# Mutual Information Measure for Image Segmentation Using Few Labels

Eduardo H. Sanchez[1,2](✉), Mathieu Serrurier[1,2], and Mathias Ortner[3]

[1] IRT Saint Exupéry, Toulouse, France
{eduardo.sanchez, mathieu.serrurier}@irt-saintexupery.com
[2] IRIT, Université Toulouse III - Paul Sabatier, Toulouse, France
[3] Airbus, Toulouse, France
mathias.ortner@airbus.com

**Abstract.** Recently several models have been developed to reduce the annotation effort which is required to perform semantic segmentation. Instead of learning from pixel-level annotations, these models learn from cheaper annotations, e.g. image-level labels, scribbles or bounding boxes. However, most of these models cannot easily be adapted to new annotations e.g. new classes since it requires retraining the model. In this paper, we propose a similarity measure between pixels based on a mutual information objective to determine whether these pixels belong to the same class. The mutual information objective is learned in a fully unsupervised manner while the annotations (e.g. points or scribbles) are only used during test time. For a given image, the unlabeled pixels are classified by computing their nearest-neighbors in terms of mutual information from the set of labeled pixels. Experimental results are reported on the Potsdam dataset and Sentinel-2 data is used to provide a real world use case where a large amount of unlabeled satellite images is available but only a few pixels can be labeled. On the Potsdam dataset, our model achieves 70.22% mIoU and 87.17% accuracy outperforming the state-of-the-art weakly-supervised methods.

**Keywords:** Mutual information maximization · Weakly supervised learning · Similarity measure · Image Segmentation · Satellite datasets.

## 1 Introduction

Most of the successful models for semantic segmentation rely on a supervised learning approach [17]. Even though these models achieve remarkable results, the effort of collecting carefully annotated data to train these models make them impractical to use in many contexts. Generally, these models require a training dataset composed of images with pixel-level annotations, e.g. a class label is assigned to every pixel in the image. The task of labeling images is very time-consuming, e.g. the reported time to segment a single image from the PASCAL VOC 2012 dataset is around 240 seconds [2]. Consider the particular case of satellite data, many missions have been launched to observe the Earth producing massive amounts of satellite images which are absolutely impossible

to be annotated by human operators. For example, each of the Sentinel-2 mission satellites [6] provides up to 1.6TB of images per day.

Different methods have been developed to reduce the need of carefully pixel-level annotations for large-scale data analysis. These methods propose a weakly supervised approach for semantic segmentation where the required annotations are less tedious to obtain than pixel-level annotations such as image-level annotations [13], points [2], scribbles [16] or bounding boxes [14]. These annotations are included during the training stage for learning semantic segmentation models. As a consequence, these models are not easily adaptable to new annotations (e.g. to refine the segmentation results or add new class labels) since retraining the models using these new annotations is required. For this purpose, few-shot learning techniques for semantic segmentation have been proposed [25] but it still requires a significant number of labeled samples from seen classes to perform well on unseen classes. Additionally, these methods often produce suboptimal results without providing the user with an interactive way to make corrections without the need to retrain the model.

Recent work has focused on mutual information estimation and maximization to perform representation learning in an unsupervised manner [3, 9, 19, 20]. The main goal of these unsupervised approaches is to capture the most salient attributes of data to perform downstream tasks using the learned representations. Extensions of the previous models have been proposed using a self-supervised approach in order to capture the shared attributes from multiple views of a common context [1, 22, 24]. We think that designing a self-supervised task to learn suitable representations for semantic segmentation is an appealing idea. In particular, our work is inspired by these models [1, 9, 22] to learn a similarity measure without supervision.

In this work, we take a step forward and propose a model that performs semantic segmentation by computing the similarity between pixels based on a mutual information approach without requiring annotations during training. Using an ideal similarity measure as distance metric, pixels belonging to the same class are close while simultaneously distant from pixels belonging to other classes. A very few pixel-level annotations are only used during test time. Our model computes the mutual information similarity between labeled pixels and unlabeled pixels and then performs a per-pixel nearest-neighbor search from the set of labeled pixels to classify the unlabeled pixels.

Our model provides several advantages. First, there is no need to retrain our model when new annotations are included since the similarity measure is learned using an unsupervised learning approach. Second, our model requires a small amount of annotated data which can be acquired in multiple formats e.g. points, scribbles, bounding boxes. Third, we propose a simple neural network architecture that achieves competitive semantic segmentation results while keeping a reasonable processing time.

The following contributions are made in this paper:

- We propose a model that combines a similarity measure based on mutual information between pixels using self-supervised techniques [1, 9, 22] and a nearest-neighbor search to perform semantic segmentation.
- We show that excellent results can be achieved by labeling less than 0.75% of the total number of pixels in an image.
- We present quantitative results for image segmentation on the Potsdam dataset [12] outperforming the state-of-the-art weakly-supervised methods and qualitative results on Sentinel-2 data [6] to show a real world use case.
- We analyze the impact of using multiple views via data augmentation techniques [1] on the segmentation performance and we perform an ablation study to evaluate the contribution of each element of the model.

## 2 Related work

**Image segmentation** Exceptional results have been achieved by fully supervised models on semantic segmentation [17]. To reduce the annotation effort required by supervised learning settings, several methods have been proposed which use cheaper annotations e.g. points [2], scribbles [16], image annotations[13] or bounding boxes [14]. Labels provided by points or scribbles are then propagated to unlabeled pixels during training [2, 16]. The main drawback is that these models are not easy to adapt to new annotations for refining the segmentation results or adding new class labels as it requires retraining the whole model. GrabCut [21] performs interactive image segmentation using a bounding box to separate foreground and background. On the other hand, Khoreva et al. [14] propose a semantic segmentation method requiring a costly recursive training where bounding boxes are refined iteratively. Recent work has been presented [25] to segment classes containing few labels in the dataset. However, this method still requires many training examples from the known classes to perform well on the unknown classes.

