

TUNISIAN REPUBLIC



Ministry of Higher Education
and Scientific Research
University of Carthage



National School of Engineering of Carthage

GRADUATION PROJECT REPORT

Presented to obtain the
National Engineering Diploma in Computer Science

By

Farouk JAZIRI

Decay Segmentation and Detection using Panoramic X-Rays

Professional supervisor : **Mr. Anoir NECHI** CTO

Academic supervisor : **Mr. Haythem GHAZOUANI** Professor

Realized within AI Diagnosis Vision



Academic Year 2021 - 2022

TUNISIAN REPUBLIC
Ministry of Higher Education
and Scientific Research
University of Carthage
National School of Engineering of Carthage

GRADUATION PROJECT REPORT

Presented to obtain the
National Engineering Diploma in Computer Science

By

Farouk JAZIRI

Decay Segmentation and Detection using Panoramic X-Rays

Professional supervisor : **Mr. Anoir NECHI** CTO
Academic supervisor : **Mr. Haythem GHAZOUANI** Professor

Realized within AI Diagnosis Vision



Professional supervisor

M. Anoir NECHI

Signature et cachet

STE.AL DIAGNOSIS
VISION
168 0055 / G
TEL: 27382472


Academic supervisor

M. Haythem GHAZOUANI

Signature

Dedication

I dedicate this priceless effort to everyone who has always believed in my abilities :

I would not have gotten this far without my parents, Hedi & Samiha, who made everything possible. Their words of encouragement and endless support have accompanied me on this challenging but wonderful path.

I shall be eternally thankful to you, my siblings, Haydar & Yosr, for your unconditional love and support and the faith you have placed in me.

To All of my Spicy Gang, I became motivated and inspired. I'll never be able to repay you for all the creativity & spices you shower on me. This friendship will always be grateful and HOT in my heart.

To my AIESEC Fam, thank you for ignoring my imperfections and being my strength in weakness over the years. You are the kind of people everyone wishes they had. A massive portion of my heart is reserved for you.

To everyone who stood by my side during my journey, my dearest friends, you are more than valuable to me.

Thank you for the time & love you provided for me.

Farouk Jaziri

Acknowledgement

This work would not have been possible without the valuable cooperation of a number of people I want to pay tribute to. I would like to thank all those who have made this internship a rewarding and enjoyable experience, especially :

I wish to thank all the professors and the administration of the best engineering school in Tunisia; this would never be possible without their hard effort dedicated to us.

My academic supervisor Mr. Haythem Ghazouani, My Professor and mentor, for his availability, remarks, and valuable advice. I express my respect and sincere gratitude for your precious help and advice, contributing enormously to my work progress.

Special thanks to Mr. Anouar Nechi, a Research Assistant and Ph.D. student with three years of experience working alongside the University of Lübeck, for his guidance throughout the project and patience with me.

I would particularly like to thank my big bother, Mr. Ahmed Haddaoui, the CMO within AIDV, for his endless help and availability during my internship. Words can't be enough to describe your support. I express my sincere appreciation for your precious assistance in contributing enormously to my professional life.

Finally, I express my recognition to the members of the jury for accepting to evaluate my work, and I hope they find the expected clarity qualities in this report.

Table of Content

General Introduction	1
1 Scope of the Project	3
1 General scope	4
1.1 Presentation of the host institution: AI Diagnosis Vision	4
1.2 Team and Services	5
1.2.1 Team	5
1.2.2 Services	5
1.3 Technologies	6
2 Presentation of the project: Decay Segmentation and Detection Using Panoramic X-Rays	6
2.1 Problem statement	6
2.2 Purpose of the project	7
2.3 Project steps	7
3 Data Science project management methodologies	8
3.1 SEMMA	8
3.2 KDD	9
3.3 CRISP-DM	9
3.4 Comparison and approach chosen	10
2 State of the art	12
1 Business Understanding	14
1.1 Desired Output	14
1.2 Business Requirements	14
2 Stomatology	15

2.1	Dental X-rays	16
2.2	Application	16
2.2.1	Panoramic X-Rays	17
2.2.2	Limitation of Panoramic X-Rays	17
2.3	Potential Diseases	17
3	Latest Techniques Used on Dental Radiography	18
3.1	Automatic and Effective Tooth Isolation Method	18
3.1.1	Tooth isolation method of Nomir and Abdel-Mottaleb . .	18
3.1.2	The proposed tooth isolation method	19
3.1.3	Summary	20
3.2	Unsupervised Caries Detection in Non-standardized Periapical Dental X-Rays	20
3.2.1	Segmentation	21
3.2.2	Caries Detection	22
3.2.3	Experimental Results	23
3.3	Learning compact and discriminative hybrid neural network for caries classification	25
3.3.1	The research method	25
3.3.2	Results and Discussion	27
3	Data Processing	29
1	Data Understanding	31
1.1	Data Collection	31
1.1.1	Properties	32
1.1.2	Data Quality	32
1.2	Radiograph Disfavors	32
2	Data Summary	33
3	Data Cleaning	33
3.1	ROI. Definition	34
3.2	Image Enhancement	35
4	Data Construction	36
4.1	Jaw Partition	36

4.1.1	Integral Projection	37
4.1.2	Linking Consecutive Points	38
4.2	Localization of the Teeth Gap Valley	39
4.2.1	Neck Detection	39
4.2.2	Line Sum Intensities	40
4.2.3	Generating Borders	41
5	Data Formation	43
5.1	Infected Teeth Isolation	43
5.2	Teeth Segmentation	44
5.2.1	Find Contours	44
5.2.2	Annotation Reform	45
5.3	Dataset Health Check	46
6	Data Augmentation	47
6.1	Resize Images	48
6.2	Random Brightness and Contrast	48
6.3	Random Gamma	49
6.4	Sharpen	49
6.5	Flip	49
6.6	Affine	50
6.7	Grid Distortion	50
6.8	Random Sized Crop	51
6.9	Dataset Health Check	51
4	Modeling and Evaluation	54
1	Modeling	56
1.1	Instance Segmentation	56
1.2	Detectron2 - Mask R-CNN	57
1.2.1	Network Architecture	58
1.2.2	Results	58
1.3	Test Design Generation	60
1.4	Model Build	61
1.4.1	Architecture of Detectron2	61

1.4.2	Tools and Packages	62
1.4.3	Load and Register the Dataset	64
1.4.4	Customize Configurations	65
1.4.5	Customize Default Trainer	66
1.4.6	Train the Model	68
1.4.7	Prediction and Visualization	68
1.5	Model Assessment	69
1.5.1	Initial Results	70
1.5.2	Intermediate Results	74
1.5.3	Final Results	78
2	Evaluation	82
2.1	Results	82
2.2	Next Step	82
Conclusion and perspectives		83
References		84

List of acronyms

- **AI.** Artificial Intelligence
- **AIDV.** AI Diagnosis Vision
- **ANN.** Artificial Neural Network
- **AP.** Average Precision
- **BMD.** Bone Mineral Density
- **CNN.** Convolution Neural Network
- **CT.** Computed Tomography
- **DNN.** Deep Neural Network
- **FC. Layers** Fully Connected Layers
- **FPN.** Feature Pyramid Network
- **HNN.** Hybrid Neural Network
- **JSON.** JavaScript Object Notation
- **LR.** Learning Rate
- **MRI.** Magnetic Resonance Imaging
- **R-CNN.** Region based Convolutional Neural Network
- **ROI.** Region Of Interest
- **RPN.** Region Proposal Network
- **PNG.** Portable Network Graphics
- **X-Ray.** X-Radiation

List of Figures

1.1	AIDV's logo	4
1.2	SEMMA methodology	8
1.3	KDD methodology	9
1.4	CRISP-DM methodology	10
1.5	Comparison of KDD, CRISP-DM and SEMMA Processes	11
2.1	Teeth Anatomy and Numbering	15
2.2	X-Ray Types: (a) Bitewing, (b) Periapical, (c) Panoramic	16
2.3	Potential Dental Diseases	18
2.4	Isolation result examples for panoramic radiograph (a), (c) and periapical radiograph (b), (d).	20
2.5	Dental Caries Classification System Architecture	25
2.6	Simulation Process	27
3.1	Dataset Preview	31
3.3	Spinal-column Cover Example (in the red circle)	32
3.2	Blurry X-Ray Example	33
3.4	Image Sample from the Data Set Annotated with Decay Bounding Boxes .	34
3.5	The Extracted Lengths from the Research Study	34
3.6	Result Examples of the ROI. Definition	35
3.7	Applied Enhancements	36
3.8	The Final Result after Enhancement	36
3.9	Horizontal Projection of the Intensities for the Starting Point	37
3.10	Initial Partition	38
3.11	Results of the Polynomial Least Squares Fitting	39
3.12	Tooth Cross-sectional Anatomy [1]	39

3.13	The Splines passing through the dental pulp on which the necks can be found (red lines) and the spline separating upper and lower jaw (green line)	40
3.14	Line Sum Intensities Applied on the three Splines	41
3.15	Gaussian Filter used to Reduce Noise	41
3.16	Filtered Line Sum Intensities	42
3.17	The Upper Jaw Curve with its respective intensity values	42
3.18	Generated Separation Lines	43
3.19	Generated Images From Original Sample	44
3.20	The Last Updated Bounding Boxes	44
3.21	Contours Finding in the Decay Area	45
3.22	Teeth with Segmented Decay	45
3.23	Annotation Files	46
3.24	Size Distribution extracted from the previous processing	47
3.25	Annotation Heatmap extracted from the previous processing	47
3.26	Data Augmentation samples	52
3.27	Size Distribution extracted after augmentation	52
3.28	Annotation Heatmap After Image Augmentation	53
4.1	The Comparison between image classification, sementatic segmentation, object detection and instance segmentation [2]	57
4.2	The Mask R-CNN framework for instance segmentation	58
4.3	Faster R-CNN, a single unified network for object detection [3]	59
4.4	FCIS+++ (top) vs. Mask R-CNN (bottom, ResNet-101-FPN). FCIS exhibits systematic artifacts on overlapping objects [3]	59
4.5	Train Samples	60
4.6	Validation Samples	60
4.7	Test Samples	60
4.8	Detectron2 Architecture	62
4.9	Google Colab Selected Plan	63
4.10	1 st Tensorboard Results of the Configuration described in Table 4.3	70
4.11	Comparaison between Ground Truth and Predictions of the 1 st model	71
4.12	2 nd Tensorboard Results of the Configuration described in Table 4.4	72
4.13	Comparaison between Ground Truth and Predictions of the 2 nd model	72

4.14	3^{rd} Tensorboard Results of the Configuration described in Table 4.5	73
4.15	Comparaison between Ground Truth and Predictions of the 3^{rd} model	74
4.16	4^{th} Tensorboard Results of the Configuration described in Table 4.6	75
4.17	Comparaison between Ground Truth and Predictions of the 4^{th} model	75
4.18	5^{th} Tensorboard Results of the Configuration described in Table 4.7	76
4.19	Comparaison between Ground Truth and Predictions of the 5^{th} model	77
4.20	6^{th} Tensorboard Results of the Configuration described in Table 4.8	78
4.21	Comparaison between Ground Truth and Predictions of the 6^{th} model	78
4.22	7^{th} Tensorboard Results of the Configuration described in Table 4.9	79
4.23	Comparaison between Ground Truth and Predictions of the 7^{th} model	80
4.24	8^{th} Tensorboard Results of the Configuration described in Table 4.10	81
4.25	Comparison between Ground Truth and Predictions of the 8^{th} model	81

List of Tables

2.1	Comparison of tooth isolation accuracy	20
2.2	Region specific segmentation results comparison	23
2.3	Overall Segmentation Results Comparison	24
2.4	Caries identification results comparison	24
2.5	Comparaison of Performance Results	26
3.1	Statistic data retrieve of our images dataset	33
3.2	Size Distribution extracted from the previous processing	46
3.3	Aspect Ratio extracted from the previous processing	47
3.4	Parameters of the Resize Function	48
3.5	Parameters of the RandomBrightnessContrast Function	49
3.6	Parameters of the RandomGamma Function	49
3.7	Parameters of the RandomGamma Function	50
3.8	Parameters of the Flip Function	50
3.9	Parameters of the Affine Function	50
3.10	Parameters of the GridDistortion Function	51
3.11	Parameters of the RandomSizedCrop Function	51
3.12	Size Distribution extracted after augmentation	51
3.13	Aspect Ratio extracted after augmentation	52
4.1	Mask R-CNN ResNet101-C4, 3x model with detectron2 [4]	57
4.2	Instance Segmentation Results on COCO test-dev [3]	59
4.3	1 st Training Results	70
4.4	2 nd Training Results	71
4.5	3 rd Training Results	73

4.6	4^{th} Training Results	74
4.7	5^{th} Training Results	76
4.8	6^{th} Training Results	77
4.9	7^{th} Training Results	79
4.10	8^{th} Training Results	80

General Introduction

Most individuals are strange about the concept of artificial intelligence (**AI**). In 2017, barely one-fifth of 1,500 senior business leaders in the United States said they were familiar with **AI**. Several were unaware of what it was or how it might impact their businesses.

Tooth decay, often known as dental carie, is a condition that affects millions of individuals daily. Dental caries can be classified as beginning, moderate, or severe based on the degree of the lesion. By avoiding more invasive operations, early detection of tooth caries can save money in the long term. **X-rays** are the gold standard for identifying demineralized proximal caries, which are difficult to detect clinically. Radiography and a visual examination make detecting proximal caries straightforward. Fluorescence-based technologies such as DIAGNOdent (KaVo Charlotte, NC, USA) and fiber optic transillumination can be employed to diagnose dental caries. One downside of these techniques is their inability to identify posterior starting proximal caries and the cost of additional equipment. As a result, radiographs are still the most commonly used method in clinical settings.

Even though radiographs are frequently used as a diagnostic tool, it might be challenging to identify dental caries. Observers disagreed on whether caries lesions are visible using the same X-Ray. Factors like the quality of the radiograph, how the dentist perceives it, and the time it takes to perform each test can all impact inter-rater agreement.

Deep learning and **CNNs** have been actively researched for their application in interpreting many medical images, with promising results. Deep learning has become more prevalent in illness diagnosis, resulting in better clinical results. Deep convolutional networks have been investigated in dentistry since 2015.

During this project, we will try our best to achieve good results with our approach; thus, this report will be organized into four chapters.

In the first chapter entitled Scope of the project, we start with a presentation of the host institution, then describe our project's main context, issue statement, and goals.

The second chapter, entitled State of the art, first introduces the business background required to understand the dentistry field and the requirement for this project. Second, we present Stomatology (Oral medicine), its tools, limits, and potential diseases. Then, we put State of the art approaches into three primary kinds of research with their objective, process, and results.

The third chapter, Data Processing, begins by describing our collected data, properties, quality, and morphology. After that, we start the preprocessing phase by cleaning and constructing data with a solid explanation of each method implemented in this section. Last, we describe each technique used in the data augmentation step.

Finally, in the last chapter, entitled Modeling, and evaluation, we describe our proposed approach to how we will train our model, then we expose the obtained results and predictions for the assessment.

We conclude our work with a broad conclusion in which we analyze our contribution and establish our vision for future enhancements to this project.

1

Scope of the Project

Content

1	General scope	4
1.1	Presentation of the host institution: AI Diagnosis Vision	4
1.2	Team and Services	5
1.2.1	Team	5
1.2.2	Services	5
1.3	Technologies	6
2	Presentation of the project: Decay Segmentation and Detection Using Panoramic X-Rays	6
2.1	Problem statement	6
2.2	Purpose of the project	7
2.3	Project steps	7
3	Data Science project management methodologies	8
3.1	SEMMA	8
3.2	KDD	9
3.3	CRISP-DM	9
3.4	Comparison and approach chosen	10

Introduction

This chapter seeks to define the project's overall context. First, let us introduce the host institution: AIDV. Then we describe the problem we are trying to address and the measures we took to get there. Finally, we describe the results of all approaches and compare them.

1 General scope

This section is dedicated to the presentation of the hosting institution as well as its vision and areas of expertise.

1.1 Presentation of the host institution: AI Diagnosis Vision



Figure 1.1: AIDV's logo

AI diagnosis vision [5] is a machine learning based web platform that provides an intelligent practice management solution for better financial and clinical outcomes.

The AI solutions help dental professionals identify problems earlier and with greater accuracy. Applying proprietary technologies and artificial intelligence to read and interpret dental imagery and provide instantaneous image reporting, AIDV recognizes and measures dozens of dental pathologies, restorative features, and natural anatomy. These detections form the basis of a robust set of information that clinicians, practice owners, laboratories, and insurers use to improve dental care efficiency, accuracy, and consistency.

Apart from detecting common pathologies, AIDV's additional expertise is detecting rare and malignant pathologies. Another functionality of the platform is to optimize and propose a treatment plan adapted to each patient and assist in prescribing drugs in case of interaction or allergy. These features are either implemented in the dental office management system platform or the analysis API, and the creation of a report of the radiological imagery can be implemented in a dental radiology machine. Today AIDV work on three types of radiographs (Panoramic, Retroalveolar, and bitewings), working on 20 classes of pathologies. In addition to detecting common pathologies (caries, granulomas, prostheses, fillings...), they also work on complex pathologies such as tumors (Ameloblastomas, keratocysts, Carcinomas...). AIDV has collected data from different hospitals and universities (Tunisia, Algeria, Peru, Algeria, Egypt, and Canada...) with diagnoses already confirmed by anatomopathological examinations.

AIDV has tested the suitability of the AI solution for the market by implementing the algorithm on the Internet. They have just released the platform's second version, and they have also developed an X-ray analysis API that can be used in X-ray machines.

Hundreds of millions of dentists worldwide are looking for similar services, with a market worth of 138 million euros. The Mena regions constitute a vast and untouched AI and oral health market. Exploring this market will benefit the entire area, accelerating dentists' workflow and improving report analysis of patients.

1.2 Team and Services

1.2.1 Team

- **CEO** Saoussen Ayari:

A dentist with four years of experience working in the public sector and 2 years in the private sector. Saoussen specializes in oncology with a thesis on adenoid cystic carcinoma and thesis award 2020.

- **CTO** Anoir Nechi:

Anouar Nechi is a Research Assistant and Ph.D. student with three years of experience working alongside the research team of the University of Lübeck. Anouar specializes in high-performance computing and deep learning algorithmsacceleration along with Hardware-Software codesign.

- **COO** Koussai Barhoumi:

A dentist with four years of experience working in the public sector and 2 years in the private sector. A thesis onkeratocysts and thesis award 2020.

1.2.2 Services

- **Intelligent Office:**

The AI business intelligence model merges detailed information to optimize diagnosis and treatment recommendations with artificial intelligence to achieve better clinical and financial outcomes for the dental practice.

- **Second Opinion:**

A dentist's AI-based virtual assistant can instantly and reliably identify dozens of pathologies. Lesions can sometimes go unnoticed during a stressful day and leave more time for patient interaction.

- **Perfect documentation:**

Automatically create a report of results with a reading of the radiological images marked on the radiograph and treatment plans for a care sheet and a conversation with the patient.

- **functionality:**

Step 1: **Upload** - Upload a digital image directly or take a good quality photo of the panoramic X-Ray.

Step 2: **Analyze** - The neural network will analyze the panoramic image and recognize dental anatomy and anomalies.

Step 3: **Visualize** - The solution analyzes, processes, and documents all the necessary data with a much better reporting quality than other solutions.

Step 4: **Report** - Create a patient report for perfect documentation.

1.3 Technologies

The team introduced annotations from tens of thousands of radiological images into the artificial neural networks to train the AI with several modifications and hyperparameters to develop AI diagnosis vision. The result of this intensive training of the software is the detection model with an accuracy of 92%.

The software loads, processes, and documents all necessary data, and according to a study, a report quality is significantly higher than many dentists' reports.

What is the Procedure for obtaining these models ?

- **Data Collection:**

Collection of thousands of panoramic radiological images with a known diagnosis.

- **Data Annotation**

Annotate the specific characteristics and abnormalities of the radiological images (decays, prosthesis, fillings, peri apical images, tumors, cysts....).

- **Model Training**

Train the AI models on these labeled data so that they can identify these features on new features on new data.

- **Deployment**

Run the trained AI models on new images.

