AIL 862

Lecture 12

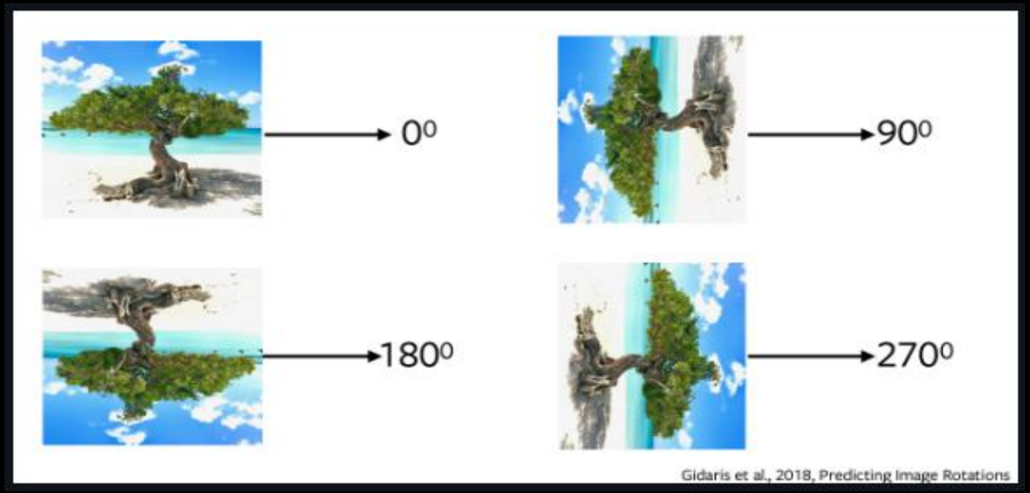
Self-Supervised Learning

- Supervised learning tasks have pre-defined (and generally human-provided) labels.
- Unsupervised learning has just the data samples without any supervision, label or correct output.
- Self-supervised learning derives its labels from a co-occurring modality for the given data sample or from a co-occurring part of the data sample itself.

Pre-Text Task

- The pretext task is the self-supervised learning task solved to learn visual representations.

- E.g., Rotation of images



Assumption

- Accuracy in pre-text tasks is closely linked to accuracy in downstream task.

- Generally, more difficult pre-text task will satisfy the assumption better.

Context Prediction, 2016

- Key Idea: By predicting the relative position of image patches, the model learns to understand object structures and scene layouts.

Context Prediction

- Patch Extraction: Extract a central patch and one of its eight neighboring patches. This results in pairs of patches with known spatial relationships.

- Prediction Task: Train a CNN to predict the position of the neighboring patch relative to the central patch. The network learns to classify the relative position into one of eight possible categories.

Context Prediction

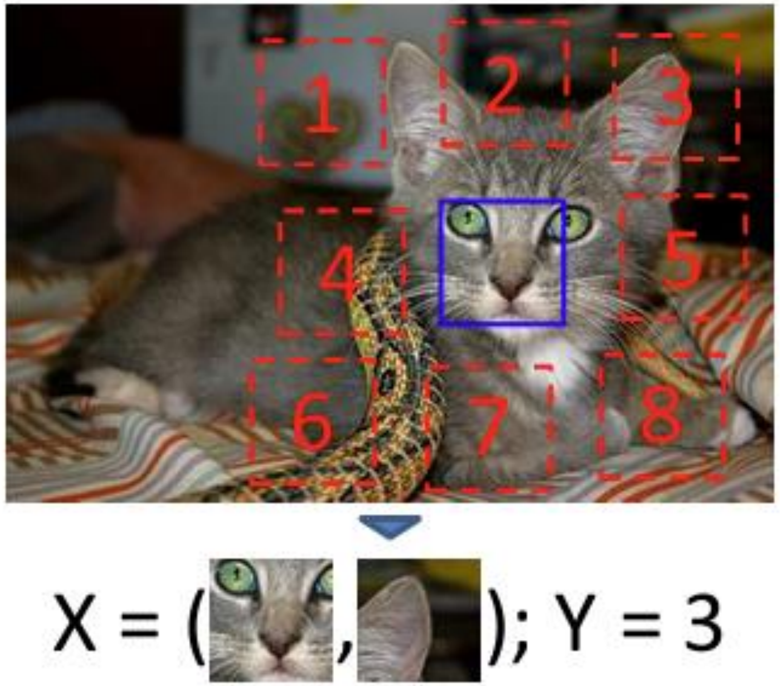
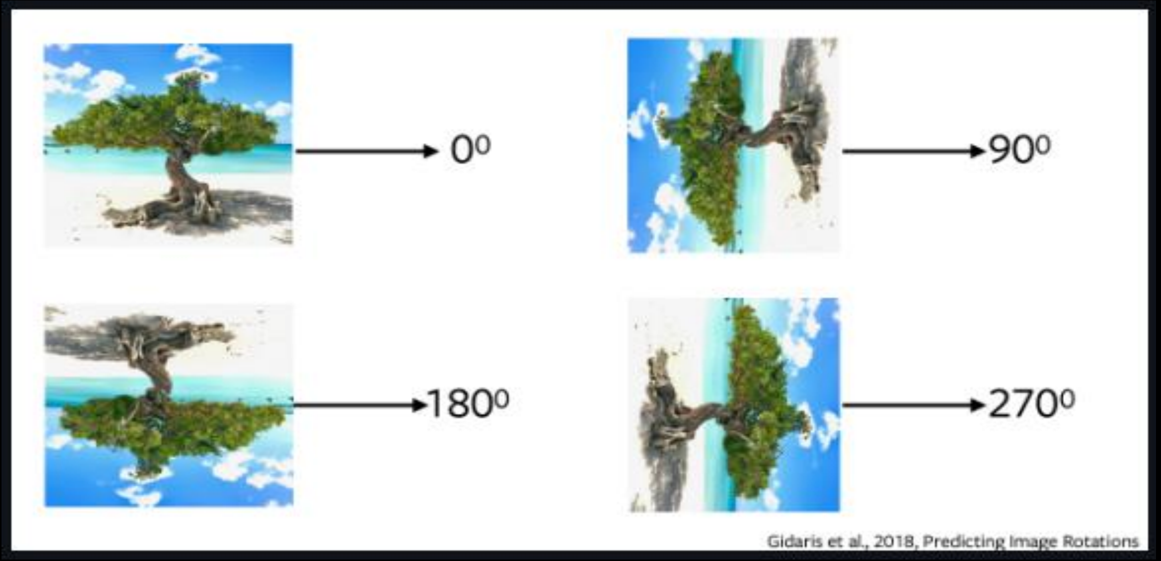