**Self-supervised learning** In contrast to the prevalent paradigm based on generative or reconstructive models, recent work has been focused on mutual information maximization for representation learning. These models maximize the mutual information between an input and its representation. Mutual information is computed using different estimators based on the Kullback–Leibler [3], Jensen-Shannon [9], Wasserstein [20] divergences or noise-contrastive estimation[19]. Interesting extensions of these mutual information based frameworks have been presented to capture the common attributes from paired images [1, 22, 24]. Learning representations that capture the most significant attributes of an image from multiple views is useful for semantic segmentation.

**Deep metric learning** Measuring the similarity between pixels is a useful tool for image segmentation since similar pixels under a given criterion belong to the same class while dissimilar pixels belong to different classes. Generally,

raw pixels are mapped to a representation space by a deep neural network and then similarity between pixels is computed in the representation domain [5, 8, 23]. For instance, Sun et al. [23] propose a neural diffusion distance to perform segmentation. However, it requires labeled data during training to be consistent with a human criterion. For video segmentation, Chen et al. [5] propose a metric based on the triplet loss [4] which is trained in a supervised manner.

In this paper, we propose a model that performs image segmentation in a weakly-supervised setting. The segmentation procedure is split into two stages. First, the model learns a mapping function from the pixel domain to a representation domain which captures relevant attributes for image segmentation using a mutual information based framework that combine the approaches [1, 22]. After training the mapping function, we use a mutual information objective to measure the similarity between pixels. In contrast to the models [5, 8, 23], the similarity measure is learned in a completely unsupervised manner. Second, our model computes the mutual information similarity between labeled pixels provided by an operator and unlabeled pixels and then performs a per-pixel nearest-neighbor search from the set of labeled pixels to propagate the labels to unlabeled pixels. The labeled pixels are only used during test time instead of training time like the models [2, 13, 14, 16].

## 3 Background

### 3.1 Mutual information

The mutual information between two random variables $X \in \mathcal{X}$ and $Z \in \mathcal{Z}$ is defined in Equation 1 where $p(x, z)$ is the joint probability density function of $X$ and $Z$ while $p(x)$ and $p(z)$ are the marginal probability density functions of $X$ and $Z$, respectively.

$$I(X, Z) = \int_{\mathcal{X}} \int_{\mathcal{Z}} p(x, z) \log \left( \frac{p(x, z)}{p(x)p(z)} \right) dxdz \tag{1}$$

It is straightforward to see that $I(X, Z)$ is defined as the Kullback-Leibler divergence between the joint probability distribution $\mathbb{P}_{XZ}$ and the product of the marginal distributions $\mathbb{P}_X\mathbb{P}_Z$, i.e. $I(X, Z)$=$\mathrm{D}_{KL}\left(\mathbb{P}_{XZ} \parallel \mathbb{P}_X\mathbb{P}_Z\right)$. Generally, computing the mutual information between high dimensional variables is a difficult task since the distributions $\mathbb{P}_{XZ}$ and $\mathbb{P}_X\mathbb{P}_Z$ are unknown. Thus, some methods based on deep neural networks have recently been proposed [3, 9, 19, 20].

### 3.2 Representation learning

Equation 1 can be used as objective for unsupervised learning where $X$ is a variable corresponding to a given input (image, speech, text, etc) and $Z$ is the representation of $X$. The representation $Z$ is extracted by an encoder function defined by a deep neural network of parameters $\psi$, $E_\psi : \mathcal{X} \to \mathcal{Z}$, i.e. $Z = E_\psi(X)$.

The Deep InfoMax framework [9] proposes a mutual information estimator $\hat{I}$(X,Z) based on the Jensen-Shannon divergence instead of the Kullback-Leibler divergence, i.e. $I^{(\mathrm{JSD})}(X,Z) = \mathrm{D}_{JS}\left(\mathbb{P}_{XZ} \parallel \mathbb{P}_X\mathbb{P}_Z\right)$.

Intuitively, let $X_i$ and $X_j$ be two observations of $X$. Let $Z_i$ and $Z_j$ be the representations of $X_i$ and $X_j$ respectively extracted via $E_\psi$. Therefore, $(X_i, Z_i)$ is an input-representation pair sampled from the joint probability density function $p(x,z)$ while $(X_i, Z_j)$ is an input-representation pair sampled from the product of the marginal probability density functions $p(x)p(z)$.

We define a discriminator function defined by a deep neural network of parameters $\rho$, $D_\rho : \mathcal{X} \times \mathcal{Z} \to [0,1]$ which represents the probability of a sample $(X,Z)$ coming from $p(x,z)$ instead of $p(x)p(z)$, i.e. the probability that $Z$ is the representation of $X$. The discriminator $D_\rho$ and the encoder $E_\psi$ are trained to assign a high probability to samples from $p(x,z)$ (close to 1) and a low probability to samples from $p(x)p(z)$ (close to 0) as shown in Equation 2.

$$\max_{E_\psi, D_\rho} \hat{I}(X,Z) = \mathbb{E}_{p(x,z)}\left[\log D_\rho(X,Z)\right] + \mathbb{E}_{p(x)p(z)}\left[\log\left(1 - D_\rho(X,Z)\right)\right] \quad (2)$$

By redefining the discriminator function [18] $D_\rho(X,Z) = \frac{e^{-T_\theta(X,Z)}}{1+e^{-T_\theta(X,Z)}}$ where $T_\theta : \mathcal{X}\times\mathcal{Z} \to \mathbb{R}$ is called the statistics network, we obtain the mutual information objective proposed by the Deep InfoMax framework [9] in Equation 3.

$$\max_{E_\psi, T_\theta} \hat{I}(X,Z) = \mathbb{E}_{p(x,z)}\left[-\log\left(1+e^{-T_\theta(X,Z)}\right)\right] - \mathbb{E}_{p(x)p(z)}\left[\log\left(1+e^{T_\theta(X,Z)}\right)\right] \quad (3)$$

Two mutual information objectives are proposed in the Deep InfoMax framework. Maximizing the mutual information between an input $X$ and a representation $Z$ is called *global mutual information*, i.e. $\mathbf{L}^{\mathrm{global}}_{\theta,\psi}(X,Z) = \hat{I}(X,Z)$. Additionally, maximizing the mutual information between patches of the image $X$ represented by a feature map $C_\psi(X)$ of the encoder $E_\psi$ and a feature representation $Z$ is called *local mutual information* i.e. $\mathbf{L}^{\mathrm{local}}_{\phi,\psi}(X,Z) = \sum_i \hat{I}(C^{(i)}_\psi(X), Z)$.