2 Presentation of the project: Decay Segmentation and Detection Using Panoramic X-Rays

In this section, we present our project: we begin by stating the overall problem and target goals. Then, we enumerate the necessary steps we took towards achieving the final objective.

2.1 Problem statement

Despite advances in oral health, dental caries remains the most common oral disease, with approximately 36% of the global population showing signs of infection [6]. This has led to many attempts to increase the detection rate of tooth decay to prevent developing more serious oral diseases.

Traditionally, dentists have used dental radiographs to assess unobservable areas of teeth and make a diagnosis by observation [7].

Recent advances in computer vision have led to the development of computer-aided diagnostic systems to aid identification and diagnosis. Unfortunately, these systems have a high false-positive rate in detecting caries and thus have not been used as stand-alone systems [8].

Several factors contribute to the unfavorable identification results associated with caries diagnosis, including noisy and low-contrast dental radiographs due to the low dose rates used during acquisition [9]. These low dose rates also reduce the visibility of cavities because X-rays cannot penetrate the teeth completely. Since low dose rates ensure patient health [10], there is no workaround but using image enhancement techniques to support computer vision.

2.2 Purpose of the project

This work's primary goal is to be included in an area of great interest to the scientific community. Computer vision has become increasingly part of our daily lives, whether for safety or leisure reasons.

Many researchers have implemented decay detection, yet our goal is not just to localize where the decay is. However, we want to provide the localization, the area, and the position of the infected teeth. These features will help us implement the proposed method within the company project in the future.

In other words, our primary goal is to detect caries in panoramic dental X-ray images. Based on the panoramic radiograph input, our proposal is to mark the infected teeth in the image and then segment the area of tooth decay.

2.3 Project steps

The main milestones of our project are:

1. Exploring the dental panoramic X-Ray image dataset.
2. Preprocessing the dataset
3. Applying different operations of data augmentation.
4. Constructing the detection model
5. Evaluating the model

These steps are a rough outline of the main tasks performed throughout our project. However, the further detailed steps, the order of the functions, and the iterations of the chapters, especially the first one, are more related and dependent on the project methodology we will adopt, which is presented and compared with other methods in the following section.

3 Data Science project management methodologies

This part describes and compares the benefits and drawbacks of three data-driven project management techniques. We will discuss the phases of the approach we used in our project near the end of the section.

3.1 SEMMA

SEMMA [11] is the acronym for Sample, Explore, Modify, Model, and Assess, and it relates to the steps involved in performing a Data Mining project. The SAS Institute views the procedure to be a five-stage cycle:

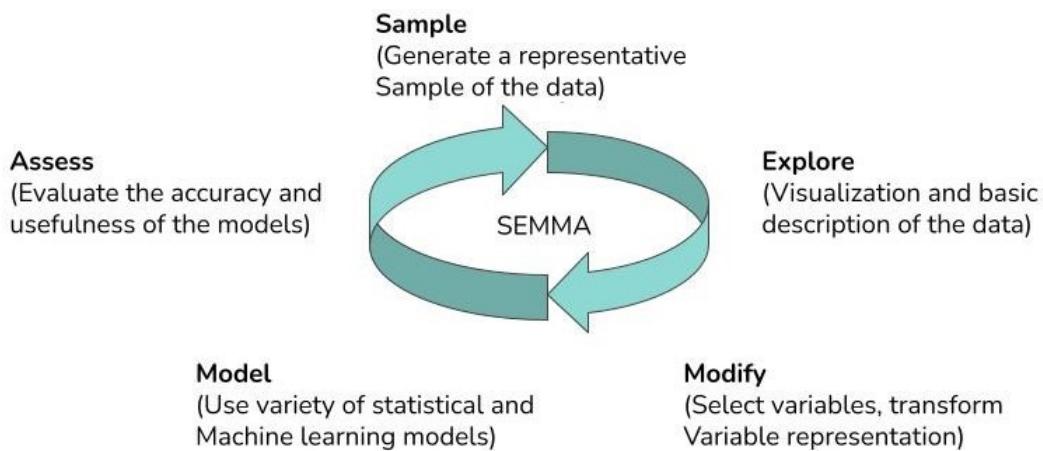


Figure 1.2: SEMMA methodology

- **Sample:** Create one or more data tables to sample the data. The samples should be big enough to hold all the essential data but tiny enough to process.
- **Explore:** In order to gain understanding and ideas, search the data for expected relationships, unexpected patterns, and anomalies.
- **Modify:** This stage entails updating the data by creating, choosing, and altering the variables to address the data quality issues found in the previous step. In general, here is where feature selection and engineering can be used.
- **Model:** Model the data by searching for a combination of data that reliably predicts a desired outcome using analytical techniques.
- **Assess:** Evaluate the data by determining the usefulness and veracity of the data mining discoveries.

3.2 KDD

Knowledge Discovery in Databases [11], or KDD, is a method for extracting, altering, and refining relevant facts and patterns from a raw database for usage in different domains or applications. The technique is iterative and is based on following the stages outlined in the diagram (Figure 1.3):

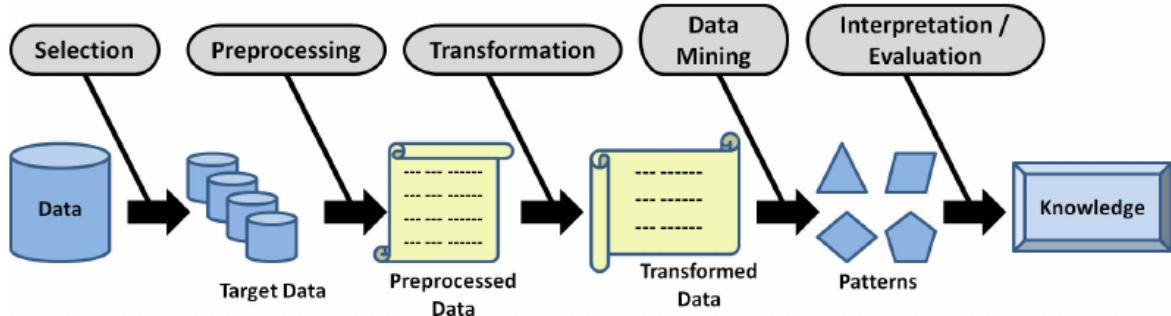


Figure 1.3: KDD methodology

- **Selection:** The information gathered must be selected and organized into meaningful sets depending on its availability, importance, and quality.
- **Preprocessing:** In order to increase the dependability and effectiveness of data, this stage entails searching for missing data and deleting noisy, redundant, and low-quality data from the dataset.
- **Transformation:** This stage prepares the data for the data mining algorithms to process. As a result, the data must be transformed and aggregated using methods such as dimensionality reduction, feature selection, etc...
- **Data Mining:** Algorithms are used to extract significant patterns from altered data, which aid in the development of prediction models.
- **Interpretation/Evaluation:** Interpretation and evaluation of the mined results in the context of problem.

3.3 CRISP-DM

IBM introduced the **CRoss Industry Standard Process for Data Mining** [11], an open standard for data mining project management, in 1999. It's a six-step strategy that increases the success of data-driven projects while reducing frequent mistakes.

This procedure is broken down into the following steps:

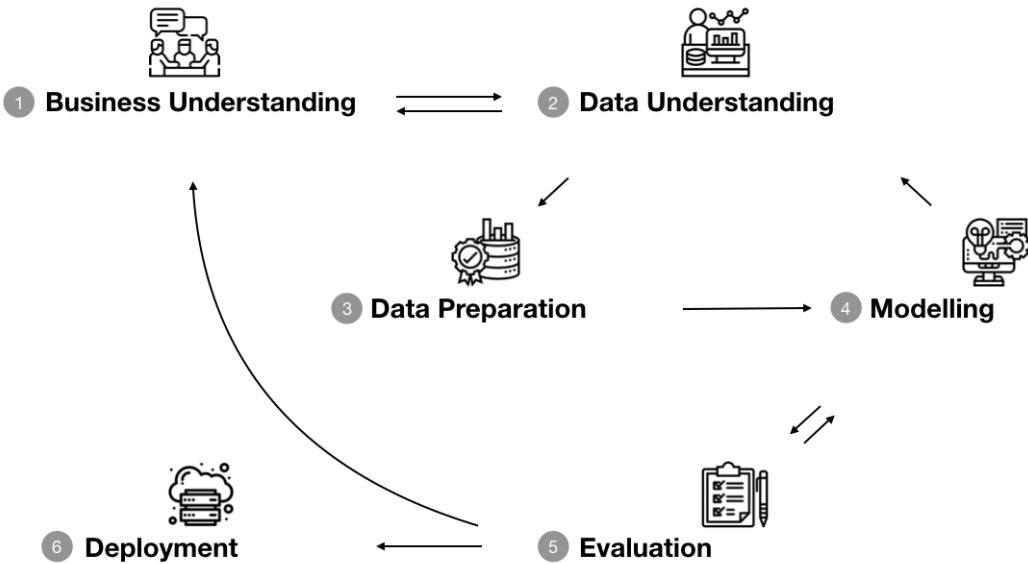


Figure 1.4: CRISP-DM methodology

- **Business understanding:** This initial phase focuses on gaining a business understanding of the project, translating these specifications into data mining problems, and developing a project strategy to achieve the stated goals.
- **Data understanding:** This phase includes data collection and review to gain initial insights, identify data quality issues (imbalanced datasets, missing values, Etc...), and/or discover important patterns or subsets.
- **Data preparation:** This phase includes preprocessing and cleaning tasks to obtain the final dataset to feed the model. Feature selection and engineering, removing or filling missing values, resizing, cropping, and other modifications are some of the many examples.
- **Modeling:** At this stage, various data mining methods and modeling techniques are applied to the prepared data (such as classical machine learning algorithms and neural networks). Create the model to achieve the desired result.
- **Evaluation:** The focus of this phase is to review the completed steps and comprehensively examine the results the model provides for unmet business goals.
- **Deployment:** This phase aims to plan deployment, monitoring, and maintenance so that the model can be used daily. At this point, the overall experience of the project is also reviewed and recorded for future reference.

3.4 Comparison and approach chosen

As presented in the [Figure 1.5](#), SEMMA and KDD are extremely similar as they contain the same number of steps that can be identified to be almost the same.

Because it clearly maps the business understanding and deployment phases, CRISP-DM is the most thorough process when compared to the other two. It also gives a bit

more detail abstraction, making it more industry-oriented. In 2002, 2004, 2007, and 2014, KD Nuggets, a popular data science online platform, named it the "The main methodology employed by industrial data miners."

For these reasons, we chose CRISP-DM to be our main project management methodology.

KDD	SEMMA	CRISP-DM
Pre KDD	-----	Business understanding
Selection	Sample	Data Understanding
Pre processing	Explore	Data preparation
Transformation	Modify	Modeling
Data mining	Model	Evaluation
Interpretation/Evaluation	Assessment	Deployment
Post KDD	-----	

Figure 1.5: Comparison of KDD, CRISP-DM and SEMMA Processes
[11]

Conclusion

This chapter introduces our hosting company, AIDV, followed by our problem statement. We also defined our goals and implemented a project management methodology.

In the following chapter, we will provide general information about the dental field and various approaches from related researches.

2

State of the art

Content

1	Business Understanding	14
1.1	Desired Output	14
1.2	Business Requirements	14
2	Stomatology	15
2.1	Dental X-rays	16
2.2	Application	16
2.2.1	Panoramic X-Rays	17
2.2.2	Limitation of Panoramic X-Rays	17
2.3	Potential Diseases	17
3	Latest Techniques Used on Dental Radiography	18
3.1	Automatic and Effective Tooth Isolation Method	18
3.1.1	Tooth isolation method of Nomir and Abdel-Mottaleb . .	18
3.1.2	The proposed tooth isolation method	19
3.1.3	Summary	20
3.2	Unsupervised Caries Detection in Non-standardized Periapical Dental X-Rays	20
3.2.1	Segmentation	21

3.2.2	Caries Detection	22
3.2.3	Experimental Results	23
3.3	Learning compact and discriminative hybrid neural network for caries classification	25
3.3.1	The research method	25
3.3.2	Results and Discussion	27

Introduction

In our study, we aim to address the concerns and the problems we will face during this project. We will present the project's business perspective, outcomes, and requirements. Finally, We will focus on the various methods for detecting decay tooth infections in the panoramic X-Ray to see how each system and approach handles this objective.

1 Business Understanding

The business understanding stage is critical because it helps clarify the dentist's goals.

At this stage, we need to ask the dentist many questions about every aspect of the problem; in this way, we are sure that we will study data related, and by the end of this stage, we have a list of business needs.

1.1 Desired Output

Here we describe the plan for achieving our goals and define the criteria that we will use to determine whether the project has been successful from the business point of view.

- **Project Plan:** During this project, the phase of preprocessing the data will be the most sensitive part. Hence, we will implement different algorithms in order to take as much information as possible from the panoramic X-Ray.
- **Business success criteria:** From the business point of view, the accuracy of detection dental decays is a priority, as well as giving useful insights on X-Ray images and creating a faithful segmentation of caries.

1.2 Business Requirements

We will define the needs and requirements for the project solution to assist in performing this work.

- **Mouth Anatomy:** Understanding the teeth numbering and their partition, The Jaw and bone form, the location of nasal fossa ([Figure 2.1](#)).
- **Dental Diseases Knowledge:** In order to understand this project, understanding the difference between dental diseases and their effect on X-Ray is a must.
- **Decay Detection:** How do Doctors identify Caries? This question aims to know the difficulty of this identification and the steps doctors take to find tooth decay.

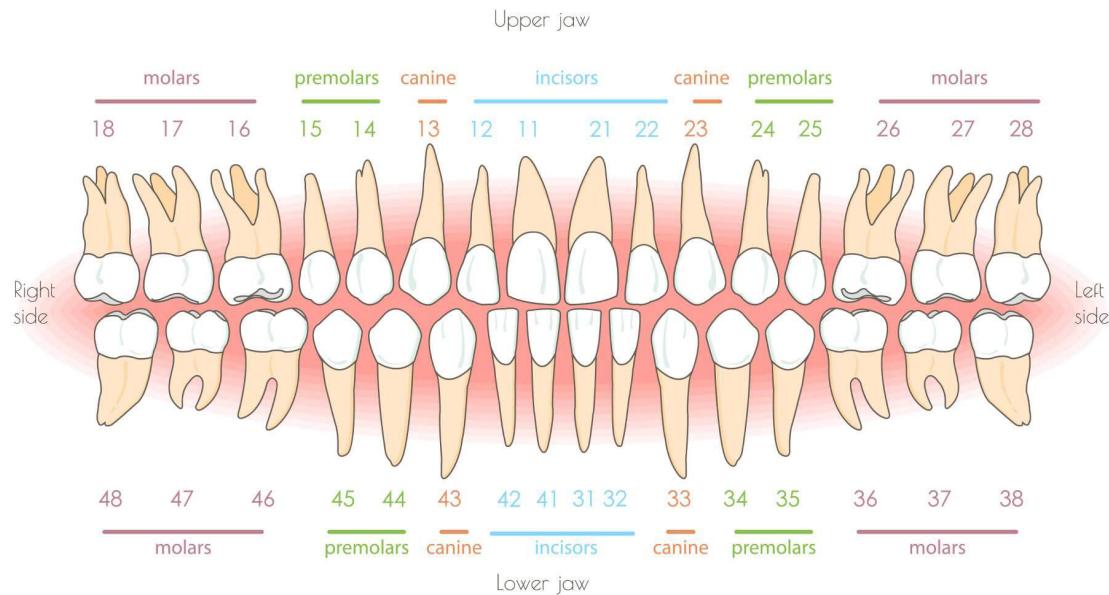


Figure 2.1: Teeth Anatomy and Numbering

2 Stomatology

Stomatology is the branch of medicine related to the mouth and its diseases, initially practiced by physicians, and it was a standard medical specialty until the early 20th century.

Oral medicine today is a field that employs only university-trained specialists, ie. doctors after completing a five to six-year college degree. Dentists - The staff of dentists is healthcare workers: nurses, dental technicians, X-ray technicians, and dental hygienists.

Primary Oral Specialties include Oral Therapeutics, Orthopedics, and Oral Surgery. Therapeutic dentistry (protective, conservative dentistry) deals with diagnosing, treating, and preventing tooth decay and its complications. Related to this essential branch of stomatology is pedodontics, which deals with the care of the primary dentition or the development of permanent dentition in adolescents.

Periodontology deals with diseases of the periodontal tissue and the oral mucosa. These diseases are mainly found on X-rays of apical teeth.

Orthodontics (prosthodontics) deals with the replacement of partial crowns, the loss of a single tooth, or the complete replacement of a lost tooth by making and using a variety of dentures (crowns, bridges, removable dentures). A separate focus is an orthodontics, which deals with diagnosing, treating, and preventing abnormal deformities of individual teeth, groups of teeth, and jaws.

Oral surgery deals with treating diseases of the mouth (alveolar surgery), or as a particular field (maxillofacial surgery), provides surgical treatment of more significant oral diseases mainly in the form of inpatient care.

Graduates of a Dentistry degree are theoretically and practically prepared for pre-

ventive and curative practice in the fundamental areas of Dentistry. He or she receives only general knowledge of the specialized discipline, enabling responsible and consistent treatment decisions in a highly specialized dental practice. As a doctoral student, after passing the prescribed examinations, he or she can obtain further qualifications and work as a specialist in orthodontics or maxillofacial surgery.

2.1 Dental X-rays

Dental X-rays are images of teeth, bones, and surrounding soft tissue that can help find problems with teeth, mouth, and jaw. X-rays show cavities, hidden tooth structures, and bone loss that are not visible to the naked eye.

Dental X-rays can also be used as follow-up care after dental treatment. There are three main types of dental radiographs: bitewing, periapical, and panoramic.

A series of periapical X-rays of the entire mouth is most often taken during a person's first visit to the dentist. A bitewing radiograph is used to look for cavities during the exam (see Figure 2.2). Depending on expert opinion, panoramic X-rays can sometimes be used.

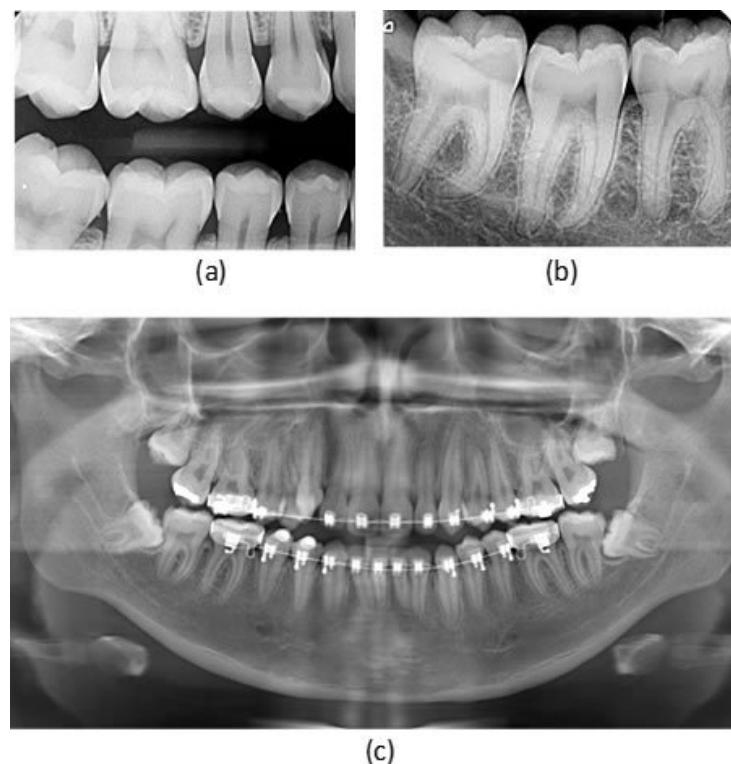


Figure 2.2: X-Ray Types: (a) Bitewing, (b) Periapical, (c) Panoramic

2.2 Application

During this project, we will focus on the panoramic X-rays Panoramic. How can we use them?

2.2.1 Panoramic X-Rays

X-rays are examinations frequently performed by dentists and oral surgeons in their daily practice and are an essential diagnostic tool.

It covers a larger area than a traditional oral X-ray, providing valuable information about the maxillary sinuses, tooth position, and other skeletal abnormalities. This exam also plans treatments for full and partial dentures, braces, extractions, and implants.