Figure 2. The algorithm receives two patches in one of these eight possible spatial arrangements, without any context, and must then classify which configuration was sampled.

Unsupervised Visual Representation Learning by Context Prediction

Rotation of Images



Simple Experiment on MNIST

Random initialization, freeze everything except FC layers

```
model = Net()

for name, param in model.named_parameters():
    if not name.startswith('fc'):  # Freezing everything except fc layers
        param.requires_grad = False
model = model.to(device)
```

Random initialization, freeze everything except FC layers

```
model = Net()

for name, param in model.named_parameters():
    if not name.startswith('fc'):  # Freezing everything except fc layers
        param.requires_grad = False
model = model.to(device)
```

Accuracy after 1 epoch of training = 95%

Pre-train based on simple rotation

```
for batch_idx, (data, target) in enumerate(train_loader):

    pretextTarget = torch.randint(0, 2, (target.shape))   ##if low and high are 0 ar

    for pretextIter in range(target.shape[0]):
         if pretextTarget[pretextIter]==1:
           thisImage = data[pretextIter,:,:,:]
           thisImageTransposed = torch.transpose(thisImage,1,2)
           data[pretextIter,:,:,:] = thisImageTransposed


    data, pretextTarget = data.to(device), pretextTarget.to(device)
    optimizer.zero_grad()
    output = model(data)
    loss = F.nll_loss(output, pretextTarget)
    loss.backward()
```

Start from this pre-trained model, freeze everything except FC layer

```
model = torch.load('mnistPretrained.pt')
model.fc2 = nn.Linear(128, 10)

for name, param in model.named_parameters():
    if not name.startswith('fc'):  # Freezing everything e
        param.requires_grad = False
model = model.to(device)
```

Start from this pre-trained model, freeze everything except FC layer

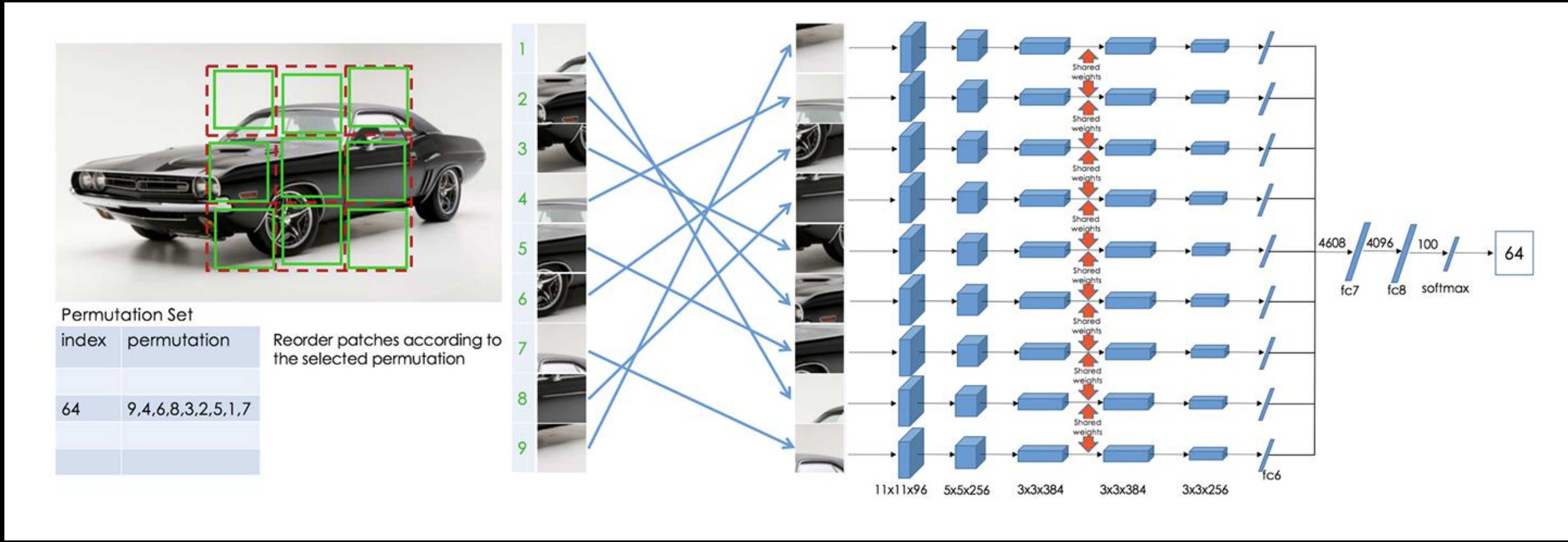
```
model = torch.load('mnistPretrained.pt')
model.fc2 = nn.Linear(128, 10)

for name, param in model.named_parameters():
    if not name.startswith('fc'):  # Freezing everything e
        param.requires_grad = False
model = model.to(device)
```