## 4 Method

In this paper, we propose a model that combines the mutual information based methods [1, 22] to learn a suitable representation domain to measure the similarity between pixels. Our model is trained in a fully unsupervised manner by leveraging large amounts of unlabeled data. Sanchez et al. [22] extends the Deep InfoMax framework to separate the common information and the exclusive information for paired images. Bachman et al. [1] use the Deep InfoMax framework to perform self-supervised representation learning by maximizing the mutual information between representations extracted from multiple views of a shared context, e.g. the context is provided by an image and the multiple views are generated via data augmentation techniques. Learning the common information between images [1, 22] provides a way to compute how similar these images are. In Section 4.1, we present the mutual information objective to learn the similarity measure and we explain how to use this similarity measure to perform image segmentation in Section 4.2.

### 4.1 Shared mutual information

To create a suitable representation domain for image segmentation, we propose to capture the common information between images of the same context (e.g. satellite images from the same forest) into a shared representation. By removing the particular information of each image, we create a representation that distills the class information which is useful for image segmentation. We propose to learn this shared representation by using the principle presented in [1, 22]. Let $X$ and $Y$ be two images of the same context and let $S_X$ and $S_Y$ be the respective shared representations extracted by an encoder $E_\psi$. In order to enforce learning only the common information between images $X$ and $Y$, the methods [1, 22] maximizes the mutual information between the image $X$ and the representation $S_Y$ and similarly, between the image $Y$ and the representation $S_X$. In order to create pairs of images of the same context, we follow the approach of Bachman et al. [1] and we use data augmentation techniques (rotation, flip, pixel shift, color jitter) to create a second image from a given image, i.e. $X = f(Y)$ where $f$ is a data augmentation function. We use the objective function proposed by Sanchez et al. [22] since it is simpler to optimize. Equations 4 and 5 displays the global and local mutual information maximization objectives.

$$\mathbf{L}_{MI}^{\text{global}}(X, Y) = \mathbf{L}_{\theta,\psi}^{\text{global}}(X, S_Y) + \mathbf{L}_{\theta,\psi}^{\text{global}}(Y, S_X) \tag{4}$$

$$\mathbf{L}_{MI}^{\text{local}}(X, Y) = \mathbf{L}_{\phi,\psi}^{\text{local}}(X, S_Y) + \mathbf{L}_{\phi,\psi}^{\text{local}}(Y, S_X) \tag{5}$$

Sanchez et al. [22] also includes a $L_1$ constraint to force the shared representations to be identical as shown in Equation 6. The final objective function is displayed in Equation 7, where $\alpha$, $\beta$ and $\gamma$ are constant coefficient.

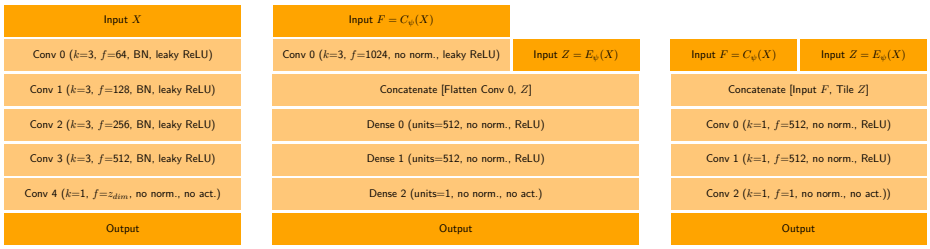
$$\mathbf{L}_1(X, Y) = \mathbb{E}_{p(s_x, s_y)} \left[ |S_X - S_Y| \right] \tag{6}$$

$$\max_{\psi,\theta,\phi} \mathcal{L}^{\text{shared}} = \alpha \mathbf{L}_{MI}^{\text{global}}(X, Y) + \beta \mathbf{L}_{MI}^{\text{local}}(X, Y) - \gamma \mathbf{L}_1(X, Y) \tag{7}$$

### 4.2 Mutual information as similarity measure

Similarly to Chen et al. [5], we perform per-pixel retrieval to find the closest pixel from the reference pixel set using the learned representations. A k-nearest-neighbors approach is used to determine the class of unlabeled pixels by propagating the information from labeled pixels. A common way of computing the distance between pixels is to measure the $L_1$ or $L_2$ distance between their corresponding representations [5]. Alternatively, we propose to use the global and local mutual information objectives introduced in Section 3.2.

During training, the mutual information objective is computed using an image $X$ and a different view of $X$ generated via data augmentation techniques, i.e. $Y = f(X)$. In contrast, during test time the mutual information objective is computed using two different images. Let $X_i$ and $X_j$ be two image patches centered at the pixels $i$ and $j$ respectively and let $S_{X_i}$ and $S_{X_j}$ be the shared




a) Encoder network $E_\psi$ b) Global statistics network $T_\theta$ c) Local statistics network $T_\phi$

**Fig. 1.** Network architecture. The encoder and statistics networks are implemented using convolutional and dense layers defined by the number of units, $k$: kernel size, $f$: feature maps, BN: batch normalization [11] and activation function. The statistics networks and the encoder share weights: the input $F$ of the statistics network is the output of the Conv 2 layer of the encoder, $C_\psi(X)$.

representations provided by the encoder function. The similarity between pixels $i$ and $j$ is measured by computing $\mathbf{L}^{\text{global}}_{\theta,\psi}(X, S_X)$ and $\mathbf{L}^{\text{local}}_{\phi,\psi}(X, S_X)$.

