Bachelor Project

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Kybernetics

Segmentation of teeth restoration from X-ray bitewing images

David Grundfest

Supervisor: Prof. Dr. Ing. Jan Kybic
Subfield: Medical electronics and bioinformatics
February 2023

Acknowledgements

I would like to express my gratitude to prof. Kybic for supervision of this project, his kind attitude and will to help at any time. I also need to thank MDDr. Nagyová for kind cooperation during annotating and editing annotation. Finally I have to express my gratitude to my family for their never-ending support.

Declaration

I hereby declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the Methodical instructions for observing the ethical principles in the preparation of university thesis.

In Prague, 24. May 2023

Abstract

Dental caries belongs among the most prevalent diseases on a global scale with more than 3.5 billion people affected. It significantly damages teeth and untreated could cause even loss of the tooth. After clearing the tooth from caries, the damages must be cleaned and closed and we need to restore the tooth's integrity and functionality. A restoration is chosen with respect to the damage done and the tooth's position. With progressing technologies, there is no need to use exclusively amalgam fillings. There is quite a number of other materials for fillings or possibilities of different methods. However, these methods are developed to hide damage and make the restoration as invisible as possible. This makes its detection quite challenging.

Teeth restorations are made in such a way, that they resemble teeth texture. This makes it difficult to find them, but it is still possible when cautious. However, it is impossible to tell the restoration's overall shape and how deep it goes. Following this fact, X-ray images, which are widely used for caries detection and overall view of the structure of teeth, are being used even for restoration recognition. However, due to evolving materials, many of them are barely visible and resemble teeth which makes them easy to overlook.

This work addresses this problem and tries to develop a tool for teeth restoration segmentation which could help with the detection and estimation of their size and shape. It follows and improves the work of Mr Kunt. We came up with an idea to filter out images without any restorations with another tool which does not have to make decisions for every pixel. With the employment of an object detection model, our results achieve an average 0.88 in DSC. With the help of our model, dentists could focus their work on troublesome images and double-check their work. Thanks to its high sensitivity it could be used as a screening tool for dentists to quickly check where are any restorations and then annotate them.

Keywords: filling, restoration, teeth, denture, artificial intelligence, convolutional neural network, segmentation

Supervisor: Prof. Dr. Ing. Jan Kybic

Abstrakt

Zubní kaz je jedním z nejrozšířenějších onemocnění na světě. Postihuje více než 3.5 miliard lidí. Výrazně poškozuje chrup a může způsobit až jeho ztrátu. Po odstranění zubního kazu je nutné zbylé tvrdé zubní tkáně překrýt, zabránit dalšímu šíření zubního kazu a obnovit funkčnost zubu. Podle rozsahu poškození zubu a podle jeho polohy v dutině ústní se zvolí vhodná metoda pro opravu. Jak technologie pokročily, nemusí se již využívat pouze amalgámových zubních výplní. Využívá se celá škála materiálů pro zubní výplně nebo i jiné typy ošetření. Tyto ošetření jsou s postupujícími technologiemi méně viditelné přímým pohledem i na rentgenových snímcích.

Novodobá zubní ošetření imitují barvou i materiálem tvrdé zubní tkáně a byť jsou většinou detekovatelné pomocí pečlivé inspekce za pomoci zubní sondy, určení jejich rozsahu zejména do hloubi zubních struktur vyžaduje použití pomocných vyšetřovacích metod. Široce se tedy využívá rentgenových snímků, které se používají i pro detekci zubních kazů a pohledu na celkovou strukturu chrupu.

Tato práce se zaměřuje na segmentaci zubních výplní a ošetření pro usnadnění práce při jejich lokalizaci a odhadu tvaru a velikosti. Navazuje na práci pana Kunta a vylepšuje jeho stávající model. Přišli jsme s tím, nejdříve odfiltrovat snímky, které neobsahují zubní výplně, nějakým jiným nástrojem, který nemusí udělat rozhodnutí pro každý pixel. Po zapojení detekce naše výsledky dosahují průměrných hodnot až 0.88 v DSC. Za pomoci našeho modelu zubaři dále upravují anotace pro další vývoj tohoto nástroje. Díky vysoké citlivosti jej lze využít i jako screening techniku pro nalzení výplní, které se tak snáze označí.

Klíčová slova: výplň, náhrada, zuby, chrup, umělá inteligence, konvoluční neuronová síť, segmetace

Překlad názvu: Segmentace zubních výplní v X-ray bitewing snímcích

Contents

Figures

Tables

ČVUT
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Grundfest** Jméno: **David** Osobní číslo: **499118**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra teorie obvodů**

Studijní program: **Lékařská elektronika a bioinformatika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Segmentace zubních výplní z bitewing rentgenových snímků

Název bakalářské práce anglicky:

Segmentation of dental restorations from bitewing X-ray images

Pokyny pro vypracování:

Seznam doporučené literatury:

1.Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." MICCAI: International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.
2.Kunt L. "Dental caries detection from bitewing X-ray images.", diplomová práce ČVUT FEL, 2022.
3.Redmon, Joseph and Farhadi, Ali YOLOv3: An Incremental Improvement. (2018). , cite arxiv:1804.02767

Jméno a pracoviště vedoucí(ho) bakalářské práce:

prof. Dr. Ing. Jan Kybic algoritmy pro biomedicínské zobrazování FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **20.02.2023** Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce: **16.02.2025**

prof. Dr. Ing. Jan Kybic podpis vedoucí(ho) práce	doc. Ing. Radoslav Bortel, Ph.D. podpis vedoucí(ho) ústavu/katedry	prof. Mgr. Petr Páta, Ph.D. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

.
Datum převzetí zadání Podpis studenta

Chapter 1

Introduction

As of 2017, dental caries was the most prevalent disease globally[12], with more than 3.5 billion affected. There are several ways to treat this condition, each having similar goal - to remove the dental caries in order to prevent it from spreading, to hermetically cover the remaining tooth structures and to restore the integrity and function of the affected tooth [16]. The vast amount of people suffering from dental caries reflects the need for vast amounts of different dental restorations. However not all of them are easily recognisable or even visible.

Attention paid to machine learning and especially neural networks caused a massive improvement in this field over the last decade. It even outperformed human recognition skills in classification task in 2015 [18]. This improvement led to the idea to use image analysis models in a medical field to speed up diagnostic techniques and help doctors to validate their diagnoses. It excels in the early detection of future medical problems, such as breast cancer detection[38].

Given these results, we aim to use a deep-learning model to find and segment dental restorations from X-ray images. Such model could help doctors quickly visualise where the restorations are located and assess their size both superficially and to depth. Clear visualisation of the restoration margins may aid in finding secondary or residual caries which often arise right at the edge of dental restoration. Finally, marking the restored dental surfaces would enable quick export of found restorations into teeth charts (see Figure 1.1). The full code is in this GitHub repository https://github.com/GrunyD/Bc_project

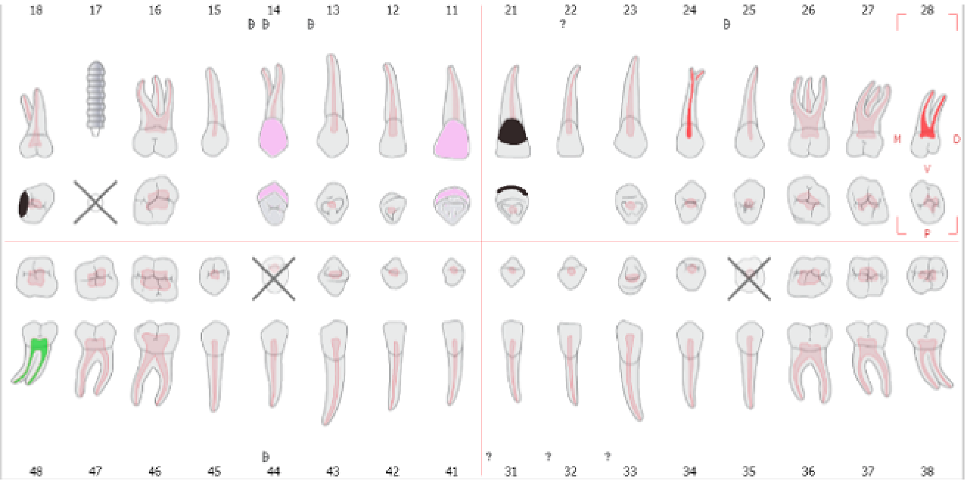


Figure 1.1: Teeth chart used by dentist to write down any work done. Source: www.xdent.cz

Chapter 2

Medical background

2.1 Human teeth

Human dentition is composed of two sets of teeth - primary and permanent. The primary, also called deciduous, consists of 20 teeth and begins to erupt at six months of age. This dentition is completely replaced at the approximate age of 13 years by a permanent set of 32 teeth.

2.1.1 Structure of teeth

Teeth are composed of three structures: Enamel, pulp-dentin complex, and cementum. A picture of teeth structure is depicted in Figure 2.1. The superficial layer covering the anatomic crown of a tooth consists of a highly mineralized crystalline structure called the enamel. More than 90% of the volume is taken up by minerals (hydroxyapatite), making enamel the hardest substance of teeth and even the human body. Its thickness varies from one class of tooth to another, but it ranges from 2 to 3mm on average. Enamel is produced in the process of amelogenesis by cells occurring only in the development stage, meaning that it cannot regenerate. The biggest threat to enamel are acidic conditions, which can cause its demineralization. Enamel has the ability to remineralize, but if the cause is not removed, the enamel is irreversibly damaged, and a cavity is formed.

Pulp-Dentin complex

Pulp and dentin are two specialized connective tissues. However, some sources consider them a single tissue forming a complex [37]. The dental pulp is located in the pulp chamber of the tooth, and it serves four functions: formative, nutritive, sensory, and reparative. The pulp is circumscribed by

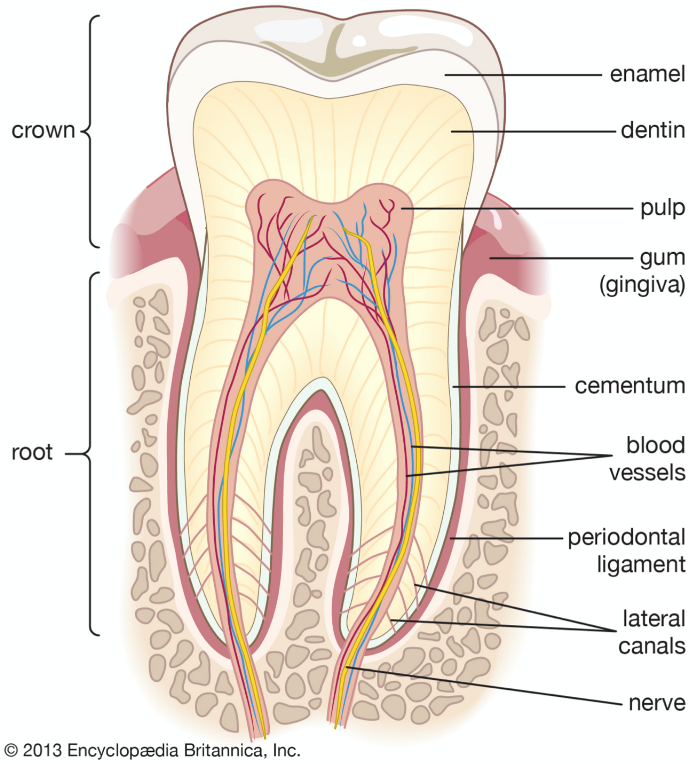


Figure 2.1: Structure of teeth

dentin formed by specific cells in the process of dentinogenesis. Their cell bodies are found in the pulp chamber, but their cytoplasmic cell processes, located in dentinal tubules, extend into the mineralized dentin. Thanks to those processes, dentin is considered to be a living tissue. Its function is to provide the ability to regenerate and react to pathological stimuli, such as blocking the advancement of carious lesions by precipitating minerals in the affected area. Dentin forms the most significant portion of the tooth. In the coronal part, it is covered by the enamel, and on the root of the tooth overlayed by cementum. There are different types of dentin.

- **Primary dentin** forms the outer and most prominent layer of dentin closest to the enamel. It is produced in the development stage of the tooth.
- **Secondary dentin** is formed after the root development is completed.
- **Tertiary (reactive) dentin** production is encouraged as a response to pathological stimuli, such as injury or caries. It is produced at the pulp-dentin interface in order to protect the pulp.
- **Transparent dentin** is characterized by the presence of mineral precipitates in dentinal tubules as a result of injury or aging.

Cementum

Cementum covers the roots of teeth. Its structure consists of approximately 50 % of an anorganic material, 50 % of organic matter, and water, making it slightly softer than dentin and far more delicate than enamel. Together with gingiva, periodontal ligaments, and the alveolar bone, cementum forms periodontium, ensuring that the tooth is attached to the bone. Cementum possesses the ability to repair itself to a limited degree.

2.2 Dental caries

2.2.1 Cause

Dental caries is an infectious disease characterized by the demineralization and destruction of hard dental tissues. The leading cause is dental plaque (also called a biofilm). Plaque is composed of bacteria, their by-products, and salivary proteins, and it has the ability to adhere to the tooth structures. Some bacteria in the plaque metabolize refined dietary carbohydrates and produce organic acid by-products. If present in the biofilm for an extended period of time, those acids can lower the pH in the biofilm to below a critical threshold (5.5 for enamel, 6.2 for dentin)[37]. Low pH drives phosphate and calcium from the tooth into the biofilm in an attempt to reach an equilibrium. This loss of minerals in a tooth is called demineralization and, if not stopped, can lead to a macroscopic loss of hard dental tissues. Once the described pathological process reaches dentin, caries spreads not only by loss of minerals but also by destruction of collagen fibers. While the demineralisation of enamel may be controlled and eventually reverted if the pH returns to neutral and the relative concentration of soluble calcium and phosphate in the biofilm is higher than in the tooth. The cycle of demineralization and remineralization occurs multiple times a day and is modulated by many highly individual and tooth-specific factors.

2.2.2 Treatment

Treatment is suggested based on the progression of the lesion and the patient's risk profile. In some cases, only instructions to increase oral hygiene together with fluoride toothpaste are enough to stop the progression and lead to remineralization of the enamel. The dentist can suggest an application of a sealant to prevent further progression of the lesion. If this treatment is perceived as insufficient or if the carious lesion is already cavitated, restoring the tooth is required. This consists of removing all dental decay and filling the cavity with restorative material such as dental composite or amalgam [37] [14].

2.3 Restorations

Once the irreversible carious lesion is removed, the remaining tooth structures must be sealed and the cavity should be restored to the original anatomy of the tooth to ensure its function and aesthetics. There are different types of restorations, which are indicated based on the position and degree of damage of the affected teeth.