Automated image analysis methods have achieved higher relevance in many uses, and in medical image processing, the results can be considered satisfactory. Detecting health conditions as early as possible increases the usefulness of such automated systems.

Dental radiographs are also used in clinical settings to detect and predict bone mineral density (**BMD**) to diagnose osteoporosis. In [12], The authors assessed the trabecular patterns on dental X-rays to predict **BMD**.

2.2.2 Limitation of Panoramic X-Rays

A panoramic radiograph does not provide precise and detailed information about individual teeth or soft tissues such as muscles. It is typically used as a preliminary examination of the bones and teeth.

Because the mouth is curved, the panoramic x-ray may produce a slightly blurry image in which accurate metrics of the teeth and jaw are impossible. A computed tomography (**CT**) scan or magnetic resonance imaging (**MRI**) may be ordered if the dentist or surgeon requires additional information. This may include Dental Cone Beam **CT**, a specialized imaging exam for this specific body area.

2.3 Potential Diseases

There are a lot of oral diseases that could appear in a radiograph; among the common ones we list:

- **Decay:** Decay is also known as dental carie. It damages the surface of teeth over time.
- **Abscess:** A tooth with an abscess at its root is generally sensitive to touch or pressure.
- **Impacted Tooth:** An affected tooth is one whose eruption is considerably delayed.
- **Periodontal:** It results from infections and inflammation of the gums and bones surrounding and supporting the teeth.

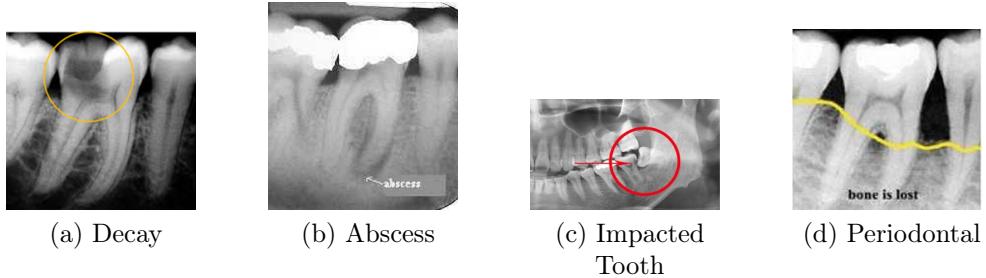


Figure 2.3: Potential Dental Diseases

3 Latest Techniques Used on Dental Radiography

3.1 Automatic and Effective Tooth Isolation Method

Tooth separation is a critical step for both computer-aided dental diagnosis and automatic tooth recognition systems, as it directly affects the accuracy of feature extraction and, thus, the final results of both systems.

This paper proposes an efficient and fully automatic method for dental fragment isolation, including maxillary/mandibular separation, single-tooth isolation, over-segmentation validation, and under-segmentation detection.

The mandibular separation mechanism is based on overall gray-scale projection to avoid potential information loss and integrates angle adjustment to handle distorted images.

An adaptive window scheme for locating gap valleys is proposed in single-tooth isolation to improve accuracy. In over-segmentation, an isolation curve verification scheme is proposed to remove excessive curves; and in the case of sub-segmentation, a scheme for detecting missing teeth is proposed instead.

The experimental results show that in the test database of 60 bitewing films, this method achieves 95.63% and 98.71% accuracy on maxillary and mandibular images, respectively, and performs better on images with severe occlusion and heavy dental work. Well, the illumination is not uniform compared to the method of Nomir and Abdel-Muttalib. The method without the maxillary separation step is also suitable for panoramic and periapical images [13].

3.1.1 Tooth isolation method of Nomir and Abdel-Mottaleb

Nomir and Abdel-Mottaleb [14] proposed a tooth isolation method for thresholding binary images. Their method can be described as the following three steps: iterative swelling, separation of the upper and lower jaws, and isolation of each tooth.

3.1.2 The proposed tooth isolation method

The proposed method for tooth isolation includes separating the maxilla and mandible, isolating individual teeth, checking the isolation curve for over-segmentation, and detecting missing teeth for under-segmentation.

Since dental radiographs always suffer from noise, low contrast, and uneven exposure, the image is enhanced before the teeth are isolated.

- **Image enhancement:**

For tooth isolation, the image is better enhanced so that the intensity of the tooth area is much raised than the rest of the image, which is almost as dark. Hence, many methods of image enhancement for dental radiographs, e.g., Adaptive strength stretching, top hat/bottom hat morphological transformation, and homomorphic filtering, are implemented.

- **Upper-lower jaw separation:**

Assuming that a horizontal or near-horizontal line can be used as an approximation to separate the maxilla and mandible, such a dividing line can be obtained using a horizontal projection histogram. Therefore, we make an integral projection for each horizontal line of the X-ray image.

- **Single tooth isolation:**

We performed single-tooth isolation by applying vertical separation using the proposed adaptive windowing scheme to draw a dividing line between each pair of adjacent teeth and then applying a thinning scheme to make each dividing line a segmentation curve.

- **Isolation-curve verification for over-segmentation:**

For each isolation curve, we obtain its pixel intensity distribution. If the intensity distribution of a particular isolation curve starts low and rises for a short time, then drops close to the baseline, accept the curve as valid and keep it; otherwise, consider them false and delete them.

- **Missing teeth detection for under-segmentation scheme:**

1. Obtain the vertical integral projection $V(x)$ and its first derivative $V'(x), x = 1, \dots, n$ (the width) of the rotated upper (lower) jaw image and calculate the mean value of $V(x)$, denoted as V_{mean} .
2. For $x = 1$ to n , assign position x to missing teeth region if $abs(V'(x) - V'(x+1)) < 45$ and $V(x) < V_{mean}$; otherwise, passing position x to normal region.
3. Discard the missing teeth region if its width is less than half of the width of a normal tooth, which is determined in the adaptive-windowing-based vertical separation scheme.
4. Add a vertical line at both ends of each detected missing teeth region.

- **Evaluation:**

	Upper jaw		Lower jaw	
	proposed method	N and AM	proposed method	N and AM
No. of teeth	252		233	
Correctly isolated no.	241	197	230	181
Isolation accuracy	95.63%	78.17%	98.71%	77.68%

Table 2.1: Comparison of tooth isolation accuracy

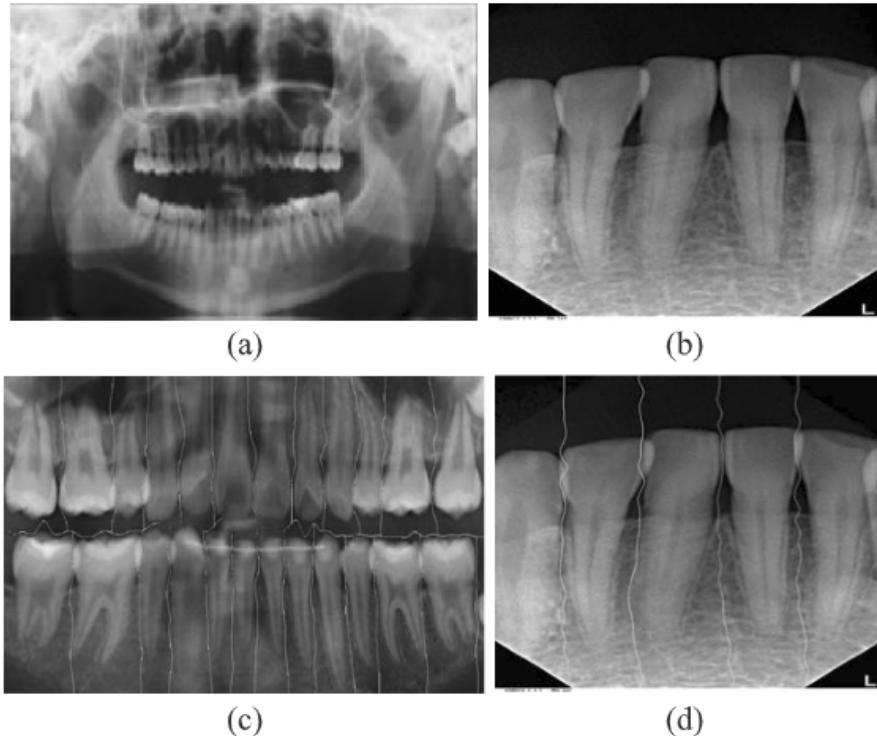


Figure 2.4: Isolation result examples for panoramic radiograph (a), (c) and periapical radiograph (b), (d).

3.1.3 Summary

The experimental results show that in the test database of 60 dental X-ray films, the accuracy of the maxillary image reaches 95.63%, and the accuracy of the mandibular image reaches 98.71%. , excessive dental work, and uneven tooth illumination than the Nomir and Abdel-Mottaleb methods.

3.2 Unsupervised Caries Detection in Non-standardized Periapical Dental X-Rays

This research presents an efficient model for caries detection on various non-uniform radiographs using single tooth segmentation, boundary detection, and caries detection via image analysis techniques.

Tooth segmentation is achieved using integral projection and analytical segmentation algorithms. Boundary detection is achieved using top and bottom hat transformations and active contours.

Finally, caries detection is achieved by performing blob detection and cluster analysis in various areas. Cluster analysis yields results relevant to the analyzed images and thus constitutes the unsupervised evaluation method of this work.

The results described in this document demonstrate the feasibility of this unsupervised learning model and its relative effectiveness compared to current systems in accurately diagnosing dental caries, with the proposed model achieving 96% correct diagnosis [15].

3.2.1 Segmentation

A three-stage procedure was used to segment the X-rays into individual teeth. This procedure includes picture enhancement and pre-processing, adaptive and iterative thresholding, and separation line selection utilizing a novel algorithm.

- **Pre-processing:**

Enhancing the quality of the processed pictures increased the diagnosis rate of the method described in this study. This optimization was accomplished by employing image enhancement techniques to reduce noise from the image and increase overall image quality. A preliminary cleanup was done to eliminate any anomalies in the pictures owing to the radiograph process to achieve the best possible outcome of the image improvements.

- **Image Enhancement:**

After removing all non-organic structures except dental fillings, the picture contrast was increased to enable more excellent delineation of the dental features. Noise reduction via blur filters was avoided to maintain feature detail. Following the findings of Yoonet al. [16] that Adaptive Histogram Equalization improved contrast for computer vision, a combination of a median filter followed by histogram equalization was adopted.

- **Thresholding:**

Iterative Thresholding. Because the X-Ray pictures being processed were comparable, the model presented in [14] was utilized as the foundation for the thresholding implementation. In each X-ray picture, a clever edge detector was employed to acquire the general contour of the teeth. The pixels in the area presumed to represent the toothed border were then obtained by applying morphological dilation to these edges. Approximately half of the pixels collected in this manner were of the teeth, while the remainder were of the jaw bone and other background objects. The initial threshold value was determined by averaging the pixel values of the supposed tooth pixels and the background pixels, and successive thresholds were determined as follows:

$$\mu_D^i = \frac{\sum_{(i,j) \in dental} f(i,j)}{\#dental_pixels} \quad (2.1)$$

$$\mu_B^i = \frac{\sum_{(i,j) \in dental} f(i,j)}{\#background_pixels} \quad (2.2)$$

$$T_{i+1} = \frac{\mu_B^i + \mu_D^i}{2} \quad (2.3)$$

Where $f(i, j)$ is the grayscale value of a pixel at point (i, j) , μ_D^i and μ_B^i are the mean grayscale values for their respective regions, and T_i , the threshold value for the whole image, is calculated from the average values of the background and teeth pixels.

Adaptive Thresholding. An adaptive thresholding approach was used to accurately distinguish tooth pixels from background, jaw, and gum pixels. The adaptive thresholding approach developed by [8] calculated the threshold value for an image; a pixel undergoes thresholding if, as the center pixel of a window of size IJ pixels, its grayscale value is less than the mean value of all non-zero pixels inside the window. As a result, the formula is:

$$T(i, j) = \frac{\sum_{s=-1/2}^{1/2} f(i + s, j + t)}{\#nonzero_pixels} \quad (2.4)$$

- **Tooth Separation:**

The separation of teeth was accomplished in two stages. The prospective separation lines are first constructed using integral projection, which defines the gap regions between detected teeth. The line of most excellent fit is then determined by an evaluation method.

3.2.2 Caries Detection

Caries detection took place in two stages. Potential regions of interest were first detected using an edge detector, revealing all places with dark patches and potential caries.

After defining a zone of interest, a new algorithm was used for the area in issue to determine the veracity of the caries flag.

- **Blob Detection:**

Following the testing of numerous approaches, a blob detector was built to detect possible cavities in the defined search region. Due to the high contrast of the picture from the top and bottom hat modification conducted during the boundary detection phase, probable decay regions were significantly darker compared to the surrounding dental substance. Blob detection algorithms used this due to their capacity to discover local maxima. Lindeberg's [17] blob detection model was used since it was unaffected by scaling difficulties caused by the varied sizes of the teeth being processed. To detect darker regions, the model employed a Laplacian of the Gaussian method, which was specified as a convolution kernel of this form:

$$LoG = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} e^{-\frac{x^2 - y^2}{2\sigma^2}} \quad (2.5)$$

where σ was the width of the kernel.

Using a 4-connected kernel resulted in some loss of definition along the borders of the caries clusters and hindered the diagnostic approach; hence, an 8-connected kernel was used.

- **Caries Analysis:**

Region of Interest Generation. Image analysis strategies were used to analyze the regions of interest using standard dentistry techniques to detect whether dental cavities were present with a non-supervised evaluation model. Because the pictures were periapical rather than panoramic, the particular tooth being studied was uncertain because the X-rays were collected at several sites. The following formula was developed to approximate the width:

$$W = T_{max} - \frac{T_{variance}(P_{max} - P_{calculated})}{P_{variance}} \quad (2.6)$$

Where W is the estimated width, T is the tooth's width, and P is the search space's percentage depth, determined to be 10–15%. $T_{variance}$ was obtained by calculating the maximum and minimum tooth width values and was determined to be 1,5. The value for $P_{variance}$ was calculated to be five following the same process.

Cluster Analysis. Several acceptance characteristics were satisfied for positive caries classification from flagged areas of interest. The mean pixel value of the cluster region was used to calculate a threshold value. A second threshold value was obtained by computing the mean pixel intensity of the area surrounding the suspected caries' location. Because caries originate in the tooth's enamel, the search space was limited to this area. This was mainly done to avoid inaccurate caries diagnoses caused by the darker dentin area interfering with the evaluation algorithm. The calculations were based on enamel thicknesses ranging from 0.87 to 1.45 mm, as specified in Reference [18].

3.2.3 Experimental Results

- **Segmentation Results:**

If the separation line did not produce partial separation or division of the teeth, the teeth were appropriately regarded as separated. Teeth that were already partial due to being on the X-edge rays were regarded as successfully separated if no more partiality was produced.

Teeth that were not accurately segmented were either owing to shallow contrast in the source picture, where enhancing methods could not distinguish between the tooth and non-teeth features, or impacted teeth.

Table 2.2 presents a comparison of the findings on a jaw-by-jaw basis. The results obtained by Oliveira, Nomir, and Abdel-Mottaleb are utilized as a comparison due to the similarity of the implemented methodologies used to produce dental segmentation.

	Upper Jaw	Lower Jaw
Oliveira [19]	72%	72%
Nomir and Abdel-Mottaleb [14]	84%	81%
Proposed approach	85%	90%

Table 2.2: Region specific segmentation results comparison

As can be observed, the segmentation results improved above current approaches by using a mix of the modified and innovative algorithms mentioned in this research.

	Accuracy (%)	Implementation
Nomir and Abdel-Mottaleb [14]	82.5	Thresholding
Said et al. [20]	83	Thresholding
Shah et al. [21]	58.1	Active contour
Phong-Dinh et al. [22]	77.23	Thresholding and integral projection
Oliveira [19]	71.91	Active contour without edge
Lai and Lin [23]	83	Region growing
Proposed approach	87.5	Thresholding and integral projection

Table 2.3: Overall Segmentation Results Comparison

The Table 2.3 compares the proposed approach to previous segmentation process implementations.

These findings imply that the strategy described in this study significantly improved over previous models. It also implies that the diagnostic algorithm obtained the maximum number of accurately segmented pictures for review.

- **Caries Detection Results:**

A comparison was made using the various diagnostic procedures available.

Caries detection by dentists utilizing the Logic on Caries Detector system, as mentioned by Tracy et al. [24], unsupported caries diagnosis by dentists, as discussed by Dykstra [25], and caries detection using a supervised learning model, as presented by Oliveira [19], Table 2.4 shows the outcomes of this comparison.

	Correctly categorized (%)	False positives (%)	Missed Caries (%)
Tracy [24]	94	-	6
Dykstra [25]	60	20	20
Oliveira [19]	98	-	2
Proposed approach	96	2	2

Table 2.4: Caries identification results comparison

The suggested model is implemented by utilizing a segmentation approach to split the X-rays into individual teeth, a boundary detection method to determine the margins of the teeth for caries analysis, and an image analysis methodology to analyze the border.

Because of the innovative methodologies mentioned in this work, both the suggested segmentation method and the caries detection algorithm achieved favorable results compared to similar models.

As a result, the caries detection model described in this study offers a feasible alternative to existing caries detection methods.

3.3 Learning compact and discriminative hybrid neural network for caries classification

Early detection and diagnosis of carious lesions are mandatory for proper dental care and health. For this, image processing techniques are used, further assisting specialists in making an accurate diagnosis.

Dental caries can be divided into different types based on location and severity. This classification is essential for the diagnosis and treatment planning of dental diseases.

This method recommends using HNN. technique to detect caries according to the type of caries to classify the layers affected by caries accurately. HNN. is a hybrid of ANN. and DNN. techniques. For image classification, deep neural networks rely on Stacked Sparse Auto-Encoder.

After that, ground-truth information can be retrieved from the data through supervised fine-tuning and unsupervised pre-training. On the other hand, ANN. relies on logistic regression to classify layers affected by caries. HNN. classifies dental caries into four distinct layers: enamel, dentin, pulp, and root lesions.

The proposed system provides efficient outputs for the input images and accurately classifies the degree of caries. HNN. is an innovative technique that provides good accuracy and processing time results.

3.3.1 The research method

Dental radiographs help to identify and analyze the extent of caries present in dental radiographs.

There is a wide range of imaging applications in dental radiography, and if applied correctly, it can improve diagnostic results and provide the perception of different stages of the imaging process that is limited in the field of diagnosis.

The proposed architecture is shown in the FIGURE. Dental X-rays are used as input images for preprocessing [26].

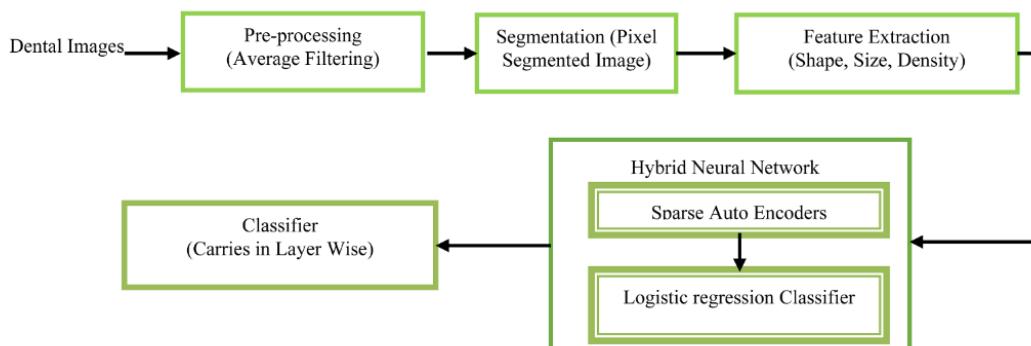


Figure 2.5: Dental Caries Classification System Architecture

- Pre-processing:

Filtering techniques are complex functions used to remove any noise from an image. The filter produces sharp and smooth images. First, convert the gray-scale image to a normalized image as output, then reduce the blurring effect. A selective median filter is used to remove noise in the image. The selective median filter [27] acts as a median filter for pixel values 0 and 255 and a Wiener filter for other pixel values.

$$\text{SelectiveMedianFilter} = \begin{cases} \text{MedianFilter} & \text{for } p(x) = 0 \text{ or } p(x) = 255 \\ \text{WienerFilter} & \text{otherwise} \end{cases} \quad (2.7)$$

- **Segmentation:**

In the Segmentation process, The tooth image is segmented or subdivided into individual pixels or objects. Thresholding is used to generate binary images, taking into account grayscale images and spatial regions that depend on image properties [28]. Thresholding techniques are used to detect blank areas present in an image.

