RecViz 2022/2023 - Project - Topic D

Théo Di Piazza (theo.dipiazza@gmail.com)
Ecole normale supérieure Paris-Saclay (MVA)
January 2023

Abstract

In computer vision, it is necessary to have a large amount of labelled data in order to develop supervised learning algorithms. However, the process of labelling this data is costly and time consuming, and can be biased towards the user responsible for the labels. To address this issue, contrast methods are developed using self-supervised learning to generate embeddings for a given image. For this project, we will focus on VICReg, a self-supervised learning method and also on the pre-training of a visual blackbone. The objective of this project will be to take this method in hand, in order to train a self-supervised backbone on the CIFAR-10. Finally, a linear evaluation will be performed to assess the performance of the backbone on the CIFAR-10 test set and CIFAR-100 datasets.

1 VICReg method

Recent self-supervised methods for image representation learning maximize the agreement between embedding vectors produced by encoders fed with different views of the same image. This is why contrastive learning methods have been developed. These methods are based on the importance of data augmentation in predictive tasks and the add of a learnable nonlinear transformation between representation and a contrastive loss. Below is a schematic of the method [1].

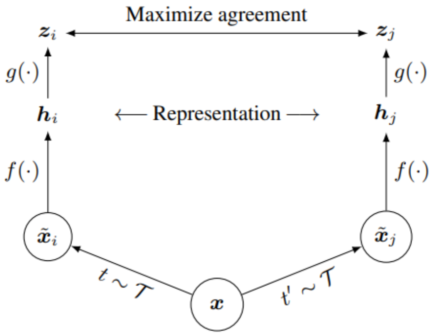


Figure 1: Contrastive Learning of visual representations.

Where : *x is the original image. t and t' are data augmentation operators sampled from a set of operators. $\tilde{x_i}$ and $\tilde{x_j}$ are augmented image. $f(.)$ is a base encoder network which extracts representation vectors from augmented data. h_i and h_j are ouput of the encoder network after the average pooling layer. $g(.)$ a projection head which is a small neural network which maps representations to the space to compute the loss. z_i and z_j are vectors obtained from the projection head. Finally l the contrastive loss, defined for a contrastive prediction task. Given a set $(\tilde{x_k})_k$ of augmented image, whose goal is to identify $\tilde{x_j}$ from $\tilde{x_i}$.*

However, a collapse can happen: it's when the encoders $f(.)$ produce constant or non-informative vectors. That's why VICReg method is introduced [2], method based on the contrastive learning method, with the addition of two regularisation terms in the loss function: one that maintains the variance of each encodding vector above a threshold and a second that decorrelates each pair of variables. Morever, these terms stabilise the training of other methods and lead to better performance. Below is a schematic of the VICReg method.

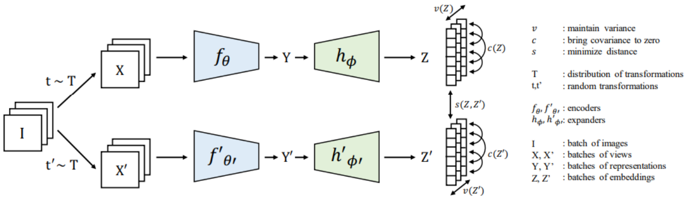

$t \sim$ T
$t' \sim$ T
I
X
X'
f_θ
$f'_{\theta'}$
Y
Y'
h_ϕ
$h'_{\phi'}$
Z
Z'
$v(Z)$
$c(Z)$
$s(Z,Z')$
$c(Z')$
$v(Z')$
v : maintain variance
c : bring covariance to zero
s : minimize distance
T : distribution of transformations
t,t' : random transformations
$f_\theta, f'_{\theta'}$: encoders
$h_\phi, h'_{\phi'}$: expanders
I : batch of images
X, X' : batches of views
Y, Y' : batches of representations
Z, Z' : batches of embeddings

Figure 2: VICReg method

The difference with the previous method is in the loss function. First of all, distance between 2 embeddings from the same image is minimized. Moreover, covariance between pairs of embedding variable over a batch are attracted to 0 which decorrelate variables, and the variance of each embedding variable over a batch is maintained above a threshold.

