Self-Supervised Learning for Annotation Efficient Biomedical Image Segmentation

Luca Rettenberger*, Marcel Schilling*, Stefan Elser, Moritz Böhland, and Markus Reischl

***Abstract*— *Objective*: The scarcity of high-quality annotated data is omnipresent in machine learning. Especially in biomedical segmentation applications, experts need to spend a lot of their time into annotating due to the complexity. Hence, methods to reduce such efforts are desired. *Methods*: Self-Supervised Learning (SSL) is an emerging field that increases performance when unannotated data is present. However, profound studies regarding segmentation tasks and small datasets are still absent. A comprehensive qualitative and quantitative evaluation is conducted, examining SSL's applicability with a focus on biomedical imaging. We consider various metrics and introduce multiple novel application-specific measures. All metrics and state-of-the-art methods are provided in a directly applicable software package (`https://osf.io/gu2t8/`). *Results*: We show that SSL can lead to performance improvements of up to 10%, which is especially notable for methods designed for segmentation tasks. *Conclusion*: SSL is a sensible approach to data-efficient learning, especially for biomedical applications, where generating annotations requires much effort. Additionally, our extensive evaluation pipeline is vital since there are significant differences between the various approaches. *Significance*: We provide biomedical practitioners with an overview of innovative data-efficient solutions and a novel toolbox for their own application of new approaches. Our pipeline for analyzing SSL methods is provided as a ready-to-use software package.**

***Index Terms*— Biomedicine, Contrastive Learning, Deep Learning, Segmentation, Self-Supervised Learning**

I. Introduction

Recently, Deep Learning (DL) has shown great potential in various research areas, including biomedicine [16], [26], [40], [52], [64]. However, since data is essential for DL algorithms and annotating samples can be tedious, these approaches are often constrained by the lack of suitable annotated datasets. Generating annotations for segmentation tasks is especially time-consuming; hence such datasets are particularly affected by annotation scarcity. This is further amplified within biomedical imaging. High inter- and intra-subject variations are ubiquitous. Relevant regions are often difficult to separate as they appear heterogeneous and the shape or borders are fuzzy and hard to determine [12], [49]. Further, when developing diagnostics solutions, the analysis algorithms need to be robust [42]. All these difficulties conflict with deep learning algorithms as they usually need vast amounts of accurately annotated samples to perform well [51].

Self-Supervised Learning (SSL) is an emerging approach to counteract the gap between the large amount of data needed in deep learning and the difficulties in annotating. It finds characteristics in unannotated data and builds a knowledge base [24]. This can increase the performance on a small portion of annotated samples of the same or a similar domain. Especially within biomedical image segmentation, this introduces multiple benefits. First, the general robustness of the Machine Learning (ML) system is enhanced by task-agnostic knowledge [65]. Second, the importance of annotations is reduced since SSL creates knowledge without needing any segmentation masks. Third, the delineation quality of relevant regions increases as SSL does not depend on possibly erroneous human annotations [8]. Despite the great potential, in-depth investigations regarding SSL for segmentation tasks, small-scale datasets, and applicability within the biomedical domain are missing.

This work's contributions are: (i) A comprehensive study regarding the potential and applicability of SSL in segmentation applied to biomedical imaging with limited data is conducted. (ii) We evaluate the state-of-the-art methods concerning visual representation learning, dense predictions, and biomedical applications using various qualitative and quantitative metrics. (iii) Multiple novel metrics are introduced, focusing to evaluate SSL in-depth depending on the number of available annotations. (iv) A software package is deployed, including all SSL methods and evaluation metrics.

All code and data employed in this paper are open-source and available at `https://osf.io/gu2t8/`.

II. Related Work

Solutions to computer vision challenges usually revolve around building Supervised Learning (SL) systems with remarkable solutions for a particular problem [16], [22], [30], [56] but no transferability to other challenges. Such methods are gradually recognized as a limiting factor [37]. Further, human annotations are often erroneous, so numerous works try to reduce the human component in ML or focus on data-efficient learning approaches like active learning, semi-supervised learning, or transfer learning [44]–[46], [50], [51],

* Equal contribution.

Luca Rettenberger, Marcel Schilling, Moritz Böhland, and Markus Reischl are with the Institute for Automation and Applied Informatics (IAI), Karlsruhe Institute of Technology (KIT), Hermann-von-Helmholtz-Platz 1 76344 Eggenstein-Leopoldshafen (e-mail: Luca.Rettenberger@kit.edu).

Stefan Elser is with the Institute for Artifical Intelligence, RWU Hochschule Ravensburg-Weingarten University of Applied Sciences Doggenriedstraße, 88250 Weingarten.

[58], [69], [70]. A particularly promising approach to reduce annotation efforts is SSL, which finds inherent structures in unannotated images [33].

SSL consists of two phases: the pretext task $\mathcal{P}$ (or pre-training) and the downstream task $\mathcal{D}$. The pretext task learns features of a given unannotated dataset by transforming samples into a vector (= representation learning). Pseudo annotations are generated by the SSL method to train $\mathcal{P}$ applying conventional SL approaches [29], [43]. The learned parameters of $\mathcal{P}$ are subsequently used as initialization in $\mathcal{D}$, which may be any ML challenge. SSL for imaging data consist of three categories: i) *generative*, ii) *generative-discriminative* and, iii) *discriminative* methods. *Generative* methods employ Autoencoders (AEs) with reconstruction error-based losses (denoising AEs) [3], [55]. *Generative-discriminative* methods use Generative Adversarial Networks (GANs) [20] to learn image representations [14], [15], [17]. *Discriminative* methods apply metrics on image representations directly [6], [8], [9], [21], [65].

Contrastive Learning (CL) belongs to the *discriminative* methods and currently dominates the state of the art in SSL [8], [65]. Here, pseudo annotations are generated by modifying the appearance of an image $\mathbf{x}$. The augmentation process $\mathcal{T}$ is used, which produces modification functions $t \sim \mathcal{T}$ that are employed to obtain two views $t(\mathbf{x}) = \mathbf{x}'$ and $t'(\mathbf{x}) = \mathbf{x}''$ of the same depicted object. Using the similarity function $\mathcal{F}$, CL maps $\mathbf{x}'$ and $\mathbf{x}''$ to a similarity value $s \in \mathbb{R}_+$

$$\mathcal{F} : \mathbf{x}' \times \mathbf{x}'' \mapsto s. \tag{1}$$

$\mathcal{F}$ is trained to output high values for similar ($\mathbf{x}'$ and $\mathbf{x}''$) samples and low values for dissimilar (any other image) samples [36]. To prevent $\mathcal{F}$ from yielding a constant distance of 0 (model collapse), it must be trained on as many dissimilar samples as possible. In DL, $\mathcal{F}$ is implemented with two Neural Network (NN) encoders f_Θ and f_ξ, that map the views into a lower-dimensional feature space, combined with a distance metric $d()$ (see Fig. 1). The parameters of f_Θ and f_ξ are then employed into a downstream task. Considering dense predictions (classifying each pixel of an image), the encoder of an Encoder-Decoder Network (EDN) is provided with f_Θ and f_ξ for the downstream training.

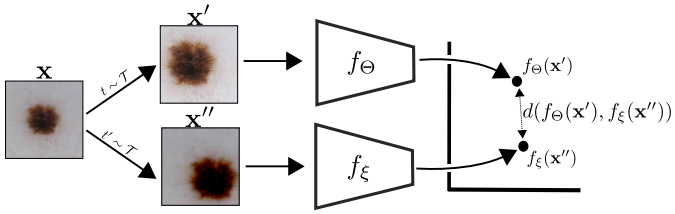

Fig. 1. **Concept of contrastive learning:** The sample image $\mathbf{x}$ is modified by an augmentation process $\mathcal{T}$ to produce two different views $t(\mathbf{x}) = \mathbf{x}'$ and $t'(\mathbf{x}) = \mathbf{x}''$. Both views are then mapped into a feature space by the encoders f_Θ and f_ξ. During training the distance $d(f_\Theta(\mathbf{x}'), f_\xi(\mathbf{x}''))$ between the two mappings is minimized.