Accuracy after 1 epoch of training = 97.5%

Image Jigsaw Puzzle

- Patch Extraction: Divide the input image into a grid of patches.
- Shuffling: Randomly permute the patches to create a shuffled version of the image.
- Puzzle Solving Task: The model predicts the original order of the patches from the shuffled input.

Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles

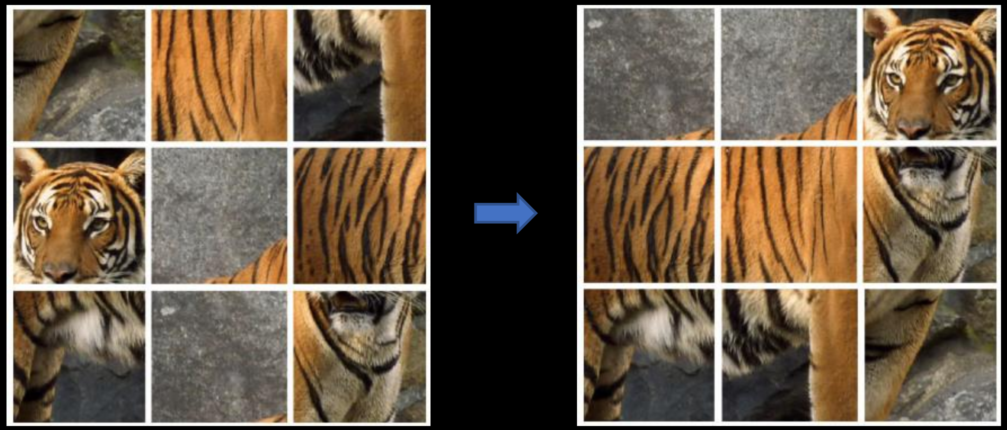


Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles

Figure 1. **Learning image representations by completing damaged jigsaw puzzles.** We sample 3-by-3 patches from an image and create damaged jigsaw puzzles. (a) is the puzzles after shuffling the patches, removing one patch, and decolorizing. We push a network to recover the original arrangement, the missing patch, and the color of the puzzles. (b) shows the outputs; while the pixel-level predictions are in *ab* channels, we visualize with their original L channels for the benefit of the reader.

Learning Image Representations by Completing Damaged Jigsaw Puzzles

Jigsaw Alternative



Pre-Text Task: Rotation / Jigsaw / …

- Rarely used in Earth observation
- Such spatial correlation is less dominant in EO images

Pre-Text Task: Geolocation Classification

- Geolocation metadata is often available

- Cluster the dataset according to lat/long

- Train a model to predict these clusters

Geography-Aware Self-Supervised Learning, 2021

Image Colorization, 2016

- Utilize the CIE Lab color space, where 'L' represents lightness, and 'a' and 'b' represent color channels.

- Input: Grayscale image (L channel).

- Output: Predicted 'a' and 'b' color channels.