2.3.1 Direct restorations

Such restorations are placed into the cavity while still soft and are shaped with dental tools to reproduce the original anatomy of the tooth. They set directly inside the cavity, either chemically or upon light curing, or as a combination of both. The most commonly used materials for direct restorations include amalgam, resin composite and glass ionomer cement. Historically, gold was also a material of choice for direct restorations. [16]

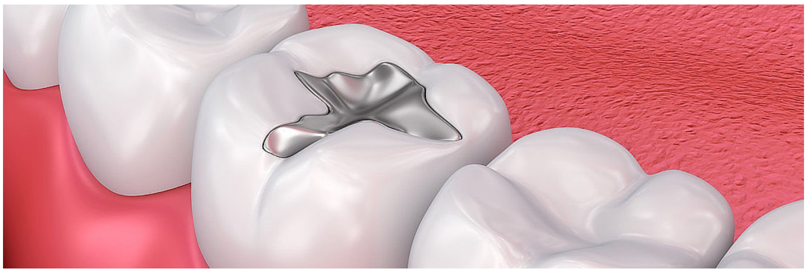


Figure 2.2: Direct restoration using amalgam. Source: www.nzip.cz

Amalgam

The General meaning of the word amalgam is an alloy of mercury and several other metals. Dental amalgam consists mostly of mercury, silver, tin, and copper and has traces of other metals. The debate over consequences of mercury still continues but it is mostly agreed that it is no threat to human body. Due to its grey metallic colour this material is not indicated to restore frontal teeth for poor aesthetic outcome. Thus , dental amalgam is only used to restore cavities in premolars and molars where it serves as a long lasting restorative material if used when specific indication criteria are met.[16] [13].

Resin composite

Resin composites are manufactured in various shades with different optical properties and are perfected to closely resemble the original tooth structures and may even be difficult to detect by inspection and probing only. They are the most commonly used material in modern dentistry for direct restorations of both anterior and posterior teeth. They consist of resin matrices, glass fillers, and silane coupling agents. They also include initiators and inhibitors of polymerization, which is the reaction leading to the setting of the material inside the dental cavity. The polymerization reaction may be initiated by light curing, by chemical reaction, or by both processes simultaneously.

Glass ionomer cement

Glass ionomer cements are mostly used for temporary restorations but may also serve as long-term restorations under certain indication criteria. In literature, they may also be found as glass polyalkenoate cements. Their setting process is based on an acid-base reaction between glass powder and a water solution of polyalkenoic acid. Their advantage is that they bond to dental hard tissues and release fluoride ions over prolonged period of time.

Gold

Gold was used historically for long lasting durable direct restorations with improved mechanical and chemical properties over dental amalgam.

2.3.2 Indirect restorations

Indirect restorations are manufactured and set outside the oral cavity and are attached onto the remaining tooth structures using various materials reffered to as cements. They are commonly made of ceramics or resin composite. Indirect restorations include inlays, onlays and overlays and are classified according to their extent:

- **Inlays** are pre-molded fillings that don't cover the cusps.
- **Onlays** cover at least one cusp but not all of them
- **Overlays** cover all cusps resulting in the coverage of the whole occlusal surface.

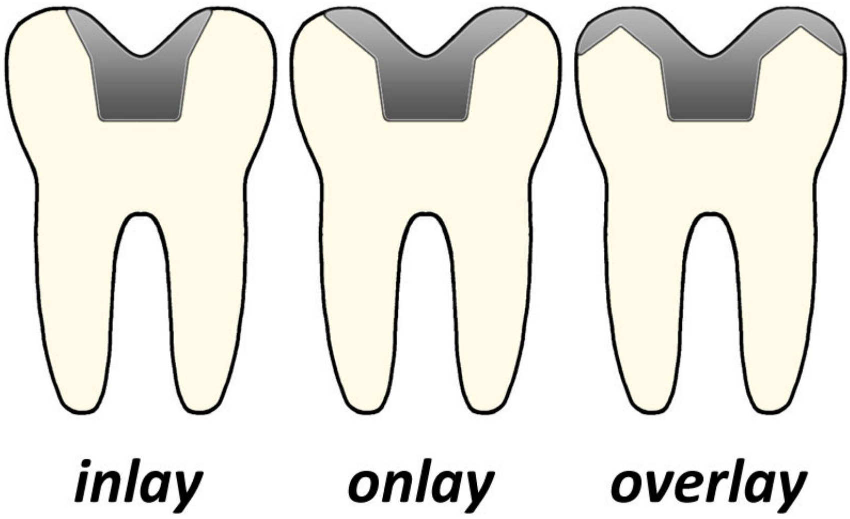


Figure 2.3: Difference among inlays, onlays and overlays. Source: www.dubrovnik-dental.clinic

2.3.3 Dental crowns

Dental crowns cover the anatomical crown of tooth completely and are indicated after a major dental tissue loss to restore the tooth's chewing abilities, its structure and aesthetics. They are also usually manufactured outside the oral cavity and secured onto the tooth using cements.

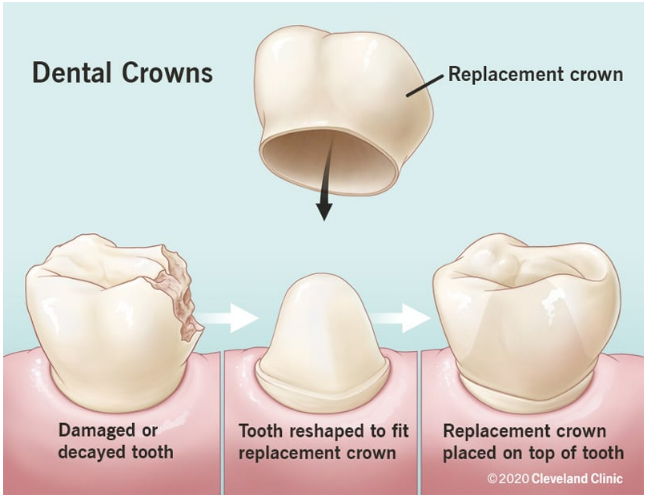

Dental Crowns
Replacement crown
Damaged or decayed tooth
Tooth reshaped to fit replacement crown
Replacement crown placed on top of tooth
©2020 Cleveland Clinic

Figure 2.4: Dental crowns are cemented on the prepared affected tooth. Source: www.clevelandclinic.com

2.3.4 Dental implants

Implants replace missing teeth by placing a fixture directly into a jaw bone. The fixture is covered by an abutment and a dental crown. As it is put directly in bone, patient experience a strong stability of the implant.

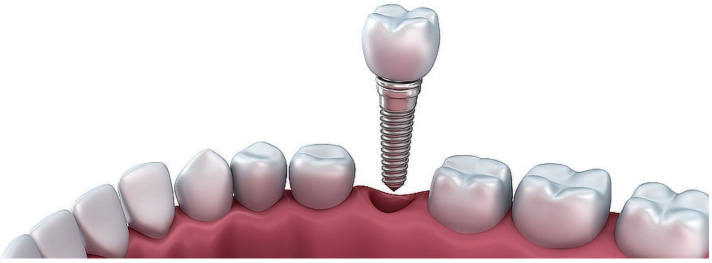


Figure 2.5: Dental implant, Source: www.nzip.cz

2.3.5 Dental bridges

Bridges replace more than one missing teeth. Bridges are used to replace one or more missing teeth. They consist of abutments and pontics. Abutment is the remaining tooth which is prepared and covered by a crown. Pontic is an artificial crown that replaces the missing tooth.

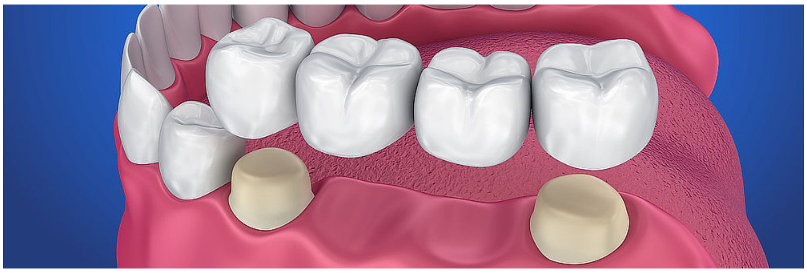


Figure 2.6: Dental bridge covering empty spaces after missing teeth. Source: www.nzip.cz

2.4 X-ray

2.4.1 Diagnosis

Visual-tactile diagnosis is the primary way to inspect teeth. Dentists use a mouth mirror and sharp probe to perform the examination. It is indispensable to dry teeth since the difference in the refractive index between sound and carious enamel is higher when water is removed from the tissue. This increases the chance of spotting a carious lesion before it has an opportunity to progress and cavitate the tooth. The second most used method clinicians use to complement the visual examination is a dental X-ray. In dentistry, two main types of X-ray imaging are taken during the examination: intraoral (the X-ray film is located inside the mouth) and extraoral (the X-ray film is outside the mouth). The intraoral images are the most commonly taken ones. This category includes bitewing and periapical X-rays, each featuring different aspects of the teeth. Extraoral imaging is mainly used to detect dental problems in the jaw and skull area. The most common one to be used is a panoramic radiograph [14]. Less common diagnostic measures are:

- Laser light-induced fluorescence
- Digital imaging fiber-optic transillumination
- Electrical conductance and impedance measurement

Bitewing X-ray

The bitewing radiograph is an image that depicts the crowns of upper and lower teeth on the left or right side, as seen in Figure 2.7. It gives a clear sight of the interproximal surfaces allowing good caries detection in this area. Interproximal caries are challenging to diagnose by the visual-tactile method; thus, using the bitewing X-ray can lead to an early diagnosis and a chance for the enamel to remineralize. Also, bitewing X-rays portray the alveolar crest, where the dentist may notice any bone thickness changes due to periodontal disease. Unlike the other intraoral method, it does not show the entire length of the teeth. This type of dental X-ray is the most commonly taken for preventive purposes [14].

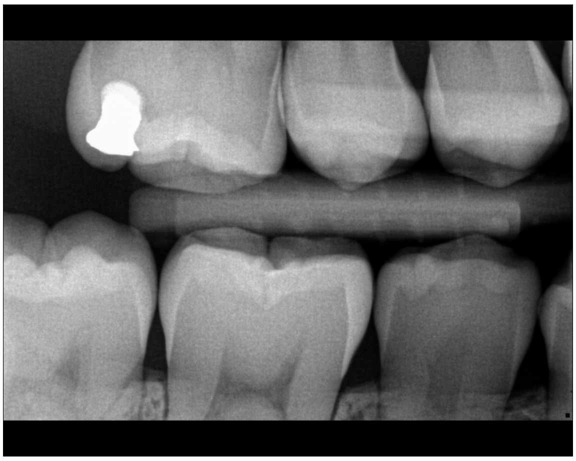


Figure 2.7: Image from bitewing X-ray with clearly visible restoration

Periapical X-ray

Periapical X-ray portrays the tooth from the crown to where the root attaches to the jaw; hence, the whole tooth length is visible. As illustrated in Figure 2.8, it only shows the upper or lower teeth in one part of the jaw. Periapical X-ray detects any abnormalities in the root and any periapical lesions.

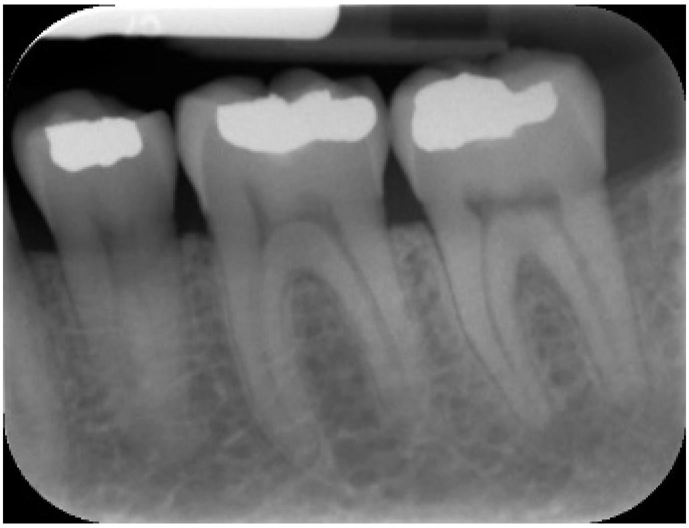


Figure 2.8: Image from periapical x-ray showing whole teeth. Source: www.dentist-manila.com

Panoramatic X-ray

This extraoral dental image shows the entire mouth area, including the upper and lower jaw and adjacent structures. It depicts the full dentition, including teeth that have not erupted yet. Impacted teeth, i.e. wisdom teeth as seen in Figure 2.9, can be identified as well. Panoramic X-ray is often used before major procedures or to diagnose jaw tumors, cysts, fractures, or sinusitis. Nevertheless, it is not usually taken to diagnose dental caries.

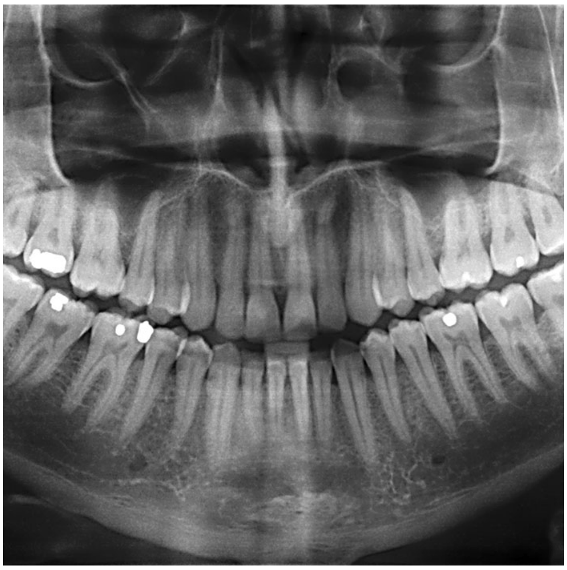


Figure 2.9: Image from panoramatic x-ray. Source: www.minthilldentistry.com

Chapter 3

Theoretical background

3.1 Computer vision tasks

This section provides a brief overview of standard computer vision tasks.

3.1.1 Classification

Let us say we have an image x. In a classification task, our goal is to assign one of n possible classes to the image:

$$\hat{y} = f_\theta(x) \tag{3.1}$$

where f is a mapping, sometimes called a model, and θ represents model parameter. If it holds that $\hat{y} = y$, where y is a true class of the image x, the classification is considered to be correct. It is possible to output $\mathbf{p} \in \mathbb{R}^n$ instead of $\hat{y}$, where $p_i \in \mathbf{p}$ is a probability of $i = y$, modeled by f_θ.

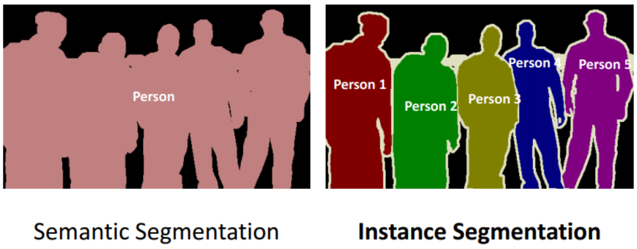
3.1.2 Semantic segmentation

For an input image $x \in \mathbb{R}^{n \times m}$, the goal is to output $\hat{\mathbf{y}} \in \mathbb{Z}^{n \times m}$, where $\hat{y}_i$ is the predicted class of pixel i in image x. Similarly to the classification problem, we can output matrix $\mathbf{P} \in \mathbb{Z}^{n \times m \times c}$, where $p_{i,c}$ is the probability of pixel i to belong to class c. A sample of semantic segmentation output can be seen in Figure 3.1.

