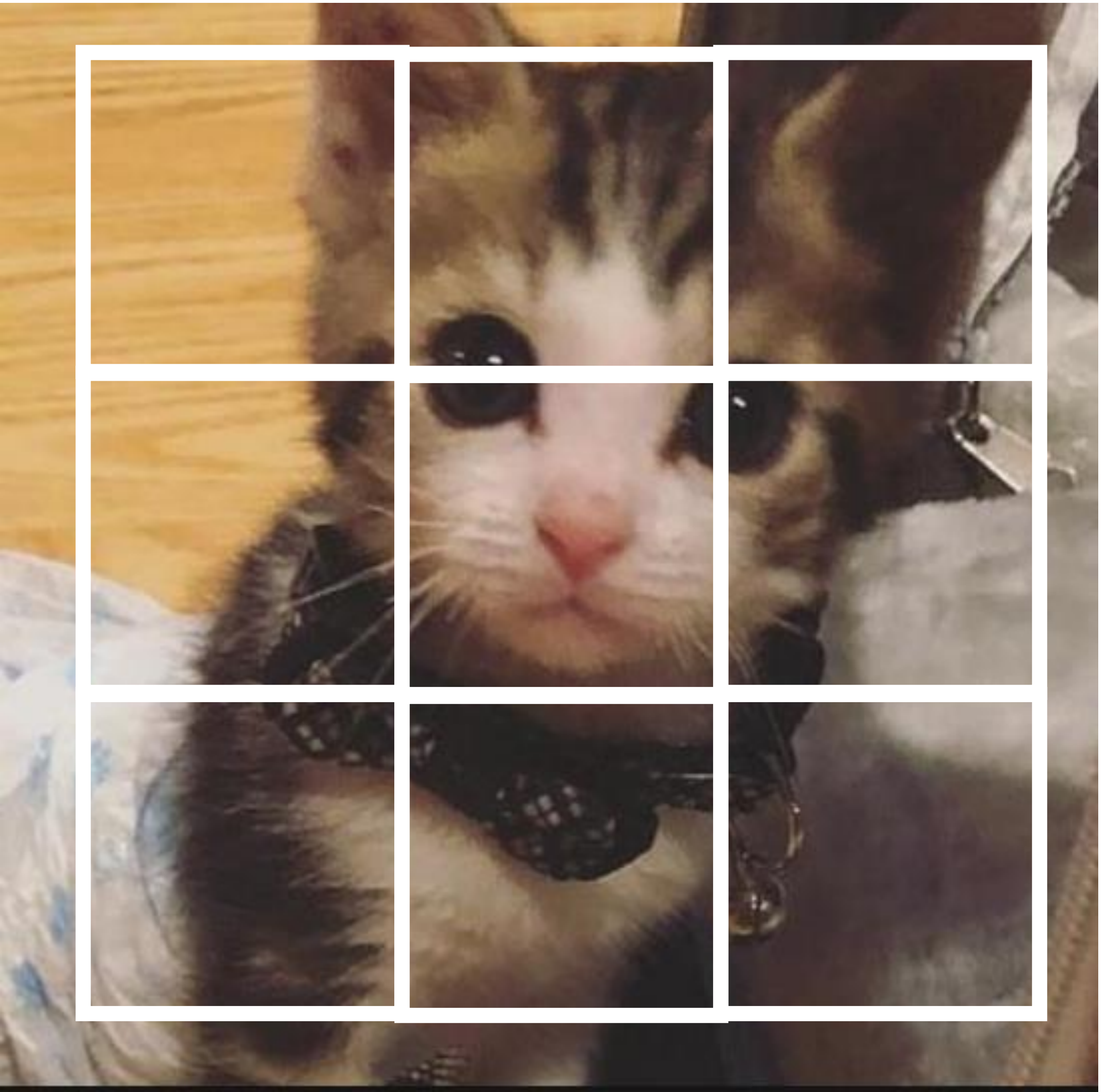
Find the bug

- The network will CHEAT if it can
- When designing a pretext task, care must be taken to ensure that the task forces the network to extract the desired information (high-level semantics, in our case), without taking “trivial” shortcuts.



Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to “**understand**” the data in order to solve it

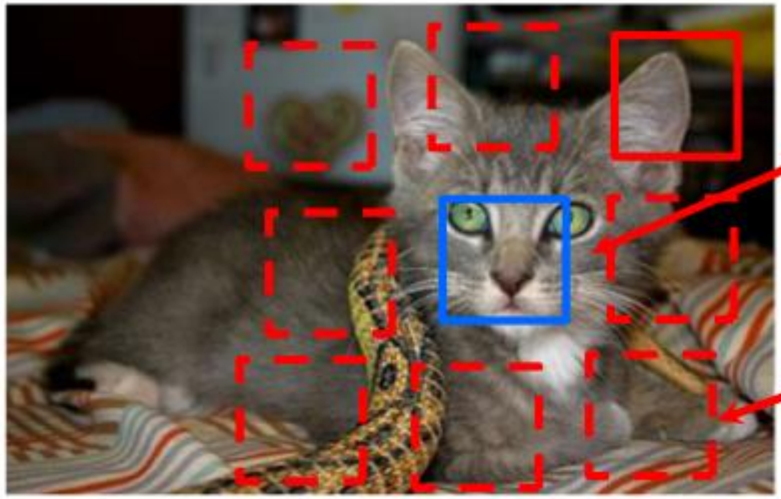
Brainstorm: you are a lazy neural network



You are a network that, given the center patch and one of the others, has to predict the relative position of the second wrt the first (8 possible classes).

Think of lazy ways to solve the problem without actually understanding the image!

Anti-cheat 1 and 2!



Include a gap

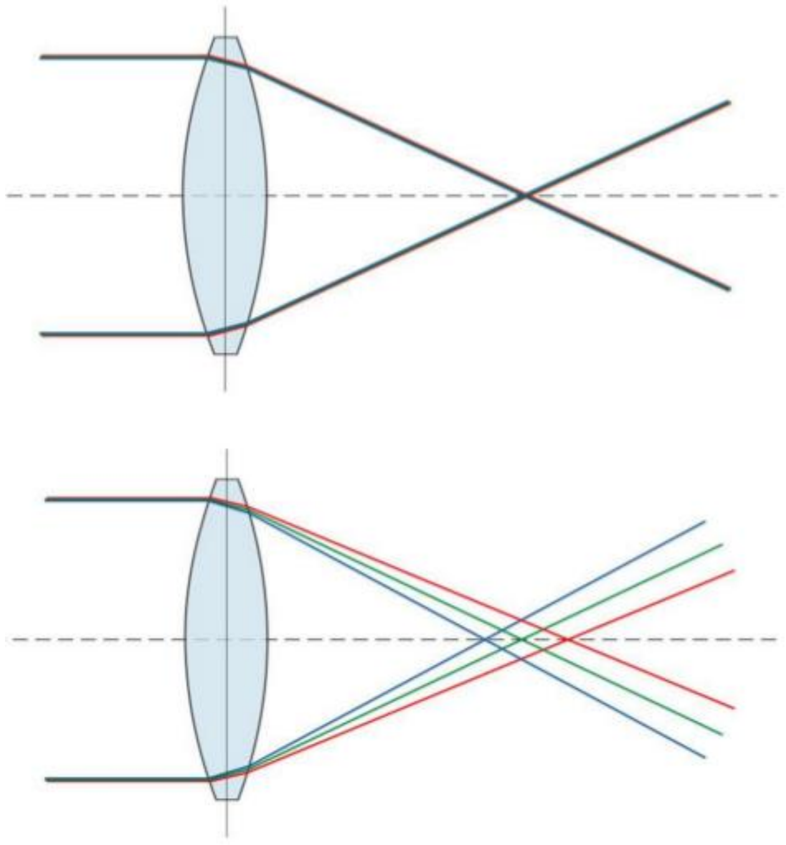
low-level cues like boundary patterns or textures continuing between patches could potentially serve as a lazy shortcut

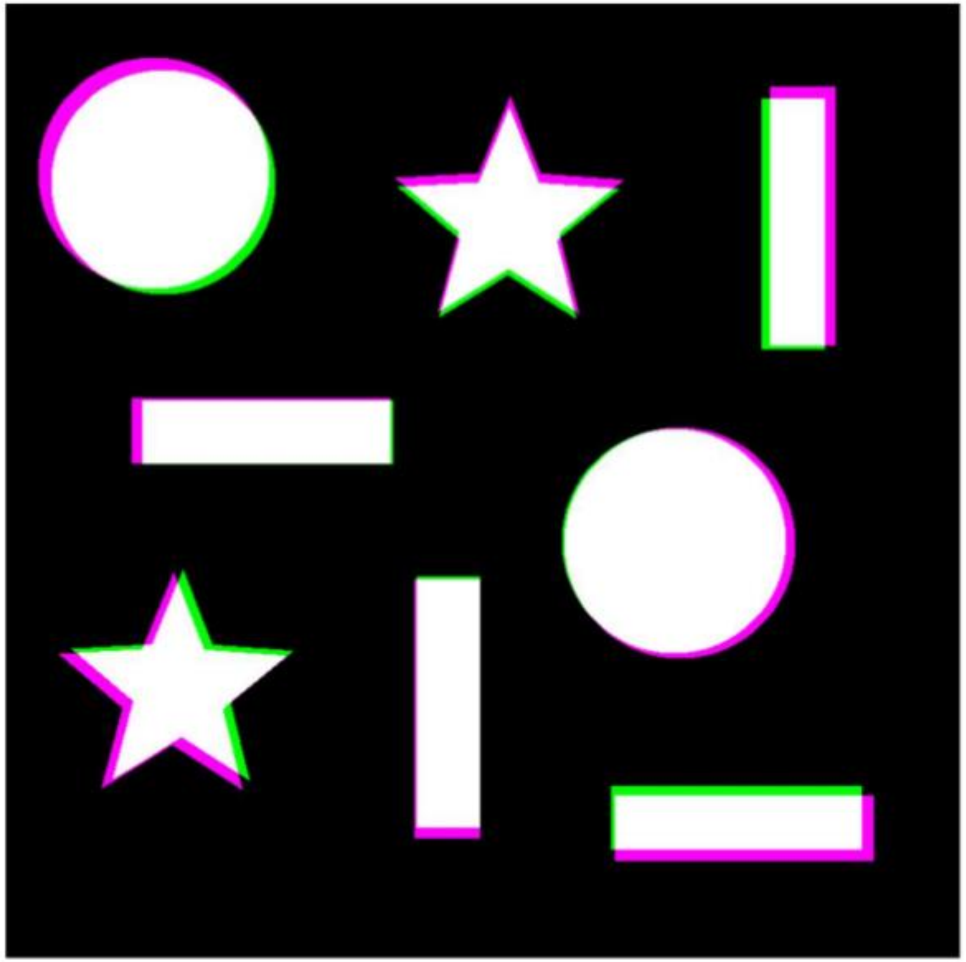
Jitter the patch locations

it is possible that long lines spanning neighboring patches could could give away the correct answer