After training, our model is capable to predict whether a shared representation $S_{X_i}$ corresponds to the image $X_i$. Since the shared representation $S_{X_i}$ contains the class information of $X_i$, it provides a means to identify pixels belonging to the same class of $X_i$. For example, consider that $X_i$ and $X_j$ are two different images (e.g. a satellite image from an urban area and another from an agricultural area), the mutual information objective $\mathbf{L}^{\text{global}}_{\theta,\psi}(X, S_X)$ (or $\mathbf{L}^{\text{local}}_{\phi,\psi}(X, S_X)$) achieves a high score since it is easy to distinguish both images. On the other hand, suppose that $X_i$ and $X_j$ are two similar images (e.g. satellite images from the same forest), the mutual information objective $\mathbf{L}^{\text{global}}_{\theta,\psi}(X, S_X)$ (or $\mathbf{L}^{\text{local}}_{\phi,\psi}(X, S_X)$) achieves a low score since it is hard to distinguish the images.

### 4.3 Implementation details

Our model is composed of three deep neural networks: the encoder $E_\psi$, the global statistics network $T_\theta$ and the local statistics network $T_\phi$. The architecture details are provided in Figure 1. Every network is trained from scratch by using randomly initialized weights. To optimize the objective $\mathcal{L}^{shared}$ defined in Equation 7, we use the Adam optimizer with a learning rate of 0.0001, $\beta_1$=0.9 and $\beta_2$=0.999. We use a batch size of 512 images. Images pairs are created by applying data augmentation techniques (flip, rotation, pixel shift, color jitter). Our baseline model use the following coefficients to weight the terms of the objective function $\mathcal{L}^{shared}$: $\alpha$=0.5, $\beta$=1.0 and $\gamma$=0.1. The size of the shared representation is $z_{dim}$=10. The training algorithm was executed on a NVIDIA Tesla K80 GPU. The training and image segmentation procedures are summarized in Algorithms 1 and 2. More details are provided in the additional material section.

**Algorithm 1** Training algorithm.

Random initialization of model parameters $\psi^{(0)}, \theta^{(0)}, \phi^{(0)}$.
**for** $k = 1$; $k = k + 1$; $k <$ number of iterations **do**
  Sample a batch of $C$ image patches $\{X_1, \ldots, X_C\}$. Image patches have a size $s$.
  Create a new view of $X_i$ via a data augmentation technique $Y_i = f(X_i)$.
  Create a batch of $C$ paired images $\mathbf{X} : \{(X_1, Y_1), ..., (X_C, Y_C)\}$.
  Create a batch of $C$ unpaired images $\tilde{\mathbf{X}}$ by shuffling the $Y$ dimension of $\mathbf{X}$.
  Compute $\mathcal{L}^{(k)} = \mathcal{L}^{\text{shared}}(\mathbf{X}, \tilde{\mathbf{X}}, \psi^{(k)}, \theta^{(k)}, \phi^{(k)})$:

$$\begin{aligned}\mathcal{L}^{(k)} = \alpha \big[ &- \textstyle\sum_{\mathbf{X}} \text{sp}\left(-T_\theta(C_\psi(X_i), E_\psi(Y_i))\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\theta(C_\psi(X_i), E_\psi(Y_i))\right) \\ &- \textstyle\sum_{\mathbf{X}} \text{sp}\left(-T_\theta(C_\psi(Y_i), E_\psi(X_i))\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\theta(C_\psi(Y_i), E_\psi(X_i))\right) \big] + \beta \textstyle\sum_j \big[ \\ &- \textstyle\sum_{\mathbf{X}} \text{sp}\left(-T_\phi^{(j)}(C_\psi(X_i), E_\psi(Y_i))\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\phi^{(j)}(C_\psi(X_i), E_\psi(Y_i))\right) \\ &- \textstyle\sum_{\mathbf{X}} \text{sp}\left(-T_\phi^{(j)}(C_\psi(Y_i), E_\psi(X_i))\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\phi^{(j)}(C_\psi(Y_i), E_\psi(X_i))\right) \big] \\ &- \gamma \textstyle\sum_{\mathbf{X}} \left(|E_\psi(X_i) - E_\psi(Y_i)|\right)\end{aligned}$$

  where the softplus function is defined by $\text{sp}(x) = (1 + e^x)$

  Update the parameters $\psi^{(k+1)}, \theta^{(k+1)}$ and $\phi^{(k+1)}$ by gradient ascent of $\mathcal{L}^{(k)}$.
**end for**

**Algorithm 2** Image segmentation algorithm.

Select an image $X_t$ from the dataset. Image $X_t$ has a size $t \gg s$.
Label a set of $L$ pixels into $N$ classes $P = \{(p_1, c_1), \ldots, (p_L, c_L)\}$ from $X_t$.
where $p_i$ defines the coordinates and $c_i$ is the class of the $i$-th labeled pixel.
**for** unlabeled pixel at $q_j \in X_t$ **do**
  **for** labeled pixel at $p_i \in P$ **do**
    Select the image patches $X_j$ and $X_i$ of size $s$ centered at $q_j$ and $p_i$.
    Extract the representations $S_{X_i} = E_\psi(X_i)$ and $S_{X_j} = E_\psi(X_j)$.
    Extract the feature maps $C_{X_i} = C_\psi(X_i)$ and $C_{X_j} = C_\psi(X_j)$.
    Create the image-representation sets $\mathbf{X} = \{(C_{X_i}, S_{X_i}), (C_{X_j}, S_{X_j})\}$
    and $\tilde{\mathbf{X}} = \{(C_{X_i}, S_{X_j}), (C_{X_j}, S_{X_i})\}$
    Compute the global/local mutual information between $X_j$ and $X_i$:
    $\mathbf{D}_i = \mathbf{L}_{\theta,\psi}^{\text{global}} = -\sum_{\mathbf{X}} \text{sp}\left(-T_\theta(C_{X_k}, S_{X_k})\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\theta(C_{X_k}, S_{X_k})\right)$ or
    $\mathbf{D}_i = \mathbf{L}_{\phi,\psi}^{\text{local}} = \sum_j \left[-\sum_{\mathbf{X}} \text{sp}\left(-T_\phi^{(j)}(C_{X_k}, S_{X_k})\right) - \sum_{\tilde{\mathbf{X}}} \text{sp}\left(T_\phi^{(j)}(C_{X_k}, S_{X_k})\right)\right]$
  **end for**
  Assign the pixel $q_j$ the class $c_{i^*}$ of the nearest pixel $i^* = \arg\min_i \{\mathbf{D}_i\}_{i=1}^L$.
**end for**

