HAW HAMBURG

MASTER THESIS
Anja Witte

Masked Autoencoder: Influence of Self-Supervised Pretraining on Object Segmentation in Industrial Images

Faculty of Engineering and Computer Science
Department Computer Science

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG
Hamburg University of Applied Sciences

Anja Witte

Masked Autoencoder: Influence of Self-Supervised Pretraining on Object Segmentation in Industrial Images

Master thesis submitted for examination in Master´s degree
in the study course *Master of Science Informatik*
at the Department Computer Science
at the Faculty of Engineering and Computer Science
at University of Applied Science Hamburg

Supervisor: Prof. Dr. Christian Lins
Second Supervisor: Dr. Sascha Lange

Submitted on: 13.10.2023

Anja Witte

Thema der Arbeit

Masked Autoencoder: Einfluss von Self-Supervised Pretraining auf Objektsegmentierung in industriellen Bildern

Stichworte

Masked Autoencoder, Self-Supervised Pretraining, Semantische Segmentierung, UNETR, Label-Effizienz, Holzplatzkräne

Kurzzusammenfassung

Die Menge der gelabelten Daten ist in industriellen Anwendungsfällen häufig gering, da der Annotierungsprozess zeit- und kostenaufwändig ist. In der Forschung wird zunehmend self-supervised Pretraining z. B. mit Masked Autoencodern (MAE) angewandt, um Segmentierungsmodelle mit weniger Labeln zu trainieren. In der vorliegenden Masterthesis wird daher der Einfluss des MAE-Pretrainings auf die Label-Effizienz für die semantische Segmentierung mit U-Net Transformers am Beispiel von Holzplatzkränen analysiert. Zusätzlich wird Transfer Learning in Bezug auf den Krantyp und die Perspektive im Kontext der Label-Effizienz betrachtet. Die Ergebnisse zeigen, dass der MAE erfolgreich anwendbar ist und die Label-Effizienz erhöht. Der stärkste positive Einfluss wird bei allen Experimenten in den niedrigeren Labelmengen gefunden. Der höchste Effekt wird beim Transfer Learning in Bezug auf Kräne erzielt, wobei die IoU und der Recall um 4,31 % bzw. 8,58 % verbessert werden. Weitere Analysen zeigen, dass die Erhöhung der Label-Effizienz aus einer besseren Unterscheidung zwischen dem Hintergrund und den segmentierten Kranobjekten resultieren.

Anja Witte

Title of Thesis

Masked Autoencoder: Influence of Self-Supervised Pretraining on Object Segmentation in Industrial Images

Keywords

Masked Autoencoder, Self-Supervised Pretraining, Semantic Segmentation, UNETR, Label-efficiency, Log-yard Cranes

Abstract

The amount of labeled data in industrial use cases is limited because the annotation process is time-consuming and costly. In research, self-supervised pretraining such as Masked Autoencoder (MAE) are used to train segmentation models with fewer labels. Therefore, this master thesis analyses the influence of MAE pretraining on the label-efficiency for semantic segmentation with U-Net Transformers. This is investigated for the use case of log-yard cranes. Additionally, two transfer learning cases with respect to crane type and perspective are considered in the context of label-efficiency. The results show that MAE is successfully applicable and increases label-efficiency. The strongest positive influence is found for all experiments in the lower label amounts. The highest effect is achieved with transfer learning regarding cranes, where IoU and Recall increase by 4.31 % and 8.58 %, respectively. Further analyses show that improvements result from a better distinction between the background and the segmented crane objects.

Contents

List of Figures

Abbreviations

AI Artificial Intelligence.

CNN Convolutional Neural Network.

KAI Kran AI.

MAE Masked Autoencoder.

ML Machine Learning.

MLP Multi-Layer Perceptron.

NLP Natural Language Processing.

UNETR U-Net Transformers.

ViT Vision Transformer.

1 Introduction

Automation is an increasingly important topic for industrial companies as it positively impacts speed, safety and quality. In particular, routine tasks such as packaging, material handling, and goods unloading can be automated well [1].

One of the benefiting sectors is the growing pulp and paper industry. Production automation, including logistics, processes and remote process control, offers diverse use cases to improve efficiency and reduce costs [3]. Among others, current activities comprise the automation of the log-yard operation. With an autonomous log-yard crane, the required workforce can be reduced as operating processes, including unloading and storage strategy, are optimised and handled by an Artificial Intelligence (AI) [2].

To achieve this, an AI requires continuous perception of its environment through sensors such as LiDAR scanners or industrial cameras. With the help of Machine Learning (ML) methods for computer vision, the images can be processed to gain information about wood piles, crane parts and other relevant objects. To obtain an object's location, image segmentation methods can be applied. The images are separated into semantically related areas, showing e. g. the grapple [2].

ML models for semantic segmentations are trained supervised and require many annotated images. These image masks include a pixel-wise classification of each pixel [4]. As the model performance depends on the quality of the training data, the manual labeling process is time-consuming and costly.

To limit this effort, a reduction of the needed data is possible. This can be achieved through a two-stage training approach, including pretraining and finetuning. A model is trained to learn general image representations while using a large amount of unlabeled pretraining data. Based on this knowledge, less labeled data is required to train the model task-specific features [12]. Recently, Masked Autoencoder (MAE) [11] has shown promising results for pretraining Vision Transformer (ViT).

As research focuses on datasets such as ImageNet [16], the question of applicability in the industrial sector arises. Therefore, this thesis aims, on the one hand, to analyse the usage of MAE for the industrial use case of autonomous cranes. On the other hand, an investigation on the influence of MAE regarding the required amount of labeled data for image segmentation is targeted. For this purpose, label-efficiency is analysed in the context of the segmentation architecture U-Net Transformers (UNETR) [9].

Furthermore, research has shown that pretraining knowledge from a domain can be advantageously transferred to a related target domain [8]. Based on this idea, the thesis further investigates the influence of using different but related datasets for pretraining and finetuning on label-efficiency. Two cases are considered in this context. On the one hand, two cameras are mounted on the trolley that differ slightly in perspective as one is on the left side and the other on the right side. This raises the question of whether labeled images from one perspective are sufficient for training and how label-efficiency is influenced by pretraining with a different camera perspective.

On the other hand, two cranes that differ in their appearance and environment are considered. Since the relevant objects are similar, the influence of transferring knowledge between different crane types on label-efficiency is also analysed. From these investigation topics, the following research questions are posed:

1. Can self-supervised pretraining with MAE increase the label-efficiency of semantic segmentation with UNETR in industrial images?
2. Can self-supervised pretraining with MAE using images of a crane perspective increase the label-efficiency of semantic segmentation with UNETR for another crane perspective?
3. Can self-supervised pretraining with MAE using images of a crane increase the label-efficiency of semantic segmentation with UNETR for another crane?

To answer these questions, the thesis is structured as follows. Section 2 introduces the relevant background knowledge. This includes an explanation of the industrial use case of autonomous cranes where the used datasets come from. In addition, the theoretical foundation of ViTs is addressed. This topic is introduced as ViT is the network architecture on which the chosen segmentation model UNETR is based. In section 3, the research questions are investigated as a paper. Finally, section 4 presents practical implications of the results and possibilities for further research.

2 Background

This section provides an introduction to the relevant background to this thesis. Firstly, the industrial use case of autonomous cranes is addressed. Secondly, the network architecture of ViT is explained.

2.1 Industrial Use Case

As the thesis aims to perform analyses in an industrial use case, it is carried out in the context of the Kran AI (KAI) project[1] at PSIORI GmbH. The project goal is the automation of log-yard cranes (see figure 2.3) used in the paper industry's wood yard. The objective of automation is the crane steering and includes unloading a truck and moving wood to an infeed conveyor or a log pile.

Figure 2.1: Log-yard crane [2]

To automate these processes, a system needs ongoing knowledge about the crane and its environment. Based on this information, decisions are made about the control of the sway and the grapple.

[1] https://crane.psiori.com/

Various sensors, such as LiDAR scanners and industrial cameras, are attached to the structure for perception. The information provided enables the perception of the environment in 3D and the position detection of relevant objects. Images from cameras mounted on the trolley are used to determine the location and orientation of these objects. These cameras provide a birds-eye perspective of the grapple and the headblock. While moving the grapple along the crane's boom, the trolley camera moves accordingly and always centres the grapple. Figure 2.2 shows an exemplary image from the trolley camera perspective, which includes the objects of interest: the grapple (here in red), the headblock (here in yellow), and the wood bundles.

Figure 2.2: Sample image for the trolley camera perspective [14]

Based on the camera images, pose estimation of the grapple and the headblock is performed. This is achieved through applying semantic segmentation with U-Net [15] and post-processing steps.

2.2 Vision Transformer

The concept of ViT is a network architecture based on transformer [17] for computer vision that was introduced in 2020 by Dosovitskiy et al. [6]. It was the first scalable and efficient approach to implementing a multihead self-attention mechanism without using convolutional operations. Before the architecture was introduced, Convolutional Neural Networks (CNNs) dominated the vision domain [10], [13].

Inspired by the success of transformer in the Natural Language Processing (NLP) domain [5], it is used in ViT with few modifications to work with image data. An overview

of the ViT architecture is shown in figure 2.3. In NLP, the transformer input is a sequence of token embeddings that is one-dimensional in contrast to two-dimensional images. In order to use this data type as input, an image with resolution (H, W) is divided into a sequence of fixed-size non-overlapping patches. This results in $N = \frac{H \cdot W}{P^2}$ patches with a patch size of (P, P). To obtain the patch embeddings, a trainable linear projection is applied on the flattened patches, ensuring that they match the latent vector size D of the transformer. Additionally, the patch embeddings are enhanced through trainable position embeddings that provide information about the patch position in the image.

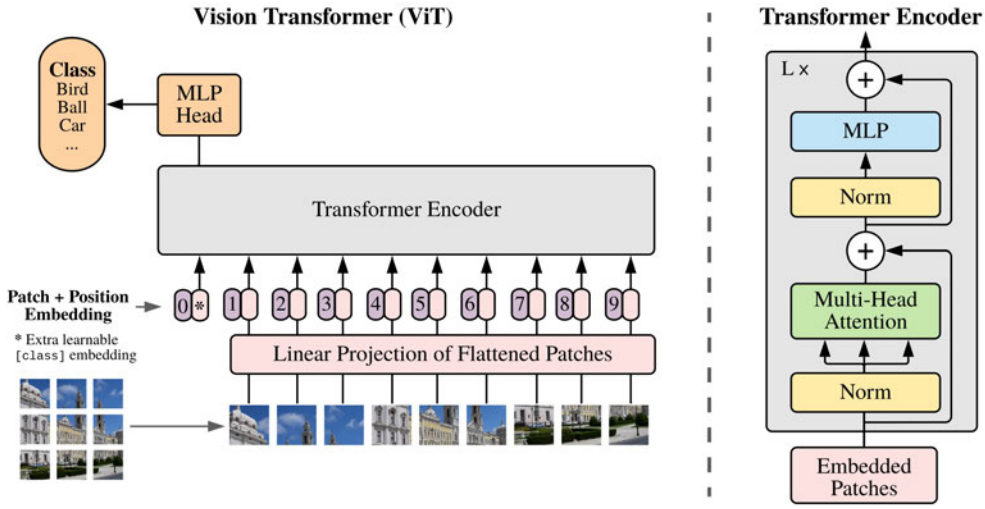


Figure 2.3: Overview of ViT architecture (left) and the transformer encoder (right) [6].