What is cheat 3? Hint…

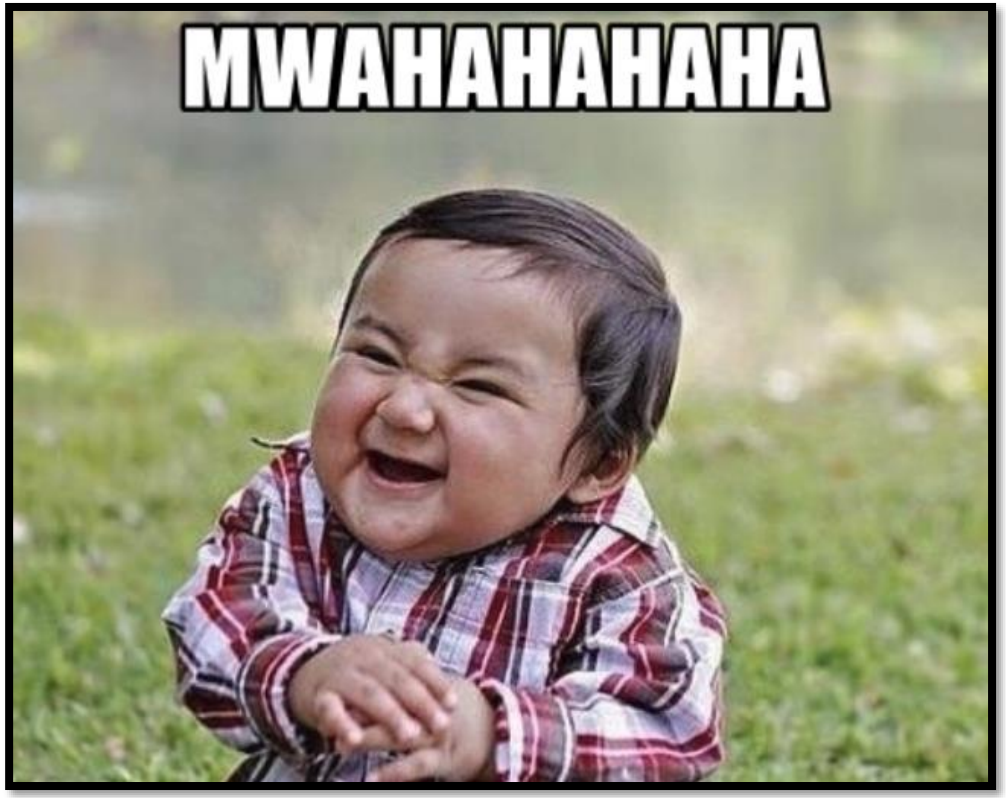
Chromatic aberration





Cheat 3 (genius!)

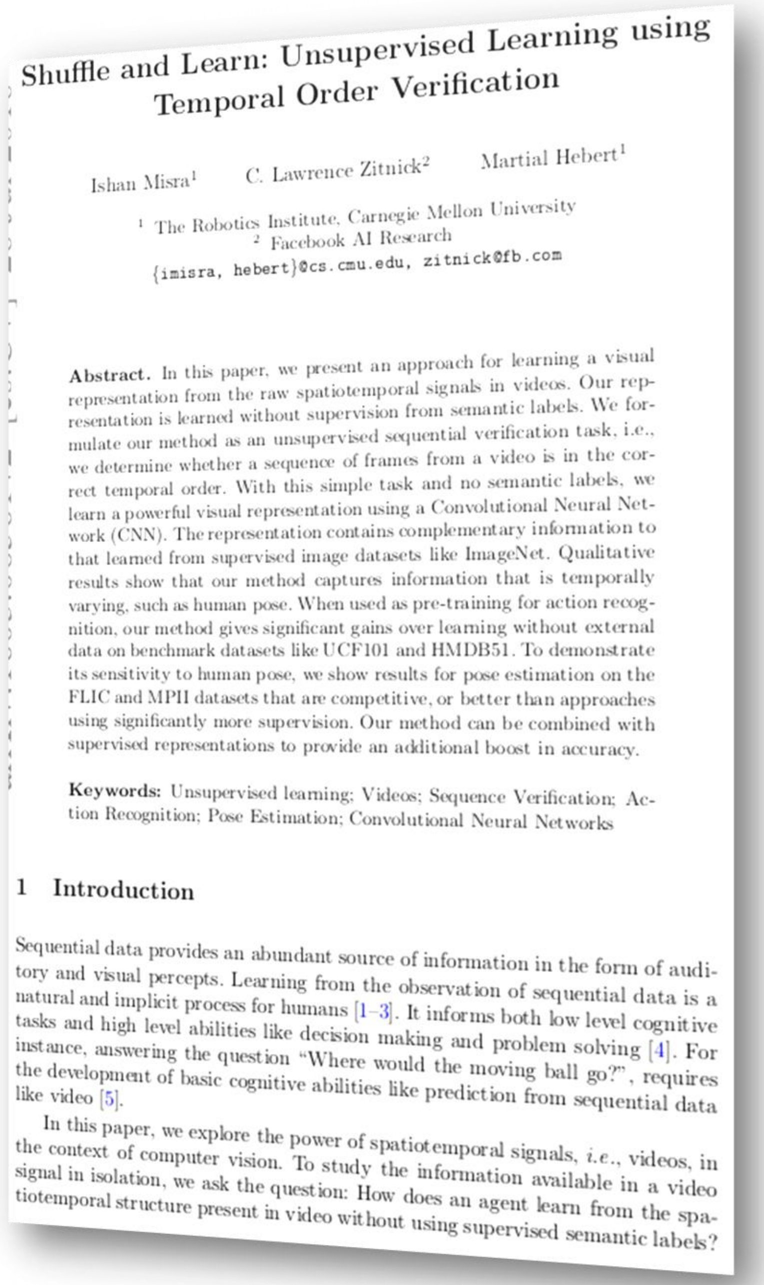
- Chromatic aberration arises from differences in the way the lens focuses light at different wavelengths. In some cameras, one color channel (commonly green) is shrunk toward the image center relative to the others.
- A ConvNet, it turns out, can learn to localize a patch relative to the lens itself simply by detecting the separation between green and magenta (red + blue).
- Once the network learns the absolute location on the lens, solving the relative location task becomes trivial.



Anti-cheat 3

Shuffle and Learn: Unsupervised Learning using Temporal Order Verification

https://arxiv.org/abs/1603.08561, 2016



Shuffle and Learn: Unsupervised Learning using Temporal Order Verification

Ishan Misra[1] C. Lawrence Zitnick[2] Martial Hebert[1]

[1] The Robotics Institute, Carnegie Mellon University
[2] Facebook AI Research
{imisra, hebert}@cs.cmu.edu, zitnick@fb.com

Abstract. In this paper, we present an approach for learning a visual representation from the raw spatiotemporal signals in videos. Our representation is learned without supervision from semantic labels. We formulate our method as an unsupervised sequential verification task, i.e., we determine whether a sequence of frames from a video is in the correct temporal order. With this simple task and no semantic labels, we learn a powerful visual representation using a Convolutional Neural Network (CNN). The representation contains complementary information to that learned from supervised image datasets like ImageNet. Qualitative results show that our method captures information that is temporally varying, such as human pose. When used as pre-training for action recognition, our method gives significant gains over learning without external data on benchmark datasets like UCF101 and HMDB51. To demonstrate its sensitivity to human pose, we show results for pose estimation on the FLIC and MPII datasets that are competitive, or better than approaches using significantly more supervision. Our method can be combined with supervised representations to provide an additional boost in accuracy.

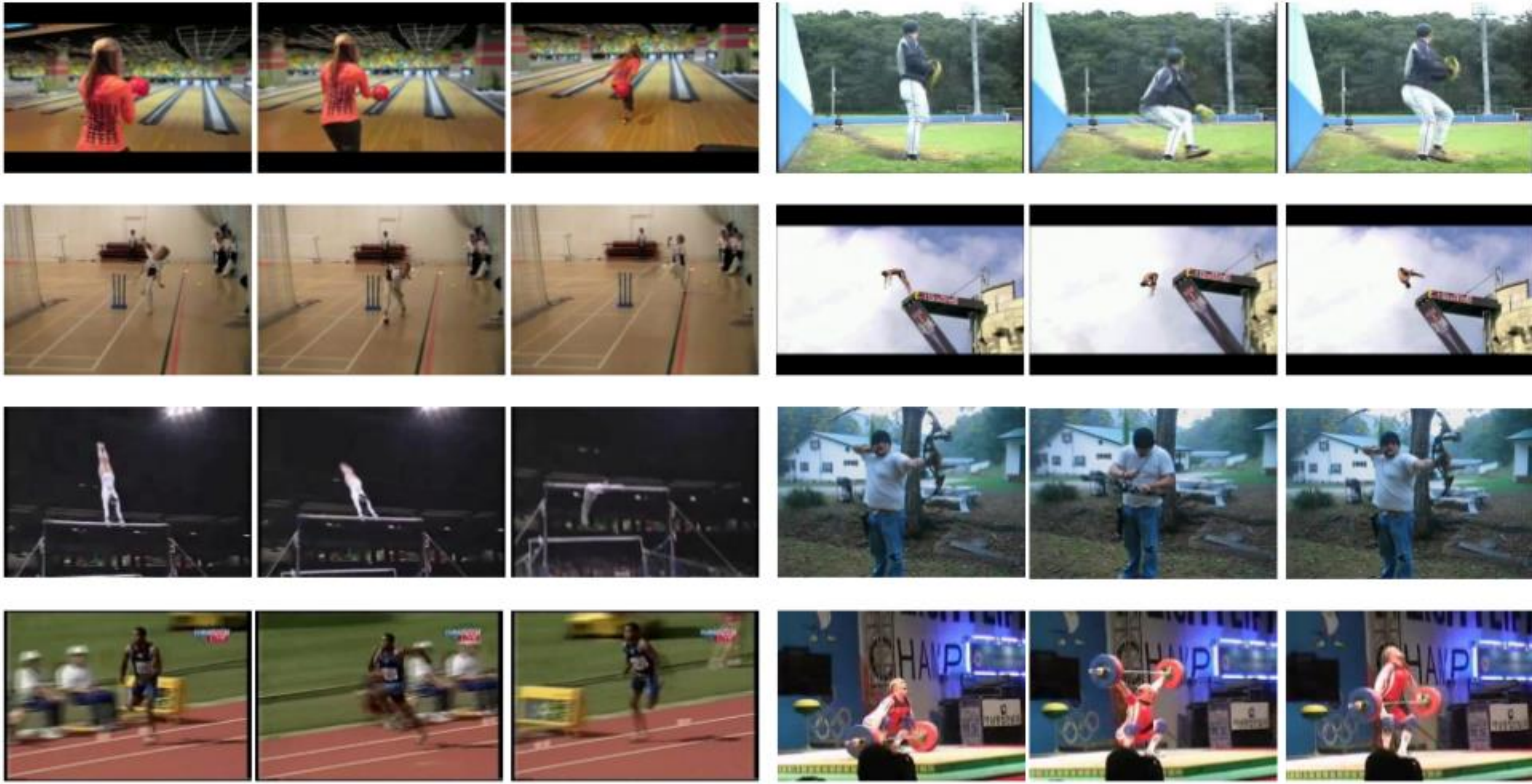
Keywords: Unsupervised learning; Videos; Sequence Verification; Action Recognition; Pose Estimation; Convolutional Neural Networks

1 Introduction

Sequential data provides an abundant source of information in the form of auditory and visual percepts. Learning from the observation of sequential data is a natural and implicit process for humans [1–3]. It informs both low level cognitive tasks and high level abilities like decision making and problem solving [4]. For instance, answering the question "Where would the moving ball go?", requires the development of basic cognitive abilities like prediction from sequential data like video [5].