Within CL multiple approaches exist. MoCo [24] uses a query encoder f_Θ that is trained with regular backpropagation and a momentum encoder f_ξ that is updated with a linear interpolation of f_Θ and f_ξ itself. MoCo also introduces an encoding queue containing the representations from previous training batches to access many dissimilar samples during training. SimCLR [8] introduces projection heads consisting of multiple Multilayer Perceptrons (MLPs), that map the views yet again into a lower vector space. Within SimCLR f_Θ and f_ξ share all parameters. Bootstrap Your Own Latent (BYOL) [21] avoids model collapse without the need for dissimilar samples by defining the parameters ξ as an exponential moving average of Θ. Barlow Twins [65] tries to make the cross-correlation matrix of a batch of samples close to identity, effectively moving together similar samples (on-diagonal) and decorrelating the other samples (off-diagonal). DenseCL [59] extends MoCo by employing one projection head as introduced in SimCLR and a second one tailored for dense predictions. DetCo [60] is a second extension to MoCo that uses many projection heads to improve dense predictions. DenseCL and DetCo have been proven on dense predictions, SimCLR and MoCo in the biomedical environment [59], [60], [63], [67]. Barlow Twins, BYOL, AE were evaluated neither on dense predictions nor biomedical data. A graphical overview is given in the supplementary.

A popular benchmark for CL is to pre-train a NN on the ImageNet [48] classification challenge and use the obtained parameters as initialization for the actual task [41], [61]. While this works well for large datasets within similar domains as ImageNet, it is unknown whether this benefit can be transferred to more specific domains like the biomedical field, small-scale datasets, or segmentation challenges.

Most studies within SSL for biomedical imaging focus on classification tasks [4], [53], [57]. Such studies cannot be considered for segmentation since the semantic consistency is disregarded if certain essential image augmentations are applied [8], [66][1]. Some works address this issue by tailoring frameworks for dense prediction tasks [7], [25], [38], [59], [62] and a few studies examine segmentation tasks, however, with very large datasets [10].

The fundamental challenges are summarized as follows: (i) Current studies assume that the database for training consists of millions of samples, which is not the case in real-world scenarios. (ii) Studies on SSL in the biomedical imaging field are scarce; there is no comprehensive application-related study. (iii) Statements about whether the attention to semantic consistency in SSL makes a difference for the actual application are missing. (iv) There is no collection of evaluation methods to reliably and comparably evaluate SSL methods. No metrics evaluate SSL depending on available annotations. (v) No framework combines different SSL techniques and additional tools and methods.

III. Concept

A. Experiments

We systematically compare the state of the art in SSL on biomedical data with various metrics to conclude about the applicability of the methods. Our experiments are divided into

[1]For example, if an image is cropped at two random positions, the two crops may contain objects of different semantic categories

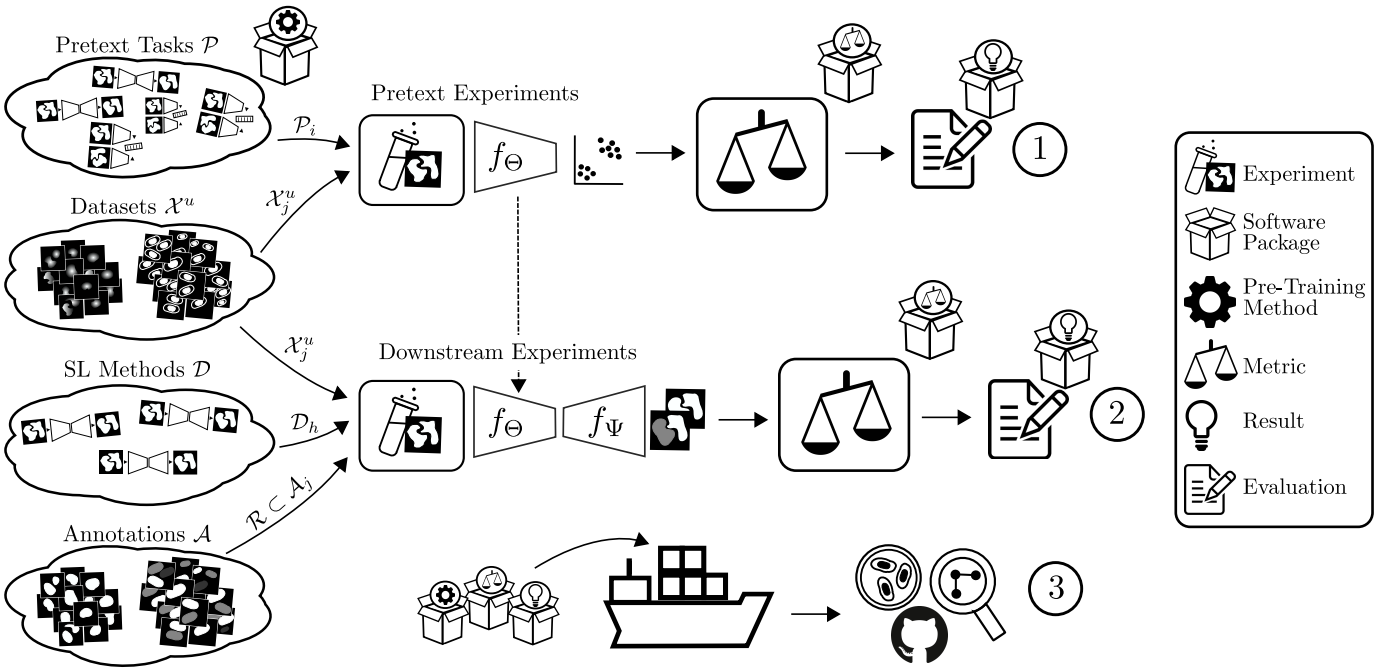

Fig. 2. **Overview of our work:** Our experiments are conducted in three stages ①, ②, and ③. In the first stage, a set of pretext tasks $\boldsymbol{\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \dots\}}$ are combined with multiple unannotated biomedical imaging datasets $\boldsymbol{\mathcal{X}^u = \{\mathcal{X}_1^u, \mathcal{X}_2^u, \mathcal{X}_3^u, \dots\}}$ and used in the pretext experiments to parameterize useful encoders $\boldsymbol{f_\Theta}$ and evaluated with multiple metrics (①). The encoders $\boldsymbol{f_\Theta}$ are then employed into the downstream experiments, which consist of regular Supervised Learning (SL) methods $\boldsymbol{\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \dots\}}$ (②) and are trained with a EDN consisting of the contracting path $\boldsymbol{f_\Theta}$ and an expanding path $\boldsymbol{f_\Psi}$. For the downstream tasks, the datasets are extended with corresponding annotations $\boldsymbol{\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \dots\}}$. Usually, only a subset $\boldsymbol{\mathcal{R}}$ of the available annotations $\boldsymbol{\mathcal{A}_j \supset \mathcal{R}}$ are provided. Multiple metrics evaluate the downstream experiments to assess the performance, depending on the available annotations $\boldsymbol{\mathcal{R}}$. All pre-training methods, evaluation metrics, and formalizable conclusions are provided as a self-contained software solution, ready to be used in industrial deployment, biomedical applications, or further research (③).

two stages: the first focuses on pretext tasks, and the second employs the learned representations into the downstream tasks.

1) Pretext Comparison: In the first part a set of SSL pretext tasks $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \dots\}$ are combined with unannotated biomedical datasets $\mathcal{X}^{\mathrm{u}} = \{\mathcal{X}_1^{\mathrm{u}}, \mathcal{X}_2^{\mathrm{u}}, \mathcal{X}_3^{\mathrm{u}}, \dots\}$ to learn useful representations and parameterize the encoder f_Θ (Fig. 2, ①). The learned parameters Θ are evaluated independently of the subsequent downstream task with various metrics. *ImageNet* pre-training is used as a baseline. Further, an *Autoencoder* is trained, which is the simplest type of representation learning. For the SSL methods, a combination of classical methods (*SimCLR*, *Barlow Twins*, *BYOL*, and *MoCo*) and the latest approaches designed for dense tasks (*DenseCL* and *DetCo*) are used to obtain an all-encompassing overview.

2) Downstream Application: The second part employs the learned representations into the downstream tasks $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \dots\}$ to evaluate the performance on real-world segmentation challenges. For this $\mathcal{X}^{\mathrm{u}}$ is enriched with corresponding annotations/segmentation masks $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \dots\}$ and the provided parameters f_Θ are used as initialization. The decoder f_Ψ of the employed EDN is randomly initialized. Since SSL is especially interesting for partially annotated datasets, we observe how the performance behaves depending on the number of annotated samples. The downstream tasks are evaluated with multiple metrics to assess the performance on each dataset $\mathcal{X}_j^u$ with annotations $\mathcal{A}_j$, a pretext task $\mathcal{P}_i$, and downstream task $\mathcal{D}_h$ depending on the number of available annotations $\mathcal{R} \subset \mathcal{A}_j$ (Fig. 2, ②). Powers of two are used as annotation rates $\rho_s = 2^s/100$, $s = 0, \dots, 6$ to focus on situations with few annotations. We also investigate whether freezing the learned parameters f_Θ is a viable option, as this reduces the training effort and the number of required annotated samples [27]. Since 2% available annotations are difficult but not impossible to solve if the pretext task provides good representations, we choose to select exemplary results for each pretext task for this annotation rate. In our work, *No Pretraining* means that the parameters of the encoder are randomly initialized [22] and not changeable during training.