3.1.3 Instance segmentation

Instance segmentation is similar to semantic segmentation, with the alteration saying that two objects of the same category would have different ground truth values. If we have $\mathcal{O}_\infty, \mathcal{O}_\in$, where $\mathcal{O}_i \subset x$ are pixels of object i in image x. Then

$$o_{1,i} \neq o_{2,j} \text{ for } o_{1,i} \in \mathcal{O}_1, o_{2,j} \in \mathcal{O}_2, \forall(i,j)$$



Semantic Segmentation

Instance Segmentation

Figure 3.1: Difference between semantic and instance segmentation. Source: www.analyticsvidhya.com

3.1.4 Object detection

In object detection, the goal is to locate and recognize objects of interest in image x. A rectangle and a category represent a ground truth object. Model predicts $\hat{\mathbf{Y}} \in \mathbb{R}^{n \times 6}$ values for each image. Each row of $\hat{\mathbf{Y}}$ consists of four numbers, which describe a rectangle, then the category of the object inside the rectangle, and a number in the range from 0 to 1 called the confidence. In literature, we can see the term score instead of confidence. Nevertheless, the meaning remains the same: Certainty of the network regarding the particular prediction described by the bounding box and category. Please note that the confidence of predictions does sum to one. In other words, we are not talking about probabilities since multiple detections per image can correspond to the ground truth.

3.2 Supervised and semi-supervised learning

Formulation of supervised learning

Supervised learning relies on on fully annotated data by an expert. Model in training uses the annotation as a feedback for its prediction and is able then change its parameters based on the results. We are given a dataset $\mathcal{D}_L$ which consists of M samples x (images) and their corresponding labels y (annotations, ground truth).

$$\mathcal{D}_L = \{x_i^l, y_i\}_{i=1}^M$$

Then a loss function $\mathcal{L}_{sup}$ is defined. The model in training then aims to minimize the result of this loss function over the dataset $\mathcal{D}_L$.

$$f_\theta^* = \text{argmin}_\theta \left(\sum_{i=1}^{M} \mathcal{L}_{sup}(f_\theta(x_i), y_i) \right) \tag{3.2}$$

Where f_θ^* is the optimal trained model with parameters θ. Because the dimension of θ is extremely large, it can not be minimized analytically, instead numerical methods, such as gradient descent, are in play. Basic algorithm for learning is then as follows:

1. Set initial values of parameters θ
2. Produce model's outputs $\hat{y}_i^l = f_\theta(x_i^l)$
3. Calculate loss of the model $L = \sum_{i=1}^{M} \mathcal{L}_{sup}(\hat{y}_i^l, y_i)$
4. Calculate gradient of the loss function in this point
5. Adjust parameters θ
6. If termination condition is met, end learning, otherwise go to 2

Formulation of semi-supervised learning

With semi-supervised learning we create a larger dataset $\mathcal{D}$ which consists of mentioned labeled data $\mathcal{D}_L$ and unlabeled data $\mathcal{D}_U = \{x_i^u\}_{i=1}^N$ of N samples.

$$\mathcal{D} = \{\mathcal{D}_L, \mathcal{D}_U\}$$

However now we do not have suitable loss function, because we do not have ground truth for all images to compare model's predictions with. Thus we add another part to a loss function.

$$\mathcal{L} = \mathcal{L}_{sup} + \lambda \mathcal{L}_{uns} \tag{3.3}$$

Where λ balances the impact of unsupervised loss function. How we define $\mathcal{L}_{uns}$ and how we change basic learning algorithm to make the best use of unlabeled data is what distinguishes different methods.

Assumptions

For semi-supervised learning to work properly, distribution of data should obey these general assumptions.
The smoothness assumption says that if two inputs x_1 and x_2, where $x_1 \neq x_2$, are close to each other in the input space (in the same cluster), then their outputs y_1 and y_2 should also be similar.

The low density assumption is directly derived from smoothness assumption. It says that the decision boundary lies in low density area (area with low number of samples). If it lied in high density area, the smoothness assumption would be violated.
The manifold assumption says that the smoothness assumption is valid even after some arbitrary transformation to lower dimension.

3.2.1 Methods

In this section we focus on different methods. They differ with their basic algorithm and unsupervised loss function.

Pseudo labels

The model in training plays the role of both a teacher and a student. After a round of learning on labeled data, the model produces segmentations for unlabeled data.

$$\hat{y}_i^u = f_\theta(x_i^u) \tag{3.4}$$

These output segmentations are called pseudo-labels and are then used as a ground truth for unlabeled data to compare with. We use the same loss functions for both unlabeled and labeled data. The algorithm then goes as follows:

1. Set initial values for parameters θ
2. Apply one round of learning on labeled data
3. Add unlabeled data into dataset
4. Produce pseudo-labels for unlabeled data
5. Produce predictions for the whole dataset
6. Calculate loss and its gradient
7. Adjust parameters θ
8. If termination condition is met, then end training, otherwise go to 4

Within this category methods mainly differ by different weights initialization and pseudo-labels noise handling. Noise could be created when pseudolabel is not correct by is treated as correct. If the noise is too strong it propagates through all generations of model and then reinforce itself. Thus it is important to handle that carefully. Yao *et al.* [48] propose confidence-aware supervision to improve pseudo labels quality. Li *et al.* [29] propose a self-ensembling strategy to build predictions via exponential moving average to avoid noisy and unstable pseudo-labels. Thompson *et al.* [43] came up with refining pseudo-labels with use of superpixels which should improve accuracy of masks of irregularly shaped targets.

Consistency learning

Consistency learning enforces variance of predictions with respect to image transformations to be as low as possible. To achieve that, model makes a prediction of an image $\hat{y}_i = f_\theta(x_i)$ and compares it to a prediction of the same image which is perturbated (small changes which are also used as an augmentation technique). We define set of transformations $\mathcal{T}$. For each image we recieve, let $(T_i^{input}, T_i^{output})$ be tuple of mappings, where $T = (t_1, t_2, ..., t_i), t_i \in \mathcal{T}$ is set of transformations from distribution $\mathcal{T}$:

$$T = (t_1, t_2, ..., t_i), t_i \in \mathcal{T}$$

T^{input} is transformation pipeline used for input and T^{output} is transformation pipeline used for output. They are created in such manner, that the following equation holds:

$$T^{output}(f(x)) = f(T^{input}(x)) \tag{3.5}$$

For each image x_i we create

$$T_i^{input}, T_i^{output}$$

and make predictions:

$$\begin{aligned}\hat{y}_{i1} &= f(T_i^{input}(x_i)),\\ \hat{y}_{i2} &= f(x_i)\end{aligned}$$

The the goal of this method is

$$\mathcal{L}\left(\hat{y}_{i1}, T_i^{output}(\hat{y}_{i2})\right) \to 0$$

Where $\mathcal{L}$ is some sort of similarity function (such as IoU or DSC).This requires using such T^{output}, that $T^{output-1}$ exists. Thus intensity transformations used in T^{input} are okay as they do not have output equivalent. However, one must be careful when using spatial transformations such as translation or cropping. As Bortsova et al. [11] pointed out, it is necessary for T^{output} to have inverse function, otherwise the backpropagation works with wrong weights. In the end it would lead into moving the weights in favour of one of the predictions, even though this is unsupervised loss and the prediction could be wrong. Bortsova states that if:

$$\tilde{y}_i = T^{output}(\hat{y}_i) \tag{3.6}$$

then

$$\frac{\partial \mathcal{L}(\hat{y}_{i1}, \hat{y}_{i2})}{\partial \hat{y}_{i2}} = T_i^{output-1}\left(\frac{\partial \mathcal{L}(\hat{y}_{i1}, \tilde{y}_{i2})}{\partial \tilde{y}_{i2}}\right) \tag{3.7}$$

Bortsova et al. [11] suggests to use it IoU loss as supervised and unsupervised loss function:

$$\mathcal{L}_{sup}(\hat{y}, y) = \mathcal{L}_{uns}(\hat{y}, y) = 1 - \frac{1}{C}\sum_{c=1}^{C} \frac{\sum_{i=1}^{N} \hat{y}_c^{(i)} y_c^{(i)}}{\sum_{i=1}^{N} y_c^{(i)} + (1 - y_c^{(i)})\hat{y}_c^{(i)}} \tag{3.8}$$

Where C is number of classes excluding background. Laradji et al. [27] introduce a unsupervised loss function which computes sum of absolute differences. This loss function helps them learn achieve surprisingly high results using point loss as supervised loss. They also use geometric transformation such as flip and rotation.

$$\mathcal{L}_{uns}(x_i) = \sum_{i=1}^{N} |f(T_i^{input}(x_i)) - T_i^{output}(f(x_i))| \tag{3.9}$$

Among other loss function are Kullback-Leibler divergence ($\mathbf{D}_{KL}$), mean squared error (MSE) or Jensen-Shannon divergence ($\mathbf{D}_{JS}$). There are many perturbations to be used. Among the most common belongs Gaussian blur, Gaussian noise, adjusting contrast or brightness, rotation and fliping. Xu *et al.* [47] propose using shadow augmentation which simulates low quality images.

Co-training

Blum and Mitchell in [10] build on the idea of pseudo-labels. However there is not one model that creates pseudo-labels for itself, there are two models. This work relies on assumption that there are (at least) two different views on data that are independent on each other and a model is able to learn from only one view. Two models are then in play, each training only on one view. After first round of training each of them will produce pseudo-labels for unlabeled data. Those labels with higher confidence are selected. The architecture introduced by Yao et al. [48] mentioned in section 3.2.1 should also be mentioned here as it works with confidence of predictions and two models.

Entropy minimization

This method builds on the low density assumption. The decision boundary should be in low density area. Therefore we adjust the loss function by adding entropy of possible decisions on given input. If the model is not sure and produces similar probabilities for several classes than entropy in this point is high. As model gets more confident in decision the entropy gets lower. Thus it is convenient to place the decision boundary in low density area. Without unlabeled data, we might have too little data to create a satisfying and properly working boundary. Other problem occurs when model is overconfident and produces confident prediction (which have low entropy) even in high density areas. Therefore this methods does not have to work in all cases [17]. However added to another technique it could boost its performance [34].

Adversarial learning

Adversarial learning consists of two neural networks, segmentation and evaluation, fighting against each other. Segmentation network (SN) creates a prediction which is then evaluated by evaluation network (EN) with respect to ground truth annotation. Specific approach was described by Zhang et al.[51]. They trained segmentation model to return decent segmentation maps. Then trained a classifier to distinguish between predictions on unlabeled data and predictions on training data (which should be quite similar to ground truth). Then they combined both models. SN creates segmentation map, EN evaluates it and backpropagation runs trough both models. They formulated the whole loss function as:

$$\begin{aligned}\mathcal{L} = &\sum_{m=1}^{M} \mathcal{L}_{CE}(SN(X_m), Y_m) + \\ &+ \lambda \left[\sum_{m=1}^{M} \mathcal{L}_{CE}(EN(SN(X_m), X_m), 1) + \sum_{n=1}^{N} \mathcal{L}_{CE}(EN(SN(X_n), X_n), 0) \right]\end{aligned} \tag{3.10}$$

Where $\mathcal{L}_{\mathcal{CE}}$ is a Cross Entropy loss, X_m is an image from labeled dataset, Y_m is corresponding ground truth, and X_n is an image from unlabeled dataset.

Informed learning (with prior knowledge)

Knowledge priors are general information about the task that could help the model. Medical images have many anatomical priors which could be used to our benefit. Among these priors belong organ's shape, color (intensity), relative position to other organs. These knowledge priors could be incorporated into model as adjusting the loss function. With semi-supervised learning this method is usually seen as pretraining the model for some proxy task, which helps the model to learn general features of images. Huang *et al.* [20] add a reconstruction pretraining to improve parameters initializing. Zheng *et al.* [53] proposed using probability atlas based on labeled images. Huang *et al.* [19] builds on that approach and reuse it for semi-supervised learning, where probability atlas is used to give segmentation masks pixel wise confidence to select reliable pixels.

3.3 Metrics

A metric helps in evaluating the performance of any designed model. The metrics provide the accuracy of the designed model. The popular metrics employed for assessing effectiveness of any designed segmentation algorithm are represented in terms of the following [31]: In context of binary classification or binary segmentation we have a collection of data and want to retrieve those, which are somehow relevant. The collection could be pixels in case of image segmentation or whole images in case of image classification.

- True positive (TP) represents data that are relevant and were retrieved.
- True negative (TN) represents data that are not relevant and were not retrieved.
- False positive (FP) represents data that are not relevant but were retrieved.
- False negative (FN) represents data that are relevant but were not retrieved.

Precision (P) says how many of retrieved instances are relevant.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.11}$$

Recall (R) also known as Sensitivity says how many of relevant instances were retrieved:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.12}$$

Specificity (S) says how many of not relevant instances were correctly ignored

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{3.13}$$

Accuracy (A) tells us how many instances were correctly classified

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3.14}$$

F1 score tells about models accuracy as represented in the following equation. It is defined as the harmonic average of the precision and recall values:

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.15}$$

Intersection over union (IoU) is a metric commonly used for checking the performance of image segmentation algorithm. It is the amount of intersecting area between the predicted image segment and the ground truth mask, divided by the total area of union between the predicted segment mask and the ground truth mask:

$$\text{IoU} = \frac{TP}{TP + FN + FP} \tag{3.16}$$

Dice similarity coefficient (DSC) is similar metric to IoU, with a significant difference and that is that this function is differentiable [9]:

$$\text{DSC} = \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \tag{3.17}$$

Given that in this specific task not all images contain restoration, DSC would not be good evaluation metric for such images as those could only have DSC 1 or 0. Thus we will also use metric **True Posistive DSC (TP DSC)** which will compute DSC only over those images which truly contain restorations. This metric could better represent segmentation ability of the model. Using the fact that DSC can not exceed 1 and that it is differentiable function, there is loss function based upon this metric simply computed as:

$$\text{Dice Loss} = 1 - \text{DSC} \tag{3.18}$$

Cross Entropy loss (CE) is widely used loss function which take advantage of logarithm and can shoot the loss function into great heights if the answer is wrong. The binary cross entropy loss is computed as [35]:

$$L_{CE}(\hat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{N} \sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) * \log(1 - \hat{y}_i) \tag{3.19}$$

where $\mathbf{y}$ refers to ground truth label and $\hat{\mathbf{y}}$ refers to model prediction.

3.4 Deep learning architecturs

3.4.1 SegNet

SegNet introduced semantic pixel-wise image labeling [7]. It comprised a stack of encoders followed by a corresponding decoder stack, which feeds into a softmax classification layer. The decoders help map low-resolution feature maps at the encoder stack's output to full-size feature maps identical to the input size. During downsampling using maxpool layer, the network saves indices of pooled values and forwards them through a skip connection to the decoder. It can assign correct position to values during upsampling. It addressed an essential drawback of recent deep learning approaches, which have adopted networks designed for object categorization for pixel-wise labeling.