As the standard ViT aims at image classification, an extra learnable embedding is added to the sequence. The class token finally provides an image representation that is useful for classification. All embeddings are then fed into the transformer encoder with L layer. Each layer mainly consists of a multi-head self-attention block followed by a Multi-Layer Perceptron (MLP). The final output is used for the classification task with a MLP head.

The advantage of ViT is the attention mechanism that enables a model to learn long-range dependencies [17]. As ViTs are highly scalable concerning parametric complexity, they require high computational costs to be trained. Additionally, transformer-based networks are not specifically designed for visual data. As a result, they need large amounts of training data to learn data-specific concepts such as e. g. translation invariance [12]. Nevertheless, ViT and architectures based on it are successfully used for various computer vision tasks, e. g. image classification [6], object detection [7] and image segmentation [18].

3 Contributing Paper

Masked Autoencoder: Influence of Self-Supervised Pretraining on Object Segmentation in Industrial Images

Anja Witte[1]

[1]Department of Computer Science, Hamburg University of Applied Sciences, Stiftstraße, Hamburg, 20999, Germany.

Contributing authors: anja.witte@haw-hamburg.de;

Abstract

The amount of labeled data in industrial use cases is limited because the annotation process is time-consuming and costly. As in research, self-supervised pretraining such as Masked Autoencoder (MAE) resulted in training segmentation models with fewer labels, this is also an interesting direction for industry. The reduction of required labels is achieved through the usage of large amounts of unlabeled images for the pretraining that aims to learn image features. This paper analyses the influence of MAE pretraining on the efficiency of label usage for semantic segmentation with U-Net Transformers (UNETR). This is investigated for the use case of log-yard cranes. Additionally, two transfer learning cases with respect to crane type and perspective are considered in the context of label-efficiency. The results show that MAE is successfully applicable to the use case. With respect to the segmentation, an Intersection over Union (IoU) improvement of 3.26 % is reached while using 2000 labels. The strongest positive influence is found for all experiments in the lower label amounts. The highest effect is achieved with transfer learning regarding cranes, where IoU and Recall increase about 4.31 % and 8.58 %, respectively. Further analyses show that improvements result from a better distinction between the background and the segmented crane objects.

Keywords: Masked Autoencoder, Self-Supervised Pretraining, Semantic Segmentation, UNETR, Label-efficiency, Log-yard Cranes

1 Introduction

Recent advances in self-supervised pretraining such as MAE [25] have shown promising results for training computer vision models with less labeled data. This is an important research direction, especially for real-world use cases, as annotations are hardly available and labeling is time-consuming and costly [2]. However, research mainly focuses on pretraining and evaluation with datasets such as ImageNet [49]. On the one hand, these datasets offer a large number of labeled and unlabeled samples that are necessary for training and include diverse object classes. On the other hand, they are in contrast to industrial datasets that are used in real-world scenarios of machine learning models.

Industrial images usually have a lower quality, which is crucial for the accuracy of Machine Learning (ML) models, e. g. for classification or segmentation tasks [11]. In addition, the objects of interest might be smaller than e. g. in ImageNet that has an average object scale of 24.1 % of an image. As a result, details have a higher importance in industrial images.

Moreover, the intention for ImageNet is diversity in terms of object classes, textures and number of instances, among others. This has less relevance in industrial datasets as the images have a lower semantic diversity. It becomes clear that research datasets such as ImageNet are in contrast to industrial images. Accordingly, research results of self-supervised pretraining might not be directly transferable to industrial use cases. Therefore, this paper analyses the applicability of the self-supervised pretraining method MAE for semantic segmentation with UNETR in industrial images received from crane cameras. Since the goal of MAE is reducing the required labeled data for downstream tasks, the focus of analysis is the influence on label-efficiency. This results in the following research questions:

1. Can self-supervised pretraining with MAE increase the label-efficiency of semantic segmentation with UNETR in industrial images?
2. Can self-supervised pretraining with MAE using images of a crane perspective increase the label-efficiency of semantic segmentation with UNETR for another crane perspective?
3. Can self-supervised pretraining with MAE using images of a crane increase the label-efficiency of semantic segmentation with UNETR for another crane?

As [43] concludes that various self-supervised pretraining methods lead to a downstream performance increase with less labeled data, the hypothesis is put forward that MAE increases the label-efficiency of semantic segmentation. In addition, it is assumed that the influence decreases with an increasing amount of labeled samples.

To address the objectives, the paper is structured as follows. Section 2 introduces essential and recent research carried out in the context of the paper. The following section 3 covers the chosen methods, which include an explanation of the used industrial dataset and training pipeline. In addition, implementation details and the experimental setup and evaluation are described. The obtained experimental results are presented and discussed in sections 4 and 5. Finally, a conclusion is drawn in section 6.

2 Related work

The following section covers relevant topics in the context of the paper and presents the current state of the art. Therefore, it is subdivided into semantic segmentation, self-supervised pretraining and transfer learning.

2.1 Semantic segmentation

The task of semantic segmentation refers to an image classification at a pixel level. In order to predict the corresponding label for each pixel, information about the location and the semantics are needed [39]. Recent approaches apply Deep Learning (DL) to achieve semantic segmentation and can be mainly divided into based on Convolutional Neural Network (CNN) or Vision Transformer (ViT) [54].

CNNs are classical architectures in the segmentation domain as they enable high and low-level feature extraction (e. g. [18], [39], [50]). A popular approach is U-Net [48], which follows an encoder-decoder architecture with skip-connections to combine multiple-resolution feature maps. Despite the successful extraction of basic image structures, CNN-based architectures have a limited performance when it comes to global feature relations. Since transformer-based approaches can better learn global semantics, they are increasingly used for segmentation tasks [54].

In the context of computer vision, ViT [14] is the first successful architecture with a pure transformer. The ViT processes an image as splitted fixed-size patches that are linearly embedded and extended with a position embedding. A transformer is then applied to encode the input. Finally, the output is used for the downstream task, e. g. through a classification head [14].

In order to perform segmentation, transformer-based approaches often follow an encoder-decoder structure. In this context, the transformer encoder is the backbone, which represents the feature extractor and is extended through a segmentation-specific decoder. Segmentation Transformer (SETR) [60] proposes progressive upsampling and multi-level feature aggregation as two kinds of CNN-based decoder. SegFormer [56] uses a hierarchical transformer encoder and relies on a lightweight Multi-Layer Perceptron (MLP) as a decoder. In contrast, Segmenter [51] is entirely based on a transformer. Additionally, some approaches integrate U-Net concepts, such as the U-shape and the skip connections. Various architectures enhance the CNN-based encoder-decoder with a transformer encoder following (e. g. TransUNet [9]) or fusing (e. g. TransFuse [59]) with the CNN encoder. Other architectures such as UNETR [22] or nnFormer [61] propose fully ViT-based encoder or encoder-decoder. As UNETR has shown promising results in the context of medical images and in combination with self-supervised pretraining (introduced in the next section) [63], a similar architecture is used for the paper.

The proposed approaches show a successful application of transformer-based segmentation as it is advantageous in modelling long-range relations [16]. However, in order to achieve this performance, large labeled datasets are required, and high computational effort must be expended [29]. To address the challenge of large data amounts, the paradigm of self-supervised pretraining is increasingly used in DL [37].

2.2 Self-supervised pretraining

The goal of self-supervised pretraining is to train a model to learn meaningful representations of unlabeled data. This is achieved through a pretext task that requires predicting or recovering the original input partly or fully. The trained model represents a feature extractor that is afterwards extended by downstream specific layers and finetuned [37].

With regard to the training objective, a distinction can be made between contrastive, generative and adversarial approaches for self-supervised pretraining. Contrastive self-supervised pretraining comprises concepts with the goal of differentiation between instances. Therefore, representation learning focuses on features that help to measure similarity [37]. Some approaches aim to model the relation between the features and the context in a sample, e. g. Pretext-Invariant Representation Learning (PIRL) through solving jigsaw [42]. Other approaches focus on similarities on the instance level, e. g. Momentum Contrast (MoCo) [24] and Simple Framework for Contrastive Learning of Visual Representations (SimCLR) [10] follow the idea of instance discrimination.

As contrastive approaches usually train only an encoder, they are mainly used for discriminative tasks such as classification. In contrast, generative self-supervised pretraining relies on encoder-decoder training with the goal of reconstruction. Common architectures are based on auto-regressive, flow-based or autoencoding models [37]. Various recent generative approaches can be categorised as masked image modelling inspired by generative Natural Language Processing (NLP) pretraining methods. These pretext tasks are based on reconstructing masked patches of an input image to learn the general image pattern [44]. On the one hand, some approaches, such as BERT Pre-Training of Image Transformers (BEiT) [7], Masked Feature Prediction (MaskFeat) [55] or Perceptual Codebook for BERT Pre-training of Vision Transformers (PeCo) [13] aim to reconstruct the masked patches as tokens. On the other hand, the direct reconstruction of the original pixel values can be targeted. Proposed methods are Simple Framework for Masked Image Modeling (SimMIM) [57] and MAE [25].

Adversarial methods combine ideas of contrastive and generative methods in the way that, on the one hand, an encoder-decoder architecture is trained to generate fake images. On the other hand, a discriminator is used to differentiate between the fake and the real samples. Approaches either aim at the generation of an image (e. g. adversarial autoencoder [41]) or at recovering parts of an image (e. g. Image Super-Resolution Using a Generative Adversarial Network (SRGAN) [34]).

Recently, the aforementioned generative approach of MAE has attracted attention in research (e. g. [26], [32], [58]) as it is simple and effective. Moreover, [63] shows that MAE can be successfully applied as pretraining for UNETR. Therefore, it is chosen as the self-supervised pretraining method for the paper.

In addition, some papers analyse the influence of self-supervised pretraining on downstream tasks. The authors in [33] show that contrastive methods, e. g. MoCo [24] and PIRL [42], result in a better downstream performance than with supervised pretraining. Moreover, the encoder performance is better in the case of similar datasets for pretraining and finetuning [33]. In addition, [43] shows that self-supervised pretraining with Variational Autoencoder (VAE) [31], Rotation [17], Contrastive Multiview Coding (CMC) [52] or Augmented Multiscale Deep InfoMax (AMDIM) [6] leads to a lower

demand of labeled data to achieve a similar performance as trained from scratch. The difference is particularly high for scenarios with fewer labels and decreases with an increasing number of labels [43]. In [12], similar findings are presented but with a focus on the influence of the contrastive method SimCLR [10] on image classification. Moreover, results are validated with MoCo [24] and Bootstrap Your Own Latent (BYOL) [19] as well [12]. No paper specifically analyses the influence of generative self-supervised pretraining on label-efficiency.

In the case of generative methods, the authors of [15] evaluate the influence of the pretraining data amount on classification performance. The paper shows that masked image modelling methods such as BEiT [7] are less influenced than supervised pretraining with Self-Distillation with no Labels (DINO) [8]. In addition, [4] analyses label-efficiency for few-shot learning. Using 1, 2 or 5 images per class, image BERT Pre-Training with Online Tokenizer (iBOT) [62], DINO [8] and Masked Siamese Networks (MSN) [4] outperform MAE [25] [4].

Within research, most papers focus on the downstream performance and label-efficiency evaluation of contrastive methods. Few papers include generative approaches, but none address label-efficiency without few-shot. Therefore, this paper evaluates the label-efficiency of generative self-supervised pretraining with the example of MAE.

2.3 Transfer learning

Self-supervised pretraining can be performed with data within or outside the downstream task domain. In both cases, the knowledge learned in the pretraining phase is transferred to the downstream model to enhance performance. Transfer learning is the general idea of using previously learned knowledge from a source domain for a target domain [21].