To make all results easily accessible and to be able to transfer our evaluation pipeline to other challenges, all methods, employed metrics, and formalizable conclusions are provided as a self-contained software package ready to be applied to any biomedical challenge (Fig. 2, ③). Details about the training configurations are given in the supplementary.

B. Evaluation Methods

We provide a collection of qualitative and quantitative metrics to investigate SSL.

1) Implementation and Hardware Requirements: We consider the number of hyperparameters ψ and the qualitative implementation overhead κ for each SSL method. Since SSL methods extend the employed encoder f_θ for training and require larger batch sizes than SL, we also evaluate the relative number of parameters $\Delta\theta$ compared to f_θ and the required batch size b.

2) Class Activation Maps: Observing the the Class Activation Maps (CAMs) [68] of the encoder trained with SSL enables the visual interpretation of the learned features with attention maps. We evaluate the quality of a CAM as how much the computed focus is on the segmented object.

3) Centered Kernel Alignment: Centered Kernel Alignment (CKA) [34] is a similarity metric that is invariant to orthogonal transformations and isotropic scaling. It observes the compared NNs at multiple feature layers. It is not relevant if identical filters of two NNs are located at different positions in a layer. Therefore, it is a suitable measure to quantitatively evaluate features of multiple SSL methods and to evaluate how similar the layers of the encoders are compared to the network being trained supervised with annotations. High similarity to the supervised case means good performance since the same parameters could be found without annotations.

C. Novel Evaluation Methods

We introduce three novel metrics, as the existing evaluation methods do not analyze SSL approaches in a sufficient quantification degree and considering the given annotations.

1) Neighborhood Quality Criterion (NQC): Considering a representation $\mathbf{p}$ of a dataset $\mathcal{X}$, the nearest $D_\mathrm{n}()$ and farthest $D_\mathrm{f}()$ neighbors calculated with the Euclidean distance $\mathrm{d}()$

$$D_\mathrm{n}(\mathbf{p}) = \underset{\mathbf{k}\in\mathcal{X}}{\operatorname{argmin}}\{\mathrm{d}(\mathbf{p},\mathbf{k})\}$$
$$D_\mathrm{f}(\mathbf{p}) = \underset{\mathbf{k}\in\mathcal{X}}{\operatorname{argmax}}\{\mathrm{d}(\mathbf{p},\mathbf{k})\},$$

provide valuable insights since samples of the same latent class should be clustered together and be located far away from dissimilar samples [28]. We additionally introduce the quantifying NQC measure that summarizes the neighborhood quality in one value. NQC iterates over the test data $\mathcal{X}^\mathrm{t}$ and outputs 1 if the nearest neighbor $D_n(\mathcal{X}_i^\mathrm{t})$ of each sample $\mathcal{X}_i^\mathrm{t}$ is from the same class and 0 otherwise

$$Q_\mathrm{NQC} = \frac{1}{|\mathcal{X}^\mathrm{t}|}\sum_{i=1}^{|\mathcal{X}^\mathrm{t}|}\begin{cases}1, & \text{if } \mathcal{X}_i^\mathrm{t} \text{ and } D_\mathrm{n}(\mathcal{X}_i^\mathrm{t}) \text{ are of same class}\\ 0, & \text{otherwise,}\end{cases}$$

where $|\mathcal{X}^\mathrm{t}|$ is the cardinality of $\mathcal{X}^\mathrm{t}$. Since NQC is dependent on the number of classes k in $\mathcal{X}^\mathrm{t}$, it can be employed to compare different representation spaces, but not different datasets and has to be > 1, to be able to find the nearest neighbor. To reduce the effect of the curse of dimensionality [31], we additionally employ a Principal Component Analysis (PCA) [1] before evaluating $D_\mathrm{n}()$, that maps the representations into a 10-dimensional feature subspace. Assume a random representation distribution for a dataset $\mathcal{D}$ of length l with classification classes $\mathcal{C}$. The expected value $\mathbb{E}_\mathrm{random}$ of Q_NQC is calculated as the sum over the probability $\mathrm{P}()$ multiplied by the number of samples for each class

$$\mathbb{E}_\mathrm{random}\left[Q_\mathrm{NQC}\right] = \frac{1}{l}\sum_{c\in\mathcal{C}}\mathrm{P}(c)|c|, \tag{2}$$

where $|c|$ denotes the cardinality of c.

2) Runtime Quality Criterion (RQC): We evaluate the temporal requirements of SSL approaches, as they require substantial training effort and powerful hardware. The training time is quantified into a comparable value by the novel RQC metric that uses the point of stabilized loss values (convergence) in the non-convex objective function. The time $\mathrm{t}()$ until this convergence happens for a method δ is compared relative to the fastest converging method t*

$$Q_\mathrm{RQC} = \frac{\mathrm{t}(\delta)}{\mathrm{t}^*}. \tag{3}$$

3) Integrated Quality Criterion (IQC): Considering that SSL methods strongly depends on the number of available annotations, we introduce a novel metric called the Integrated Quality Criterion (IQC), which evaluates the performance of downstream tasks, taking the number of available annotations into account.

In the following $\Omega()$ is some quality criterion, like the Aggregated Jaccard Index (AJI+) [35] or the Dice-Sørensen coefficient (DSC) [13]. The annotation rate is defined as $0.0 \leq \varrho \leq 1.0$ with $\varrho = 0.0$ meaning no annotations and $\varrho = 1.0$ fully annotated. Let a be the maximum and b be the minimum annotation rate. All annotation rates $\{b,\ldots,a\}$ with $0.0 \leq a,b \leq 1.0$ and $a > b$, are applied to $\Omega()$ to acquire the measurements $\mathrm{P} = \{\Omega(b),\ldots,\Omega(a)\}$. P is linearly interpolated to obtain a continuous function $f_{\Omega,\mathrm{lin}}(\varrho)$. To describe the overall quality concerning different annotation rates ϱ we introduce the IQC formula as

$$Q_\mathrm{IQC} = \frac{1}{\Omega(\varrho=1)(a-b)}\int_b^a f_{\Omega,\mathrm{lin}}(\varrho)\mathrm{d}\varrho.$$

Since the achieved quality with a completely annotated data set $\Omega(\varrho = 1)$ can be assumed to be the maximum value, the integral is normalized with the product of $\Omega(\varrho = 1)$ and the interval length $a - b$ (the maximum possible area). An illustration of the IQC is given in the supplementary.

To verify IQC, we conduct a one-tailed t-test [32]. The null hypothesis H_0 states that the respective method is equal to or worse than random initialization.

IV. Results

A. Datasets

Two small-scale biomedical imaging datasets reflecting realistic scenarios are observed in this work. They are different in nature and cover various aspects that the practitioner may encounter in biomedicine. Fig. 3 displays samples of both datasets.

1) ISIC Melanoma: The first dataset stems from the 2017 International Skin Imaging Collaboration (ISIC) challenge [11] (ISIC Melanoma dataset) and contains 2.600 close-up RGB images, split into 2.000 train and 600 test samples. The segmentation task is to generate binary masks which locate the lesion in the respective image. The dataset also contains a categorization task with three classes: *Seborrheic Keratosis*, *Melanoma*, and *Unknown*.

2) MoNuSeg: The second dataset is part of the 2018 MICCAI challenge [35] (MoNuSeg dataset) and contains histopathological images of different types of organs. There are 30 training and 14 test images. The challenge is to segment and identify each nucleus in the multi-organ images (instance segmentation). Additionally, there are classification annotations given that differentiate the respective organs. The organ classes are *Kidney*, *Colon*, *Breast*, *Bladder*, *Prostate Liver*, *Stomach*, *Brain*, and *Lung*. The classes *Liver* and *Stomach* are only contained in the training, *Brain* and *Lung* only in the test set.