3.4.2 U-Net

The U-Net architecture is the most prominent medical image segmentation model applied to various medical problems, published in 2015. It is a common belief that successful architecture training needs a massive amount of data [39]. U-Net presents a strategy that strongly depends on the data augmentation technique to use limited available data more effectively. It is quite similar to mentioned SegNet except for two significant differences. It passes maxpool output through skip connections instead of maxpool indices and it doubles the number of features every time it downsample the image. The U-Net architecture is depicted in Figure 3.2

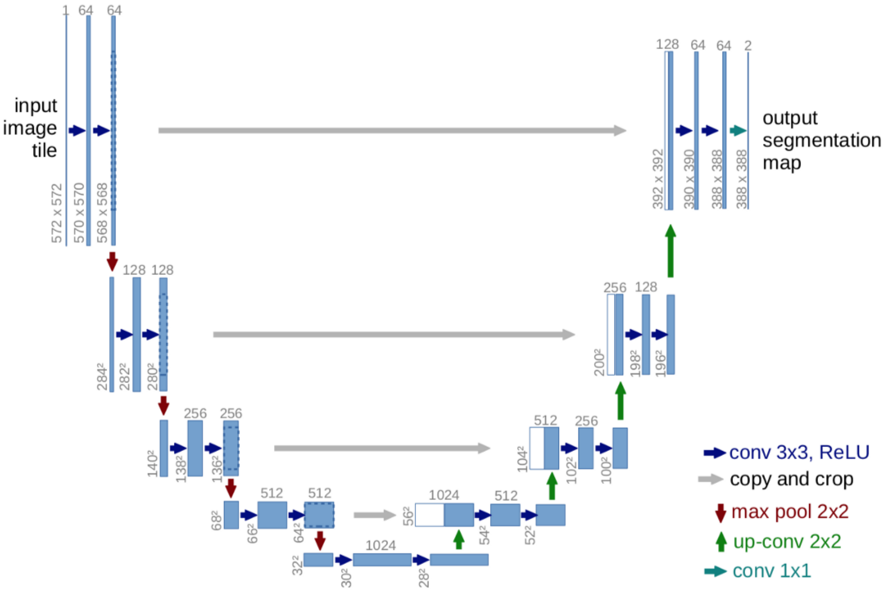


Figure 3.2: Architecture of neural network proposed by [39]

3.4.3 U-Net++

In 2018, U-Net++ was proposed by Zhou et al [54]., USA to overcome U-Net's limitation of utilizing same-scale feature maps alone. The architecture used the concept of Dense-Block to improve original U-Net performance as shown in Fig. 15. Unlike the foundation model (U-Net), it included convolutions and dense skip connections on skip-pathway to fill the gap between feature maps across modules and to improve gradient flow. The proposed architecture is evaluated in a multi-modal environment by considering four different medical image repositories, including; cell nuclei, colon polyp, liver, and lung nodule. During testing it outperformed U-Net and wide U-Net. The architecture is shown below in Figure 3.3

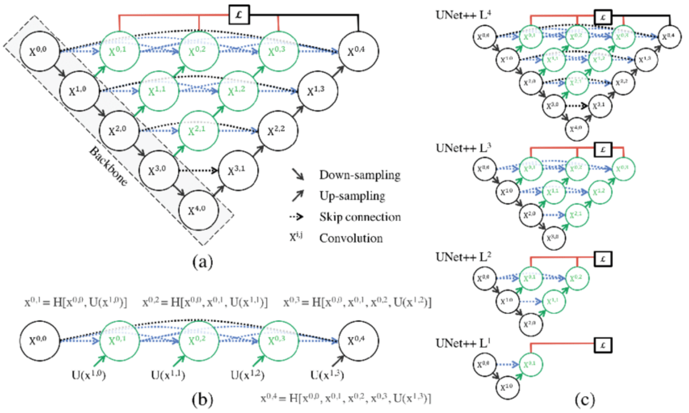


Figure 3.3: U-net++ is supposed to be improved version with more dense skip connections

3.4.4 YOLO

YOLO (You Only Look Once) is an object detection neural network which is designed to be fast and able to detect objects in real time in video. Similar to the mentioned segmentation networks, it has a encoder part, which uses double strided convolution instead of maxpool layer, to downsample the input image to a grid with few squares containing locations of bounding boxes and probabilities for each class.

Chapter 4

Related work

- L. Kunt's masters thesis [26] focused mainly on dental caries detection but managed to do some progress in this field too. He managed to achieve DSC of 0.76 and IoU 0.676. This project aims to follow Kunt's work and improve used models. Kunt's work will be used as the main benchmark.

- Baydar et al. [8] conducted similar study as we are about to. They explored the usage of U-Net regarding bitewing images. However, they used it as a detector machine. When segmentation achieved more than 0.50 in IoU, it was considered to be true positive, otherwise it was false positive. They also used more than just one annotation type. They distinguished among some type of restorations and among teeth, teeth root canals and restorations. Even though they used U-Net, which is network designed for image segmentation, they used it as some kind of detection network.

- Mao et al. [32] classified dental segmentations in previously extracted image patches with unilateral teeth

- Lee et al. [28] did not focus directly on the segmentation of restorations, yet it was one of the classes segmented out by their U-net architecture. There are no metrics available regarding the algorithm's performance on dental restorations[26].

- Abdalla-Aslan et al. [6] used methods of classical computer vision to segment out restorations in panoramic images. Their pipeline consisted of: Adaptive gaussian thresholding, morphological operations, and deleting regions in peripheral areas of the image. The final algorithm had the precision and sensitivity of 0.33 and 0.946, respectively. After successful detection, the restoration was classified as: dental implant, crown, amalgam filing, etc.

- Yeshua et al. [49] were solving the same problem as Abdalla-Aslan. Even the approach was more-less the same, except theirs achieved a precision of 0.568. They classified detected areas similarly to Abdalla-Aslan, having an extra category for false detections. After the removal of false detections, the precision was boosted to 0.98.

Chapter 5

Dataset

All data have been provided and annotated by MDDr. Tichý and MDDr. V. Nagyová. The images came from at least four different stomatologic clinics as we received different sizes of x-rays (as we can see in Figure 5.2). All images have been padded to same rectangular size 847×1068 pixels. All annotations were done in CVAT [1] which allowed them to annotate restorations using polygons for best accuracy. All restorations with exceptions of braces and retainers have been annotated.

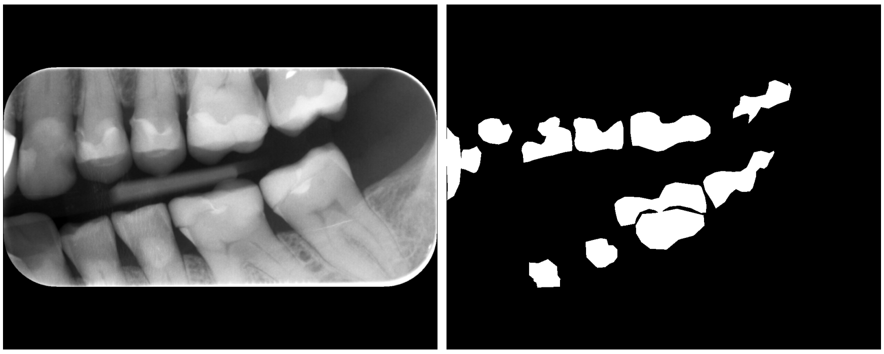
We split our work in two stages because we received another set of annotated data later in this work. We will explicitly mark which dataset was used for particular experiments in later sections. We can see one data sample with corresponding segmentation mask in Figure 5.1. All tests were conducted after normalizing the images with mean and standard deviation computed over training data and without the black paddings.

5.1 Stage 1

In stage one we followed work of Kunt [26] and used the exact same dataset. It consists of 521 images. We split the data to training and validation in ratio 90:10 knowing that more data are to come before an end of this work, which we could use as testing data. Also we wanted to split the data in the exact same way as Kunt in order to be able to compare our results. To compare how the dataset changes after adding new images we compare the following histograms: How many restorations are in one image and how much space takes up one restoration (in % of image size) and how much space take up all restorations in an image. These are to be seen in Figure 5.3. Overall there are 2336 restorations marked in 521 images. The type of restorations was not marked in this task but they cover all kinds of different restorations covered in chapter 2.

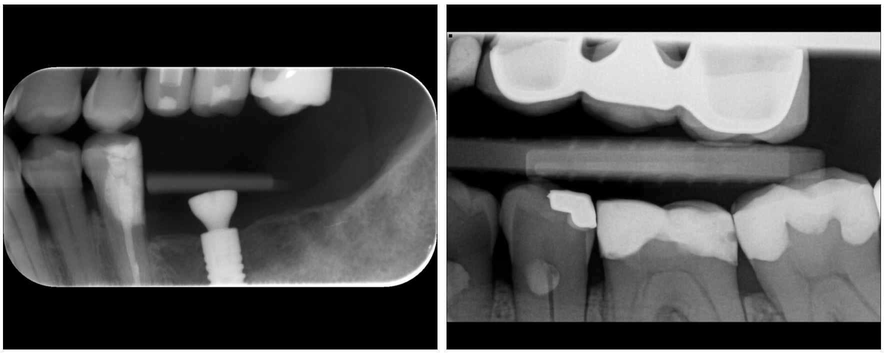
5.2 Stage 2

MDDr. Nagyová was able to annotate another 447 x-ray images with another 1654 restorations, leaving us with 969 images in total. We divided those data in training, validation and testing in ratio 80:10:10. Histograms describing this enlarged dataset are in Figure 5.4.
During this work we also retrieved annotations from students of dentistry from Charles University. Student A studies at Faculty of Medicine in Plzeň (second year) and Student B studies at First Faculty of Medicine in Prague (third year).
These students were kind enough to try to annotate one set of validation data consisting of 98 images. Student B made time to annotate testing data which consists of another 99 images.



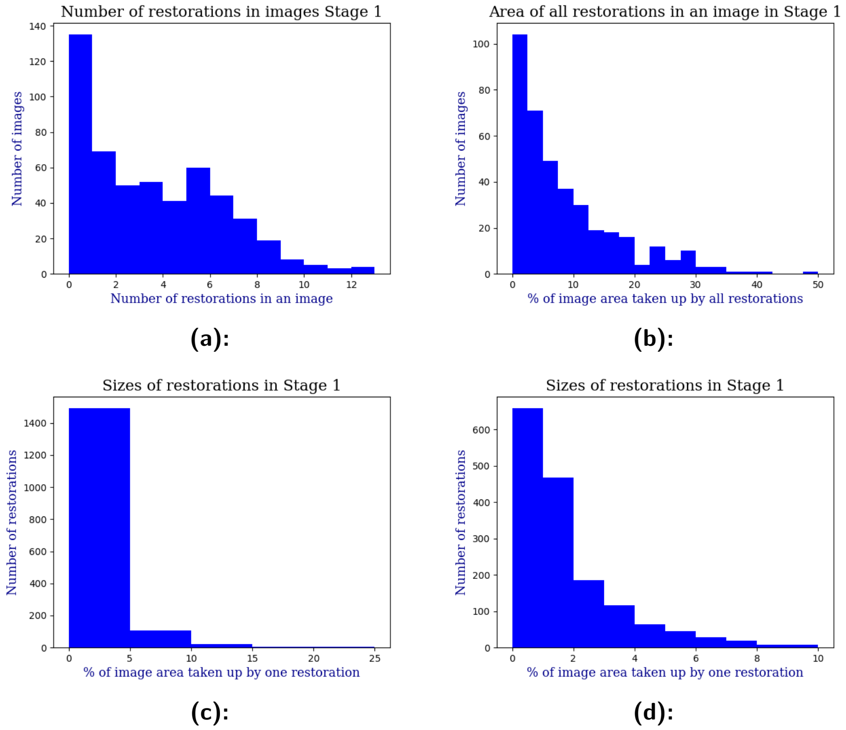
(a): (b):

Figure 5.1: Sample x-ray image and corresponding segmentation mask. We can see metalic (bright white) and composite (lower contrast compared to tooth) direct restoration



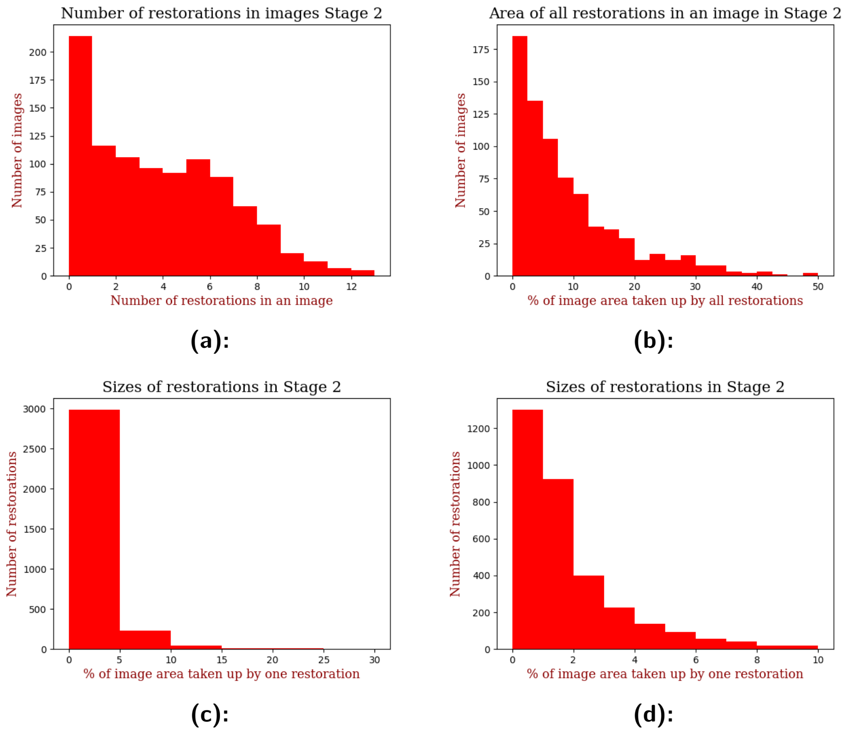
(a): (b):

Figure 5.2: Sample images containing dental implant and dental bridge. We can even notice white artifact at the top of the right image. Such phenomena originates from lower precision when taking the image.



(a): **(b):** **(c):** **(d):**

Figure 5.3:



(a): **(b):** **(c):** **(d):**

Figure 5.4:

Chapter 6

Methods

Until said otherwise, the following experiments used satge 1 dataset.

6.1 Experiments with backbone model architecture

Following Kunt's work [26] we put most of our focus into using U-Net. Based on his work, where he optimized loss function, LR scheduler and optimizer we used the same hyperparameters as he did. We implemented U-Net with PyTorch open-source deep learning library and used th following settings:

- Cosine Annealing Lr scheduler with half-period = 100
- AdamW optimizer with default settings
- Combination of Dice Loss and Cross Entropy loss
- We set minimum number of epochs to be 50 and end the training if there was no improvement in Dice score over 10 epochs
- Learning rate was arbitrary set to 1e-5

The basic architecture shown in Figure 3.2 has depth of 4 levels. To outperform Kunt's work, which used this basic architechture, we tested if making the U-Net deeper will make a difference. To further improve performance we also tested U-Net++ with the same settings. Results are depicted in Table 6.1 and Figure 6.1

Transposed convolution or Interpolation

Mentioned architectures heavily rely on upsampling layers to return the tensor to the original size. We can either use **Transposed convolution** (also called

Model	Depth	Upsampling layer	Batch number	DSC(median)
U-Net	4	Transposed convolution	3	0.6577
U-Net	4	Interpolation	3	0.5398
U-Net	5	Transposed convolution	2	**0.8705**
U-Net	5	Interpolation	2	0.7977
U-Net	6	Interpolation	1	0.848
U-Net++	5	Interpolation	1	0.6577
U-Net++	6	Interpolation	1	0.7739
U-Net++	7	Interpolation	1	0.798

Table 6.1: Displaying the influence of depth, upsampling layer and model architecture on DSC. All of these tests were conducted with no scaling of the images.

deconvolution, explained here [2]) which is has learnable parameters, or we can decide that interpolation layer would suffice. Transposed convolution contribute to stacking gradient and thus needs larger memory. Interpolation layer does not have that problem, However, the model can not learn on its way up. Thus we tested both these layers, if possible, to see if one is superior or it does not matter.