For a successful application, some relation must be available between the domains or tasks. According to the learning setting, the concept of transfer learning can be divided into unsupervised, transductive and inductive methods. Unsupervised approaches are contrary to the classical machine learning setup. It requires differences in the domains and tasks, but both have to be related for source and target. Additionally, no labels are available.

In contrast, transductive methods use the same task for different domains, whereas source domain labels are needed. The opposite is true for the domains and tasks of the inductive learning setting. It only demands the same domains and at minimum target domain labels [45]. As self-supervised pretraining is based on a pretext task that differs from the downstream task, it can be categorised as inductive transfer learning. Therefore, self-supervised pretraining requires common knowledge between pretraining and finetuning data domains.

In order to transfer knowledge, [27] categorises various methods in the context of DL as either adversarial or network-based. The first relies on enhancing feature extraction in an adversarial manner, e. g. with Conditional Generative Adversarial Network (CGAN) [38]. The latter relates to approaches that retrain a model. Most frequently, the whole model, including the pretrained layers, is trained again, referred to as finetuning (e. g. in [3]). As it can lead to forgetting the learned features, partial

finetuning is conducted. All feature-extracting layers are frozen and downstream task-related layers are retrained (e. g. [5]). Moreover, progressive learning is possible where the pretrained model is partly or not retrained, but some layers are added and trained again (e. g. [20]) [27].

In [35], the authors show that self-supervised pretraining with ImageNet data [49] leads to an increase in performance for object detection and instance segmentation using COCO [36]. They compare generative (MAE [25] and BEiT [7]) and contrastive (MoCo [24]) methods with supervised pretraining. As a result, generative approaches outperform as a backbone for a Mask R-CNN [23] [35]. As using different datasets for pretraining and finetuning can positively influence downstream performance, it is additionally analysed in the paper.

3 Methods

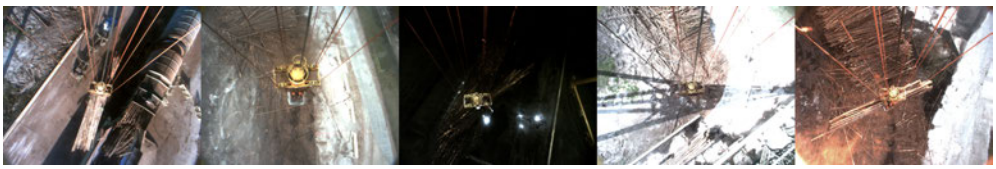
The methods used to conduct the experiments for this paper are addressed in the following section. To evaluate the label-efficiency of MAE with an industrial scenario, the used dataset with its classes is described. Additionally, the section focuses on the experiments' training pipeline and the corresponding implementation details. Finally, the different experimental setups that use the training pipeline are explained and experimental evaluation metrics and methods are introduced.

3.1 Datasets

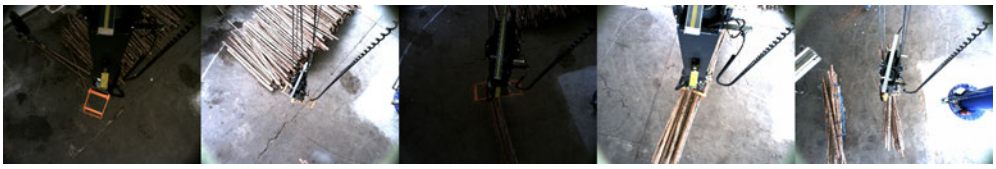
In order to validate the label-efficiency of MAE with an industrial use case, images from cameras placed on two cranes are used. One crane is located in a hall (minicrane), and the other one is outside in Palatka (palatka-crane). As the minicrane is indoors, weather conditions such as rain and snow are not present in the images in contrast to the palatka-crane. The used cameras are mounted on the trolley and show the grapple and the headblock from above. Figure 1 shows exemplary images of the different cranes. For the palatka-crane, there are two different trolley cameras. The difference between them is the perspective, as one is located on the left and the other on the right side of the trolley. Each image initially has a size of 1024 x 1280 pixels but is resized to 128 x 160 pixels to reduce the computational costs of the training. In total, there are 20,000 labeled palatka-crane and 5,000 minicrane images.

For all images, the segmentation classes of interest are the background (class 0), the grapple (class 1) and the headblock (class 2). Figure 2 shows an exemplary segmentation for a minicrane image where classes 1 and 2 are coloured red and blue, respectively.

For all images, the classes 1 and 2 overlap whereas the headblock is located above the grapple and therefore covers parts of the grapple. Additionally, the grapple is always located around the image centre and is shown in different rotations. Both classes can vary in size depending on their distance from the camera. Figure 3 shows the distribution of the relevant classes over the images for both cranes. For the palatka-crane, there are two highly probable positions for the grapple and the headlock, which result from the two different trolley cameras (left and right).



(a) Palatka-crane



(b) Minicrane

Fig. 1: Exemplary images of the used datasets.

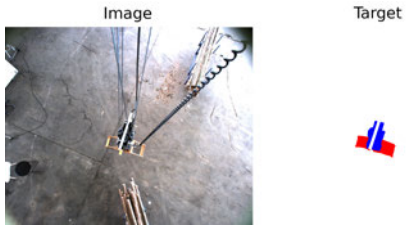


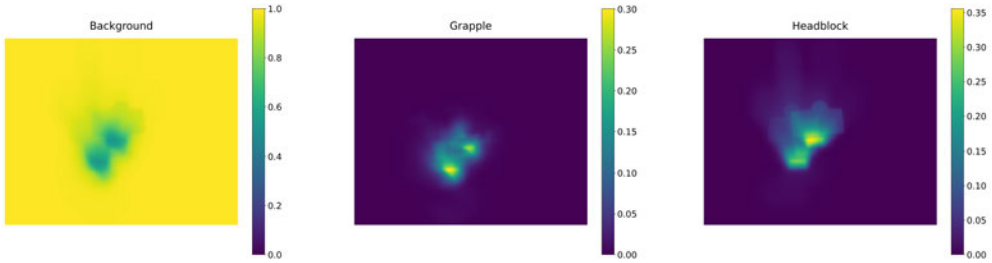
Fig. 2: Exemplary segmentation for a minicrane image. Classes of interest are the grapple (red) and the headblock (blue).

3.2 Training pipeline

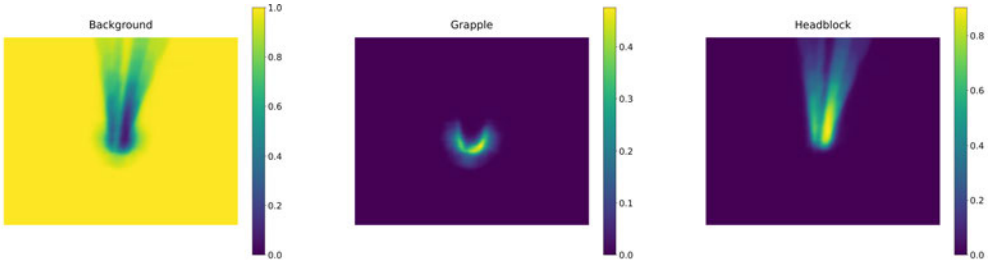
The experiments of this paper are based on a two-step pipeline comprising pretraining and finetuning. The architecture is shown in figure 4 and follows the setup in [63].

The self-supervised pretraining method MAE [25] is applied in the first step. The backbone architecture consists of a ViT-based encoder trained as a feature extractor. For the MAE pretraining, each input image is separated into fixed-size non-overlapping patches. Afterwards, a certain amount of patches is randomly removed. The MAE encoder linearly embeds the remaining patches and enhances them with a positional embedding. Finally, transformer blocks are used to process the embeddings. The resulting tokens are combined with masked tokens for the removed patches to be used as MAE decoder input. With transformer blocks, the MAE decoder conducts the reconstruction task and is trained to predict the pixel values of the masked patches.

After training the encoder-decoder architecture, the MAE encoder represents a feature extractor. To use this knowledge for the downstream task of semantic segmentation, the encoder weights are extracted and transferred to a UNETR [22]. The UNETR is inspired by the concepts of U-Net [48] and is therefore a "U-shaped" encoder-decoder architecture that integrates skip-connections. In contrast to U-Net, it



(a) Palatka-crane



(b) Minicrane

Fig. 3: Class distribution in datasets. Figures show the probability of each image pixel belonging to a specific class in the Palatka-crane (3a) or Minicrane (3b) dataset.

uses transformer blocks in the encoder. The input images are divided into patches and embedded as in MAE, but no patches are removed. The following decoder consists of convolutional and deconvolutional blocks. Accordingly, only the UNETR encoder and the prior patch embedding layer are initialised with the pretrained MAE weights. The UNETR decoder is randomly initialised and outputs the predicted segmentation.

3.3 Implementation details

All implementations are done using TensorFlow [1]. For all experiments, ten-fold cross-validation is used.

For self-supervised pretraining, the MAE architecture[1] is implemented with a ViT as a backbone model. As previous experiments with the dataset have shown that large models tend to overfit, a lightweight version of ViT as used in the implementation of Keras [28] is chosen. Accordingly, the encoder has a projection dimension of 128, 4 transformer blocks and 6 layers. As stated in [25], the decoder can be lightweight and has therefore a projection dimension of 64 and 4 transformer blocks with 2 layers. Additionally, the final dense layer of the decoder is replaced by a series of convolutional and deconvolutional operations to save computational costs.

The random masking of patches results in varying training samples and can therefore be seen as data augmentation. As the authors in [25] found that additional data

[1]Implementation is based on [28].

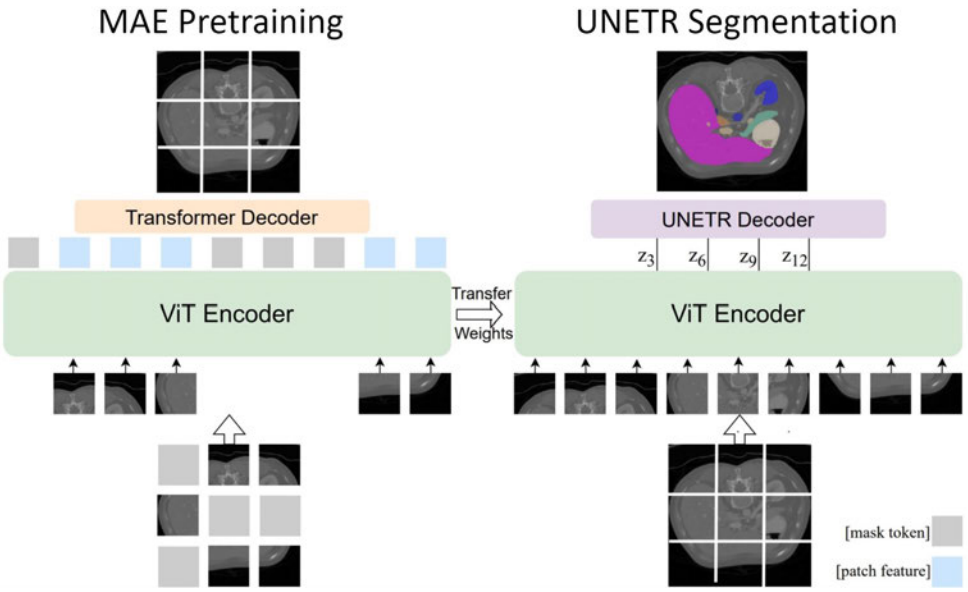


Fig. 4: Training Pipeline for experiments. At first, a MAE is trained self-supervised with unlabeled data (pretraining). Afterwards, the encoder weights are transferred to a UNETR. Finally, the UNETR is trained supervised with labeled data (finetuning). (Figure adapted from [63])

augmentation steps have little impact, the used architecture applies only random horizontal flipping.