- **Feature extraction:**

The feature extraction process deals with the extraction of different features of the tooth image. The following properties are taken into account when computing the features: shape, size, and density.

- **Hybrid Neural Network:**

This section illustrates and encourages using DNNs with multiple hidden layers to train and classify tooth images. HNN. is a hybrid of ANN. and DNN. techniques. Using these techniques, HNNs can be trained.

Deep Neural Network:

DNN. is a neural network with multiple layers and a certain complexity. Here, each layer of nodes is trained on a unique set of features based on the previous layer's output. To train on input images, DNNs rely on stacked sparse auto-encoders.

Artificial Neural Network:

Each neuron is assigned a pixel. There is a non-linear hidden layer where all neurons are based on a radial distance function. ANN. uses logistic regression to classify layers affected by caries.

- **Evaluation:**

S no	Techniques	Accuracy (%)	Mean processing time (msec)
1	Convolution Neural Network	78	1.234
2	Multi Layer Perceptron Neural Network	85	0.956
3	Hybrid Neural Network	96	0.732

Table 2.5: Comparaison of Performance Results

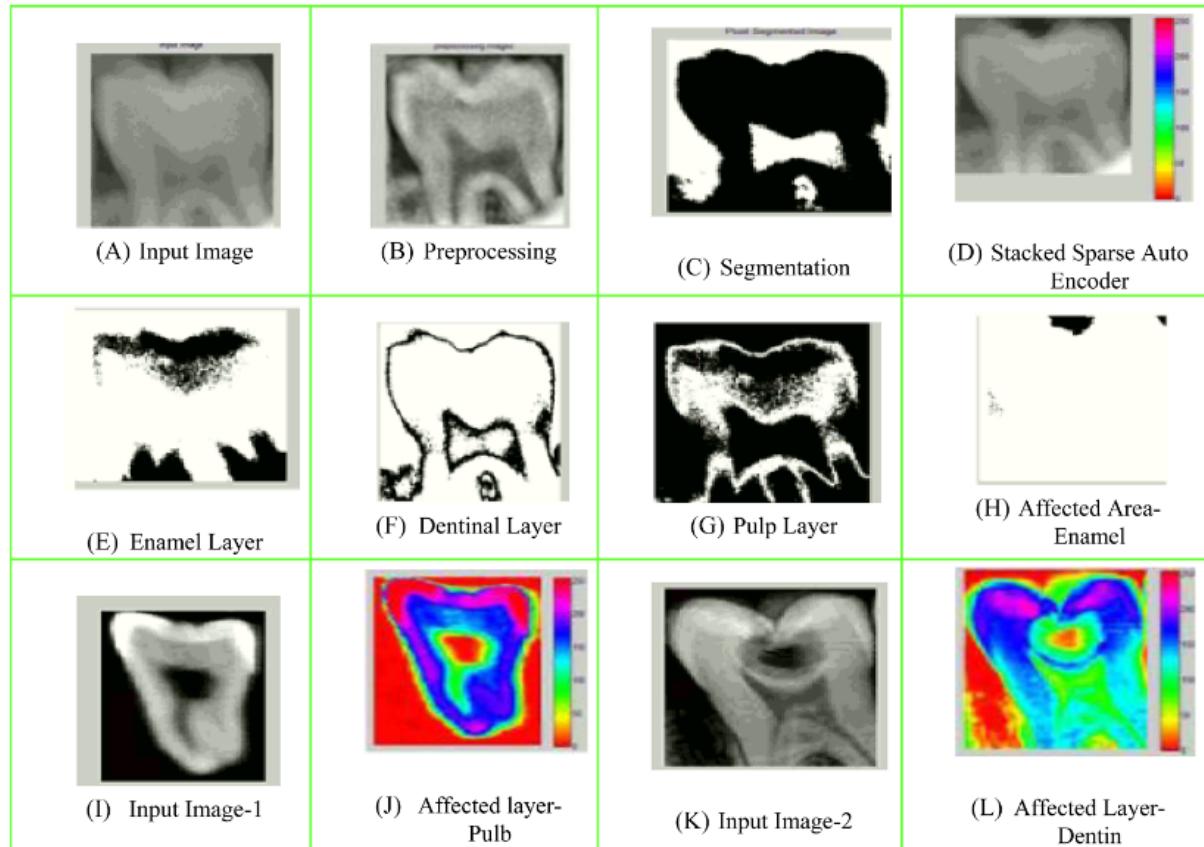


Figure 2.6: Simulation Process

3.3.2 Results and Discussion

Eighty images are specified in the dataset. There are 40 images in the training dataset and 40 in the test dataset.

The final dataset contains 480 tooth images captured, 300 images for training, and 180 for testing. About 173 images are accurately classified and recognized layer by layer, and accurate results are obtained.

The output of the simulation process (Figure 2.6), segmented images, and caries images are shown. (A) (I) (K) Input tooth images. (B) Preprocessing the image helps to remove noise. (C) Pixel-based segmentation of the image by hiding the background image. (D) Feature extraction image - this facilitates extraction of pixel values based on outliers. (E)(F)(G) Three layers, i.e., enamel, dentin, and pulp (H) The area affected by caries - Enamel. (J) The area affected by caries - Pulp. (L) Affected area - Dentin.

As a consequence of Table 2.5, compared with traditional techniques such as CNN. and multi-layer perceptual neural network, the proposed HNN. technique takes the least time on average. Unlike other techniques, the proposed HNN. technique provides efficient and improvised performance.

Conclusion

In this chapter, we presented the specification of our problem and the requirements needed to achieve our goal. We also presented the research done before starting this project, which will benefit us in the next steps. We went through different approaches on how to detect and segment tooth decay.

The next chapter presents the first steps of our method, which are the image pre-processing steps for segmentation and detection of a decay.

3

Data Processing

Content

1	Data Understanding	31
1.1	Data Collection	31
1.1.1	Properties	32
1.1.2	Data Quality	32
1.2	Radiograph Disfavors	32
2	Data Summary	33
3	Data Cleaning	33
3.1	ROI. Definition	34
3.2	Image Enhancement	35
4	Data Construction	36
4.1	Jaw Partition	36
4.1.1	Integral Projection	37
4.1.2	Linking Consecutive Points	38
4.2	Localization of the Teeth Gap Valley	39
4.2.1	Neck Detection	39
4.2.2	Line Sum Intensities	40

4.2.3	Generating Borders	41
5	Data Formation	43
5.1	Infected Teeth Isolation	43
5.2	Teeth Segmentation	44
5.2.1	Find Contours	44
5.2.2	Annotation Reform	45
5.3	Dataset Health Check	46
6	Data Augmentation	47
6.1	Resize Images	48
6.2	Random Brightness and Contrast	48
6.3	Random Gamma	49
6.4	Sharpen	49
6.5	Flip	49
6.6	Affine	50
6.7	Grid Distortion	50
6.8	Random Sized Crop	51
6.9	Dataset Health Check	51

Introduction

According to the CRISP-DM process, data understanding and preparation is the lengthiest part of a data science project; thus, this section will be the most challenging.

This chapter will begin by describing the first steps of our project. The first stage is elaborated on understanding the dataset and the morphological properties. Next, we will carry out the correction of the data; after that, we will describe how we construct the data to extract the main features needed for segmentation. We will also discuss formatting and using the new data in the following steps. Finally, we are going to implement some image augmentation methods.

1 Data Understanding

The dataset consists of panoramic X-ray images from patients of different ages, genders, and health conditions([Figure 3.1](#)). These X-rays help to identify and analyze the limits of caries present in dental X-rays. Dental X-rays are used as input images for preprocessing stage.

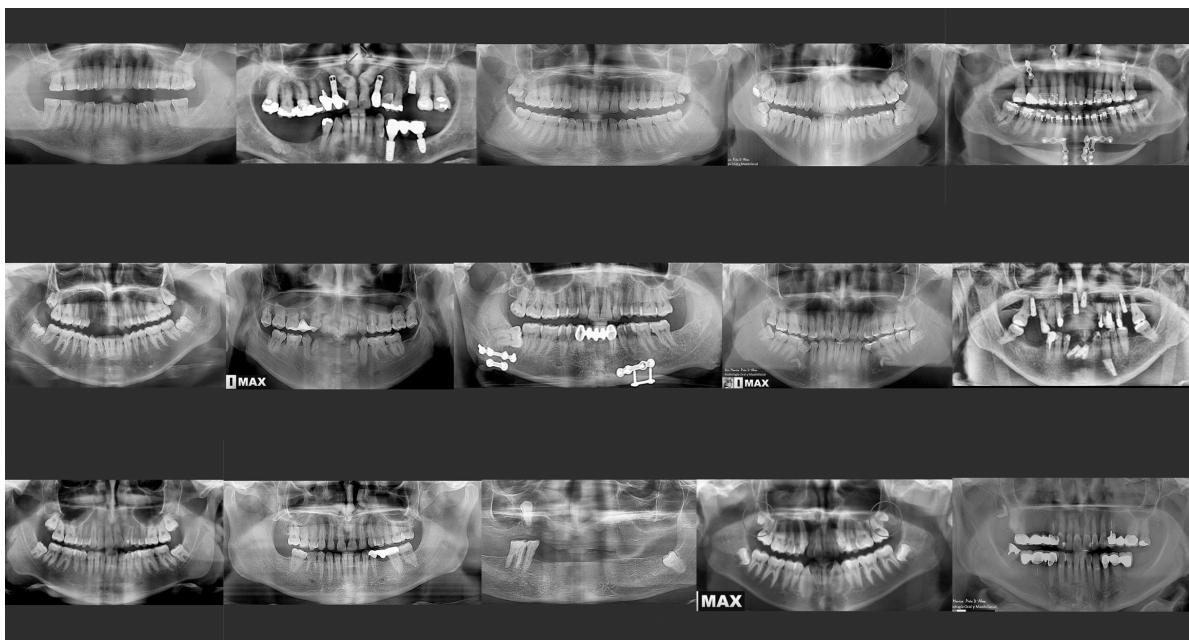


Figure 3.1: Dataset Preview

1.1 Data Collection

This section is where we determinate the necessary data content, formats, and sources for initial data collection.

The data set images were taken from different resources such as hospitals and private domains. Thus they were captured by other X-Ray machines.

1.1.1 Properties

There are 237 images in the dataset, with varying types of dental structure, sizes of the mouth, and the number of teeth per image, as seen in the [Figure 3.1](#)

- **Content:** Two hundred thirty-seven images accompanied with their annotation files and a text file dedicated to classes.
- **Image shape:** Each image has a 600 x 600 shape (height and width, respectively.)
- **Format:** All images are in **PNG** format with annotation text files in coco format.
- **Color Scale:** The gray levels of each image of our dataset were stretched to the [0, 255] scale.

1.1.2 Data Quality

The images are low-quality X-Rays, which may affect the processing of pixels in the wrong way. In addition, not just decays are mentioned in the annotation files, but prostheses too, and they will be removed afterward.

1.2 Radiograph Disfavors

When compared with different types of stomatology images, X-Ray images are very challenging for several reasons that increase their heterogeneity:

1. Different noise levels due to the moving imaging device capture a global perspective of the patient's mouth.
2. Low contrast, topological and morphological features are very complex, whether a global or local area of an image.
3. Blur and random objects within the image make it challenging to see edges easily (example [Figure 3.2](#)).
4. The noise from the spinal column covering the front teeth in some pictures is shown in the image (example [Figure 3.3](#)).

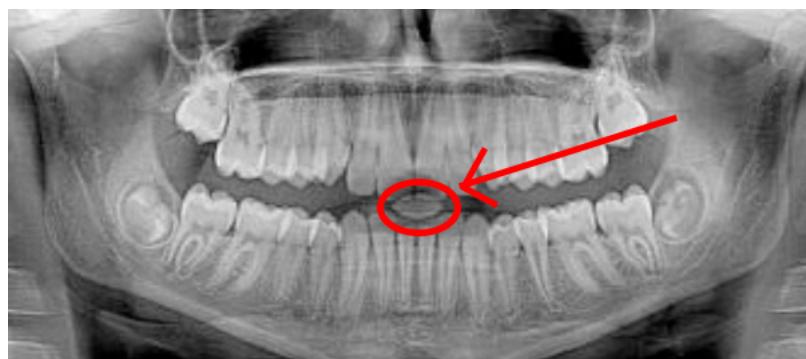


Figure 3.3: Spinal-column Cover Example (in the red circle)

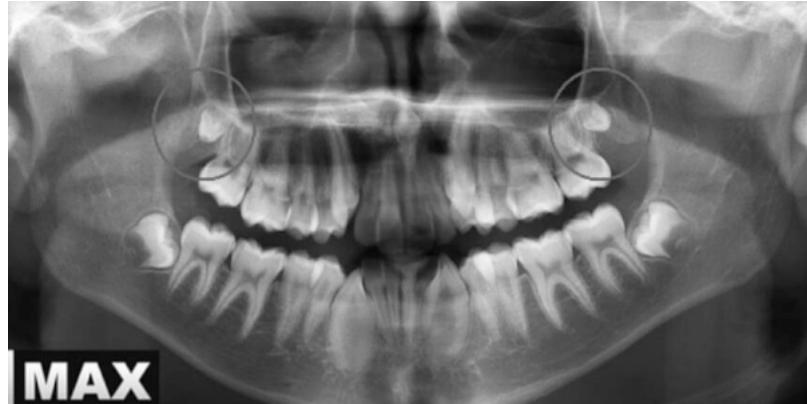


Figure 3.2: Blurry X-Ray Example

2 Data Summary

Our primary goal in this part is to demonstrate the main statistical features of our image dataset, which consists of 237 grayscale images. Note that the value is the average corresponding to all images, which is i.e., The mean intensity shown in the table 3.1 corresponds to the mean intensity of all images.

Since we can observe that all images present in the dataset have a low mean intensity value, this means that, in general, the intensity of each image tends to be darker. This is also confirmed by the median values shown in table 3.1. The maximum and minimum values indicate diversity in almost all possible grayscale values.

Statistic Data	Values
Minimum intensity value	0
Maximum intensity value	252.1335
Mean intensity value	135.0296
Mode	240.7918
Variance	5227.9080
Standard deviation	72.2699
Median	138.7947

Table 3.1: Statistic data retrieve of our images dataset

3 Data Cleaning

This section begins by describing the first steps in data preparation.

The first stage analyzes the distance from the center of the image to the four corners of the mouth; this creates a region of interest (**ROI**). In the second stage, we describe the image enhancement techniques used.

In the upcoming parts of this section, we will take this Figure 3.4 as a sample for the processing display.

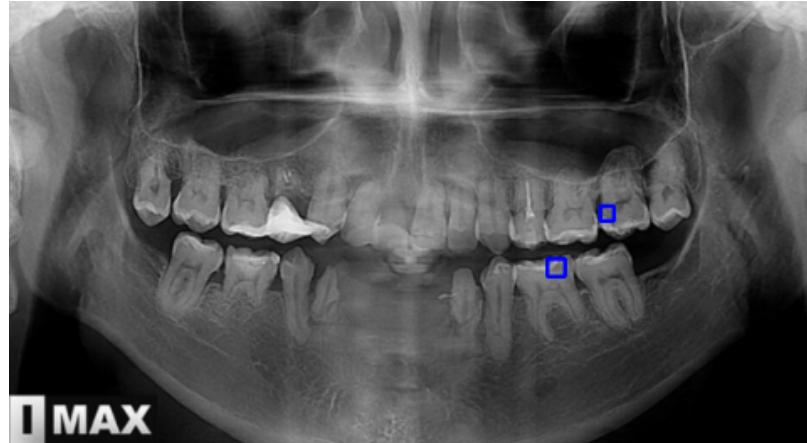


Figure 3.4: Image Sample from the Data Set Annotated with Decay Bounding Boxes

3.1 ROI. Definition

The earliest stages are based on a statistical analysis of the size and location of similar dataset in a research paper [29] to define initial regions of interest.

This eliminates useless information from the nose and jaw bones. Our goal is to crop the area that includes the entire mouth and remove as much noise as possible. For each data set image, we measured four distances (R_1 , R_2 , R_3 , R_4) from the image center (x_{center} , y_{center}) = ($w/2 = 300$, $h/2 = 300$) as shown in Figure 3.5.

Therefore, Using these values from analyzing this data set images, we get:

$$R_1 \approx 897.77, R_2 \approx 863.36, R_3 \approx 406.31, R_4 \approx 471.27$$

The images were cropped to 188×374 pixels containing only teeth and jaws.

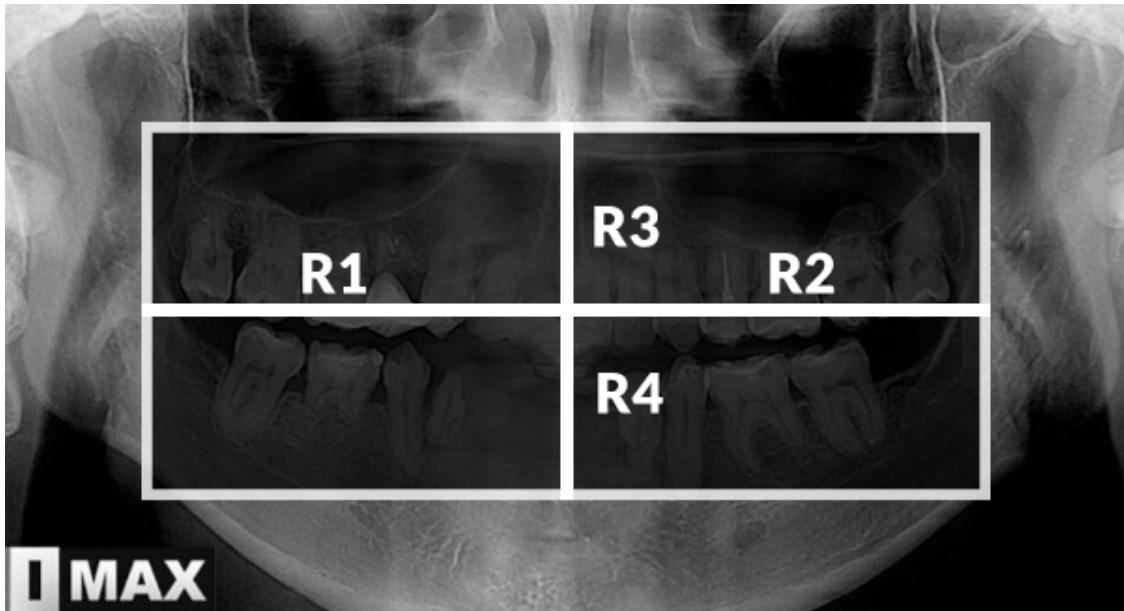


Figure 3.5: The Extracted Lengths from the Research Study

Consequently, borders ensure that slightly different images are appropriately cropped, even at the edges of small useless areas. The [Figure 3.6](#) illustrates the final results of this stage.

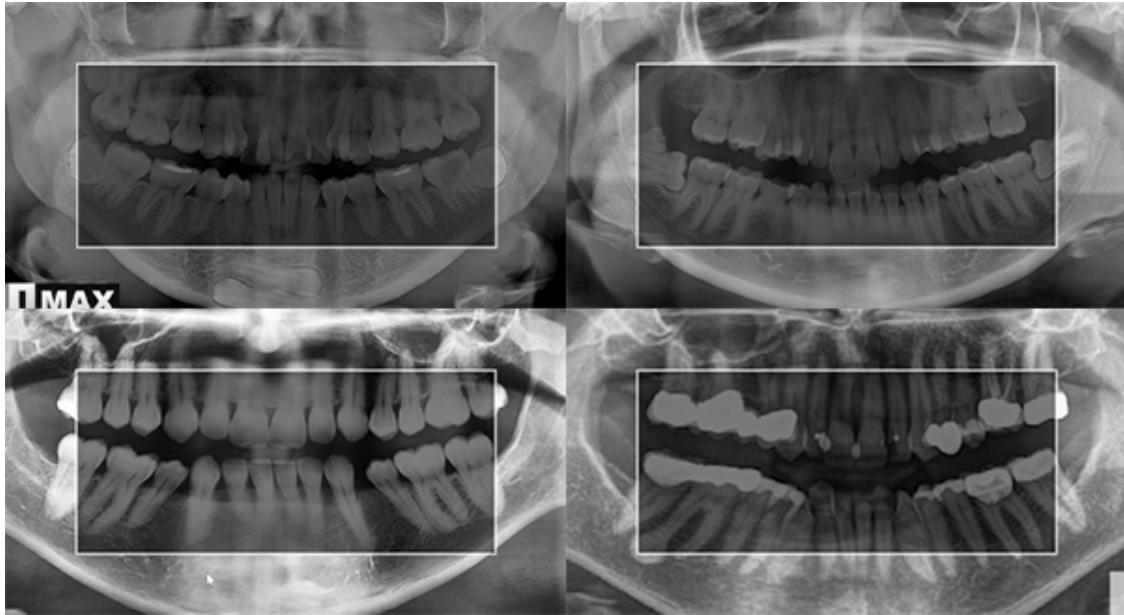


Figure 3.6: Result Examples of the ROI. Definition

3.2 Image Enhancement

In this part, we will discuss image quality optimization.