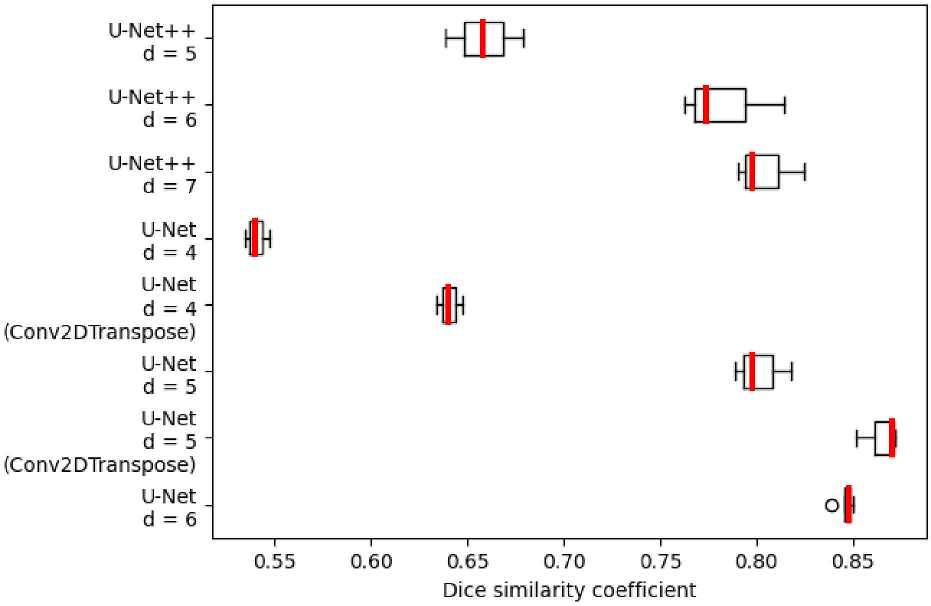


Figure 6.1: Displaying the influence of depth, upsampling layer and model architecture. We tried to use transposed convolution with U-Net++ and deeper U-Net, but it did not fit into GPU's memory even if the image was scaled with factor 0.1. We did not try to scale it down even more, because as we can see from charts 5.3 and 5.4 most of the restorations are quite small and we would probably lose crucial information regarding these restorations.

Given these results we will continue to use U-Net with transposed convolu-

Transform	Probability	Parameters	DSC (Validation set)	DSC (Test set)
None	-	-	0.8705	0.7557
Vertical flip	0.5	-	0.8983	0.7586
Horizontal flip	0.5	-	0.8711	0.7546
Gaussian blur	0.3	$kernelsize \in (7, 31)$	0.8548	0.7373
Gamma correction	0.3	$\gamma \in (0.6, 1.4)$	0.8659	0.7457
Translation	0.4	translation limit = 20% of image size	0.8661	0.7692
Rotation	0.4	rotation limit = 20°	0.8892	0.7484
Scaling (Resized Crop)	0.4	$scale \in (0.5, 1.0)$	0.8914	0.7677
Gaussian noise	0.3	$\sigma \in (0.01, 0.1)$	0.863	0.7514
Elastic deformation	0.2	$\alpha \in (50, 150)$	0.8795	0.7318
Everything			0.8809	0.7339
Selection			0.904	0.7844

Table 6.2:

tion with depth 5 as a backbone for all following experiments unless it is said otherwise.

6.2 Finding best working augmentation

Ronneberger et al. [39] mention that U-Net heavily rely on data augmentation. Kunt [26] used several transformations to help network to better generalize. However, there was no specific study conducted whether all of them are useful. Thus we decided to test several image transformations independently and then all together to see if any transformations could be pulling the performance down. We tested the following transformations with probability p:

- Horizontal flip, p = 0.5
- Vertical flip, p = 0.5
- Rotation with limit 20°, p = 0.4
- Translation with limit 20% of image size, p = 0.4
- Resized crop (Scaling) with scale factor ranging from 0.5 to 1, p = 0.4
- Gaussian blur with kernel size from 7 to 31, p = 0.3
- Gamma correction with γ ranging from 0.6 to 1.4, p = 0.3
- Gaussian noise with σ ranging from 0.01 to 0.1, p = 0.3
- Elastic deformation with α ranging from 50 to 150, p = 0.2

At first we used each of those listed at its own and compared results to controlling runs without any data augmentation. Each setting was trained 10 times to be sure good or bad result was not just a coincidence. Results of those runs can be seen in Table 6.2 and Figure 6.2.

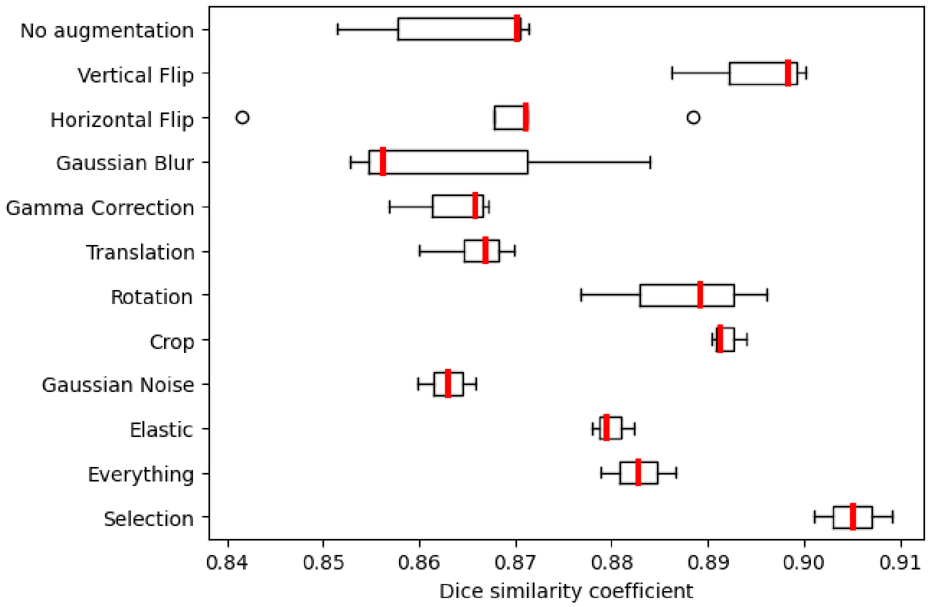


Figure 6.2: Results of models with only some augmentation enabled

Based on those results we compared two settings. One with all mentioned transformation put to use and one where only the following selection of transformations was used:

- Vertical flip
- Horizontal flip
- Rotation
- Translation
- Crop
- Elastic transform

Even though translation and horizontal flip did not exactly improve performance, there is no logical reason for them to worsen the result, thus we included them. We can see results again in the same table and figure. It is important to note that the beginning of the axis is not in the zero, thus the difference is not as big as might seem. However given the number of runs we conclude that it is not by mistake and that we find an improvement to how to augment data with this specific task. At the time of this experiment we could use about 90 newly annotated images. Thus we ran the same learned models on those to test our results. We can see that in Figure 6.3 and Table 6.2 in column Test set. Even though the overall performance of models has significantly dropped, we can see that the selected transformations still provided an improvement. From now on we continue to experiment with these transformation in place.

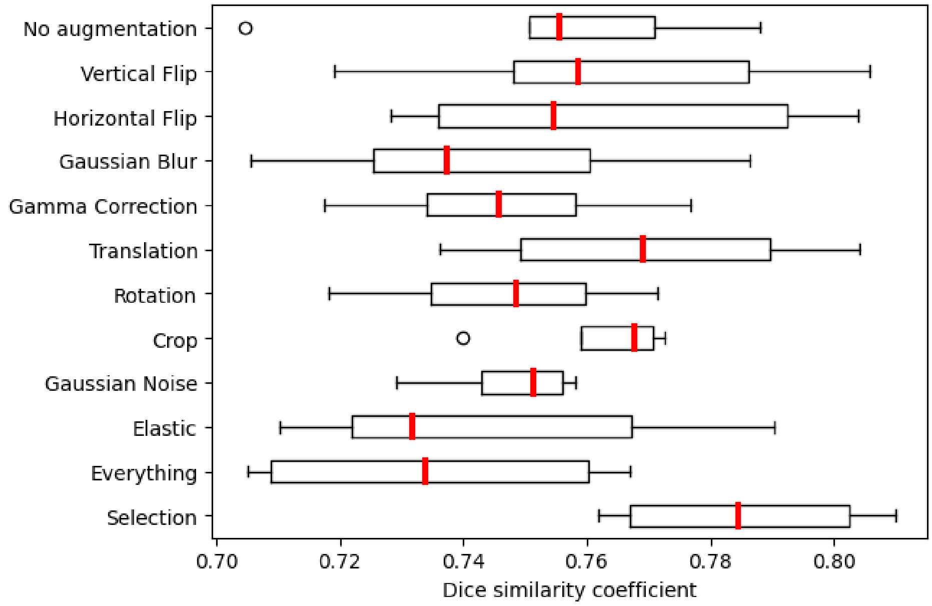


Figure 6.3: Results of the learned model on newly annotated testing data

6.3 Filtering out images without restorations

From this section on we used stage 2 dataset. Not everyone had to go to his dentist to get his teeth repaired. Thus not all x-ray images contain restorations. Actually, as we could see in Figures 5.3 and 5.4, those images make about one quarter of all images. Combined with the reality of false positive segmentation of model, which finds segmentation at places without any obvious reason, we deduced it could be helpful to come with some sort of discriminator network (DN) which could filter such negative images before they get to segmentation network. For this purpose we tried several deep learning models.

All the following method were trained on dataset consisting of nearly 800 images and validated on nearly 100 images of dental bitewing x-ray.

Baseline

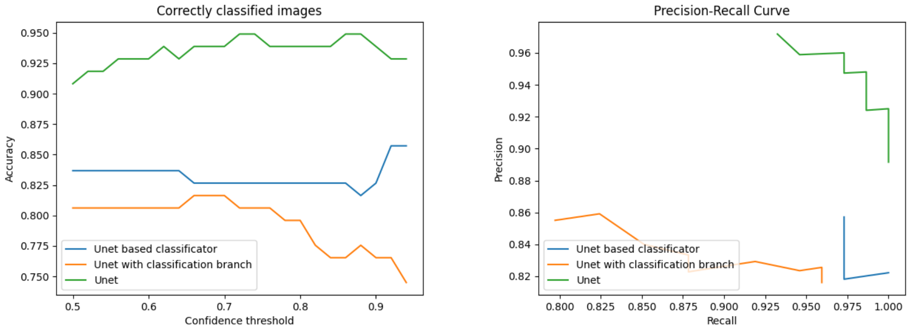
We use U-Net trained as segmentation model as the classification baseline. Classification process of an image goes as follows: We let the model predict probability segmentation map. If any pixels hold probability value higher than set threshold, let the class of an image be Positive. Negative otherwise. The U-Net model was trained for 50 epochs with loss function being combination of mean Crossentropy and Dice loss. We loaded so far best performing U-Net model (0.81 in DSC, 0.85 in TP DSC) and tried to find the best confidence threshold (going from 50 to 96). Using U-Net purely as segmentation model, where confidence threshold is set to 0.5, we receive following results:

U-Net with classification branch

To take advantage of decent U-Net results, we created new model that we trained alongside with U-Net. For purpose of this paper we will call it classification branch. It shares encoder with U-Net (shared weights). After that it continues with double convolutions with the same number of channels as U-Net, but does not have to upsample, as the result does not have to describe whole image. After the same number of double convolutions, it has one linear layer, which outputs two dimensional vector.
We trained this model with loss function being a combination of mean cross entropy and Dice loss for segmentation output and cross entropy for classification output.

U-Net based classificator

Finally we tried to use only the classification branch. We used weights from trained U-Net model described in section 6.3 and fine tuned the net only on classification cross entropy loss for 100 epochs.

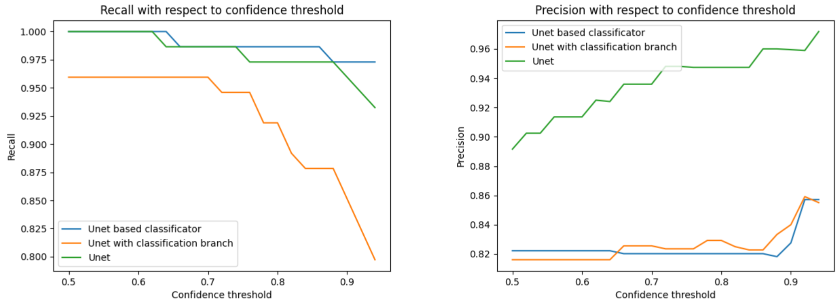


(a): **(b):**

Figure 6.4: Results of U-Net based classification techniques

YOLO

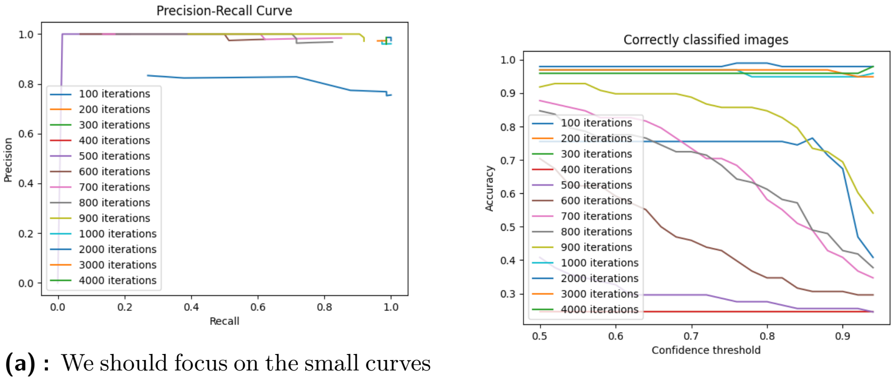
As said YOLO is an object detection model. It works with bounding boxes. We used Yolov3, which has more layers compared to original YOLO and creates outputs from different parts of model, meaning we end up with several sets of predictions. Each prediction contains center coordinates of bounding box, its height and width, and confidence value going from 0 to 1 for each



(a): **(b):**

Figure 6.5: Results of U-Net based classification techniques

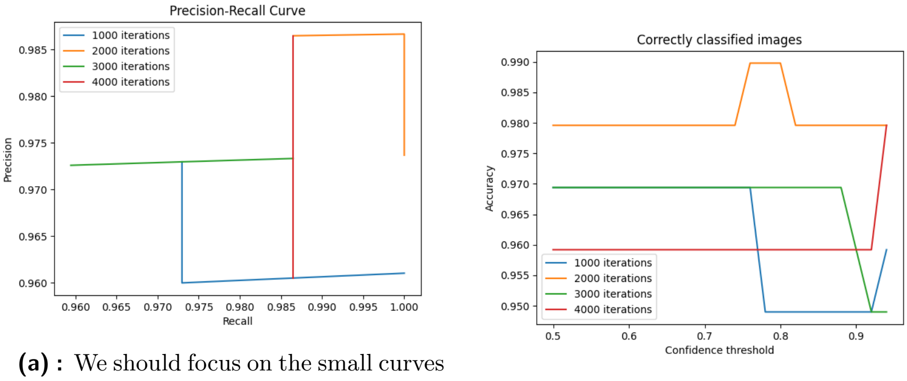
class we are predicting. We used yolo-voc.cfg with batch 64, subdivision 16, width and height 416 and 1 channel. We let the YOLO train and saved its weight every 100 epochs until 1000 epochs and then every 1000 epochs up to 4000. We then evaluated each weights on validation data with confidence threshold for bounding box to be valid going from 50 to 96 with step of 2. We can see the results in Figure 6.6a, which depicts how recall decreases while precision increases with increasing confidence threshold, and in Figure 6.6b which shows us how many images were classified correctly depending on confidence threshold. From close look at Figure 6.7 we can see that after 2000 epochs, the model is at the peak performance. With confidence threshold set to value between 0.76 and 0.8 it filtered out only negative images.