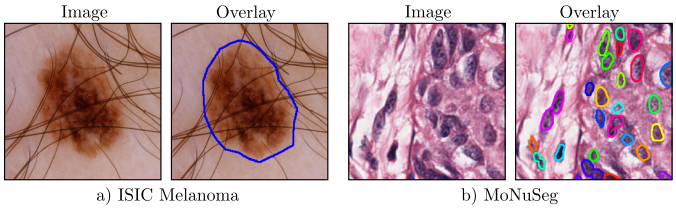

Fig. 3. **Example images:** a) ISIC Melanoma dataset [11] and b) MoNuSeg dataset [35]. Segmentation masks are displayed as overlay. Color markings are provided to assist in separating the individual instances.

Details regarding the distribution of samples for both datasets are provided in the supplementary.

B. Architecture, Training, and Implementation

The dimensions of the samples of the MoNuSeg dataset are fairly large (1000×1000 pixels). Hence, we divide each image into multiple crops of dimensions of 256×256 to not distort the content while keeping the individual images processable. For the ISIC Melanoma dataset, the samples are resized to 256×256 pixels.

The correct image augmentations are crucial for SSL. Therefore, the augmentations from [59] used in many SSL frameworks are chosen for the ISIC Melanoma dataset. Since the MoNuSeg dataset consists of histopathological images, other augmentations must be chosen. As there is no standard for this type of data, we identified a set of fitting augmentation parameters in an extensive hyperparameter search. For all augmentations except *Gaussian blurring*, the *Range* value describes ranges of relative percentage changes. With *Gaussian blurring*, the *Range* value describes the standard deviation. P specifies the probability that the respective augmentation is applied

- Random Cropping. Range $[0.2,\ 1.0]$ and P=100%.
- Brightness modifications. Range: $[0.4,\ 1.6]$ and P=80%.
- Contrast modifications. Range: $[0.2,\ 1.8]$ and P=80%.
- Saturation modifications. Range: $[0.2,\ \ 1.8]$ and P=80%.
- Brightness modifications. Range: $[0.8,\ 1.2]$ and P=80%.
- Gaussian blurring. Range: $[0.1,\ 2.0]$ and P=80%.
- Horizontal and vertical flipping. P=50%.

All SSL pretext tasks are trained with the SGD optimizer with a weight decay of $1\text{x}10^{-4}$, momentum of 0.9, and learning rate of 1×10^{-3}. Additionally, we employ Cosine Annealing to the learning rate [39]. As downstream task, we either solve a semantic segmentation (ISIC Melanoma) or instance segmentation (MoNuSeg) task. We employ the U-Net [47] architecture with a ResNet-50 [23] backbone. For semantic segmentation, we use the Dice Loss and for instance segmentation the smooth L1 loss, both in combination with the Adam optimizer and a learning rate of 1×10^{-3}. To segment instances, we employ a subsequent seed-based watershed post-processing.

The whole architecture and training loop is implemented in PyTorch Lightning [18]. The Albumentations [5] library is used to implement the image augmentations. For visualizing the Gradient Class Activation Maps (Grad-CAMs), we use the PyTorch Grad-CAMs library [19], and PyTorch Model Compare [54] to calculate and display the CKA matrices.

As evaluation metrics, we employ the DSC for semantic segmentation and the AJI+ for instance segmentation.

Training is performed on cluster nodes equipped with NVIDIA A100 Tensor Core GPUs.

C. Pretext Comparison

1) Nearest and Farthest Neighbor Retrival: Fig. 4a displays the *Nearest* and *Farthest* neighbors considering the ISIC Melanoma dataset for three *Reference* images. The first image stems from the *Seborrheic Keratosis* class and for most approaches the *Nearest* neighbor is also within this class and clear qualitative similarities are visible for *DenseCL* and *MoCo*: the reference image and the two methods display a round border around the object of interest. Qualitative similarities are also present in the second sample (*Melanoma* class). The *Nearest* neighbors for *ImageNet*, *BYOL*, *DenseCL*, and *MoCo* are not only within the same class but also look similar (patchy, disseminated melanoma with fuzzy borders). Even though the *Nearest* neighbor of *DetCo* and *SimCLR* are not from the *Melanoma* class, the images still appear similar. Only for the *Autoencoder* and *Barlow Twins* the *Nearest* neighbors appear less similar. In the last sample (*Unknown* class) all methods have the *Nearest* neighbor in the same class and appear similar as small black melanoma with sharp edges (except for *Barlow Twins*). For all reference images and methods, the *Farthest* neighbors look dissimilar. However, the results indicate that the methods are not always able to accurately capture the classifications, as the farthest neighbors occasionally belong to the same class as the reference image. This discrepancy can likely be attributed to large visual variations within the individual classes.

The qualitative observations are supported by the NQC. Assuming a random mapping of samples into the representation space, the expected value of Q_{NQC} calculates to $\mathbb{E}_{\text{random}}[Q_{\text{NQC}}] = 0.49$ (see Eq. 2). The *Autoencoder*, *Barlow Twins*, and *DetCo* do not surpass this value. All other methods are noticeably better than $\mathbb{E}_{\text{random}}$, which shows that a class-dependent clustering while training the SSL approaches occurred even though it is not flawless. *DenseCL* in particular is striking with a value of 0.64 (see Tab. I, where $\uparrow$ means that large, and $\downarrow$ that small values are better. The best method is marked in bold).

Fig. 4b shows the *Nearest* and *Farthest* neighbors for the MoNuSeg dataset. As for the leftmost sample, the *Nearest*

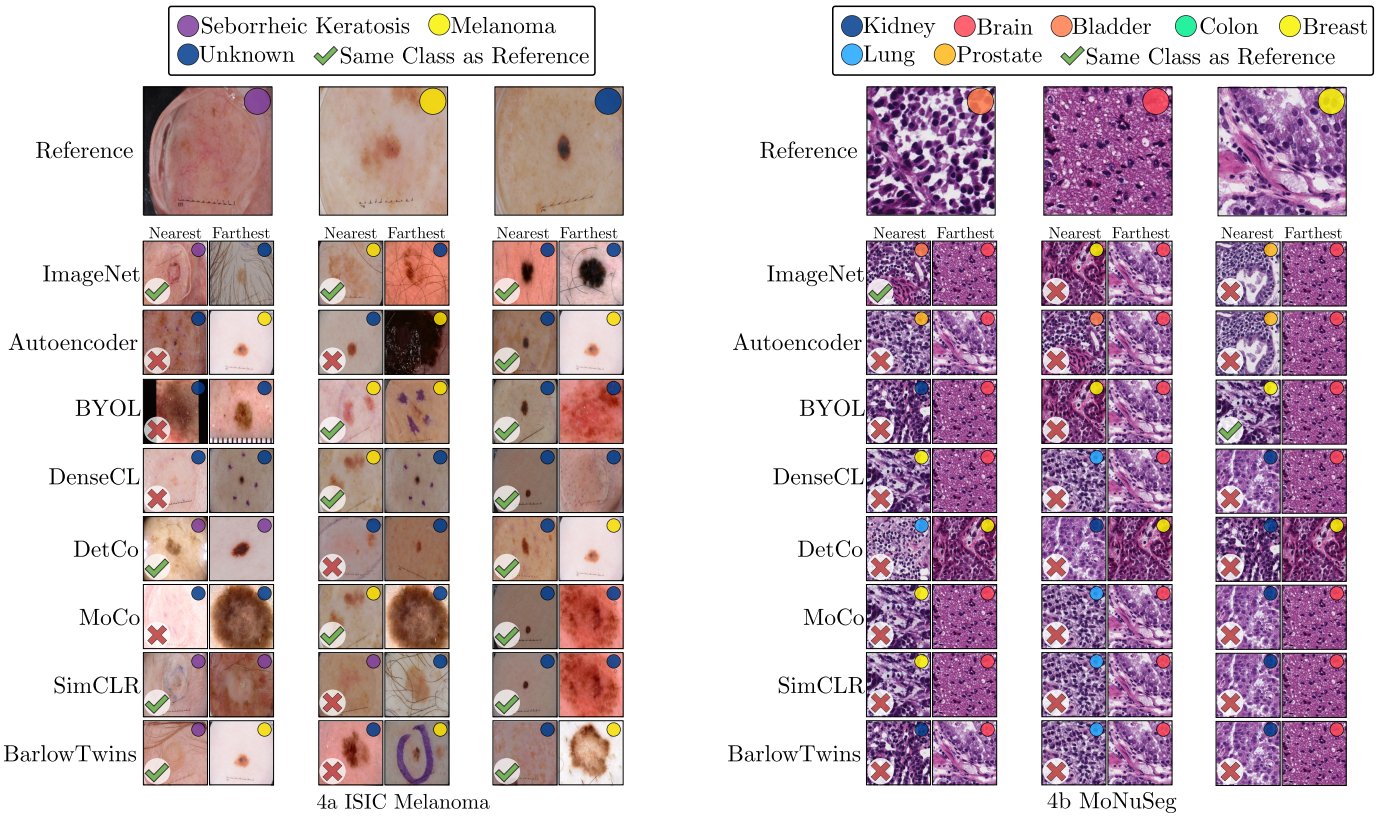

Fig. 4. **Nearest and farthest neighbors:** The circle in the upper right corner shows either the ground-truth class (reference image) or the predicted classes. Samples are marked with green check marks if they are of the same class as the reference image and with red crosses if not. The nearest and farthest neighbors are calculated as the Euclidean distance after a Principal Component Analysis (PCA) with 10 output components.