This optimization is achieved by implementing image enhancement techniques to adjust the brightness of images and improve overall image quality. A preliminary cleaning process is implemented to ensure the best results in image enhancement to remove any anomalies that appear in the images due to the X-ray process.

After removing all inorganic structures except dental fillings, the image contrast is increased to define the tooth structure better.

To preserve feature details, denoising through blur filters is avoided. According to the study by Yoonet et al. [16], who suggested that adaptive histogram equalization provides more remarkable contrast improvement for computer vision, implemented a combination of median filter and histogram equalization shown in [Figure 3.7](#).

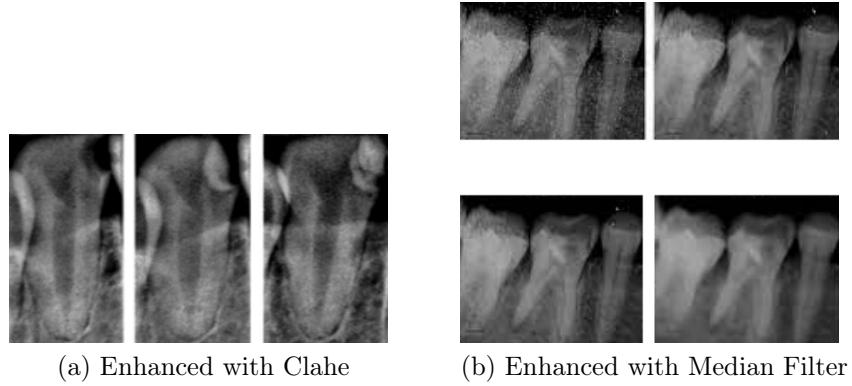


Figure 3.7: Applied Enhancements

After applying these adjustments, we get the enhanced panoramic image as shown in the Figure 3.8.



Figure 3.8: The Final Result after Enhancement

4 Data Construction

In this part, we will attempt to transform the data to be processed afterward. It is a crucial step to combine and reformat data for our interest.

We will mainly discuss separating the upper and lower jaws at this stage. Second, we will detect the gaps between the teeth in order to segment the infected teeth in the last part.

4.1 Jaw Partition

The first step is determining the dividing line between the upper and lower jaws. We will use the same algorithm described in [30].

4.1.1 Integral Projection

In order to automate the segmentation procedure, the user is not prompted for an initial segmentation point but is chosen by the horizontal integral projection with the lowest value, which is also near the center of the image, (usually between 40% and 60% of its height).

Integral projection can analyze pixel intensities across an image and detect regions of darker pixels. As such, it provided the best solution for detecting gaps between teeth, where the spaces between two adjacent teeth are easily identifiable from the thresholded mask obtained in the previous stage. Areas where black pixels were present between pairs of adjacent clusters of white pixels were identified as valleys.

Because the jaw form an arc on the image, rather than using the full horizontal line that would pass through the teeth farther away from the incisors, only a few pixels closest to the previously selected anterior space were used to define the calculated projection. After that, the algorithm continues to use the width of the image. The set of points is obtained based on the horizontal projection $v(u)$ of the image, given by [Equation 3.1](#), where $I(x, i)$ represents the intensity values in row x and column i :

$$v(u) = \sum_{i=0}^w I(x, i) \quad (3.1)$$

The starting point is defined at the right edge of the image and at the line with the minor $v(u)$ value given by [Equation 3.2](#), where w is the image width, as shown in [Figure 3.9](#).

$$p_0(x_0, w - 1) = \operatorname{argmin}_x(v(u)) \quad (3.2)$$

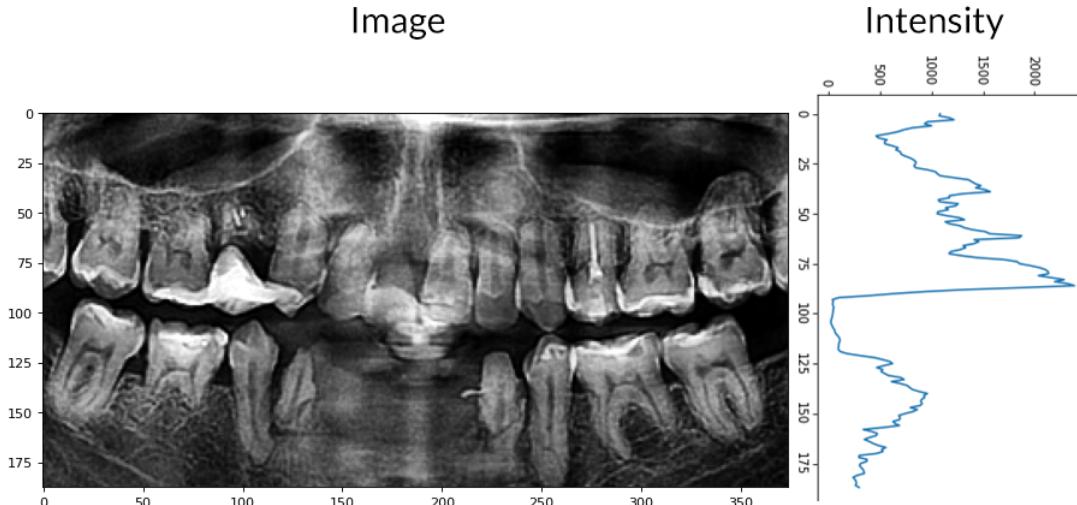


Figure 3.9: Horizontal Projection of the Intensities for the Starting Point

4.1.2 Linking Consecutive Points

The remaining points p_i are regularly spaced from $p_0 : p_i(x_i, (w - 1) - W/21)$, where x_i is obtained similarly to x_0 . To avoid excessive vertical distance between consecutive p_i , we add the following constraints in the [Equation 3.3](#).

$$p_i(x_i, y_i) = \begin{cases} p_i(x_{i+1} + T, y_i), & |p_i(x_i, y_i) - p_{i+1}(x_{i+1}, y_{i+1})| > T \\ p_i(x_i, y_i), & \text{otherwise} \end{cases} \quad (3.3)$$

The output of the described algorithm using Horizontal Integral Projection is shown in the [Figure 3.10](#) below.

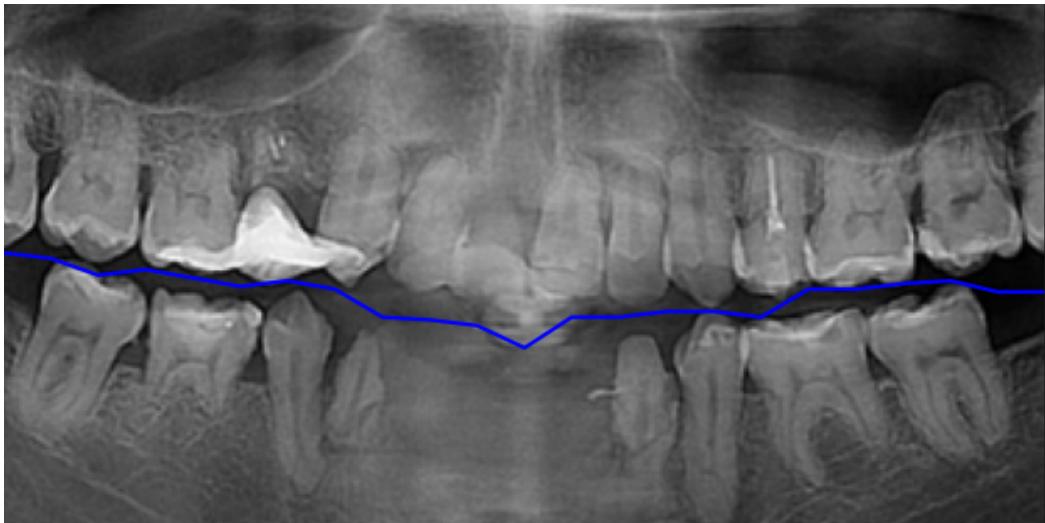


Figure 3.10: Initial Partition

As a rule of thumb, T is the threshold we define to avoid high vertical distances with a value of $T = 18$. We observed that this step plays an essential role in dealing with missing teeth.

Using a set of $p_i(x_i, y_i)$ primitive points, the spacing of the jaws is given by the 10^{th} order polynomial given in [Equation 3.4](#), obtained by the polynomial least squares fitting algorithm.

$$p(x) = a_0 + \dots + a_{10}x^{10} \quad (3.4)$$

If we do not subdivide the image into small strips, the algorithm will depend on the size of the missing teeth in the input image.

When teeth are missing and neighbors are present at the same time, our method is error-prone because the points we extract tend to follow darker areas of the image. Sometimes areas with missing teeth are darker compared to the gap between the jaws, and it is missing areas. After a large missing teeth area, in some cases, this method may overcome the tooth's presence. So we extracted our points from the strips.

In return, however, this may make the method more sensitive to small changes in intensity in the image, making it difficult to separate the jaws.

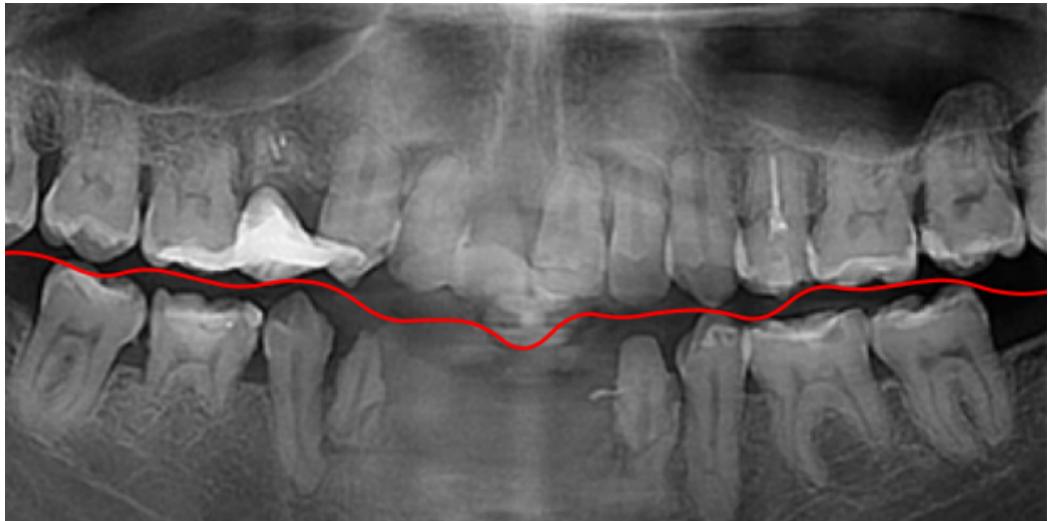


Figure 3.11: Results of the Polynomial Least Squares Fitting

4.2 Localization of the Teeth Gap Valley

4.2.1 Neck Detection

The obtained curve is then used to estimate the position of the neck of each tooth, the end of the root, and the part of the tooth where crown and enamel formation began (Shown in Figure 3.12).

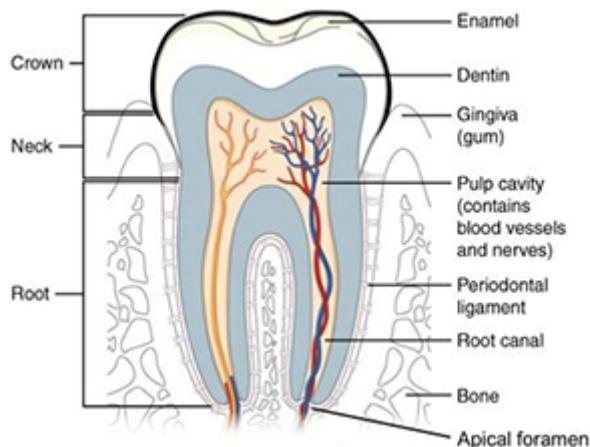


Figure 3.12: Tooth Cross-sectional Anatomy [1]

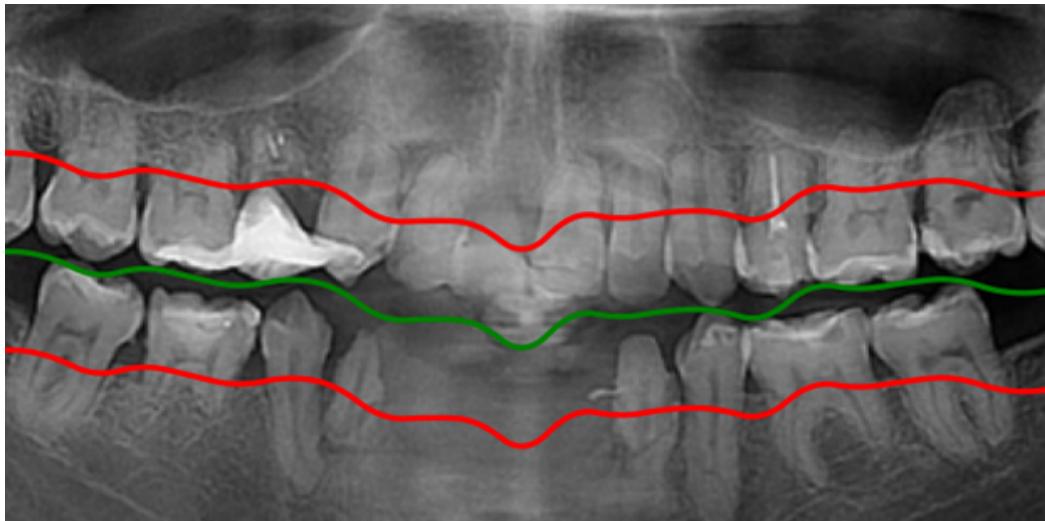
While the crowns of individual teeth tend to lock together and the roots are difficult to separate from the underlying bone, the area between the necks of two adjacent teeth is clear enough to be easily found on the panorama.

Since the neck is usually flush with the pulp, darker than the surrounding teeth, the easiest and quickest way to find a line through the neck is to add the vertical displacement of the curve between the upper and lower jaws.

After that, we sum the intensities of pixels the line passes through for every translation and select pixels whose values drop significantly, indicating that the line passes through

darker pulp areas.

This is similar to the integral projection method, but the sum is not calculated along a horizontal or vertical line but along a curve that has the same shape as the spline separating the upper and lower jaws and is translated vertically. The sample image curves calculated in this way are shown in [Figure 3.13](#).



[Figure 3.13](#): The Splines passing through the dental pulp on which the necks can be found (red lines) and the spline separating upper and lower jaw (green line)

In order to make sure neither the original gap between jaws nor a gap between the roots of teeth and the edge of the image are selected instead of the desired dental pulp line, the vertical translation range should be selected so that the search can be performed on a limited area, too far from the separating jaw line. Translations that are near and too far are automatically discarded.

4.2.2 Line Sum Intensities

The next step is the selection of points on each curve representing a gap between the necks of two adjacent teeth.

The resulting image of the radiographic sample is shown in [Figure 3.4](#). For the maxilla (upper jaw) and mandible (lower jaw), we stored a series of values containing the intensities of the points conforming to the splines passing through the respective dental pulps.

The spikes on the graph of these values represent dark spots surrounded by bright areas, indicating gaps between the tooth necks ([Figure 3.14](#)).

In order to remove the false spikes (also known as noise), the values are Gaussian filtered to smooth the function ([Figure 3.15](#)). This algorithm is applied to all of the curve functions and they are shown on [Figure 3.16](#) and [Figure 3.17](#).

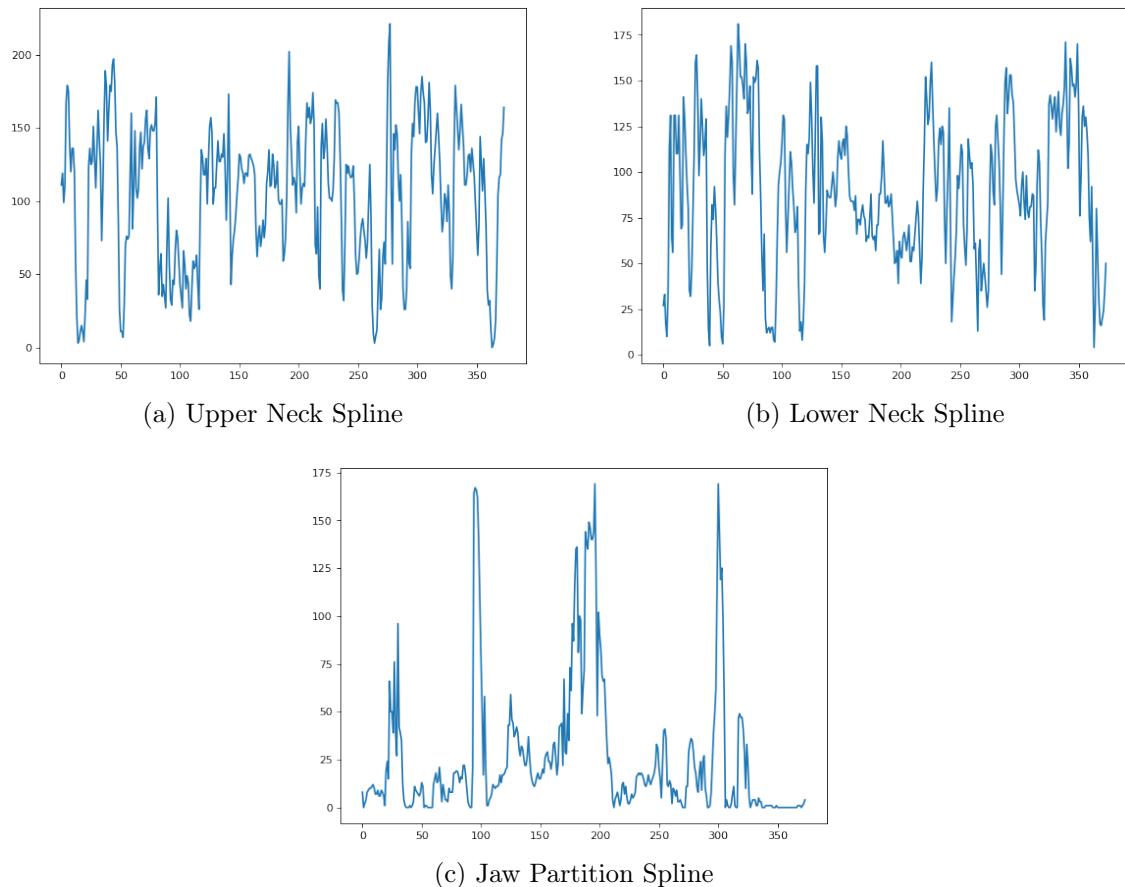


Figure 3.14: Line Sum Intensities Applied on the three Splines

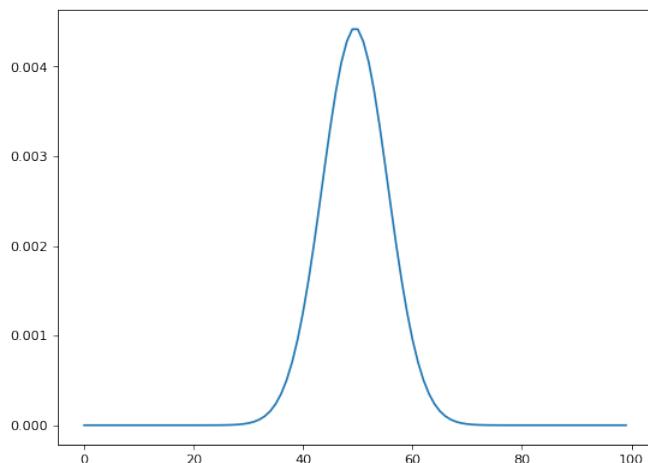


Figure 3.15: Gaussian Filter used to Reduce Noise

4.2.3 Generating Borders

The next step is to extract the points from the splines where gaps may be present and form a subset for each spline. To select the proper spike among these values, for every plot, we search for local minimum values along the image width.