The loss (between Z and Z') function is defined by:

$$l(Z,Z') = \lambda s(Z,Z') + \mu[v(Z)+v(Z')] + \nu[c(Z)+c(Z')]$$

Where:

$$c(z) = \sum_{i \neq j}[C(Z)]^2_{i,j} \ ; \ s(Z,Z') = \frac{1}{n}||z_i - z'_i||_2^2$$

$$C(Z) = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \hat{z})(z_i - \hat{z})^T$$

$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} max(0, \gamma - \sqrt{Var(z_j) + \epsilon})$$

1.1 Getting started with the method

In order to get a feel for the method, the representative loss is calculated for 3 cases: one case between 2 different transformations of the same image, case with 2 identical images and case with 2 different images. It is computed with the full model associated to the ResNet-50 backbone available on the github [3]. The images are from ImageNet [5].

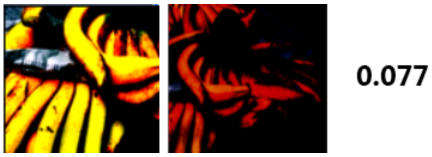


Figure 3: 2 different transformations from an image (banana, ImageNet) and the representative loss.

Figure 4: Identical images (banana, ImageNet) and the representative loss.

Figure 5: Different images (banana and animal, ImageNet) and the representative loss.

It can be seen logically that: 2 identical images have a loss associated with the quality of the representation that is zero (0.000, Figure 4), 2 augmented images resulting from the data augmentations of an image have a non-zero loss with low value (0.077, Figure 3) and 2 different images have a significant non-zero loss with high value (3.242, Figure 5).

2 Pretrain backbone

From now on, the objective is to use the presented method to pretrain a backbone on a different dataset. Here, the encoder used will be a ResNet-18. It is an encoder with fewer parameters than ResNet-50, so it will be faster to train and we'll still benefit from the architecture to tackle the vanishing gradient problem [6]. The dataset used is CIFAR-10 [4]. CIFAR-10 consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. The model will be pretrained on 40000 images from the train set.
Here examples of images from CIFAR-10 dataset:

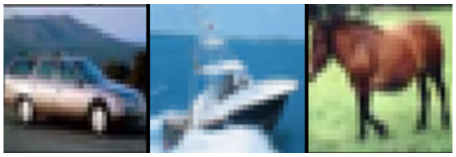


Figure 6: Samples from CIFAR-10: car, boat, horse.

CIFAR-10 was used in order to use a different dataset than the paper (ImageNet). CIFAR-10 contains 60,000 images for 10 classes, while ImageNet contains more than one million images for 100 classes. In our results, we expect to have a less efficient pretraining model than the paper one, because the encoder used (ResNet-18) has a shallower architecture than the paper one (ResNet-50); and the dataset used (CIFAR-10) contains fewer images than the paper dataset (ImageNet). Two backbones will be pre-trained: a first backbone with a ResNet-18 and the same transformation parameters as the VICReg paper, then a second backbone with a ResNet-18 and different transformation parameters, more adapted to the CIFAR-10 dataset. The details are given in the next section.

2.1 Data Augmentation

Data Augmentation is a key element of the method studied, as two augmented images from the same image are compared. The quality of the augmented images therefore has a direct impact on model learning. First, I trained my first ResNet-18 backbone from the data augmentations of the paper, which were adapted to the ImageNet dataset. However, the results were not satisfactory enough because the data augmentation operations were not adapted to the CIFAR-10 dataset.
The CIFAR-10 images have a lower resolution than the ImageNet images. When the transformations used for the VICReg paper (adapted to ImageNet) are used on the CIFAR-10 images, the augmented images can be difficult to interpret by the human eye.
The first step is to adjust the normalization coefficients to the CIFAR-10 dataset with the following values : $mean : (0.4914, 0.4822, 0.4465) - std : (0.247, 0.243, 0.261)$. In order to adapt to the CIFAR-10 dataset, the different transformations will be used : Random Resized Crop with scale constrait (0.4 to 1.0), horizontal flip, modifications on the brightness

(0.2 instead of 0.4), contrast (0.2 instead of 0.4), saturation, hue (0.025 instead of 0.1), grayscale, normalization, solarization. The data augmentation operator that had the biggest impact on the performance of the model was the addition of a scaling constraint on the random resize crop, and secondly the modification of the parameters on the colours. The application probabilities of these transformations were not modified.. Below are examples of transformations (Figure 7).