## 5 Experiments

### 5.1 Datasets

**Potsdam** The Potsdam dataset [10] contains 8550 aerial images of the city of Potsdam. Each image has a size of $t$=200 × 200 pixels and is composed of four channels: red, green, blue and infrared (RGBI). The dataset is split into three

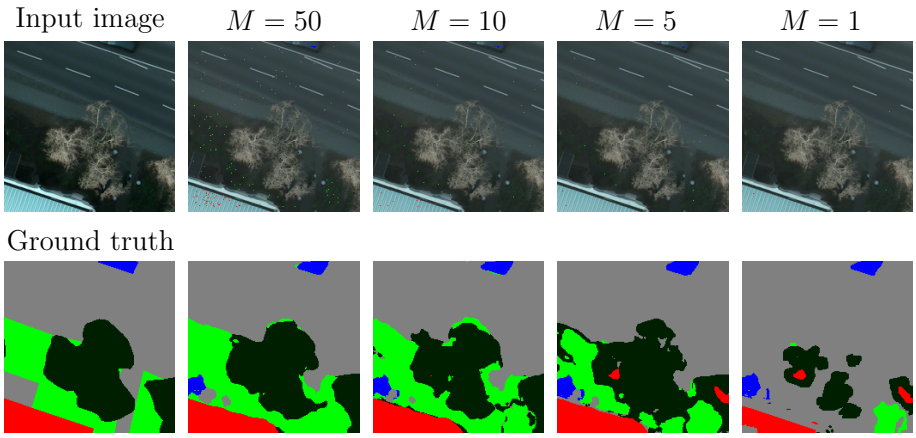



**Fig. 2.** Image segmentation examples. During test time, only $M$ points per class randomly sampled from the ground truth are used to perform image segmentation. As $M$ increases the accuracy and mIoU are improved. Best viewed in color and zoom-in.

parts: 3150 unlabeled images, 4545 training labeled images and 855 test labeled images. Images are labeled into 6 classes (road, car, vegetation, tree, building and clutter). Similarly to [12], we also perform image segmentation using a 3-label version by merging classes (road and car, vegetation and tree and building and clutter). Image patches of size $s$=$13 \times 13$ pixels are randomly sampled from the unlabeled images to optimize the model objective (Equation 7) and the test labeled images to report the experimental results. We use this dataset to provide quantitative results and comparisons to other models.

**Sentinel-2** We collected 100GB of Sentinel-2 time series [6] by selecting several regions of interest on the Earth's surface. Images are acquired at 13 spectral bands using different spatial resolutions. We use the RGBI bands which correspond to bands at 10m spatial resolution. Our dataset is composed of 4200 time series of 12 images acquired at different dates between 2016 and 2018. The size of each image is $t$=$512 \times 512$ pixels. Image patches of size $s$=$9 \times 9$ pixels are randomly sampled from these images. In addition to data augmentation techniques, the function $f$ creates an image pair by selecting an image patch $Y$ from the same location of $X$ but on a different date. Since there are no labels available, we use this dataset to provide qualitative results in a real world use case where a huge amount of unlabeled data is available and a few annotated pixels are provided by a human operator. Data can be downloaded from the Sentinel Hub [7]. More dataset construction details are provided in the additional material section.

| Mutual information | Metric | $N = 6$ | | | | $N = 3$ | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $M = 1$ | $M = 5$ | $M = 10$ | $M = 50$ | $M = 1$ | $M = 5$ | $M = 10$ | $M = 50$ |
| Global | Accuracy | 0.4576 | 0.6366 | 0.7147 | 0.8517 | 0.5310 | 0.6626 | 0.7362 | 0.8843 |
| | mIoU | 0.2793 | 0.4598 | 0.5354 | 0.6777 | 0.3333 | 0.4888 | 0.5691 | 0.7407 |
| Local | Accuracy | 0.5013 | 0.6894 | 0.7670 | 0.8717 | 0.5397 | 0.7274 | 0.8045 | 0.9163 |
| | mIoU | 0.3332 | 0.5085 | 0.5818 | 0.7022 | 0.3632 | 0.5589 | 0.6415 | 0.7866 |

**Table 1.** Segmentation results. Accuracy and mIoU for $N$ classes, $M$ points per class and $z_{dim}$=10 using the global/local mutual information in the Potsdam dataset.

### 5.2 Image segmentation on Potsdam

**Global and local mutual information** We train our model as described in Section 4.3 using the unlabeled images of the Potsdam dataset. Image segmentation is performed on test images where $M$ pixels per class are known. Typically, these annotated pixels are provided by a human operator. To simplify the evaluation, annotated pixels are simulated by randomly sampling $M$ pixels per class from the ground truth. We use several values of $M \in \{1, 5, 10, 50\}$ to evaluate the performance on image segmentation. An example of the impact of $M$ on the segmentation results is shown in Figure 2. By using the learned mutual information based similarity measure, nearest neighbor search is applied to classify pixels into one of $N \in \{3, 6\}$ classes. The performance is reported in terms of mean intersection over union (mIoU) and accuracy. To measure the pixel similarity, we use either the global mutual information objective or the local mutual information objective (see Algorithm 2). Results are reported in Table 1. As expected, the performance is improved as $M$ increases. Our experiments suggest that using the local mutual information objective achieves a better performance than the global mutual information when a few pixels are annotated while the performance is similar when a larger amount of annotated pixels is provided ($M = 50$). Many segmentation examples are shown in Figure 3.

**Model comparison** To provide a comparison, we perform image segmentation using different similarity measures to search the nearest neighbor of the unlabeled pixels. First, we compute the nearest neighbor using the $L_1$ distance between raw pixels. Secondly, we use the $L_1$ distance between the representations extracted from the VAE model [15], Deep InfoMax model [9] and our model. Similar images do not necessarily have to be close in the representation domain in terms of the $L_1$ distance. Therefore, a low performance is expected at image segmentation using the $L_1$ distance between representations. Finally, we use the mutual information objective of Deep InfoMax [9]. As Deep InfoMax representations keeps all the image information, i.e. more than just class information, we expect this representation to be less appropriate for image segmentation. Table 2 displays the segmentation results. As shown, the local mutual information objective outperforms the other similarity measures for image segmentation. Segmentation examples are shown in Figure 4.

