Momentum Contrast on Imagenette

Shai Kimhi Moshe Kimhi

Jan 2022

1 Introduction

1.1 Momentum Contrast for Unsupervised Visual Representation Learning

Unsupervised representation learning,or MOCO [1], provide a simple and inference efficient framework to better perform Computer Vision tasks with no external labeled data (unsupervised), using contrastive loss between query image and key image, both augmented from unlabeled images.
The Unsupervised representation learning task also called the Upstream Task, when the image classification task also called the Downstream Task.

First, we train the feature extractor part of the network with an embedding head and force the features of augmented images from the same origin to be close in the latent space
Then, we use this features to train a linear classifier that shall outperform a classifier that is pre-trained on the same task.
The algorithm also suggest some improvements such as saving the older key images in a queue and thus to be time efficient when training the latent space representation of the features (the upstream task).

1.2 The Data

The Imagenette Dataset [2] contain 13,394 (9,469 training set and 3,925 test) images from 10 classes.
We measured the mean and standard deviation of the each channel of the dataset, and normalized the images with measured values:
$\mu = [0.4661, 0.4581, 0.4292]$
$\sigma = [0.2382, 0.2315, 0.2394]$.
*Imagenette statistics slightly differ from Image-Net statistics.

1.3 The Model

The common setup for a visual task model is to compose a feature extractor, denote by $f_{extractor}(.)$ and a classifier, we will denote by $f_{class}(.)$. when concatenated, (i.e- we take the features and use as input to the classifier) denote by f and the prediction: $\hat{y} = f(x)$.
The Upstream Task setup is composed from $f_{extractor}(.)$ and Visual encoded projection head $f_{enc}(.)$.

The backbone architecture for this project was ResNet50 from Torchvision lib.The model has 26.5M parameters.
For the contrastive learning task (denote by Upstream task), we used a MLP head of two layers (with non-linearity activation: ReLU) that has 1.12M parameters.
For the Downstream task, we used a single linear layer to the output dimensions with 20,490 parameters.
Note: The original paper [1] used higher dimension for $f_{enc}(.)$. Our implementation considered hardware memory limitations.

1.4 The objective

When training the Upstream Task, we define a query image as q and several key images as k_i were keys that produced from the same original image denote by k_+.
A contrastive loss function (similarity objective function) yields low values when a queue is similar to k_+, and dissimilar to the other keys:

$$l_{contrastive} = -log(\frac{exp(q \cdot k_+)/T}{\sum_{i=0}^{K} exp(q \cdot k_i)}) \tag{1}$$

The classification task (denote by Downstream task) objective function is the Cross-Entropy loss.

2 Upstream Task experiment

2.1 Settings and Parameters

In the contrastive learning task, we yield each time 2 images, one reference image that we call query image q, and a set of key images k_i, and we train the model in order to minimize $l_{contrastive}$ mentioned above.
To better utilize the memory, we used the queue version were each time we calculate $l_{contrastive}$ for k_+ and concatenate the current queue to the new batch of k_+.
For each element in the batch, we apply 2 augmentations both on q and k_+, each augmentations activated with probability 0.6.

We used a pool of 14 different augmentations from Albumentations library [3] listed below.

None	Blur	Gaussian Blur
Channel Shuffle	Coarse Dropout	Cutout
Random Rain	Gaussian Noise	Random change Heu and Saturation
Random Fog	Flip	Color Jitter
Vertical Flip	Random Sharpening	Random change Brighteness and Contrast