Moreover, [53] analyses the importance of the patch size for ViT-training with DINO. As smaller patch sizes increase performance, a patch size of 8 is chosen. According to the results in [25], the masking ratio is set to 75 %.

A batch size of 256 is chosen for training purposes, and early stopping is applied. Additionally, the optimizer AdamW [40] with weight decay of 1e-4 is used. The warm-up epoch percentage is 15 % with a cosine decay learning rate schedule. The base learning rate is 1e-3. To measure the reconstruction loss between the original and reconstructed image and as loss function, the Mean Squared Error (MSE) is used.

The semantic segmentation model follows the architecture of UNETR[2] with adaptions to work with two-dimensional images as UNETR was designed for three dimensions. Similarly to MAE, the model capacity is reduced by decreasing the encoder width to 128 with 4 transformer blocks and 6 layers.

The patch size is chosen according to MAE and is 8. In contrast to MAE, data augmentation is applied as it has proven useful in previous segmentation training with U-Net and the crane data. Therefore, the images are randomly scaled by a factor of 1 - 1.2. The brightness (delta=0.4), contrast (range=[1, 1.2]) and saturation (range=[0.6, 1.4]) are augmented as well.

[2]Implementation is based on [47].

The training is performed for a maximum of 100 epochs with Adam optimiser [30] and a learning rate of 1e-3. A batch size of 64 is chosen. The used loss function is Sparse Cross-entropy, which is adapted to handle class weights. From experience with the crane datasets, segmentation results improve through using a higher class weight for the grapple than for the background and the headblock. Accordingly, the class weight of the grapple is set to 4, whereas the other classes are weighted to 1.

3.4 Experimental setup

In order to analyse the influence of self-supervised pretraining with MAE on semantic segmentation, different setups are chosen, which are shown in table 1. All pretraining experiments are conducted with 20,000 unlabeled samples. Moreover, the number of labeled samples for UNETR-training is step-wise reduced from 100 % to 50 %, 10 % and 1 % of the available images. For label-efficiency (LE) analysis, only palatka-crane data from both perspectives is used. To investigate the label-efficiency with transfer learning between perspectives (TL (p)), two separate datasets for palatka trolley-left and trolley-right are taken as training input of MAE and UNETR. The transfer learning between cranes (TL (c)) is conducted through the usage of the full palatka-crane dataset for MAE and a labeled minicrane dataset. The results are compared with experiments without pretraining to evaluate the influence of MAE on label-efficiency. For these experiments, the setup follows the same as in table 1.

Table 1: Experimental setups. Objective refers to the three research topics: label-efficiency (LE), transfer learning for perspectives (TL (p)) and for cranes (TL (c)). The used datasets are from the palatka-crane (P) or minicrane (M). In order to analyse the label-efficiency of each objective, different amounts (100 %, 50 %, 10 %, 1 %) of labeled samples are used.

Objective	MAE			UNETR		
	Dataset	Perspective	Num. images	Dataset	Perspective	Num. images
LE	P	trolley	20,000	P	trolley	20,000
	P	trolley	20,000	P	trolley	10,000
	P	trolley	20,000	P	trolley	2,000
	P	trolley	20,000	P	trolley	200
TL (p)	P	trolley-left	20,000	P	trolley-right	5,000
	P	trolley-left	20,000	P	trolley-right	2,500
	P	trolley-left	20,000	P	trolley-right	500
	P	trolley-left	20,000	P	trolley-right	50
TL (c)	P	trolley	20,000	M	trolley	4,400
	P	trolley	20,000	M	trolley	2,200
	P	trolley	20,000	M	trolley	440
	P	trolley	20,000	M	trolley	44

3.5 Experimental evaluation

Different metrics and methods are used to evaluate the described experiments for MAE and UNETR. As the goal of MAE is to reconstruct corrupted images, a visual analysis of exemplary reconstructions is conducted. These images provide insights about the performance regarding the level of detail and the different settings, e. g. light situations. Nevertheless, the reconstructions are highly dependent on the MAE decoder, which is discarded after pretraining. Therefore, the final interest is the encoder, as it serves afterwards as a feature extractor. The corresponding performance is indirectly evaluated by analysing the UNETR results.

The evaluation of UNETR is performed through the counting metrics Recall and IoU, which are calculated as the mean value of the folds for an experiment. Moreover, a deeper analysis of specific experiments with class-wise metrics calculation is performed. To investigate the influence on label-efficiency, the numerical difference between experiments with and without pretraining is relevant. Therefore, the metric delta for each fold and the mean value of the folds are used.

The Recall measures the amount of correctly identified positives [46]:

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

As this can result in misleading information about the performance with imbalanced datasets, IoU is used as another metric. It measures the overlap between the predictions A and the targets B [46]:

$$IoU = \frac{|\ A \cap B\ |}{|\ A \cup B\ |}$$

As Recall and IoU are both single-threshold metrics, the Receiver Operating Characteristic (ROC) curve and the Area under the Receiver Operating Characteristic curve (AUROC) are used as threshold-invariant metrics for deeper analysis. With these, the performance of distinguishing between classes can be measured even for imbalanced datasets. As it is used for binary problems, it is calculated in the paper using either one class vs the others or discarding the background class [46].

In addition, the UNETR segmentations are visually evaluated similarly to MAE. A comparison between the semantic segmentations for experiments with and without pretraining can be helpful in gaining insights into the strengths and weaknesses.

4 Results

The following section presents the results from the experimental setups described before. The section is subdivided into the evaluation of MAE and label-efficiency, using either the same dataset or different datasets (transfer learning) for pretraining and finetuning.

4.1 MAE reconstructions

In order to investigate the applicability of MAE for autonomous cranes, figure 5 shows the MAE training curve for fold 10. The training curves for the other folds are shown in the appendix A. From the loss curves, it becomes clear that the training and validation

losses are decreasing during the training and converge. Similar results are found for the other folds. This shows that the MAE model effectively learns from the training data.

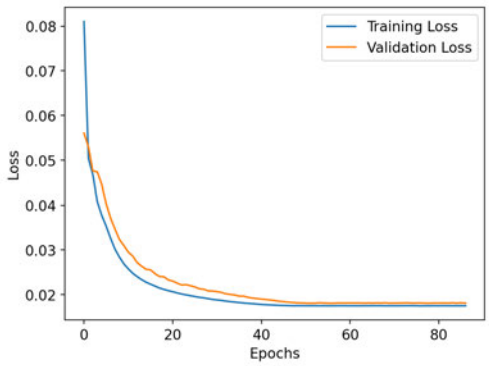


Fig. 5: Training curve for MAE training (fold 10).

As the MAE encoder is used as the backbone model for the UNETR, the learned features are of interest. To gain insight, figure 6 shows different input images of the palatka-crane dataset with example maskings and the corresponding reconstructions. The examples suggest that the MAE learns to reconstruct the basic image structures. Besides that, a low level of detail reconstruction is found in the first two images and the crane classes are barely visible. In the third example image, the grapple and the headblock are larger compared to the image size and are reconstructed in more detail.

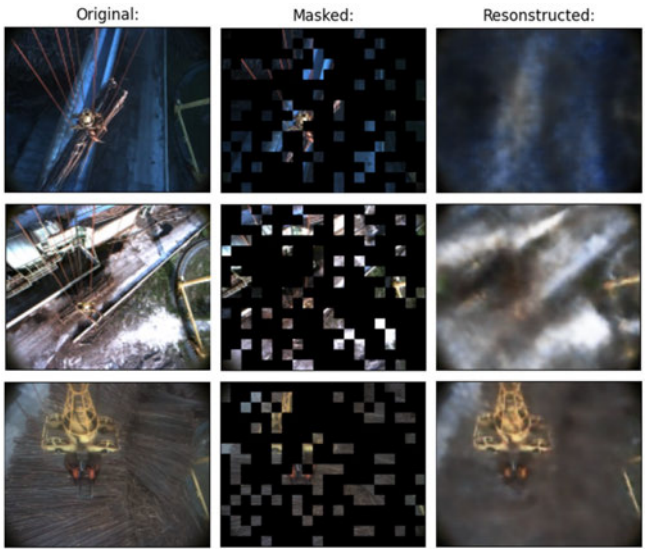


Fig. 6: Reconstruction results using MAE.

4.2 Label-efficiency

To evaluate the influence of MAE pretraining on label-efficiency, figure 7 compares it with only supervised training of UNETR for different label amounts. For both experimental setups, IoU and Recall are reported as mean values of the folds. Additionally, the lower diagrams present the delta between the two setups for each fold and the mean value thereof.

The results show that self-supervised pretraining offers a performance increase starting from 2000 labels. In the case of 200 labeled samples, pretraining results in a decrease of 8.34 % for IoU and 11.18 % for Recall [3]. Moreover, the dispersion of the delta values between the individual folds is greater than for other label amounts. The strongest positive influence is found with 2000 labeled images and is about 3 % for both metrics. With an increasing number of labels, the segmentation performance increases, but the delta values decrease.

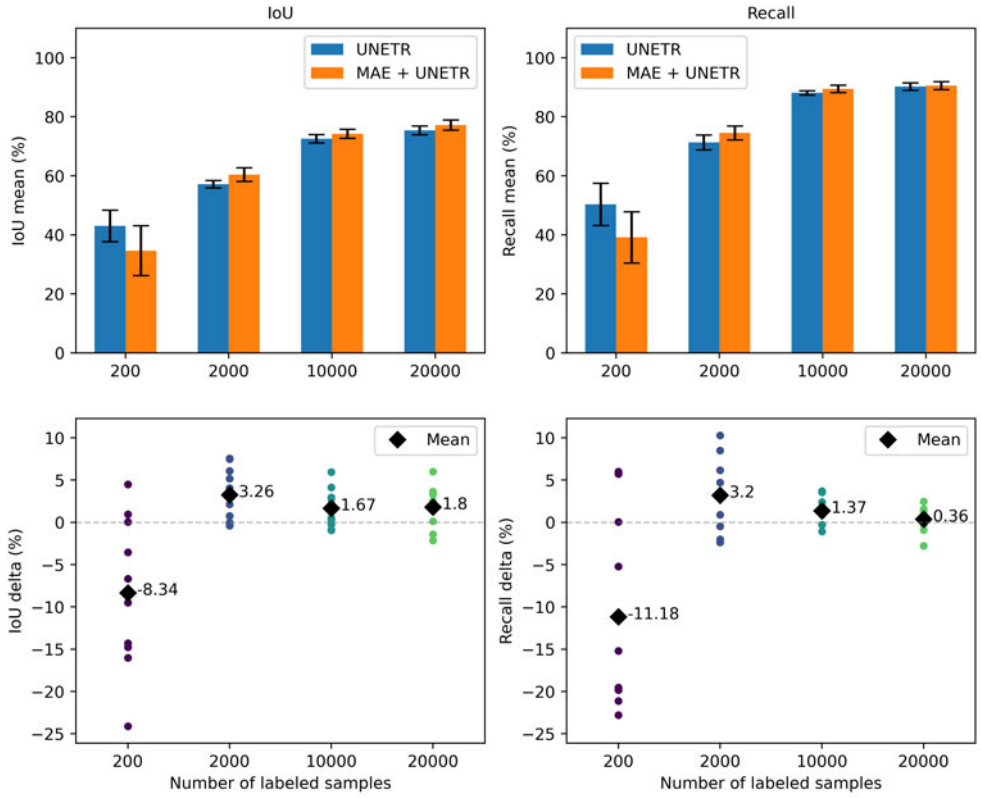
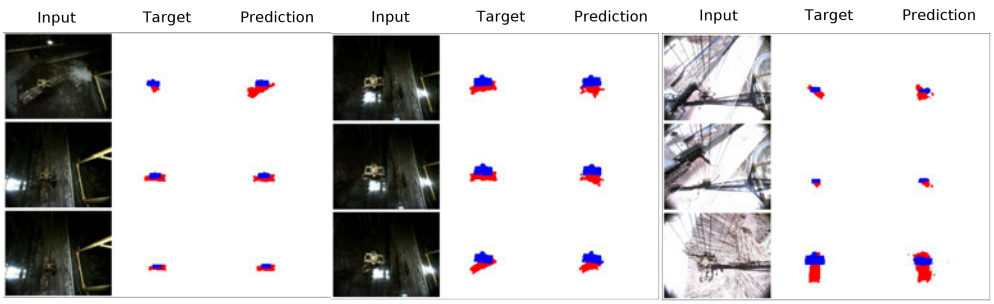