neighbors for most methods appear similar to the reference image, with strong, dark colors and contrasting red and white tones. Only the *Autoencoder* and *DetCo* stand out with a marginally different appearance. The middle sample is less clear. For *ImageNet*, *Autoencoder*, and *BYOL*, the *Nearest* neighbors appear similar. For the other methods, this is not the case. For the third reference image, the neighborhood is similarly indistinct. Four methods (*DenseCL*, *MoCo*, *SimCLR*, and BarlowTwins) have the same sample as *Nearest* neighbor. *ImageNet* and the *Autoencoder* have a different *Nearest* neighbor than the other methods, but it is also identical between the two approaches. With the MoNuSeg dataset the importance and expressiveness of the correlation between class label clustering and actual representation quality is lower than with ISIC Melanoma, as the challenge of this dataset is instance segmentation and not classification. Class labels are available in the MoNuSeg dataset, but it is not designed as a classification challenge.

The difficulties mentioned above are also apparent in calculating Q_{NQC}. Random representation mapping is computed as $\mathbb{E}_{\text{random}}[Q_{\text{NQC}}] = 0.15$. Almost all methods do not outperform this value. The *Autoencoder* even falls below it. This is due to *Autoencoder* pre-training providing parameters that are more informative than *Random* but are not aligned with the given class labels, leading to a representation space that is worse in terms of class-based clustering compared to no pre-training. Only *DenseCL* is able to surpass $\mathbb{E}_{\text{random}}$ with a value of 0.23 (see Tab. I).

TABLE I

Nearest NEIGHBORS. Q_{NQC} IS THE *Nearest* NEIGHBOR QUALITY CRITERION.

Method	$Q_{\text{NQC}} \uparrow$ ISIC Melanoma	MoNuSeg
ImageNet	0.58	0.15
Autoencoder	0.48	0.07
BYOL [21]	0.57	0.15
DenseCL [59]	**0.64**	**0.23**
DetCo [60]	0.48	0.15
MoCo [24]	0.56	0.15
SimCLR [8]	0.56	0.15
Barlow Twins [65]	0.48	0.15

Evaluating the *Nearest* neighbors shows that the methods seem to build representation spaces that share visible qualitative characteristics. Our novel NQC metric quantifies these results. *DenseCL* emerged as the most promising method for observing the neighborhood for both observed datasets.

2) Application and Hardware Requirements: Tab. II displays relevant factors for evaluating hardware requirements and application overhead. *ImageNet* and *Autoencoder* do not contain any hyperparameters ($\psi = 0$). *ImageNet* classification results are provided online, reducing κ to a download and no additionally trained parameters ($\Delta\theta = +0$M). *Autoencoder* needs the skip connections of the U-Net to be removed and increases the number of parameters by $\Delta\theta = +11$M, as the expanding path must be additionally trained. For both, b is

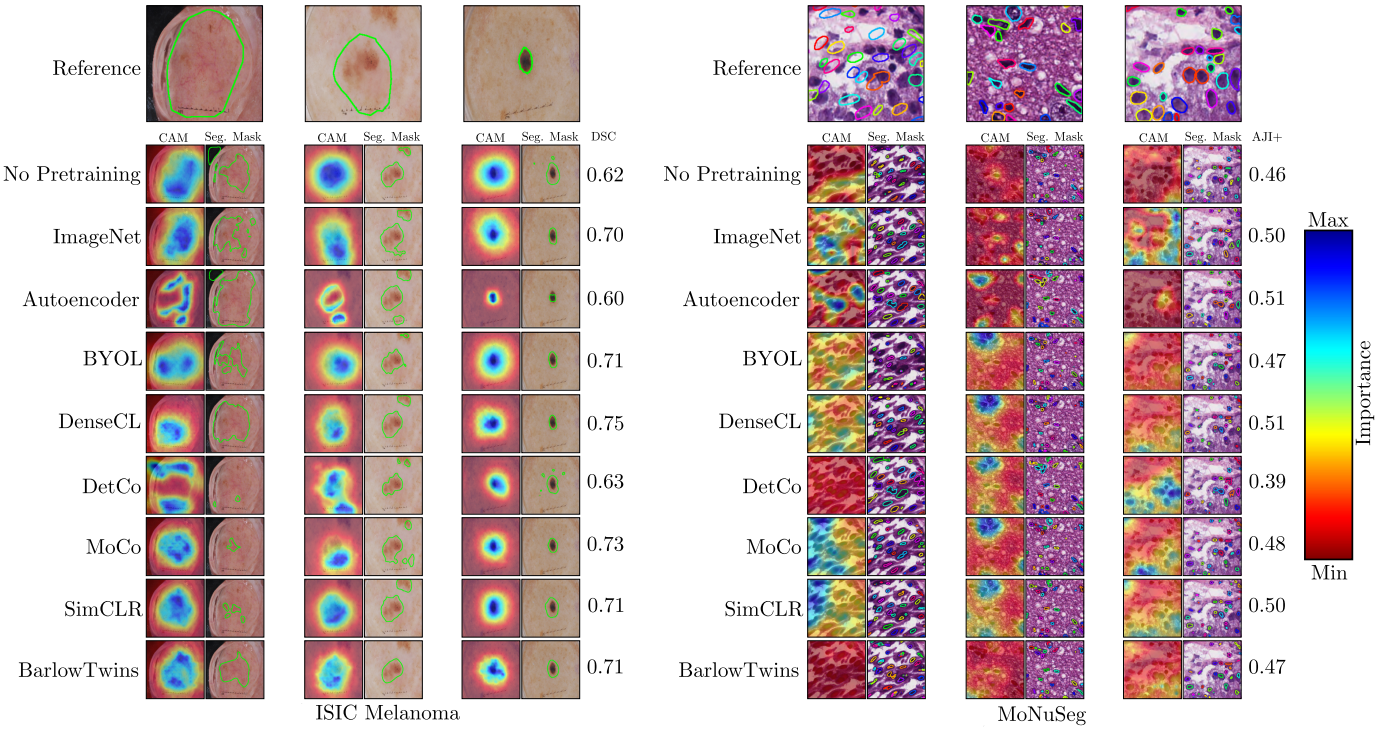

Fig. 5. **Class activation maps:** For each reference image, the ground-truth segmentation mask is given as an overlay. For each method, the CAM for the respective reference image and the predicted segmentation mask is given. The U-Net was trained with 2% labeled samples for each dataset (the remaining 98% unlabeled samples were removed for the downstream task). The encoder of the U-Net was frozen. The DSC and AJI+ are given, and calculated over the whole test split of the respective dataset.

about the same size as in SL.

SimCLR, *Barlow Twins*, and *BYOL* need the generation of image pairs, the contrastive loss function, and the projection head to be implemented. All three have one hyperparameter to be tuned ($\psi = 1$) and require batch sizes $\mathbf{b}$ big enough to provide enough negative samples for approximating the similarity space. *SimCLR* contains the fewest parameters ($\Delta\theta$ = +2.2M), followed by *Barlow Twins* ($\Delta\theta$ = +12.6M) and lastly *BYOL* ($\Delta\theta$ = +46.6M).

All *MoCo*-based approaches (*MoCo*, *DenseCL*, and *DetCo*) employ queue-based methods for negative samples, greatly reducing the required batch size b while increasing the complexity κ. *MoCo* and *DetCo* contain two hyperparameters ($\psi = 2$) and *DenseCL* three ($\psi = 3$). *DetCo* requires by far the most parameters of all methods due to the many projection heads ($\Delta\theta$ = +946.5M).