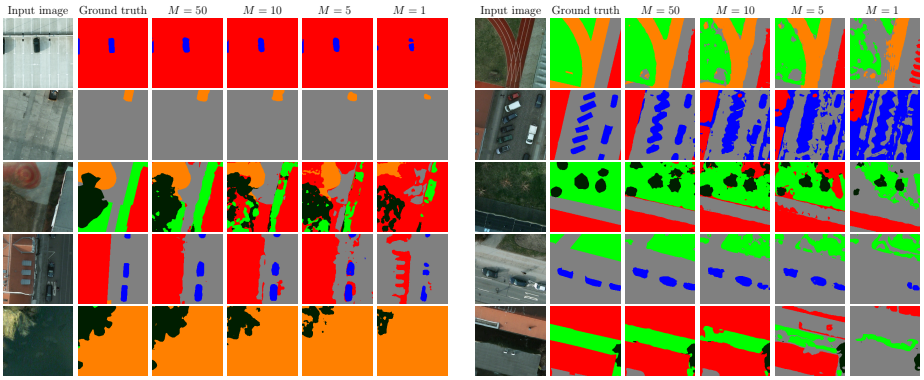



**Fig. 3.** Image segmentation examples using $M$ points per class randomly sampled from the ground truth. The image segmentation performance is improved as $M$ increases.

| Model | $N = 6$ | | $N = 3$ | |
|---|---|---|---|---|
| | Accuracy | mIoU | Accuracy | mIoU |
| Raw pixels (L1) | 0.6073 | 0.3962 | 0.7267 | 0.5337 |
| VAE (L1) | 0.5844 | 0.3826 | 0.7045 | 0.5230 |
| DIM (L1) | 0.4063 | 0.2103 | 0.4754 | 0.2887 |
| Ours (L1) | 0.6498 | 0.4570 | 0.7391 | 0.5685 |
| DIM | 0.5973 | 0.4114 | 0.6497 | 0.4649 |
| **Ours** | **0.8717** | **0.7022** | **0.9163** | **0.7866** |

**Table 2.** Model comparison in terms of accuracy and mIoU for $N$ classes, $M = 50$ and $z_{dim} = 10$ using the local mutual information in the Potsdam dataset.

**Ablation study** We analyze two important factors in our model: the influence of data augmentation techniques to generate multiple views (pixel shift, color jitter, image flip and image rotation) and the importance of some model components, e.g. the statistics networks. Results are displayed in Table 3. Several conclusions can be drawn from our experiments. First, the model architecture can be simplified since the global statistics network can be removed ($\alpha = 0$) without modifying the performance on image segmentation. The local statistics network plays the most important role during training as pointed out by Bachman et al. [1]. Second, removing the $L_1$ distance between shared representations ($\gamma = 0$) leads to a slightly reduction in the performance. Third, when the shared representations are not swapped in Equations 4 and 5 (no SSR) the performance drastically decreases since these representation contains more information than the class information required for image segmentation. Concerning the data augmentation techniques, we surprisingly notice that the performance remains the same by individually removing the color jitter, image rotation and image flip. We believe that the effect of the color jitter is ignored since it is an attribute which is not captured in the shared representation. Additionally, the impact of removing the image rotation or image flip is minimal due to the local information

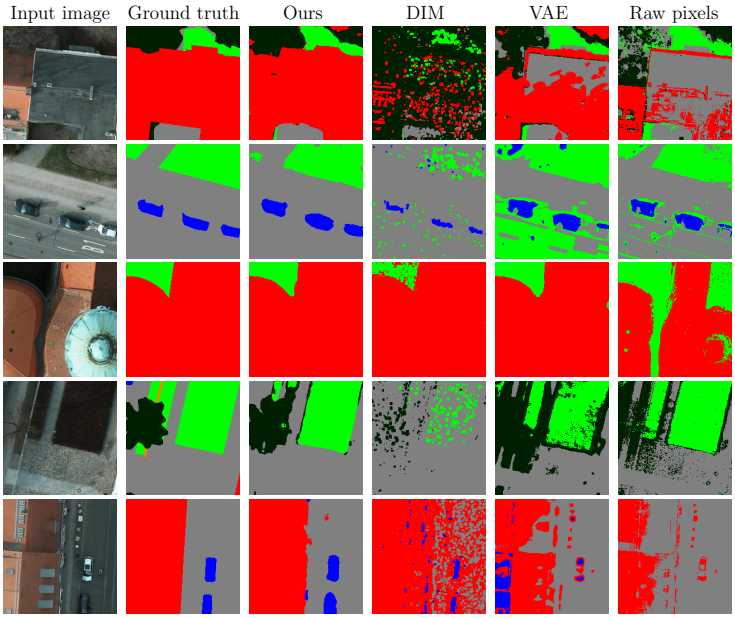



**Fig. 4.** Image segmentation model comparison. Our model produce the closest predictions to the ground truth using $N$=6, $M$=50 and $z_{dim}$=10 in the Potsdam dataset.

objective where the mutual information is maximized between the representation and image patches instead of the whole image. On the other hand, removing the pixel shift degrades the performance considerably. By removing the data augmentation techniques and not swapping the shared representations (no SSR + no DA) the performance is significantly degraded. We also study the impact of the representation space dimension without noticing significant differences between $z_{dim}$=10 and $z_{dim}$=32.