For the last step, every value in the lower and upper subsets, its closest value from

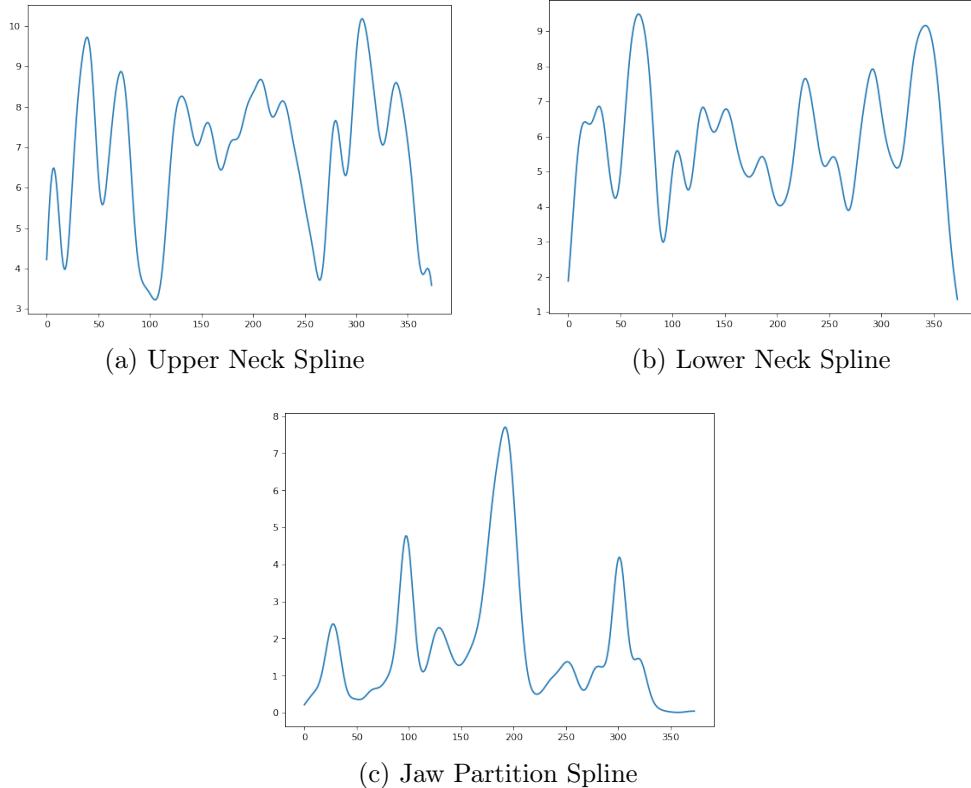


Figure 3.16: Filtered Line Sum Intensities

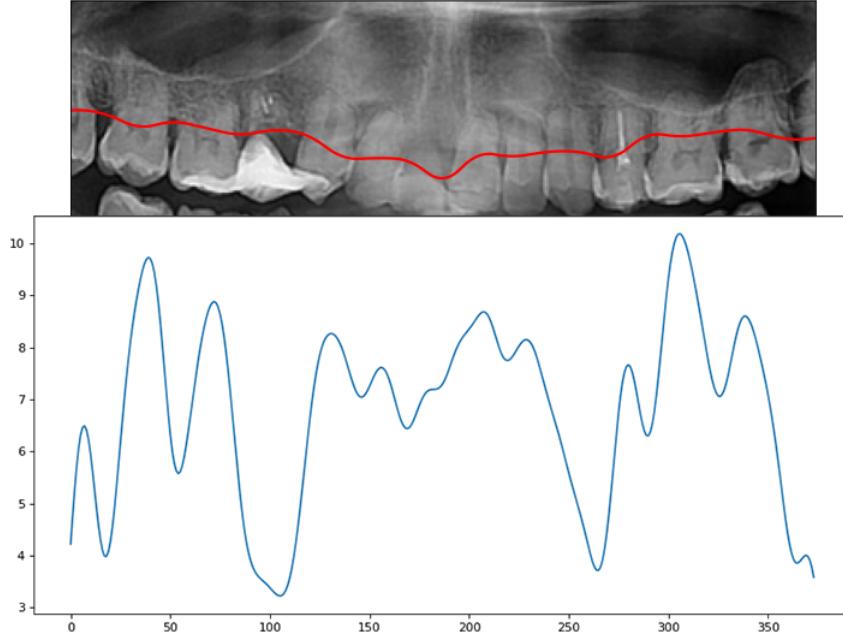


Figure 3.17: The Upper Jaw Curve with its respective intensity values

the gap subset, is linked to it, and the results are saved to form a dividing function. This function is then extended to the edges of the image.

The disadvantage of this method of finding the second separation point is that the root of the tooth must be sufficiently distinguished from the background and adjacent

teeth, and using this algorithm for incisors and canines often gives incorrect results. The separating lines acquired using this algorithm on a sample radiograph are presented on Figure 3.18.

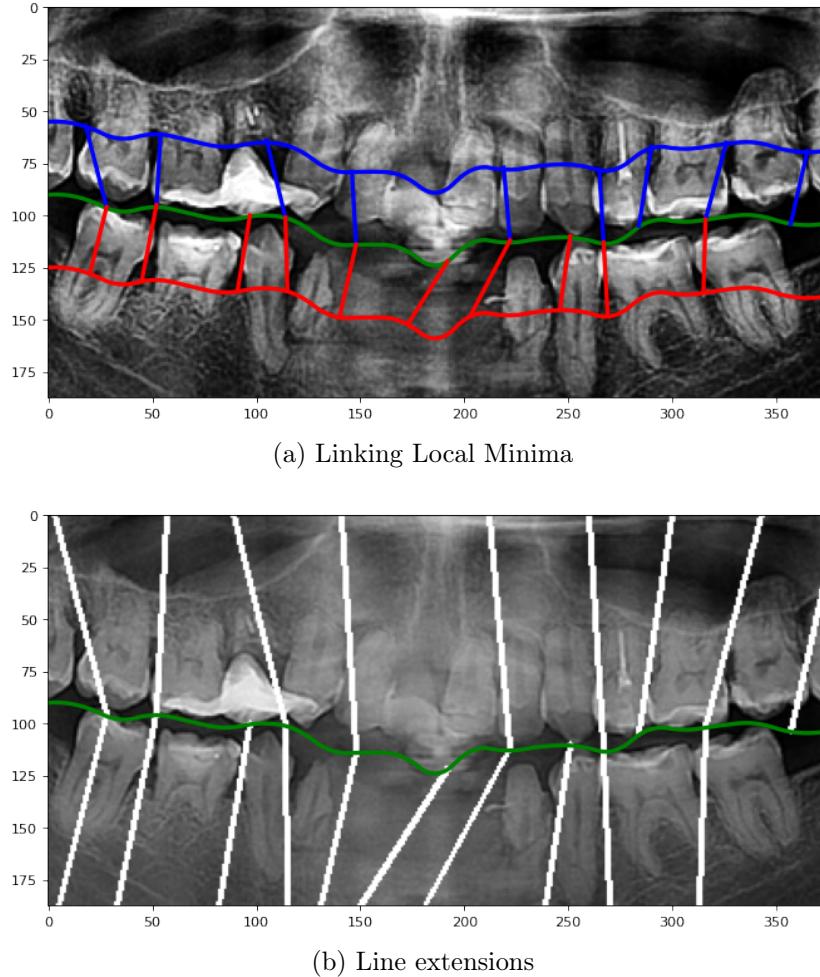


Figure 3.18: Generated Separation Lines

5 Data Formation

Reformatting data is necessary to extract useful features, which we will discuss in this section as well as creating automatic segmentation algorithm.

5.1 Infected Teeth Isolation

We perform single tooth isolation by looping on separation lines and searching for the proper position of the decay to isolate the infected tooth within its borders.

Within the sample X-Ray, We have two caries, one on the upper side and the second one is in the lower side; as a result, we will have two newly generated images. Hence, the isolation step is illustrated in Figure 3.22.

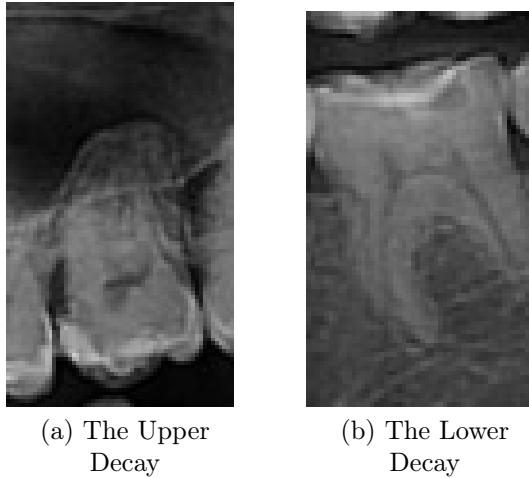


Figure 3.19: Generated Images From Original Sample

5.2 Teeth Segmentation

The Final step of this chapter is to build the silhouette of the decay in the bounding boxes and store the data in an annotation file. we will start the process from [Figure 3.20](#).

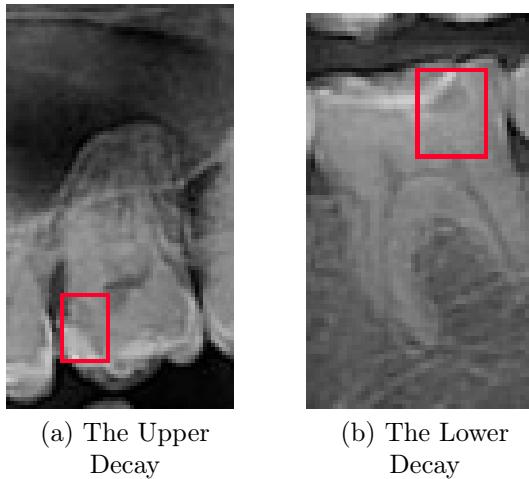


Figure 3.20: The Last Updated Bounding Boxes

5.2.1 Find Contours

Due to contours, we will be able to segment the decay automatically. The Finding Contours technique is one of the most essential and used algorithms in computer vision problems. The primary motivation for using this algorithm is its excellent ability to segment objects in the images.

Based on this work, we choose the finding contours method applied within the bounding boxes where decay is identified to optimize and focus on the region of interest, the decay region ([Figure 3.21](#)).

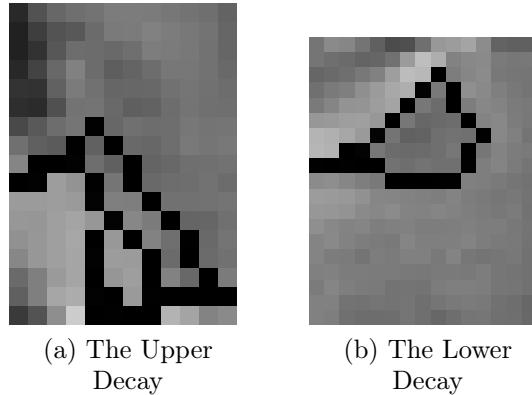


Figure 3.21: Contours Finding in the Decay Area

In the [Figure 3.22](#) is the results of this step concerning the lower and upper decay in our example.

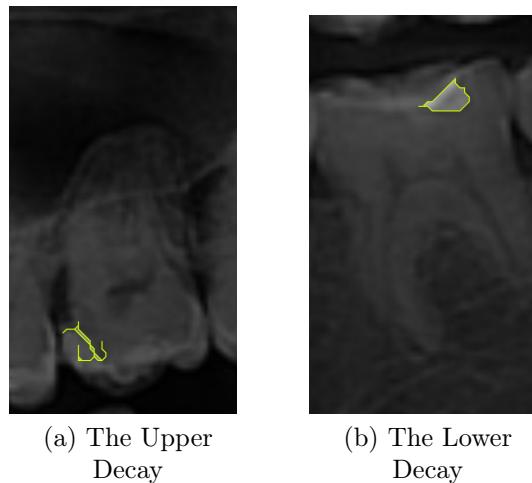
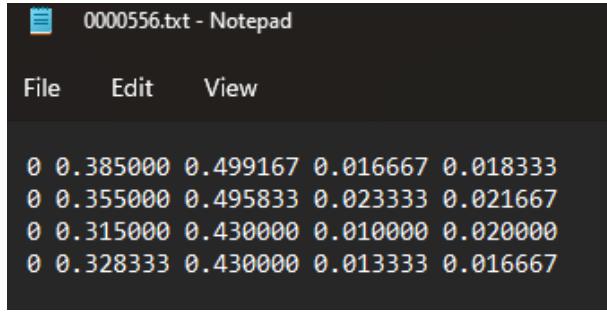


Figure 3.22: Teeth with Segmented Decay

5.2.2 Annotation Reform

As we can observe in the last part, we have been editing the position of the anomalies in every step that requires cropping or editing image dimensions (see [Figure 3.20](#)).

Before the previous processing steps, we had one text file for each image describing the Class (0: Decay Class) and coordinates ($x, y, width, height$) of the bounding boxes present in the related image. However, we will reform it to one **JSON**. file containing the annotations of all the segmented images. Therefore we have these files described in [Figure 3.23](#).



```
0 0.385000 0.499167 0.016667 0.018333
0 0.355000 0.495833 0.023333 0.021667
0 0.315000 0.430000 0.010000 0.020000
0 0.328333 0.430000 0.013333 0.016667
```

(a) Text File Annotation Before Changes

```
1 {
2   "0000561_1.jpg285":
3   {
4     "filename": "0000561_1.jpg",
5     "size": 285,
6     "regions": [
7       {
8         "shape_attributes": {
9           "name": "polygon",
10          "all_points_x": [32, 31, 30, 29, 28, 27, 26, 25, 24, 25, 26, 27,
11          "all_points_y": [14, 15, 16, 17, 18, 19, 19, 20, 20, 20, 20, 21
12          "region_attributes": {
13            "class": "decay"
14          }
15        ],
16        "file_attributes": {}
17      },
18      "0000561_2.jpg204":
19      {
20        "filename": "0000561_2.jpg",
21        "size": 204,
```

(b) The Final JSON File Annotation

Figure 3.23: Annotation Files

5.3 Dataset Health Check

After the processing done in the previous part, we obtained 222 images filtered from 237 original X-Ray images. The median image ratio is 79x97 (width x height). We only have one class which is decay. Along with all the data, we acquired 222 caries, one in each image.

- **Dimension Insights:** The size distribution is described in the [Table 3.12](#) and in the [Figure 3.24](#). Besides, the aspect ration distribution features three main categories: (tall, square and wide) as described [Table 3.13](#)

tiny	186 (< 100x100)
small	36

Table 3.2: Size Distribution extracted from the previous processing

- **Annotation Heatmap:**

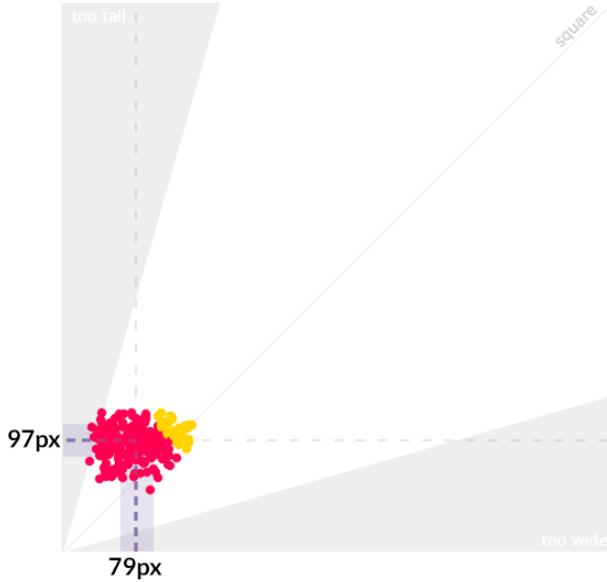


Figure 3.24: Size Distribution extracted from the previous processing

tall	190
square	17
wide	15

Table 3.3: Aspect Ratio extracted from the previous processing

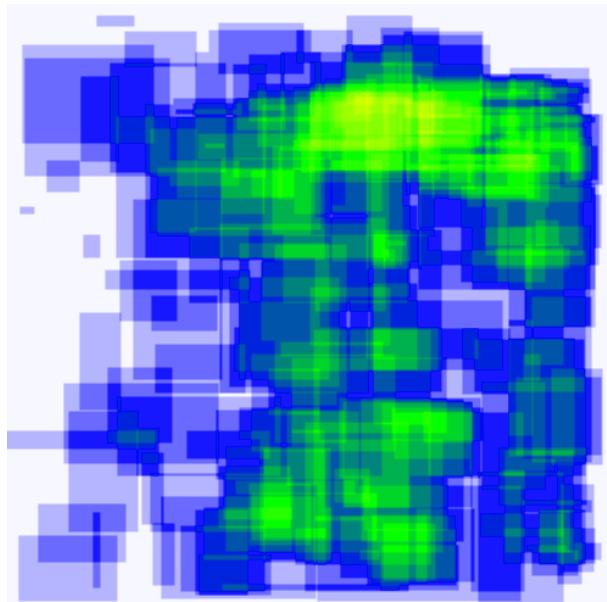


Figure 3.25: Annotation Heatmap extracted from the previous processing

6 Data Augmentation

Due to our small dataset after the modifications applied in the previous section, we need to make minor alterations to our existing dataset. This augmentation process will help to increase the amount of relevant data in our dataset. In this project, we will use a

famous python library called Albumentations.

Albumentations is a Python package for image augmentation. Image augmentation is used in deep learning and computer vision tasks to improve the quality of trained models. The goal of image augmentation is to create new training samples from the existing data [31].

Why Albumentations?

- Albumentations supports all everyday computer vision tasks such as classification, semantic segmentation, instance segmentation, object detection, and pose estimation.
- The library contains over 70 different extensions to generate new training samples from existing data.
- It works with popular deep learning frameworks like PyTorch and TensorFlow. By the way, Albums are part of the PyTorch ecosystem.
- The library is widely utilized in industry, deep learning study, machine learning competitions, and open-source projects.

6.1 Resize Images

The first step to do is to resize all the images to a specific size obtained during the health check analysis of the dataset, as explained in the [Equation 3.5](#).

$$size_{average} = \text{round}\left(\frac{height_{average} + width_{average}}{2}\right) \quad (3.5)$$

```
class albumentations.augmentations.geometric.resize.Resize (height=88,
    width=88, interpolation=1, always_apply=True, p=1)
```

Name	Type	Description
<i>height</i>	int	desired height of the output
<i>width</i>	int	desired width of the output.
<i>interpolation</i>	OpenCV flag	flag that is used to specify the interpolation algorithm.
<i>p</i>	float	probability of applying the transform.

Table 3.4: Parameters of the Resize Function

6.2 Random Brightness and Contrast

Randomly vary the brightness and contrast of the input image.

```
class albumentations.augmentations.transforms.RandomBrightnessContrast
    (brightness_limit=0.01, contrast_limit=0.1, brightness_by_max=True,
     always_apply=False, p=0.5)
```

Name	Type	Description
<i>brightness_limit</i>	[float, float] or float	factor range for changing brightness. If limit is a single float, the range will be (-limit, limit). Default: (-0.2, 0.2).
<i>contrast_limit</i>	[float, float] or float	factor range for changing contrast. If limit is a single float, the range will be (-limit, limit). Default: (-0.2, 0.2).
<i>brightness_by_max</i>	Boolean	If True adjust contrast by image dtype maximum, else adjust contrast by image mean.
<i>p</i>	float	probability of applying the transform.