In this paper, we explore the power of spatiotemporal signals, *i.e.*, videos, in the context of computer vision. To study the information available in a video signal in isolation, we ask the question: How does an agent learn from the spatiotemporal structure present in video without using supervised semantic labels?

Are these frames in the correct order or not?

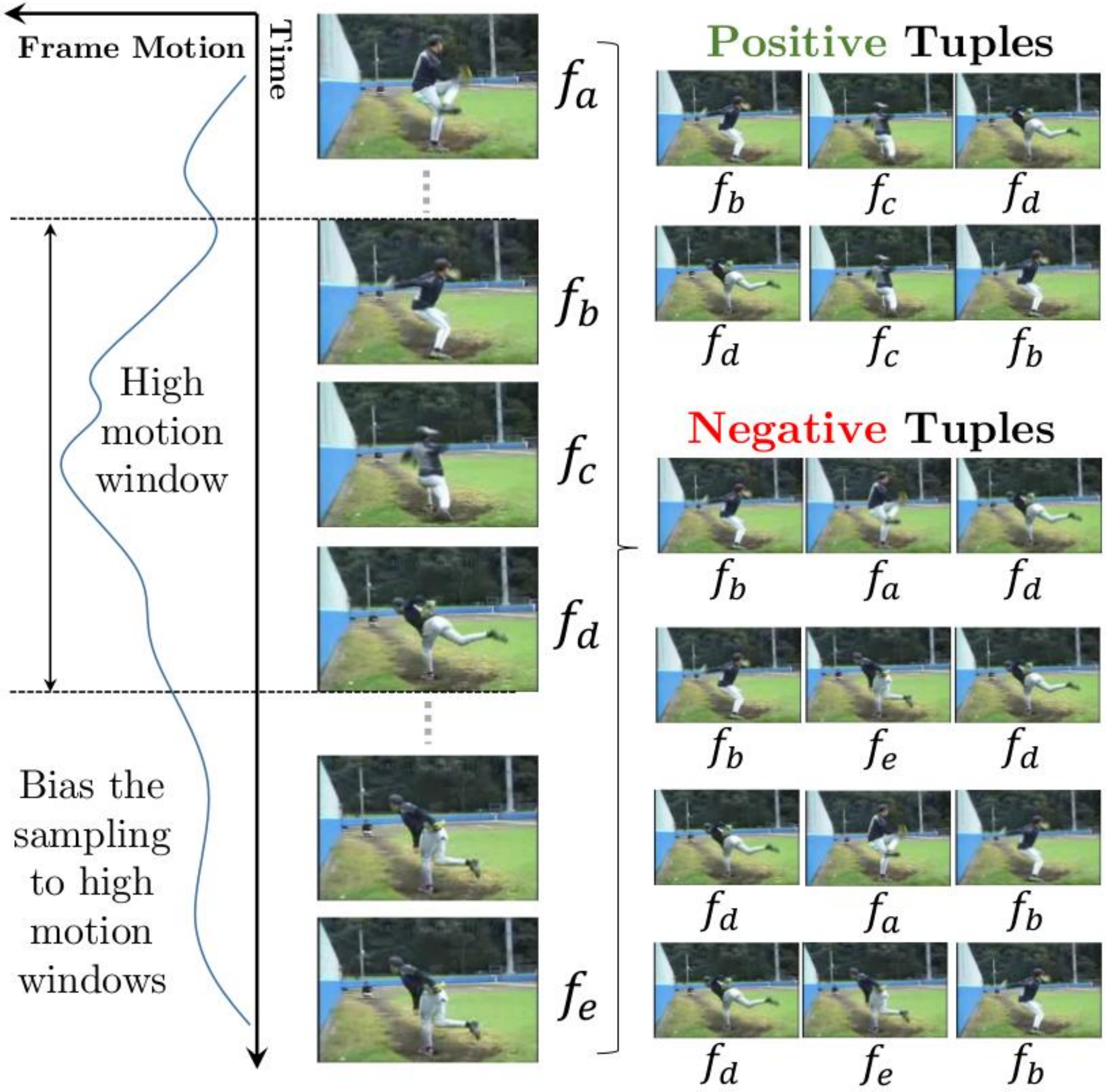


Pretext problem (classification): are these frames in the correct order?

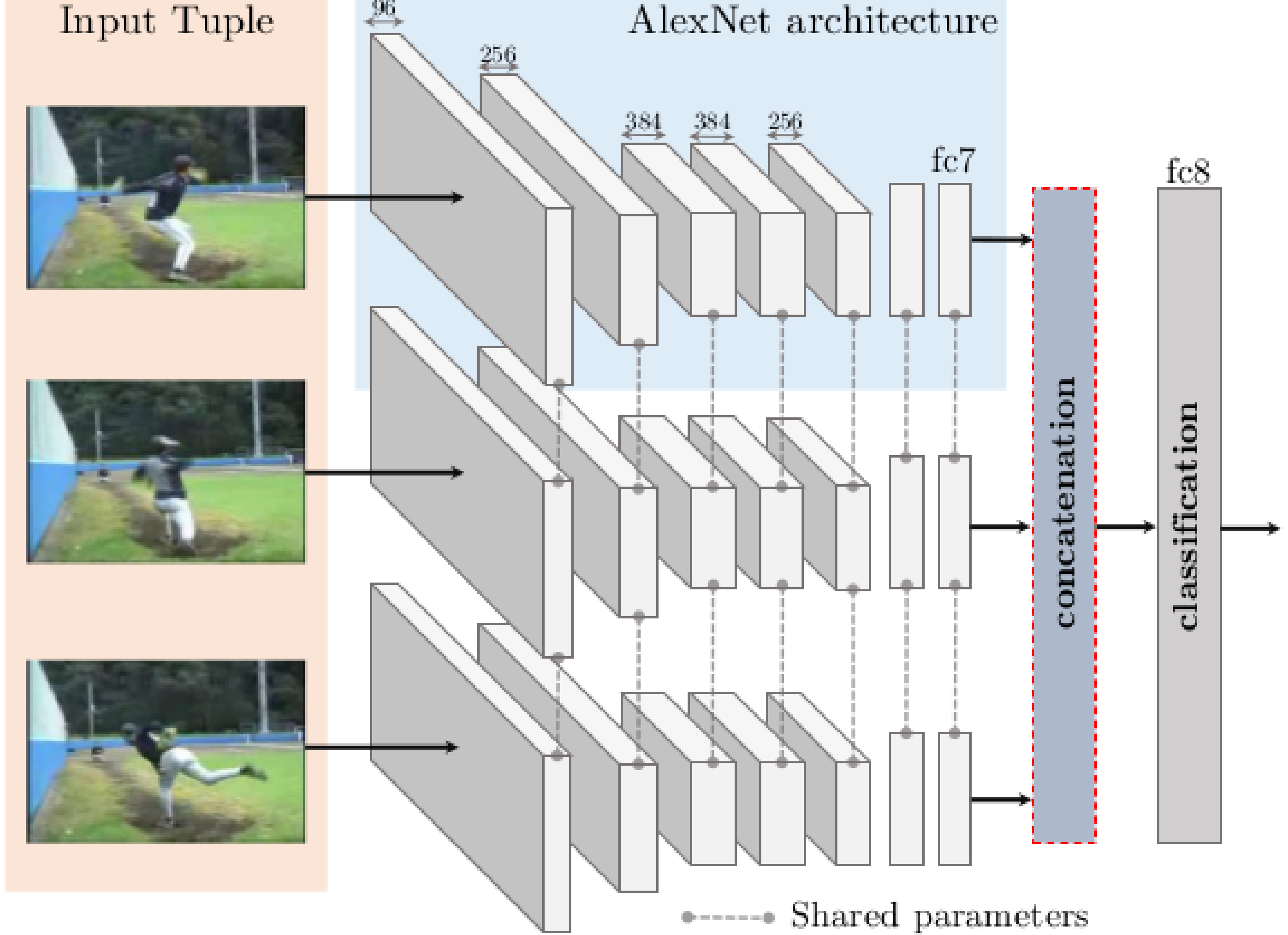
Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to "**understand**" the data in order to solve it

Sampling reasonable instances

- What is the problem if you sample frames from any video?
- That most samples will be impossible to predict due to almost no motion
- Then, only sample from high-motion windows