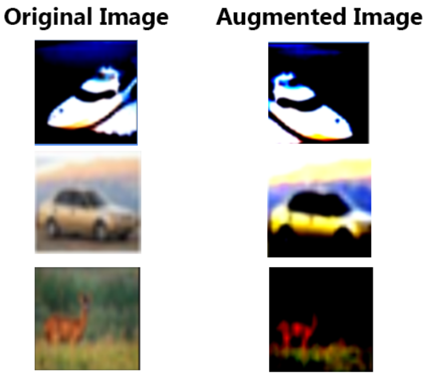


Figure 7: Samples Augmented images from CIFAR-10.

2.2 Pre-Training details

Different optimizers and different learning rates were tried for pre-training the backbone. Adam was tested with different learning rates (10^{-2} to 10^{-5}). LARS (base lr: 0.2, weight decay: 10^{-6}, eta: 0.001 and momentum: 0.9) was also tested. All configurations were tested over 20 epochs, and every 4 epochs a linear evaluation was performed in order to see wether it was learning good visual representations, or not. Then, the best configuration is retained for the final training. Both Adam optimizer with lr: 10^{-4} and LARS optimizer result in a stable loss function that decreases with epochs, and an increase in accuracy in the linear evaluation, epoch after epoch. For the other configurations, the model does not seem to learn. In the end, the chosen optimizer is Adam with a learning rate of 10^{-4} for the first 30 epochs, then 10^{-5} for the following epochs.

3 Linear Evaluation

In this section, a focus on the linear evaluation of pre-trained models is made. This consists of training a linear classifier on top of frozen representations learned by self-supervised methods. Below is a diagram of the method presented (Figure 8).

3.1 ResNet-18 backbones on CIFAR-10

The ResNet-18 pretrained backbone is tested on the CIFAR-10 test dataset. The model is trained on 15k images and will be tested on 5k images. To the frozen backbone is added a non-frozen linear layer (512 to 10 neurons). Training parameters: 20 epochs, SVM Optimizer, lr: 0.3, weight decay: 10^{-6} and momentum: 0.9.

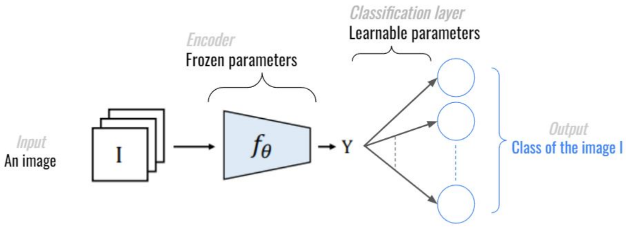


Figure 8: Scheme of the linear evaluation protocol.

On validation for the best backbone (the one with transformations adapted to CIFAR-10 images), Top-1-Acc = 80.2% and Top-5-Acc = 99.1%.

3.2 ResNet-18 backbones on CIFAR-100

The ResNet-18 pretrained backbone is tested on the CIFAR-100 dataset. CIFAR-100 consists of 60000 32x32 colour images in 100 classes, with 6000 images per class. The model will be trained on 45000 images from the train set and will be test on 15000 images from the test set.

Here examples of images from CIFAR-100 dataset:

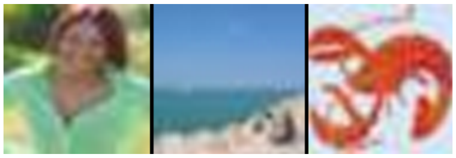


Figure 9: Samples from CIFAR-100: women, sea, lobster.

To the frozen backbone is added a non-frozen linear layer (512 to 100 neurons). Training parameters: 20 epochs, SVM Optimizer, lr: 0.3, weight decay: 10^{-6} and momentum: 0.9. n validation for the best backbone (the one with transformations adapted to CIFAR-10 images), Top-1-Acc = 45.2% and Top-5-Acc = 74.9%.

3.3 VICReg ResNet50 on CIFAR-10

The ResNet50 backbone pretrain of the VICReg paper model is tested on the CIFAR-10 dataset. The model will be trained on 15,000 images of the training set and will be tested on 5,000 images of the test set.