Table 3.5: Parameters of the RandomBrightnessContrast Function

6.3 Random Gamma

It Controls the overall luminance of an image.

```
class albumentations.augmentations.transforms.RandomGamma (gamma_limit=(80, 120), eps=None, always_apply=False, p=0.6)
```

Name	Type	Description
<i>gamma_limit</i>	[float, float] or float	If <i>gamma_limit</i> is a single float value, the range will be (<i>-gamma_limit</i> , <i>gamma_limit</i>).
<i>eps</i>	[float, float] or float	Deprecated.
<i>p</i>	float	probability of applying the transform.

Table 3.6: Parameters of the RandomGamma Function

6.4 Sharpen

Sharpen the input image and overlays the output with the initial image.

```
class albumentations.augmentations.transforms.Sharpen (alpha=(0.2, 0.5), lightness=(0.5, 1.0), always_apply=False, p=0.5)
```

6.5 Flip

Flip the initial image horizontally, vertically, or horizontally and vertically.

```
class albumentations.augmentations.geometric.transforms.Flip (p=0.7)
```

Name	Type	Description
<i>alpha</i>	[float, float] or float	range to choose the visibility of the sharpened image. At 0, only the original image is visible, at 1.0 only its sharpened version is visible.
<i>eps</i>	[float, float] or float	range to choose the lightness of the sharpened image. Default: (0.5, 1.0).
<i>p</i>	float	probability of applying the transform.

Table 3.7: Parameters of the RandomGamma Function

Name	Type	Description
<i>p</i>	float	probability of applying the transform.

Table 3.8: Parameters of the Flip Function

6.6 Affine

Augmentation to use affine transformations to images. This augmentation is primarily a wrapper around the corresponding classes and functions in OpenCV.

```
class albumentations.augmentations.geometric.transforms.Affine (scale=None,
    translate_percent=0.2, translate_px=None, rotate=None, shear=0.3,
    interpolation=cv2.BORDER_REFLECT, mask_interpolation=0, cval=0,
    cval_mask=0, mode=0, fit_output=False, keep_ratio=False,
    always_apply=False, p=0.4)
```

Name	Type	Description
<i>translate_percent</i>	None, number, tuple of number or dict	Translation as a fraction of the image height/width (x-translation, y-translation)
<i>shear</i>	number, tuple of number or dict	Shear in degrees (NOT radians), i.e. expected value range is around [-360, 360], with reasonable values being in the range of [-45, 45]
<i>interpolation</i>	int	OpenCV interpolation flag.
<i>p</i>	float	probability of applying the transform.

Table 3.9: Parameters of the Affine Function

6.7 Grid Distortion

An image warping technique driven by the mapping between equivalent families of curves arranged in a grid structure.

```
class albumentations.augmentations.geometric.transforms.GridDistortion
    (num_steps=5, distort_limit=0.3, interpolation=1,
```

```
border_mode=cv2.BORDER_REFLECT, value=None, mask_value=None,
normalized=False, always_apply=False, p=0.7)
```

Name	Type	Description
<i>num_steps</i>	int	count of grid cells on each side.
<i>border_mode</i>	OpenCV flag	flag that is used to specify the interpolation algorithm.
<i>normalized</i>	bool	if true, distortion will be normalized to do not go outside the image.
<i>p</i>	float	probability of applying the transform.

Table 3.10: Parameters of the GridDistortion Function

6.8 Random Sized Crop

```
class albumentations.augmentations.crops.transforms.RandomSizedCrop
(min_max_height=[64, 80], height=80, width=80, w2h_ratio=1.0,
interpolation=1, always_apply=False, p=0.6)
```

Name	Type	Description
<i>min_max_height</i>	[int, int]	crop size limits.
<i>height</i>	int	height after crop and resize.
<i>width</i>	int	width after crop and resize.
<i>p</i>	float	probability of applying the transform.

Table 3.11: Parameters of the RandomSizedCrop Function

6.9 Dataset Health Check

After applying image augmentation techniques, we enlarged the dataset from 222 images to 1209 images with segmented decay, as shown in the [Figure 3.26](#).

Most of the images within our dataset are an 88x88 pixel shape. Therefore our median image ratio is a square equal to 88x88.

tiny	1173 (< 100x100)
small	36

Table 3.12: Size Distribution extracted after augmentation

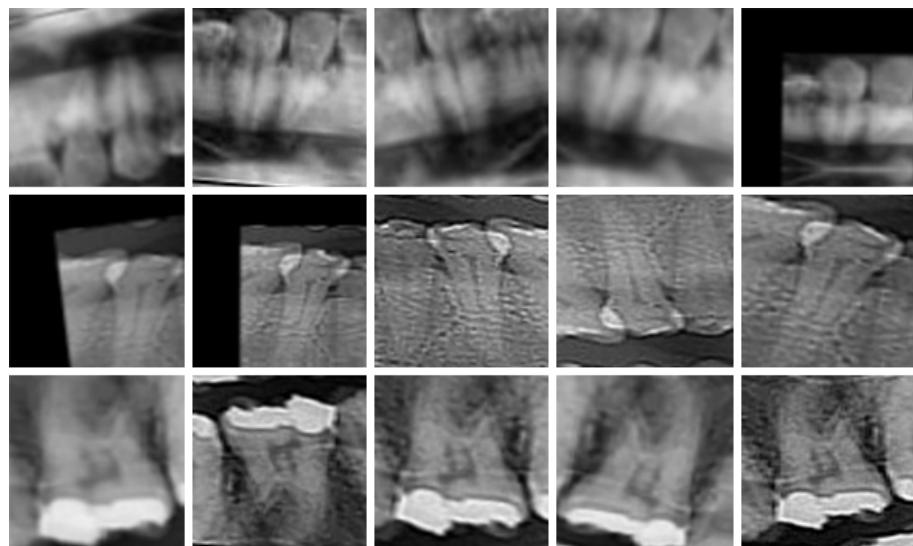


Figure 3.26: Data Augmentation samples

tall	190
square	1004
wide	15

Table 3.13: Aspect Ratio extracted after augmentation

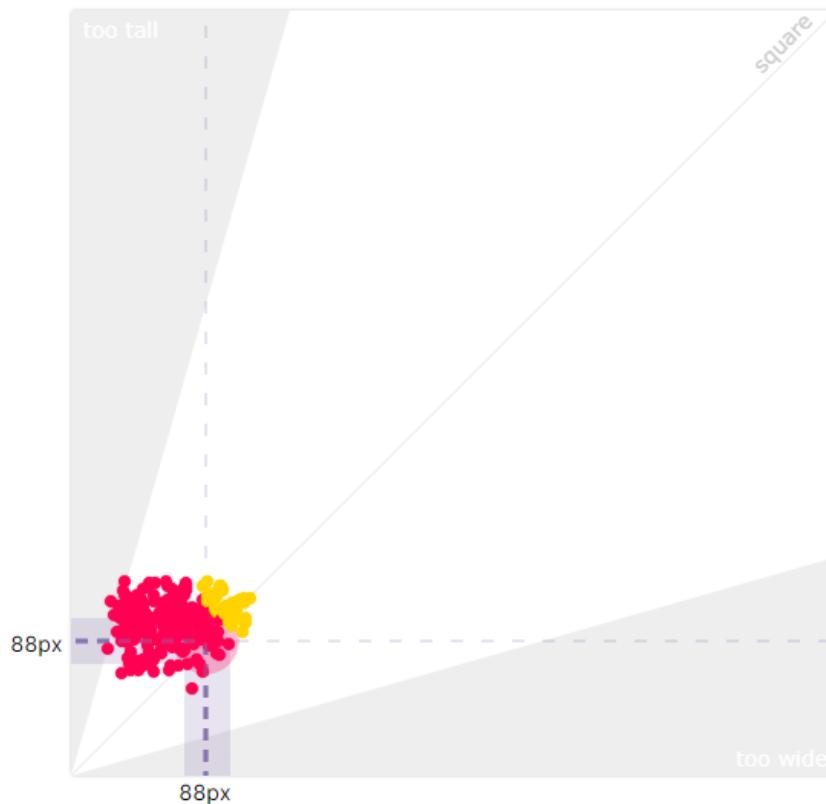


Figure 3.27: Size Distribution extracted after augmentation

The annotation heatmap described in the Figure 3.28.

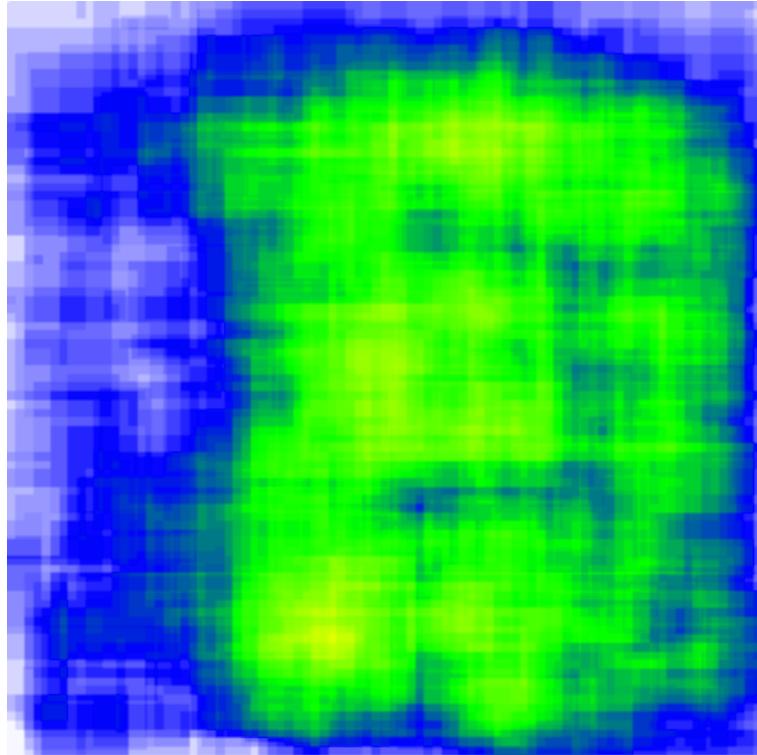


Figure 3.28: Annotation Heatmap After Image Augmentation

Conclusion

In Conclusion, in this chapter, we started with the properties of the data. Next, we introduced the first stage of our processing, which consists of defining the first region of interest, our ROI., by which most of the noise in the input image can be removed.

Besides, we applied numerous enhancements to the panoramic X-Ray. In the second step, we separated the dataset images' jaws using the horizontal integration projection method.

In the third step, we performed the teeth gap valley detection using line sum intensities and selecting the local minima points to extract the line borders. In addition, we applied the segmentation technique to the decay region after cropping this area. Indeed, we have produced a new annotation file for the dataset images.

Last but not least, we have implemented various data augmentation techniques to expand our dataset. In the next chapter, we will discuss our model and its evaluation.

4

Modeling and Evaluation

Content

1	Modeling	56
1.1	Instance Segmentation	56
1.2	Detectron2 - Mask R-CNN	57
1.2.1	Network Architecture	58
1.2.2	Results	58
1.3	Test Design Generation	60
1.4	Model Build	61
1.4.1	Architecture of Detectron2	61
1.4.2	Tools and Packages	62
1.4.3	Load and Register the Dataset	64
1.4.4	Customize Configurations	65
1.4.5	Customize Default Trainer	66
1.4.6	Train the Model	68
1.4.7	Prediction and Visualization	68
1.5	Model Assessment	69
1.5.1	Initial Results	70

1.5.2	Intermediate Results	74
1.5.3	Final Results	78
2	Evaluation	82
2.1	Results	82
2.2	Next Step	82

Introduction

Since the advent of deep learning, we have developed our data processing and decision-making tools to levels comparable to, and even exceeding, human capabilities in some circumstances.

In particular, in this chapter, we will discuss the training architecture we chose to train, then we will talk about the extracted results from the training and the predictions. Before that, we will clarify how the selected architecture works and the code implemented during the training process. Altogether, we will evaluate the approved model by the key metrics and the prediction results performed on the test dataset.

1 Modeling

1.1 Instance Segmentation

Object detection builds a bounding box around items of interest and classifies them. Segmentation extends object detection by outlining the object of interest. As a result, modeling is significantly more challenging, and segmentation should be utilized only when a downstream application requires the precise contour of the object.

What is the difference between semantic and instance segmentation?

Semantic segmentation assigns a class name to each pixel of an image, such as a person, flower, or automobile. It considers several objects of the same class to be one entity. On the other hand, instance segmentation considers numerous objects of the same category as different individual instances.

Why instance segmentation?

Object segmentation is needed for further implementation and advanced development, for example, decay classification. We need the masks predicted by instance segmentation to indicate the category of each decay in a tooth.

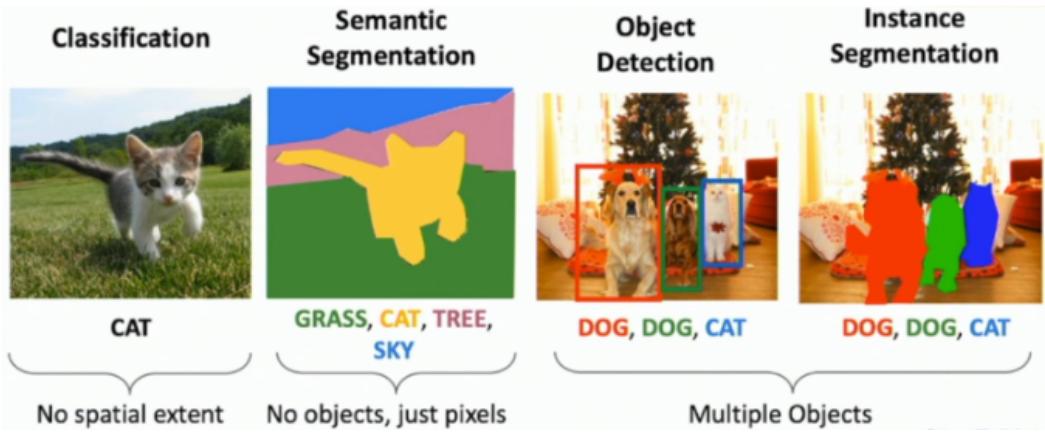


Figure 4.1: The Comparison between image classification, sementatic segmentation, object detection and instance segmentation [2]

1.2 Detectron2 - Mask R-CNN

Detectron2 is the next-generation library from Facebook AI Research that delivers state-of-the-art detection and segmentation algorithms. It is the replacement for Detectron and the Mask R-CNN benchmark. It contributes to various computer vision research projects and Facebook production apps.

The premise of Mask R-CNN is straightforward: Faster R-CNN provides two outputs for each candidate item, a class label, and a bounding-box offset; we add a third branch that outputs the object mask to this. Thus, Mask R-CNN is a straightforward and obvious concept. However, the additional mask output differs from the class and box outputs, necessitating the extraction of an item's more acceptable spatial arrangement. The fundamental parts of Mask R-CNN are then introduced, including pixel-to-pixel alignment, which is the main missing piece in Fast/Faster R-CNN [3].

Parameters	FLOPs	File Size
55 Millions	937 Billion	210.10 MB
Architecture		
Convolution, ROIAlign, Softmax, RPN, Dense Connections, ResNet		
Training Data	Lr Sched	Training Resources
COCO	3x	8 NVIDIA V100 GPUs
Training Time No	Max Iter	FLOPs Input No
2.04 days	270000	100
Backbone Layers	Train Time (s/iter)	
101	0.652	
Training Memory (GB)	Inference Time (s/im)	
6.3	0.145	

Table 4.1: Mask R-CNN ResNet101-C4, 3x model with detectron2 [4]

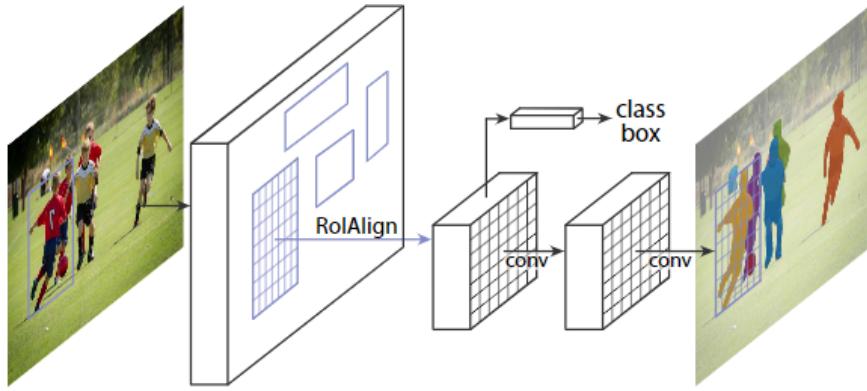


Figure 4.2: The **Mask R-CNN** framework for instance segmentation

1.2.1 Network Architecture

Mask R-CNN is a cutting-edge segmentation model built on top of Faster R-CNN. Faster R-CNN is a region-based convolutional neural network that outputs bounding boxes, class labels for each item, and a confidence score. To understand Mask R-CNN, first consider the design of Faster R-CNN, which operates in two stages:

- **Stage 1:** The initial stage includes a backbone network (ResNet, VGG, Inception...) and a region proposal network. These networks run once per image to give a set of region proposals. Region proposals are regions in the feature map which contain the object.
- **Stage 2:** In this step, the network predicts bounding boxes and object classes for each proposed region generated in stage 1. Each proposed region can vary, but fully connected layers in networks require a set-size vector to make predictions. The size of these proposed regions is determined by either the ROI pool (similar to MaxPooling) or the ROIAlign method.

1.2.2 Results

Mask R-CNN outperforms baseline variations of earlier state-of-the-art models in all instantiations, and this comprises the winners of the COCO 2015 and 2016 segmentation challenges, MNC and FCIS. Mask R-CNN with ResNet-101-FPN backbone beats FCIS++, which features multi-scale train/test, horizontal flip test, and online complex example mining without bells and whistles (OHEM). [Figure 4.4](#) shows a comparison of our Mask R-CNN baseline and FCIS++. FCIS++ displays systematic artifacts on overlapping instances, implying that the underlying problem of instance segmentation constrains it. There are no such artifacts visible in Mask R-CNN.

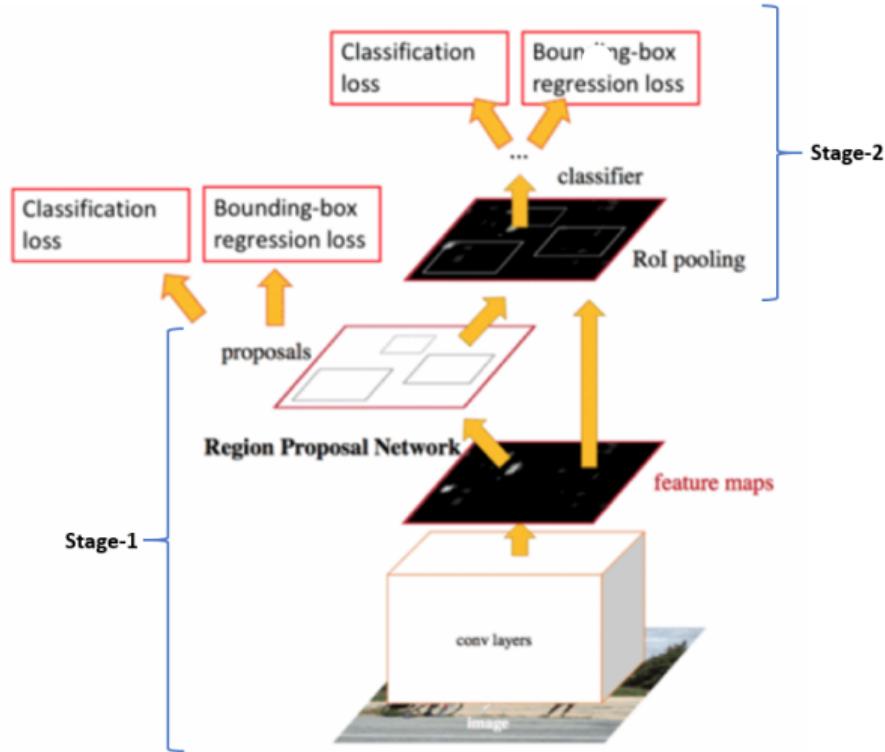


Figure 4.3: Faster R-CNN, a single unified network for object detection [3]

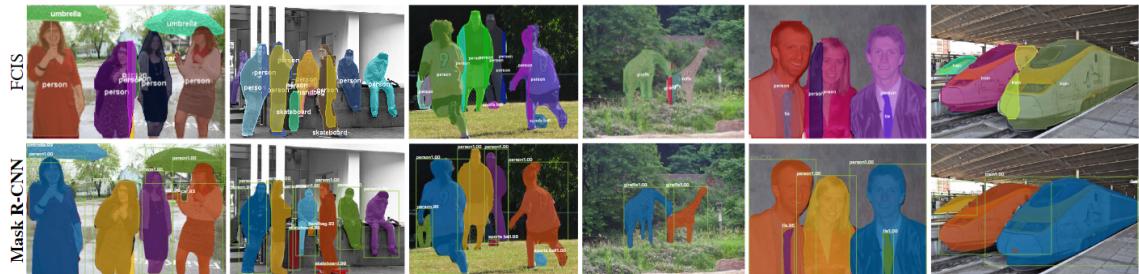


Figure 4.4: FCIS+++ (top) vs. Mask R-CNN (bottom, ResNet-101-FPN). FCIS exhibits systematic artifacts on overlapping objects [3]

	backbone	<i>AP</i>	<i>AP₅₀</i>	<i>AP₇₅</i>	<i>AP_S</i>	<i>AP_M</i>	<i>AP_L</i>
MNC	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
FCIS	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.1	50.0
FCIS+++	ResNet-101-C5-dilated	33.6	54.5	-	-	-	-
Mask R-CNN	ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	551.1
Mask R-CNN	ResNet-101-FPN	355.7	58.0	37.8	15.5	38.1	52.4
Mask R-CNN	ResNet-101-FPN	37.1	60.0	39.4	16.9	39.9	53.5

Table 4.2: Instance Segmentation Results on COCO test-dev [3]

1.3 Test Design Generation

When data is split, it is partitioned into two or more subsets. This project uses a three-part split to analyze or test the data; the second is to train the model, and the other is to validate the results during training:

Why is splitting data important?