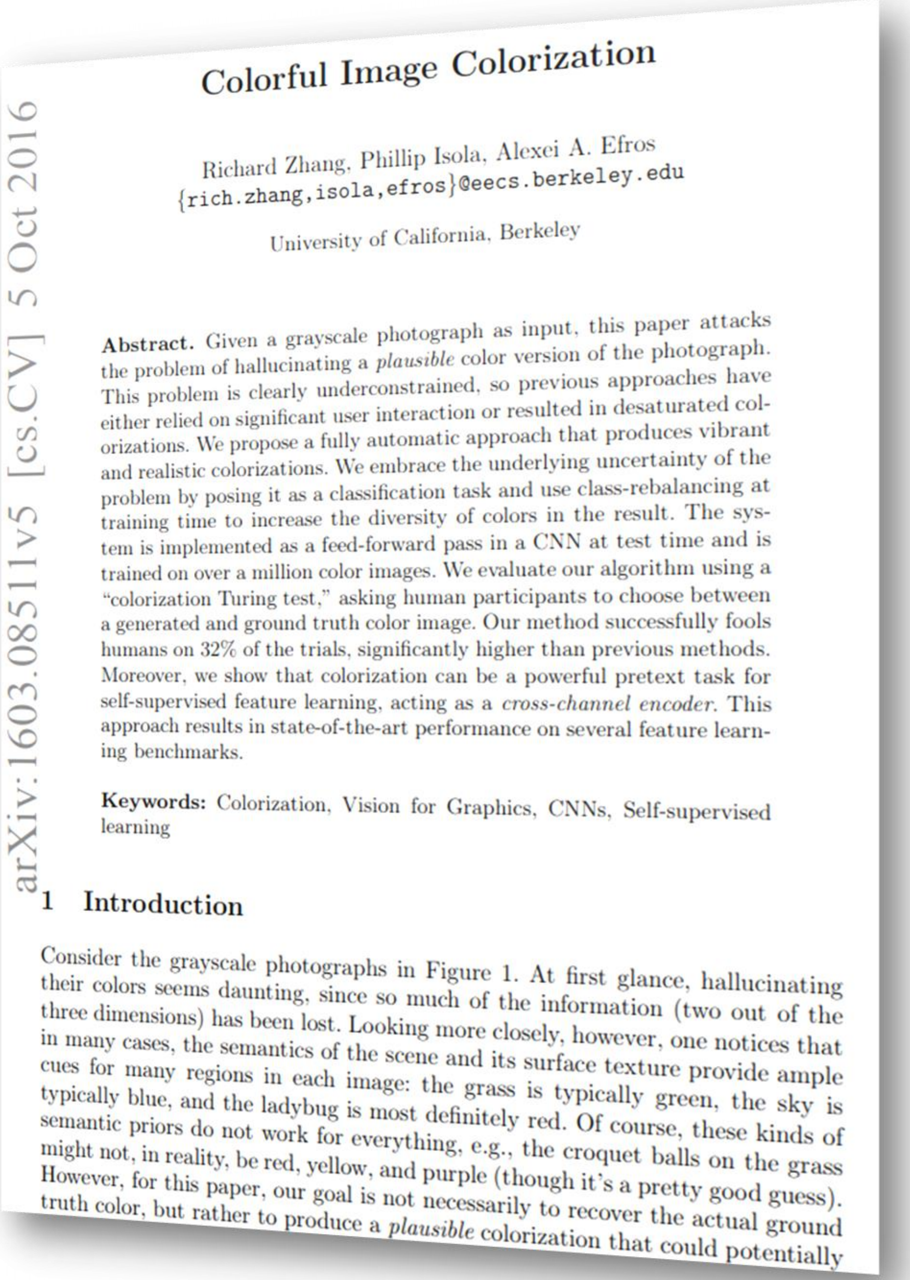


NN architecture



Colorful image colorization

2016



Colorful Image Colorization

Richard Zhang, Phillip Isola, Alexei A. Efros
{rich.zhang,isola,efros}@eecs.berkeley.edu

University of California, Berkeley

Abstract. Given a grayscale photograph as input, this paper attacks the problem of hallucinating a *plausible* color version of the photograph. This problem is clearly underconstrained, so previous approaches have either relied on significant user interaction or resulted in desaturated colorizations. We propose a fully automatic approach that produces vibrant and realistic colorizations. We embrace the underlying uncertainty of the problem by posing it as a classification task and use class-rebalancing at training time to increase the diversity of colors in the result. The system is implemented as a feed-forward pass in a CNN at test time and is trained on over a million color images. We evaluate our algorithm using a "colorization Turing test," asking human participants to choose between a generated and ground truth color image. Our method successfully fools humans on 32% of the trials, significantly higher than previous methods. Moreover, we show that colorization can be a powerful pretext task for self-supervised feature learning, acting as a *cross-channel encoder*. This approach results in state-of-the-art performance on several feature learning benchmarks.

Keywords: Colorization, Vision for Graphics, CNNs, Self-supervised learning

1 Introduction

Consider the grayscale photographs in Figure 1. At first glance, hallucinating their colors seems daunting, since so much of the information (two out of the three dimensions) has been lost. Looking more closely, however, one notices that in many cases, the semantics of the scene and its surface texture provide ample cues for many regions in each image: the grass is typically green, the sky is typically blue, and the ladybug is most definitely red. Of course, these kinds of semantic priors do not work for everything, e.g., the croquet balls on the grass might not, in reality, be red, yellow, and purple (though it's a pretty good guess). However, for this paper, our goal is not necessarily to recover the actual ground truth color, but rather to produce a *plausible* colorization that could potentially

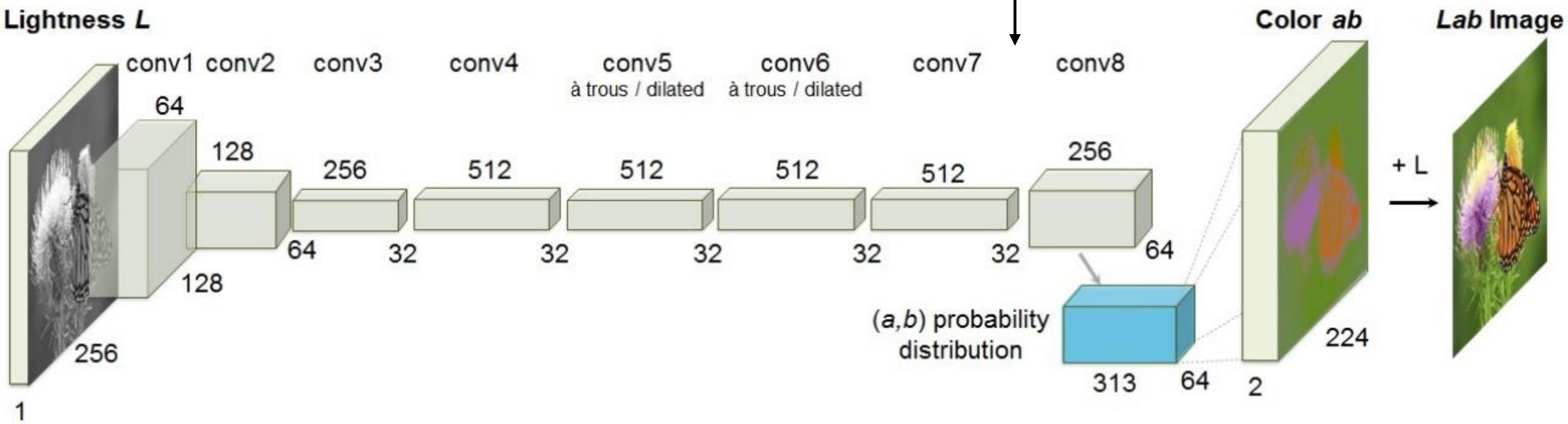
Image colorization (hallucinate colors)

Pretext task desiderata:

- you can get a ground truth for free from the data itself
- you need to "**understand**" the data in order to solve it

Main idea

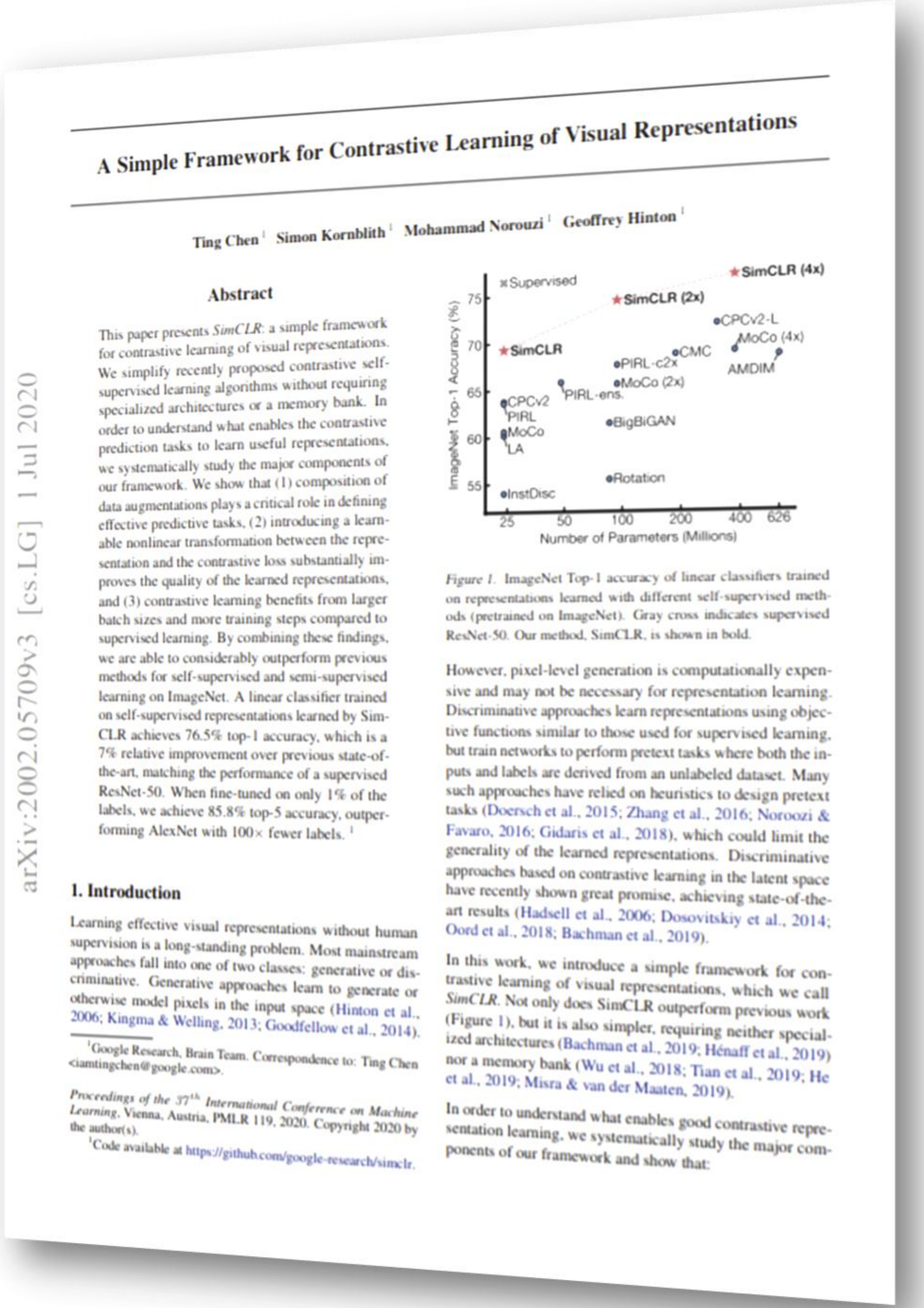
Learned representation



SimCLR: Contrastive Learning of Visual Representations

https://arxiv.org/pdf/2002.05709.pdf

2020



A Simple Framework for Contrastive Learning of Visual Representations

Ting Chen [1] Simon Kornblith [1] Mohammad Norouzi [1] Geoffrey Hinton [1]

