Corso di dottorato di ricerca in:

"Scienze Matematiche Informatiche e Fisiche"

Ciclo 36°

Titolo della tesi

"Deep learning techniques for image segmentation and anomaly detection in low data regimes"

Dottorando

Axel De Nardin

Supervisore

Prof. Gian Luca Foresti

Co-supervisore

Prof. Claudio Piciarelli

Anno 2024

Dedicated To my mother, Paola, for supporting me through the ups and downs of this journey

Abstract

This thesis focuses on the problem of image segmentation in low-data settings. In particular, the two specific problems that are tackled in the present work are the ones revolving around anomaly segmentation for industrial quality control and document layout segmentation of ancient manuscripts. For the first problem, two novel attention-based approaches are proposed, one based on the popular U-Net architecture and the second one on the more recent Vision Transformer which has been enhanced for the task at hand with a masking module and a multi-resolution self-attention component. As for the document layout analysis, we introduce a few-shot segmentation framework based on the combination of DeepLabV3+, a robust deep learning architecture for semantic segmentation, with a traditional computer vision algorithm for image binarization while at the same time relying on a novel instance generation strategy that allows to leverage the small amount of data available fully. Furthermore, we provide an analysis of the effects of transfer learning in this domain-specific context, showing the drawbacks of pre-training on large general-purpose datasets compared to smaller domain-specific ones. For each of the proposed approaches, we provide the experimental results obtained on popular publicly available datasets for the corresponding task.

List of publications

- **De Nardin A.**,Zottin S, Piciarelli C. Colombi E., Foresti G. L, A One-Shot Learning Approach to Document Layout Segmentation of Ancient Arabic Manuscripts, WACV 2024, (Accepted).

- Zottin S, **De Nardin A.**, Piciarelli C. Colombi E., Foresti G. L, U-DIADS-Bib: a full and few-shot pixel-precise dataset for document layout analysis of ancient manuscripts, International Journal of Neural Computing and Applications, NCAA, (Accepted).

- **De Nardin A.**,Zottin S, Piciarelli C. Colombi E., Foresti G. L, Few-shot pixel-precise document layout segmentation via dynamic instance generation and local thresholding, International Journal of neural systems, IJNS, 33(10), 2350052.
https://doi.org/10.1142/S0129065723500521

- **De Nardin A.**, Zottin S., Paier M. Forest G. L, Colombi, E., Piciarelli C., Efficient few-shot learning for pixel-precise handwritten document layout analysis, Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) 2023, pages 3680-3688.

- **De Nardin, A.**, Mishra, P., Foresti, G. L., & Piciarelli, C. (2022). Masked Transformer for Image Anomaly Localization. International journal of neural systems, IJNS, 32(7), 2250030.
https://doi.org/10.1142/S0129065722500307

- Daniele Veritti, Rubinatto Leopoldo, Valentina Sarao, **Axel De Nardin**, Gianluca Foresti, Paolo Lanzetta, Behind the mask: a critical perspective on the ethical, moral, and legal implications of AI in ophthalmology, BMJ Open Ophthalmology (2023).
https://doi.org/10.1007/s00417-023-06245-4

- **De Nardin A.**,Zottin S, Piciarelli C. Colombi E., Foresti G. L, s ImageNet always the best option? An overview on transfer learning strate-

gies for document layout analysis, International conference on Image Analysis and Processing. ICIAP 2023

- Beltrami AP, De Martino M, Dalla E, Malfatti MC, Caponnetto F, Codrich M, Stefanizzi D, Fabris M, Sozio E, D'Aurizio F, Pucillo CEM, Sechi LA, Tascini C, Curcio F, Foresti GL, Piciarelli C, **De Nardin A**, Tell G, Isola M. Combining Deep Phenotyping of Serum Proteomics and Clinical Data via Machine Learning for COVID-19 Biomarker Discovery. Int J Mol Sci. 2022 Aug 15;23(16):9161. doi: 10.3390/ijms23169161. PMID: 36012423; PMCID: PMC9409308.

- **De Nardin, A.**, Mishra, P., Piciarelli, C., Foresti, G.L. (2022). Bringing Attention to Image Anomaly Detection. In: Mazzeo, P.L., Frontoni, E., Sclaroff, S., Distante, C. (eds) Image Analysis and Processing. ICIAP 2022 Workshops. ICIAP 2022. Lecture Notes in Computer Science, vol 13373. Springer, Cham. https://doi.org/10.1007/978-3-031-13321-3_11

- **Axel De Nardin**, Marino Miculan, Claudio Piciarelli, and Gian Luca Foresti. A time-series classification approach to shallow web traffic deanonymization. Proceedings of the fifth Italian conference on cyber security, ITASEC 2021, volume 2940, pages 156-165. CEUR-WS

Acknowledgments

"Ku areba raku ari - There are hardships and there are delights", is a Japanese saying that was probably invented by an ancient Ph.D. student, and I must say I can see his point!

My research journey in the past three years has been undeniably filled with many delightful, tough, and even some very tough moments, but as they say "if the path that you are following seems too easy, then you are on the wrong path".

It's precisely because of the struggle I went through during my Ph.D. that I want to take some time here to thank all the people who helped me succeed in this incredible journey, people without whom I probably couldn't have achieved all I did in the past three years and that undoubtedly had a great influence in shaping the person I've become. First of all, I want to thank Prof. Gian Luca Forest and Prof. Claudio Piciarelli for offering me the possibility to join their lab in the first place and, subsequently for introducing me with their expertise to the wild and beautiful world of academic research. If I think back to when I started it seems unbelievable how much I've grown, both personally and professionally, since then and how my perspective changed on so many things, and I can honestly say this was mostly due to the opportunity to work side by side with them.

Furthermore, I want to thank our new acquisition in the lab, Silvia Zottin for collaborating with me on a wide variety of research projects in the past year and a half and making my Ph.D. journey far less lonely than it would have been otherwise.

I also want to give a special thanks to Clara Veronese, my high school computer science professor who, during a particularly hard time of my life, made the effort to foster my passion for this field and believed in my potential when many others didn't.

Finally, I want to thank my friends and especially my family for putting up with my occasional (some may say frequent) rants about failed experiments and rejected papers, i know I'm not always easy to deal with in those moments of struggle, which made their support even more valuable.

To conclude with another inspiring quote, Ryūnosuke Akutagawa once said "Individually we are a drop, together we are an ocean", I find this to be absolutely true, and it does not apply only to the aforementioned people, but to the entire scientific community which collectively works toward a greater goal even when our individual contributions may seem very limited or unimportant.

Contents

List of Figures

List of Tables

1

Introduction

1.1 An introduction to image segmentation

Image segmentation has always been a fundamental task in the field of computer vision and image understanding and it gained even more popularity in recent years with the advent of deep learning techniques. The problem of segmenting images is highly relevant for a plethora of important applications including, but not limited to:

- Medical imaging [69, 137] (e.g. tumor detection, tissue volume estimation)
- Video surveillance [48] (e.g. moving objects identification, novelty detection)
- Autonomous driving vehicles [45, 121] (e.g. obstacle and navigable surface detection)
- Document analysis [41, 127] (e.g. layout segmentation)
- Anomaly detection [97] (segmentation of the anomalous areas for increased explainability)

But what is image segmentation? Image segmentation can be defined as the problem of individually classifying each pixel contained in an image based on one or multiple criteria. There are mainly three macro-categories

of image segmentation tasks, semantic segmentation, instance segmentation and panoptic segmentation [72](Fig. 1.1). Semantic segmentation is the task of separating the different elements of an image by individually classifying each of its pixels based on the semantic category they belong to. Typically this task is much more complex than simply classifying the image as a whole, as the required level of detail is much higher. As an example, we can think about the perception system of an autonomous driving vehicle, where we want to identify the different components of the scene that is captured by the cameras mounted on it, such as the road, pedestrians, other cars, potential obstacles, etc. This approach can result in the problem statement being poorly defined, especially if there are several instances belonging to the same class. Think about a very crowded street, in that case, all the people would be grouped together resulting in a segmentation that presents a relatively low level of detail.

Instance segmentation adds a further level of complexity and detail by requiring that each individual instance belonging to a semantic class is identified as a separate entity. This would mean, going back to the previous example of the crowded street, that each individual must be segmented separately. Finally, panoptic segmentation combines the two tasks by requiring a segmentation both at a semantic and at an individual level. This type of segmentation is usually required for applications where a high degree of perceptual detail is required.

1.2 Types of approaches

Over time a wide variety of approaches has been employed to solve this type of problem. Traditional computer vision techniques adopted for this task can be grouped into 5 main categories:

1. **Threshold based [110]:** Threshold-based algorithms represent the simplest form of image segmentation techniques and work by classifying each pixel based in an image based on a pre-defined and task-dependant

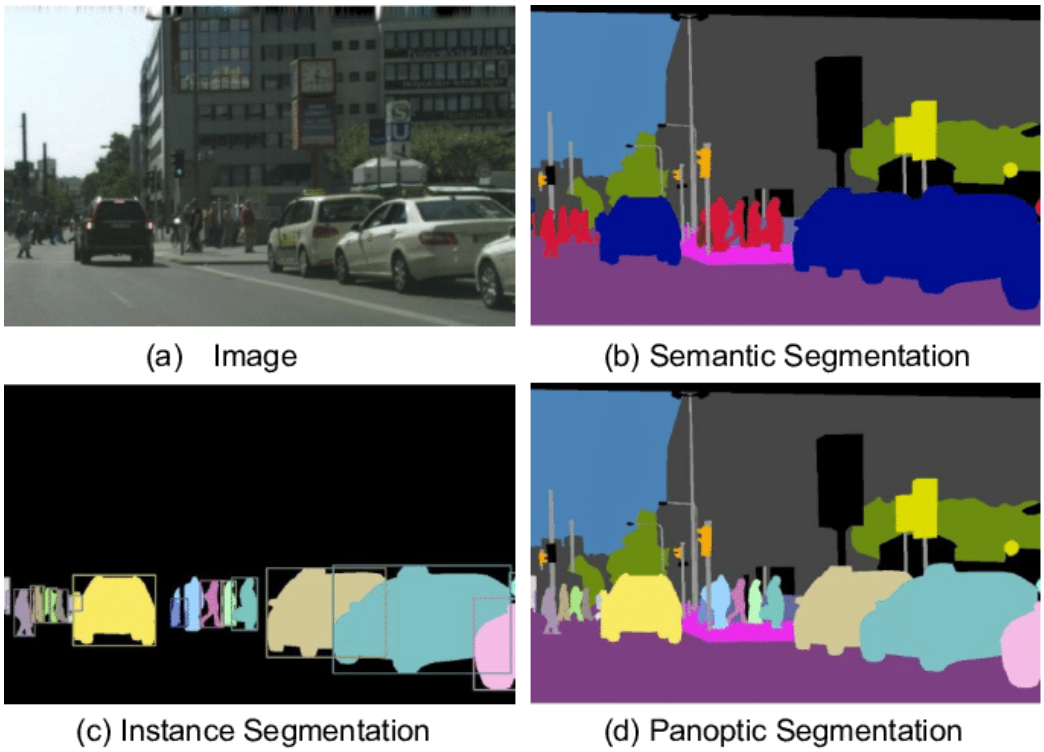


Figure 1.1: Example of the three types of image segmentation problems. [59]

threshold which can be either globally defined for the entire image or, in more sophisticated algorithms it can be dynamically and locally defined inside sub-regions of the original image. A common application for threshold-based image segmentation is represented by the sub-task of image binarization, widely adopted in the context of document analysis to separate the text from the background. While their scope is somewhat limited they can be very efficient for some applications as they are usually fast to implement and execute.

2. **Edge based [53, 3, 67]:** Approaches belonging to this category work by identifying the edges of the different elements belonging to an image by analyzing different types of features such as contrast, texture, color and saturation variations. the main downside of this type of approach is that they are only reliable on high-contrast images with a limited amount of edges. Furthermore, they tend to be computationally

expensive.

3. **Region-growth [70]:** In region-based image segmentation, the image is divided into a set of regions represented by a group of pixels located based on a seed. These regions are then progressively expanded or shrunk based on the similarity of the surrounding pixels until convergence. The main issue with this type of approach is that they are very heavily reliant on the selection of the hyperparameters, being represented by the starting seed and the similarity measure.

4. **Clustering-based [128, 6]:** Similarly to region-based approaches, clustering ones try to group together the parts of the image that share common characteristics but are not necessarily located spatially close to each other.

5. **Watershed [75]:** Watershed segmentation algorithms treat grayscale images like topographic maps, where the elevation is determined by the brightness of each pixel. Then, the image is divided into multiple regions being represented by pixels with the same height (i.e. same gray value). Watershed segmentation approaches are typically very efficient, however, they suffer from a major drawback, represented by over-segmentation, which is the situation in which the algorithm detects an overly large amount of boundaries and ends up segmenting the various regions composing an image into smaller, potentially meaningless, sub-regions.

In general, the majority of the most successful approaches were based on the extraction of handcrafted sets of features from the input images, such as HOG [31] and SIFT [114]. While these approaches have proved to be effective to a certain extent for the task of image segmentation, they pose a series of challenges that are typically hard to overcome. In fact, Handcrafted feature approaches require a high degree of domain knowledge which can and has severely limited the adoption of these systems to a wide range of

application areas while at the same time making the developed strategies hard to seamlessly transfer to another domain.

1.2.1 Deep Learning approaches

A huge step forward in terms of effectiveness, efficiency and applicability has been represented by the advent of deep learning systems. The main advantage of deep learning approaches is that they represent an end-to-end framework that relies solely on data to provide highly accurate outputs for the tasks at hand, making them much more general-purpose compare to traditional approaches.

The fact that nowadays data availability is rarely an issue, thanks to the plethora of datasets that have been publicly released for all kinds of applications, including image segmentation, led to an exponential growth in the adoption of deep learning systems which, in turn, achieved higher and higher degrees of accuracy in their predictions. Since the advent of deep neural networks, a wide variety of models and architectures have been explored with the goal of solving the problem of image segmentation in its different declinations. Consistently with what happened for other computer vision-oriented tasks the most prominent architectures in this field have been historically based on Convolutional Neural Networks (CNNs) which overtime appeared in different forms while preserving their core idea of spatial invariance that allows capturing the features that characterize the different elements of an image without being affected by their position.

A convolution-based architecture that gained a lot of attention in semantic segmentation is represented by Fully Convolutional Networks (FCN), characterized by the removal of the fully connected layers that appeared in the first CNNs, mainly used for the task of image classification, and the introduction of a set of deconvolution operations aimed at providing a dense prediction by reconstructing an output of the same size as the input image, namely the segmentation map [65]

Different types of convolutions have also been explored over time, with

prominent examples such as the Deeplab architecture which introduced the use of Atrous convolutions in all its different iterations [22, 23, 24, 26] with the aim of leveraging contextual information at multiple scales in the image to achieve improved accuracy in the segmentation task.

A further example of CNN architecture that has been widely used in the context of semantic segmentation is represented by U-shaped networks, introduced in the U-net paper [95], and characterized by a contraction of the input through an encoder component, which extracts the most meaningful features of the input image and a decoder component which takes this feature map and expands it back to the original image size via upconvolution layers. Many different improvements have been implemented in U-Nets over time including, for example, multires blocks [56].

Recent works also explored the use of different types of attention mechanisms to further improve the performance of the models on the image segmentation task Some notable examples include the adoption of criss-cross attention [54], the aggregation of long-range contextual information through the use of global attention [135] and, more recently, the introduction of a Vision Transformer (ViT) [39] base architecture as the feature extraction module of the segmentation framework [111].

1.3 Motivation

As previously stated, the advent of deep learning techniques greatly advanced the field of image segmentation, which represents a step of paramount importance in the field of image analysis, as it is necessary to perform further processing such as recognition or description tasks. However, many of the modern deep neural networks require very large amounts of data to be trained. In particular, the harder it is the task the larger the amount of data needed to train a model that can effectively tackle it. While data has become increasingly available in the past years, there are still settings in which gathering the necessary data can be a problem for a variety of reasons, be

it an intrinsic lack of it as in the context of anomaly detection, where we typically have a huge amount of available data for the positive samples but almost no data available for the negative ones, or the difficulty in obtaining an appropriate amount of ground truths which is a common problem for semantic segmentation tasks where manually labeling each pixel in an image is a very time-consuming task and may also require a certain degree of domain knowledge as in the context of document layout analysis. This lack of data in certain settings leads to the necessity to develop specific strategies in order to be able to effectively perform the task at hand and, while there has been an increasing effort toward the goal of data efficiency in image segmentation tasks this is still a relatively young area of research which presents many opportunities for further improvements.

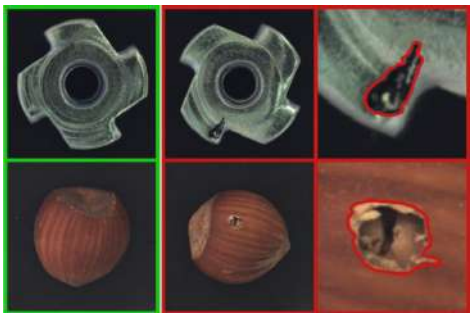
1.4 Problems overview

In this section, we will provide an overview of the 2 main problems which this thesis focuses on, namely Anomaly Detection and Segmentation and Document Layout Segmentation.

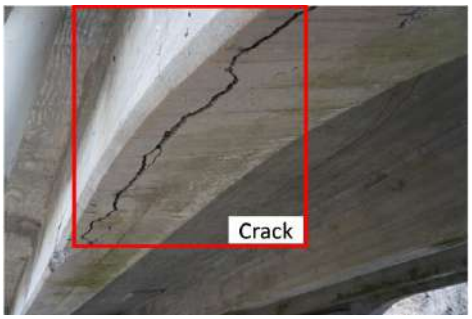
1.4.1 Anomaly Detection and Segmentation

Anomaly detection is referred to as the process of identifying novel samples that exhibit significantly different traits compared to an accepted and predefined model of normality. In real-life scenarios, like Visual Inspection Systems (VIS), the novel sample can show a previously unseen and considerable difference from the reference data, and labeling novel examples is not possible. Systems that can perform such a task in an autonomous way are in high demand, ranging from banking [129], medical imaging [68], defect segmentation [91, 74], inspection [91], quality control [76], video surveillance [92], etc (Fig. 1.2).

In fact, while this kind of task could be easy for a human being, the same does not hold for an autonomous system trained over a small set of data. A

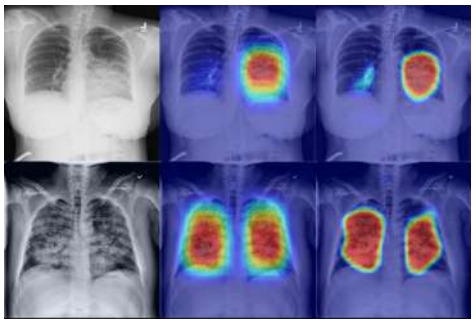


(a) Industrial quality inspection



(b) Structural monitoring

(c) Video survaillance



(d) Medical Imaging

Figure 1.2: Examples of anomaly localization tasks in different application fields.

significantly pressing reason that makes this task challenging is the fact that, while it is a classification problem, the traditional methods are intrinsically flawed by the way they process the data. In fact, most often the industrial-grade data available in real-life are highly imbalanced [73] or labeled data exist only for the normal class. When dealing with images we also have the added problem of the high dimensionality of the data which often leads to more classical methods for anomaly detection, such as clustering techniques, to achieve poor performance. The task of anomaly segmentation brings the problem a step further by requiring not only a classification of the instances based on their normal or anomalous nature but also the identification of the region inside the image that contains the anomaly.

1.4.2 Document Layout Segmentation

Layout analysis is the process of identifying and recognizing the physical and logical organization and structure of document images [64, 14]. Document layout analysis includes three main tasks each with a specific purpose and which leads to the study of different characteristics of the document. These three key sub-processes of layout analysis are layout segmentation, text line segmentation and baseline detection.

Document Layout Segmentation is a prerequisite step of DIA. Layout segmentation is the process that segments the document pages into different semantically meaningful regions such as main text, paratexts, decorations and background. In particular, the page segmentation of historical manuscripts allows humanists to study documents more quickly and easily because it allows the paratexts (i.e all the semantic elements which are part of the foreground but don't belong to the main text) to be analyzed separately. Furthermore, layout segmentation is a step of paramount importance in document analysis as it enables further processing steps such as, for example, Optical Character Recognition (OCR) and automated transcription.However, performing this task in historical manuscripts is much more difficult than in printed documents [94], due to many variations, such as layout structure, decoration, different writing styles, texture, and degradation.

1.5 Goals and outline of the thesis

The main goal of this thesis is to investigate new techniques to perform image segmentation in low-data settings. The focus will be specifically on two main application settings. The first one is represented by anomaly segmentation in industrial settings, where the core problem is represented by a high imbalance in the available data due to the lack of negative (anomalous) instances. The second task is document layout segmentation in ancient manuscripts for which the motivation to develop a data-efficient models setting comes from the fact that generating the ground truths necessary to train a regular deep

learning model is an extremely daunting task due to the level of precision and domain knowledge required, which often varies from document to document due to their historical nature and due to the fact they may be written in different languages. The rest of the thesis is organized as follows. Chapter 2 outlines previous works in the field of image segmentation with a particular focus on the current state-of-the-art approaches. Chapter 3 introduces an attention-based variant of a previously established anomaly segmentation framework, Chapter 4 builds upon the previous chapter by introducing an entire attention-based anomaly segmentation framework relying on the use of a novel iteration of the Vision Transformer architecture featuring both a masking and a multi-resolution component. In Chapter 5 a few-shot document layout segmentation framework, based on the combination of a robust semantic segmentation network with a traditional binarization algorithm is proposed. Chapter 6 tackles the efficiency aspect of segmentation systems from a different point of view, focusing on a thorough analysis of transfer learning approaches in the context of document layout segmentation. Finally, in chapter 7 a conclusion of the thesis is provided together with potential ideas for further development in this research area.

2

Related works and State of the art

In the previous chapter, we have provided an introduction that clearly shows how important and active the research field of image segmentation is. All the efforts put into improving the quality of the predictions of the systems employed for this task have led to a variety of highly effective deep-learning models and frameworks for a wide range of application scenarios. In this chapter, a brief description of the most prominent of these systems will be provided, with a particular focus on those focusing on the problems of anomaly segmentation and document layout segmentation.

2.1 General purpose semantic segmentation

In this section, we will provide a thorough description of two very prominent state-of-the-art models for general-purpose semantic segmentation based, respectively, on the CNN and Vision Transformer architectures. These two models are namely DeepLabV3+[26] and SegFormer [126].

2.1.1 DeepLabV3+

DeepLabV3+ [26] represents the latest iteration, introduced by Google in 2018, of the DeepLab framework series of segmentation models. The success of this model is to be attributed to a set of architectural choices, some of

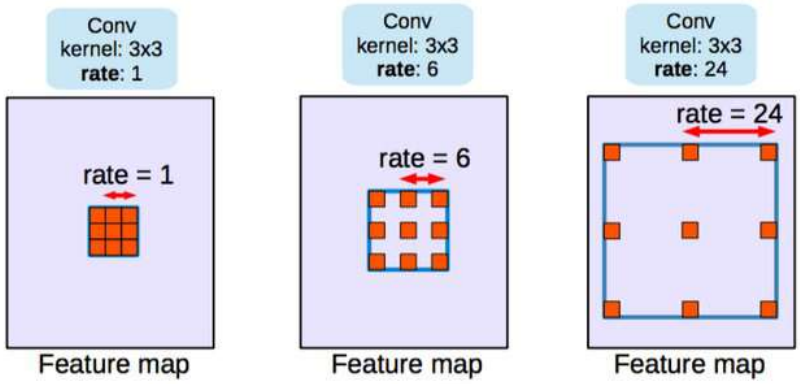


Figure 2.1: Atrous convolution with different dilation rates [24]

which were inherited from its predecessors (such as the introduction of atrous convolution) and some others which represented a departure from them.