To the frozen backbone is added a non-frozen linear layer (2048 to 10 neurons). Training parameters: 10 epochs, SVM Optimizer, lr: 0.3, weight decay: 10^{-6} and momentum: 0.9. On validation, Top-1-Acc = 82.2% and Top-5-Acc = 98.8% .

3.4 VICReg ResNet50 on CIFAR-100

The ResNet50 pretrain backbone of the VICReg paper model is tested on the CIFAR-100 dataset. Training parameters: 10 epochs, SVM Optimizer, lr: 0.3, weight decay: 10^{-6} and momentum: 0.9.
On validation, Top-1-Acc = 54.6% and Top-5-Acc = 81.9%.

3.5 Comparison of results

Below are the results of the linear evaluation of the different models on the test sample.

Metrics on CIFAR-10		
Model	Top-1 accuracy	Top-5 accuracy
ResNet-18 ∅	29.3%	79.8%
ResNet-18 - IN	59.9% ↑ +30.6%	93.9% ↑ +14.1%
ResNet-18 - C10	80.2% ↑ +52.0%	**99.1%** ↑ +23.0%
ResNet-50	**82.2%** ↑ +52.9%	98.8% ↑ +19.0%

Table 1: Metrics of Linear Evaluation on CIFAR-10.

Metrics on CIFAR-100		
Model	Top-1 accuracy	Top-5 accuracy
ResNet-18 ∅	11.7%	25.4%
ResNet-18 - IN	28.2% ↑ +16.5%	54.8% ↑ +29.4%
ResNet-18 - C10	45.2% ↑ +33.5%	74.9% ↑ +42.9%
ResNet-50	**54.6%** ↑ +49.5%	**81.9%** ↑ +56.5%

Table 2: Metrics of Linear Evaluation on CIFAR-100.

*Remark: **ResNet-18 ∅** is the baseline : backbone without pretraining. **ResNet-18 - IN** is the backbone pretrained with the transformations adapted for **ImageNet**. **ResNet-18 - C10** is the backbone pretrained with the transformations adapted for **CIFAR-10**.*

For the linear evaluation on CIFAR-10: it can be seen that all 3 backbones perform better than the baseline. Moreover, the ResNet-18 - C10 backbone (with transformations adapted to CIFAR-10) has a much better accuracy than ResNet-18 - IN (with transformations not adapted to CIFAR-10), which shows the importance of working with data augmentation operators adapted to the dataset. A bad management of these operators can lead to a bad learning of the visual representations. Finally, the ResNet-50 backbone presents the best Top1-Accuracy. This is consistent, as it has been trained on a large dataset, and its more complex architecture allows it to learn more relevant visual representations. However, it is important to note that even though the datasets used for pre-training and for linear evaluation are not the same, the images in the CIFAR-10 test set are of the same nature as the images in the CIFAR-10 train set. Thus even if the performances are very close (80.2% & 82.2%), it is expected that the ResNet-50 backbone performs better than my backbones, on new data.

Concerning the evaluation on CIFAR-100 the observations and comparisons of backbones are the same. We notice a drop in performance in terms of Top-1/5 Accuracy, which is expected, as CIFAR-100 is a more complex dataset than CIFAR-10, with 100 classes instead of 10. It is therefore more difficult for models to learn more general visual representations when there are 100 classes, than on 10.

Conclusion and Perspectives

For this project, I was able to discover a supervised learning method: VICReg. After learning the method, I was able to pretrain 2 backbones from ResNet-18 on CIFAR-10: a first backbone with transformations not adapted to the training dataset, and a second backbone with transformations adapted to the training dataset. Thus, the choice of data augmentation operators has a strong impact on the qualities of the visual representations learned by the backbone. Furthermore, I was able to compare these backbones to a baseline, as well as to the backbone proposed in the paper, and obtain similar results on the CIFAR-10 dataset. Constrained by the technical resources (GPU, memory) available, I did not have the opportunity to use ResNet-50, nor to benefit from a larger batch size.

Regarding the limitations of using such methods, I would say that they require large datasets of qualities and complex encoding architectures, in order to be really effective.