arXiv:2002.05709v3 [cs.LG] 1 Jul 2020

Abstract

This paper presents *SimCLR*: a simple framework for contrastive learning of visual representations. We simplify recently proposed contrastive self-supervised learning algorithms without requiring specialized architectures or a memory bank. In order to understand what enables the contrastive prediction tasks to learn useful representations, we systematically study the major components of our framework. We show that (1) composition of data augmentations plays a critical role in defining effective predictive tasks, (2) introducing a learnable nonlinear transformation between the representation and the contrastive loss substantially improves the quality of the learned representations, and (3) contrastive learning benefits from larger batch sizes and more training steps compared to supervised learning. By combining these findings, we are able to considerably outperform previous methods for self-supervised and semi-supervised learning on ImageNet. A linear classifier trained on self-supervised representations learned by SimCLR achieves 76.5% top-1 accuracy, which is a 7% relative improvement over previous state-of-the-art, matching the performance of a supervised ResNet-50. When fine-tuned on only 1% of the labels, we achieve 85.8% top-5 accuracy, outperforming AlexNet with 100× fewer labels. [1]

1. Introduction

Learning effective visual representations without human supervision is a long-standing problem. Most mainstream approaches fall into one of two classes: generative or discriminative. Generative approaches learn to generate or otherwise model pixels in the input space (Hinton et al., 2006; Kingma & Welling, 2013; Goodfellow et al., 2014).

[1]Google Research, Brain Team. Correspondence to: Ting Chen <iamtingchen@google.com>.

Proceedings of the 37th International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020. Copyright 2020 by the author(s).

[1]Code available at https://github.com/google-research/simclr.

Supervised
SimCLR (4x)
SimCLR (2x)
SimCLR
CPCv2-L
MoCo (4x)
CMC
AMDIM
PIRL-c2x
MoCo (2x)
PIRL-ens.
CPCv2
PIRL
MoCo
LA
BigBiGAN
Rotation
InstDisc
ImageNet Top-1 Accuracy (%)
Number of Parameters (Millions)

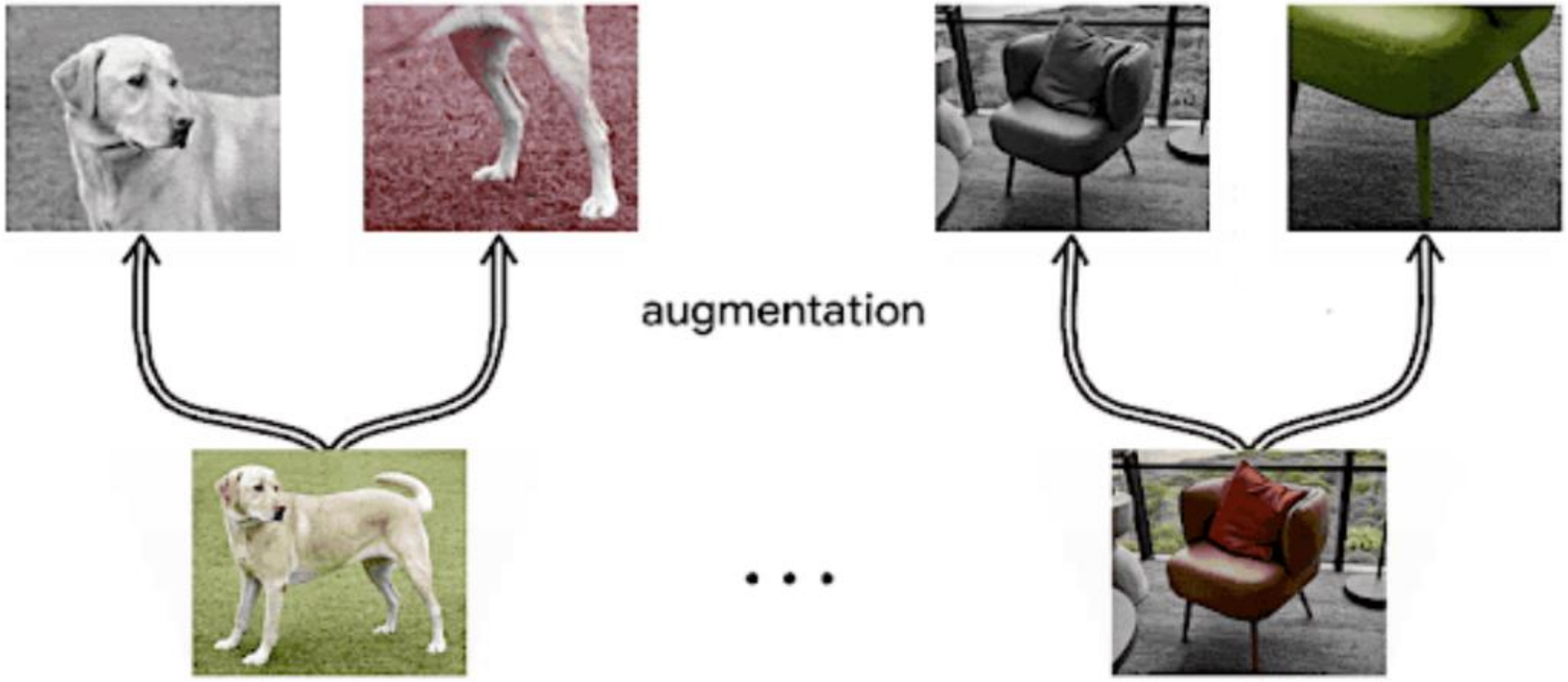
Figure 1. ImageNet Top-1 accuracy of linear classifiers trained on representations learned with different self-supervised methods (pretrained on ImageNet). Gray cross indicates supervised ResNet-50. Our method, SimCLR, is shown in bold.

However, pixel-level generation is computationally expensive and may not be necessary for representation learning. Discriminative approaches learn representations using objective functions similar to those used for supervised learning, but train networks to perform pretext tasks where both the inputs and labels are derived from an unlabeled dataset. Many such approaches have relied on heuristics to design pretext tasks (Doersch et al., 2015; Zhang et al., 2016; Noroozi & Favaro, 2016; Gidaris et al., 2018), which could limit the generality of the learned representations. Discriminative approaches based on contrastive learning in the latent space have recently shown great promise, achieving state-of-the-art results (Hadsell et al., 2006; Dosovitskiy et al., 2014; Oord et al., 2018; Bachman et al., 2019).

In this work, we introduce a simple framework for contrastive learning of visual representations, which we call *SimCLR*. Not only does SimCLR outperform previous work (Figure 1), but it is also simpler, requiring neither specialized architectures (Bachman et al., 2019; Hénaff et al., 2019) nor a memory bank (Wu et al., 2018; Tian et al., 2019; He et al., 2019; Misra & van der Maaten, 2019).

In order to understand what enables good contrastive representation learning, we systematically study the major components of our framework and show that:

What can we expect about the representations of these four images?



Main idea

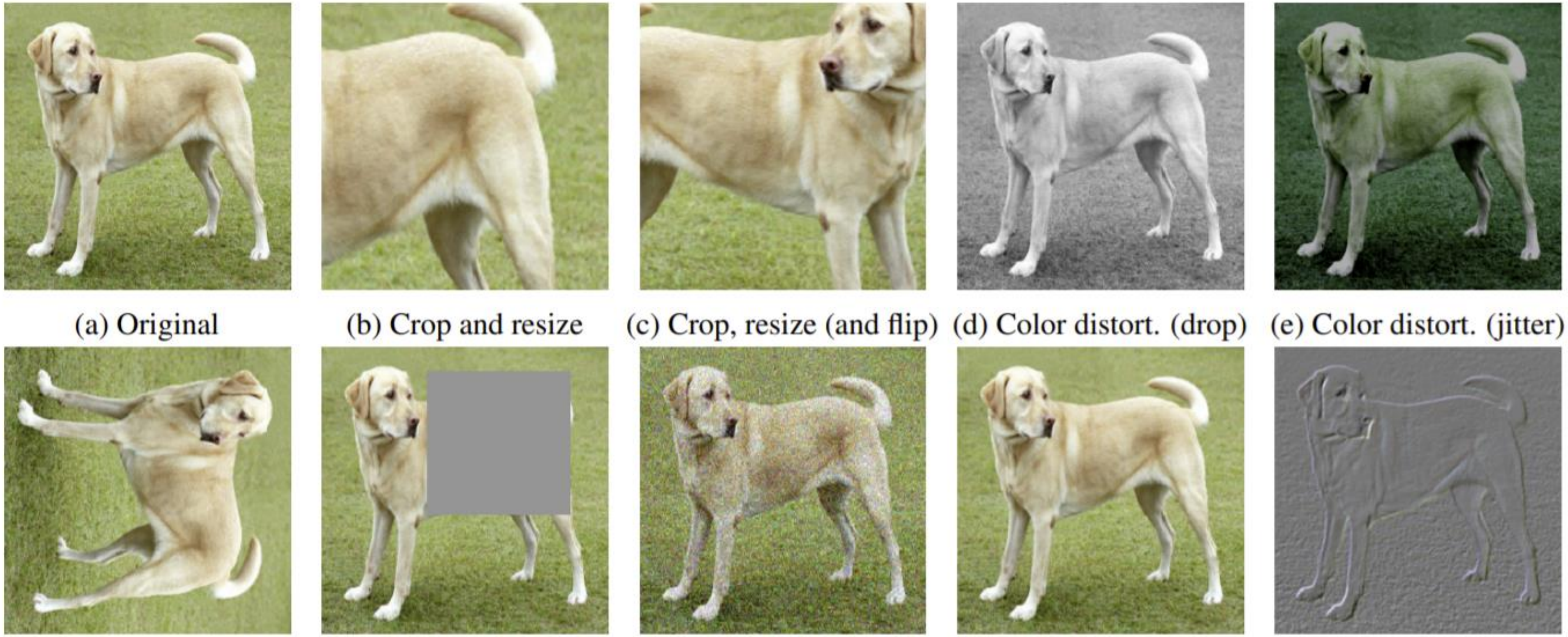
Use a contrastive learning loss to train CNN and MLP such that:

- similar outputs for different augmentations of the same image
- different outputs for different images

https://ai.googleblog.com/2020/04/advancing-self-supervised-and-semi.html

Which augmentation is best?

Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to "**understand**" the data in order to solve it



(a) Original (b) Crop and resize (c) Crop, resize (and flip) (d) Color distort. (drop) (e) Color distort. (jitter)

(f) Rotate $\{90°, 180°, 270°\}$ (g) Cutout (h) Gaussian noise (i) Gaussian blur (j) Sobel filtering

Seems fine… ?

(a) Original (c) Crop, resize (and flip)

I can be lazy and just check if the color histograms approximately match!

Seems fine… ?

(a) Original (e) Color distort. (jitter)

I can be lazy and just check if the geometry approximately matches!

The data!

- Any individual augmentation is not very helpful
- Applying two augmentations at the same time (Color and Crop) forces the model to actually learn semantics!

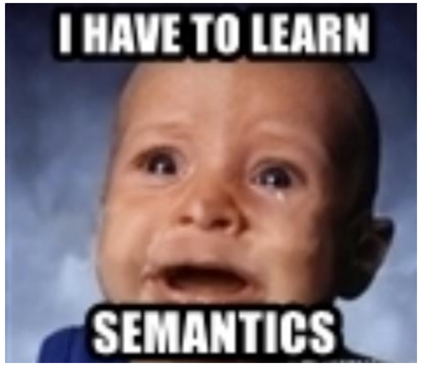


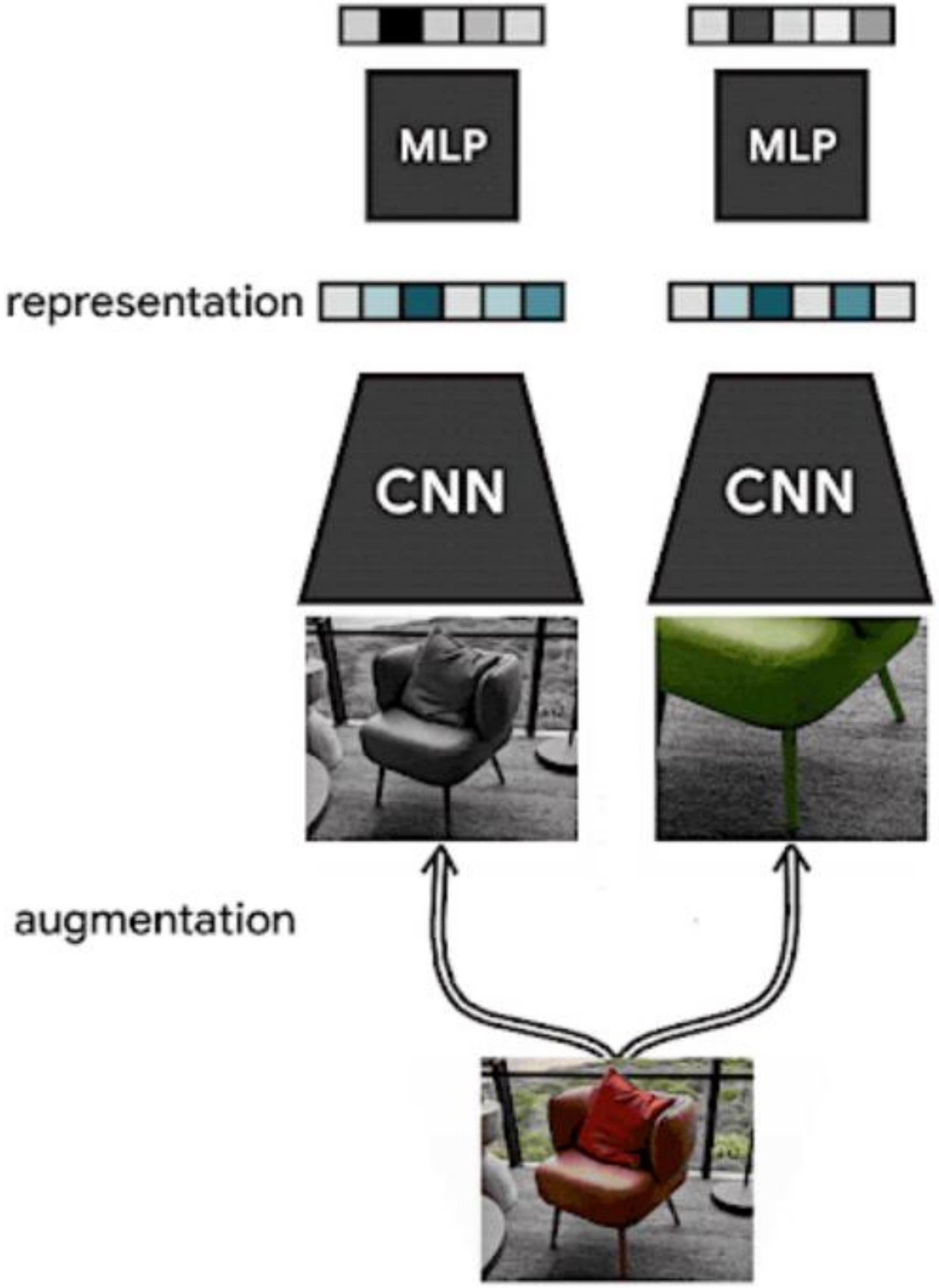
Figure 5. Linear evaluation (ImageNet top-1 accuracy) under individual or composition of data augmentations, applied only to one branch. For all columns but the last, diagonal entries correspond to single transformation, and off-diagonals correspond to composition of two transformations (applied sequentially). The last column reflects the average over the row.

Why the projection?

It turns out that the best representation to use for downstream tasks is not the MLP output, but its input.

But the MLP is useful during training. Why?

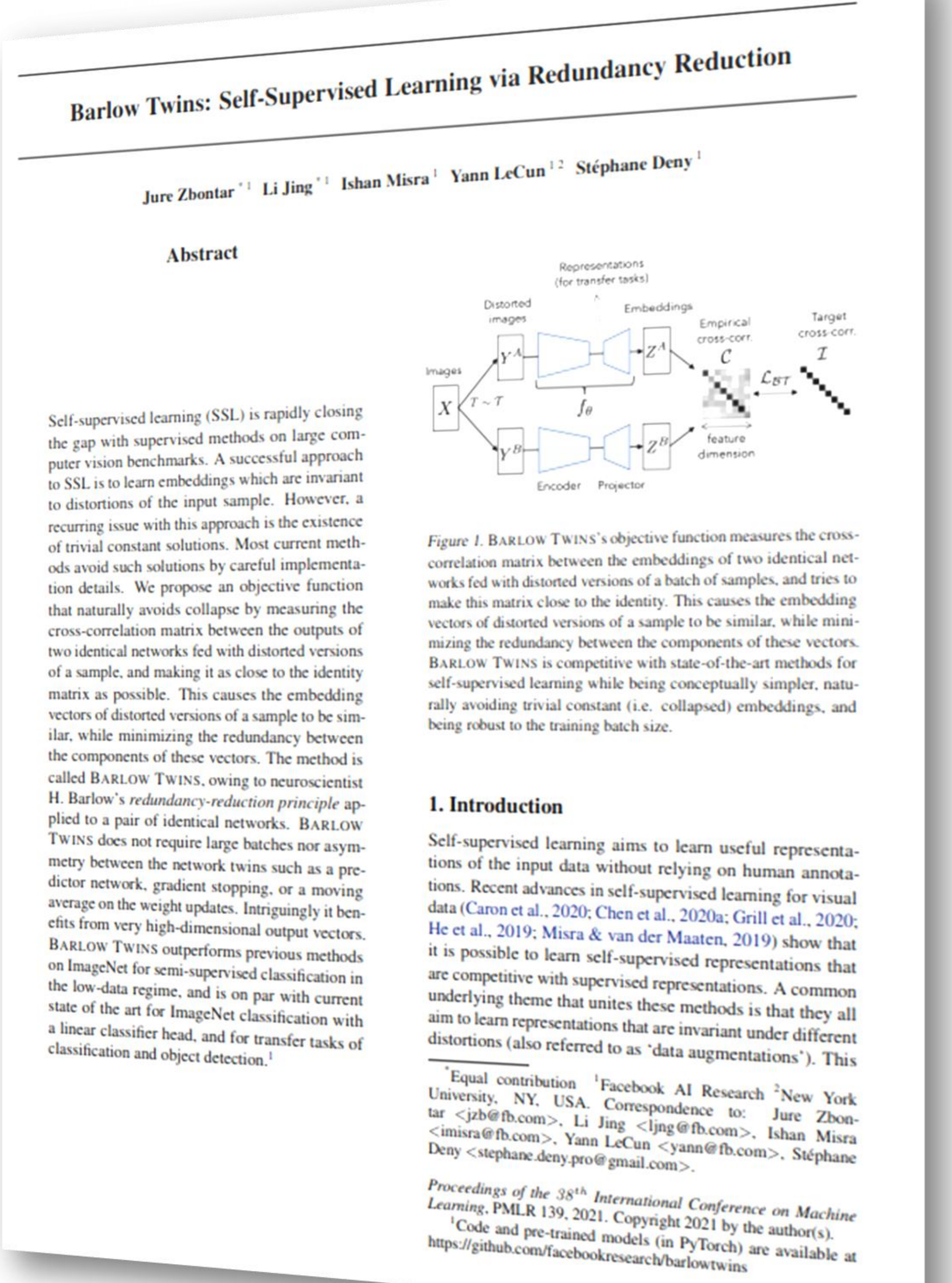
The MLP *loses information*, e.g. color, in order to achieve the contrastive loss. This information might be relevant for downstream tasks!



Barlow Twins: Self-Supervised Learning via Redundancy Reduction

https://arxiv.org/pdf/2103.03230.pdf
2021

The method is called Barlow Twins, owing to neuroscientist H. Barlow's redundancy-reduction principle applied to a pair of identical networks.