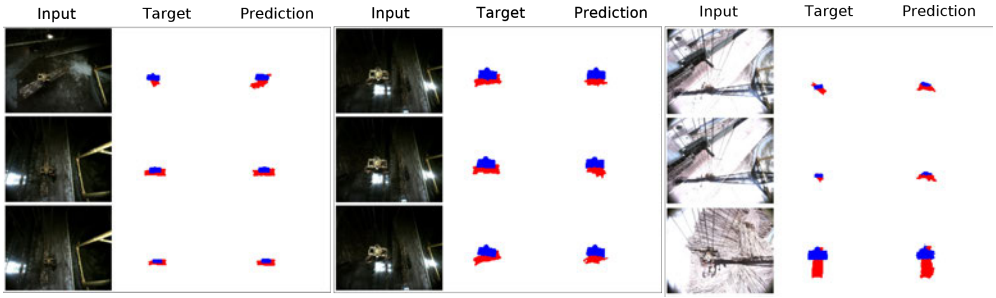
Fig. 7: Label-Efficiency. Results report mean IoU and Recall for different numbers of labeled samples for finetuning. The delta between the results of training UNETR with and without MAE-pretraining before are shown in the lower figure.

[3]This result is contrary to the expectations. A discussion can be found in appendix B.

In addition, figure 8 shows some exemplary UNETR segmentations with and without MAE pretraining. The segmentations result from models that were trained with 20,000 labeled images. As the previously mentioned metrics deltas demonstrated, the visual segmentation predictions are similar for both setups. Furthermore, figure 7 showed a higher delta for the IoU (1.8 %) than for the Recall (0.36 %), which suggests that MAE pretraining leads to less over-segmentation. This result is also visible in figure 8, e. g in the example in the top left corner.



(a) UNETR



(b) MAE + UNETR

Fig. 8: Exemplary UNETR segmentations without (8a) and with pretraining (8b) for palatka-crane dataset.

As MAE pretraining positively influences the semantic segmentation of the crane dataset, the following section investigates two experimental setups based on it. For both cases, the analyses focus on the additional influence of transfer learning on label-efficiency.

4.3 Label-efficiency with transfer learning

With regard to the label-efficiency of MAE and transfer learning, two different experimental setups are presented in the following. First, the results of knowledge transfer across different crane perspectives are shown. Then, transfer learning for different crane types is addressed. Furthermore, some results of the latter are analysed in more detail.

4.3.1 Transfer learning between crane perspectives

To transfer knowledge between crane perspectives, the MAE pretraining was conducted with data from the left trolley perspective, whereas segmentation training is based on the right trolley perspective. The performance is compared with supervised UNETR training for the right perspective in figure 9.

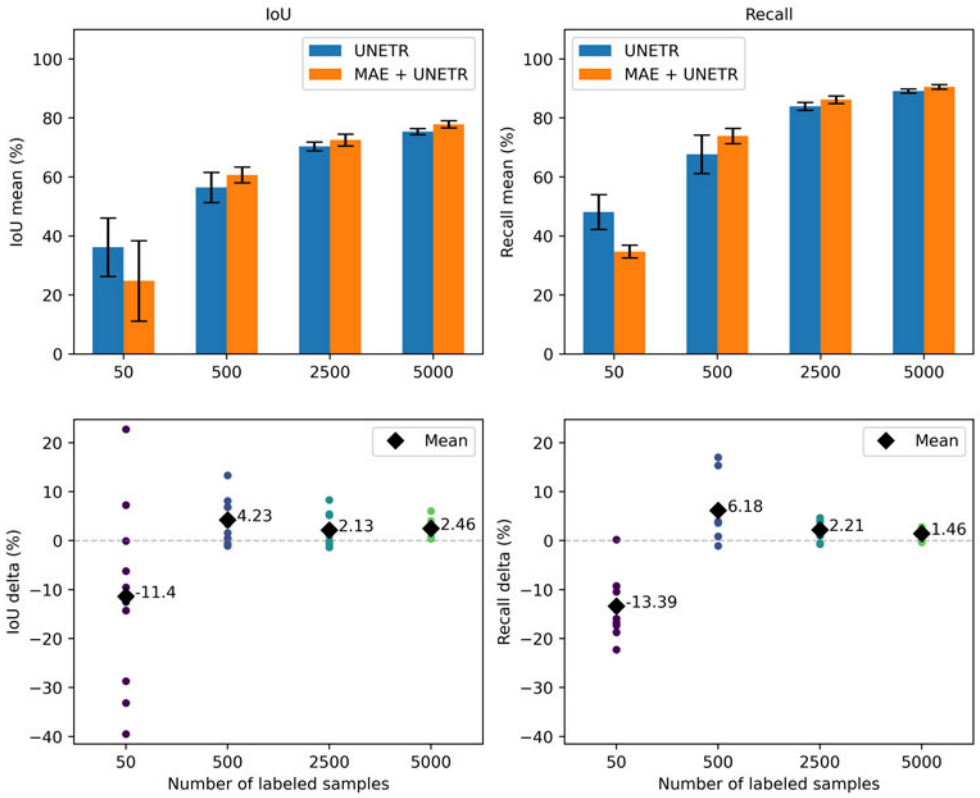


Fig. 9: Transfer Learning (perspectives). Results report mean IoU and Recall for different numbers of labeled samples for finetuning. The delta between the results of training UNETR with and without MAE-pretraining before are shown in the lower figure.

The comparison shows that the performance of both setups increases with the number of labeled samples. In addition, UNETR achieves higher IoU (+11.4 %) and Recall (+13.39 %) results on 50 labeled samples without pretraining [4]. Regarding IoU, results vary for individual folds more than for other experimental setups. Results show an improvement with pretraining for 500 up to 5000 labeled samples. The highest

[4]Similarly to the previous section, this is an unexpected result. Appendix B provides further information for the previous case, which might impact this result.

performance increase is seen for both metrics at 500 labels. In this case, IoU improves on average by 4.23 % and Recall by 6.18 %. Using more labeled data for UNETR training results on average in lower increases between 1.4 % and 2.5 % for both metrics. Despite less improvement, the results are more consistent for the individual folds with increasing labels.

4.3.2 Transfer learning between cranes

As the palatka-crane and the minicrane share similar objects shown on the trolley camera images, transfer learning is analysed. Figure 10 shows the corresponding comparison for the semantic segmentation performance with and without pretraining.

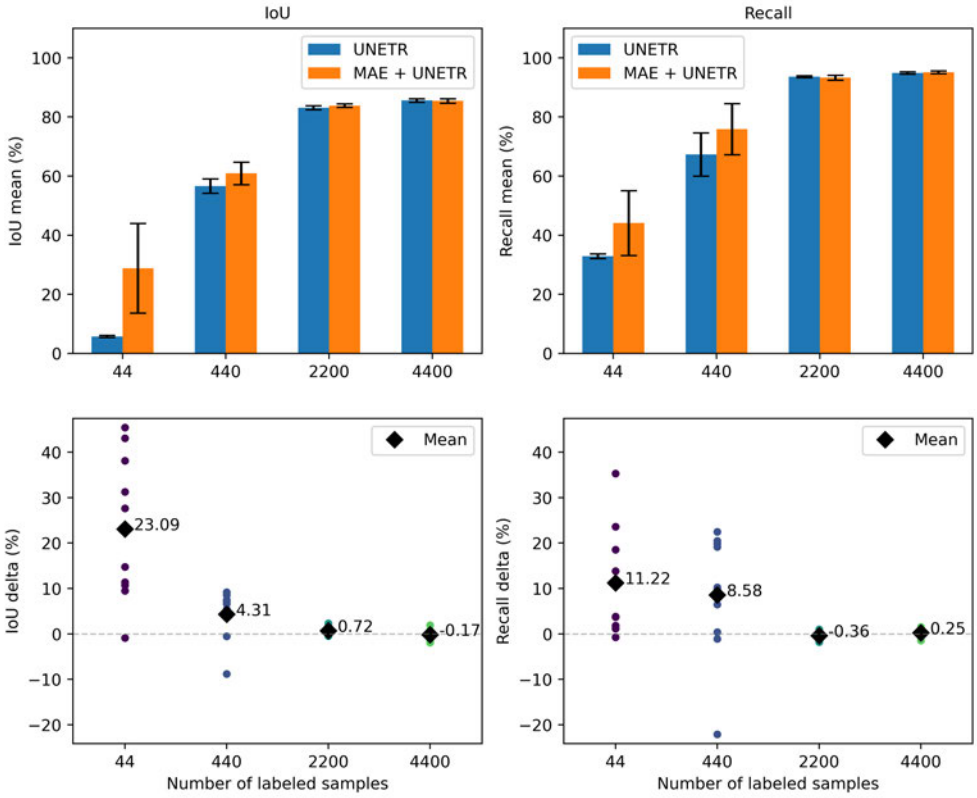


Fig. 10: Transfer Learning (cranes). Results report mean IoU and Recall for different numbers of labeled samples for finetuning. The delta between the results of training UNETR with and without MAE-pretraining before are shown in the lower figure.

Concerning the mean IoU, the pretraining results in an increasing improvement as the number of labels decreases. In the case of 4400 labeled samples, the pretraining on palatka data shows on average a negative influence (−0.17 %) on the final segmentation

performance. The lowest number of labeled samples (44) leads to the most significant improvement of 23.09 % [5]. Nevertheless, the corresponding delta values for individual folds are highly dispersed because of varying UNETR performances when using pretraining.

The second highest positive influence on IoU is 4.31 % on average and results from 440 labeled images, whereas the metrics are less dispersed than for 44 labels. The results are similar for the Recall. The main difference is that pretraining leads to a negative influence of −0.36 % for 2200 labels and with 4400 labels, the Recall slightly improves (0.25 %) with pretraining. Similar to the previous transfer learning experiment, the setups with more labels lead to more consistent results across the individual folds.

In addition, the results show that for 440 labels, the IoU delta is about half of the Recall delta. This means that while pretraining leads to more correctly identified class pixels, it also causes more over-segmentation as the overlap between the predicted and true segmentation is not rising in the same ratio.

4.4 Further analysis

The experimental setup with the highest improvement is analysed in more detail in the following to gain a deeper understanding of the influence of MAE pretraining on UNETR. Concerning the mean values of the metrics, transfer learning between cranes with 44 labels showed the best performance. Nevertheless, the setup with 440 labels (second best result) is chosen for further analysis as it shows less dispersion for individual folds.