The first element characterizing the Deeplabv3+ architecture is represented by the replacement of the traditional convolution operation with the atrous one, also called dilated convolution (Fig. 2.1). Atrous convolution is characterized by the weights of the kernel being spaced apart instead of adjacent as in traditional convolutional layers. To control the amount of spacing between them an additional parameter "r" is introduced. The key advantage of this type of convolution is that it allows for wider receptive fields at the same computational cost as a traditional convolution which, in turn, allows for more contextual information to be captured which is of great importance in the context of semantic segmentation. DeepLab3+, as its predecessor DeepLab3, employs atrous convolution in both a cascade arrangement, which allows for bigger feature maps to be retrieved, as well as in a parallel fashion by assigning a different dilation rate to each of the modules to capture information at multiple scales in the original image. The first improvement introduced by DeepLabv3+ is represented by the encoder-Decoder structure that wasn't present in the previous iterations of the architecture. In particular, DeepLabV3 is used as the encoder module of this new version of the framework to extract the feature map from the input image. This feature map is then fed to the decoder module which processes it in the fol-

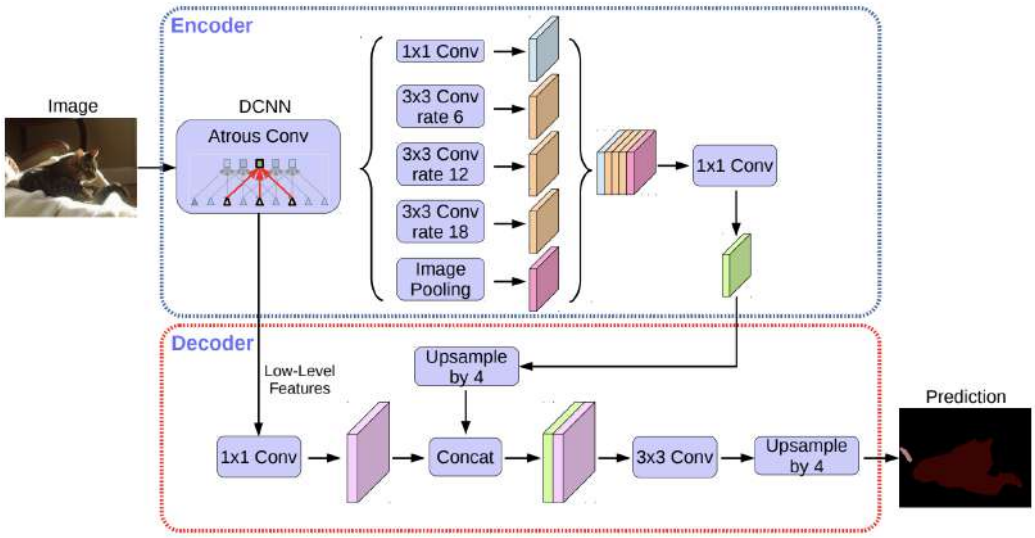


Figure 2.2: Visual representation of the full DeepLabV3+ architecture. [26]

lowing way: first, the encoded features are bilinearly upsampled by a factor of 4 and concatenated with the corresponding lower-level feature, which are passed through a 1x1 convolution layer to reduced the amount of channels that usually characterizes them. After the concatenation the resulting feature map is refined through another 3x3 convolutional layer and finally, a last upsampling step is performed to ensure that the output segmentation map has the same size of the input image. This process proved to provide much more precise segmentation maps compared to just upsampling the original feature map by a factor of 16. Finally, a modified version of the Aligned Xception [30] replaced the more traditional ResNet [49] as the backbone of the framework. The first change compared to the original Aligned Xception paper is that a deeper network is used in this context. Furthermore, all the max pooling operations are replaced by depthwise separable convolutions with striding which allow to extract feature maps at an arbitrary resolution. Finally, inspired by the MobileNet [52] architecture design, each 3x3 depthwise convolution is followed by batch normalization and ReLU activation functions.

The full architecture is reported in Fig. 2.2.

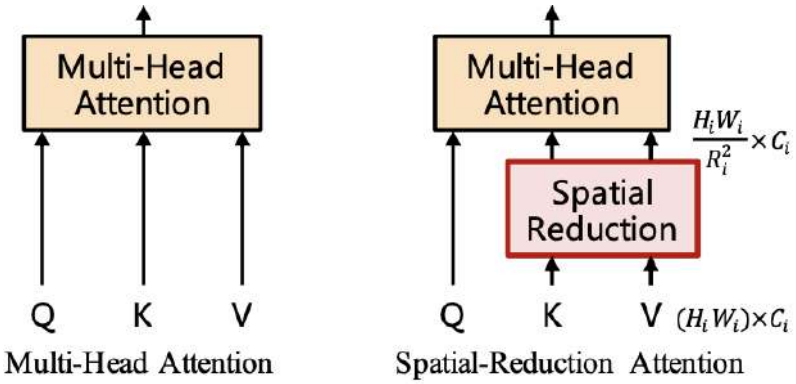


Figure 2.3: Visual depiction of the difference between the traditional Multi-Head Attention (MHA) module and the Spatial Reduction Attention module. [126]

2.1.2 SegFormer

In the last few years methods relying on the Transformer architecture to perform different types of computer vision tasks became more and more popular. The Transformer represents an alternative to traditional CNNs where the main component is based on a self-attention mechanism that allows capturing relationships between distant parts of the images more effectively. The drawback, however, is that its original form is not a very scalable solution as the complexity of the networks increases quadratically with the resolution of the image to be analyzed. For this reason, alternative, more efficient, solutions have been proposed. One of these is represented by SegFormer [126], a ViT-based Encoder-Decoder architecture that improved the SoTA for semantic segmentation when introduced in 2021. Compared to the original ViT architecture, there are 3 main differences introduced by SegFromer to improve efficiency. The first change is represented by the removal of the positional encoding that was originally used to identify the position of each region of the image. The main problem with this approach is that it loses its effectiveness when the resolution of the test set images is different from the ones from the training set, as in this scenario the positional information encoded by it needs to be interpolated leading to a reduction in the model's

accuracy. In its stead, SegGFormer introduced a new way to retrieve location information, with a module named MiX-FFN (Feed Forward Network) which considers the effect of zero padding to leak location information and is defined as follows:

$$X_{out} = MLP(GELU(Conv_{3x3}(MLP(X_{in})))) + X_{in} \tag{2.1}$$

where X_{in} is the feature extracted by the self-attention module. The second change introduced by SegFormer involves the Self-Attention module which is the main bottleneck in the original ViT architecture as it is characterized by a quadratic complexity. To avoid this a sequence reduction process (Fig. 2.3), as proposed in [123], is used in the SegFormer architecture. The key idea behind this process is to use a reduction ratio R to reduce the length of the input sequence in the following way:

$$\begin{aligned} \hat{K} &= Reshape(\frac{N}{R}, C \cdot R)(\hat{K}) \\ K &= Linear(C \cdot R, C)(\hat{K}) \end{aligned} \tag{2.2}$$

where K is the sequence to be reduced, Reshape is the function used to bring K to the shape of $\frac{N}{R} \times C \cdot R$ and Linear is a linear layer that takes a C_{in} tensor as input and produces a C_{out} dimensional output, leading to a new K with shape $\frac{N}{R} \times C$. This process allows for a reduction in complexity from $O(N^2)$ to $O(\frac{N^2}{R})$.

These two newly introduced modules, coupled with an Overlapped Patch Merging module that allows performing the self-attention process on overlapped patches, instead of separated ones as in the original ViT architecture, allow obtaining an Encoder for the SegFormer framework with improved effectiveness and efficiency.

The third and last key departure from the original ViT architecture introduced by SegFormer involves the decoder structure. In fact, the latter implements a lightweight decoder that consists only of MLP layers and is enabled by the hierarchical Transformer encoder that is characterized by a

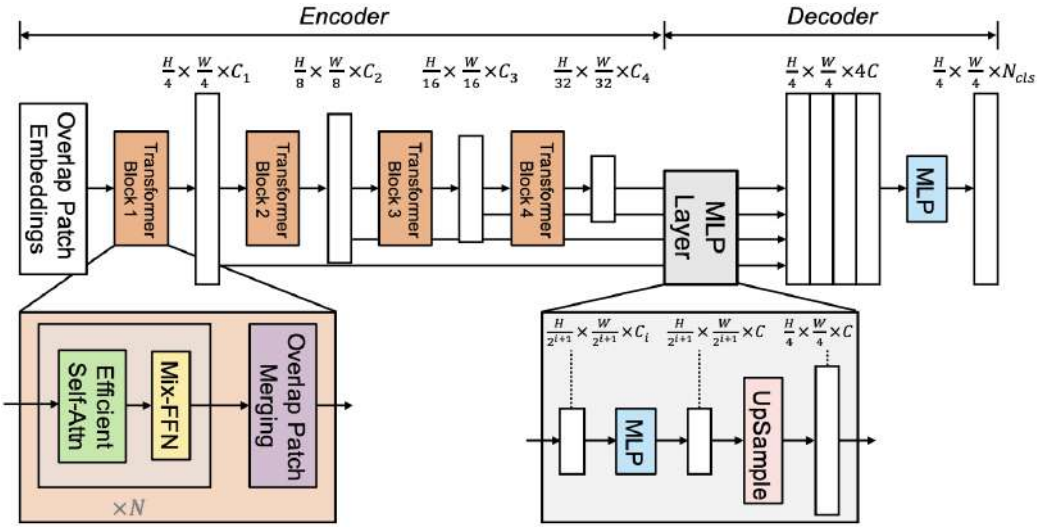


Figure 2.4: Visual representation of the SegFromer architecture [126]

larger, more effective, receptive field compared to traditional CNN ones.

The Decoder architecture of SegFormer consists of four main steps reported hereafter:

$$\begin{aligned} \hat{F}_i &= Linear(C_i, C)(F_i), \forall_i \\ \hat{F}_i &= Upsample(\frac{W}{4} X \frac{W}{4})(\hat{F}_i, \forall_i \\ F &= Linear(4C, C)(Concat(\hat{F}_i)), \forall_i \\ K &= Linear(C, N_{cls})(F) \end{aligned} \tag{2.3}$$

In the first step, the feature maps extracted by the Encoder are passed through an MLP layer to unify their channel dimension. Then, they are upsampled to 1/4th and concatenated together. Subsequently, the concatenated maps are fused through an additional MLP layer. Finally, one last MLP layer processes the fused feature maps and produces the segmentation mask M with a $\frac{H}{4} \times \frac{W}{4} \times N_{cls}$ resolution, with N_{cls} representing the number of categories contained in the segmentation mask. As for DeepLabV3+, the SegFormer architecture is shown in Fig. 2.4.

2.2 Anomaly segmentation

When it comes to Anomaly detection and segmentation, traditional training strategies can rarely be used, as typically this type of problem is characterized by a very marked data imbalance in favor of the normal instances. This means that the data containing the abnormalities or defects is rarely available in large enough quantities to be used to effectively train a deep learning model. For this reason, the majority of approaches dealing with anomaly detection or segmentation task rely on alternative learning paradigms such as self-supervised and unsupervised learning. In this section, we will provide both an overview of the deep learning approaches for this class of problems and an in-depth description of the current state-of-the-art approach.

When it comes to Deep Learning approaches to anomaly detection and segmentation, we can classify them into 3 main categories: deep learning for feature extraction, deep learning approaches that learn a feature representation of normality and end-to-end anomaly score learning frameworks [83].

Deep learning for feature extraction

In the systems belonging to the first category, the power of deep learning is leveraged to extract a low-dimensional representation that capsules the key features of the high-dimensional input, in our case consisting of images, provided to the network. In this set of approaches, the feature extraction and the anomaly scoring modules are completely disjoint and the performance of the latter is strongly dependent on the quality of the features extracted by the former.

A popular way of approach to improve the quality of the extracted features while minimizing the computational cost is to employ a pre-trained network, such as the popular VGG [108] or ResNet [49], and then fine-tune it on the target domain dataset or even use it as it is and train only the anomaly scoring module [119, 7, 86]. In recent years ImageNet [36] has become the de-facto standard dataset on which to perform the pre-training step

for the feature extraction networks in the field of computer vision, the reason behind this is that it consists of a wide variety of classes and a large number of instances.

While pre-training on a general-purpose dataset is often very convenient in terms of computational cost, it doesn't always lead to the best results. In fact, when working within domains that present very specific characteristics, such as medical imaging, for example, this approach is not always effective. For this reason, as a second class of approaches explicitly trains the feature extraction network on the target domain dataset [42, 57, 32]. While this approach requires more time and computational resources it ensures that the extracted features specifically target the criticalities of the task at hand, therefore leading, most of the time, to improved results.

Learning normality representation

Compared to the previous category of approaches, normality learning ones typically introduce a connection between feature extraction and anomaly scoring. These approaches can be categorized into two main groups: Generic Normality Feature Learning and Anomaly Measure-dependent feature learning.

The first category includes all those approaches that try to optimize a generic feature learning objective, non necessarily designed for the task of anomaly detection/localization but that can still lead to learning the regularities of the data. Self-supervised learning approaches, such as Autoencoders (AE) and Generative Adversarial Networks (GANs) typically fall in this category.

A GAN network is typically composed of two main components, a Generator which starting from a latent representation tries to generate images as close as possible to the one present in the dataset used, and a Discriminator which receives as input both the images generated by the Generator and the original ones and tries to discriminate between the two. One way of performing anomaly detection through the use of GANs is to train the model

only with "good" images so that during the testing process it should be able to recognize them but not the anomalous one and therefore it could be used to discriminate between data with and without anomalies. A different approach, which also allows performing Anomaly localization, was proposed by Schlegl et al. with the introduction of the AnoGAN architecture[102]. The main idea behind this model is represented by the introduction of an additional component which is trained to learn the inverse transformation of the Generator in order to produce the latent representation of the original images, which can then be used to perform a reconstruction of said images and thus compare it with the respective original ones. The main issue with this idea is that the inversion process is very computationally expensive. For this reason, another model, known as f-AnoGAN [103] (faster AnoGAN) was introduced. This approach, compared to the original AnoGAN network, provides much faster convergence thanks to the introduction of an additional encoder network used to learn a function that maps the original images into their respective latent space, which makes the expensive process of finding the inverse of the generator superfluous.

The idea behind the AE architecture, on the other hand, is to use an encoder to map the inputs to a latent space that is much smaller than the original one and which is then used as the input for a Decoder which tries to learn how to reconstruct the original input starting from this latent representation. The assumption is that the model should learn to map only the most important, or more common, features regarding the instances of the dataset leaving out every superfluous or specific information. Therefore it shouldn't be able to properly reconstruct anomalies, leading to a greater difference between input and reconstruction for this class of instances compared to the normal ones. For this reason, one possible approach to anomaly detection is to use an Autoencoder, usually trained only on the normal instances, to obtain a latent representation (which is typically much smaller than the original one) for each element of the dataset and then apply a clustering algorithm in order to discriminate between the good and the anomalous ones.

The limit of this approach is that it cannot be used for the localization of the anomalies but only for the detection. Many approaches based on AEs have been proposed over time with different characteristics related to their structure and type of loss functions adopted. Recent works specifically focused on Anomaly Segmentation[12] in images, showed the benefit of using a structural similarity-based loss (e.g. SSIM) to assess the quality of the reconstruction in substitution, or addition, to the pixel-wise one (e.g. MSE) adopted in previous works.

The second category of approaches is represented by all those systems that try to learn a representation of normality via a domain-specific optimization task. The simplest systems belonging to this category typically rely on the optimization of a distance-based task [132, 82, 122] that allows distinguishing between normal and anomalous instances at inference time. The drawback of optimizing a distance measure is that it becomes exponentially harder as the dimensionality of the input increases and therefore can be only applied to relatively simple problems or, as an alternative, it must be coupled with a dimensionality reduction step. An alternative is represented by one-class-classification systems, that typically involve the use of a Support Vector Machine (SVM) either on its own [104, 117] or paired with neural networks [77, 125], to learn the description of a set of data instances as to be able to detect whether new instances come from the same set (are normal) or not (are anomalous).

End-to-end anomaly score learning

End-to-end anomaly detection approaches share some similarities with the ones that try to learn a normality representation through the optimization of an anomaly measure. However, while the latter focus on synthesizing existing neural network models and anomaly measures, with all the inherent disadvantages they carry, the former aim at defining new loss functions to direct the learning of the anomaly score. There are four main types of approaches belonging to this category.

The first one is represented by ranking models, which aim at learning a ranking model to sort the data instances based on an ordinal value that represents their level of abnormality and that is used to drive the behavior of the anomaly scoring neural network [84, 87, 115]. Ranking models usually require the availability of some labeled anomalous instances which may not be available in certain application scenarios. Furthermore by relying on a small set of labeled anomalies the models belonging to this category may not generalize well to previously unseen ones.

A second type of approach is one relying on prior-driven models that use a prior distribution to drive the learning process for the anomaly scores [85, 80]. While these approaches provide a flexible framework that allows to incorporate of different priors into the anomaly scoring process and also potentially allows more interpretable results compared to other methods, the main drawback that characterizes them is that it's really hard, if not impossible to design an effective prior that generalizes to different application scenarios.

Softmax likelihood models represent yet another way of tackling the anomaly detection task. They work by maximizing the likelihood of events in the training data. The idea behind this category of approaches is that typically normal and anomalous instances are characterized by frequent and rare patterns respectively, therefore it's fair to assume that the former are high-probability events while the latter are low-probability ones [44]. The anomaly score that drives the learning of this category of approaches is then naturally defined as the negative of the event likelihood.

Finally, end-to-end one-class classification approaches learn to discriminate between the instances belonging or not to what is considered the normal class. Compared to more traditional one-class classification approaches the one belonging to this category does not rely on any existing one-class classification measures. One common example of an approach performing end-to-end one-class classification is represented by a subset of GANs that are trained to perform this task in an adversarial fashion by learning a discriminative criterion that allows to effectively discriminate between the normal

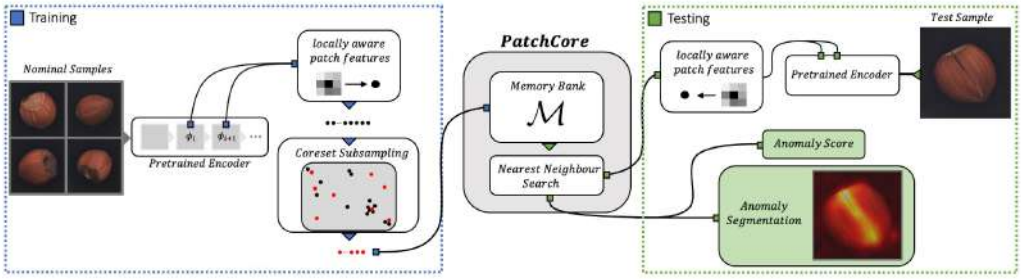


Figure 2.5: Visual representation of the Patchcore architecture [96]

instances, provided by the dataset, and the pseudo-anomalies generated by the Generator component of the network [136, 89, 98].

2.2.1 Patchcore

The Patchcore framework (Fig. 2.5), introduced in 2021 [96], represented a big step forward in the field of anomaly detection, becoming the state-of-the-art approach for this class of problems. Patchcore is a patch-based anomaly detection approach that builds on the idea that if a single patch is found to be anomalous then the whole image can be classified as anomalous. This framework consists of three main components that combined together lead to an efficient and robust architecture for the task of anomaly detection, namely: the adoption of local patch features aggregated into a memory bank. a coreset reduction method and the final anomaly detection and localization module.

The first step involves the adoption of a ResNet-based architecture, pre-trained on the popular Imagenet dataset [36], to extract a set of patch-level locally aware features from the input images. In particular, the extracted features come from the intermediate layers of the pre-trained network to avoid relying on features that are either too generic (first layers of the CNN) or too heavily biased towards a specific task, in this case, Imagenet classification (last layers of the CNN). Each patch is formally defined as the c^*-dimensional feature slice at position $h \in \{1, ..., h^*\}$ and $w \in \{1, ..., w^*\}$ of the feature

map $\phi_{i,j} \in \mathbb{R}^{c^* \times h^* \times w^*}$. Each patch is then enhanced by incorporating some local information from the neighboring ones, this allows for an increased robustness to small spatial deviation while preserving the spatial resolution of the feature maps. The way this goal is achieved is by incorporating into each patch representation the feature vectors from its neighborhood, leading to the following new patch definition:

$$\phi_{i,j}(N_p^{(h,w)} = f_{agg}(\{\phi_{i,j}(a,b)|(a,b) \in N_p^{h,w}\}) \tag{2.4}$$

where

$$\begin{aligned} N_p^{(h,w)} = \{(a,b)|a \in [h - [p/2], ..., h + [p/2]], \\ b \in [w - [p/2], ..., w + [p/2]]\} \end{aligned} \tag{2.5}$$

represents the feature vectors from the neighborhood of size p surrounding the patch, and F_{agg} is some aggregation function for the neighborhood, in this case, the one used is adaptive average pooling. For each feature map tensor $\phi_{i,j}$ we can now define its locally aware patch feature collection as:

$$\begin{aligned} P_{s,p}(\phi_{i,j}) = \{\phi_{i,j}(N_p^{(h,w)})| \\ h, w \bmod s = 0, h < h^*, w < w^*, h, w \in \mathbb{N}\} \end{aligned} \tag{2.6}$$

where s is an optional striding parameter that is typically set to 1. Finally, a memory bank containing the patch representations for all the normal (non-anomalous) training samples $x_{\in} X_N$, is created and defined as:

$$M = \bigcup_{x_i \in X_N} P_{s,p}(\phi_j(x_i)). \tag{2.7}$$

The main problem with the memory bank construction is that for increasing sizes of X_N it becomes too large to handle, leading to a drastic increase in inference time and storage space needed. This leads us to the second step in the Patchcore framework, represented by the reduction of the patch-feature memory bank to a reduced coreset. This process aims at finding a subset $S \subset A$ such that solutions over A can be more efficiently computed over

S while maintaining a close approximation to the original ones. The way the coreset (M_C^*) is obtained is by applying a greedy variant of the facility location coreset selection algorithm [105] (eq. 2.8) which is made even more efficient by the reduction in the dimensionality of the elements of M through random linear projections that allow preserving a close resemblance to the characteristics of the starting, higher dimensionality space [109].

$$M_C^* = \underset{M_C \subset M}{\operatorname{argmin}} \max_{m \in M} \min_{n \in M_C} ||m - n||_2 \tag{2.8}$$

Finally, through the patch-feature memory bank M an estimated image-level anomaly score s is produced for each test image. The way this score is obtained is by calculating the maximum distance between the feature vector of each patch in the test patch collection and the respective nearest neighbor $m^* \in M$, denoted as s^*, then this score is scaled based on the distance between m^* and the neighbor patches in M. The idea behind this scaling is that if m^* is an "outlier" itself with respect to the other patches in M then the patch we are currently analyzing, $m^{test,*}$, is more likely to be anomalous, consequently its anomaly score is increased. The final formula used to calculate the anomaly score is reported in Eq. 2.9, where $N_b(m^*)$ are the b nearest patch-features in m to test patch-feature m^*

$$s = \left(1 - \frac{\exp||m^{test,*} - m^*||_2}{\sum_{m \in N_b(m^*)} \exp||m^{test,*} - m||_2}\right) \cdot s^*. \tag{2.9}$$

Starting from Eq. 2.9 the image-level anomaly map can be computed by realigning the computed anomaly scores for each patch of the test image based on their spatial location and then upscaling the obtained map via bi-linear interpolation.

2.3 Document layout segmentation

Semantic segmentation in the context of document layout analysis, commonly referred to as document layout segmentation, is characterized by very specific challenges compared to its general-purpose counterpart, especially when the target documents are handwritten and ancient. The first key challenge is represented by the very high degree of similarity between the semantic classes we want to distinguish. In fact, since the main goal of this type of problem is to classify different types of text, the adopted models need to be very sensitive to small details and contextual information. Furthermore, when it comes to ancient manuscripts there is an added challenge introduced by the degradation of the document pages due to aging, ink stains, and scratches which coupled with a potentially low quality of digitally acquired instances makes the segmentation task even harder.

Many different approaches have been proposed to tackle the layout analysis, especially for handwritten historical documents. This section reviews some representative state-of-the-art methods for historical document image segmentation. In general, the techniques employed for document layout analysis are usually divided into three categories: bottom-up, top-down and hybrid [13].

The bottom-up strategy derives document analysis dynamically from smaller granularity data levels such as pixels and connected components. Then, the analysis grows up to form larger document regions and stops once it reaches a page segmentation into different regions with uniform elements. These techniques are flexible and do not require any prior knowledge of the layout structure. However, usually, they demand many labeled training data that is often not available, especially in the domain of historical documents where highly specialized expertise is needed to label the data.

On the contrary, top-down approaches assume that pages have a well-defined structure and layout. Various characteristics of the document page structure are then considered, such as white space between text regions, size of text blocks and the measures between main texts and paratext [33].