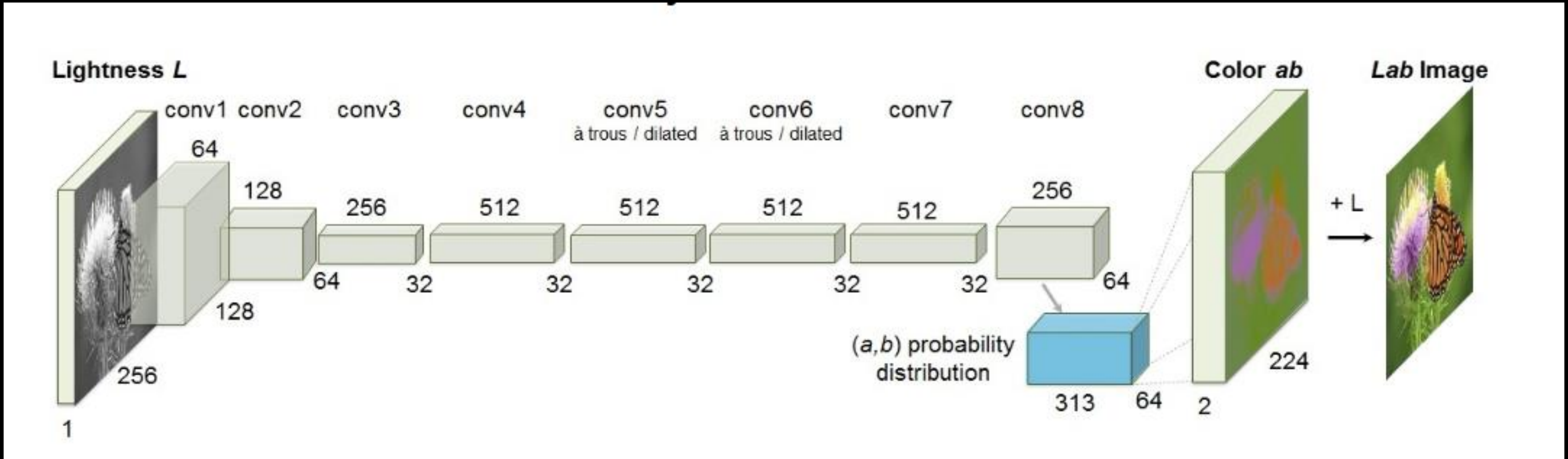


Figure from: https://richzhang.github.io/colorization/

Pre-Text Task Predict Bands

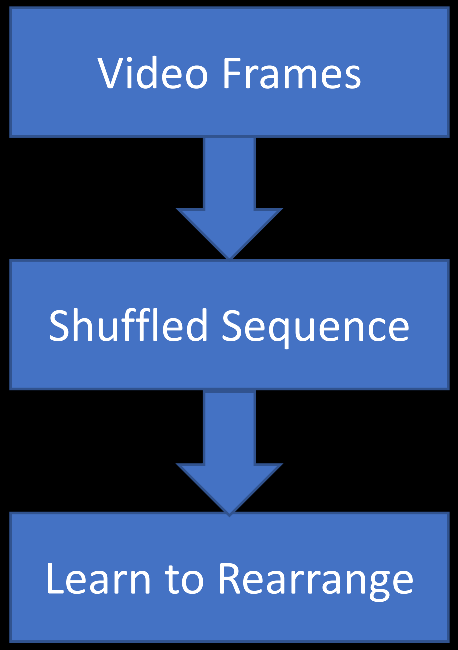
- Drop a band from the image (or alternatively create a synthetic band like NDVI)