Previous result figures focused on the average IoU and Recall across folds and different segmentation classes. One of the classes is class 0, which represents the background. Since the background covers a large part of the images to be segmented, it highly influences the mean metrics. Therefore, figure 11 shows the mean IoU and Recall separately for the classes.

It is visible that the background class achieves the highest IoU (about 97 %) and Recall (about 98 %) values on average for both setups. The second best performance is found for the headblock class for which UNETR including pretraining leads to an IoU of 61.17 % and a Recall of 70.05 %. Both metrics are increased compared to UNETR results without pretraining. The delta improves up to 5.87 % for IoU and 9.17 % for Recall. The lowest mean IoU is found for the grapple class. Here, the mean IoU is 23.50 % with pretraining which represents an improvement of 6.84 %. Similarly, pretraining leads to a Recall increase of 16.79 % to 58.8 % on average.

The result that the Recall for the grapple and headblock is much higher than IoU leads to the conclusion that while the model identifies more true positives, it also tends to produce false negatives and therefore over-segmentation. As the IoU delta is lower than the Recall delta, the pretaining leads on average to a higher gap between IoU and Recall and more false positives.

As the mean metrics for the grapple and headblock classes are lower than for the background class, the corresponding false negatives are of interest to gain insights

[5] The strong positive influence of MAE pretraining might be impacted by the issues discussed for the previous experiments with the lowest label amount. Readers are referred to appendix B.

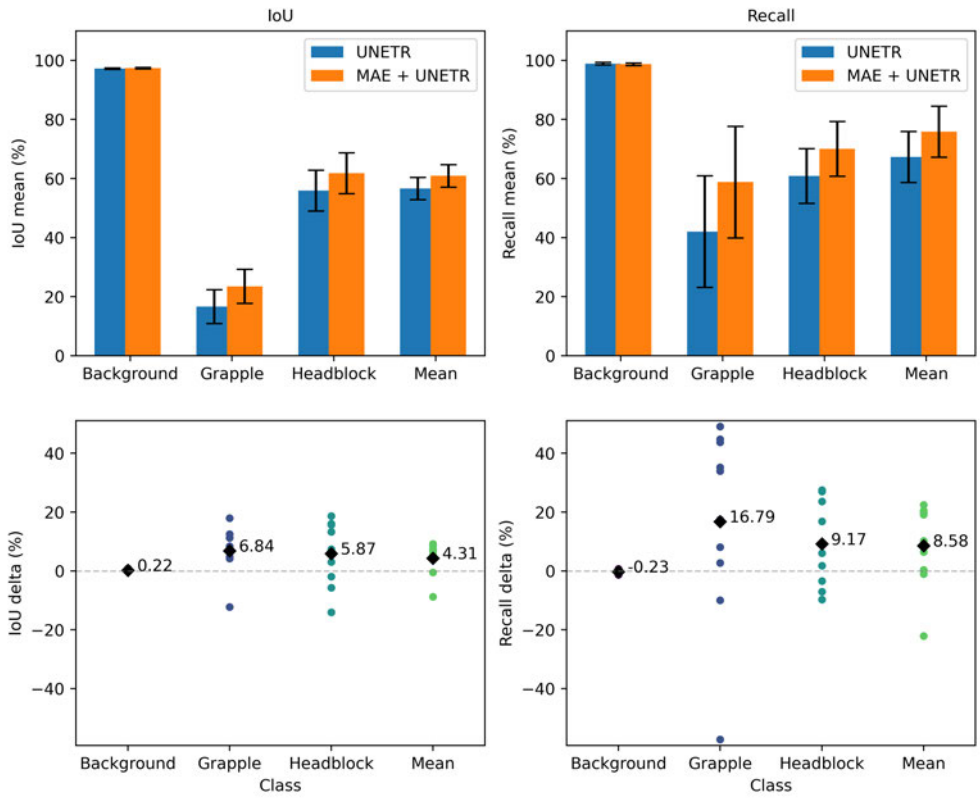


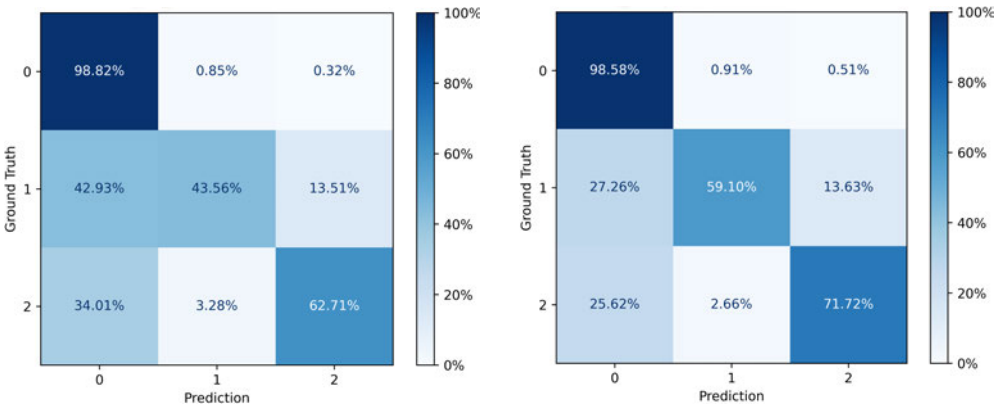
Fig. 11: Transfer Learning (cranes) with 440 labeled samples. Results report mean IoU and Recall for the segmentation classes. The delta between the results of training UNETR with and without MAE-pretraining before are shown in the lower figure.

into the model performance. Therefore, figure 12 shows the confusion matrices for the segmentation results of both setups.

Without self-supervised pretraining (figure 12a), most of the false negative predictions for class 1 (42.93 %) and 2 (34.01 %) are found in the background class. This demonstrates that the UNETR tends to classify the two crane objects as background and is partly not able to separate the classes.

Moreover, 13.51 % of positive class 1 samples are incorrectly predicted as class 2. The other way round, the UNETR predicts 3.28 % of class 2 as class 1. This result shows that the headblock is better differentiable from the grapple than the other way around. Overall, the trained model reaches a higher performance for the headblock than for the grapple.

In comparison, pretraining (figure 12b) results in differences with respect to the false positives in class 0. Both decrease about 15 % and 9 % for class 1 and class 2, respectively. Since at the same time, the other false negatives increase at a maximum of about 0.2 % or slightly decrease, the ratio of true positives increases for classes 1



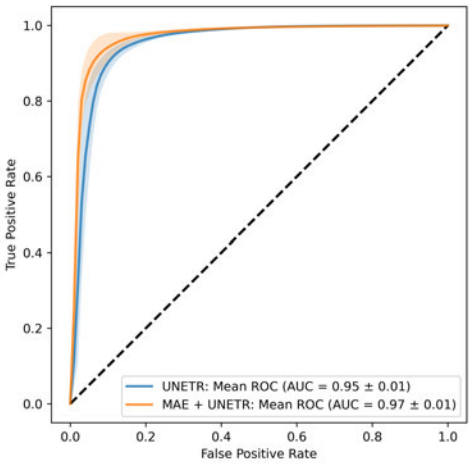
(a) UNETR (b) MAE + UNETR

Fig. 12: Confusion Matrices.

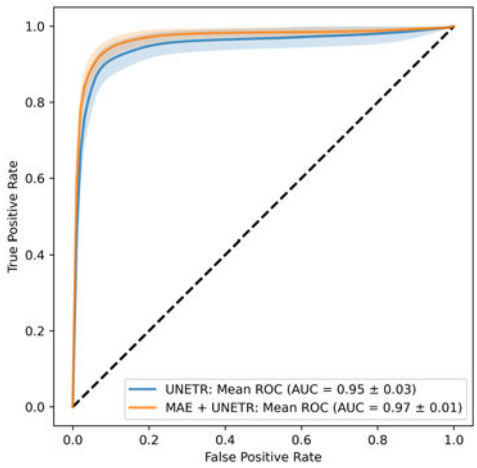
and 2. The highest increase of true positives is achieved for class 1 with a delta of 15.54 %. The comparison suggests that the MAE has learned different image features as e. g. the crane objects and is therefore helpful as pretraining for the UNETR to distinguish between the background and the crane objects. In particular, the grapple segmentations are improved.

In addition to the confusion matrices, the ROC curves in figure 13 and 14 put the true positive rate in relation to the false positive rate for different thresholds. For figure 13, the curves are computed for one class vs the others. The curve and the corresponding AUROC results show that the ability of the model to distinguish between the individual classes improves with MAE pretraining. The AUC increases for the background and the grapple class by about 0.02 and for the headblock class by about 0.01 with pretraining. The results show that the UNETR with and without pretraining achieves the best performance for the headblock class. The headblock is the most straightforward to distinguish from the other classes.

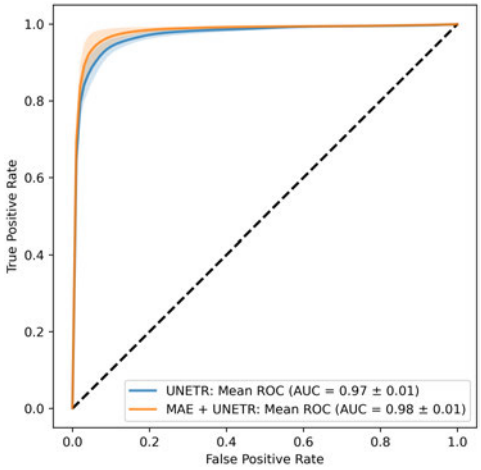
In contrast to figure 13, the background class is discarded for figure 14, and the curves only compare classes 1 and 2. For both comparisons, the pretraining improves the ROC curve and the resulting AUROC. Moreover, it is visible that the stronger positive influence is found for the grapple class in figure 14a. The AUROC increases on average by 0.06 whereas the comparison in figure 14b shows an AUROC improvement of 0.02. This leads to the result that the UNETR is on average more confident in distinguishing the grapple from the headblock than the other way around. Moreover, the confidence increases with pretraining.



(a) Class 0 (Background) vs other classes

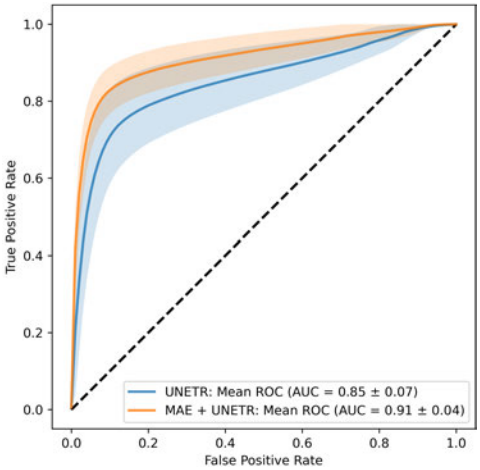


(b) Class 1 (Grapple) vs other classes

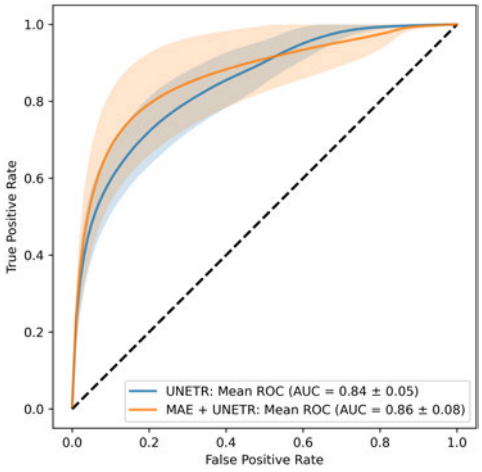


(c) Class 2 (Headblock) vs other classes

Fig. 13: ROC curves. Results report each class vs the other two classes combined.