(a) : We should focus on the small curves in the right upper corner where are the top performing models

(b) : Accuracy showing how many images were classified correctly

Figure 6.6: Results of YOLO image classification based on the number of epochs it was training for



(a) : We should focus on the small curves in the right upper corner where are the top performing models

(b) : Accuracy showing how many images were classified correctly

Figure 6.7: Results of YOLO image classification with focus on the best performing models. We can see that after while the model is overfiting the data.

Model	**Threshold**	**Precision**	**Recall**	**Accuracy**
Baseline	0.5	0.912	**1.00**	0.928
U-Net	0.76	0.938	0.985	0.939
U-Net with classification	0.70	0.827	0.96	0.819
U-Net based classificator	0.92	0.86	0. 973	0.857
Yolo	0.76	**0.985**	**1.00**	**0.99**

Table 6.3: Models and their results if we set such a threshold to achieve the highest accuracy

Results

As Figure 6.5a and 6.4a shows, at the beginning the model U-Net had perfect recall as the threshold was just above 0.5. However precision depicted in Figure 6.5b was at 90%, which is not satisfying as the point of this work is to get as high precision as possible. With increasing threshold we quickly lose perfect recall, which is crucial, because filtering out positive images should be avoided at all costs. Even with threshold set to a high number, the precision does not reach 100%.
It is clear that using yolo's detection mechanism outperforms suggested classification techniques. It is interesting that even if we transformed the loss function to improve classification result, the performance did not improve. In addition, segmentation ability of U-Net with classification branch has significantly decreased (DSC to 0.5891 and TPDSC to 0.7293).
Second right after YOLO model is normal U-Net and it does not matter if we maximize accuracy or maximize precision subjected to maximum possible recall.
Results for maximum precision are displayed, because those are not relevant as maintaining highest possible recall is crucial in this task.

Based on these results, we can further only focus on improving segmentation on TP images, as yolo should effieciently filter TN images out. Thus we will from now on use metric TP DSC as defined in section 3.3. This will better ilustrate model's ability to properly segment restorations, because it will not be noised with DSC from images without restorations.

6.4 Hyperparameters experiments

Learning rate

To further optimize settings for supervised learning, we tried setting starting learning rate to the following values:

$$10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$$

The settings of other hyperparameters remains the same. The cosine annealing value was set 100, which makes the learning rate to drop by cca half of an order in 50 epochs. We can see the process of learning during 50 epochs in the Figure 6.8

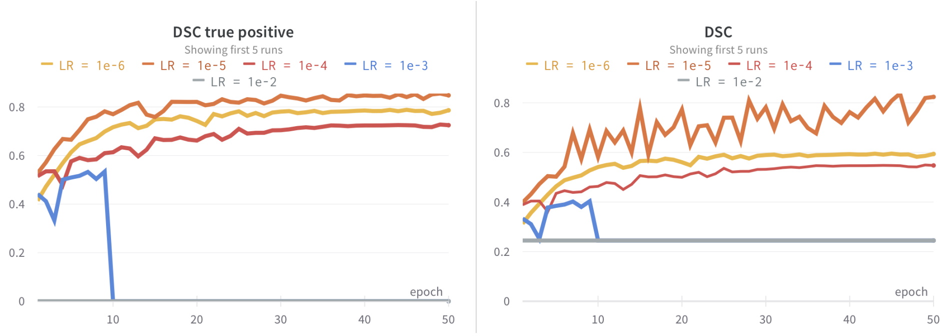
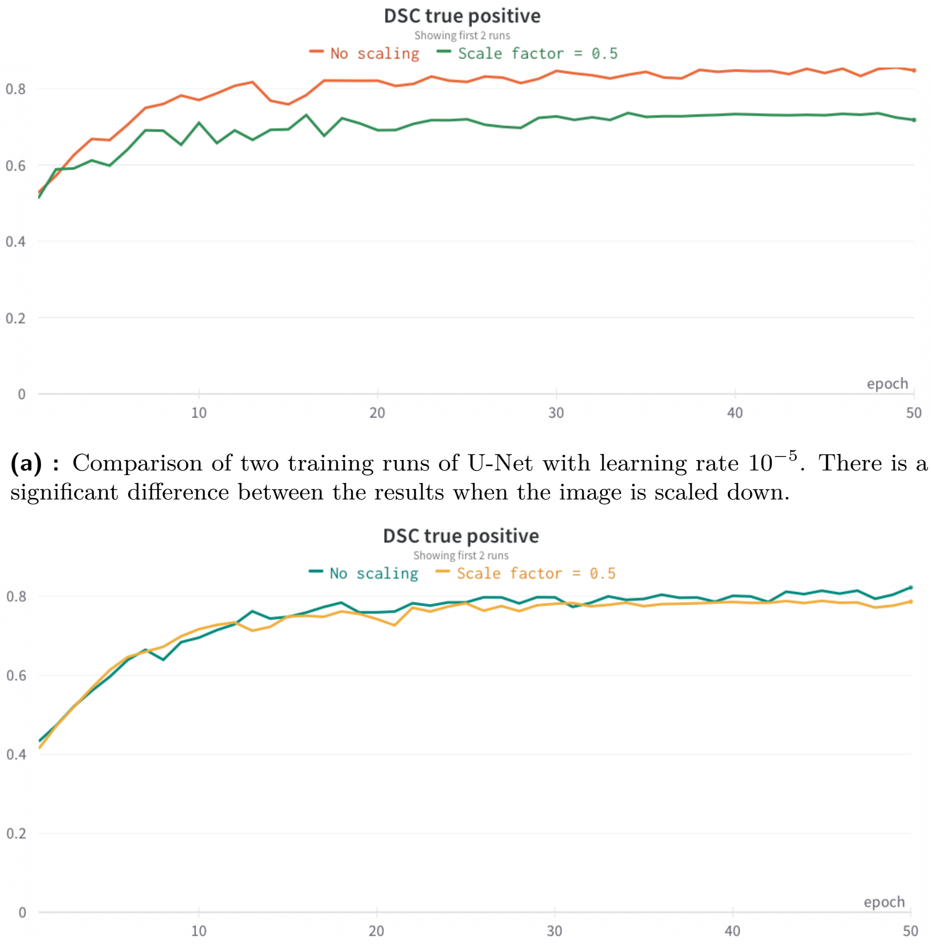


Figure 6.8: Process of learning with different learning rate settings. 1e-5 turned out to be optimal. It could be optimized even more with more specific search, but it would not probably bring such significant improvement.

More epochs would probably lead to better results for higher values of learning rate, considered that the learning rate would decrease with use of learning rate scheduler. However it is better to start with optimal learning rate, which we showed to be 10^{-5}.

To speed up training and to utilize higher batch number we tried to downsample images before passing them to a model. We used torchvision resize with scale factor 0.5 (decreases each side by half making the image one quarter of its former resolution). We can see how it affected performance in Figure 6.9.



(a) : Comparison of two training runs of U-Net with learning rate 10^{-5}. There is a significant difference between the results when the image is scaled down.

(b) : Comparison of two training runs of U-Net with learning rate 10^{-6}.

Figure 6.9: Difference between training with downscaled images and images in full resolution. Compared when training U-Net.

We can see that lowering the resolution resulted in significant and not random (we tested this with five runs for each setting, only one run of each is displayed for better visibility) decrease in performance when scaled down. The reason for this to happen could once again found in Figures 5.3 and 5.4. We can see that lot of restorations are below 1% of images size. Bigger downscaling can erase them completely which then affects model's ability to find such small dental correction.

6.5 Semi-supervised learning

For the purpose of segmentation of dental bitewing x-ray images, we used several semi-supervised methods of those described in section 3.2. As a supervised baseline we used model trained with the settings described higher in section 6.4 which achieved following results:

TP DSC	DSC
0.8554	0.8015

Table 6.4: DSC and True positive DSC of best performing supervised U-Net model

Pseudolabels

First of all we tried to use pseudolabels which only uses the algorithm described in section 3.2.1. We loaded so far best trained model described higher as baseline. We let it train for 50 epochs while every 10 epochs the model regenerated pseudolabels. As it is possible to see in Figure 6.10, the model converged quite quickly. Thus we did not continue in adding more epochs. We did not change the original loss function and continued with $\mathcal{L} = \text{CrossEntropy} + \text{Dice Loss}$. But we achieved growth in DSC which could further help with identifying images without any restoration.

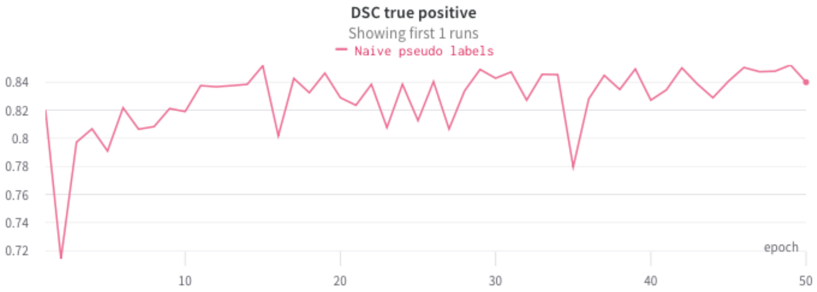


Figure 6.10: Evolution of TP DSC in time with naive pseudo labels learning

Consistency training

We defined transformation pipelines as follows:

$$T_i^{output}(x_i) = T_i^{input}(x_i) = t_3(p_3) \circ t_2(p_2) \circ t_1(p_1)(x_i)$$

where t_i as applied with probability p_i and inspired by [27] we set

$$p_1 = p_2 = p_3 = 0.8$$

$$t_1 = \text{HFlip}$$

$$t_2 = \text{VFlip}$$

$$t_3 = \text{Rotation}$$

Angle of rotation is chosen uniformly with maximum 20 degrees.
We tested the following unsupervised loss functions:

Probability difference

Inspired in [27] we used the following loss function:

$$\mathcal{L}_{uns}(x_i) = \frac{\lambda}{N}\sum_{i=1}^{N} |f(T_i^{input}(x_i)) - T_i^{output}(f(x_i))| \tag{6.1}$$

We set $\lambda = 5$ to get numbers of similar value as Dice loss

Soft IoU

I did not find a way to implement IoU to be differentiable and thus it does not have a backward pass. Therefore I implemented soft IoU. After normalizing logits from model with softmax, I exponentiate them, to increase the difference and then ran through softmax function again.

$$\hat{y}_i = \text{Softmax}(f_\theta(x_i))$$

$$\tilde{y}_i = \text{Softmax}(\hat{y}_i^{\gamma})$$

I arbitrary chose $\gamma = 6$. Then such "thresholded" segmentation maps went through IoU lost function:

$$\mathcal{L}_{uns}(\hat{y}, y) = 1 - \frac{1}{C}\sum_{c=1}^{C} \frac{\sum_{i=1}^{N} \hat{y}_c^{(i)} y_c^{(i)}}{\sum_{i=1}^{N} y_c^{(i)} + (1 - y_c^{(i)})\hat{y}_c^{(i)}} \tag{6.2}$$

Dice loss

$$\mathcal{L}_{uns}(\hat{y}, y) = 1 - \frac{1}{C}\sum_{c=1}^{C} \frac{\sum_{i=1}^{N} \hat{y}_c^{(i)} y_c^{(i)}}{\sum_{i=1}^{N} y_c^{(i)} + \sum_{i=1}^{N} \hat{y}_c^{(i)}} \tag{6.3}$$

First we ran experiments on labeled data only. We trained models from scratch and we fine tuned pretrained models which can be seen in Table 6.5 and Table 6.6. And then put to use also our unlabeled dataset. Based on the previous results of loaded model v. trained from scratch, we continued only with training pretrained models. Result are in Table 6.7.

Used $\mathcal{L}_{sup}$	**Used $\mathcal{L}_{unsup}$**	DSC	TP DSC
CE + (6.3)	(6.3)	0.7932	0.8114
CE + (6.3)	(6.1)	**0.8146**	**0.8356**
(6.2)	(6.2)	0.7745	0.7933

Table 6.5: Results of consistency learning with new model trained from scratch and only with labeled data

Pseudolabels with consistency learning

We also employed combination of both methods described above. Based on Table 6.6 and Table 6.7 we figured not to use IoU anymore. Instead we

Used $\mathcal{L}_{sup}$	**Used** $\mathcal{L}_{unsup}$	DSC	TP DSC
CE + (6.3)	(6.3)	0.8265	0.8525
CE + (6.3)	(6.1)	**0.8414**	**0.8586**
(6.2)	(6.2)	0.7824	0.8151

Table 6.6: Results of consistency learning with pretrained model (DSC 0.842, TP DSC 0.853) on labeled data only

Used $\mathcal{L}_{sup}$	**Used** $\mathcal{L}_{unsup}$	DSC	TP DSC
CE + (6.3)	(6.3)	0.8185	0.8517
CE + (6.3)	(6.1)	**0.8303**	**0.8574**
(6.2)	(6.2)	0.6805	0.6437
CE + (6.3)	(6.3)+ (6.1) + (6.2)	0.8093	0.8469
CE + (6.3)	(6.3)+ (6.1)	0.8163	0.8513

Table 6.7: Results of consistency learning with pretrained model (DSC 0.842, TP DSC 0.853) on both labeled and unlabeled data

employed combination of Dice loss and Probability difference. Results are shown in Table 6.8

Used $\mathcal{L}_{sup}$	**Used** $\mathcal{L}_{unsup}$	DSC	TP DSC
CE + (6.3)	(6.3)	**0.8397**	0.853
CE + (6.3)	(6.1)	0.8135	0.8402
CE + (6.3)	(6.3)+ (6.1)	0.8288	**0.8597**

Table 6.8: Results of combined pseudolabels and consistency learning

Chapter 7

Results

To acquire statistically meaningful results, we conducted 9 fold cross validation. Then we evaluated models on testing data and compared with dentist students.

7.1 Image classification

First we mention results of YOLO. This idea showed to be very effective. As we can see in Table 7.1, this detection model is very strong. It made only few mistakes, all justifiable after revision (braces, questionable cases). When consulted with dentists, they even corrected their annotations with newly found restorations. There were only handful of false negative results where dentists are sure about their annotation. We can see such image in Figure 7.1. During cross validation we tested more confidence thresholds and came to conclusion, that it should be set around 0.56 for best performance. We also wanted to know if could utilize YOLO directly in segmentation. For example if we could add extra weight to pixels which are inside bounding boxes created by this model. However, YOLO does not detect all restorations, it found only 912 out of 1053 restorations in testing dataset. For now we do not consider that enough and only use YOLO as classifier.

7.2 Supervised segmentation

First we can see comparison of fully supervised model in this work and Kunt's fully supervised in Figure 7.2. Green color refers to true positive pixels, red color to false positive pixels and blue color to false negative pixels. We can see that model in this work is more capable of segmenting restorations with lower contrast. This is result of work on stage 1 dataset and testing which backbone and augmentations would work the best.