Finally, the results presented in the paper allow us to be optimistic about the use of such methods. It allows to learn a good representation of the data that can be used for downstream tasks such as classification or regression. In the future they could, for example, be used for image labelling.

References

[1] *A Simple Framework for Contrastive Learning of Visual Representations, Chen et al, ICML 2020*
[2] *VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning, Bardes et al, ICLR 2022*
[3] *VICReg Code : https://github.com/facebookresearch/vicreg*
[4] *Learning multiple layers of features from tiny images. Technical Report, University of Toronto, Krizhevsky, A. (2009).*
[5] *Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., Fei-Fei, L. (2009).*
[6] *Deep Residual Learning for Image Recognition, Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun (2015)*

Appendix

A.1. More details about the architecture that I used to pretrain backbones.

Below is an illustration of the architecture used to train my own backbone on the CIFAR-10 dataset.

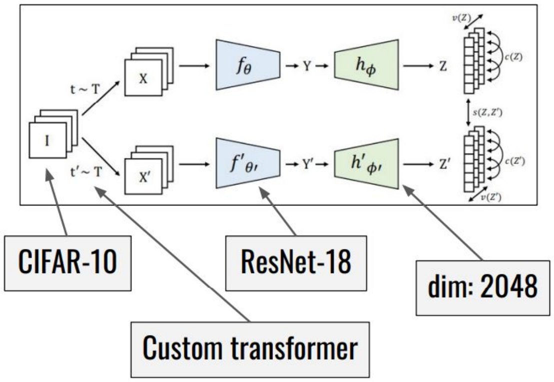


Figure 1: Proposed architecture to pretrain backbones.

A.2. Evolution of the cumulative loss during training.

Below is the evolution of the cumulative loss for the first 20 epochs during the pretraining of the 2 backbones. Even if the loss decreases very smoothly, this is not enough to say that the backbone learns quality visual representations. At each iteration, different transformations are used on the images, so new representations have to be learned. For this, the linear evaluation protocol must be used.

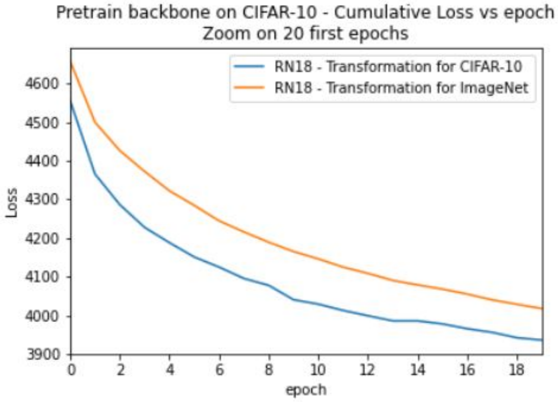


Figure 2: Cumulative Loss vs epoch.

B. Evolution of the Top1/5-Accuracy during linear evaluation.

Below, the evolution of the Top1/5-Accuracy during the linear evaluation of the backbones on CIFAR10 and CIFAR100, after pretraining.

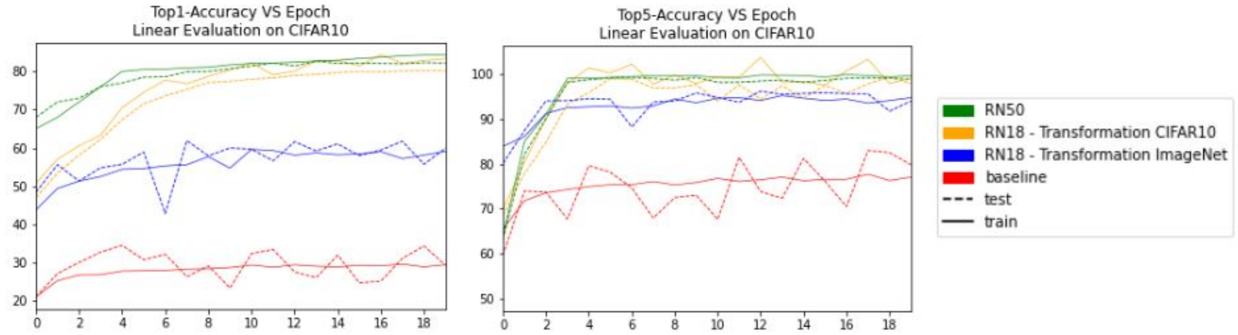


Figure 3: Top 15 Accuracy vs epoch - Linear evaluation on CIFAR-10.