The page segmentation process then starts from the whole page and cuts it into areas to produce small homogeneous regions. In general, the top-down methods are easily applicable but not suitable for complex layouts such as handwritten historical documents. In addition, these methods depend on the layout structure of the document, so they have a low generalization capability.

Even though the research of this technique is well established, there are still many challenging issues that neither bottom-up nor top-down strategies can address appropriately. For this reason, the hybrid strategy has been identified and derives from the integration of the other two main categories [13]. Over the years, many techniques have been used to address this task, from classical computer vision algorithms to deep learning methods.

Chen et al. [20] used a convolutional autoencoder to learn the features directly from the pixel intensity values. Then, by using these features to train Support Vector Machines (SVM), this method got high-quality segmentation without any assumption of specific topologies and shapes of document layouts.

A different approach, which also allows performing layout analysis, was proposed by Mehri et al. [71] with the method based on learning texture features. This method used the simple linear iterative clustering super-pixels, Gabor descriptors, the co-occurrence matrix of the gray level, and an SVM to classify pixels into foreground and background. A super-pixel is a set of pixels that shares similar spatial and intensity information.

Many researchers have approached the page segmentation problem as a pixel labeling problem such as the work by Chen et al.[19]. In this paper, the features are learned directly from randomly selected image patches by using stacked convolutional autoencoders. With an SVM trained with the features of the central pixels of the super-pixels, an image is segmented into four regions. Finally, the segmentation results are refined by a connected components-based smoothing procedure. The authors show that by using super-pixels as units of labeling, the speed of the method is increased.

Following the same idea of [19], in Chen et al. [21] local features are

learned with stacked convolutional autoencoders in an unsupervised manner for the purpose of initial labeling. Then a conditional random field model is applied for modeling the local and contextual information jointly to improve the segmentation results. The graph nodes are represented by super-pixels, so the label of each pixel is determined by the label of the super-pixels to which it belongs.

[118] proposed a hybrid method for page segmentation problems. In the first stage, the text and non-text elements are classified by using a minimum homogeneity algorithm which is the combination of connected component analysis and multilevel homogeneity structure. Then, in the second stage, a new homogeneity structure is combined with an adaptive mathematical morphology in the text document to get a set of text regions. [33] proposed a novel method for document layout analysis that reduces the need for labeled data. This method is a dictionary-based feature learning model where a sparse autoencoder is first trained in an unsupervised manner on a document's image patch. Then, the latent representation of image patches is then used to classify pixels into various region categories of the document using a feed-forward neural network. Also, [4] used the patching of the document image to train a siamese network model that takes in input a pair of patches and gives as an output a distance that corresponds to their similarity. The trained model is also used to calculate a distance matrix which in turn is used to cluster the patches of a page as either main text, side text, or a background patch. [112] tackle the problem of the limited presence of annotated data by introducing the use of pre-trained segmentation models on images from a different domain and then fine-tuning them on historical handwritten documents. The results demonstrated that on some manuscripts pre-training on ImageNet increases the performance, but on others, the pre-trained network performs much worse. Finally, [116] propose the few-shot learning approach Deep&Syntax to segment historical handwritten registers. Their work uses a hybrid system that exploits recurring patterns to delimit each record, combining U-shaped networks and logical rules such as filter and text alignment.

While the presented approaches have different degrees of effectiveness when trying to solve the document layout segmentation task, they all rely on large amounts of data for their training. The main contribution we bring with the present work is the ability to achieve similar, or even better performance while relying on just a fraction of the available data.

2.3.1 MLA

In 2018 Xu et al. proposed a Multi-task Layout Analysis framework (MLA) for the layout analysis of historical handwritten documents which established the current state of the art in this research field [127].

The proposed framework is based on a Fully Convolutional Network (FCN) for semantic segmentation trained on 4 different tasks concurrently. While the first 5 convolutional modules of the network are kept consistent with previous works, the authors introduced three main modifications to the architecture. The first one is represented by a stronger focus on low-level features which are concatenated to the final feature map extracted by the network (FIG 2.6, Stage 2). This choice is motivated by the idea that when dealing with text classification tasks, where the difference between the various elements we want to partition lies in very small and subtle details, we are more interested in features that represent simple characteristics as opposed to the ones extracted by the deeper layers of the network, which usually capture more complex structures.

Furthermore, the depth of the network is increased by adding three additional 3×3 convolutional layers on top of stage 5, in order to extract feature maps characterized by a larger receptive field. The reason behind this is that when it comes to document layout segmentation contextual information is very important as, for example, the relative position between text blocks can indicate to which category they belong.

Finally, as previously stated, instead of training the network on just the task of layout segmentation three more heads are introduced to leverage the full annotation information provided by the dataset. The first head is

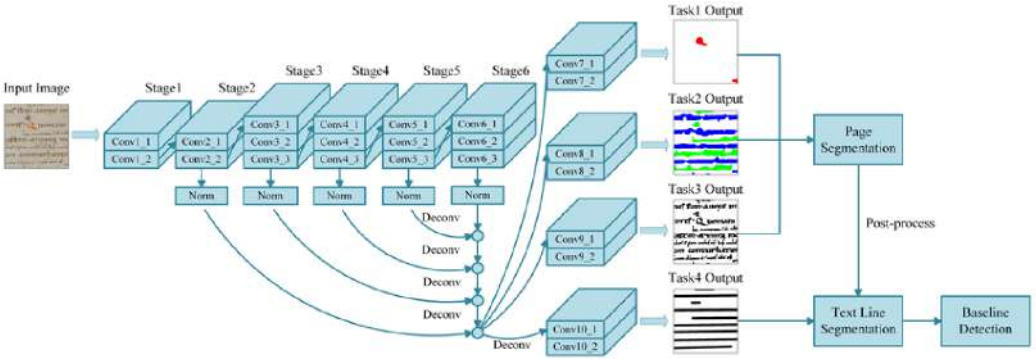


Figure 2.6: Multi-task Layout Analysis framework [127]

specialized in the segmentation of decoration regions, which represent the only layout class not characterized by textual information. The second head focuses on the recognition of the main text and comment regions which typically represent the bulk of the content of a document page. These first two heads not only extract the category information but also provide a coarse contour outside of the text lines. The third head is trained to perform a binarization task on the document page, to separate the foreground pixels from the background ones. Finally, the last head performs the text line segmentation task or, in other words, learns the center line of each line of text.

The way the segmentation task is performed is thus by combining the predictions of the first three heads, in particular, the coarse segmentation masks produced by heads one and two are combined with the binarization masks produced by head three to obtain a more precise, segmentation mask. However, at this stage, the obtained mask is still characterized by noise and misclassified regions. For this reason, some additional refinement steps are performed. The first one is based on the observation that small isolated regions are typically characterized by the same class as their surroundings and also, the length of contacted boundary between the elements belonging to the main text and comments classes is short. Therefore, supposing C_a and C_b are two adjacent connected components (CC) belonging to different classes (A and B respectively) and let L_a be the boundary length of C_a and

L_{ab} the length of the boundary touching both C_a and C_b then, if $L_{ab} \geq L_a/3$, C_a will be considered an isolated CC surrounded by class B. At this point, a window of fixed size around the CCs is considered and each pixel inside it is converted to the predominant class.

As a second refinement step, each foreground CC is analyzed and if 80% of its pixels are classified as a decoration then all the remaining ones are also converted to the decoration class.

Finally, all the CCs with an area smaller than a predefined threshold value are removed and considered as background to reduce the noise.

3

Bringing attention to image anomaly detection

In the following chapter we analyze the effects of the introduction of an attention-based mechanism in a traditional reconstruction-based anomaly segmentation framework. Furthermore, a novel masking approach for the input image is adopted to prevent the accurate reconstruction of anomalous regions thus improving the anomaly localization capabilities of the system.

3.1 Introduction

In this chapter we are presenting an attention-based approach built upon the RIAD Framework [130] which leverages a patchwise inpainting process in order to detect and localize anomalies in images. In particular, while keeping the patchwise structure of said approach, we modify the original reconstruction module of the framework by introducing an Attention U-Net model in place of the original U-Net. Furthermore, we propose a new approach for the masking process of the images which works on a higher number of overlapping patches at a single scale of the image instead of using a multiple scale approach. Finally, we also propose a new evaluation setup that uses both an L2 and a multi-scale GMS loss, instead of relying just on the latter, to generate the anomaly maps. We show how the combination of these ideas significantly increases the overall performance of the model on the adopted

dataset for the metrics considered.

The rest of the chapter is organized as follows: in section 3.2 we give an overview of the proposed framework by describing in detail the masking process and the attention-based reconstruction approach that characterizes it. In section 3.3 we provide a comparison of our model with its baseline counterpart as well as with other common approaches for anomaly localization. Finally, section 3.4 is dedicated to the final remarks together with an overview of possible future works in this area.

3.2 Methods

The approach we propose in the present chapter is based on the framework presented in [130]. As in the original work, we try to address the problems of anomaly detection and localization via an inpainting approach that aims to reconstruct the missing parts of an image. The idea is to generate a set of random masks for each image before feeding it to the model, which then tries to reconstruct only the masked-out regions hopefully ignoring the anomalies potentially contained in them, thus increasing the difference between the original images and the reconstructed ones specifically in those anomalous regions. The two main novelties we present in this work are, respectively, the introduction of an attention mechanism in our reconstruction process through the adoption of an Attention U-Net [81] instead of the original standard U-Net, and the adoption of a new masking process which will now be described in detail.

3.2.1 Masked regions generator

The masking approach we are proposing in the present works differs from the one used in the original work [130] in the fact that, instead of masking out non-overlapping patches at multiple scales of the original image it focuses on patches of a single size. Our approach works by selecting N random, potentially overlapping, masks consisting in a set of $k \times k$ regions which in total

make up for the P percent of the total area of the input image, by setting the corresponding pixels to a value of 0. The masked instances are then fed to the model which tries to reconstruct the missing parts. Finally, the average of each reconstructed region of the input image is taken to determine the output of the model. The main intuition behind this approach is that by reconstructing each patch of the original instance starting from different starting visible regions the robustness of the reconstructed image should improve, making it harder for it to contain the reconstruction of any eventual anomaly present in the corresponding input.

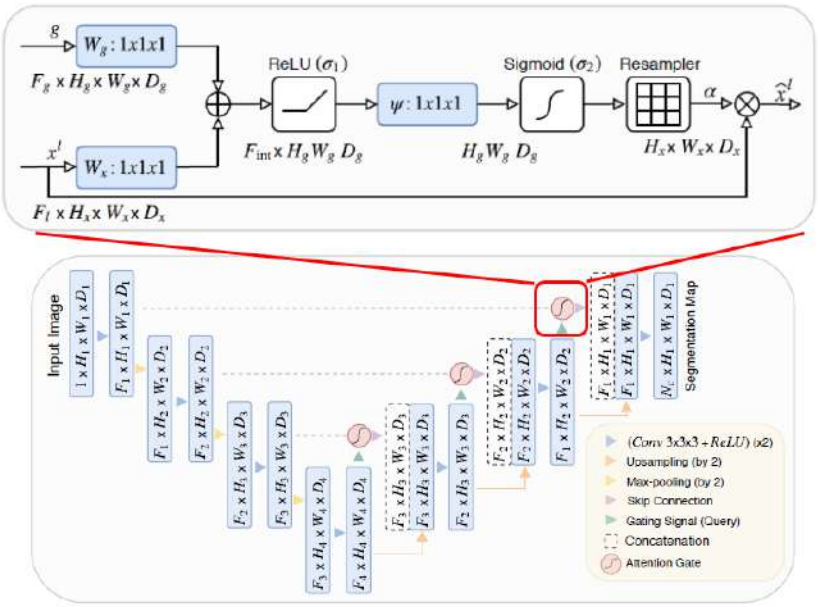


Figure 3.1: Scheme representing the function of the attention gates for the Attention U-Net model [81]. The input features (x^l) are scaled with the attention coefficients (α) computed in the AG. These coefficients are obtained through the use of additive attention computed between the input feature and the gating signal (g) which is collected from a coarser scale and provides the needed contextual information.

3.2.2 Model

The model we adopted for the Reconstruction module in our framework is represented by the popular Attention U-Net. This deep learning model was initially introduced [81] as a way to address the problem of Multi-class image segmentation in biomedical images. The Attention U-Net differs from its original counterpart for the introduction of attention gates (Fig.3.1) in the upsampling part of the network, which takes as their inputs a set of input features, represented by the output of the previous layer of the model (called the "gating signal") and the feature map obtained from the skip connection, which also defines the size of the output of the gate. This way a set of attention coefficients are learned in order to identify salient regions in the feature maps, relative to the considered task.

3.3 Experiments

3.3.1 Datasets

The dataset we choose for the training and testing processes of the presented model is represented by the MVTec Anomaly Detection dataset [8]. This dataset, thanks to its heterogeneity, has become one of the most common benchmarks for works that try to address the problem of anomaly detection and localization in images. MVTec consists of 3629 training images and 1725 test images distributed over 15 classes, 10 of which represent different products while the remaining ones cover 5 different types of textures. The classes have been chosen in order to provide heterogeneous characteristics both regarding the appearance of the elements they represent and the type of anomalies by which they can be affected. The way in which the images have been collected is also heterogeneous as for some of the classes all the instances tend to be roughly aligned while for some others a random rotation is introduced. Moreover, three of the classes are characterized by the presence of grayscale images only, as this is a common occurrence in real-world indus-

trial settings. As previously mentioned, the testing set also presents a high degree of heterogeneity. For each category, an average of 5 different types of anomalies is provided for a total of 73 anomaly types across the dataset. Furthermore, MVTec does not only provide the labels for defective images at an instance level but also provides pixel-perfect Anomaly Maps which allow us to assess the quality of our model also on the anomaly localization task. A set of samples for the MVTec dataset is provided in Fig.3.2.

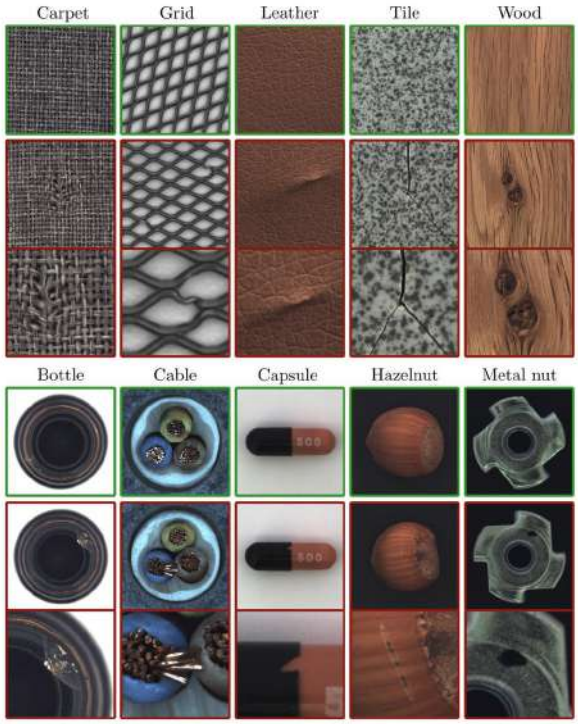


Figure 3.2: Samples for 5 texture classes and 5 product classes of the MVTec dataset [9]. For each category are shown a normal instance (top row), an anomalous one (middle row), and a close-up of the anomalous region (bottom row)

3.3.2 Training Setup

The training of the model was performed using only the normal instances of the dataset, through a self-supervised approach. The maximum number of

epochs allowed was set to 300, which proved to be enough for each model to converge, with an early stop in case the performance on the validation set wouldn't improve over the last 20 epochs. The total loss used to evaluate this performance was calculated as the sum between the Mean squared Error (MSE) loss, the Gradient Magnitude Similarity (GMS) loss, and the negative of the Structured Similarity Index (SSIM) between the original and the reconstructed images. The combination of these losses allows for focus both on the overall structure and the smaller details of the images. A formal description of the loss functions is given hereafter:

$$L_2(X, \hat{X}) = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} (X_{ij} - \hat{X}_{ij})^2 \tag{3.1}$$

$$L_{SSIM}(X, \hat{X}) = -\frac{(2\mu_X \mu_{\hat{X}} + c_1)(2\sigma_{X\hat{X}} + c_2)}{(\mu_X^2 + \mu_{\hat{X}}^2 + c_1)(\sigma_X^2 + \sigma_{\hat{X}}^2 + c_2)} \tag{3.2}$$

$$L_{GMS}(X, \hat{X}) = \frac{2g(X)g(\hat{X}) + C}{g(X)^2 + g(\hat{X})^2 + C} \tag{3.3}$$

$$L(X, \hat{X}) = \alpha * L_1(X, \hat{X}) + \beta * L_{SSIM}(X, \hat{X}) + \gamma * L_{GMS}(X, \hat{X}) \tag{3.4}$$

where:

- **X:** Is the original image
- $\hat{\boldsymbol{X}}$: Is the reconstructed image
- **h, w:** Are the height and width of the image in pixel
- $\boldsymbol{\mu_x}$: Is average value of image x
- $\boldsymbol{\sigma_x^2}$: Is variance of image x
- $\boldsymbol{\sigma_{xy}}$: Is the covariance of x and y
- $\boldsymbol{c_1}$**&**$\boldsymbol{c_2}$ **:** Are two variables used to stabilize the division with weak denominator

- $\mathbf{g(X)}$ is the gradient magnitude map for image X calculated as: $g(X) = \sqrt{(G * h_x)^2 + (G * h_y)^2}$ with h_x and h_y being $3x3$ filters along the x and y dimensions respectively and $*$ being the convolution operation.

- $\boldsymbol{\alpha, \beta, \gamma}$: are the weights of every single loss in the total loss function, for this work we kept them all equal.

As we can notice the SSIM is negated in the total loss function, the reason behind this is that it represents a similarity measure defined in the interval $[-1, 1]$ where a value of 1 indicates that the compared images are completely identical, therefore by keeping the obtained values as they are we would have a maximization problem instead of a minimization one.

Finally, in Tab.3.1 we provide the hyper-parameters employed for the model during the training process. The reported values have been selected empirically by trying to provide a good compromise between the quality of the predictions and the complexity of the model.

Table 3.1: model hyperparameters

LR	**Weight decay**	**Batch**	**Img size**	**# of masks**	**Patch sizes**	**Masked region (%)**
1eˆ-4	1eˆ-4	16	128x128	20	16x16	33

3.3.3 Anomaly localization process

Another area in which we departed from the original RIAD framework is the one regarding how anomalous regions of the images are detected. Instead of relying on just the anomaly maps obtained by calculating the gradient magnitudes similarity between the input and the reconstructed images, we combined these with the anomaly maps resulting from the normalized, pixel-wise, MSE between the two images by summing the values of each of their pixels. The idea behind this choice is that by taking in consideration different similarity measures between the original and the reconstructed images the reliability of the model in localizing the anomalies should improve.

3.3.4 Metrics

For the evaluation process of the performance of the model on the selected dataset, we opted to use two metrics. The first one is the Area Under the ROC Curve (AUROC), which plots the False Positive Rates obtained by the model versus its True positive Rates and then calculates the area under the resulting curve. This measure is a common metric for the addressed problems in recent works. Instead of considering the whole curve, we decided to set an upper limit for the FPR to 0.3 as proposed in [8]. This is because results with a high FPR tend to lead to meaningless detection and segmentation results, especially in real-world scenarios where they would lead to the rejection of many products not presenting any real defects. The second metric we adopted is the Area Under the Precision-Recall Curve, which has been chosen because it is particularly well suited for those problems where we are more interested in one specific class of the instances as it happens in the context of anomaly detection. Furthermore, another important property of this metric is that it is not affected by data imbalances in the test set.

3.3.5 Results

Hereafter we present the results achieved by our model on the MVTec dataset for the 2 different metrics considered, and show how they compare to the baseline on which our work is based, namely RIAD [130], as well as to other widely used approaches, represented by the ones reported in the most recent MVTec paper [8] from which we selected, for a fair comparison only those who didn't rely on additional data for pre-training. We additionally include, for the results regarding the ROCAUC score, a comparison with VT-ADL [74] which is of particular interest as it represents an alternative way of introducing the attention mechanism for the task of anomaly localization in images. In Table 3.2 and Table 3.3 are reported the aforementioned comparative results, for the ROCAUC and PRAUC metrics respectively. While the scores for the VT-ADL and baseline RIAD approaches have been calculated by us,

which is of particular relevance especially for the latter as it allowed us to really analyze and understand the power represented by the introduction of the attention mechanism, for all the other models they have been gathered from the original paper.

Table 3.2: Normalized area under the ROC curve up to an average false positive rate per pixel of 30% for each dataset category. The values in bold represent the best scores overall.

Class	f-anoGan [101]	l2-AE [11]	SSIM-AE [11]	Texture inspection [17]	Variation model [18]	VT-ADL [74]	RIAD [130]	**Ours**
Carpet	0.251	0.287	0.365	**0.874**	0.162	0.549	0.803	**0.906**
Grid	0.550	0.741	0.820	0.878	0.488	0.569	**0.966**	0.932
Leather	0.574	0.491	0.356	0.975	0.381	0.817	**0.983**	**0.983**
Tile	0.180	0.174	0.156	0.314	0.304	0.589	0.599	**0.719**
Wood	0.392	0.417	0.404	0.723	0.408	0.682	0.745	**0.768**
Bottle	0.422	0.528	0.624	0.454	0.667	0.687	0.893	**0.926**
Cable	0.453	0.510	0.302	0.512	0.423	0.751	0.665	0.**854**
Capsule	0.362	0.732	0.799	0.698	0.843	0.615	**0.965**	0.951
Hazelnut	0.825	0.879	0.847	0.955	0.802	0.926	0.944	**0.954**
Metal nut	0.435	0.572	0.539	0.135	0.462	0.711	0.706	**0.913**
Pill	0.504	0.690	0.698	0.440	0.666	0.748	**0.919**	0.903
Screw	0.814	0.867	0.885	0.877	0.697	0.771	0.881	**0.945**
Toothbrush	0.749	0.837	0.846	0.712	0.775	0.878	**0.974**	0.969
Transistor	0.372	0.657	0.562	0.363	0.601	0.689	0.731	**0.886**
Zipper	0.201	0.474	0.564	0.928	0.209	0.683	**0.951**	0.939
Mean	0.472	0.590	0.584	0.656	0.526	0.683	0.848	**0.893**

As we can see for the ROCAUC metric the proposed model vastly outperforms all the other ones, achieving margins going from 4.5% to 42.1% on the mean value for the baseline RIAD and f-anogan models respectively, with a significant improvement (21%) also over VT-ADL, the only other approach using an attention-based mechanism. While the margin from the former is not as large as the one from other models it is definitely noteworthy and representative of the power of the attention mechanism. This aspect is also accentuated by the fact that our model achieved the best performance on most of the classes (10 out of 15) across all models. Regarding the PRAUC scores, on the other hand, we can see that, while the results achieved by our approach aren't as markedly better than the ones reported for the other approaches on the individual classes, where it obtains the top score in 6 out of 15 of them, it still proves to be the best-performing model overall, again with a significant margin over the competition. Is it interesting to observe

Table 3.3: Normalized area under the Precision-Recall Curve for each dataset category. The values in bold represent the best scores overall.

Class	f-anoGan	l2-AE	SSIM-AE	Variation model	Texture inspection	RIAD	**Ours**
Carpet	0.025	0.042	0.035	0.017	**0.568**	0.223	0.292
Grid	0.050	0.252	0.081	0.096	0.179	**0.278**	0.208
Leather	0.156	0.089	0.037	0.072	**0.603**	0.600	0.555
Tile	0.093	0.093	0.077	**0.218**	0.187	0.157	0.198
Wood	0.159	0.196	0.086	0.213	**0.529**	0.322	0.290
Bottle	0.160	0.308	0.309	0.536	0.285	0.560	**0.571**
Cable	0.098	0.108	0.052	0.084	0.102	0.122	**0.369**
Capsule	0.033	0.276	0.128	0.226	0.071	0.184	**0.317**
Hazelnut	0.526	0.590	0.312	0.485	**0.689**	0.444	0.580
Metal nut	0.273	0.416	0.359	0.384	0.153	0.257	**0.712**
Pill	0.121	0.255	0.233	0.274	0.207	**0.567**	0.487
Screw	0.062	0.147	0.050	0.138	0.052	**0.163**	0.138
Toothbrush	0.133	0.367	0.183	0.416	0.140	0.456	**0.492**
Transistor	0.130	0.381	0.191	0.309	0.108	0.244	**0.525**
Zipper	0.027	0.095	0.088	0.038	**0.611**	0.605	0.312
Mean	0.136	0.241	0.148	0.234	0.299	0.339	**0.403**

in particular how the improvement over the baseline RIAD model is even more significant for the PRAUC metric (6.4%), which can be considered a stricter evaluation metric, than it is for the formerly analyzed ROCAUC one (4.5%). Finally it's important to mention that the computational complexity of the proposed model is comparable to the original RIAD one as both the new mask generation process and the introduction of the attention gates introduce very little overhead.