### 5.3 Image segmentation on Sentinel-2 time series

Since the Sentinel-2 mission does not provide pixel-level annotations for image segmentation, we perform only qualitative experiments. In contrast to the Potsdam case where the annotated pixels are randomly sampled from the available ground truth, now we ask a human operator to label $M$ pixels per class for each image during test time. The reader must note that scribbles, points or bounding box can be used to annotate the pixels. As these pixels are annotated under a human criterion, these pixels carry more significant information than pixels randomly sampled from the ground truth and thus the quality of the segmentation results improves significantly using just a few well-selected pixels. As shown in Figure 5 as the number of pixels per class $M$ increases, the segmentation results considerably improve. Nevertheless, the percentage of annotated pixels remains

| Model | $N = 6$ | | $N = 3$ | |
|---|---|---|---|---|
| | Accuracy | mIoU | Accuracy | mIoU |
| Baseline | 0.8717 | 0.7022 | 0.9163 | 0.7866 |
| Baseline + $\alpha = 0$ | 0.8724 | 0.7068 | 0.9147 | 0.7863 |
| Baseline + $\gamma = 0$ | 0.8636 | 0.6934 | 0.9026 | 0.7655 |
| Baseline + no jitter | 0.8767 | 0.7131 | 0.9097 | 0.7800 |
| Baseline + no flip | 0.8730 | 0.7077 | 0.9123 | 0.7815 |
| Baseline + no rotation | 0.8759 | 0.7094 | 0.9114 | 0.7819 |
| Baseline + no shift | 0.7584 | 0.5710 | 0.7949 | 0.6280 |
| Baseline + no SSR | 0.7230 | 0.5405 | 0.7576 | 0.5918 |
| Baseline + no SSR + no DA | 0.5973 | 0.4114 | 0.6497 | 0.4649 |
| Baseline + random $\phi$ | 0.3994 | 0.2384 | 0.5834 | 0.3986 |

**Table 3.** Ablation analysis results in terms of accuracy and mIoU for $N$ labels, $M = 50$ and $z_{dim} = 10$ in the Potsdam dataset.

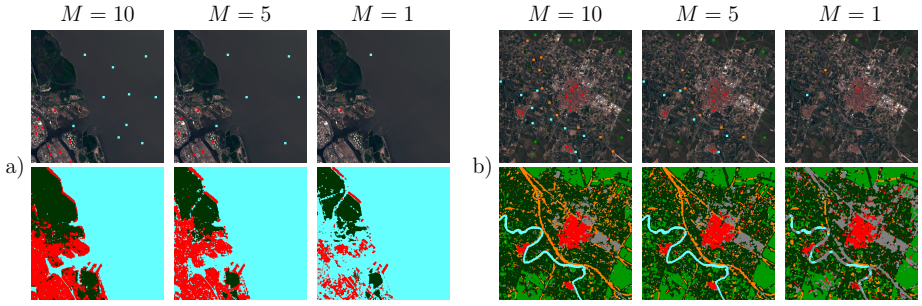



**Fig. 5.** Image segmentation examples in the Sentinel-2 dataset. A human operator identifies $N$ classes in the satellite image and selects $M$ pixels per label. a) Buenos Aires, Argentina; b) Valencia, Spain. Best viewed in color and zoom-in.

insignificant. For instance, 60 annotated pixels in a $512 \times 512$ pixel image represent less than 0.03% of the total number of pixels. Also the time required for image segmentation is reasonable, an image of $512 \times 512$ pixels with 60 annotated pixels takes around 33 seconds to be segmented.

**Segmentation over the time** Since we maximize the mutual information between images from the same time series, the learned representation ignores the temporal information. As a consequence, by annotating pixels from a single image our model is capable to segment the whole time series the image belongs to. In Figure 6, it can be seen that the segmentation results are coherent over the time. For instance, agricultural areas are belonging to the same class regardless of whether these areas are grown or harvested.

**Segmentation over the space** In the same manner we perform image segmentation over the time using a single image, our model is able to do it over the

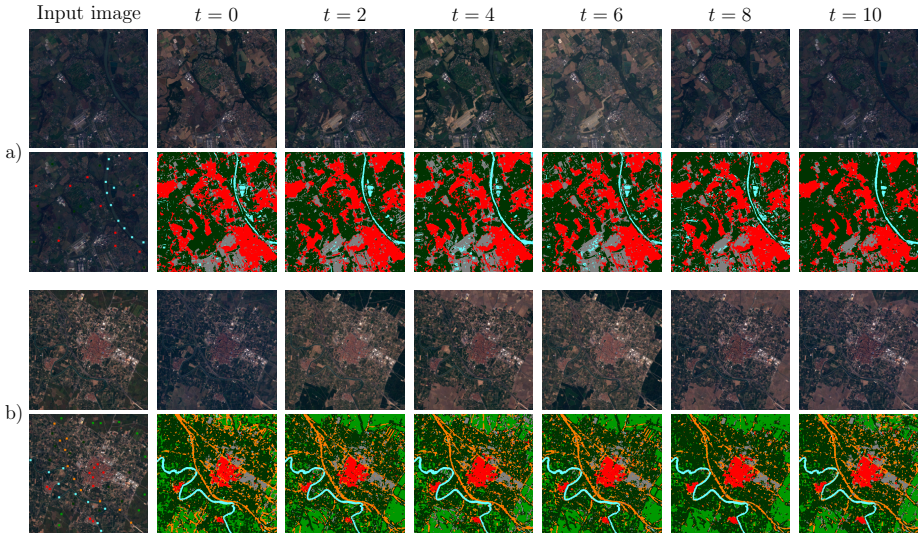



**Fig. 6.** Image segmentation over the time in the Sentinel-2 dataset. In the first column, the input image and the selected pixels are displayed for $M$=10. Our method is able to perform image segmentation with few labeled pixels on the entire time series the input image belongs to. The time series and the corresponding predictions are shown in the remaining columns for a) Toulouse, France; b) Valencia, Spain. Best viewed in color and zoom-in.

space. The annotated pixels provided by a human operator are generally used to perform image segmentation on the image these pixels are extracted from. We also use these annotated pixel to segment other images from the same area achieving satisfactory results as can be seen in Figure 7. In general, using annotated pixels from a single image we can perform image segmentation on images of the same area independently of the acquisition time.