For RQC, t* is set to *Autoencoder*, as it is the fastest method. The values are averaged over both datasets. The queue-based methods *MoCo*, *DenseCL*, and *DetCo* need the longest training times. *SimCLR* ($\mathbf{Q}_{\mathrm{RQC}} = 4.07$) is the fastest CL method, followed by *Barlow Twins* ($\mathbf{Q}_{\mathrm{RQC}} = 4.53$) and *BYOL* ($\mathbf{Q}_{\mathrm{RQC}} = 4.23$).

D. Downstream Application

1) Class Activation Maps: Fig. 5 shows three reference images, each with the predicted segmentation masks, the CAM, and the respective quality metric (DSC or AJI+). The ISIC Melanoma is trained with 32 annotated samples, and the MoNuSeg dataset with 10.

TABLE II

APPLICATION AND HARDWARE REQUIREMENTS. ψ IS THE NUMBER OF HYPERPARAMETERS TO BE TUNED, κ THE IMPLEMENTATION EFFORT, $\Delta\theta$ THE NUMBER OF PARAMETERS RELATIVE TO THE RESULTING BACKBONE (RESNET-50 [23]), $\mathbf{b}$ THE REQUIRED BATCH SIZE, AND $\mathbf{Q}_{\mathrm{RQC}}$ IS THE RELATIVE TIME UNTIL CONVERGENCE AND AVERAGED OVER BOTH DATASETS.

Method	$\psi \downarrow$	κ	$\Delta\theta \downarrow$	b	$Q_{\mathrm{RQC}} \downarrow$
ImageNet	**0**	**+++**	**+0M**	**+++**	-
Autoencoder	**0**	++	+11M	**+++**	**1.00**
BYOL [21]	1	+	+46.6M	- -	4.23
DenseCL [59]	3	-	+41.4M	++	5.45
DetCo [60]	2	- -	+946.5M	++	5.12
MoCo [24]	2	-	+32.4M	++	4.93
SimCLR [8]	1	+	+2.2M	- -	4.07
Barlow Twins [65]	1	+	+12.6M	- -	4.53

For the ISIC Melanoma dataset, the performance is acceptable even with *No Pretraining*, since looking at the CAMs the melanomas of the reference image are at least roughly detected. Using an *Autoencoder* as pretext task (DSC = 0.60) reduces the performance compared to *No Pretraining* (DSC = 0.62). This is further supported by the CAMs, since the *Autoencoder* version of the U-Net pays attention to implausible regions of the images. *ImageNet* pre-training can provide very good results (DSC = 0.70), considering that it is the leading method in terms of implementation and training effort. Still, all CL methods apart from *DetCo* outperform *ImageNet* at least by a small amount. Considering the DSC score, *BYOL* (DSC = 0.71) , *SimCLR* (DSC = 0.71), and *Barlow Twins* (DSC =

0.71) are on the same level.

For the MoNuSeg dataset, the results look less straightforward, presumably because it is more challenging, as many individual instances have to be segmented. Still, distinctive differences in the various methods can be recognized. Observing the AJI+, *DetCo* performs the worst (AJI+ = 0.39), even being inferior to *No Pretraining* (AJI+ = 0.46). *ImageNet* (AJI+ = 0.50) and the *Autoencoder* (AJI+ = 0.51) are sensible choices for the MoNuSeg dataset, outperforming most of the SSL methods. Only *DenseCL* (AJI+ = 0.51) is on par with the *Autoencoder*. Looking at the leftmost reference image, *DenseCL* pays attention to the regions where many nuclei are located, while the *Autoencoder* has a less clear focus. Overall, the differences between the various pretext methods are marginal, suggesting that the approaches can extract little information from the MoNuSeg dataset.

As *DenseCL* provides the best results for both datasets, it can be assumed that dedicated pretext methods, adjusted to the downstream task, are useful.

2) Centered Kernel Alignment (CKA): Tab. III displays the comparison of the different pretext methods compared to SL training on the whole dataset.

TABLE III
CKA BETWEEN SSL METHODS AND SUPERVISED TRAINING. *ConvX* DESCRIBES THE CKA SIMILARITY FOR THE X'TH LAYER OF THE RESNET-50 [23] ENCODER. μ DESCRIBES THE MEAN VALUE OVER ALL LAYERS.

a) ISIC Melanoma

Method	Conv1↑	Conv2↑	Conv3↑	Conv4↑	μ↑
ImageNet	**0.88**	**0.86**	0.75	0.65	0.79
Autoencoder	0.68	0.51	0.45	0.37	0.50
BYOL [21]	0.83	0.82	0.73	0.60	0.75
DenseCL [59]	0.85	0.82	**0.81**	**0.73**	**0.80**
DetCo [60]	0.65	0.46	0.40	0.36	0.47
MoCo [24]	0.77	0.70	0.74	0.70	0.73
SimCLR [8]	0.85	0.82	0.79	0.71	0.79
Barlow Twins [65]	0.72	0.61	0.66	0.70	0.67

b) MoNuSeg

Method	Conv1↑	Conv2↑	Conv3↑	Conv4↑	μ↑
ImageNet	0.95	0.78	0.45	**0.48**	**0.67**
Autoencoder	0.96	0.74	0.39	0.35	0.61
BYOL [21]	0.91	0.65	0.28	0.25	0.52
DenseCL [59]	**0.97**	**0.79**	**0.47**	0.46	**0.67**
DetCo [60]	0.62	0.31	0.11	0.13	0.29
MoCo [24]	0.96	0.76	0.45	0.43	0.65
SimCLR [8]	0.95	0.76	0.46	0.44	0.65
Barlow Twins [65]	0.76	0.61	0.45	0.45	0.57

For the ISIC Melanoma dataset, parallels between the CAMs and high values regarding the CKA are visible. Further, the methods that emerge as the most capable also have the most similar representations to SL. This means that suitable pretext tasks learn very similar parameters as fully supervised training (with enough training data). Furthermore, these parallels also show that the quantifying metric CKA can extend or even replace the purely qualitative CAMs. Comparing the best methods determined by the mean (μ) CKA over all layers, *ImageNet* (μ = 0.79) has the best representations in the upper layers (Conv1 and Conv2), *SimCLR* (μ = 0.79) does not have quite as good representations in the upper layers, but flattens less in the lower layers (Conv3, Conv4). *DenseCL* (μ = 0.80), has the overall best and least decreasing CKA values. This supports the notion that transfer learning works so well with *ImageNet* since it provides general representations that are useful for all kinds of challenges, but the task-specific parameters have to be relearned (fine-tuning). SSL on the other side learns useful features throughout the whole network.

For the MoNuSeg dataset, the CKA values are much higher in the first layers than in the lower ones. This is expected for *ImageNet*, as the domain of histopathological data is quite detached from the content of the *ImageNet* dataset. Observing the SSL methods, the upper filters of the ResNet seem to fit very well but then rapidly worsen. This is a strong indication that the training could not provide enough information for learning complex and specialized features (the lower layers of the network). Nevertheless, *DenseCL* (μ = 0.67) provides the best parameters almost throughout the whole network, on par with *ImageNet* (μ = 0.67).

3) Integrated Quality Criterion (IQC): Looking at the quantitative evaluation of our novel IQC metric, the findings can be further confirmed (see Tab. IV). We discriminate between *frozen* and *unfrozen* encoders to determine whether fine-tuning the provided parameters of the pretext task is sensible. The Q_{IQC} summarizes the performance over the entire range of label ratios into one value. Thus, it can be seen as an approximation of a fully comprehensive view. A detailed visualization is provided in the supplementary.