3.4 Conclusions and Future Works

In the presented work we investigated the potential of the attention mechanism, and of an efficient masking process, in the context of Anomaly Localization, specifically through the introduction of an Attention U-Net as the reconstruction module for the already effective RIAD Framework, which used a traditional U-Net for this task. In particular, we have shown how the

proposed approach is able to outperform other more traditional models on the popular benchmark MVTec for the metrics considered, furthermore, it provides a significant improvement in performance over its baseline counterpart which doesn't leverage the power of the Attention mechanism, showing the effectiveness of the latter. Even though the results obtained with this work clearly show the potential of attention-based approaches for anomaly detection, we think that further investigation in this area could lead to even more interesting outcomes. In future works, we would like to build an entire framework that fully leverages the attention component as opposed to simply introducing it in a preexisting one. Furthermore, we believe that a higher degree of heterogeneity in the datasets considered for the benchmarking process would definitely provide a more complete picture of the capabilities of this approach in different real-world scenarios.

4

Masked Transformer for image Anomaly Localization

Most of the current deep learning approaches rely on image reconstruction: the input image is projected in some latent space and then reconstructed, assuming that the network (mostly trained on normal data) will not be able to reconstruct the anomalous portions. However, this assumption does not always hold. We thus propose a new model based on the Vision Transformer architecture with patch masking: the input image is split into several patches, and each patch is reconstructed only from the surrounding data, thus ignoring the potentially anomalous information contained in the patch itself. We then show that multi-resolution patches and their collective embeddings provide a large improvement in the model's performance compared to the exclusive use of the traditional square patches.

4.1 Introduction

Most of the models used in recent years to tackle the anomaly detection problem make use of Convolutional layers which exploit the typical characteristics of images by detecting progressively more complex features starting from the most basic ones (e.g. edges). In particular, there are two kinds of architectures, with their respective variations, which gained a lot of popularity for their effectiveness in dealing with this kind of problem, which are

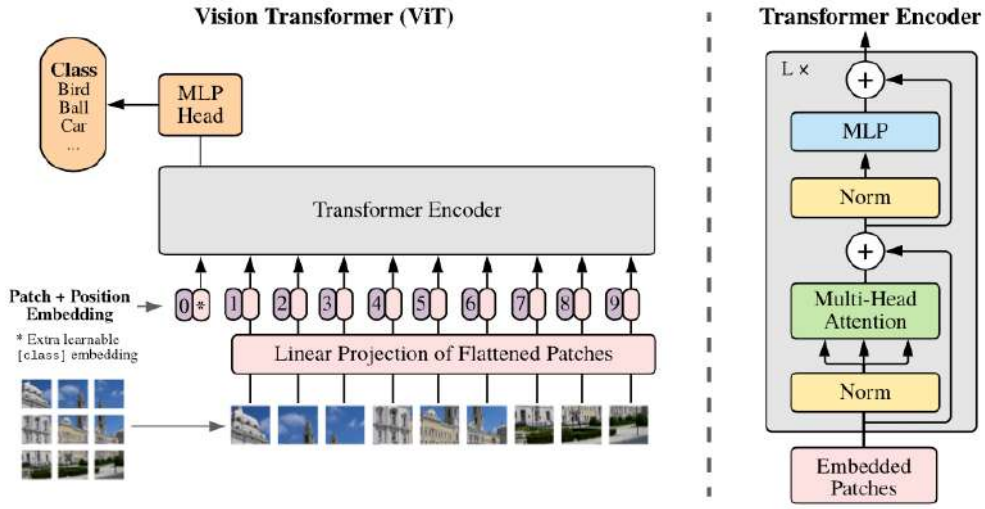


Figure 4.1: Vision Transformer architecture [40]

Autoencoders [28] and Generative Adversarial Networks (GANs) [131]. One problem with classical image reconstruction approaches though, is that they use the information extracted from the whole image to obtain the output image, which can lead the model to be able to reconstruct also the anomalies and therefore to not be able to identify them.

In this chapter, we propose a new approach a novel method for image Anomaly Detection, called Masked Transformer for image Anomaly Localization (MeTAL), that adopts as its backbone the recently presented Vision Transformer (ViT) architecture [40], which instead of leveraging the prior knowledge granted by the use of convolutional layers, is characterized by the adoption of a masked multi-head self-attention mechanism that allows the model to learn a relationship between different patches of the input images. In particular, with the present work, we introduce two main novelties to the original architecture. the first one regards a new masking component we added to the multi-head self-attention module of the ViT encoder which allows us to reconstruct each patch of the image without using any information coming from the patch itself but using only the information extracted from the surrounding patches based on the importance given to each of them

by the attention module. The second idea we present regards, instead, the way patches are generated from the original image. In particular, instead of relying on just the square patches as in the original work presenting ViT, we introduce the idea of calculating attention between patches of different shapes, which are then combined to obtain the final image reconstruction. As we will show in section 4.4 both ideas resulted in an improvement in performance over the baseline model for the task at hand.

Furthermore, we show that the Vision Transformer architecture is a valid option for anomaly detection problems and can be adopted effectively even in scenarios where the amount of data available is relatively small without necessarily relying on a pre-training procedure.

The rest of the chapter is organized as follows. In section 4.2 we give a brief introduction to the Vision Transformer architecture. Then in section 4.3, a detailed overview of the training process is given together with a thorough description of the proposed architecture. The obtained results are outlined in section 4.4 where a more in-depth description of the adopted dataset is also provided. Finally, in section 4.5 we summarize our work and discuss our ideas for future work.

4.2 Vision Transformers

The Vision Transformer is a deep neural network architecture proposed by Dosovitskiy et al. [40] as an alternative to convolutional-based architectures for computer vision applications. This model builds upon the idea of Self-Attention introduced in the original Transformer paper [120], which has since become the model of choice for Natural Language Processing (NLP) Applications, replacing Recurrent Neural Networks.

Architecture:

The Vision Transformer architecture (Fig. 4.1) differs from the original transformer one in the fact that it only uses the encoder module leaving out the

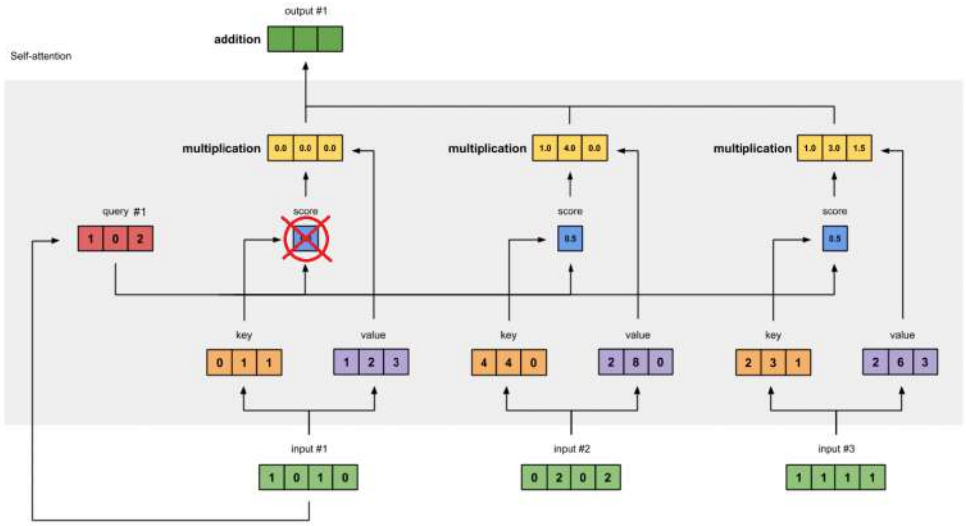


Figure 4.2: Illustration of the masking process performed in the self-attention module of our model. In the example we are calculating the attention values for patch #1, therefore we set the dot product between Query #1 and Key #1 to 0 in order to take into account only the attention values of the remaining patches

Decoder. The encoder module, which takes as its input a set of flattened representations of the different patches composing the image we are trying to analyze, consists of a Stack of N identical Layers each containing two sub-layers: the first one is a multi-head self-attention block while the second one is a fully connected feed-forward layer. Around each of the 2 blocks, a residual connection is applied, followed by a layer normalization step. The role of the encoder module in the ViT architecture is that of learning a correlation between the different patches composing an image. In order to preserve the spatial information regarding the position of the different patches a positional embedding is added to the patches representations before feeding them to the ViT encoder. The positional embedding can be fixed or learned alongside the other parameters.

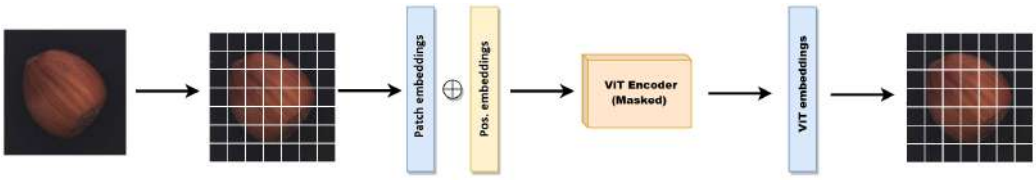


Figure 4.3: Overview of our model in its basic form: the original image is split into patches of the same size which is used as the input of a Multi-Layer Perceptron (MLP) in order to get the corresponding embedding. Then, to each patch embedding, a position embedding is added. The resulting tensor is then processed through a masked ViT encoder module in order to obtain the final embedding which will be used to reconstruct the respective patch by using an MLP Decoder

Multi-head Attention:

The attention mechanism can be described as a function that maps triplets of vectors, represented by a query (Q) and key-value (K, V) pairs, to an output which is computed as a weighted sum of the values where the weights of each value are computed based on a compatibility relationship between a query and the corresponding key. In Vision Transformers instead of performing the self-attention step only once for each set of queries, keys and values, we project them in h different spaces via learned linear projections. Attention is then calculated for each of these different projections and the final outputs are concatenated and projected again to obtain the final values. The whole self-attention calculation process can be summarized by the following equation:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{4.1}$$

This approach allows the model to consider different representation subspaces when calculating attention between the different parts of the image.

4.3 Methods

4.3.1 Data preparation

As far as data preparation is concerned we tried to adopt a very minimal approach. The only way the images themselves are pre-processed is through a resizing process that brings them all to the same final size, which is 128x128 for all the classes. This step was necessary in order to have a consistent number of patches when dividing the images in the next steps and it was also helpful for reducing the overall complexity of the model. The only other operation applied to the dataset before training is a shuffling of the instances in order to avoid any potential bias for the model based on their order.

4.3.2 Model description

The model on which the work for this chapter is based on the classical Vision transformer architecture presented in the previous section.

In this work, however, we introduce two very important changes to the aforementioned architecture which we will describe in detail in the present section. The first change is represented by a masking process by which we try to remove the focus of each patch of the original image on its own features, redirecting the attention to the remaining ones. While the second idea we introduced regards the division of the original images into multi-shaped and multi-scale patches, as opposed to relying only on the traditional square patches presented in the original work.

Masking process

While many of the image-reconstruction-based models for anomaly detection work on the entire image, this approach leads to a very common problem. Since the model uses all the information available in the input it tends to learn how to reconstruct correctly the anomalous images as well as the normal one which, of course, is not the desired behavior since the goal is to discriminate

between the two classes. For this reason, the idea behind our proposed method is to mask out some of the information of the original image and use only the remaining data to perform the reconstruction. This as we will show, greatly reduces the problem mentioned in the previous paragraph. More specifically we decided to work on patches representing non-overlapping subsets of the original image and to reconstruct each of these patches based only on the content of the remaining ones. For this purpose, we decided to adopt the ViT architecture as a baseline for our model, which as we have previously shown, allows finding a correlation between different parts of an image by calculating an attention score between its patches. In its basic form, though, the ViT model provides for each patch an embedding obtained by processing the whole information of the original image, including the patch itself. For this reason, we altered the concept of self-attention introduced by the original model in order to mask this piece of information. In order to do so, we added a masking module in the multi-head self-attention component of the original architecture which forces the dot product between the key generated for each patch and the respective query to be set to 0, as shown in Fig. 4.2. In other words, after obtaining the nxn (where n is the number of patches) matrix in which the cell in position ij represents the correlation between patches i and j, we set its diagonal, representing the internal information of each patch, to 0 in order to use only external information for its reconstruction. An overview of the model described so far is given in Fig. 4.3.

Multi shape patch structure

Another direction in which we expanded the original idea of the ViT model regards the way the patches fed to it are obtained from the original image. While the use of square patches is a common choice, it is ultimately an arbitrary one, for this reason, we introduced in our model a set of patches with different shapes, in particular horizontal and vertical stripes, each of which was processed in the same way as the square ones. The encodings of

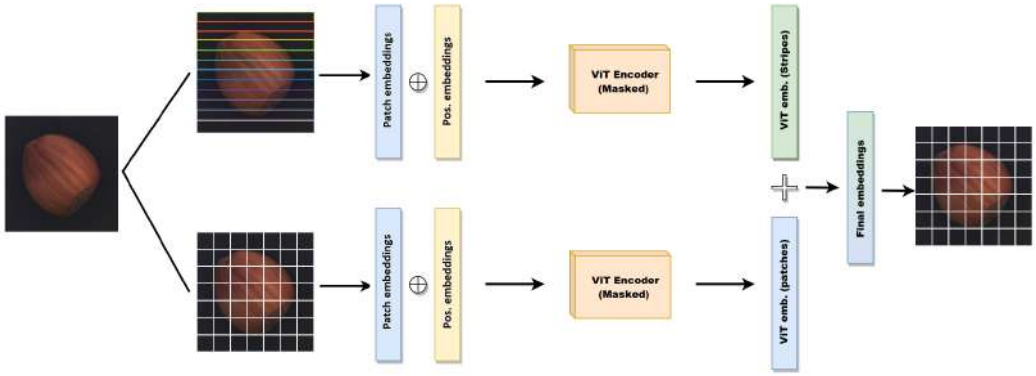


Figure 4.4: Overview of the proposed model using horizontal stripes only, in addition to the square patches

the different types of patches have then been concatenated to form the final representation for each patch which has then been used for its reconstruction. The reconstruction process involved the use of a simple MLP Decoder which worked on each patch embedding singularly to reproduce the respective patch. This process has been carried out by ensuring that the masking property of the network was held true, to do so the concatenation between patches of different shapes has been carried on based on their spatial location in the original image so that each square patch was fully contained in the respective horizontal (or vertical) stripe patch. Furthermore, in order to keep the final embedding size as small as possible, we split the embedding of each stripe into p segments of the same size, where $p = N/K$ (with N=size of the image and K=size of the square patches) is the number of square patches contained in each stripe, and then concatenated each of these segments to the embedding of a different patch thus forcing the model to learn specific information about each of them in different locations of the stripe embedding (Fig. 4.5). A high level overview of our final model is provided in Fig. 4.4.

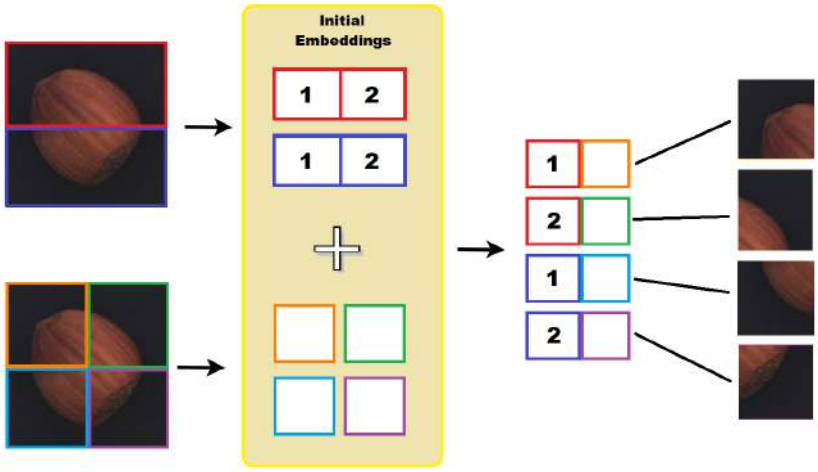


Figure 4.5: Illustration of the process carried out for the concatenation of the patch embeddings. Two components are concatenated together, the first one is the embedding obtained through the self-attention process performed on square patches, while the second one is a subset of the embedding resulting from the self-attention process carried out on strip patches.

4.4 Evaluation

4.4.1 datasets

To test our Framework we relied on two popular datasets, namely MVTec which is currently one of the most popular and heterogeneous datasets in the context of anomaly detection for industrial quality control, and HeadCT which is instead a dataset representing a collection of X-Ray head scans in which the anomalies are represented by the presence of hemorrhages. While the use of larger datasets is usually preferable, the options available in the context of anomaly detection are still very limited. MVTec already represents a big step forward when it comes to the size and heterogeneity of datasets in this area of research compared to the ones that were available before it and, combined with the HeadCT dataset, we believe they provide and reliable option for the evaluation of the proposed approach.

The MVTec Dataset

As in the previous chapter he dataset we used for both the training and testing of our model is the MVTec Anomaly Detection dataset [9], since it is the de-facto benchmark in recent works on anomaly detection and localization. In addition to the already provided description regarding this dataset, inn Table 4.1 the details regarding each class are reported. As we can notice there are no anomalous instances in the training set, this is because usually the anomaly detection models are trained only on normal data.

Table 4.1: MVTec dataset details

	Class	Train (Normal)	Test (Normal)	Test (Anomaly)	Image side
Textures	**Carpet**	280	28	80	1024
	Grid	264	21	57	1024
	Leather	245	32	92	1024
	Tile	230	33	84	1024
	Wood	247	19	60	1024
Products	**Bottle**	209	20	63	900
	Cable	224	58	92	1024
	Capsule	219	23	109	1000
	Hazelnut	391	40	70	1024
	Metal nut	220	22	93	700
	Pill	267	26	141	800
	Screw	320	41	119	1024
	Toothbrush	60	12	30	1024
	Transistor	213	60	40	1024
	Zipper	240	32	119	1024

Head CT Dataset

As a further benchmark for our model, we used the Head CT dataset [60], which we selected because of its medical nature and because of its relatively

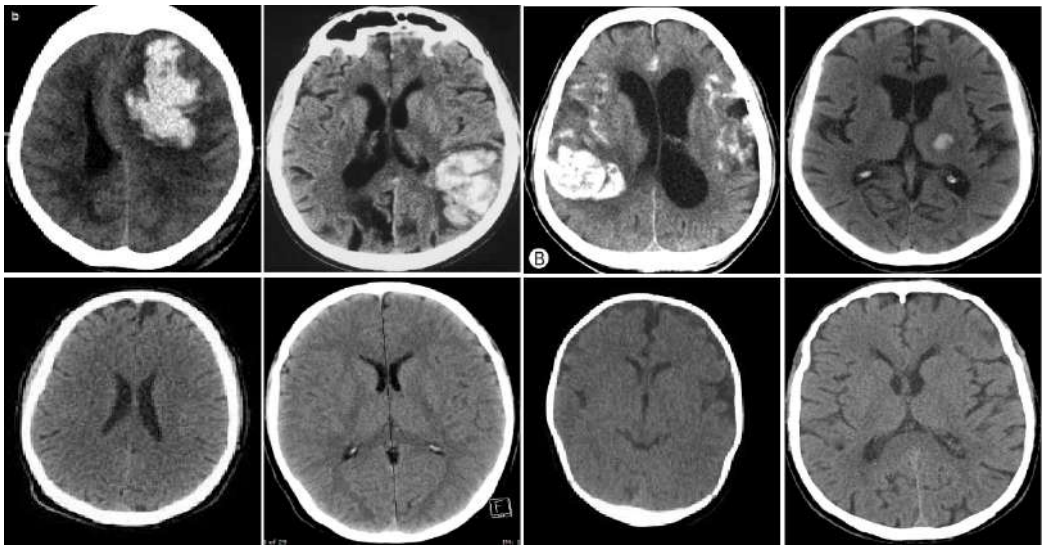


Figure 4.6: Samples of anomalous (top row) and normal (bottom row) images from the Head CT dataset

small size, which allowed us to prove the potential of the Vision Transformer architecture even in this particularly challenging type of setting. The head ct dataset is, in fact, composed of a total of 200 images 100 of which represent head ct of healthy individuals while the remaining 100 represent scans of patients with a head hemorrhage. For the purpose of this study, we adopted 80 normal images for the training of the model, while the remaining 120 instances (20 normal, 100 with an hemorrhage) were used in the testing process. Some samples representing normal and anomalous images from the dataset are reported in Fig. 4.6.

4.4.2 Training setup

The training of the model was performed in a self-supervised fashion, using only the normal images of the dataset, over a maximum of 3000 epochs with an early stop introduced after epoch 500 which would trigger when the performance of the model on the validation set didn't improve in the previous 50 epochs. The validation set was composed of 10% of the images present in the training data of the datasets. While often larger percentages are used,

both MVTec and HeadCT are relatively small datasets, which led us to prefer keeping as much data as possible for the training of the model. The total loss function we used is obtained by summing the L1 loss and the negative of the SSIM Similarity, as defined in [124], calculated between the original image given as input and the reconstructed image returned as the output of the model. A formal description of the two functions is given hereafter:

$$L_1(X, \hat{X}) = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} |X_{ij} - \hat{X}_{ij}| \tag{4.2}$$

$$L_{SSIM}(X, \hat{X}) = -\frac{(2\mu_X \mu_{\hat{X}} + c_1)(2\sigma_{X\hat{X}} + c_2)}{(\mu_X^2 + \mu_{\hat{X}}^2 + c_1)(\sigma_X^2 + \sigma_{\hat{X}}^2 + c_2)} \tag{4.3}$$

$$L(X, \hat{X}) = L_1(X, \hat{X}) + L_{SSIM}(X, \hat{X}) \tag{4.4}$$

Where:

- **X:** Is the original image
- $\hat{\boldsymbol{X}}$: Is the reconstructed image
- **h, w:** Are the height and width of the image in pixel
- $\boldsymbol{\mu_x}$: Is mean value of image x
- $\boldsymbol{\sigma_x^2}$: Is variance of image x
- $\boldsymbol{\sigma_{xy}}$: Is the covariance of x and y
- $\boldsymbol{c_1 = (k_1 L)^2 \& c_2 = (k_2 L)^2}$ **:** Are two variables used to stabilize the division with weak denominator
- **L:** Is the dynamic range of the pixel-values (usually $2^{\#bitsperpixel} - 1$)
- $\boldsymbol{k_1 \& k_2}$: are two costants set to 0.01 and 0.03 respectively.

It is important to notice that the SSIM function represents a similarity measure defined in the interval $[-1, 1]$ where 1 means that the two images being

compared are identical. For this reason, to use it as a loss function it needs to be negated.

The use of an L_1 loss instead of a more classical MSE loss is motivated by the fact that it reduces blurriness and color artifacts in the reconstructed images [133].

The hyperparameters adopted during the training process are shown in Table 4.2. We tried to keep the Structure of the TF encoders as shallow as possible and actually noticed that increasing the number of blocks improved the performance of the model only marginally for the texture classes while for the product classes didn't help at all. A possible reason for this is that by increasing the depth, the model would become too complex for the relatively small dataset we used for training and evaluation and thus lead to overfitting.

Another important hyperparameter that needs to be selected when adopting a ViT architecture is the size of the patches in which the image needs to be divided. Following the original ViT paper with adopted 16x16 patches as the baseline for our model, which empirical tests confirmed to be a proper value for the task considered. In general, the idea to keep in mind during patch size selection is that Vision Transformers tend to work better with a high number of sequences as their input, for this reason, the size of the patches needs to be kept relatively small compared to the image size. Furthermore, in the context of the proposed approach, it was important not to use patches that are too small, as this would make much more frequent the presence of anomalies crossing multiple patches, which are easier to reconstruct since they can be inferred from the surroundings of the current patch analyzed and, therefore are more difficult to detect. The size of the short side of the stripes was selected to match the square patch sizes, while the long side is determined by the image size.