Barlow Twins: Self-Supervised Learning via Redundancy Reduction

Jure Zbontar [*1] Li Jing [*1] Ishan Misra [1] Yann LeCun [1 2] Stéphane Deny [1]

Abstract

Self-supervised learning (SSL) is rapidly closing the gap with supervised methods on large computer vision benchmarks. A successful approach to SSL is to learn embeddings which are invariant to distortions of the input sample. However, a recurring issue with this approach is the existence of trivial constant solutions. Most current methods avoid such solutions by careful implementation details. We propose an objective function that naturally avoids collapse by measuring the cross-correlation matrix between the outputs of two identical networks fed with distorted versions of a sample, and making it as close to the identity matrix as possible. This causes the embedding vectors of distorted versions of a sample to be similar, while minimizing the redundancy between the components of these vectors. The method is called BARLOW TWINS, owing to neuroscientist H. Barlow's *redundancy-reduction principle* applied to a pair of identical networks. BARLOW TWINS does not require large batches nor asymmetry between the network twins such as a predictor network, gradient stopping, or a moving average on the weight updates. Intriguingly it benefits from very high-dimensional output vectors. BARLOW TWINS outperforms previous methods on ImageNet for semi-supervised classification in the low-data regime, and is on par with current state of the art for ImageNet classification with a linear classifier head, and for transfer tasks of classification and object detection.[1]

Figure 1. BARLOW TWINS's objective function measures the cross-correlation matrix between the embeddings of two identical networks fed with distorted versions of a batch of samples, and tries to make this matrix close to the identity. This causes the embedding vectors of distorted versions of a sample to be similar, while minimizing the redundancy between the components of these vectors. BARLOW TWINS is competitive with state-of-the-art methods for self-supervised learning while being conceptually simpler, naturally avoiding trivial constant (i.e. collapsed) embeddings, and being robust to the training batch size.

1. Introduction

Self-supervised learning aims to learn useful representations of the input data without relying on human annotations. Recent advances in self-supervised learning for visual data (Caron et al., 2020; Chen et al., 2020a; Grill et al., 2020; He et al., 2019; Misra & van der Maaten, 2019) show that it is possible to learn self-supervised representations that are competitive with supervised representations. A common underlying theme that unites these methods is that they all aim to learn representations that are invariant under different distortions (also referred to as 'data augmentations'). This

[*]Equal contribution [1]Facebook AI Research [2]New York University, NY, USA. Correspondence to: Jure Zbontar <jzb@fb.com>, Li Jing <ljng@fb.com>, Ishan Misra <imisra@fb.com>, Yann LeCun <yann@fb.com>, Stéphane Deny <stephane.deny.pro@gmail.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

[1]Code and pre-trained models (in PyTorch) are available at https://github.com/facebookresearch/barlowtwins

Main idea

When we train a visual classification model, our ideal features are:

- Invariant to transformations that do not affect the class
- Not correlated to each other

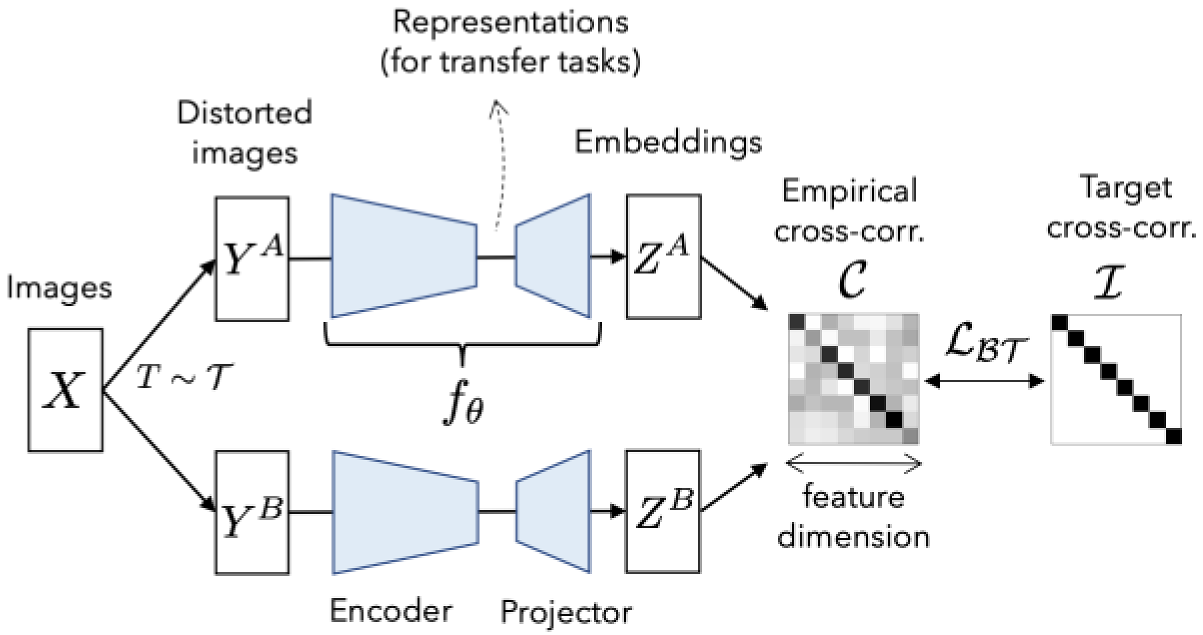
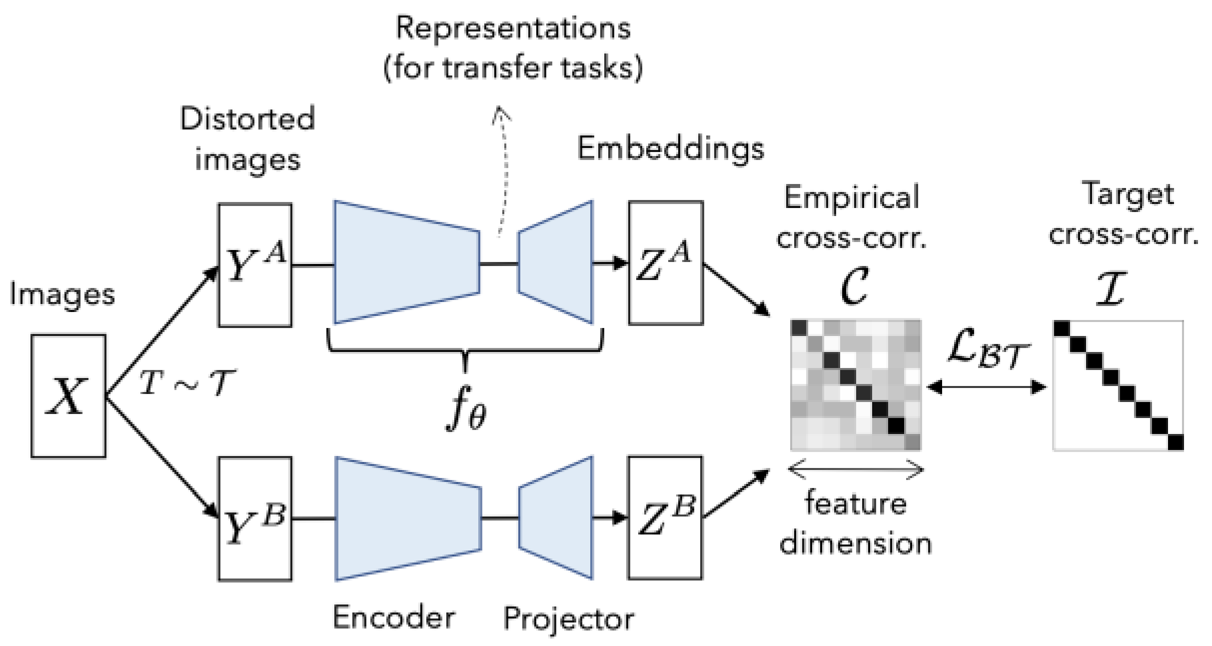
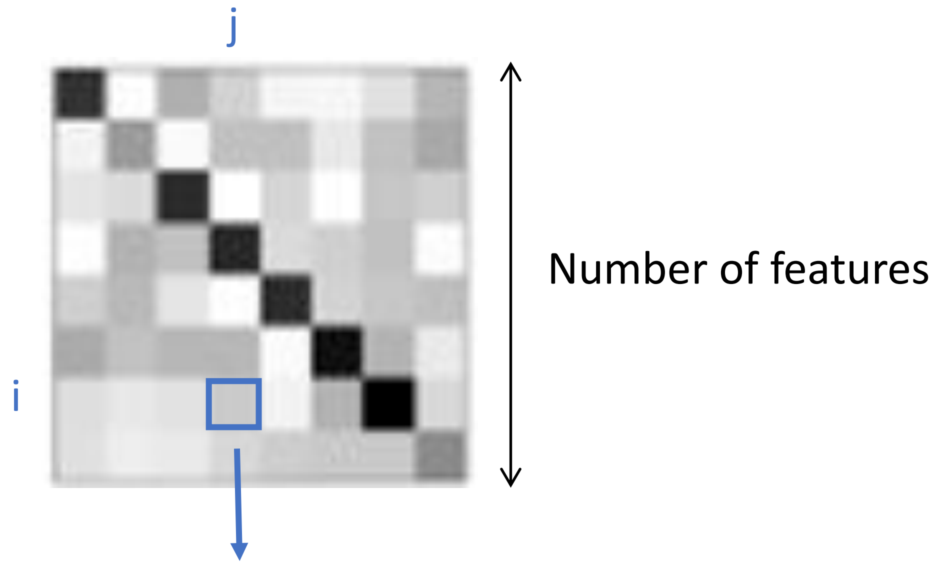


Figure 1. BARLOW TWINS's objective function measures the cross-correlation matrix between the embeddings of two identical networks fed with distorted versions of a batch of samples, and tries to make this matrix close to the identity. This causes the embedding vectors of distorted versions of a sample to be similar, while minimizing the redundancy between the components of these vectors. BARLOW TWINS is competitive with state-of-the-art methods for self-supervised learning while being conceptually simpler, naturally avoiding trivial constant (i.e. collapsed) embeddings, and being robust to the training batch size.

More about the cross correlation matrix…





Correlation between two vectors:

- feature i in Z^A for all samples in the batch
- feature j in Z^B for all samples in the batch

Self-supervised deep learning conclusions

- You are interested in solving problem A
- Take a lot of data similar to the one you'll use, without labels
  (of course: you are lazy)
- Invent a problem B (*pretext task*) on the data for which
  - you can get a ground truth for free from the data itself
  - you need to "**understand**" the data in order to solve it
- Train a network for B

→ The network has learned something valuable for A, i.e. to understand the data

Plan of the lecture

- Part **1**: introduction
- Part **2**: warm-up on the CIFAR-10 dataset
- Part 3: what is self-supervised learning?
- **Part 4: implement&test a simple self-supervised learning method**
- Part **5**: some examples of self-supervised learning in robotics

We are now going try on CIFAR-10!