TABLE IV
QUANTITATIVE COMPARISON SSL METHODS IN THE CONTEXT OF SEMANTIC SEGMENTATION (ISIC MELANOMA) AND INSTANCE SEGMENTATION (MONUSEG). Q_{IQC} IS CALCULATED WITH THE DSC (ISIC MELANOMA) OR WITH THE AJI+ (MONUSEG). THE p VALUES OF THE t-TEST ARE GIVEN IN BRACKETS. WE DISCRIMINATE BETWEEN **FROZEN** ENCODERS AND **UNFROZEN** ENCODERS. PERFORMANCE METRICS ARE PROVIDED IN PERCENTAGES.

	a) ISIC Melanoma		b) MoNuSeg	
	Q_{IQC} (p) ↑ [%]		Q_{IQC} (p) ↑ [%]	
Method	Frozen	Unfrozen	Frozen	Unfrozen
Random	85.6 (-)	91.9 (-)	81.0 (-)	90.3 (-)
ImageNet	92.7 (0.000)	95.4 (0.000)	**90.5** (0.000)	92.5 (0.021)
Autoencoder	89.4 (0.000)	92.7 (0.000)	80.0 (0.76)	90.1 (0.000)
BYOL [21]	92.5 (0.000)	94.9 (0.000)	83.1 (0.048)	90.5 (0.410)
DenseCL [59]	**94.1** (0.000)	**96.3** (0.000)	86.9 (0.001)	**93.5** (0.003)
DetCo [60]	80.1 (0.999)	90.3 (0.991)	78.2 (0.997)	84.8 (0.992)
MoCo [24]	93.4 (0.000)	96.0 (0.000)	87.8 (0.000)	93.4 (0.003)
SimCLR [8]	93.9 (0.000)	95.6 (0.000)	85.5 (0.002)	93.3 (0.016)
Barlow Twins [65]	91.7 (0.000)	92.8 (0.019)	79.6 (0.869)	86.9 (0.999)

As in the previous experiments, *DenseCL* is the leading method, followed by *ImageNet*. However, the performance gaps are much smaller if we look at the whole range of available annotations. Observing the ISIC Melanoma dataset, Q_{IQC} improves noticeably, compared to *Random*, for all SSL methods apart from *DetCo*. For *frozen* encoders, the best method *DenseCL* improves Q_{IQC} by 8.5%. If we *unfreeze* the encoder, each method gets better. This means that the learned features of the SSL methods are not sufficient to be considered optimal in the downstream task and should at least be fine-tuned for the best results. *DenseCL* still achieves an

improvement of 4.4% compared to *Random*. Moreover, all methods (apart from *DetCo*) are at least slightly better than *Random*. This shows that even with varying annotation rates employing a pretext task is reasonable.

Observing the MoNuSeg dataset with *frozen* encoders, almost all methods enhance the performance in the downstream task compared to *Random* ($Q_{\text{IQC}} = 81.0$). Only *Barlow Twins* ($Q_{\text{IQC}} = 79.6.0$) and *DetCo* ($Q_{\text{IQC}} = 78.2$) worsen the results. This displays parallels to the CKA similarities from Tab. III, as these two methods have the lowest values with a considerable gap to all others. Unlike *DetCo*, *Barlow Twins* was not bad for the ISIC melanoma dataset. This may suggest that *Barlow Twins* is not suitable for instance segmentation tasks. Further, if the encoder is frozen, *ImageNet* has the best representations, showing that it can provide good representation even if the domains have no apparent commonalities. Further, this either means that SSL is not suitable for this downstream task or that the dataset is not large enough. With unfrozen encoders, the results look quite different. While *ImageNet* only enhances marginally, the SSL approaches can demonstrate significant improvements. This shows that SSL, unlike *ImageNet*, learns useful representations throughout the whole network. If the features deep in the network are not very suitable, relearning them takes a lot of effort, which is most likely the case with the parameters provided by *ImageNet*. An additional evaluation on a dataset containing breast ultrasound images [2] is available in the supplementary material.

V. Discussion

Our evaluation shows that *ImageNet* pre-training is a capable pre-training method that can be used with little effort. Still, most SSL approaches perform better than *ImageNet* if the parameters are adjustable during training, challenging the previous assumption that SSL requires millions of samples to deliver good results. *DenseCL*, in particular, consistently displays a clear performance advantage compared to other methods, which shows that attention to spatial information improves segmentation. Observing the pretext comparison, especially regarding the application overhead and hardware requirements, it is questionable whether SSL methods are cost-effective. However, a closer look at the representations of SSL approaches shows that SSL learns task-specific, complex, features while transfer learning provides simple ones. Also, our results show that the parameters of SSL methods most likely converge to the same parameters found in SL when sufficient data is available. This strongly indicates that the potential of SSL is not fully exhausted yet.

Looking at different number of available annotations, our novel evaluation method IQC approximates a fully comprehensive view of the respective dataset and the employed SSL methods. The NQC provides quantifications of classification annotations and the RQC summarizes training time comparisons into one value.

Many practical insights are obtained. *ImageNet* and *Autoencoder* provide good results regarding pre-training, if little time can be invested. When employing SSL for segmentation, methods tailored specifically for this challenge are the best option. *DenseCL* is especially promising. Even though DetCo was designed for segmentation tasks as well, it does not yield sufficient results. There is a strong correlation between the results of SSL methods and clustering regarding the classification annotations, which should be observed if such annotations are available. Further, SSL is more effective for semantic segmentation than for instance segmentation. As SSL shows great potential but also noticeable differences between the ISIC Melanoma and MoNuSeg, new approaches and datasets should always be compared with our framework in a structured way.

VI. Conclusion

We presented a comprehensive analysis regarding Self-Supervised Learning (SSL) in biomedical image segmentation and developed a framework for evaluating SSL methods with a variety of existing and novel qualitative and quantitative criteria. All methods and evaluation metrics are provided as a self-contained software solution, ready to be used in industrial deployment, biomedical applications, or further research (`https://osf.io/gu2t8/`). Our results on two small-scale biomedical datasets show that SSL improves segmentation tasks, especially if annotations are missing. In particular, methods explicitly tailored for segmentation tasks can produce improvements of up to 10% in overall performance.

Our evaluation pipeline offers in-depth insights into the inner workings of SSL and clearly quantifies which methods are best suited for a specific dataset. These detailed examinations sparked many inspirations for future works: new methods that focus on the segmentation of single instances of objects, targeting specific layers in the neural network for optimization, or the combination of transfer learning and SSL are ideas for future work.

Acknowledgment

This work was supported in part by the HoreKa Supercomputer through the Ministry of Science, Research, and the Arts Baden-Württemberg, in part by the Federal Ministry of Education and Research, and the Helmholtz Association Initiative and Networking Fund on the HAICORE@KIT partition.

References

[1] Hervé Abdi and Lynne J Williams, Principal component analysis, *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[2] Al-Dhabyani *et al.*, Dataset of breast ultrasound images, *Data in brief*, 28:104863, 2020.

[3] Guillaume Alain and Yoshua Bengio, What regularized auto-encoders learn from the data-generating distribution, *The Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

[4] Shekoofeh Azizi *et al.*, Big self-supervised models advance medical image classification, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 3478–3488. IEEE, 2021.

[5] Alexander Buslaev *et al.*, Albumentations: fast and flexible image augmentations, *arXiv:1809.06839*, 2018.

[6] Mathilde Caron *et al.*, Unsupervised learning of visual features by contrasting cluster assignments, *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

[7] Krishna Chaitanya and volume=33 pages=12546–12558 year=2020 publisher=MIT Press others, journal=Advances in Neural Information Processing Systems, Contrastive learning of global and local features for medical image segmentation with limited annotations.

[8] Ting Chen *et al.*, A simple framework for contrastive learning of visual representations, In *Proc. of the International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[9] Ching-Yao Chuang *et al.*, Debiased contrastive learning, *arXiv:2007.00224*, 2020.

[10] Ozan Ciga *et al.*, Self supervised contrastive learning for digital histopathology, *Machine Learning with Applications*, 7:100198, 2022.

[11] Noel CF Codella *et al.*, Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic), In *IEEE 15th International Symposium on Biomedical Imaging (ISBI)*, pages 168–172. IEEE, 2018.

[12] Thomas M Deserno, Fundamentals of biomedical image processing, In *Biomedical Image Processing*, pages 1–51. Springer, 2010.

[13] Lee R Dice, Measures of the amount of ecologic association between species, *Ecology*, 26(3):297–302, 1945.

[14] Jeff Donahue *et al.*, Adversarial feature learning, *arXiv:1605.09782*, 2016.

[15] Jeff Donahue and Karen Simonyan, Large scale adversarial representation learning, *Advances in Neural Information Processing Systems*, 32:10541–10551, 2019.

[16] Michele D'Orazio *et al.*, Electro-optical classification of pollen grains via microfluidics and machine learning, *IEEE Transactions on Biomedical Engineering*, 69(2):921–931, 2021.

[17] Vincent Dumoulin *et al.*, Adversarially learned inference, *arXiv:1606.00704*, 2016.

[18] William Falcon and The PyTorch Lightning team, PyTorch Lightning, Accessed: Sep. 19, 2022, GitHub: `https://github.com/Lightning-AI/lightning`, version 1.4, 2019.