- Use rest of bands to predict the missing band

Pre-Text Task: Image Inpainting

- Mask and restore local salient regions
- Also applicable to Earth observation

Pre-Text Task for Video

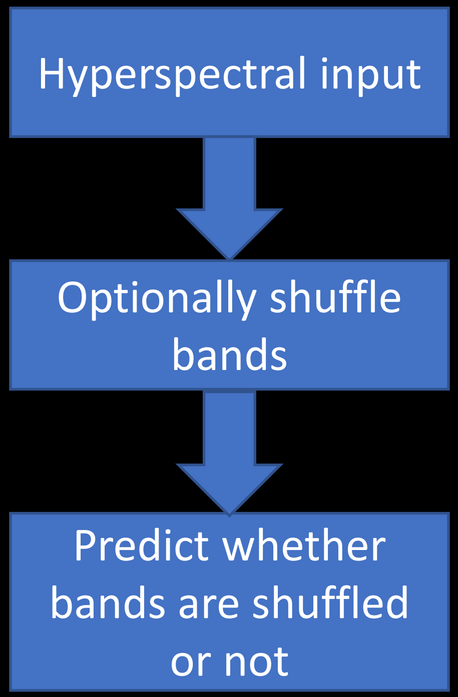


Time-Series: Learning to Reorder

- Learn in unsupervised way from time-series

- Jumble the temporal data in random order and then learn to rearrange

Pre-Text Task for Hyperspectral Images



Contrastive SSL

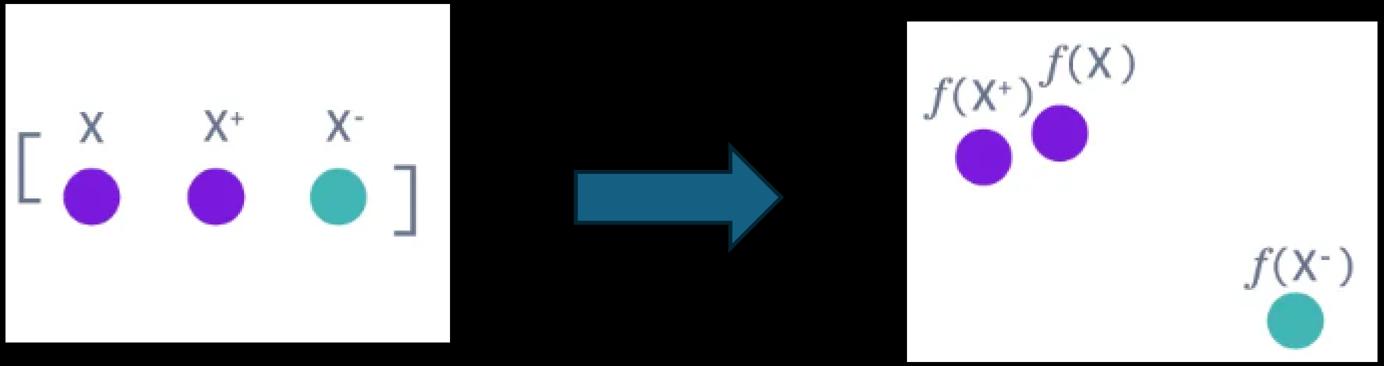
- The goal of contrastive representation learning is to learn such an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart.

- Contrastive learning can be applied in both supervised and unsupervised setting.

Contrastive learning

- Multiple data points simultaneously

Contrastive SSL



Concept of triplet

Related/unrelated

- Sometimes depends on definition

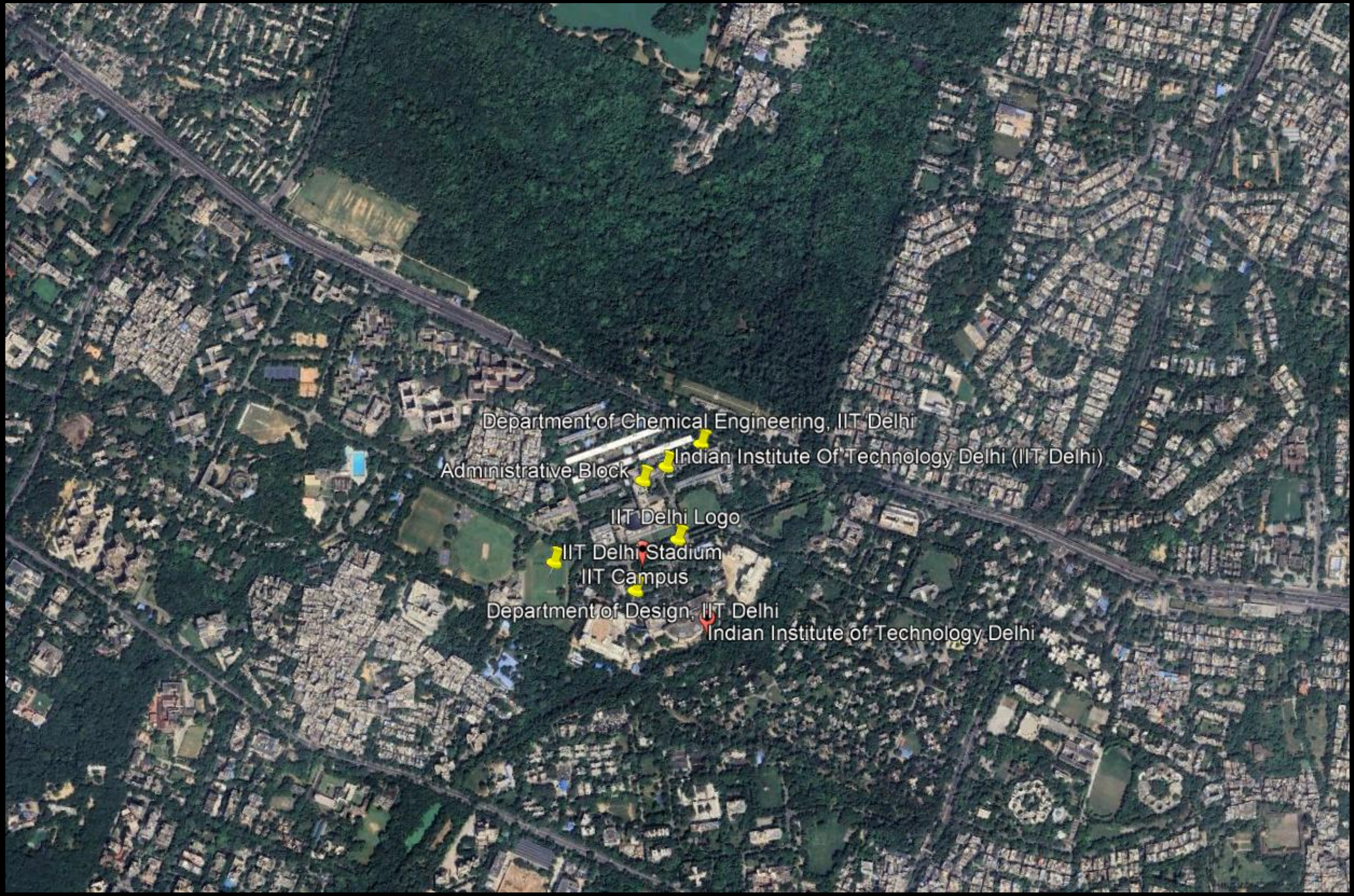
Before diving into the methods, we will discuss for a while about the potential generation mechanisms of positive-negative samples