Table 4.2: Model hyperparameters

Classes	LR	Batch	Img size	Patch size	Stripe size	Emb. size	#Heads	#Blocks
Textures	1eˆ-4	64	128x128	16x16	128x16	128	4	2
Products	1eˆ-4	64	128x128	16x16	128x16	128	4	1

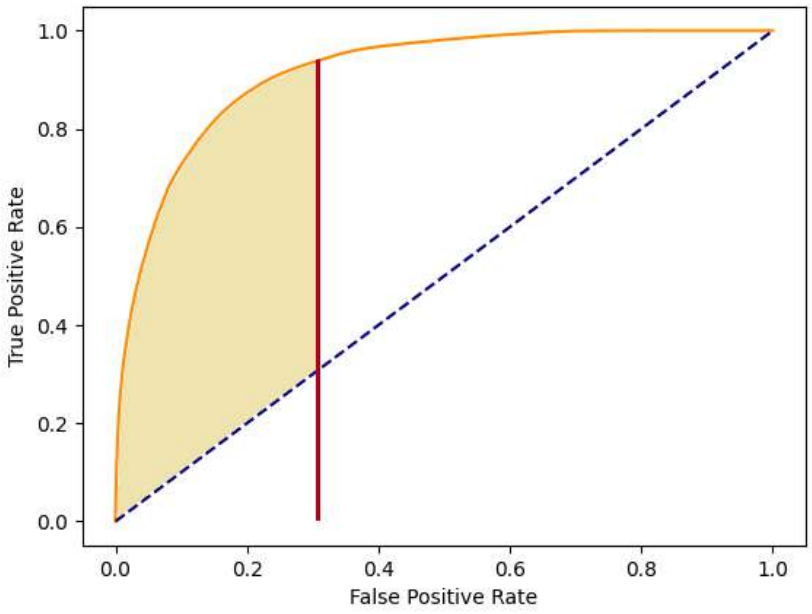


Figure 4.7: Illustration of the thresholded AUROC metric. The red line represents the selected threshold while the yellow region is the area taken into consideration for the evaluation of the model.

4.4.3 Metrics

For the evaluation of our model, since our objective is to assess its anomaly segmentation capabilities, we opted to use the Area Under the ROC Curve (AUROC), which plots the False Positive Rate versus the True Positive Rate and is a very commonly used metric for this type of problems. In order to obtain consistent results with those presented by the authors of the MVTec dataset, the values of the metric are computed up to a False Positive Rate of 0.3. The reason behind this choice is that thresholds that yield to a high FPR lead to meaningless segmentation results, especially for industrial scenarios where such results would lead to wrongly rejecting products not presenting any defects. An illustration of the thresholding process is provided in Fig. 4.7

4.4.4 Results

All the values presented in this section for our models have been obtained by first generating the reconstructed images, patch by patch, for each of the instances in the test set of the MVTec dataset, each of them has then been compared to the respective original image by applying a pixel-wise MSE loss (the deltas over the three channels of each pixel have been summed), in order to obtain a heatmap that highlighted the largest differences between the two, thus revealing the potentially anomalous regions. A Gaussian filter has also been applied to the map to smooth out anomalous regions and reduce noise.

In Table 4.3 the results of our ablation study, in which we compare the performance of our model when applying the masking process and different combinations of patch shapes, are reported. As we can see the approach which relies solely on the traditional square patches is outperformed by every other method using a combination of different shapes. In particular, we show that the best-performing approach is the one represented by a combination of the square patches with horizontal, or vertical stripes, which leads to an improvement in the pixelwise AUROC value of more than 4% compared to the baseline approach. As we can see the orientation of the stripes combined with the square patches didn't affect the overall performance of the model, which achieved, on mean, the same performance when using horizontal or vertical ones. On the other hand, we can see how using both orientations together leads to a decrease in the performance of the model. Our guess is that in this scenario the model becomes too complex compared to the relatively small dataset we used and therefore becomes too specialized on the training set, leading to overfitting. Finally, we can notice how completely removing the square patches and relying only on the horizontal and vertical stripes also affect the model performance negatively. A possible explanation for this behavior is that the square patches are able to provide some more locally specific information that the model needs to perform the reconstructions of the single patches effectively.

In Table 4.4 we provide a comparison between our methods and other

Table 4.3: results of the ablation study in which we show the effect of using different combinations of patch shapes on the model performance. All the values reported represent the normalized area under the ROC curve up to a mean FPR per pixel of 30%.

Class	Only Squares (No Mask)	Only Squares (Masked)	Squares + Rows	Squares + Cols	Rows + Cols	Squares + Rows + Cols
Carpet	0.495	0.510	0.712	0.723	**0.755**	0.661
Grid	0.814	0.835	**0.884**	0.883	0.800	0.817
Leather	0.734	0.792	**0.976**	0.905	0.723	0.723
Tile	0.727	0.754	0.771	**0.772**	0.659	0.690
Wood	0.701	0.757	0.836	**0.851**	0.808	0.840
Bottle	0.796	0.812	**0.850**	0.847	0.821	0.842
Cable	0.683	0.715	0.701	0.701	**0.753**	0.700
Capsule	0.863	**0.895**	0.891	0.885	0.889	0.893
Hazelnut	0.927	0.951	**0.953**	0.945	0.951	0.942
Metal nut	0.712	0.721	0.773	0.779	**0.865**	0.806
Pill	0.815	0.835	0.852	**0.852**	0.806	0.850
Screw	0.770	0.818	0.901	**0.905**	0.903	0.903
Toothbrush	0.926	0.949	**0.975**	0.966	0.967	0.971
Transistor	0.848	0.855	0.841	0.860	**0.866**	0.865
Zipper	0.783	**0.805**	0.750	0.770	0.634	0.720
Mean	0.773	0.800	**0.844**	**0.843**	0.814	0.815

approaches. In particular, we focus on the methods proposed in the MVTec paper [8] as our benchmarks and on VT-ADL as the only other approach using a ViT architecture for anomaly localization. For the latter, the results shown have been calculated by us as the original paper didn't provide the values for the AUROC metric, while for every other model, the result has been gathered from the original paper. As we can see our approach vastly outperforms the previous method based on ViTs represented by VT-ADL, in particular, by referring back to Table 4.3, we can see that even the approach relying solely on square patches still achieves better results than VT-ADL by improving over its results by almost an 12% margin, therefore proving the effectiveness of the masking process introduced in the Self-attention module

of our model. As for the remaining models, we show that while our approach

Table 4.4: Normalized area under the ROC curve up to a mean false positive rate per pixel of 30% for each dataset category. The values in bold represent the best scores overall, while the underlined ones represent the best scores between the models not using extra data.

Class	f-anoGan	Feature dictionary	Student teacher	l2-AE	SSIM-AE	Texture inspection	Variation model	VT-ADL	Ours
Carpet	0.251	**0.943**	0.927	0.287	0.365	0.874	0.162	0.549	0.712
Grid	0.550	0.872	**0.974**	0.741	0.820	0.878	0.488	0.569	0.884
Leather	0.574	0.819	**0.976**	0.491	0.356	0.975	0.381	0.817	**0.976**
Tile	0.180	0.854	**0.946**	0.174	0.156	0.314	0.304	0.589	0.771
Wood	0.392	0.720	**0.895**	0.417	0.404	0.723	0.408	0.682	0.836
Bottle	0.422	**0.953**	0.943	0.528	0.624	0.454	0.667	0.687	0.850
Cable	0.453	0.797	**0.866**	0.510	0.302	0.512	0.423	0.751	0.701
Capsule	0.362	0.793	**0.952**	0.732	0.799	0.698	0.843	0.615	0.891
Hazelnut	0.825	0.911	**0.959**	0.879	0.847	0.955	0.802	0.926	**0.959**
Metal nut	0.435	0.862	**0.979**	0.572	0.539	0.135	0.462	0.711	0.773
Pill	0.504	0.911	**0.955**	0.690	0.698	0.440	0.666	0.748	0.852
Screw	0.814	0.738	**0.961**	0.867	0.885	0.877	0.697	0.771	0.901
Toothbrush	0.749	0.916	0.971	0.837	0.846	0.712	0.775	0.878	**0.975**
Transistor	0.372	0.527	0.566	0.657	0.562	0.363	0.601	0.689	**0.860**
Zipper	0.201	0.921	**0.964**	0.474	0.564	0.928	0.209	0.683	0.750
Mean	0.472	0.836	**0.922**	0.590	0.584	0.656	0.526	0.683	0.844

performs worst than the best one from the MVTec paper, represented by the student-teacher architecture (7.8% AUROC score difference), it outperforms every other method by a margin going from 1% to 37%. One important aspect to notice is that the two top-performing methods presented in the MVTec paper, namely the student-teacher and the Feature Dictionary models which are the only ones exceeding a mean AUROC score of 0.7, both rely on feature extractors pre-trained on much larger datasets such as the popular imagenet one [37] while our model is trained from scratch on the MVTec dataset making it the best-performing model not relying on extra data for training and thus showing the possibility of adopting transformer-based models, typically considered very heavy, even to scenarios where we have a relatively small amount of data available.

Finally, in Fig. 4.8, we provide some qualitative results of our model by showing a comparison between the original images and ground truth anomaly maps with the reconstructed images and anomaly maps generated by our

model. As we can see the model is able to effectively mask out the smaller anomalies from the reconstructed images, leaving only small artifacts in their places. As for larger anomalies that are spread through different regions of the original images, the model is usually not able to completely remove them as it can infer their structure from the surrounding patches. Nonetheless, in many cases, the defective part has a more "washed-out" appearance in the reconstructed image which allows for its localization.

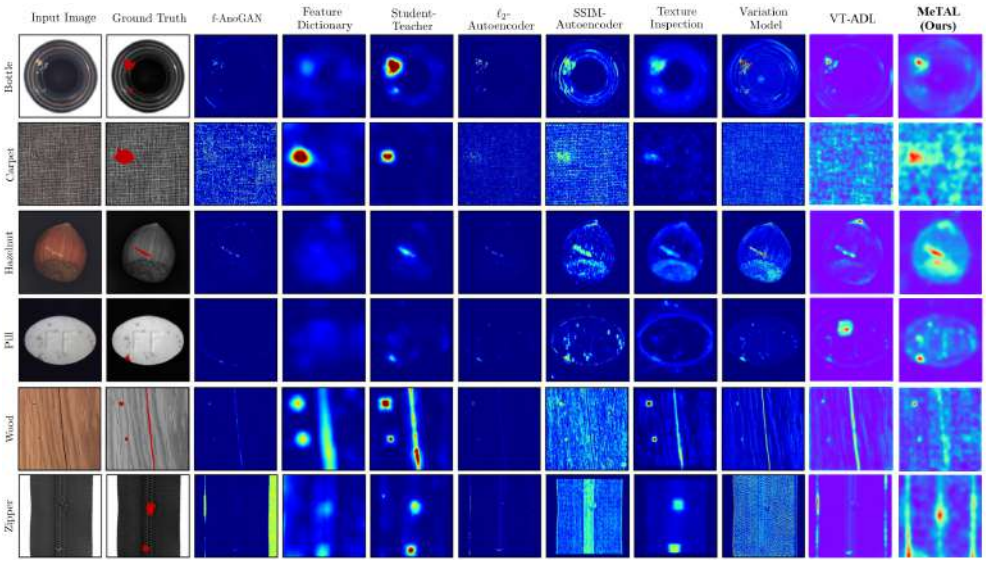


Figure 4.8: In order from left to right, we have: (1) The original image used as input, (2) the ground truth mask showing the location of the anomaly in the original image, and (3-11) the anomaly maps generated by our model and the competition

Furthermore, we present the results for the Head CT-hemorrhage dataset. In particular, since this dataset doesn't provide masks for the instances anomalies, we use as a comparison metric the images ROCAUC scores. We report our results, together with the ones presented by Salehi et al. [99] which, as far as we know, are the best ones available online for the headct dataset, in Tab. 4.5. As we can see our model, even in its most basic configuration, achieves results comparable to the State of the Art, while when horizontal stripes are added it is able to surpass the other presented approaches by a significant margin showing the effectiveness of the presented approach even

for scenarios when the data available is very limited. The downside, however, is that the increased complexity of the model appears to make it less stable as we can observe from the higher variance characterizing the results achieved by this configuration.

Table 4.5: AUROC for anomaly detection on Head CT dataset.

	AUROC Head CT (%)
OCGAN [**90**]	51.20 ± 0.358
LSA [**1**]	81.67 ± 3.626
GT [**47**]	49.5 ± 3.873
MKD [**99**]	80.4 ± 0.006
MeTAL (Squares Only)	81.35 ± 1.153
MeTAL (Squares+Rows)	**86.32** ± 4.03

Table 4.6: Comparison of the number of parameters defining the compared models architectures

	# of parameters
l^2 **- AE / SSIM-AE**	1.2M
f-AnoGAN	24.6M
Feature dictionary	11.5M
Student teacher	26M
VT-ADL	25M
MKD	15M
Ours (Products)	17.8M
Ours (textures)	26.3M

For completeness, we are also providing a comparison of the model sizes for the different approaches, expressed in the number of parameters defining their structure (Tab. 4.6). As we can see the proposed approach size is

generally comparable to the ones of the other frameworks, excluding the most simple ones being represented by the two autoencoder variations which, however, achieve very poor performances on the selected dataset.

4.5 Conclusions and Future Work

In the present chapter, we investigated the use of a Framework based on the Self-Attention mechanism introduced by the Vision Transformer model. In particular, we proposed an image Inpainting approach to anomaly detection which leverages the ability of the ViT encoder to find correlations between different regions of a given image in order to reconstruct each of them based only on the information contained in the surrounding ones. Furthermore, we have shown that the model's performance is affected positively by the use of heterogeneous shapes for the subsets into which the original image is split. Our opinion is that this approach allows the model to learn correlations between patches at different scales of the image, therefore increasing the quality of the generated reconstructions. Finally, we have shown how the ViT model, while usually considered a heavy architecture, can be used effectively even on a relatively small dataset without the need to rely on extra data from external sources, achieving the best performance compared to other models trained in a similar setting on the MVTec dataset.

Nonetheless, as future work, we believe it would be interesting to explore our model's capabilities when pre-trained on a larger dataset before fine-tuning it on the final task's data and we would also like to investigate the possibility of a more general approach to the multi-scale patch acquisition we introduced in this work.

As another line of research we believe it would be also worth investigating the adoption of ideas introduced in other recent works present in the literature, such as [2], [88], [5], [93], in order to try to further improve the performance of the proposed architecture.

Furthermore, we believe that the main limitation of the proposed ap-

proach regards the ability to handle large-scale anomalies since these can be inferred from the context of the image even when no information is available for the specific patch we are reconstructing. For this reason, in future works, we would like to investigate more sophisticated masking approaches that allow for a better generalization of the model to different anomaly scales. Finally, in the future, we will also extend our idea to process 3D applications [10].

5

Few-shot layout segmentation via dynamic instance generation and local thresholding

Over the years, the humanities community has increasingly requested the creation of artificial intelligence frameworks to help the study of cultural heritage. Document Layout segmentation, which aims at identifying the different structural components of a document page, is a particularly interesting task connected to this trend, specifically when it comes to handwritten texts. While there are many effective approaches to this problem, they all rely on large amounts of data for the training of the underlying models, which is rarely possible in a real-world scenario, as the process of producing the ground truth segmentation task with the required precision to the pixel level is a very time-consuming task and often requires a certain degree of domain knowledge regarding the documents at hand. For this reason, in the present chapter, we propose an effective few-shot learning framework for document layout segmentation relying on two novel components, namely a dynamic instance generation and a segmentation refinement module.

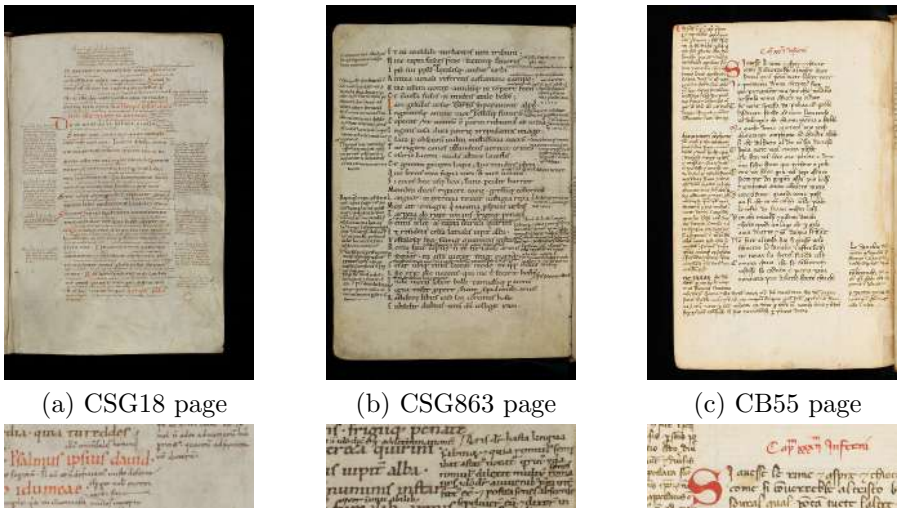
5.1 Introduction

Page segmentation of historical manuscripts allows humanists to study documents more quickly and easily because it allows the paratexts (i.e all the semantic elements which are part of the foreground but don't belong to the main text) to be analyzed separately.However, performing this task in historical manuscripts is much more difficult than in printed documents [94] due to many potential variables, such as layout structure, decoration, different writing styles, texture, and degradation.

While, in recent years, machine learning and deep learning-based approaches have been more and more commonly adopted for this kind of task as they represent a more effective alternative compared to traditional approaches, they require a large amount of carefully defined ground truth maps in order to be properly trained and compared with other approaches.

Furthermore, for ground truths to be suitable for training accurate deep learning models, the annotation of the segmentation masks must be as precise as possible down to the pixel level [46]. The disadvantage is that the pixel-precise annotation of the entire historical document page dataset is a very time-consuming process and requires domain-specific knowledge, which only an expert humanist can satisfy, especially when working with historical manuscripts [79], making this type of information rarely available in a real-world scenario. Nonetheless, few-shot learning approaches in the context of document layout segmentation are still under-explored in the literature.

For this reason, in the present work, we propose a novel few-shot learning framework for efficient pixel-precise page segmentation of historical manuscripts, which is able to accurately segment the different components of a document page (e.g. text, paratext, images) achieving results comparable to the current state-of-the-art approaches on the popular Diva-HisDB dataset (Fig. 5.1) while using only a fraction of the available data for the training process. The rest of the chapter is organized as follows: Sections 5.2 and 5.3 contain, respectively, a detailed description of the proposed framework and the experimental setup used to train and test it. Section 5.4 provides an in-depth



(a) CSG18 page (b) CSG863 page (c) CB55 page

(d) CSG18 detail (e) CSG863 detail (f) CB55 detail

Figure 5.1: Samples from the three representative manuscripts (CSG18, CSG863 and CB55) present in DIVA-HisDB. Fig. 5.1a– 5.1c show a full page for each manuscripts, while Fig. 5.1d– 5.1f show a detail extracted from each of them.

description of the results achieved by our model, both from a quantitative and a qualitative perspective. Finally, in Section 5.5, the conclusions are drawn.

5.2 Proposed approach

The proposed approach is built on three core components, namely a robust segmentation backbone used to retrieve the semantic components of each document page, a dynamic instance generation module that allows us to fully leverage the limited amount of data available at training time and finally a segmentation refinement module that makes it possible to further improve the quality of the segmentation maps produced by our model. A visual representation of the proposed framework pipeline is reported in Fig. 5.2.

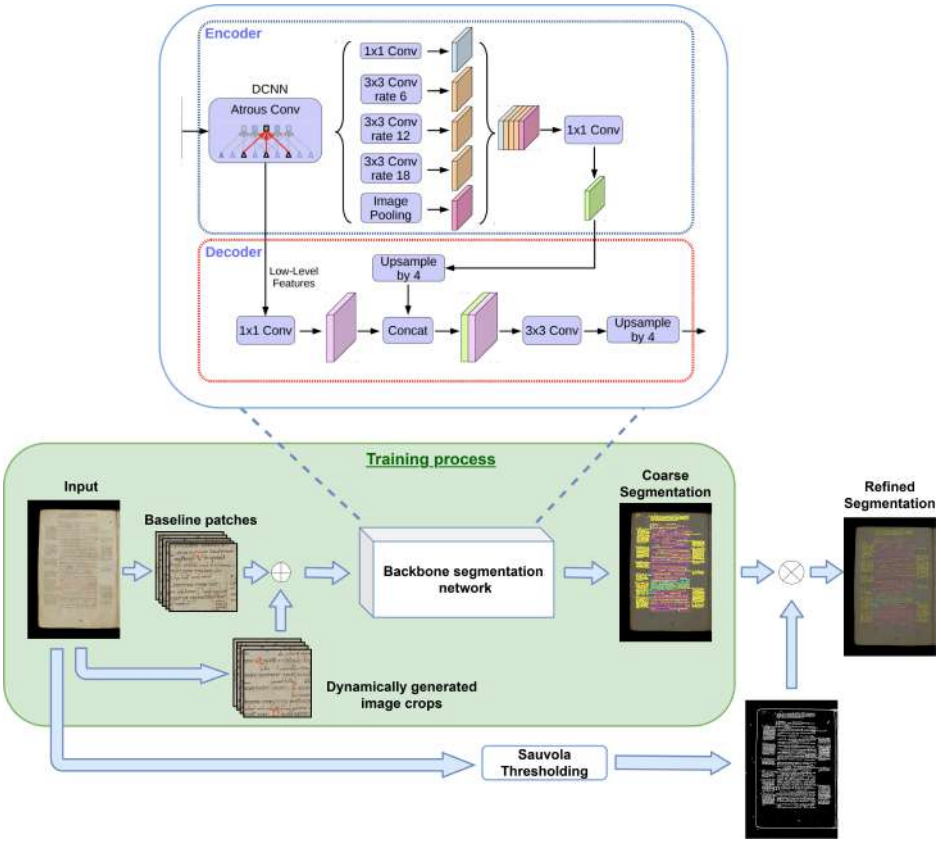


Figure 5.2: Visual representation of the segmentation pipeline for the proposed framework. The green area represents the processes carried out during the training phase, where each input image is split into 2 sets of patches: the baseline patches, which are non-overlapping patches of size $k \times k$ providing a complete representation of the original image and a set of C random crops which are extracted from random locations of the image at each training epoch. These 2 sets of patches are then combined and given in input to the backbone segmentation model which provides a predicted coarse segmentation map for each of them. These maps are compared with the ground truth ones through the application of a weighted cross-entropy loss. At inference time the dynamic instance generation step is removed while a segmentation refinement process is applied to the outputs of the backbone architecture to obtain more precise segmentation maps

5.2.1 Segmentation backbone

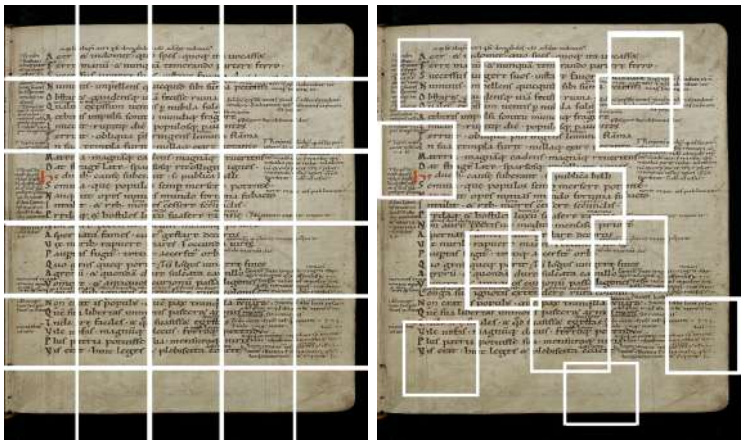
Adoption of a robust backbone is a crucial step in each Deep Learning framework. When working in a few-shot setting in particular we need a network

that is able to capture a sufficient level of detail while being given in input just a handful of samples. For this reason, we selected DeepLabV3+ [27] as the backbone of our framework. DeepLabV3+ is a popular pixel-wise semantic segmentation model built on its predecessor DeepLabV3 [25]. The latter is a ResNet [50] based architecture heavily relying on atrous convolutions which are employed both in parallel and in a cascade in order to enlarge the receptive fields of the filters and consequently retain a higher spatial resolution throughout the network. The key advantage of this approach is that it allows for deeper neural networks that provide larger feature maps at no additional computational cost. Finally, the Atrous Spatial Pyramid Pooling (ASPP) is introduced in DeepLabV3 as a way of capturing features at different scales in the original image by relying on a heterogeneous set of dilation rates in the network. DeepLabV3+ introduces two substantial changes compared to the aforementioned architecture. The first one regards the substitution of the ResNet encoder with a custom version of the Aligned XCeption [29] model in which all max pooling operations are replaced by depth-wise separable convolution. Furthermore, it adds a simple yet effective decoder which refines the segmentation results. The decoder module employs depth-wise separable convolutions to enhance the spatial resolution of the feature maps, resulting in sharper and more detailed output segmentation maps.