(a) Class 1 (Grapple) vs 2 (Headblock)



(b) Class 2 (Headblock) vs 1 (Grapple)

Fig. 14: ROC curves. Results report ROC for the classes Grapple and Headblock. Background class is excluded.

5 Discussion

The conducted experiments showed that MAE is applicable for the crane dataset. The analysis of the reconstructions resulted in varying levels of reconstruction detail degree, which depends on e. g. the object size. This result is reasonable since in some cases the grapple and the headblock have a similar size as the patch size. Depending on the random placement of masked patches, the objects of interest are either no longer or only with a few parts visible. As a result, the MAE has too little information for a reasonable reconstruction. Similar results can be found in the original MAE paper [25].

Furthermore, the final reconstructions highly depend on the decoder adapted for this paper. Therefore, the reconstruction performance might be lower. Nevertheless, the feature extraction capability of the MAE encoder is more relevant as it is used in the UNETR model.

The different label-efficiency experiments resulted mainly in similar findings. Independent of the pretraining usage, the segmentation performance increases with an increasing number of labeled samples for the UNETR training. The reason is that more training data offers the model more information about the underlying features and patterns. This enables the model to generalise well.

With the usage of MAE, the label-efficiency increases and the UNETR performs better. The influence of MAE decreases with increasing number of labels. As expected, the strongest influence occurs with fewer labeled samples. In the case of large training datasets for UNETR, the model has enough information to learn helpful image features by itself and without pretraining. Therefore, pretraining has no or little influence on performance here. Using MAE might speed up UNETR training since the feature extraction has not been learned again. As this is out of the paper's focus, it is not analysed here.

The transfer learning experiments concerning the camera perspective showed a positive influence of MAE on label-efficiency. This indicates that the MAE learned the features of the grapple and the headblock independent of its position. Therefore, the learned features from the left perspective are transferable to the right perspective and cause an improvement in the segmentation performance.

Similarly, results from transfer learning between cranes demonstrate that the palatka-crane features are useful for segmenting minicrane images. This is particularly interesting as the corresponding grapples and headblocks differ in shape and colour. Nevertheless, the pretrained features improve the segmentation result, especially with less labelled data.

In addition, the more detailed analysis showed that the weakness of UNETR segmentation is the distinction between the crane parts and the background. MAE pretraining mainly improved the grapple and headblock segmentation performance by reducing this weakness.

Moreover, it is found that the highest improvement is reached for the grapple. This can be justified by the similar appearance of the grapple in the two datasets. From a visual perspective, the palakta-crane and minicrane headblocks look more different than the grapples. Moreover, figure 3 shows that the minicrane headblock position and size are more variable than the grapple. This increases the difficulty of the task. Therefore, the pretrained feature representations are more useful for the grapple.

6 Conclusion

This paper analysed the applicability of self-supervised pretraining with MAE on industrial crane datasets and its influence on label-efficiency for semantic segmentation with UNETR. Based on recent research, it was hypothesised that the pretraining increases label-efficiency, especially in use cases with less labeled data.

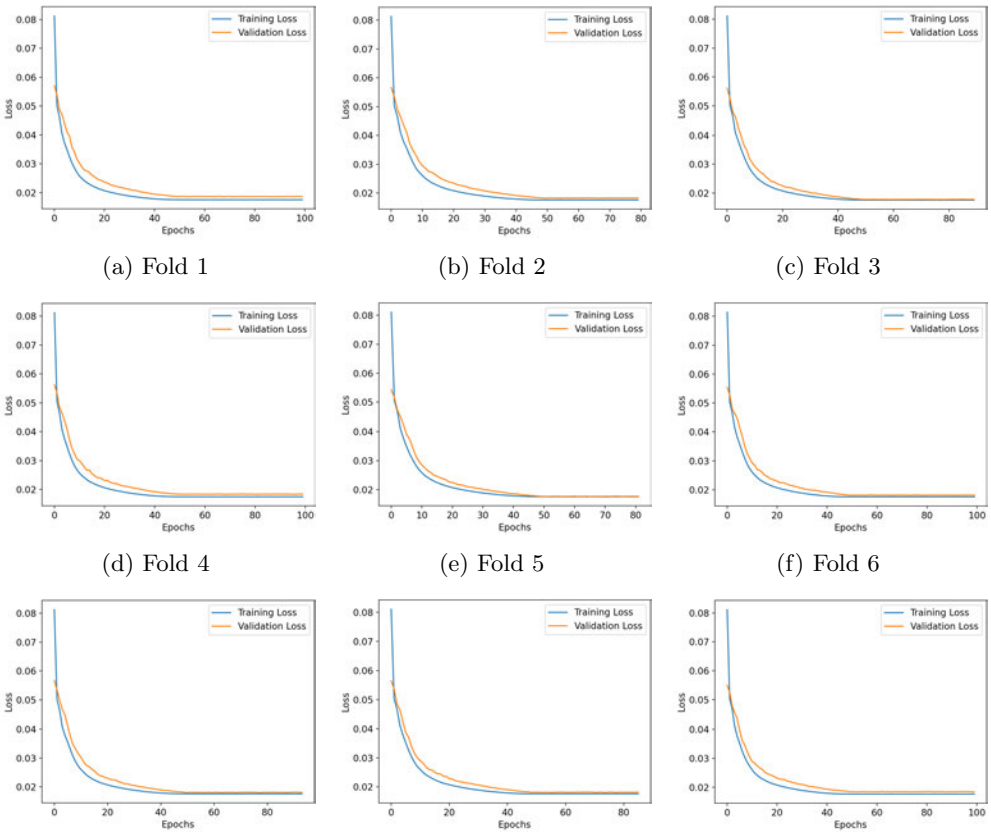
The experimental results indicate that MAE is effectively trainable on industrial data. As a direct evaluation of MAE reconstruction and feature extraction performance was beyond the scope of this paper, limited conclusions can be drawn here, and the topic is left open for future research.

Moreover, it was found that the semantic segmentation with UNETR benefits from using the pretrained feature extractor, and the available labeled data can be used more efficiently. Consequently, the label-efficiency is increased through MAE pretraining. The positive influence increases with a decreasing number of labeled samples.

Additionally, the MAE is useful as well in the case of using different datasets for pretraining and finetuning. The two transfer learning scenarios included the knowledge transfer between different perspectives and between two cranes. In both cases, findings indicate that the downstream task benefits from the pretrained features and label-efficiency increases.

Finally, this paper shows that MAE pretraining is applicable for industrial use cases. As in industry, labels are barely available, and the process is costly and time-consuming, it is an important finding that pretraining increases label-efficiency. Nevertheless, the findings are limited to the analysed industrial use case. To draw general conclusions, further investigations with different industrial datasets are needed.

Appendix A Training curves for MAE



(a) Fold 1 (b) Fold 2 (c) Fold 3

(d) Fold 4 (e) Fold 5 (f) Fold 6

(g) Fold 7 (h) Fold 8 (i) Fold 9

Fig. A1: Training curves for MAE training.

Appendix B Additional discussion of results

The negative influence of pretraining on semantic segmentation with 200 labeled samples is contrary to the expected results. Since self-supervised pretraining with MAE has shown a positive impact on downstream tasks in recent research, similar results were expected for the experiments. For the remaining experiments with more labeled samples, the results are as anticipated. Additionally, the segmentation improvement increases with a decreasing number of labeled images if the experiment of 200 labels is excluded. Therefore, the 200 labels experiment is not reasonable. Nevertheless, hypothesis testing with McNemar's test (chi-squared as test statistic) and a significance level of $\alpha = 0.05$ shows that the models' performances are statistically significantly

different. As the IoU and the Recall for the experiment have a higher variance than the others, an unsuitable experimental setup must be considered as a reason.

Further investigations on the batch size showed that lowering the hyperparameter to 6 (see figure B2) leads to an improved segmentation performance and a positive influence of pretraining. As the transfer learning experiments (see sections 4.3.1 and 4.3.2) with the lowest label amounts also showed unexpected results and high dispersion of the individual fold metrics, the experimental setup might be unsuitable as well.

Given the improved results in figure B2, similar optimisation might have an effect on the larger datasets. Due to time and especially budgetary constraints imposed by PSIORI GmbH, this could not be further evaluated as a part of this paper.

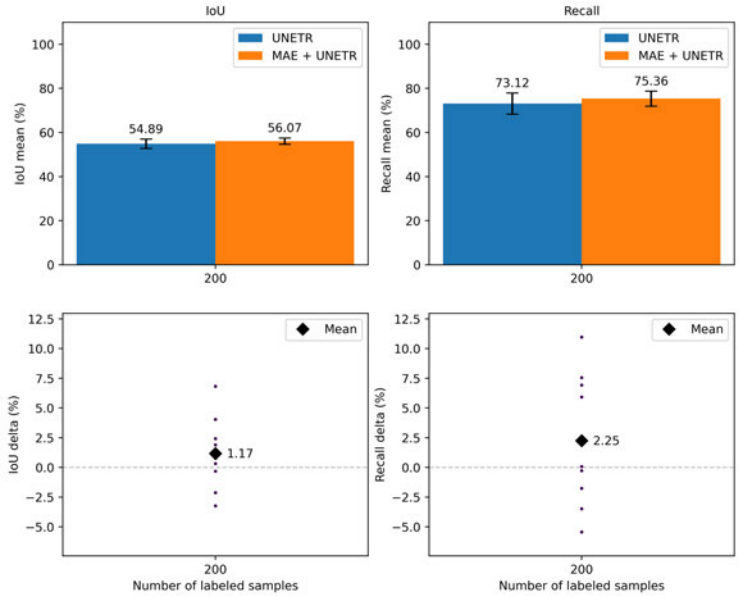


Fig. B2: Label-Efficiency results for 200 labeled samples with a batch size of 6. Results report mean IoU and Recall. The delta between the results of training UNETR with and without MAE-pretraining before are shown in the lower figure.