In image

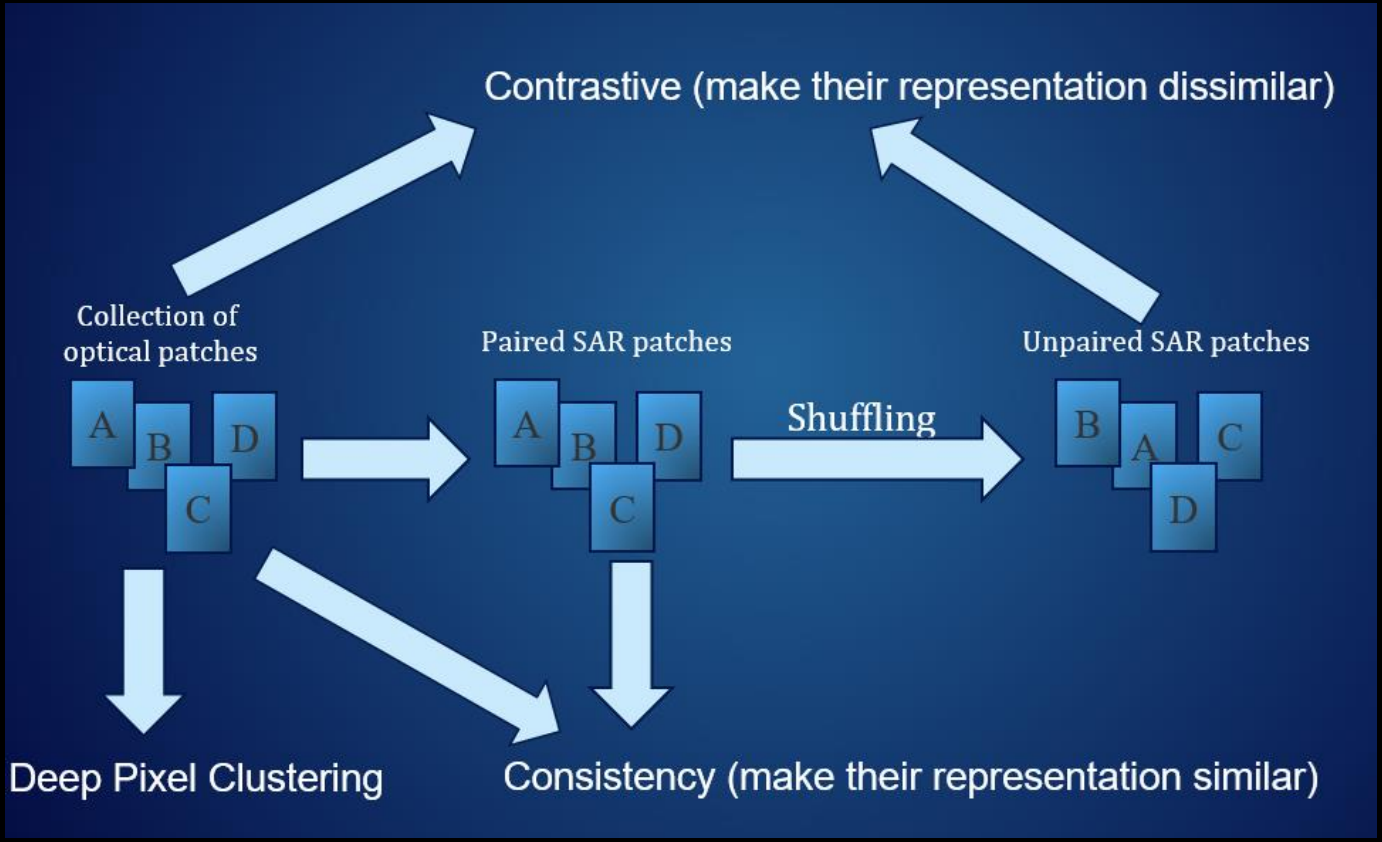
- Patches from an image – similar
- Patches from a different image - dissimilar

First law of geography

- “Everything is related to everything else, but near things are more related than distant things."


Department of Chemical Engineering, IIT Delhi
Indian Institute Of Technology Delhi (IIT Delhi)
Administrative Block
IIT Delhi Logo
IIT Delhi Stadium
IIT Campus
Department of Design, IIT Delhi
Indian Institute of Technology Delhi

Contrastive multi-sensor SSL



Temporal concept

- How to generate positive and negative pairs in unsupervised manner?
- “Temporal” concept
- For a given location, successive images can be treated as positive pairs.