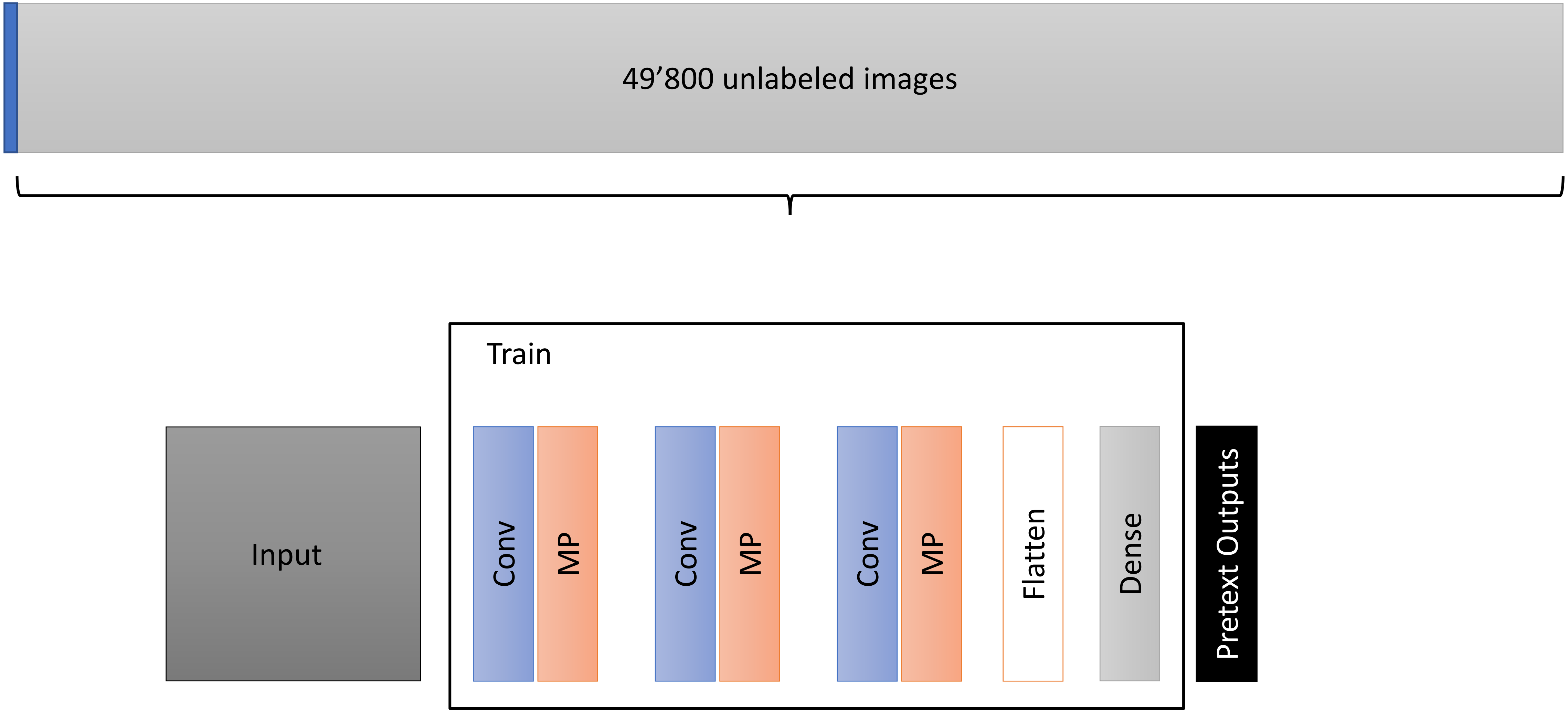
What we did before…

50’000 labeled images

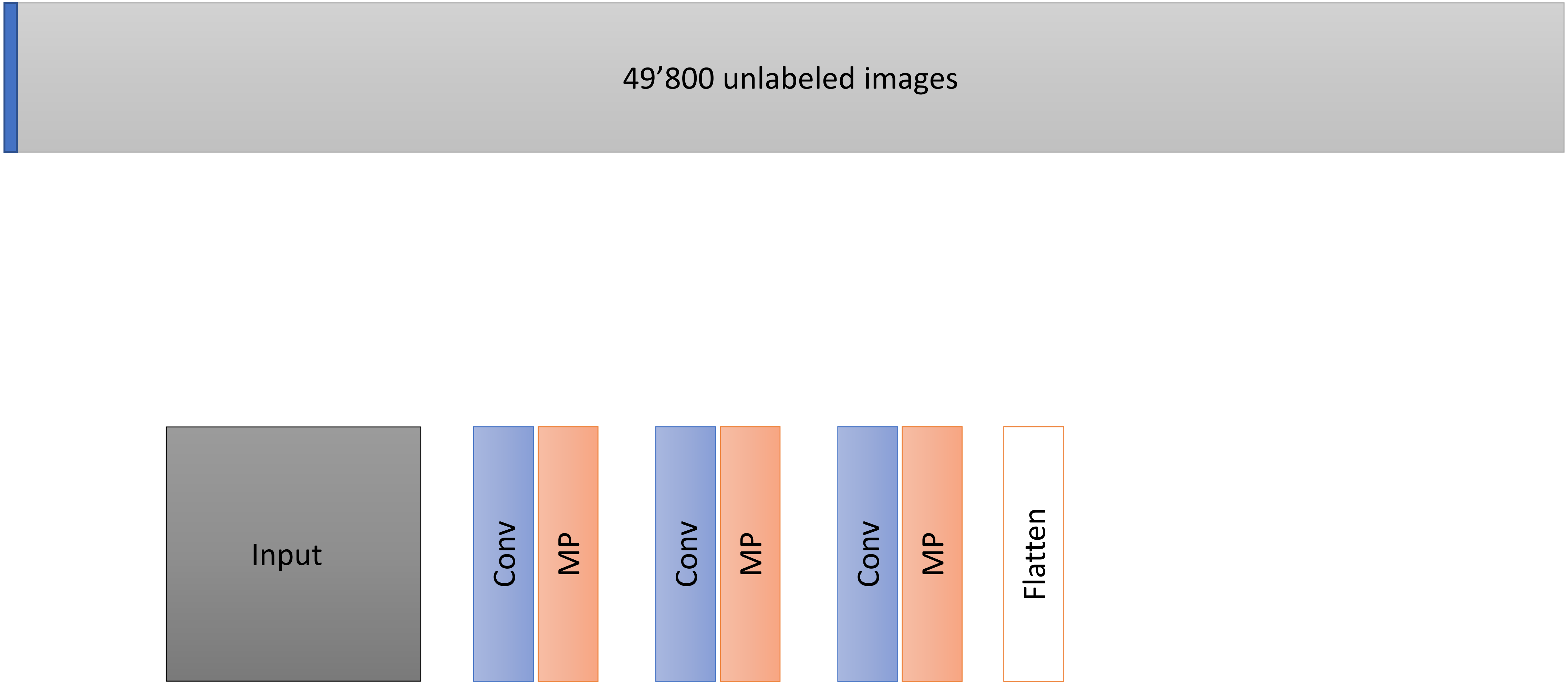
What we are going to do now…

49’800 unlabeled images

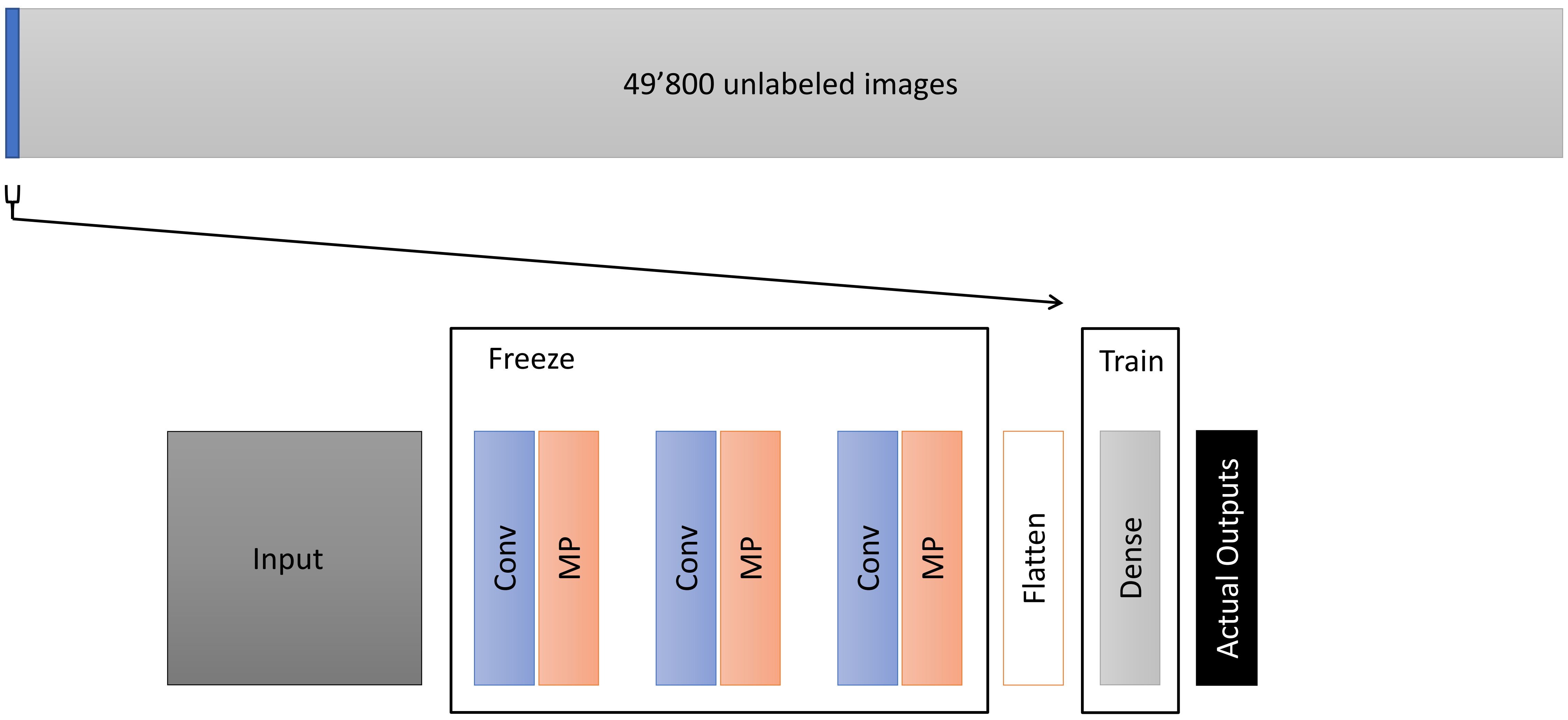
200 labeled images



Step 1: train the model on the pretext task using all unlabeled images



Step 2: discard the classification layer



Step 3: freeze the convolutional layers and train a new classification layer using only labeled data

Which pretext task should we implement?

Pretext task desiderata:
- you can get a ground truth for free from the data itself
- you need to "**understand**" the data in order to solve it