	Precision	Recall	Accuracy	Specificity
Nine fold cross validation	0.987 (SD = 0.01)	0.996 (SD = 0.0098)	0.991 (SD = 0.011)	0.996 (SD = 0.0092)
Results on testing data	1	0.989	0.987	1

Table 7.1: Results of YOLO used for classification of x-rays

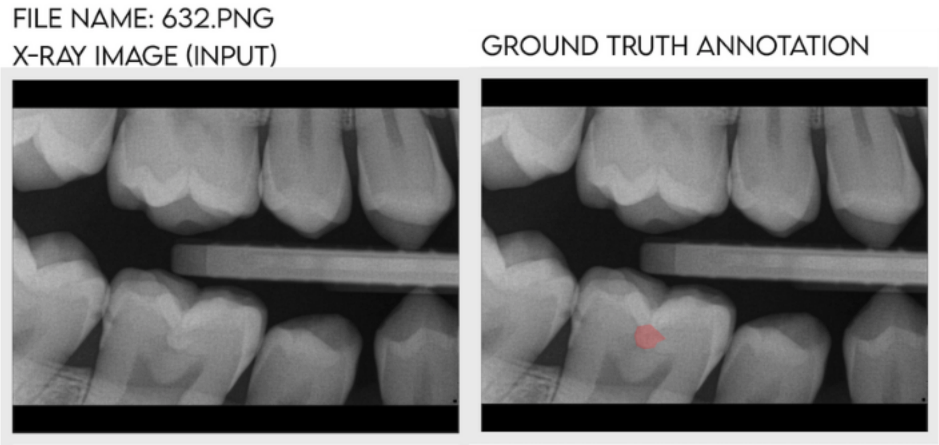


Figure 7.1: Sample from images, which YOLO could not classify correctly

7.3 Semi-supervised learning

We have chosen pseudolabels as described in 3.2.1 and Consistency learning model with 6.3 and 6.1 as loss functions. We have chosen these semi-supervised models to test them against fully supervised models and students. We compared students performance with the models on the same validation dataset, that was used through all the experiment in stage 2 (see Table 7.2. When examined closely, we could see that students approach was quite different. Student A missed quite a lot low contrast restorations. However did not have many false positives. On the other hand, student B marked a lot of places as restorations, even though there are none. Then we evaluated

Model/Student	TP DSC	DSC
Student A	0.7919	0.8327
Student B	0.0.7976	0.8422
Fully supervised U-Net (Kunt)	0.7823	0.6791
Fully supervised U-Net (Ours)	0.8514	0.8098
Consistency learning	0.8557	0.6493
Pseudolabels	**0.8596**	**0.8215**

Table 7.2: As these students were volunteers, we only asked them to annotate validation data of one fold. That consists of 98 images. This table then depicts result only for one fold of validation data

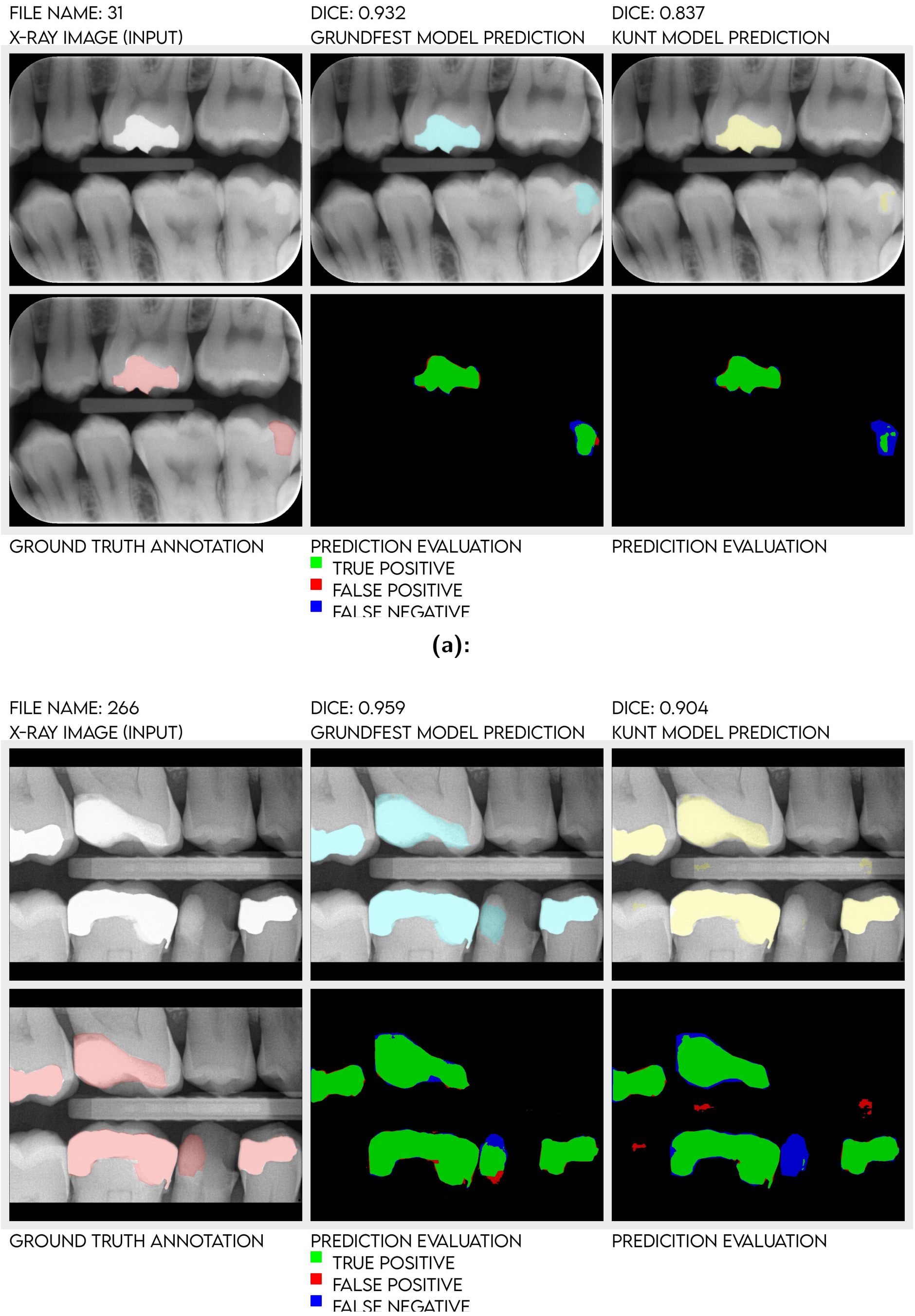
models through 9 fold cross validation. Results are in Table 7.3. At this point we did not use YOLO as a filter, because we wanted to know the behavior of the models with all images. Finally we evaluated the best performing models of each type on testing data. Here we first used YOLO to filter out images without restorations. Numeric results are in Table 7.4. To get better idea of results, it is important to look at the histograms in Figure 7.3.

Model	TP DSC	DSC
Fully supervised U-Net (Kunt)	0.8031 (SD = 0.015)	0.6359 (SD = 0.019)
Fully supervised U-Net (Ours)	0.8597 (SD = 0.009)	**0.8373** (SD = 0.012
Consistency learning	0.8624 (SD = 0.02)	0.6554 (SD = 0.026)
Pseudolabels	**0.8825** (SD = 0.017)	0.7116 (SD = 0.054

Table 7.3: Nine fold Cross-validation of models. We can see that semisupervised techniques did not outperformed fully supervised model. It even worsened the ability of the model to recognize that there are no restorations in the picture

	Supervised (Kunt)	**Supervised** (Ours)	**Consistency learning**	**Pseudolabels**	**Dentist student B**
Mean DSC	0.6489	0.8729	0.8778	**0.8930**	0.7111
DSC over all pixels	0.7309	0.8683	0.8591	**0.9023**	0.8422
Mean IoU	0.6194	0.7948	0.8084	**0.8272**	0.6251
IoU over all pixels	0.6898	0.7673	0.7530	**0.8220**	0.7274
Precision (Pixelwise)	0.8376	**0.9489**	0.9358	0.9373	0.8889
Recall (Pixelwise)	0.7957	0.8003	0.7940	**0.8699**	0.8002

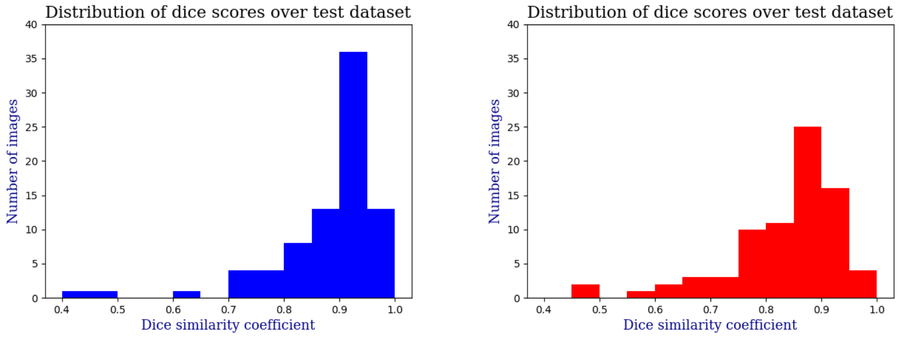
Table 7.4: This table compares performances of models and dentist student in several metrics on testing data. Mean DSC (mean IoU repectively) computes DSC (IoU) for each image and then computes mean. DSC (IoU) over all pixels counts all TP, FP and FN pixels and then computes DSC (IoU) from those. Precision and Recall in this table are also computed for each pixels, because we know the values for image classification from Table 7.1



(a):

(b):

Figure 7.2: We can see the main difference in the ability to correctly segment low contrast restorations.



(a) : Deep learning model using pseudolabels compared to ground truth.

(b) : Student B segmentation compared to ground truth

Figure 7.3: These histograms show how many images with each score was segmented. We compare deep learning model (blue) with student B (red). Such histograms can tell us that vast majority of predictions are good and only small number of troubelsome images are hard to automatically segment. Note that the y axis of histograms is not the same

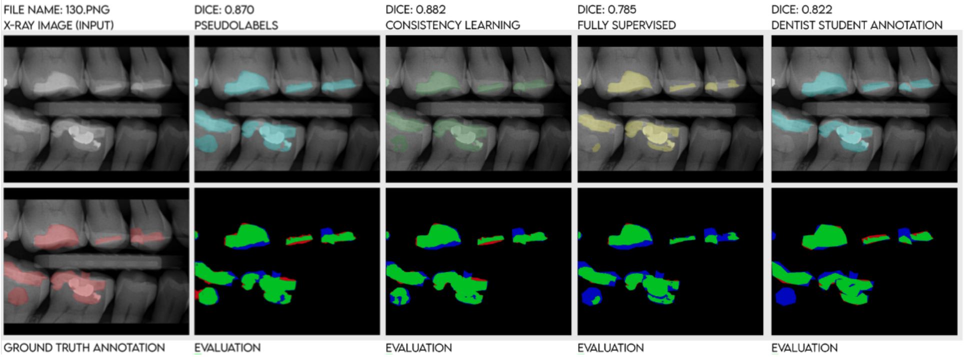


Figure 7.4: We can see comparison of several models with student B. From goes input image and ground truth (red highlights), pseudolabels, consistency learning, fully supervised learning and student B. Green pixels are true positive, red are false positive and blue are false negative. We can see that student did not notice low contrast restoration in lower left tooth, while models managed to capture it.

Chapter 8

Conclusion and further suggestions

In this work we focused on dental restoration binary segmentation. We achieved an improvement compared with Kunt's U-Net implementation. We optimized several hyperparameters specifically for this deep learning task and figured out how to apply data augmentation to best leverage U-Net's virtues. This all resulted in model's performance improvement.

Then we tried semi-supervised learning methods. Using pseudolabels, which turned out to be the most effective technique, we gained about 5% in TP DSC over supervised U-Net. We gained even more significant boost with employing YOLO as a filter for images without any segmentations. Such model is able to keep both precision and recall at high values.

To conclude, we created a tool which can create human level annotations in lot of cases, which can then be used to speed up dentists work or to help them in other ways. As we could see at the example of dentist students, it can outperform a novice dentist in restoration detection. The model is capable of creating better segmentation than humans using polygons, as it carefully fills contrastive areas.

The model still struggles with certain images. Especially when x-ray is not done precisely, which could cause appearance of white shadows. Example of such image is in Figure 8.2 These artifacts are as bright as amalgam restorations, which can then cause mistakes. Based on models predictions, our annotaters paid attention to troublesome images and corrected their annotations. This shows, that even at this level, the model can deliver second opinion which could prove to be very helpful when the load of images is too large for one human to grasp.

It is important to mention that ground truth is based only on x-ray images. Annotaters did not have any other information at hand. Thus it is possible, that even after being extremely careful they could have overlooked some low contrast restoration. Those restorations could be mixed up with enamel, which overlaps at the top of the tooth. This could create similar effect as restorations.

In future work we suggest to distinguish all restoration types, which could bring the goal of dental chart closer to reality. It could also help the model to

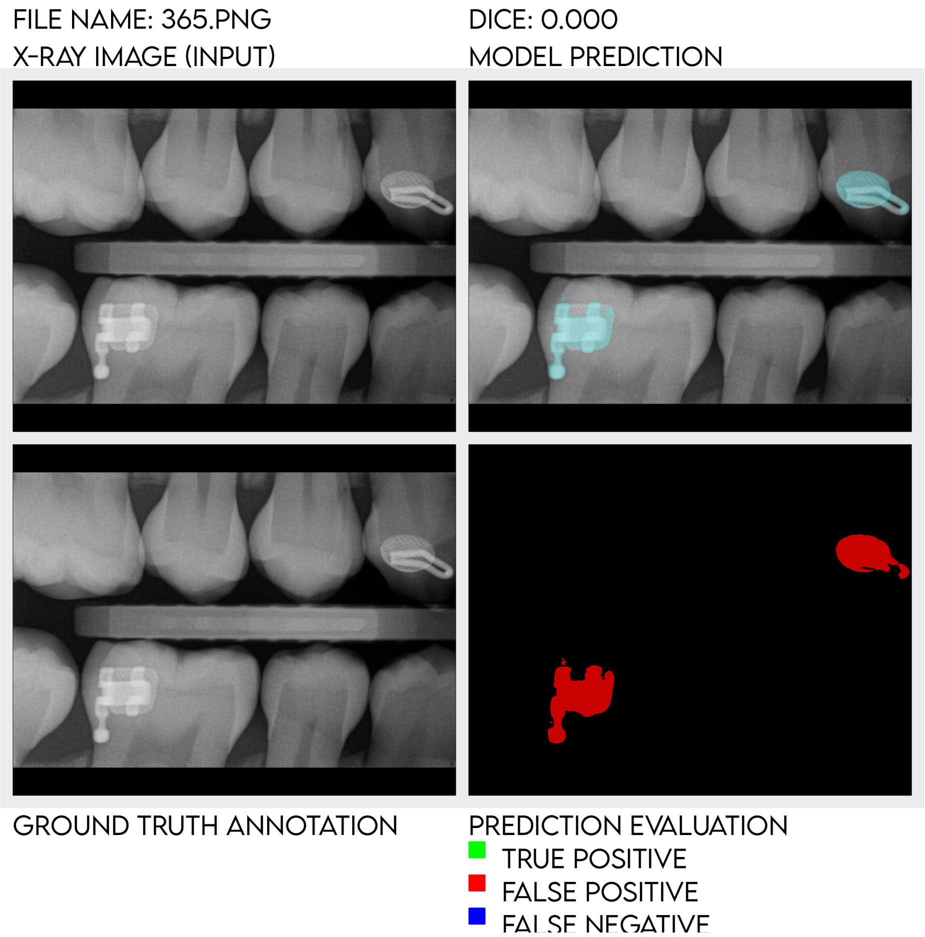


Figure 8.1: Image sample with well visible braces. The model recognizes some dental work, however ground truth would mark this prediction as false, because we do not mark braces and retainers. More of such images are in training set.

make better prediction, because it could have more options to choose from. Then we suggest to make special labeling category for braces (with retainers), because model recognizes such structures as something not usually present but then receives neagtive feedback (see Figure 8.1). With lack of images with such dental work, I think it could help the learning process.
Then we suggest to label teeth, so the model could tell if there is still a root of tooth under a bridge or if it was removed. Finally we suggest to try other backbones for segmentation network, such as Segnet or Resnet. Or it could be worth a try to find GPUs with bigger memory and run deeper U-Net or U-Net++.