## 6 Conclusion

In this paper, we have proposed to use a mutual information based similarity measure to perform image segmentation. Our approach offers the advantage of learning the proposed similarity measure in an unsupervised manner leveraging large amounts of unlabeled data. Then, per-pixel nearest-neighbor search using the proposed similarity measure is carried out to assign classes to the unlabeled pixels from the labeled pixels provided by a human operator. In particular, we have studied the case of aerial/satellite data where massive amounts of unlabeled images are available while the annotations are scarce. In the Potsdam case, our experiments suggest that the local mutual information objective is useful to measure similarity between pixels. Our approach outperforms other approaches based on state-of-the-art methods demonstrating the usefulness of our learned

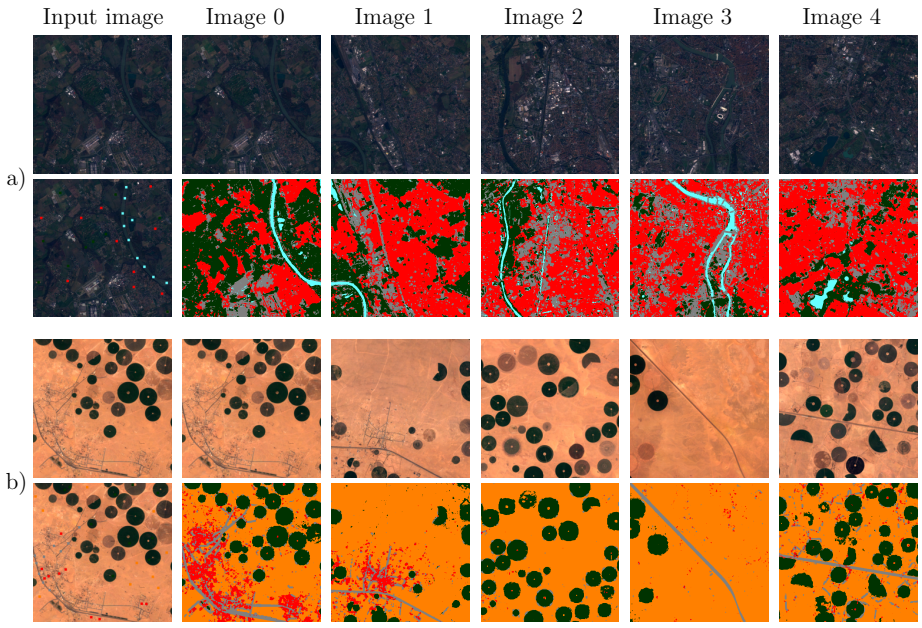



**Fig. 7.** Image segmentation over the space in the Sentinel-2 dataset. Selected pixels are not only useful for propagating the information from labeled pixels to unlabeled pixels in the same image but also in different images of the same area. a) Toulouse area, France; b) Tubarjal area, Saudi Arabia. Best viewed in color and zoom-in.

representation domain. On the other hand, the ablation experiments show that the model can be further simplified as some data augmentation techniques are more relevant and the global mutual information objective can be removed. In the Sentinel-2 case, we have shown that image segmentation can be performed over the time and over the space using a very few amount of annotated pixels, e.g. labeled pixels are less than 0.002% of the total number of pixels of a time series and it can be achieved in a reasonable amount of time.

## References

1. Bachman, P., Hjelm, R.D., Buchwalter, W.: Learning representations by maximizing mutual information across views. In: Advances in Neural Information Processing Systems 32 (2019)
2. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: Semantic segmentation with point supervision. In: European conference on computer vision (2016)
3. Belghazi, M.I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., Hjelm, D.: Mutual information neural estimation. In: Proceedings of the 35th International Conference on Machine Learning (2018)

4. Chechik, G., Sharma, V., Shalit, U., Bengio, S.: Large scale online learning of image similarity through ranking. Journal of Machine Learning Research (2010)
5. Chen, Y., Pont-Tuset, J., Montes, A., Van Gool, L.: Blazingly fast video object segmentation with pixel-wise metric learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
6. Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., et al.: Sentinel-2: Esa's optical high-resolution mission for gmes operational services. Remote sensing of Environment (2012)
7. ESA: The copernicus open access hub. https://scihub.copernicus.eu/
8. Fathi, A., Wojna, Z., Rathod, V., Wang, P., Song, H.O., Guadarrama, S., Murphy, K.P.: Semantic instance segmentation via deep metric learning. CoRR (2017), http://arxiv.org/abs/1703.10277
9. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. In: International Conference on Learning Representations (2019)
10. International Society for Photogrammetry and Remote Sensing: Isprs 2d semantic labeling contest. http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html
11. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning (2015)
12. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE International Conference on Computer Vision (2019)
13. Joon Oh, S., Benenson, R., Khoreva, A., Akata, Z., Fritz, M., Schiele, B.: Exploiting saliency for object segmentation from image level labels. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017)
14. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: Weakly supervised instance and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 876–885 (2017)
15. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (2014)
16. Lin, D., Dai, J., Jia, J., He, K., Sun, J.: Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
17. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2015)
18. Nowozin, S., Cseke, B., Tomioka, R.: f-gan: Training generative neural samplers using variational divergence minimization. In: Advances in neural information processing systems. pp. 271–279 (2016)
19. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. CoRR (2018), http://arxiv.org/abs/1807.03748
20. Ozair, S., Lynch, C., Bengio, Y., van den Oord, A., Levine, S., Sermanet, P.: Wasserstein dependency measure for representation learning. In: Advances in Neural Information Processing Systems 32 (2019)
21. Rother, C., Kolmogorov, V., Blake, A.: " grabcut" interactive foreground extraction using iterated graph cuts. ACM transactions on graphics (TOG) (2004)

22. Sanchez, E.H., Serrurier, M., Ortner, M.: Learning disentangled representations via mutual information estimation. CoRR (2019), http://arxiv.org/abs/1912.03915
23. Sun, J., Xu, Z.: Neural diffusion distance for image segmentation. In: Advances in Neural Information Processing Systems 32 (2019)
24. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. CoRR (2019), http://arxiv.org/abs/1906.05849
25. Xian, Y., Choudhury, S., He, Y., Schiele, B., Akata, Z.: Semantic projection network for zero-and few-label semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)