[19] Jacob Gildenblat and contributors, Pytorch library for cam methods, Accessed: Sep. 19, 2022, GitHub: `https://github.com/jacobgil/pytorch-grad-cam`, version 1.4.5, 2021.

[20] Ian Goodfellow *et al.*, Generative adversarial nets, *Advances in Neural Information Processing Systems*, 27:2672–2680, 2014.

[21] Jean-Bastien Grill *et al.*, Bootstrap your own latent: A new approach to self-supervised learning, *arXiv:2006.07733*, 2020.

[22] Kaiming He *et al.*, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, In *Proc. of the IEEE International Conference on Computer Vision (ECCV)*, pages 1026–1034. IEEE, 2015.

[23] Kaiming He *et al.*, Deep residual learning for image recognition, In *Proc. of the IEEE conference on computer vision and pattern recognition*, pages 770–778. IEEE, 2016.

[24] Kaiming He *et al.*, Momentum contrast for unsupervised visual representation learning, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 9729–9738. IEEE, 2020.

[25] Kaiming He *et al.*, Masked autoencoders are scalable vision learners, *arXiv:2111.06377*, 2021.

[26] Fabian Isensee *et al.*, nnu-net: a self-configuring method for deep learning-based biomedical image segmentation, *Nature Methods*, 18(2):203–211, 2021.

[27] Leo F Isikdogan *et al.*, Semifreddonets: Partially frozen neural networks for efficient computer vision systems, In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 193–208. Springer, 2020.

[28] Ashish Jaiswal *et al.*, A survey on contrastive self-supervised learning, *Technologies*, 9(1):2–22, 2020.

[29] Longlong Jing and Yingli Tian, Self-supervised visual feature learning with deep neural networks: A survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4037–4058, 2020.

[30] John Jumper *et al.*, Highly accurate protein structure prediction with alphafold, *Nature*, 596(7873):583–589, 2021.

[31] Eamonn Keogh and Abdullah Mueen, *Curse of Dimensionality*, pages 314–315, Springer US, Boston, MA, 2017.

[32] Tae Kyun Kim, T test as a parametric statistic, *Korean Journal of Anesthesiology*, 68(6):540–546, 2015.

[33] Alexander Kolesnikov *et al.*, Revisiting self-supervised visual representation learning, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 1920–1929. IEEE, 2019.

[34] Simon Kornblith *et al.*, Similarity of neural network representations revisited, In *Proc. of the International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[35] Neeraj Kumar *et al.*, A dataset and a technique for generalized nuclear segmentation for computational pathology, *IEEE Transactions on Medical Imaging*, 36(7):1550–1560, 2017.

[36] Yann LeCun and Fu Jie Huang, Loss functions for discriminative training of energy-based models, In *International Workshop on Artificial Intelligence and Statistics*, pages 206–213. PMLR, 2005.

[37] Yann LeCun and Ishan Misra, Self-supervised learning: The dark matter of intelligence, 2021, *URL https://ai. facebook. com/blog/self-supervised-learning-the-dark-matter-of-intelligence*, 2021.

[38] Zhaowen Li *et al.*, Mst: Masked self-supervised transformer for visual representation, *Advances in Neural Information Processing Systems*, 34:13165–13176, 2021.

[39] Ilya Loshchilov and Frank Hutter, Sgdr: Stochastic gradient descent with warm restarts, *arXiv:1608.03983*, 2016.

[40] William Lotter *et al.*, Robust breast cancer detection in mammography and digital breast tomosynthesis using an annotation-efficient deep learning approach, *Nature Medicine*, 27(2):244–249, 2021.

[41] Alexander Mathis *et al.*, Pretraining boosts out-of-domain robustness for pose estimation, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 1859–1868. IEEE, 2021.

[42] Ralf Mikut, *Data Mining in der Medizin und Medizintechnik*, volume 22, KIT Scientific Publishing, 2008.

[43] Andrew Owens and Alexei A Efros, Audio-visual scene analysis with self-supervised multisensory features, In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 631–648. Springer, 2018.

[44] Jialin Peng and Ye Wang, Medical image segmentation with limited supervision: a review of deep network models, *IEEE Access*, 9:36827–36851, 2021.

[45] Pengzhen Ren *et al.*, A survey of deep active learning, *ACM Computing Surveys*, 54(9):1–40, 2021.

[46] Luca Rettenberger *et al.*, Annotation efforts in image segmentation can be reduced by neural network bootstrapping, *Current Directions in Biomedical Engineering*, 8(2):329–332, 2022.

[47] Olaf Ronneberger *et al.*, U-net: Convolutional networks for biomedical image segmentation, In *Proc. of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[48] Olga Russakovsky *et al.*, Imagenet large scale visual recognition challenge, *International Journal of Computer Vision*, 115(3):211–252, 2015.

[49] Marcel P Schilling *et al.*, Impact of annotation noise on histopathology nucleus segmentation, *Current Directions in Biomedical Engineering*, 8(2):197–200, 2022.

[50] Mark Schutera *et al.*, Cuepervision: self-supervised learning for continuous domain adaptation without catastrophic forgetting, *Image and Vision Computing*, 106:104079, 2021.

[51] Mark Schutera *et al.*, Methods for the frugal labeler: Multi-class semantic segmentation on heterogeneous labels, *PLOS ONE*, 17(2):e0263656, 2022.

[52] Tianyu Shi *et al.*, C 2 ma-net: Cross-modal cross-attention network for acute ischemic stroke lesion segmentation based on ct perfusion scans, *IEEE Transactions on Biomedical Engineering*, 69(1):108–118, 2021.

[53] Kirill Sirotkin *et al.*, Improved skin lesion recognition by a self-supervised curricular deep learning approach, *arXiv:2112.12086*, 2021.

[54] Anand Subramanian, Pytorch model compare, Accessed: Sep. 19, 2022, GitHub: `https://github.com/AntixK/PyTorch-Model-Compare`, version 0.21, 2021.

[55] Pascal Vincent *et al.*, Extracting and composing robust features with denoising autoencoders, In *Proc. of the International Conference on Machine Learning*, pages 1096–1103. PMLR, 2008.

[56] Athanasios Voulodimos *et al.*, Deep learning for computer vision: A brief review, *Computational intelligence and neuroscience*, 2018, 2018.

[57] Dan Wang *et al.*, Unlabeled skin lesion classification by self-supervised topology clustering network, *Biomedical Signal Processing and Control*, 66:102428, 2021.

[58] Guotai Wang *et al.*, Interactive medical image segmentation using deep learning with image-specific fine tuning, *IEEE transactions on medical imaging*, 37(7):1562–1573, 2018.

[59] Xinlong Wang *et al.*, Dense contrastive learning for self-supervised visual pre-training, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 3024–3033. IEEE, 2021.

[60] Enze Xie *et al.*, Detco: Unsupervised contrastive learning for object detection, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 8392–8401. IEEE, 2021.

[61] Yiting Xie and David Richmond, Pre-training on grayscale imagenet improves medical image classification, In *Proc. of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0. Springer, 2018.

[62] Zhenda Xie *et al.*, Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 16684–16693. IEEE, 2021.

[63] Jiashu Xu, A review of self-supervised learning methods in the field of medical image analysis, *International Journal of Image, Graphics and Signal Processing(IJIGSP)*, 13:33–46, 2021.

[64] Xiaonan Zang *et al.*, Methods for 2-d and 3-d endobronchial ultrasound image segmentation, *IEEE Transactions on Biomedical Engineering*, 63(7):1426–1439, 2015.

[65] Jure Zbontar *et al.*, Barlow twins: Self-supervised learning via redundancy reduction, *arXiv:2103.03230*, 2021.

[66] Yucheng Zhao *et al.*, Self-supervised visual representations learning by contrastive mask prediction, In *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 10160–10169. IEEE, 2021.

[67] Siting Zheng *et al.*, Contrastive learning-based adenoid hypertrophy grading network using nasoendoscopic image, In *Proc. of the International Symposium on Computer-Based Medical Systems (CBMS)*, pages 377–382. IEEE, 2022.

[68] Bolei Zhou *et al.*, Learning deep features for discriminative localization, In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929. IEEE, 2016.

[69] Fuzhen Zhuang *et al.*, A comprehensive survey on transfer learning, *Proceedings of the IEEE*, 109(1):43–76, 2020.

[70] Maria A Zuluaga *et al.*, Learning from only positive and unlabeled data to detect lesions in vascular ct images, In *International conference on medical image computing and computer-assisted intervention*, pages 9–16. Springer, 2011.