5.2.2 Dynamic instance generation

The dynamic instance generation module is a key component of the training pipeline of our framework. The key idea behind it is that it efficiently exploits the small amount of data available at training time. To do so, instead of relying on the full document pages as the instances of our dataset, we split them into two sets of smaller patches. The first ones, which we will refer to as baseline patches, consist of a set of non-overlapping sub-regions of size $m \times n$ extracted from the original input image in order to cover its entire surface and are kept consistent between the training and inference time (Fig. 5.3a). In addition to the baseline patches, as a way to further improve

the generalization capabilities of our model, we also generate a small set of k potentially overlapping crops of the same size as the baseline patches which are extracted from random locations of the original image (Fig. 5.3b). This process is carried out at each epoch during training time, while at inference time no additional crops are generated as they are not needed to obtain the final segmentation mask. While relying on sub-patches of the original images is a common approach in computer vision-related tasks, in most cases, these patches are either limited to the ones corresponding to our baseline ones, which leads to losing potentially useful information contained in the data. As an alternative approach, they may generate a large number of patches in advance, without considering the varying complexity of different datasets [127] . As a consequence, excessive amounts of potentially unnecessary data is produced. Our dynamic instance generation approach addresses both limitations effectively at the cost of a very small computational overhead at training time.



(a) Baseline patches (b) Randomly selected crops

Figure 5.3: Representation of the instance generation process of the 2 sets of patches used to train our model: in 5.3a is shown the generation process for baseline non-overlapping patches, while 5.3b provides a visual depiction of our dynamic crop generation process

5.2.3 Segmentation refinement

Our segmentation refinement module is based on the Sauvola thresholding algorithm for document binarization [100]. The Sauvola thresholding algorithm is an evolution of Niblack's method [78], which introduced the idea of a dynamic threshold that is calculated based on the mean and standard deviation of the gray levels of a local window inside an image. The main drawback of Niblack's approach is that it didn't perform well for images with a light-textured background as it would result in very noisy binarization masks. Sauvola solved this problem by introducing the dynamic range of the standard deviation as an additional term in the equation used to calculate the local threshold, which has the effect of amplifying the contribution of the standard deviation in an adaptive manner throughout the image. The resulting equation adopted by the Sauvola algorithm is shown in Eq. 5.1, where N is the local window of size $n \times n$, $\mu(N)$ and $\sigma(N)$ are, respectively, the corresponding mean and standard deviation and R is the dynamic range of the standard deviation. Finally, k is a manually selected parameter that regulates the value of the local threshold.

$$T = \mu(N) \times \left(1 + k \times (\frac{\sigma(N)}{R} - 1)\right) \tag{5.1}$$

The refined segmentation masks are then obtained by performing the Hadamard product between the layout segmentation predictions provided by our backbone and the mask resulting from running the Sauvola algorithm on the corresponding images of the dataset.

5.3 Experimental setup

In this section, we outline a detailed description of the dataset adopted for the experiments and the training setup. Furthermore, the metrics used to evaluate and compare the performance of the proposed approach are presented, together with the results of the ablation study.

Table 5.1: Classes distribution (%) for each manuscripts of Diva-HisDB [106] (CB55, CSG18 and CSG863), and for Bukhari et al. [16] dataset

Manuscript	BG	Comment	Decoration	Text
CB55	82.41	8.36	0.55	8.68
CSG18	85.16	6.78	1.47	6.59
CSG863	77.82	6.35	1.83	14.00
Bukhari et al.	86.07	4.71	—	9.22

5.3.1 Dataset

To train and test our model we selected the popular DivaHisDB dataset [106]. Diva-HisDB is a collection of 3 medieval manuscripts (CB55, CSG18 and CSG863) selected for their heterogeneity and layout complexity. All the documents contained in the dataset are characterized by 4 classes of semantic components, namely main text, comments, decorations and background (BG), with very unbalanced distributions making the dataset particularly challenging for a few shot settings as the less common classes are present in a very small amount or not at all in some of the instances. A detail of the semantic component distributions for each manuscript is provided in Tab. 5.1.

Furthermore, the manuscripts provide a high degree of heterogeneity concerning the level of degradation of the pages, the epoch in which they were written, and both inter and intra-class differences in the layout of the pages and in writing styles, as both the CSG18 and CSG863 were written by an unspecified number of authors. The dataset consists of a total of 150 instances, 50 for each document class, of these 60 are typically used for training, 30 for validation and another 60 for testing the models. In the present work, we relied on just 6 images, 2 for each class, for training our model (Fig. 5.4). For each of the document pages, the dataset provides a corresponding ground truth segmentation mask as shown in Fig. 5.5.

Finally, to further validate the robustness of our approach we also tested it

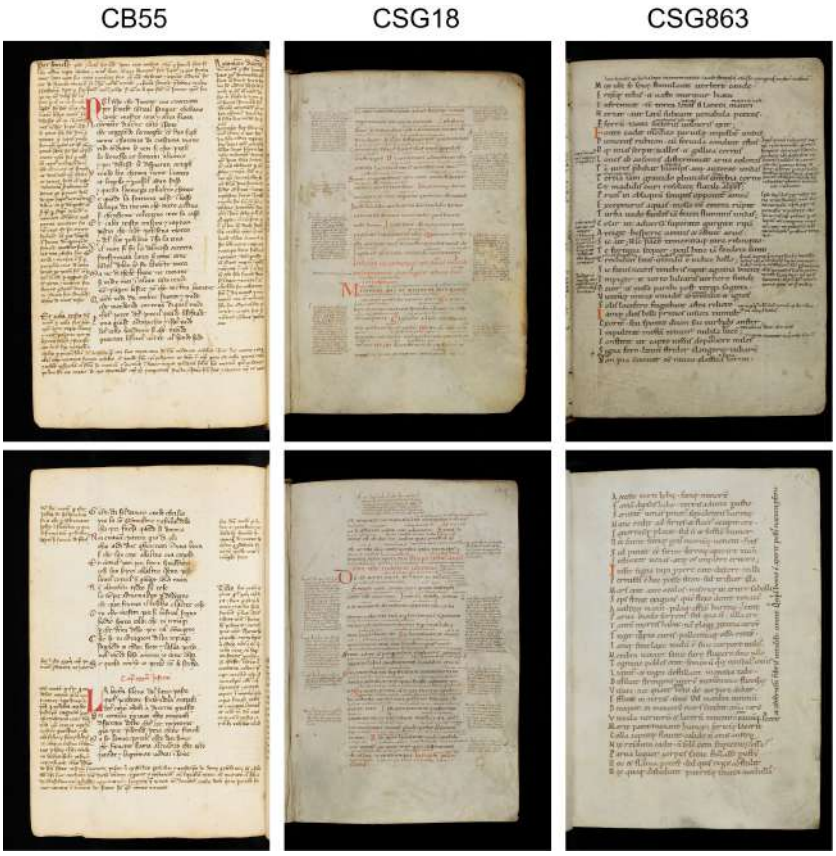
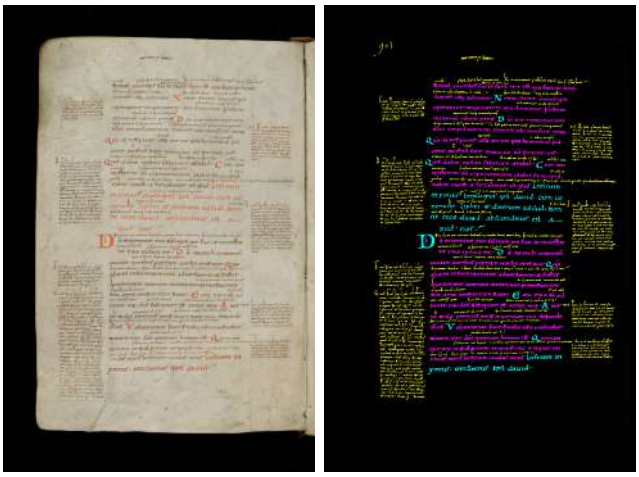


Figure 5.4: Instances selected from each manuscript in DIVA-HisDB as the training set for the proposed approach. Each of them was chosen to effectively represent the characteristics for the corresponding class

on the dataset proposed by Bukhari et al.[16] which consists of 32 images each representing a page from one of three different Arabic historical manuscripts. Out of all the samples, 24 are typically used for the training process while the remaining 8 are used for the testing, while for the purpose of this work we only relied on 3 images, one for each manuscript, to train our model. A detail of the semantic component distributions is provided in Tab. 5.1.

5.3.2 Training and inference setup

Our model was trained using the popular Adam optimizer with a learning rate of 10^{-3} and a weight decay of 10^{-5}. The maximum number of epochs for



(a) Original page (b) Page ground truth

Figure 5.5: Images showing a page of the CSG18 manuscript (5.5a) as well as its corresponding ground truth mask (5.5b), in which the magenta areas represent the main text, while the yellow and cyan areas represent the comments and decorations respectively. Finally, the black area represents the background of the image

which it was allowed to run has been set to 200 with an early stop in case the validation loss didn't improve in the last 20 epochs and a buffer of 50 epochs which guarantees that the model will be trained at least for the specified amount of iterations. During each epoch, a set of 10 dynamic crops of size 672×672 px has been generated in addition to the baseline patches of the same sizes extracted from the original image. This process led to a maximum of 4012 instances being generated for each document class during training, in case the model needed all the 200 epochs in order to converge. In order to be able to fit them in the GPU memory the images of the dataset have been resized from their original high resolution (up to $4.8k \times 6.8k$ px), down to a size of 1344×2016 px. The loss function selected to train the model is a weighted Cross Entropy Loss [58] in which the weight for each semantic element class is inversely proportional to the frequency of that element in that dataset and, more precisely is calculated as the square root of 1 over the square root of the occurrence frequency of the corresponding element in

the dataset (Eq. 5.2).

$$W_i = \sqrt{\frac{1}{F_i}} \tag{5.2}$$

This specific choice was made to take into account the high imbalance between the semantic class distribution in each document category of the datasets (Tab. 5.1). Our model was trained separately and from scratch on each document class. Regarding the inference setup, the main choice involved in it is represented by the hyperparameters of the segmentation refinement algorithm, namely the window size which was kept consistent at 15×15 px for all document classes and the control value k, which regulates the value of the threshold in the local window (the higher the k value, the lower the threshold) and was set at the value of 0.01 for all classes.

5.3.3 Evaluation metrics

In order to evaluate the performance of our proposed approach we use different metrics such as Precision, Recall, Intersection over Union (IoU) and F1-Score. These evaluation metrics are calculated individually for each one of the manuscripts that compose DIVA-HisDB dataset. Metric definitions are reported in Eq. 5.3– 5.6, where TP, FP and FN stand respectively for True Positives, False positives and False Negatives. For each metric a weighted average is performed, based on each class frequency in each manuscript. The final evaluation of a model is then obtained by averaging the metrics of all pages of the three manuscripts.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5.3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5.4}$$

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{5.5}$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.6}$$

5.4 Results

In the following section, we provide a thorough comparison between the results achieved by the proposed framework and a set of popular semantic segmentation approaches, namely DeepLabV3 [25], its improvement represented by DeepLabV3+ [27], FCN [66], Lite Reduced Atrous Spatial Pyramid Pooling (LRASPP) [51] and Pyramid Scene Parsing Network (PSPNet) [134], furthermore we also include the results obtained by current state of the art for the task of document layout segmentation, which we will refer to as MLA [127]. The comparison focuses both on a quantitative and a qualitative perspective in order to provide a complete overview of the quality of the model's predicted segmentations. To this end, we also provide a discussion about the critical cases in which our approach fails to provide the correct segmentation for the corresponding instances. All the models, excluding MLA for which we gathered the results from the respective paper, have been personally tested by us keeping the training and evaluation settings as consistent as possible.

Table 5.2: Comparison between the performance of our model and the competition on the 4 selected metrics. The best and second-best performing models are reported in a bold and underlined fashion respectively while FS indicates the models trained in a few-shot setting by using the same set of images selected for our framework

	CB55				CSG18				CSG863				Mean			
Backbone	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1
FCN (FS)	0.894	0.883	0.783	0.863	0.874	0.885	0.797	0.863	0.915	0.907	0.826	0.895	0.894	0.892	0.802	0.874
FCN	0.902	0.900	0.815	0.884	0.930	0.930	0.869	0.919	0.923	0.919	0.847	0.909	0.918	0.916	0.844	0.904
LRSAPP (FS)	0.847	0.837	0.718	0.808	0.919	0.913	0.871	0.911	0.869	0.864	0.757	0.842	0.878	0.871	0.782	0.854
LRSAPP	0.880	0.883	0.789	0.864	0.921	0.927	0.868	0.918	0.911	0.910	0.833	0.899	0.904	0.907	0.830	0.894
PSPNET (FS)	0.876	0.868	0.761	0.846	0.906	0.905	0.829	0.890	0.913	0.896	0.817	0.888	0.898	0.890	0.802	0.875
PSPNET	0.887	0.894	0.811	0.880	0.912	0.920	0.857	0.910	0.913	0.915	0.845	0.906	0.904	0.910	0.838	0.899
DeepLabV3 (FS)	0.893	0.883	0.784	0.863	0.901	0.895	0.806	0.873	0.864	0.853	0.737	0.828	0.886	0.877	0.776	0.855
DeepLabV3	0.905	0.901	0.817	0.886	0.930	0.931	0.871	0.920	0.920	0.914	0.839	0.903	0.918	0.915	0.842	0.903
DeepLabV3+ (FS)	0.908	0.903	0.821	0.888	0.931	0.929	0.867	0.918	0.936	0.933	0.875	0.927	0.925	0.922	0.854	0.911
DeepLabV3+	0.943	0.945	0.896	0.939	0.961	0.962	0.929	0.959	0.965	0.965	0.935	0.964	0.956	0.957	0.920	0.954
MLA	-	-	-	-	-	-	-	-	-	-	-	-	0.965	**0.995**	**0.989**	**0.995**
Ours	**0.989**	**0.987**	**0.977**	**0.988**	**0.983**	**0.982**	**0.967**	**0.982**	**0.986**	**0.983**	**0.971**	**0.984**	**0.986**	0.984	0.972	0.985

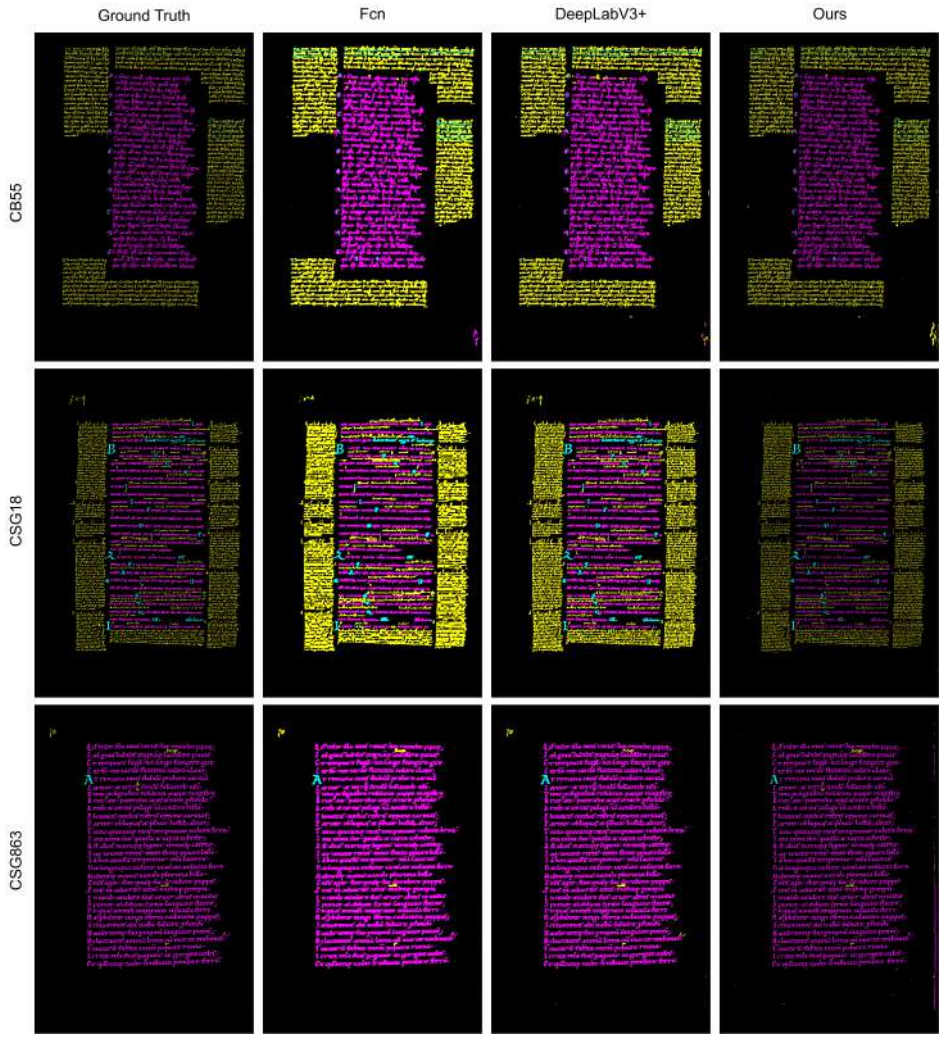


Figure 5.6: Image showing a qualitative comparison between our framework and the competition ones. Each row represents a zoomed area belonging to a different instance of the dataset, representing the three classes of manuscript contained in it. In the first column, the ground truth segmentation maps for the 3 images are shown, while on the remaining columns we provide the results produced by the three systems, FCN, DeepLabV3+ and Ours respectively

5.4.1 Quantitative results

In Tab. 5.2 the quantitative results achieved by our proposed framework for all the selected metrics across all the document classes contained in the

Diva-HisDB dataset, are shown and compared with the competitor models. In particular, for all the models, excluding MLA, we provide both the results obtained by training them on the entire available dataset and the ones obtained by training the model only on the subset of 2 pages selected for our approach (FS = Few-Shot setting). Unfortunately, MLA authors provided only the mean scores for the selected metrics and some implementation details were missing, leading our attempt at reimplementing their work to achieve sub-optimal results. As we can see our model is consistently capable of outperforming the other semantic segmentation networks on all the metrics, regardless of the setup in which they have been trained. In particular, compared to the second-best performing approach, being represented by DeepLabV3+, our model achieves a mean improvement of 7.7% when the former is trained in a few-shot setting with a peak improvement of 11.8% for the IoU metric. While, when DeepLabv3+ is trained using the full training set, our approach outperforms it by a still substantial mean of 3.5% (5.2% for the IoU metric) while using only a fraction of the available data.

Furthermore, our framework achieves very close performance even when compared with the current state-of-the-art MLA, even surpassing it by 2.1% on the mean precision metric. As for the remaining metrics our model performance is still comparable to that of MLA with a difference going from 1.7% for the IoU metric, to as little as 1% for the F1-score. It is important to notice, however, that MLA is trained on around 180000 instances extracted from all the images of the training set, while our framework, as previously mentioned, extracts at most 4012 unique instances from just 2 of the available images in the training set, resulting in a reduction of the needed data by a factor approximately 45.

Finally in Tab. 5.3 we show the comparison between our model and the competition on the Bukhari dataset for Arabic manuscript layout segmentation. As we can see our framework achieves the best performance compared to all the other approaches even when they are trained using the full training set. In particular, compared to the single best performing model, being rep-

resented by DeepLabV3+ our achieves a 2-4% improvement across all metrics against its fully trained configuration and around a 4-9% performance improvement against the few shot version of the model.

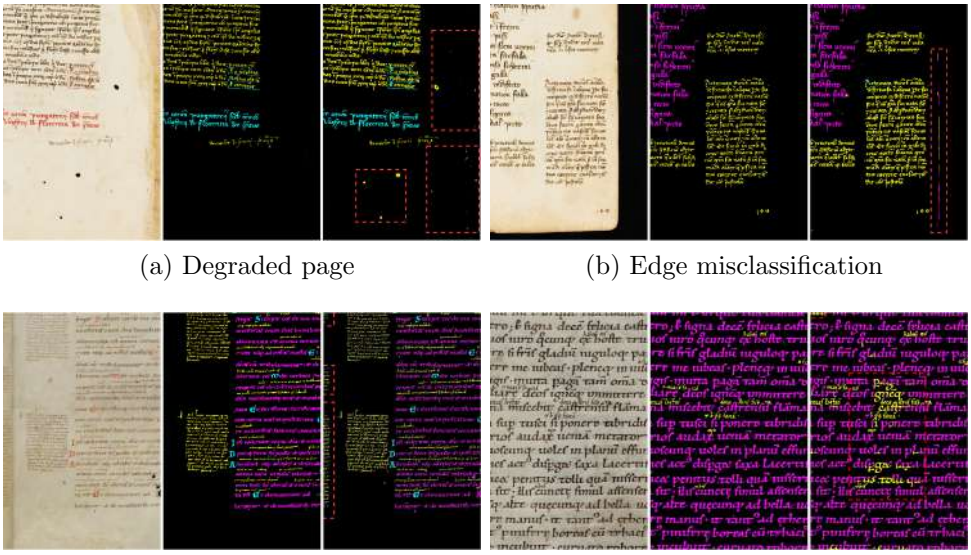
Table 5.3: Comparison between the performance of our model and the competition on the Bukhari dataset. The best and second-best performing models are reported in a bold and underlined fashion respectively while FS indicates the models trained in a few-shot setting by using the same set of images selected for our framework

Backbone	Prec	Rec	IoU	F1
FCN (FS)	0.836	0.875	0.788	0.853
FCN	0.865	0.899	0.824	0.879
LRSAPP (FS)	0.806	0.858	0.742	0.805
LRSAPP	0.899	0.876	0.806	0.884
PSPNET (FS)	0.843	0.859	0.770	0.846
PSPNET	0.911	0.861	0.790	0.875
DeepLabV3 (FS)	0.879	0.815	0.735	0.836
DeepLabV3	0.908	0.871	0.802	0.883
DeepLabV3+ (FS)	0.929	0.907	0.850	0.914
DeepLabV3+	<u>0.956</u>	<u>0.943</u>	<u>0.902</u>	<u>0.946</u>
Ours	**0.970**	**0.966**	**0.940**	**0.967**

5.4.2 Qualitative results

Fig. 5.6 shows the segmentation maps produced by our model for three document pages belonging, respectively, to the three document class present in the Diva-HisDB dataset and compared with the ones predicted by the FCN and DeepLabV3+ models, both trained on the whole available training set. Furthermore, the corresponding ground truth segmentation is provided as a reference.

While the maps produced by FCN are typically correct and with very limited amounts of noise, they tend to be very coarse, especially when observed in the areas of the pages where the text is smaller and the different components more intertwined. DeeplabV3+ provides a higher level of detail, in particular when looking at the main text component (magenta segmentation). Finally, our model provides visibly more precise segmentation maps than the competition when compared to the ground truth ones.



(a) Degraded page (b) Edge misclassification

(c) Adjacent page text (d) Foreground misclassification

Figure 5.7: Overview of the main instances of misclassification for the proposed approach. From the top left corner we have: 5.7a degraded spots in the page background being misclassified as foreground, 5.7b the same type of misclassification involving page edges, 5.7c Text belonging to the adjacent page being recognized as part of the current one, 5.7d a simple case of misclassified foreground elements in particular involving the main text being mistaken as part of the comments

Fail cases

As already mentioned in the previous section the main drawback of the presented approach is that compared to the competition it introduces more noise in the provided segmentations. For completeness, in Fig. 5.7 we provide some

more criticalities of the proposed framework together with the original image and the corresponding ground truth. In particular, other than the typical misclassification of foreground elements (Fig. 5.7d) we can notice three main instances of recurrent mistakes. The first one is represented by the edge of the pages of the documents which, being lighter than the black background introduces an area of high contrast that is identified both by the model and by the thresholding algorithm as part of the text (Fig. 5.7b). A similar occurrence can be observed for degraded areas in the page's background, these areas are, in fact, typically darker than the rest of the background and are once again misclassified as foreground elements (Fig. 5.7a). Finally, we have the misclassification caused by the text belonging to the page adjacent to the currently analyzed instance, which while correctly identified as part of the text by our model, is not included in the ground truth segmentations (Fig. 5.7c). This last case, however, is highly dependent on the coarse cropping process of the instances of the Diva-HisDB dataset which doesn't precisely include only the elements of the current page and, as such, is easily solvable by refining the crops.