Geography-Aware Self-Supervised Learning, 2021

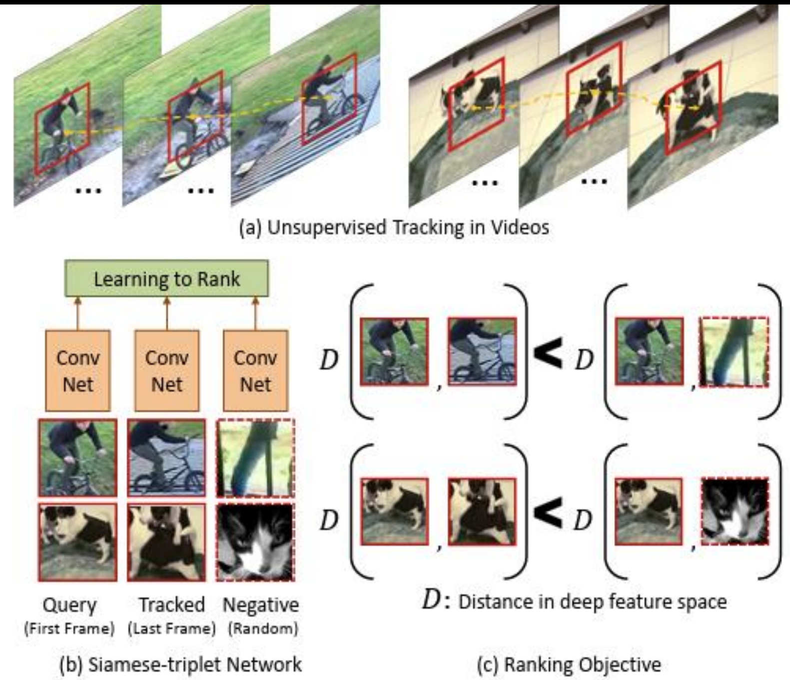


Figure 1. Overview of our approach. (a) Given unlabeled videos, we perform unsupervised tracking on the patches in them. (b) Triplets of patches including query patch in the initial frame of tracking, tracked patch in the last frame, and random patch from other videos are fed into our siamese-triplet network for training. (c) The learning objective: Distance between the query and tracked patch in feature space should be smaller than the distance between query and random patches.

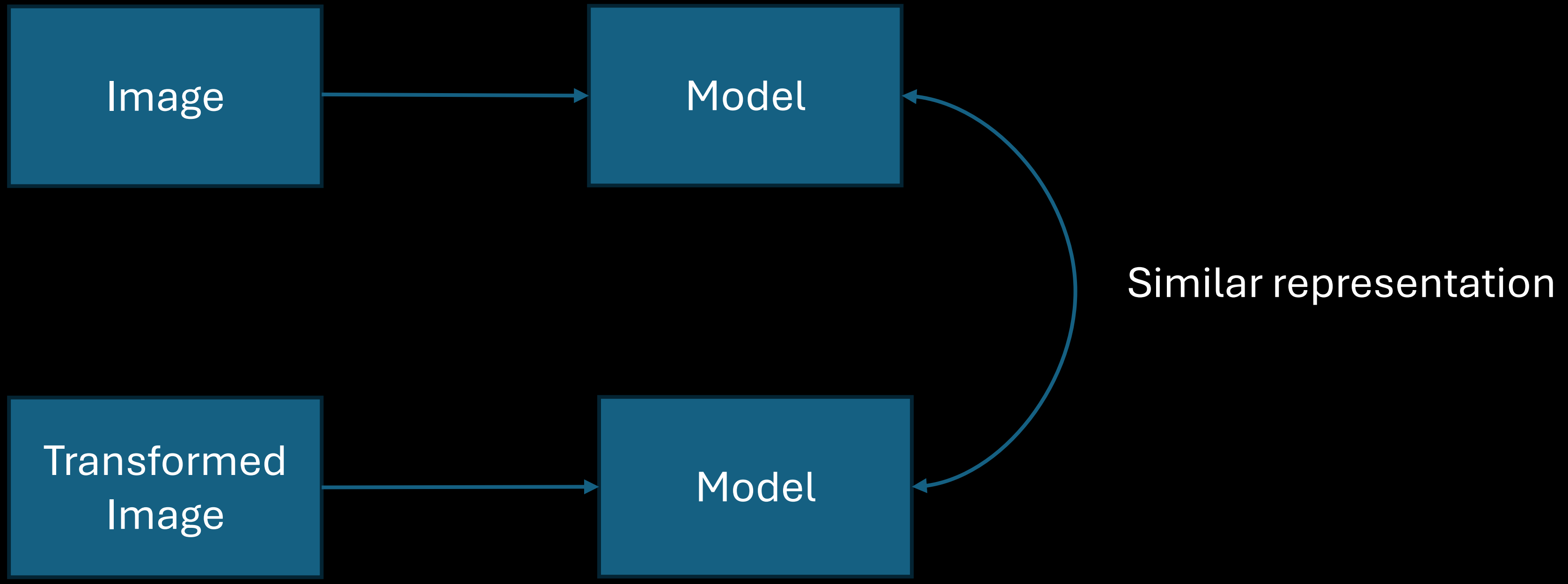
Unsupervised Learning of Visual Representations using Videos, 2015

Now let's come back to the discussing some SSL methods, considering that we have a mechanism of generating/having positive-negative samples.