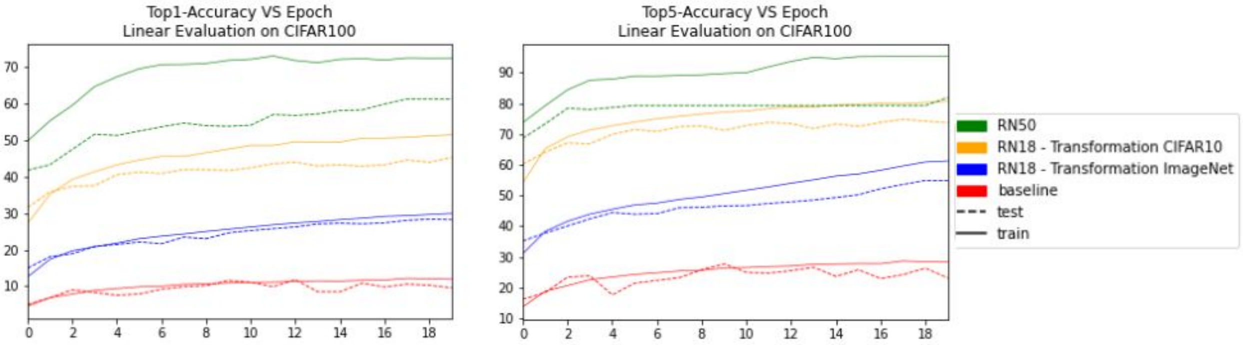
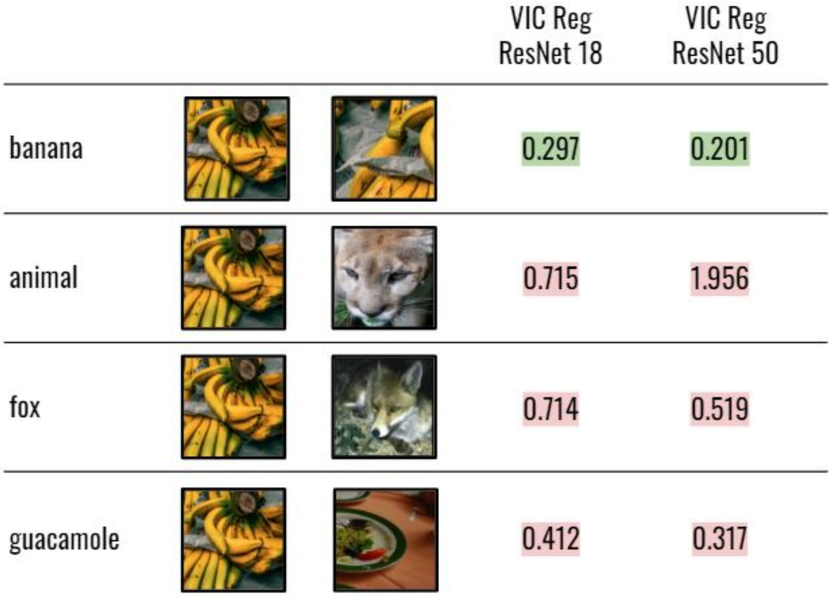


Figure 4: Top 15 Accuracy vs epoch - Linear evaluation on CIFAR-10.

C. Calculation of the representative loss for different images.

In order to propose qualitative results for a self-supervised learning method, I propose to compute the loss representation (of the ViCReg method) between an initial given image and 4 images different from the initial one. From the full model of the paper, and from the full model associated with the ResNet-18 backbone which presents the best performances. Please note that these results take into account the head projection, and are therefore not fully representative of the extraction quality of the visual representations of the backbones.



			VIC Reg ResNet 18	VIC Reg ResNet 50
banana			0.297	0.201
animal			0.715	1.956
fox			0.714	0.519
guacamole			0.412	0.317

Figure 5: 4 examples of representative loss obtained with my ResNet18 and ResNet50 from the paper.

It can be seen that for both models, the loss representation is lowest when calculated between 2 images of the same class (banana), which is consistent.