References

[1] Abadi M, Agarwal A, Barham P, et al (2015) TensorFlow: Large-scale machine learning on heterogeneous systems. URL https://www.tensorflow.org/, software available from tensorflow.org

[2] Adadi A (2021) A survey on data-efficient algorithms in big data era. Journal of Big Data 8(1):24

[3] Akhand M, Roy S, Siddique N, et al (2021) Facial emotion recognition using transfer learning in the deep cnn. Electronics 10(9):1036

[4] Assran M, Caron M, Misra I, et al (2022) Masked siamese networks for label-efficient learning. In: European Conference on Computer Vision, Springer, pp 456–473

[5] Ay B, Tasar B, Utlu Z, et al (2022) Deep transfer learning-based visual classification of pressure injuries stages. Neural Computing and Applications 34(18):16157–16168

[6] Bachman P, Hjelm RD, Buchwalter W (2019) Learning representations by maximizing mutual information across views. Advances in neural information processing systems 32

[7] Bao H, Dong L, Piao S, et al (2021) Beit: Bert pre-training of image transformers. arXiv preprint arXiv:210608254

[8] Caron M, Touvron H, Misra I, et al (2021) Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 9650–9660

[9] Chen J, Lu Y, Yu Q, et al (2021) Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:210204306

[10] Chen T, Kornblith S, Norouzi M, et al (2020) A simple framework for contrastive learning of visual representations. In: International conference on machine learning, PMLR, pp 1597–1607

[11] Chen T, Sampath V, May MC, et al (2023) Machine learning in manufacturing towards industry 4.0: From 'for now'to 'four-know'. Applied Sciences 13(3). https://doi.org/10.3390/app13031903, URL https://www.mdpi.com/2076-3417/13/3/1903

[12] Cole E, Yang X, Wilber K, et al (2022) When does contrastive visual representation learning work? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14755–14764

[13] Dong X, Bao J, Zhang T, et al (2023) Peco: Perceptual codebook for bert pre-training of vision transformers. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 552–560

[14] Dosovitskiy A, Beyer L, Kolesnikov A, et al (2020) An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:201011929

[15] El-Nouby A, Izacard G, Touvron H, et al (2021) Are large-scale datasets necessary for self-supervised pre-training? arXiv preprint arXiv:211210740

[16] Feng P, Tang Z (2023) A survey of visual neural networks: current trends, challenges and opportunities. Multimedia Systems 29(2):693–724

[17] Gidaris S, Singh P, Komodakis N (2018) Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:180307728

[18] Girshick R, Donahue J, Darrell T, et al (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587

[19] Grill JB, Strub F, Altché F, et al (2020) Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems 33:21271–21284

[20] Gu Y, Ge Z, Bonnington CP, et al (2019) Progressive transfer learning and adversarial domain adaptation for cross-domain skin disease classification. IEEE journal of biomedical and health informatics 24(5):1379–1393

[21] Han X, Zhang Z, Ding N, et al (2021) Pre-trained models: Past, present and future. AI Open 2:225–250

[22] Hatamizadeh A, Tang Y, Nath V, et al (2022) Unetr: Transformers for 3d medical image segmentation. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 574–584

[23] He K, Gkioxari G, Dollár P, et al (2017) Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp 2961–2969

[24] He K, Fan H, Wu Y, et al (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9729–9738

[25] He K, Chen X, Xie S, et al (2022) Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 16000–16009

[26] Hess G, Jaxing J, Svensson E, et al (2023) Masked autoencoder for self-supervised pre-training on lidar point clouds. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 350–359

[27] Iman M, Arabnia HR, Rasheed K (2023) A review of deep transfer learning and recent advancements. Technologies 11(2):40

[28] kerasteam (2021) masked_image_modeling. https://github.com/keras-team/keras-io/blob/master/examples/vision/masked_image_modeling.py, commit: be17b64128823cfbfc48188d62b3bea517937196

[29] Khan S, Naseer M, Hayat M, et al (2022) Transformers in vision: A survey. ACM computing surveys (CSUR) 54(10s):1–41

[30] Kingma DP, Ba J (2017) Adam: A method for stochastic optimization. 1412.6980

[31] Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:13126114

[32] Kirillov A, Mintun E, Ravi N, et al (2023) Segment anything. arXiv preprint arXiv:230402643

[33] Kotar K, Ilharco G, Schmidt L, et al (2021) Contrasting contrastive self-supervised representation learning pipelines. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE Computer Society, Los Alamitos, CA, USA, pp 9929–9939, https://doi.org/10.1109/ICCV48922.2021.00980, URL https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00980

[34] Ledig C, Theis L, Huszár F, et al (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690

[35] Li Y, Xie S, Chen X, et al (2021) Benchmarking detection transfer learning with vision transformers. arXiv preprint arXiv:211111429

[36] Lin TY, Maire M, Belongie S, et al (2014) Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, pp 740–755

[37] Liu X, Zhang F, Hou Z, et al (2021) Self-supervised learning: Generative or contrastive. IEEE transactions on knowledge and data engineering 35(1):857–876

[38] Loey M, Manogaran G, Khalifa NEM (2020) A deep transfer learning model with classical data augmentation and cgan to detect covid-19 from chest ct radiography digital images. Neural Computing and Applications pp 1–13

[39] Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440

[40] Loshchilov I, Hutter F (2019) Decoupled weight decay regularization. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=Bkg6RiCqY7

[41] Makhzani A, Shlens J, Jaitly N, et al (2015) Adversarial autoencoders. arXiv preprint arXiv:151105644

[42] Misra I, Maaten Lvd (2020) Self-supervised learning of pretext-invariant representations. In: Proceedings of the IEEE/CVF conference on computer vision and

pattern recognition, pp 6707–6717

[43] Newell A, Deng J (2020) How useful is self-supervised pretraining for visual tasks? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7345–7354

[44] Ozbulak U, Lee HJ, Boga B, et al (2023) Know your self-supervised learning: A survey on image-based generative and discriminative training. arXiv preprint arXiv:230513689

[45] Pan SJ, Yang Q (2009) A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22(10):1345–1359

[46] Reinke A, Tizabi MD, Sudre CH, et al (2021) Common limitations of image processing metrics: A picture story. arXiv preprint arXiv:210405642

[47] Rengaraju U (2022) [tensorflow]unetr + w&b. https://www.kaggle.com/code/usharengaraju/tensorflow-unetr-w-b, version 19

[48] Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells WM, et al (eds) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham, pp 234–241

[49] Russakovsky O, Deng J, Su H, et al (2014) Imagenet large scale visual recognition challenge. CoRR abs/1409.0575. URL http://arxiv.org/abs/1409.0575, 1409.0575

[50] Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556

[51] Strudel R, Garcia R, Laptev I, et al (2021) Segmenter: Transformer for semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 7262–7272

[52] Tian Y, Krishnan D, Isola P (2020) Contrastive multiview coding. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16, Springer, pp 776–794

[53] Touvron H, Cord M, Douze M, et al (2021) Training data-efficient image transformers & distillation through attention. In: Meila M, Zhang T (eds) Proceedings of the 38th International Conference on Machine Learning, Proceedings of Machine Learning Research, vol 139. PMLR, pp 10347–10357, URL https://proceedings.mlr.press/v139/touvron21a.html

[54] Wang R, Lei T, Cui R, et al (2022) Medical image segmentation using deep learning: A survey. IET Image Processing 16(5):1243–1267

[55] Wei C, Fan H, Xie S, et al (2022) Masked feature prediction for self-supervised visual pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14668–14678

[56] Xie E, Wang W, Yu Z, et al (2021) Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in Neural Information Processing Systems 34:12077–12090

[57] Xie Z, Zhang Z, Cao Y, et al (2022) Simmim: A simple framework for masked image modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9653–9663

[58] Zhang W, Ma B, Qiu F, et al (2023) Multi-modal facial affective analysis based on masked autoencoder. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5792–5801

[59] Zhang Y, Liu H, Hu Q (2021) Transfuse: Fusing transformers and cnns for medical image segmentation. In: Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part I 24, Springer, pp 14–24

[60] Zheng S, Lu J, Zhao H, et al (2021) Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6881–6890

[61] Zhou HY, Guo J, Zhang Y, et al (2021) nnformer: Interleaved transformer for volumetric segmentation. arXiv preprint arXiv:210903201

[62] Zhou J, Wei C, Wang H, et al (2021) ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:211107832

[63] Zhou L, Liu H, Bae J, et al (2022) Self pre-training with masked autoencoders for medical image classification and segmentation. arXiv preprint arXiv:220305573

4 Conclusion

This section puts the paper's results in the context of the industrial use case and explains its practical implications. Moreover, possible future research directions are presented.

4.1 Practical Implications

The results of the paper in section 3 show that the self-supervised pretraining method MAE can be successfully applied to the industrial use case of autonomous cranes. Its usage improves the label-efficiency for semantic segmentation with UNETR, particularly in the low-data regime. This is a valuable result, especially for industry, as the available amount of labeled data is usually limited due to its costs in corresponding use cases. Since larger numbers of unlabeled data often exist, they can be used to improve the segmentation without an increased label need.

Furthermore, the results lead to the conclusion that a reduction in terms of labels does not necessarily result in a performance decrease if pretraining is applied. A lower number of labels can be partly compensated by pretraining with unlabeled images. This allows the industry to reduce annotation costs while keeping the semantic segmentation performance.

Since the transfer learning experiments demonstrated similar effects on label-efficiency, the dataset selection is another relevant aspect for segmentation. The results propose that performance can be increased without further annotation effort in the case of a labeled and an unlabeled dataset that is semantically related. The knowledge from one perspective or crane data is advantageous for another. This offers the possibility to use a dataset more broadly and in various model trainings for pretraining, even if it is not the targeted use case. As a result, each dataset becomes more valuable.

4.2 Future Work

From a technical point of view, a comparative analysis concerning different self-supervised pretraining methods and ViT based segmentation architectures is needed. For this thesis, the MAE and UNETR were chosen because of their promising results in research. As research does not address industrial datasets, the transferability of the corresponding results is limited. Therefore, different architectural approaches might be more suitable for the use case of autonomous cranes.

Additionally, the MAE was also used similarly as in the original paper. Further investigations might focus on performance improvements of the self-supervised pretraining method for the use case. One direction is hyperparameter tuning regarding patch size and data augmentation. Another possibility is to investigate the influence of the pretraining data amount. For this thesis, the same number of unlabeled images was used for all experiments. It is conceivable that the used dataset is not diverse enough to represent the use case, and a larger data amount would lead to improvements. In contrast to that, it is also possible that a smaller dataset might provide similar information. In that case, a reduced dataset would lead to a decrease in the needed training time.

With respect to MAE, a further research direction concerns its evaluation. For this thesis, it was mainly visually evaluated and through the analysis of the downstream task performance. To improve its performance, gaining more insights into the learned features might be helpful.

For the semantic segmentation, a comparison of transformer and CNN-based models might be interesting as the investigated use case is currently implemented with a U-Net architecture. Future work could analyze the influence of using a ViT based architecture such as UNETR. In addition, the label-efficiency and computational cost of UNETR with MAE pretraining could be compared with the U-Net to investigate at what cost the annotation requirements are reduced.

Bibliography

[1] Femi Ajewole, Ani Kelkar, Dylan Moore, Emily Shao, and Manju Thirtha. Unlocking the industrial potential of robotics and automation. https://www.mckinsey.com/industries/industrials-and-electronics/our-insights/unlocking-the-industrial-potential-of-robotics-and-automation, 2023. Accessed: 05/10/2023.

[2] ANDRITZ AG. Innovative solutions for wood handling and logyard management. https://www.andritz.com/resource/blob/459918/ec59dd237461ad6aa06b317f1a3f2490/pp-logyard-cranes-data.pdf, 2022. Accessed: 05/10/2023.

[3] Peter Berg and Oskar Lingqvist. Pulp, paper, and packaging in the next decade: Transformational change. https://www.mckinsey.com/industries/paper-forest-products-and-packaging/our-insights/pulp-paper-and-packaging-in-the-next-decade-transformational-change#/, 2019. Accessed: 05/10/2023.

[4] cloudfactory. Image annotation for computer vision. https://www.cloudfactory.com/image-annotation-guide. Accessed: 05/10/2023.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[7] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer

in vision through object detection. *Advances in Neural Information Processing Systems*, 34:26183–26197, 2021.

[8] Xu Han, Zhengyan Zhang, Ning Ding, Yuxian Gu, Xiao Liu, Yuqi Huo, Jiezhong Qiu, Yuan Yao, Ao Zhang, Liang Zhang, et al. Pre-trained models: Past, present and future. *AI Open*, 2:225–250, 2021.

[9] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 574–584, 2022.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[12] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[14] PSIORI GmbH. Internal image.

[15] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. URL http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a.

[16] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014. URL http://arxiv.org/abs/1409.0575.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[18] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021.

Erklärung zur selbstständigen Bearbeitung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort Datum Unterschrift im Original