5.4.3 Ablation study

In this section, we provide the details regarding the ablation study we conducted in order to obtain the final version of the proposed framework. In particular, we show the effects that different segmentation backbones and patch sizes for the generated instances have on the performance of our approach for the task at hand. Furthermore, we provide a comparison between the performance of the baseline model and the models enhanced with the additional modules introduced in this work in order to provide proof of their effectiveness.

Backbones

Tab. 5.4 shows a comparison of the performance of our framework when using different backbones for the segmentation module. For this compar-

Table 5.4: Comparison between the use of different neural network architectures as the segmentation backbone for our model, in bold is reported the best-performing model

	CB55				CSG18				CSG863				Mean			
Backbone	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1
FCN	0,871	0,850	0,728	0,820	0,887	0,876	0,773	0,847	0,884	0,874	0,770	0,851	0,881	0,867	0,757	0,839
LRASPP	0,801	0,770	0,614	0,718	0,815	0,835	0,718	0,797	0,919	0,908	0,858	0,912	0,845	0,838	0,730	0,809
PSPNET	0,849	0,828	0,694	0,792	0,877	0,869	0,761	0,838	0,901	0,887	0,801	0,876	0,876	0,861	0,752	0,835
DeeplabV3	0,873	0,853	0,734	0,824	0,891	0,881	0,781	0,854	0,882	0,869	0,762	0,845	0,882	0,868	0,759	0,841
DeeplabV3+	**0.918**	**0.908**	**0.827**	**0.894**	**0.926**	**0.923**	**0.855**	**0.910**	**0.931**	**0.927**	**0.863**	**0.917**	**0.925**	**0.919**	**0.848**	**0.907**

Table 5.5: Comparison between the adoption of different patch sizes during the instance generation process of our framework, in bold is reported the best-performing model

	CB55				CSG18				CSG863				Mean			
Patch size	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1
224	0.911	0.900	0.813	0.884	0.920	0.916	0.843	0.900	0.919	0.917	0.846	0.904	0.917	0.911	0.834	0.896
336	0.916	0.906	0.823	0.891	0.925	0.920	0.849	0.905	0.928	0.926	0.860	0.916	0.923	0.917	0.844	0.904
672	**0.918**	**0.908**	**0.827**	**0.894**	**0.926**	**0.923**	**0.855**	**0.910**	**0.931**	**0.927**	**0.863**	**0.917**	**0.925**	**0.919**	**0.848**	**0.907**

ison, we selected a set of recent and popular semantic segmentation networks (DeepLabV3 [25], DeepLabV3+ [27], FCN [66], LRASPP [51] and PSPNet [134]). To allow for a fair comparison all the models have been trained and tested with the exact same setup, with 2 images for each document class as the training set and a consistent patch size of 672×672 px. As we can see all the models provide reasonably good performance on the task at hand achieving an IoU higher than 70% and a performance of over 80% for all the remaining metrics. From this analysis emerges that DeepLabV3+ consistently outperforms all other models on each of the selected metrics and on all the document classes present in the dataset, achieving an mean improvement of 6.23% over the second-best model, being represented by its predecessor DeepLabV3. A particularly interesting boost in performance is achieved for the IoU metrics where an increase of almost 9% is obtained by the former over the latter.

Patch sizes

A further comparison has been performed by exploring the adoption of different sizes for the crops of the instances being provided to the backbone networks. In particular, we selected 3 different sizes, going from the standard 224×224 which is the size used by all the pre-trained models available in PyTorch, to a much larger 672×672. The results of this comparison are shown in Tab. 5.5. In this case, the difference in performance wasn't as substantial as the one resulting from the adoption of different types of segmentation backbones. In particular, we can notice that the difference between the best and the worst performing models, which are the ones adopting the largest and smallest patch sizes respectively, is on mean around 1%. A potential explanation behind the improved performance corresponding to the adoption of larger patch sizes is that the model to which they are given in input is able to capture a higher amount of contextual information regarding the layout of the original image from which they are extracted, allowing for more accurate segmentation.

Table 5.6: Results of the ablation study. Each row shows the performance of the different versions of our system across all the selected metrics for the 4 classes of manuscripts composing the DIVA-HisDB dataset. The last four columns show the mean scores achieved by the models across the different classes

	CB55				CSG18				CSG863				Mean			
	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1	Prec	Rec	IoU	F1
Ours (baseline)	0.907	0.900	0.815	0.884	0.926	0.923	0.860	0.912	0.917	0.914	0.840	0.900	0.917	0.912	0.838	0.899
Ours (w/ dynamic crop gen.)	0.918	0.908	0.827	0.894	0.926	0.923	0.855	0.912	0.931	0.927	0.863	0.917	0.925	0.919	0.848	0.907
Ours (w/ seg. refinement)	0.979	0.978	0.967	0.976	0.981	0.978	0.963	0.979	0.982	0.980	0.965	0.980	0.981	0.979	0.965	0.978
Ours (w/ both)	**0.989**	**0.987**	**0.977**	**0.988**	**0.983**	**0.982**	**0.967**	**0.982**	**0.986**	**0.983**	**0.971**	**0.984**	**0.986**	**0.984**	**0.972**	**0.985**

Framework modules

Finally, we provide a comparison between different versions of our framework in which we systematically introduce the original modules presented in this work, namely the dynamic instance generation and the segmenta-

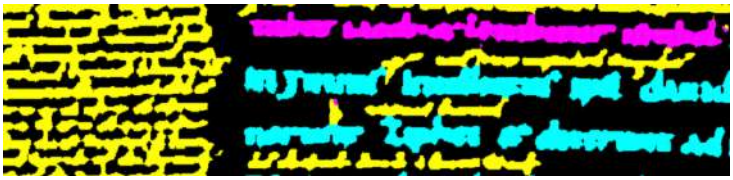
tion refinement ones. In particular, in Tab. 5.6 we show the performance obtained by our baseline model, in which the images have been split into patches but without the addition of either the dynamically generated crops or the segmentation refinement process, as well as the one achieved by introducing these 2 techniques singularly and in a combined fashion, which represents our full framework pipeline. As we can see each of the additional modules leads to a substantial improvement in performance over the baseline approach with the best performance being achieved with the use of both modules. More specifically the final framework achieves an improvement in performance going from 6.8% for the precision metric to a very substantial 13.3% for the Intersection over Union one, with a mean improvement of 9% across all metrics when compared to the baseline approach.

As additional proof of the effectiveness of the proposed approach. In Fig. 5.8 we provide a qualitative comparison between the segmentation masks provided by the baseline and the final framework, while also showing the corresponding ground truth as a reference.

5.5 Conclusions

In this chapter, we proposed an effective framework that tackles the under-explored problem of few-shot document layout analysis by introducing two original modules, namely the dynamic instance generation and segmentation refinement ones which help the core image segmentation backbone to fully leverage the small amount of training data available in order to achieve pixel-precise segmentations of the document pages. When compared to other popular image segmentation algorithms, our model consistently outperforms them, while relying only on a fraction of the training data and with a computational load that is comparable to the one of the original backbone segmentation network adopted, being represented by DeepLabV3+. Furthermore, when compared to the current State of the Art framework, our approach achieves comparable performance on all the selected metrics. While the re-

(a) Ground Truth



(b) Coarse Prediction

(c) Refined Prediction

Figure 5.8: Qualitative results showing the effects of the segmentation refinement process. Fig. 5.8a shows the original ground truth for a zoomed area of the original image. Fig. 5.8b shows the coarse segmentation mask obtained by the model. Finally, Fig. 5.8c shows the segmentation prediction resulting from the refinement process

ported results are very promising, there are still some criticalities we plan to address in the future, specifically by investigating more effective segmentation refinement strategies.

6

Effective Transfer Learning for Document Layout Analysis

Semantic segmentation models have shown impressive performance in the context of historical document layout analysis, but their effectiveness is reliant on having access to a large number of high-quality of annotated images for training. A popular approach to address the lack of training data in other domains is to rely on transfer learning to transfer the knowledge learned from a large-scale, general-purpose dataset (e.g. ImageNet) to a domain-specific task. However, this approach has been shown to lead to unsatisfactory results when the target task is completely unrelated to the data employed for the pre-training process, which is the case when working on document layout analysis. For this reason, in the present paper, we provide an overview of domain-specific transfer learning for document layout segmentation. In particular, we show how relying on document-related images for the pre-training process leads to consistently improved performance and faster convergence compared to training from scratch or even relying on a large, general-purpose, dataset such as ImageNet.

6.1 Introduction

The availability of a sufficient amount of high-quality, annotated data from historical documents is limited in the literature. As stated in the previous

chapter, this is mainly due to the specialist nature of the content. Indeed, annotating data requires domain expertise or specialist knowledge, and obtaining Ground Truth (GT) from domain experts can be time-consuming and costly. To address this problem, several alternative approaches have been proposed in the literature to perform the aforementioned task while relying on a limited amount of data. These approaches involve the adoption of unsupervised learning, few-shot learning and transfer learning techniques. Transfer learning, which is the focus of the following chapter, involves preliminary training of a model on a large annotated dataset in order to learn a set of baseline parameters that will serve as an initialization for the fine-tuning on the dataset connected to the target task. Transfer learning has become a widely adopted technique in the field of computer vision [138, 15, 61], especially following the release of large-scale, general-purpose, datasets such as ImageNet [38], CIFAR-10 [62], PASCAL [43] and COCO [63]. While these datasets provide a valuable resource for the pre-training of deep learning models on a wide array of tasks, they fall short when the downstream task is related to a very specific domain (e.g. medical imaging, handwritten document analysis, etc.). For this reason in the present chapter, we provide a study on the effects of pre-training on domain-specific datasets as opposed to general-purpose ones in the context of document layout analysis of ancient manuscripts. In particular, we show how performing the pre-training step on a dataset that is related to the downstream tasks substantially improves the performance of the model and allows for a faster convergence compared to performing the pre-training on a general-purpose dataset or training the model from scratch. The rest of the chapter is organized as follows: Section 6.2 focuses on presenting a detailed description and setup of the proposed experiments including a thorough presentation of the, currently private, dataset employed to pre-train our system; in Section 6.3 the results of the experiments are presented; finally, in Section 6.4, the conclusions and future works are drawn.

6.2 Methods

This section provides a comprehensive overview of the chosen architecture and training setup, as well as the detailed datasets used for the experiments and evaluation metrics.

6.2.1 Model Architecture

For our experiment, we chose a recent and popular model for semantic segmentation called DeepLabv3+ [27]. This architecture improves upon its predecessor DeepLabv3 [25], by incorporating an encoder-decoder architecture that combines the benefits of both high-resolution and low-resolution features. It utilizes a powerful backbone network, such as ResNet or Xception, as the encoder to extract high-level semantic features from the input image. These features are then fed into an Atrous Spatial Pyramid Pooling module, which captures multi-scale contextual information. In addition, DeepLabv3+ employs a decoder network that refines the segmentation results by upsampling the low-resolution features and combining them with the high-resolution features from the encoder. This decoder network helps to recover spatial details and produce more accurate segmentation maps. This model and DeepLabv3 have already been used in other works on the layout segmentation task of ancient manuscript datasets with excellent results [34, 113]. For the experiments presented in this paper, the ResNet50 version was chosen as the backbone of the DeepLabv3+ architecture.

6.2.2 Datasets descriptions

For the pre-training of the selected model, we relied on two datasets. The first one is the popular ImageNet dataset [38], which serves as our general-purpose pre-training baseline. The second one, on the other hand, is a private domain-specific dataset, which we will refer to as "U-DIADS-Bib" and which will be described hereafter.

(a) CB55 (b) CSG18 (c) CSG863 (d) Bukhari et al.

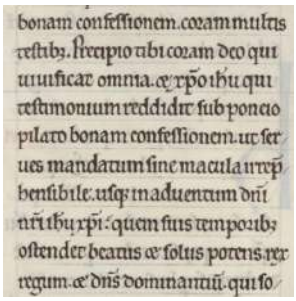
(e) CB55 GT (f) CSG18 GT (g) CSG863 GT (h) Bukhari et al. GT

Figure 6.1: Samples from the 3 manuscripts of Diva-HisDB dataset (CB55 Fig. 6.1a, CSG18 Fig. 6.1b, and CSG863 Fig. 6.1c) and from Bukhari et al. dataset (Fig. 6.1d). For each sample the relative segmentation mask is shown (Fig. 6.1e– 6.1h)

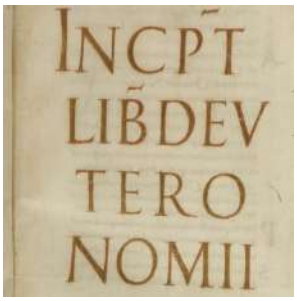
U-DIADS-Bib

U-DIADS-Bib is composed of 150 images, 50 for each of the 3 different manuscripts that characterize it. These handwritten books were selected in collaboration with humanist partners considering both the complexity of their layout and the presence of significant and semantically distinguishable elements. In particular, the images of the three manuscripts were collected from the French digital library Gallica[1]. All manuscripts are Latin Bibles published between the 6th and 12th centuries A.D. which will be briefly described hereafter:

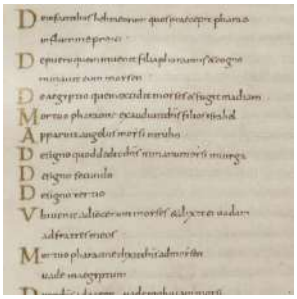
- Paris, Bibliothèque nationale de France, Latin 2[2]. The manuscript,


(a) Main Text


(b) Decoration



(c) Title


(d) Chapter Headings


(e) Paratext


(f) Background

Figure 6.2: Samples of the six segmentation classes of the pre-training dataset: Main Text (6.2a), Decoration (6.2b), Title (6.2c), Chapter Headings (6.2d) , Paratext (6.2e) and Background (6.2f).

known as the Second Bible of Charles the Bald, was produced between A.D. 871 and 877 at the Abbey of Saint-Amand (Haute-France); it was kept in the Abbey of Saint-Denis between A.D. 877 and 1595 and later transferred to the Royal Library of France. It is composed of 444 parchment pages and the layout is structured in two columns.

- Paris, Bibliothèque nationale de France, Latin 14396[3]. The manuscript was produced between A.D. 1145 and 1150 in the Abbey of Saint-Victor (Paris). It contains the biblical text from the Book of Ezra to the Book of Revelation and it is probably the final part of a three-volume Bible; this codex, known as Genesis-Kings, is the first volume. The manuscript is composed of 170 parchment pages and the layout is structured in two columns.

- Paris, Bibliothèque nationale de France, Latin 16746[4]. The manuscript

(a) Latin 2, page 144 original
(b) Latin 14396, page 325 original
(c) Latin 16746, page 187 original
(d) Latin 2 ground truth
(e) Latin 14396 ground truth
(f) Latin 16746 ground truth

Figure 6.3: Images showing a page of each manuscript (Fig. 6.3a– 6.3c) as well as its corresponding ground truth mask (Fig. 6.3d– 6.3f), in which each color represents a different semantic class of the document layout.

was produced between A.D. 1170 and 1190 at the Abbey of Saint Bertin (Pas-de-Calais). It contains the New Testament and it is the final part of a four-volume Bible (Paris, Bibliothèque nationale de France Latin 16743-16746). The Bible had been kept for a long time in the Capuchin convent of Saint-Honoré, in Paris. The manuscript is composed of 176 parchment pages and the layout is structured in two columns.

Layout classes The six segmentation classes highlighted by humanist experts for the page segmentation task are visible in Fig. 6.2 and are:

- **Main Text:** This class includes the writing area and represents the core and central content of the book. This class includes punctuation and pause marks. It can be structured in different layouts such as one or two columns.
- **Decoration:** This class includes both figurative elements in the proper sense (miniatures and decorated initials), and elements of minimal decoration, such as initials with a graphic element or color distinguishing them from the rest of the text.
- **Title:** More properly *'incipit and explicit formulae'*, identifiable by the use of a different ink color and/or by the adoption of display scripts: monumental (square) or rustic capital, uncials or mixed capital/uncials script.
- **Chapter Headings:** The chapter headings function in ancient manuscripts was to facilitate the retrieval of a particular chapter or passage, but according to different scansion and interpretation from one set to another. From a graphical point of view, they are often recognizable by a script similar to that of the main text but smaller in size.
- **Paratext:** This class consists of several elements such as glosses, marginal and interlinear notes, corrections, paragraph numbering, possession notes from different periods, page or fascicle numbering. In general, includes all hand annotations that do not fit into the other classes.
- **Background:** This class includes the background of the page and any outline visible in the scanned image.

[1]Source https://gallica.bnf.fr
[2]https://gallica.bnf.fr/ark:/12148/btv1b8452767n
[3]https://gallica.bnf.fr/ark:/12148/btv1b84429190
[4]https://gallica.bnf.fr/ark:/12148/btv1b85144288

Ground Truth construction The annotation of the U-DIADS-Bib is the result of the collaboration between humanists and computer scientists. As previously addressed, manual annotation of images is a very time-consuming task, especially in the context of document segmentation where the different layout components can be very small and detailed, whilst annotations provided by algorithms tend to present many inaccuracies and are prone to the introduction of noise. Consequently, since our goal was to produce a dataset with a large number of annotated pages, pixel-precise segmentation and very limited, if any, noise, we have defined a segmentation pipeline that involves the alternation of humanist and algorithmic work, in order to optimize the expected results.

First of all, after having chosen the manuscripts, a subset of 50 images was selected in such a way as to represent all the chosen segmentation classes for each manuscript. A subset of 10 images per manuscript was selected and binarized using Sauvola threshold technique [100] and morphological operators in order to give the human experts a starting point to work on. Then, experts in manuscript texts have manually segmented at pixel level with different colors these few images per manuscript contain examples of all expected classes. The next step was to train a machine-learning model with these subsets of images and the related GTs segmented by humanists to obtain a coarse segmentation of the entire dataset. To achieve this, the framework proposed in [35] was followed, where a few-shot pixel-precise document layout segmentation method with high performance is presented. The segmentation was done for 4 classes, so as to obtain a less detailed but well-defined segmentation on the whole dataset and with the presence of almost zero noise. Finally, the expert humanists introduced the other missing semantic classes and meticulously refined and corrected all the color masks of the GTs by comparing them to the original images. It is worth noting that, despite the task being computer-aided, the final result is always defined by a human expert, thus avoiding possible biases or errors in the dataset.

Fig. 6.3 illustrate some examples of the defined GT and corresponding

original image for each manuscript of the U-DIADS-Bib. Each selected pixel is marked by a color that symbolizes the corresponding content type.

U-DIADS-Bib is therefore composed of 50 original color page images for each manuscript, stored in JPEG image format with resolution 1344×2016 px. Each page is associated with the corresponding GT data, stored in a PNG image with the same size as the original one. GTs contain six different and non-overlapping annotated classes (background (BG), comment, decoration, text, title and chapter headings (CH)) encoded by RGB value as follows:

- RGB(0,0,0) Black: Background
- RGB(255,255,0) Yellow: Paratext
- RGB(0,255,255) Cyan: Decoration
- RGB(255,0,255) Magenta: Main Text
- RGB(255,0,0) Red: Title
- RGB(0,255,0) Lime: Chapter Headings

Out of the 150 images, 120 have been used to train the model and the remaining 30 for the validation step.

Evaluation datasets

To test our approach we chose to use the two popular datasets for document layout analysis described in the previous chapter: Diva-HisDB dataset [107] and the dataset of Bukhari et al. [16]. The layout of these manuscripts is particularly challenging and very different from each other, as visible in Figure 6.1.

6.2.3 Evaluation setup

To evaluate the performance of our proposed approach, we employ four metrics, namely Precision, Recall, Intersection over Union (IoU), and F1-Score.

In particular, for the present study, we considered the class-wise macro average of the selected metrics. These metrics are computed separately for each manuscript in the Diva-HisDB dataset and are defined as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{6.1}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{6.2}$$

$$\text{IoU} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive} + \text{False Negative}} \tag{6.3}$$

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6.4}$$

6.2.4 Training and fine-tuning setup

The pre-training of the DeepLabv3+ architecture on the private, domain-specific, dataset was carried out by relying on the popular Adam optimizer with a learning rate of 10^{-4}, a weight decay of 10^{-5} and a batch size of 3. The images were only slightly resized to 1344×2016 px, roughly one-third of the original size, to retain as much detail as possible. The same hyper-parameters were also used for the fine-tuning process and for training the model from scratch on the dataset of the downstream task.

To address the high-class imbalance in each manuscript of the dataset, we selected a weighted Cross Entropy Loss as the loss function for training the model. Specifically, the weight assigned to each class, denoted as W_i, is calculated as the square root of the reciprocal of the occurrence frequency of the corresponding class in the dataset (as shown in Equation 6.5).

$$W_i = \sqrt{\frac{1}{F_i}} \tag{6.5}$$

The same loss function was used for training the model from scratch and for the fine-tuning process, adjusting the weights to the respective segmentation classes present in the Bukhari et al. and Diva-HisDB datasets (Table 6.1).

	CB55	CSG18	CSG863	Bukhari et al.
Background	82.41	85.16	77.82	86.07
Comment	8.36	6.78	6.35	4.71
Decoration	0.55	1.47	1.83	—
Text	8.68	6.59	14.00	9.22

Table 6.1: Classes distribution (%) for each manuscript of Diva-HisDB and for Bukhari et al. dataset.

For the pre-trained ImageNet model relied on the one made available in the Segmentation Models Pytorch repository [55].

On the other hand, when pre-training on the domain-specific dataset we allowed a maximum of 500 epochs while introducing an early stop mechanism that was triggered in case the performance of the model on the validation set didn't improve in the previous 20 epochs. Finally, when training the models from scratch and for the fine-tuning process, the architectures were trained for a total 100 epochs, with a checkpoint saved every 10. The fine-tuning process, in particular, was carried out by freezing the weights of the encoder, which performs the feature extraction, while updating the weights of the decoder module of the network together with the new segmentation head, introduced to match the number of classes present in the downstream dataset.