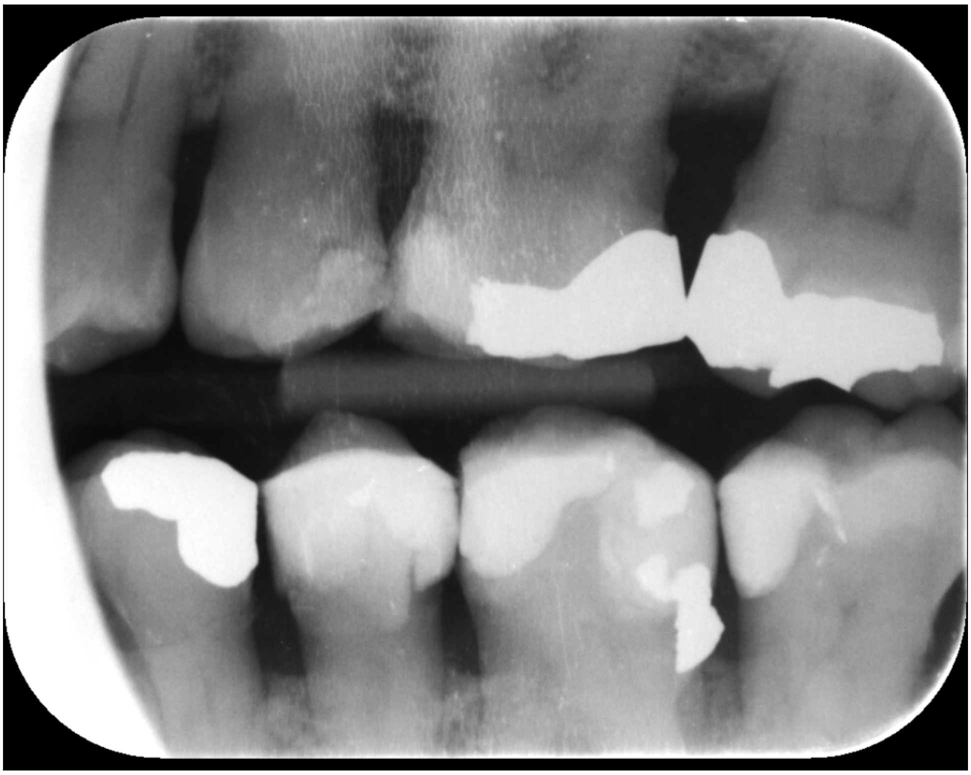


Figure 8.2: Image sample with white shadow artifact caused when taking x-ray image without sufficient precision.

Appendix A

Bibliography

[1] URL: https://www.cvat.ai.

[2] URL: https://github.com/vdumoulin/conv_arithmetic/blob/master/README.md.

[3] URL: https://stackoverflow.com/questions/50805634/how-to-create-mask-images-from-coco-dataset.

[4] Teeth crowns, dental bridges and other restorations, February 2023. URL: https://www.nzip.cz/clanek/667-korunky-mustky-a-jine-zubni-nahrady.

[5] Teeth implants, February 2023. URL: https://www.nzip.cz/clanek/668-zubni-implantaty.

[6] R. Abdalla-Aslan, T. Yeshua, D. Kabla, I. Leichter, and C. Nadler. An artificial intelligence system using machine-learning for automatic detection and classification of dental restorations in panoramic radiography. *Oral Surgery, Oral Medicine, Oral Pathology and Oral Radiology*, 130, November 2020.

[7] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39, December 2017.

[8] Oğuzhan Baydar, Ingrid Różyło-Kalinowska, Karolina Futyma-Gąbka, and Hande Sağlam. The u-net approaches to evaluation of dental bitewing radiographs: An artificial intelligence study. *Diagnostics*, 13(3), 2023. URL: https://www.mdpi.com/2075-4418/13/3/453.

[9] J. Bertels, T. Eelbode, and M. Berman et al. Optimizing the dice score and jaccard index for medical image segmentation: theory and practice. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, page 92–100, October 2019.

[10] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. *Proceedings of the eleventh annual conference on Computational learning theory*, 1998.

[11] Gerda Bortsova, Florian Dubost, Laurens Hogeweg, Ioannis Katramados, and Marleen de Bruijne. Semi-supervised medical image segmentation via learning consistency under transformations. *CoRR*, abs/1911.01218, 2019. URL: `http://arxiv.org/abs/1911.01218`, `arXiv:1911.01218`.

[12] A. Creanga, H. Geha, V. Sankar, F. Teixeira, C. McMahan, and M. Noujeim. Accuracy of digital periapical radiography and come-beam computed tomography in detecting external root resorption. *Imaging science in dentistry*, 45:153, September 2015.

[13] Richard Andrew Davies, Shaghayegh Ardalan, Wei-Hua Mu, Kun Tian, Fariborz Farsaikiya, Brian W. Darvell, and Gregory A. Chass. Geometric, electronic and elastic properties of dental silver amalgam γ-(ag3sn), γ1-(ag2hg3), γ2-(sn8hg) phases, comparison of experiment and theory. *Intermetallics*, 2009. URL: `https://doi.org/10.1016/j.intermet.2009.12.004`.

[14] O. Fejerskov, B. Nyvad, and E. Kidd. *Dental Caries: The Disease and Its Clinical Management.* BLACKWELL PUBL, May 2015. URL: `https://www.ebook.de/de/product/23695989/dental_caries_the_disease_and_its_clinical_management.html`.

[15] J. E. Frencken, P. Sharma, L. Stenhouse, D. Green, D. Laverty, and T. Dietrich. Global epidemiology of dental caries and severe periodontitis - a comprehen- sive review. *Jounral of Clinical Periodontology*, 44:94–105, March 2017.

[16] gesundheit.gv.at. Teeth restorations, February 2023. URL: `https://www.nzip.cz/clanek/666-zubni-vyplne`.

[17] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems*, 17, 2004. URL: `https://proceedings.neurips.cc/paper_files/paper/2004/file/96f2b50b5d3613adf9c27049b2a888c7-Paper.pdf`.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[19] H. Huang, Q. Chen, L. Lin, M. Cai, Q. Zhang, Y. Iwamot, X. Han, A. Furukawa, S. Kanasaki, Y.-W. Chen, R. Tong, and H. Hu. Mtl-abs3net: Atlas-based semi-supervised organ segmentation network with multi-task learning for medical images. *IEEE Journal of Biomedical and Health Informatics*, 26, 2022.

[20] W. Huang, C. Chen, Z. Xiong, Y. Zhang, X. Chen, X. Sun, and F. Wu. Semi-supervised neuron segmentation via reinforced consistency learning. *IEEE Transactions on Medical Imaging*, 2022.

[21] M. Hung, M. S. Lipsky, R. Moffat, E. Lauren, E. S. Hon, J. Park, G. Gill, J. Xu, L. Peralta, J. Cheever, D. Prince, T. Barton, N. Bayliss, W. Boyack, and F. W. Licari. Health and dental care expenditures in the united states from 1966 to 2016. *PLOS ONE*, 15, June 2020.

[22] Rushi Jiao, Yichi Zhang, Le Ding, Rong Cai, and Jicong Zhang. Learning with limited annotations: A survey on deep semi-supervised learning for medical image segmentation. 7 2022. URL: `https://arxiv.org/pdf/2207.14191.pdf`.

[23] N. J. Kassebaum, E. Bernabé, M. Dahiya, B. Bhandari, C. J. L. Murray, and W. Marcenes. Global burden of untreated caries: A systematic review and metaregression. *J Dent Res*, 94, March 2015. URL: `https://doi.org/10.1177/0022034515573272`.

[24] Muhammad Zubair Khan, Mohan Kumar Gajendran, Yugyung Lee, and Muazzam A. Khan. Deep neural architectures for medical image semantic segmentation: Review. *IEEE Access*, 9:83002–83024, 2021. `doi:10.1109/ACCESS.2021.3086530`.

[25] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL: `https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`.

[26] L. Kunt. Dental caries detection from bitewing x-ray images. Master's thesis, Czech Technical University in Prague, May 2022. URL: `http://hdl.handle.net/10467/101406`.

[27] Issam Laradji, Pau Rodriguez, Oscar Manas, Keegan Lensink, Marco Law, Lironne Kurzman, William Parker, David Vazquez, and Derek Nowrouzezahrai. A weakly supervised consistency-based learning method for covid-19 segmentation in ct images. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2453–2462, January 2021.

[28] S. Lee, S. il Oh, J. Jo, S. Kang, Y. Shin, and J. won Park. Deep learning for early dental caries detection in bitewing radiographs. *Scientific Reports*, August 2021.

[29] C. Li, L. Dong, Q. Dou, F. Lin, K. Zhang, Z. Feng, W. Si, X. Deng, Z. Deng, and P.-A. Heng. Self-ensembling co-training framework for semi-supervised covid-19 ct segmentation. *IEEE Journal of Biomedical and Health Informatics*, 25, 2021.

[30] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, June 2015.

[31] Priyanka Malhotra, Sheifali Gupta, Deepika Koundal, Atef Zaguia, and Wegayehu Enbeyle. Deep neural networks for medical image segmentation. *Journal of Healthcare Engineering*, 2022, 2022. URL: `https://doi.org/10.1155/2022/9580991`.

[32] Y.-C. Mao, T.-Y. Chen, H.-S. Chou, S.-Y. Lin, S.-Y. Liu, Y.-A. Chen, Y.-L. Liu, C.-A. Chen, Y.-C. Huang, S.-L. Chen, C.-W. Li, P. A. R. Abu, and W. Y. Chiang. Caries and restoration detection using bitewing film based on transfer learning with cnns. *Sensors (Basel, Switzerland)*, 2021.

[33] F. Milletari, N. Navab, and S.-A. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *CoRR*, October 2016.

[34] T. Miyato, S. i. Maeda, M. Koyama, and S. Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41, 2018.

[35] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *CoRR*, 2015. URL: `http://arxiv.org/abs/1506.02640`.

[36] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018. URL: `http://arxiv.org/abs/1804.02767`, `arXiv:1804.02767`.

[37] Andre Ritter. *Sturdevant's Art and Science of Operative Dentistry*. Elsevier, 2019.

[38] A. Rodriguez-Ruiz, K. Lång, A. Gubern-Merida, M. Broeders, G. Gennaro, P. Clauser, T. H. Helbich, M. Chevalier, T. Tan, T. Mertelmeier, M. G. Wallis, I. Andersson, S. Zackrisson, R. M. Mann, and I. Sechopoulos. Stand-alone artificial intelligence for breast cancer detection in mammography: Compari- son with 101 radiologists. *JNCI: Journal of the National Cancer Institute*, 111, March 2019.

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. URL: `http://arxiv.org/abs/1505.04597`, `arXiv:1505.04597`.

[40] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence°*, 39:640–651, 2017.

[41] G. Swarnendu, D. Nibaran, D. Ishita, and M. Ujjwal. Supplementary material for: Understanding deep learning techniques for image segmentation. *CoRR*, August 2019.

[42] L. Teng, H. Li, and S. Karim. Dmcnn: A deep multiscale convolutional neural network model for medical image segmentation. *J. Healthcare Eng.*, 2019, December 2019.

[43] B. H. Thompson, G. Di Caterina, and J. P. Voisey. Pseudo-label refinement using superpixels for semi-supervised brain tumour segmentation. *IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022.

[44] J. Wang, X. Li, Y. Han, J. Qin, L. Wang, and Z. Qichao. Separated contrastive learning for organ-at-risk and gross-tumor-volume segmentation with limited annotation. 2022.

[45] A. Warreth and Y. Elkareimi. All-ceramic restorations: A review of the literature. *Saudi Dent J.*, December 2020. `doi:10.1016/j.sdentj.2020.05.004`.

[46] H. Wu, Z. Wang, Y. Song, L. Yang, and J. Qin. Cross-patch dense contrastive learning for semi-supervised segmentation of cellular nuclei in histopathologic images. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[47] Xuanang Xu, Thomas Sanford, Baris Turkbey, Sheng Xu, Bradford J. Wood, and Pingkun Yan. Shadow-consistent semi-supervised learning for prostate ultrasound segmentation. *IEEE Transactions on Medical Imaging*, 41(6):1331–1345, 2022. `doi:10.1109/TMI.2021.3139999`.

[48] Huifeng Yao, Xiaowei Hu, and Xiaomeng Li. Enhancing pseudo label quality for semi-superviseddomain-generalized medical image segmentation. *CoRR*, abs/2201.08657, 2022. URL: `https://arxiv.org/abs/2201.08657`, `arXiv:2201.08657`.

[49] T. Yeshua, Y. Mandelbaum, R. Abdalla-Aslan, C. Nadler, L. Zemour L. Cohen, D. Kabla, O. Gleisner, and I. Leichter. Automatic detection and classification of dental restorations in panoramic radiographs. *Issues in Informing Science and Information Technology*, 2019. URL: `https://www.researchgate.net/publication/332821011_Automatic_Detection_and_Classification_of_Dental_Restorations_in_Panoramic_Radiographs`.

[50] R. Zhang, S. Liu, Y. Yu, and G. Li. Self-supervised correction learning for semi-supervised biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2021.

[51] Yizhe Zhang, Lin Yang, Jianxu Chen, Maridel Fredericksen, David P. Hughes, and Danny Z. Chen. Deep adversarial networks for biomedical image segmentation utilizing unannotated images. *Medical Image Computing and Computer Assisted Intervention - MICCAI 2017*, pages 408–416, 2017.

[52] Ziyuan Zhao, Jinxuan Hu, Zeng Zeng, Xulei Yang, Peisheng Qian, Bharadwaj Veeravalli, and Cuntai Guan. MMGL: Multi-scale multi-view global-local contrastive learning for semi-supervised cardiac image segmentation. In *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, oct 2022. URL: `https://doi.org/10.1109%2Ficip46576.2022.9897591`, `doi:10.1109/icip46576.2022.9897591`.

[53] H. Zheng, L. Lin, H. Hu, Q. Zhang, Q. Chen, Y. Iwamoto, X. Han, Y.-W. Chen, R. Tong, , and J. Wu. Semi-supervised segmentation of liver using adversarial learning with deep atlas prior. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2019.

[54] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: A nested u-net architecture for medical image segmentation. *IEEE Trans. Med. Imag.*, 39, June 2020.

A.1 Useful links and solutions

A.1.1 CVAT Exports Corrupt COCO File

When creating segmentation mask I ran into problem caused by COCO file which was exported from CVAT. Official library PyCOCOTools developed for processing this file would only create one and the same mask (following this procedure [3]). The problem is that CVAT gives all annotations the same id and they can not be correctly assigned. Fortunately there is a simple solution. All you have to do is go to the PyCOCOTools source code ($/.local/lib/pythonX.X/site\_packages/pycocotools/$) and redefine this function.