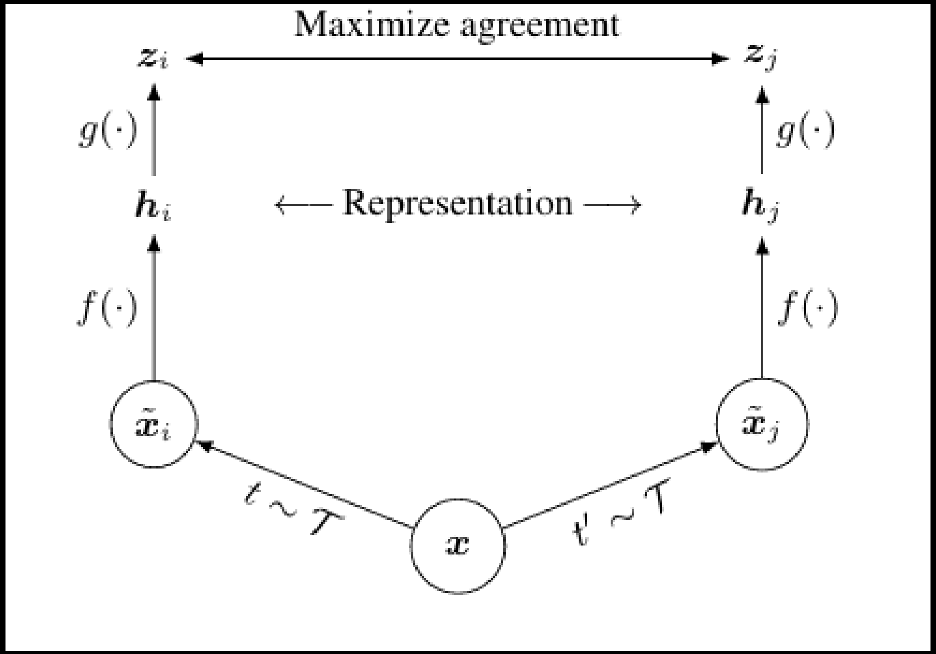
We will discuss the methods slightly not in chronological order.

A common CL framework

Applicable for several methods



SimCLR



A Simple Framework for Contrastive Learning of Visual Representations, 2020

SimCLR

Learning algorithm

Composition of data augmentation

Helps

33.1	33.9	56.3	46.0	39.9	35.0	30.2
32.2	25.6	33.9	40.0	26.5	25.2	22.4
55.8	35.5	18.8	21.0	11.4	16.5	20.8
46.2	40.6	20.9	4.0	9.3	6.2	4.2
38.8	25.8	7.5	7.6	9.8	9.8	9.6
35.1	25.2	16.6	5.8	9.7	2.6	6.7
30.0	22.5	20.7	4.3	9.7	6.5	2.6

Composition of data augmentation

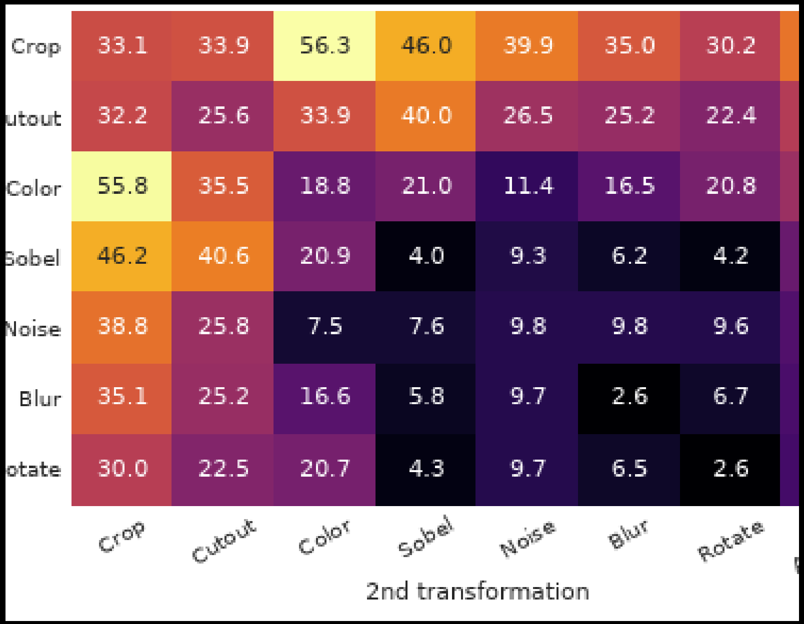


Figure 5. Linear evaluation (ImageNet top-1 accuracy) under individual or composition of data augmentations, applied only to one branch. For all columns but the last, diagonal entries correspond to single transformation, and off-diagonals correspond to composition of two transformations (applied sequentially).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