6.3 Results

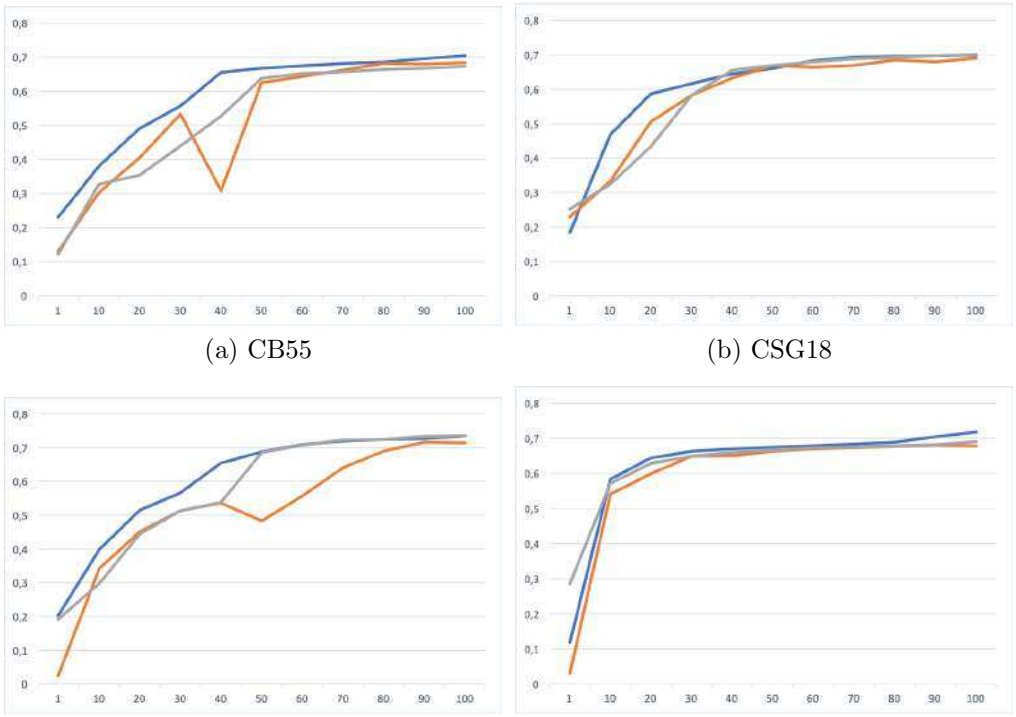
In Tab. 6.2 is reported the performance of the DeepLabv3+ model when trained from scratch on the target datasets and when pre-trained on both the ImageNet and the domain-specific datasets. As we can see the model pre-trained on the domain-specific dataset consistently outperforms the other 2 across all the selected metrics on all the manuscript classes, with the exception of the recall on the Bukhari et al. dataset. In particular, domain-specific

Manuscript	Metric	Scratch	Pre-Trained (ImageNet)	Pre-Trained (domain specific)	Δ (ImageNet)	Δ (domain specific)
Bukhari	Precision	0.730	0.738	**0.780**	+0.008	+0.050
	Recall	**0.910**	0.906	0.902	-0.004	-0.008
	IoU	0.678	0.685	**0.706**	+0.007	+0.028
	F1-Score	0.798	0.802	**0.818**	+0.004	+0.020
CB55	Precision	0.722	0.703	0.**737**	-0.019	+0.015
	Recall	0.914	**0.942**	0.920	+0.028	+0.006
	IoU	0.681	0.673	**0.698**	-0.008	+0.017
	F1-Score	0.799	0.791	0.**812**	-0.008	+0.013
CSG18	Precision	0.730	0.735	0.**740**	+0.005	+0.010
	Recall	0.917	**0.936**	0.924	+0.019	+0.007
	IoU	0.698	0.700	**0.702**	+0.002	+0.004
	F1-Score	0.813	**0.815**	**0.815**	+0.002	+0.002
CSG863	Precision	0.778	0.786	**0.792**	+0.008	+0.014
	Recall	0.891	0.896	**0.898**	+0.005	+0.007
	IoU	0.719	0.735	**0.736**	+0.016	+0.017
	F1-Score	0.828	0.834	**0.839**	+0.006	+0.011

Table 6.2: Results on the test set of the different manuscripts of Diva-HisDB and Bukhari et al. datasets. Each metric is calculated for training from scratch and for fine-tuning of ImageNet pre-training and domain-specific pre-training. The results regarding the effect of fine-tuning from ImageNet and from domain-specific is reported in Δ.

pre-training leads to a 1% average improvement on the classes belonging to the Diva-HisDB dataset. Surprisingly, this strategy proved to be even more effective for the Bukhari et al. dataset, which is the only one containing documents written in a different alphabet compared to the one present in the pre-training dataset, with improvements going from a 2% on the F1-score metric to a 5% for the Precision.

Furthermore, in Fig. 6.4 we report the performance, in terms of IoU, at different stages during the training process on the downstream task. As we can see, pre-training on a domain-specific dataset (blue line), leads to improved performance in the early stages of training, allowing for a faster convergence on all the classes with the exception of the Arabic manuscripts



(a) CB55 (b) CSG18

(c) CSG863 (d) Bukhari et al.

Figure 6.4: Overview of the performance of the DeepLabv3+ architecture at different stages of the training process for the 4 document classes. The orange line represents the model trained from scratch on the target datasets, the gray line the model pre-trained on ImageNet and the blue line the model pre-trained on the domain-specific dataset

contained in the Bukhari dataset, for which the curves described by the 3 models appear very similar to each other. Conversely, pre-training the model on a general-purpose dataset such as ImageNet doesn't seem to have the same positive effects, on the contrary, it seems to lead to a decreased performance in the initial phase of the training for two of the classes, namely the CB55 and CSG18 ones. Finally, pre-training on a domain-specific dataset consistently leads to a more stable performance curve during training. This effect is particularly visible for the CB55 and CSG863 document classes where the model trained from scratch presents some very noticeable downward spikes in performance around halfway during the training process.

6.4 Conclusion and Future Work

In the present work, we have shown the advantages of relying on a domain-specific dataset for the pre-training of models in the context of handwritten document layout segmentation. Even when relying on a small amount of data for this purpose the performance of the final model, fine-tuned on the target dataset, is consistently improved compared with the one of a model that has been trained from scratch on the latter or even compared with a model pre-trained on a large scale, general purpose, dataset such as ImageNet. While the obtained results are already promising we believe that an important limiting factor of the present study has been represented by the relatively small amount of data present in the private dataset we used for the pre-training, which is not even comparable to the large datasets commonly used for this purpose. A further limitation is represented by the homogeneity of the said dataset, mainly in terms of the alphabet in it contained. For these reasons in future works, we plan to expand and make publicly available the dataset employed in this work so as to understand the effects it would have on the performance achieved on the target task as well as to make the results obtained reproducible. Finally, we would like to gain a better understanding of the effectiveness of employing a transfer learning approach in a cross-task scenario where the model is pre-trained on a set document analysis task and fine-tuned on a different one.

7

Conclusions

In this thesis, the problem of semantic segmentation in low-data settings has been addressed. In particular, in chapters 3 and 4 two, attention-based frameworks have been proposed to tackle the problem of anomaly segmentation for industrial quality control and the identification of pathologies in medical images. The former showcased a preliminary work introducing the positive effects of attention-based mechanisms for this class of problems while the latter built upon it and presented a full framework revolving around the concept of self-attention introduced by the Vision Transformer architecture. Furthermore, Chapter 4 showcased the effectiveness of relying on self-attention at multiple scales from the original image, to leverage the full information contained in them.

The following two chapters, on the other hand, focused on the problem of layout segmentation in ancient handwritten documents which, compared to printed ones, present a much higher degree of complexity and for which generating the ground truth segmentation maps corresponding to the original page images represents an even more difficult problem. In chapter 5 a few-shot segmentation approach for this class of manuscripts is proposed. This approach relies on the combination between a robust semantic segmentation deep neural network, a dynamic instance generation module, which allows to take full advantage of the very small amount of instances available, and a traditional computer vision binarization algorithm to achieve state-of-the-art performance while relying on just a fraction of the training data compared

to the previously available approaches. Finally in chapter 6 an analysis of transfer learning strategies in the context of document layout analysis is provided. In particular, it shows how traditional pre-training approaches are not very effective in this application domain, and relying on a domain-specific, albeit much smaller, dataset is preferable.

For all the proposed approaches a quantitative comparison with competition models, through the adoption of popular metrics, has been provided to prove their effectiveness on the corresponding task. Furthermore, for chapters 4 and 5 the qualitative results achieved by the proposed approaches are reported, to provide a clearer picture of the quality of the segmentation maps they produced, as well as to showcase their fallbacks.

Bibliography

[1] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara. Latent space autoregression for novelty detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, volume -, pages 481–490, Los Alamitos, CA, USA, jun 2019. IEEE Computer Society.

[2] Mehran Ahmadlou and Hojjat Adeli. Enhanced probabilistic neural network with local decision circles: A robust classifier. *Integrated Computer-Aided Engineering*, 17:197–210, 2010.

[3] Salem Saleh Al-Amri, NV Kalyankar, and SD Khamitkar. Image segmentation by using edge detection. *International journal on computer science and engineering*, 2(3):804–807, 2010.

[4] Reem Alaasam, Berat Kurar, and Jihad El-Sana. Layout analysis on challenging historical arabic manuscripts using siamese network. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 738–742, Sydney, Australia, 2019.

[5] Kazi Md. Rokibul Alam, Nazmul Siddique, and Hojjat Adeli. A dynamic ensemble learning algorithm for neural networks. *Neural Comput. Appl.*, 32(12):8675–8690, jun 2020.

[6] Amir Alush and Jacob Goldberger. Break and conquer: Efficient correlation clustering for image segmentation. In Edwin Hancock and Marcello Pelillo, editors, *Similarity-Based Pattern Recognition*, pages 134–147, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[7] Jerone Andrews, Thomas Tanay, Edward J Morton, and Lewis D Griffin. Transfer representation-learning for anomaly detection. In *Pro-*

ceedings of the International Conference on Machine Learning (ICML). JMLR, 2016.

[8] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. The MVTec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129(4):1038–1059, 2021.

[9] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD — a comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9584–9592. IEEE, 2019.

[10] Paul Bergmann, Xin Jin, David Sattlegger, and Carsten Steger. The MVTec 3d-AD dataset for unsupervised 3d anomaly detection and localization. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, volume 5, pages 202–213. SciTePress, 2022.

[11] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *International joint conference on computer vision, imaging and computer graphics theory and applications*, 2019.

[12] Paul Bergmann., Sindy Löwe., Michael Fauser., David Sattlegger., and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 372–380. INSTICC, SciTePress, 2019.

[13] Galal M. Binmakhashen and Sabri A. Mahmoud. Document layout analysis: A comprehensive survey. *ACM Comput. Surv.*, 52(6), oct 2019.

[14] Sanket Biswas, Pau Riba, Josep Lladós, and Umapada Pal. Beyond document object detection: Instance-level segmentation of complex layouts. *Int. J. Doc. Anal. Recognit.*, 24(3):269–281, sep 2021.

[15] Andrzej Brodzicki, Michal Piekarski, Dariusz Kucharski, Joanna Jaworek-Korjakowska, and Marek Gorgon. Transfer learning methods as a new approach in computer vision tasks with small datasets. *Foundations of Computing and Decision Sciences*, 45(3):179–193, 2020.

[16] Syed Saqib Bukhari, Thomas M. Breuel, Abedelkadir Asi, and Jihad El-Sana. Layout analysis for arabic historical document images using machine learning. In *2012 International Conference on Frontiers in Handwriting Recognition*, pages 639–644, Bari, Italy, 2012.

[17] Tobias Böttger and Markus Ulrich. Real-time texture error detection on textured surfaces with compressed sensing. *Pattern Recognition and Image Analysis*, 26:88–94, 01 2016.

[18] C. Wiedemann C. Steger, M. Ulrich. *Machine Vision Algorithms and Applications*. Wiley, 2018.

[19] Kai Chen, Cheng-Lin Liu, Mathias Seuret, Marcus Liwicki, Jean Hennebert, and Rolf Ingold. Page segmentation for historical document images based on superpixel classification with unsupervised feature learning. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 299–304, Santorini,Greece, 2016.

[20] Kai Chen, Mathias Seuret, Marcus Liwicki, Jean Hennebert, and Rolf Ingold. Page segmentation of historical document images with convolutional autoencoders. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1011–1015, Nancy, France, 2015.

[21] Kai Chen, Mathias Seuret, Marcus Liwicki, Jean Hennebert, Cheng-Lin Liu, and Rolf Ingold. Page segmentation fosar historical hand-

written document images using conditional random fields. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 90–95, Shenzen, China, 2016.

[22] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[23] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915, 2016.

[24] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587, 2017.

[25] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[26] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[27] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 833–851, Cham, 2018. Springer International Publishing.

[28] Zhaomin Chen, Chai Kiat Yeo, Bu Sung Lee, and Chiew Tong Lau. Autoencoder-based network anomaly detection. In *2018 Wireless Telecommunications Symposium (WTS)*, pages 1–5. IEEE, 2018.

[29] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, Los Alamitos, CA, USA, jul 2017. IEEE Computer Society.

[30] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[31] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005.

[32] Xu Dan, Elisa Ricci, Yan Yan, Jingkuan Song, Nicu Sebe, et al. Learning deep representations of appearance and motion for anomalous event detection. In *Proceedings of the British Machine Vision Conference*, pages 8–1. BMVA Press, 2015.

[33] Homa Davoudi, Marco Fiorucci, and Arianna Traviglia. Ancient document layout analysis: Autoencoders meet sparse coding. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5936–5942, Milan, Italy, 2021.

[34] Axel De Nardin, Silvia Zottin, Matteo Paier, Gian Luca Foresti, Emanuela Colombi, and Claudio Piciarelli. Efficient few-shot learning for pixel-precise handwritten document layout analysis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3680–3688, Waikoloa, Hawaii, January 2023.

[35] Axel De Nardin, Silvia Zottin, Claudio Piciarelli, Emanuela Colombi, and Gian Luca Foresti. Few-shot pixel-precise document layout segmentation via dynamic instance generation and local thresholding. *Available at SSRN 4333692*, 2023.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, IEEE, 2009.

[38] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, Miami, FL, 2009.

[39] 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.

[40] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*. OpenReview.net, 2021.

[41] Ahmad Droby, Berat Kurar Barakat, Borak Madi, Reem Alaasam, and Jihad El-Sana. Unsupervised deep learning for handwritten page segmentation. In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 240–245, 2020.

[42] Sarah M Erfani, Sutharshan Rajasegarar, Shanika Karunasekera, and Christopher Leckie. High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. *Pattern Recognition*, 58:121–134, 2016.

[43] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010.

[44] Shaohua Fan, Chuan Shi, and Xiao Wang. Abnormal event detection via heterogeneous information network embedding. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 1483–1486, 2018.

[45] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020.

[46] Angelika Garz, Mathias Seuret, Fotini Simistira, Andreas Fischer, and Rolf Ingold. Creating ground truth for historical manuscripts with document graphs and scribbling interaction. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 126–131, Santorini, Greece, 2016.

[47] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[48] Monica Gruosso, Nicola Capece, and Ugo Erra. Human segmentation in surveillance video with deep learning. *Multimedia Tools and Applications*, 80:1175–1199, 2021.

[49] 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.

[50] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, Nevada, 2016.

[51] Andrew Howard, Mark Sandler, Bo Chen, Weijun Wang, Liang-Chieh Chen, Mingxing Tan, Grace Chu, Vijay Vasudevan, Yukun Zhu, Ruoming Pang, Hartwig Adam, and Quoc Le. Searching for mobilenetv3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1314–1324, Seoul, South Korea, 2019.

[52] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[53] Yong-Ren Huang and Chung-Ming Kuo. Image segmentation using edge detection and region distribution. In *2010 3rd International Congress on Image and Signal Processing*, volume 3, pages 1410–1414, 2010.

[54] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 603–612, 2019.

[55] Pavel Iakubovskii. Segmentation models pytorch, 2019.

[56] Nabil Ibtehaz and M. Sohel Rahman. Multiresunet : Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, 121:74–87, 2020.

[57] Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2019.

[58] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–7, 2020.

[59] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9404–9413, 2019.

[60] Felipe Kitamura. Head ct - hemorrage dataset. https://www.kaggle.com/felipekitamura/head-ct-hemorrhage, 2018.

[61] Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[62] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images, 2009.

[63] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

[64] Francesco Lombardi and Simone Marinai. Deep learning for historical document analysis and recognition—a survey. *Journal of Imaging*, 6(10):110, Oct 2020.

[65] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[66] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, Boston, MA, 2015.

[67] W.Y. Ma and B.S. Manjunath. Edge flow: A framework of boundary detection and image segmentation. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 744–749, 1997.

[68] Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognition*, 110:107332, 2021.

[69] Priyanka Malhotra, Sheifali Gupta, Deepika Koundal, Atef Zaguia, Wegayehu Enbeyle, et al. Deep neural networks for medical image segmentation. *Journal of Healthcare Engineering*, 2022, 2022.

[70] M. Mary Synthuja Jain Preetha, L. Padma Suresh, and M. John Bosco. Image segmentation using seeded region growing. In *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, pages 576–583, 2012.

[71] Maroua Mehri, Nibal Nayef, Pierre Héroux, Petra Gomez-Krämer, and Rémy Mullot. Learning texture features for enhancement and seg-

mentation of historical document images. In *Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing*, HIP '15, page 47–54, New York, NY, USA, 2015. Association for Computing Machinery.

[72] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542, 2021.

[73] Pankaj Mishra, Claudio Piciarelli, and Gian Luca Foresti. A neural network for image anomaly detection with deep pyramidal representations and dynamic routing. *International Journal of Neural Systems*, 30(10):2050060–2050060, 2020.

[74] Pankaj Mishra, Riccardo Verk, Daniele Fornasier, Claudio Piciarelli, and Gian Luca Foresti. VT-ADL: A vision transformer network for image anomaly detection and localization. In *30th IEEE/IES International Symposium on Industrial Electronics (ISIE)*, June 2021.

[75] Laurent Najman and Michel Schmitt. Watershed of a continuous function. *Signal Processing*, 38(1):99–112, 1994. Mathematical Morphology and its Applications to Signal Processing.

[76] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by cnn-based self-similarity. *Sensors*, 18(1):209, 2018.

[77] Minh-Nghia Nguyen and Ngo Anh Vien. Scalable and interpretable one-class svms with deep learning and random fourier features. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2018, Dublin, Ireland, September 10–14, 2018, Proceedings, Part I 18*, pages 157–172. Springer, 2019.

[78] W Niblack. *An Introduction to Digital Image Processing*. Prentice hall, Englewood Cliffs, 1986.

[79] Konstantina Nikolaidou, Mathias Seuret, Hamam Mokayed, and Marcus Liwicki. A survey of historical document image datasets. *Int. J. Doc. Anal. Recognit.*, 25(4):305–338, dec 2022.

[80] Min-hwan Oh and Garud Iyengar. Sequential anomaly detection using inverse reinforcement learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & data mining*, pages 1480–1490, 2019.

[81] Ozan Oktay, Jo Schlemper, Loïc Le Folgoc, M. J. Lee, Mattias P. Heinrich, Kazunari Misawa, Kensaku Mori, Steven G. McDonagh, Nils Y. Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention u-net: Learning where to look for the pancreas. *ArXiv*, abs/1804.03999, 2018.

[82] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2041–2050, 2018.

[83] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021.

[84] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. Deep weakly-supervised anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1795–1807, 2023.

[85] Guansong Pang, Chunhua Shen, and Anton van den Hengel. Deep anomaly detection with deviation networks. In *Proceedings of the 25th*

ACM SIGKDD international conference on knowledge discovery & data mining, pages 353–362, 2019.

[86] Guansong Pang, Cheng Yan, Chunhua Shen, Anton van den Hengel, and Xiao Bai. Self-trained deep ordinal regression for end-to-end video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12173–12182, 2020.

[87] Guansong Pang, Cheng Yan, Chunhua Shen, Anton van den Hengel, and Xiao Bai. Self-trained deep ordinal regression for end-to-end video anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12173–12182, 2020.

[88] Danillo Pereira, Marco Piteri, André Souza, João Papa, and Hojjat Adeli. Fema: A finite element machine for fast learning. *Neural Computing and Applications*, 32:6393–6404, 2020.

[89] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2898–2906, 2019.

[90] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, -:2893–2901, 2019.

[91] Claudio Piciarelli, Danilo Avola, Daniele Pannone, and Gian Luca Foresti. A vision-based system for internal pipeline inspection. *IEEE Transactions on Industrial Informatics*, 2018. early access.

[92] Claudio Piciarelli, Christian Micheloni, and Gian Luca Foresti. Trajectory-based anomalous event detection. *IEEE Transaction on Circuits and Systems for Video Technology*, 18(11):1544–1554, 2008.

[93] Mohammad Hossein Rafiei and Hojjat Adeli. A new neural dynamic classification algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 28(12):3074–3083, 2017.

[94] J. Y. Ramel, S. Leriche, M. L. Demonet, and S. Busson. User-driven page layout analysis of historical printed books. *International Journal of Document Analysis and Recognition (IJDAR)*, 9(2):243–261, Apr 2007.

[95] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[96] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022.

[97] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14298–14308, 2022.

[98] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3379–3388, 2018.

[99] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 14902–14912. IEEE, 2021.

[100] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236, 2000.

[101] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54:30–44, 2019.

[102] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer, Springer, 2017.

[103] Thomas Schlegl, Philipp Seeböck, Sebastian Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, 54:30–44, 2019.

[104] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[105] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[106] Foteini Simistira, Mathias Seuret, Nicole Eichenberger, Angelika Garz, Marcus Liwicki, and Rolf Ingold. Diva-hisdb: A precisely annotated large dataset of challenging medieval manuscripts. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 471–476, Shenzen, China, 2016.

[107] Foteini Simistira, Mathias Seuret, Nicole Eichenberger, Angelika Garz, Marcus Liwicki, and Rolf Ingold. Diva-hisdb: A precisely annotated large dataset of challenging medieval manuscripts. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 471–476, Shenzen, China, 2016.

[108] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[109] Samarth Sinha, Han Zhang, Anirudh Goyal, Yoshua Bengio, Hugo Larochelle, and Augustus Odena. Small-gan: Speeding up gan training using core-sets. In *International Conference on Machine Learning*, pages 9005–9015. PMLR, 2020.

[110] V. Sivakumar and V. Murugesh. A brief study of image segmentation using thresholding technique on a noisy image. In *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pages 1–6, 2014.

[111] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7262–7272, 2021.

[112] Linda Studer, Michele Alberti, Vinaychandran Pondenkandath, Pinar Goktepe, Thomas Kolonko, Andreas Fischer, Marcus Liwicki, and Rolf Ingold. A comprehensive study of imagenet pre-training for historical document image analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 720–725, Sydney, Australia, 2019.

[113] Linda Studer, Michele Alberti, Vinaychandran Pondenkandath, Pinar Goktepe, Thomas Kolonko, Andreas Fischer, Marcus Liwicki, and Rolf

Ingold. A comprehensive study of imagenet pre-training for historical document image analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 720–725, 2019.

[114] Akira Suga, Keita Fukuda, Tetsuya Takiguchi, and Yasuo Ariki. Object recognition and segmentation using sift and graph cuts. In *2008 19th International Conference on Pattern Recognition*, pages 1–4, 2008.

[115] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488, 2018.

[116] Solène Tarride, Aurélie Lemaitre, Bertrand Coüasnon, and Sophie Tardivel. Combination of deep neural networks and logical rules for record segmentation in historical handwritten registers using few examples. *International Journal on Document Analysis and Recognition (IJDAR)*, 24(1):77–96, Jun 2021.

[117] David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54:45–66, 2004.

[118] Tuan Anh Tran, In Seop Na, and Soo Hyung Kim. Page segmentation using minimum homogeneity algorithm and adaptive mathematical morphology. *International Journal on Document Analysis and Recognition (IJDAR)*, 19(3):191–209, Sep 2016.

[119] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. Unmasking the abnormal events in video. In *Proceedings of the IEEE international conference on computer vision*, pages 2895–2903, 2017.

[120] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008. Curran Associates, Inc., 2017.

[121] Hai Wang, Yanyan Chen, Yingfeng Cai, Long Chen, Yicheng Li, Miguel Angel Sotelo, and Zhixiong Li. Sfnet-n: An improved sfnet algorithm for semantic segmentation of low-light autonomous driving road scenes. *IEEE Transactions on Intelligent Transportation Systems*, 23(11):21405–21417, 2022.

[122] Hu Wang, Guansong Pang, Chunhua Shen, and Congbo Ma. Unsupervised representation learning by predicting random distances. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2950–2956, 2021.

[123] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021.

[124] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[125] Peng Wu, Jing Liu, and Fang Shen. A deep one-class neural network for anomalous event detection in complex scenes. *IEEE transactions on neural networks and learning systems*, 31(7):2609–2622, 2019.

[126] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12077–12090. Curran Associates, Inc., 2021.

[127] Yue Xu, Fei Yin, Zhaoxiang Zhang, Cheng-Lin Liu, et al. Multi-task layout analysis for historical handwritten documents using fully convolutional networks. In *IJCAI*, pages 1057–1063, 2018.

[128] Julian Yarkony, Alexander Ihler, and Charless C. Fowlkes. Fast planar correlation clustering for image segmentation. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, pages 568–581, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[129] Pengfei Yu and Xuesong Yan. Stock price prediction based on deep neural networks. *Neural Computing and Applications*, 32(6):1609–1628, 2020.

[130] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706, 2021.

[131] Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. Efficient gan-based anomaly detection. *CoRR*, abs/1802.06222, 2018.

[132] Ke Zhang, Marcus Hutter, and Huidong Jin. A new local distance-based outlier detection approach for scattered real-world data. In *Advances in Knowledge Discovery and Data Mining: 13th Pacific-Asia Conference, PAKDD 2009 Bangkok, Thailand, April 27-30, 2009 Proceedings 13*, pages 813–822. Springer, 2009.

[133] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2017.

[134] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *2017 IEEE Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 6230–6239, Honolulu, hawaii, 2017.

[135] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. PSANet: Point-wise spatial attention network for scene parsing. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 270–286, Cham, 2018. Springer International Publishing.

[136] Panpan Zheng, Shuhan Yuan, Xintao Wu, Jun Li, and Aidong Lu. One-class adversarial nets for fraud detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1286–1293, 2019.

[137] Zhiqin Zhu, Xianyu He, Guanqiu Qi, Yuanyuan Li, Baisen Cong, and Yu Liu. Brain tumor segmentation based on the fusion of deep semantics and edge information in multimodal mri. *Information Fusion*, 91:376–387, 2023.

[138] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021.