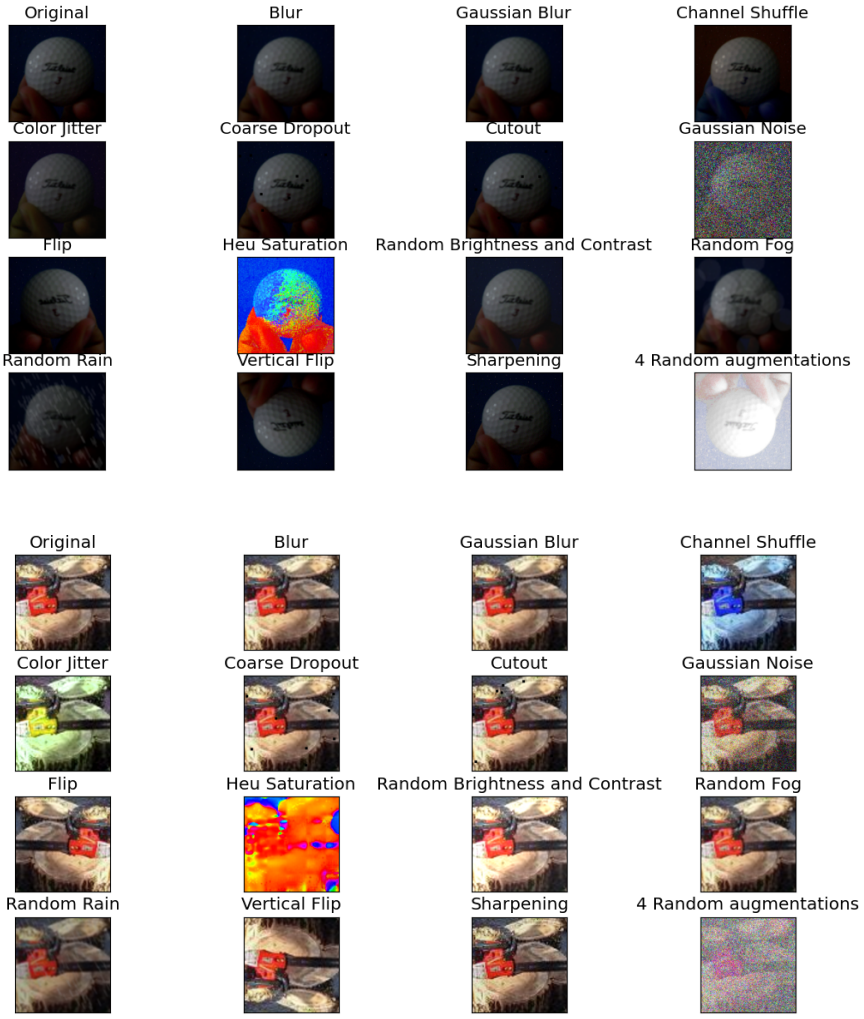

Figure 1: Two examples of activation of the augmentations, together with the original image and the activation of 4 different random augmentations. in the project implementation we used 2 augmentations only for each image and we used also lower threshold for extreme augmentations like Heu Saturation

For the Contrastive training we used Adam optimizer, with Learning rate of 0.001 and Weight decay of 0.0001. The loss temperature set to 0.07 and the Contrastive momentum was 0.999. we've train for 1000 epochs, with 256 size batches and used a Queue of k samples with the size of 4096.

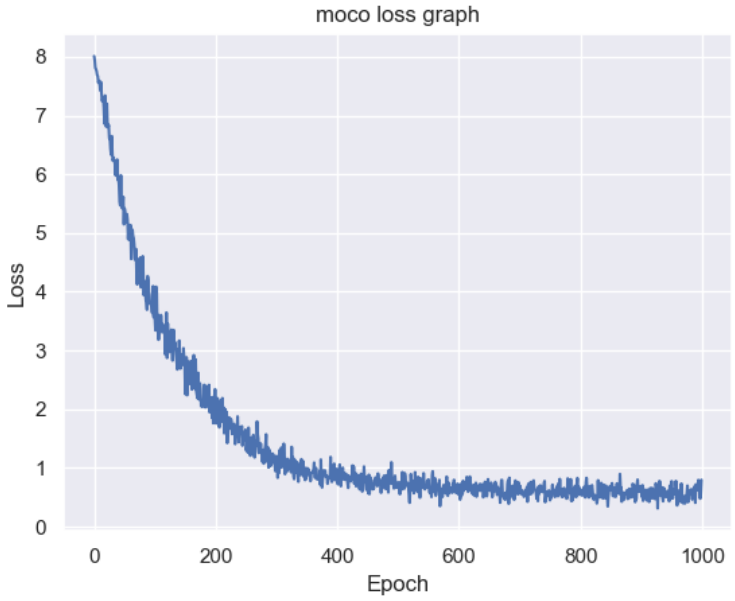

Figure 2: The loss of the contrastive learning task with respect to number of epoch

We obtain a very nice degradation in the loss of about an order of magnitude as can see in Figure2

3 Downstream Task

The Downstream task of image classification, used the $f_{extractor}$ i.e the ResNet50 conv feature extractor that we trained in the last section, and fed it through $f_{classify}$ (i.e a linear layer classifier). note that it's only differ from the ResNet from torchvision hub by the size of the output- since this dataset has only 10 classes. of course that we loaded the weights from the upstream task and fixed it in the same notion of transfer learning, so the feature extractor does not change.
We used Adam optimizer with Learning rate of 0.001, this time with LR scheduler of CosineAnnealingLR and Weight decay of 0.0001. This time we trained with batch size of 32 for 100 epochs.
For each element in the batch, we apply 3 augmentations and train on the augmneted and unaugmented image, each augmentations is activated with probability 0.6.

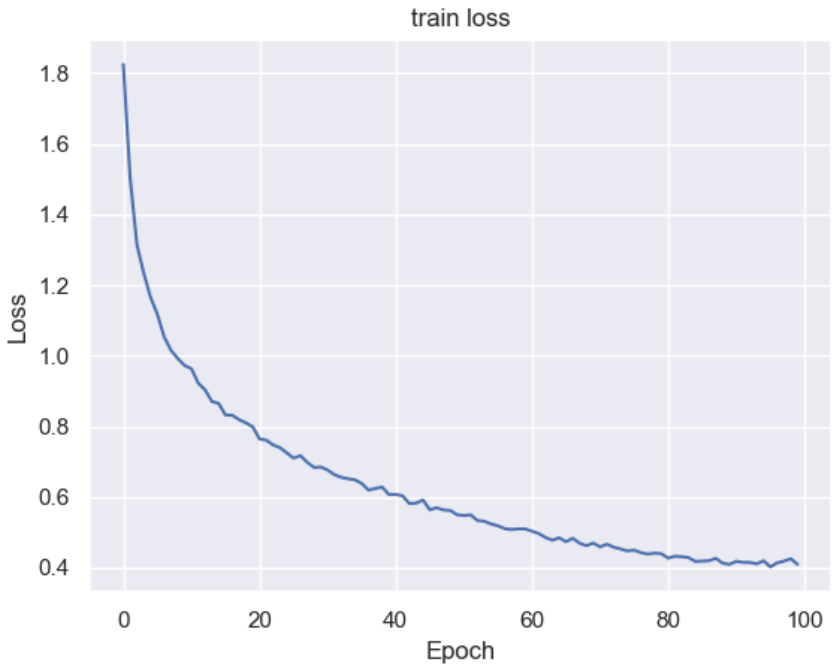

Figure 3: The training loss with respect to number of epoch

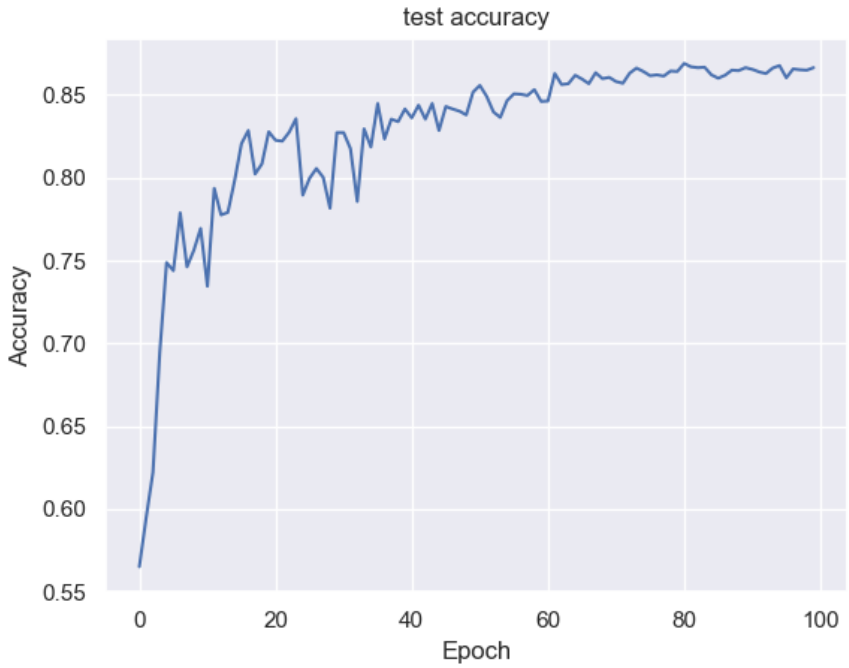

Figure 4: The test accuracy with respect to number of epoch

4 Conclusion

In this project we have explored a new framework in computer vision, Momentum Contrast for Unsupervised Visual Representation Learning [1], that utilize Contrastive loss to out preform the task of image classification, semantic segmentation and object detection in unsupervised regime (no extra labeled data). unfortunately, due to time and Hardware constrains, we experimented only with image classification and did not suggest any improvements for the method.
We believe that future improvement can be achieved when combined with Noisy labels frameworks, and the given dataset provide us the right setup for such experiments in the future.
We've tried to overcome the original paper with new augmentations and training techniques such as CosineAnnealingLR scheduler, yet again due to the Hardware was available, we could not reproduce the same experiments in the original paper.

References

[1] Kaiming He et al. *Momentum Contrast for Unsupervised Visual Representation Learning*. 2020. arXiv: 1911.05722 [cs.CV].

[2] Jeremy Howard. *Imagewang*.

[3] Alexander Buslaev et al. "Albumentations: Fast and Flexible Image Augmentations". In: *Information* 11.2 (2020). ISSN: 2078-2489. DOI: 10.3390/info11020125.