Data splitting is a crucial feature of data science, especially for constructing data-driven models. This method aids in the development of data models and data-driven processes.

- **Train Dataset:** 70% of the original dataset + all generated images from the augmentation process. As a result, we obtain 1145 images (Figure 4.5).

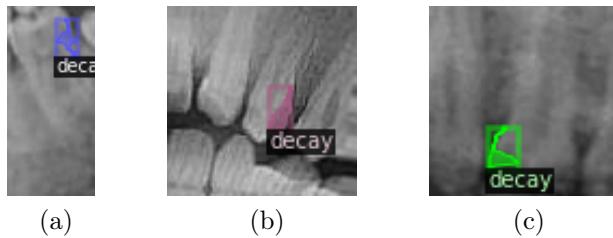


Figure 4.5: Train Samples

- **Validation Dataset:** 20% of the original dataset will be dedicated to the validation process. Consequently, we will work with 42 images in this task (Figure 4.6).

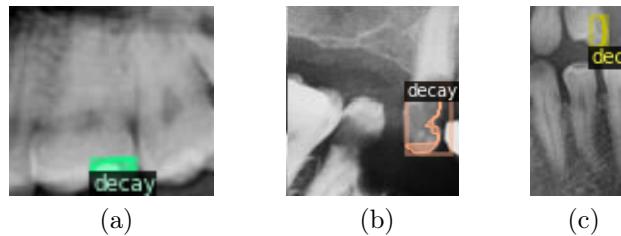


Figure 4.6: Validation Samples

- **Test Dataset:** 10% will be addressed to the operation of testing. Therefore, we will test on 22 images (Figure 4.7).

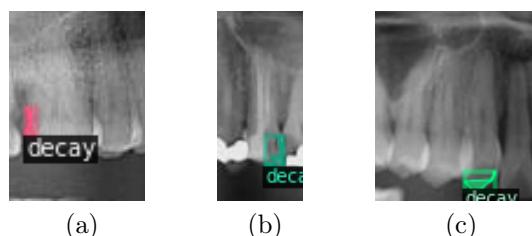


Figure 4.7: Test Samples

1.4 Model Build

1.4.1 Architecture of Detectron2

The following is the description of the meta-architecture of Detectron2, thus we have three main blocks, each block includes a primary class and multiple subclasses:

1. **Backbone Network:** Extracts feature maps at various sizes from the supplied image P2 (1/4 scale), P3 (1/8), P4 (1/16), P5 (1/32), and P6 (1/64) are the output characteristics of Base-RCNN-FPN.
2. **Region Proposal Network:** Identifies object regions using multi-scale features. By default, 1000 box proposals with confidence scores are generated.
3. **Box Head:** Crops and warps feature maps into several fixed-size features using proposal boxes to acquire fine-tuned box positions and classification results via fully-connected layers. Finally, 100 boxes (by default) in maximum are filtered out. The box head is one of the sub-classes of ROI Heads. For instance, Mask R-CNN has more ROI heads, such as a mask head.

What is inside each block? The next part and the Figure 4.8 show the detailed architecture.

Meta Architecture:

GeneralizedRCNN

which has:

1. **Backbone Network:**

- FPN: It forms pyramid features built on top of input feature maps.
 - ResNet: With two 3x3 convolution layers and a projection shortcut if needed.

2. **Region Proposal Network:**

- RPN: It takes feature maps and performs objectness classification and bounding box regression for anchors for anchors.
 - StandardRPNHead: It uses a 3x3 Conv to produce a shared hidden state from which one 1x1 Conv predicts objectness logits for each anchor and a second 1x1 Conv predicts bounding-box deltas specifying how to deform each anchor into an object proposal.
 - RPNOutput

3. **ROI Heads (Box Head):**

- StandardROIHeads: Every head independently processes the input features by each head's pooler and head.
 - ROIPooler: Map every box in "box_lists" to a feature map level index and return the assignment vector.

- FastRCNNConvFCHead: A head with multiple 3x3 Convolution layers (each followed by norm & relu) and then several FC layers (each followed by relu).
- FastRCNNOutputLayers
- FastRCNNOutputs

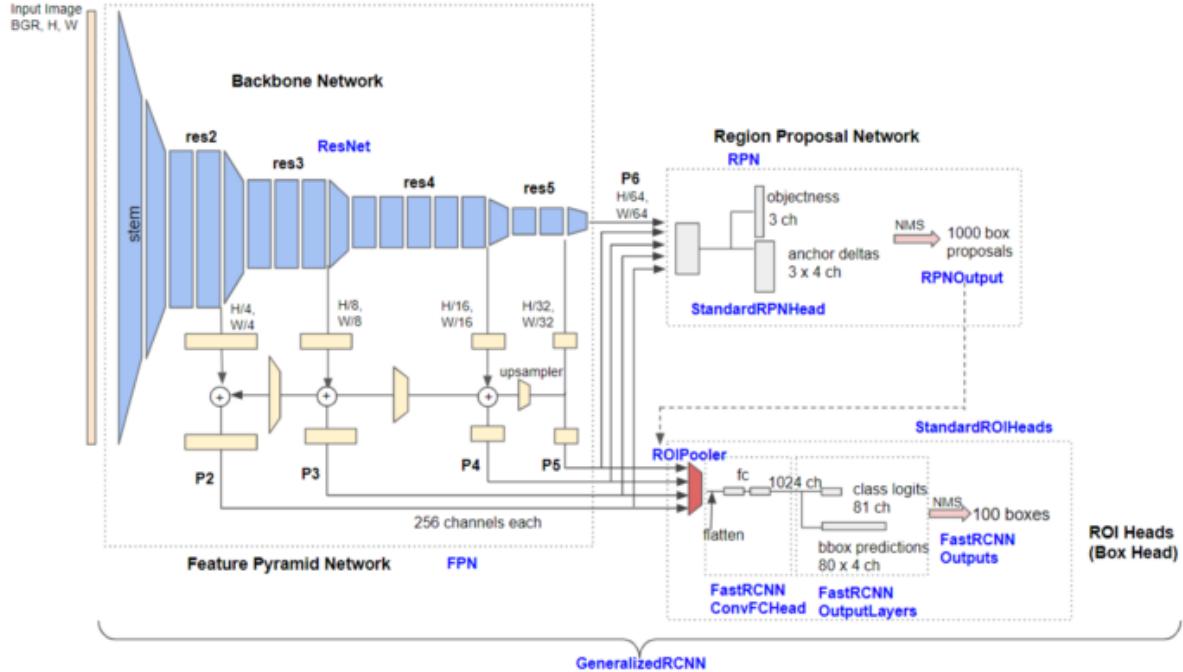


Figure 4.8: Detectron2 Architecture

1.4.2 Tools and Packages

While Python is an excellent programming language, there are certain drawbacks to executing Python scripts locally, especially if we wish to share our code. Python configurations (versions, modules) may make or break our code. Other issues include versioning, local machine performance, and non-technical user experience. Thus, we will use the *Google Colab Pro* version in this project (See Figure 4.9).

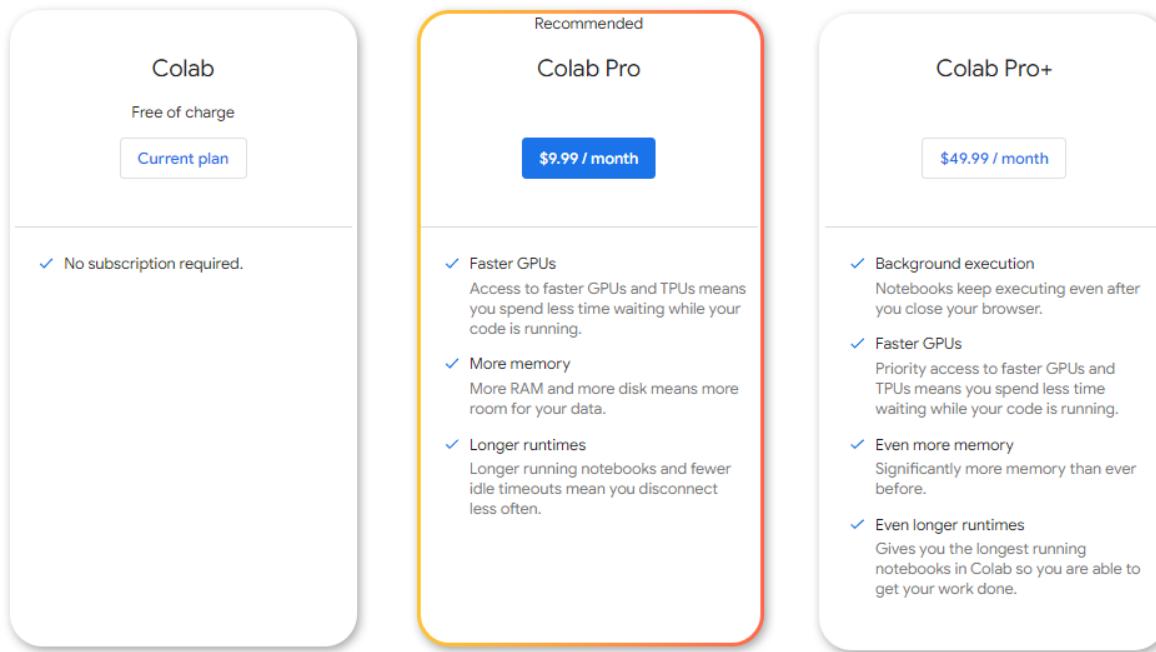


Figure 4.9: Google Colab Selected Plan

One of the essential benefits of Google Colab is its performance. Instead of a local machine, we will use the computational power of Google's servers. Training a network demand a lot of computer power and might take a long time to run. We don't have to be concerned about executing scripts on the cloud, and Our local machine's performance will not suffer when running a training process.

To start using **Detectron2**, we must install our necessary packages and import the dependencies, allowing us to use this model, as described in [Code 1](#) and [Code 2](#).

```

1 !python -m pip install pyyaml==5.1
2 !python -m pip install 'git+https://github.com/facebookresearch/detectron2.git'

```

Code 1: Install Detectron2 pip packages

```
1 # Basic setup:  
2 # Setup detectron2 logger  
3 import detectron2  
4 from detectron2.utils.logger import setup_logger  
5 setup_logger()  
6 # Import common libraries  
7 import torch  
8 import numpy as np  
9 import os, json, cv2, random  
10 from matplotlib import pyplot as plt  
11 from PIL import Image  
12 from google.colab.patches import cv2_imshow  
13 # Import common detectron2 utilities  
14 from detectron2 import model_zoo  
15 from detectron2.engine import DefaultPredictor  
16 from detectron2.config import get_cfg  
17 from detectron2.utils.visualizer import Visualizer  
18 from detectron2.data import MetadataCatalog, DatasetCatalog  
19 from detectron2.data.datasets import register_coco_instances  
20 from detectron2.engine import DefaultTrainer  
21 from detectron2.utils.visualizer import ColorMode  
22 from detectron2.solver import build_lr_scheduler, build_optimizer  
23 from detectron2.checkpoint import DetectionCheckpointer, PeriodicCheckpointer  
24 from detectron2.utils.logger import log_every_n_seconds  
25 from detectron2.utils.events import EventStorage  
26 from detectron2.engine.hooks import HookBase  
27 from detectron2.modeling import build_model  
28 import detectron2.utils.comm as comm  
29 from detectron2.data import (  
30     DatasetMapper,  
31     MetadataCatalog,  
32     build_detection_test_loader,  
33     build_detection_train_loader,  
34 )  
35 from detectron2.evaluation import (  
36     COCOEvaluator,  
37     DatasetEvaluators,  
38 )
```

Code 2: Import Dependencies

1.4.3 Load and Register the Dataset

In this section, we dedicated some lines of code to download the data uploaded in **Roboflow** using their API as described in [Code 3](#).

```

1 from roboflow import Roboflow
2 VERSION = 2
3 rf = Roboflow(api_key="API_KEY_HERE")
4 project = rf.workspace("workspace").project(PROJECT)
5 dataset = project.version(VERSION).download("coco")

```

Code 3: Download the uploaded data in Roboflow

Afterwards, we will register all downloaded dataset in 3 different instances as shown in [Code 4](#)

```

1 register_coco_instances("decay_train", {}),
2   ↪ f"./{PROJECT}-{VERSION}/train/_annotations.coco.json",
2   ↪ f"./{PROJECT}-{VERSION}/train/")
3 register_coco_instances("decay_valid", {}),
4   ↪ f"./{PROJECT}-{VERSION}/valid/_annotations.coco.json",
4   ↪ f"./{PROJECT}-{VERSION}/valid/")
3 register_coco_instances("decay_test", {}),
4   ↪ f"./{PROJECT}-{VERSION}/test/_annotations.coco.json",
4   ↪ f"./{PROJECT}-{VERSION}/test/")

```

Code 4: Register Instances

1.4.4 Customize Configurations

Detectron2 includes a default configuration with numerous hyperparameters. To customize the default configuration, in the first place, we will import `get_cfg`, which returns a dictionary of hyperparameters.

We will find configuration files in `detectron2.model_zoo`. Furthermore, we will use a pre-trained model by loading the weight from `model_zoo`. Moreover, we will configure the additional configurations, as we did in the following [Code 5](#).

```

1 cfg = get_cfg()
2 # Get configuration from model_zoo
3 cfg.merge_from_file(model_zoo.get_config_file(
4   ↪ "COCO-InstanceSegmentation/mask_rcnn_R_50_FPN_3x.yaml"))
4 # Let training initialize from model zoo
5 cfg.MODEL.WEIGHTS = model_zoo.get_checkpoint_url(
6   ↪ "COCO-InstanceSegmentation/mask_rcnn_R_50_FPN_3x.yaml")
6 # Model
7 cfg.MODEL.ROI_HEADS.NUM_CLASSES = 1 # Only has one class (decay)
8 cfg.MODEL.ROI_HEADS.BATCH_SIZE_PER_IMAGE = 64 # The "RoIHead batch size".
9 cfg.MODEL.ROI_HEADS.SCORE_THRESH_TEST = 0.15 # Set threshold for this model
10 # Solver
11 cfg.SOLVER.BASE_LR = 0.01 # pick a good Learning Rate
12 cfg.SOLVER.MAX_ITER = 1200
13 cfg.SOLVERIMS_PER_BATCH = 128 # The "batch size" commonly known in deep learning
14 cfg.SOLVER.STEPS = []
15 # DataLoader

```

```

16 cfg.DATALOADER.NUM_WORKERS = 8
17 # Test
18 cfg.TEST.EVAL_PERIOD = 200
19 cfg.TEST.AUG.ENABLED = True
20 # DATASETS
21 cfg.DATASETS.TRAIN = ("decay_train", )
22 cfg.DATASETS.TEST = ("decay_valid", )
23 os.makedirs(cfg.OUTPUT_DIR, exist_ok=True)
24 # Early Stopping Param
25 PATIENCE = 400

```

Code 5: Customize configurations

1.4.5 Customize Default Trainer

Given that we already did the augmentation process, we will eliminate the default trainer implemented by Detectron2. Not only because of augmentation but also owing to missing the evaluation and the early stopping algorithms in the Detectron2 default model. Hence, we will customize our trainer as demonstrated in the [Code 6](#).

```

1 class Trainer(DefaultTrainer):
2
3     def train(self):
4         model = build_model(cfg)
5         BEST_LOSS = np.inf
6         optimizer = super().build_optimizer(cfg, model)
7         scheduler = super().build_lr_scheduler(cfg, optimizer)
8         checkpointer = DetectionCheckpointer(
9             model, cfg.OUTPUT_DIR, optimizer=optimizer, scheduler=scheduler
10        )
11        start_iter = (
12            checkpointer.resume_or_load(cfg.MODEL.WEIGHTS,
13                ↳ resume=resume).get("iteration", -1) + 1
14        )
15        self.iter = self.start_iter = start_iter
16        self.max_iter = cfg.SOLVER.MAX_ITER
17        periodic_checkpointer = PeriodicCheckpointer(
18            checkpointer, cfg.SOLVER.CHECKPOINT_PERIOD, max_iter=self.max_iter
19        )
20        logger = logging.getLogger(__name__)
21        logger.info("Starting training from iteration {}".format(start_iter))
22        with EventStorage(start_iter) as self.storage:
23            try:
24                self.before_train()
25                data_loader = Trainer.build_train_loader(cfg)
26                for data, self.iter in zip(data_loader, range(start_iter,
27                    ↳ self.max_iter)):
28                    self.before_step()
29                    self.run_step()
30                    self.after_step()
31                    loss_dict = model(data)
32                    losses = sum(loss_dict.values())
33                    assert torch.isfinite(losses).all(), loss_dict
34                    loss_dict_reduced = {k: v.item() for k, v in
35                        ↳ comm.reduce_dict(loss_dict).items()}

```

```

33         losses_reduced = sum(loss for loss in loss_dict_reduced.values())
34         if (
35             cfg.TEST.EVAL_PERIOD > 0
36             and (self.iter + 1) % cfg.TEST.EVAL_PERIOD == 0
37             and self.iter != self.max_iter - 1
38         ):
39             super().test(cfg, model)
40             comm.synchronize()
41             if self.iter > prev_iter:
42                 prev_iter = self.iter
43                 if losses_reduced < BEST_LOSS:
44                     BEST_LOSS = losses_reduced
45                     patience_counter = 0
46                 else:
47                     patience_counter += 1
48                     if patience_counter % 100 == 0:
49                         print(f"Loss has not improved for {patience_counter}
49                           → iterations")
50                         if patience_counter >= PATIENCE:
51                             print(f"EARLY STOPPING")
52                             break
53             self.iter += 1
54         except Exception:
55             logger.exception("Exception during training:")
56             raise
57     finally:
58         self.after_train()

59
60     @classmethod
61     def build_train_loader(cls, cfg):
62         mapper = DatasetMapper(cfg, is_train=True, augmentations[])
63         return build_detection_train_loader(cfg, mapper=mapper)
64
65     @classmethod
66     def build_test_loader(cls, cfg, dataset_name):
67         mapper = DatasetMapper(cfg, is_train=False, augmentations[])
68         return build_detection_test_loader(cfg, dataset_name, mapper=mapper)
69
70     @classmethod
71     def build_model(cls, cfg):
72         model = build_model(cfg)
73         logger = logging.getLogger(__name__)
74         logger.info("Model:\n{}".format(model))
75         return model
76
77     @classmethod
78     def build_evaluator(cls, cfg, dataset_name, output_folder=None):
79         if output_folder is None:
80             output_folder = os.path.join(cfg.OUTPUT_DIR, "inference")
81         evaluator_list = []
82         evaluator_type = MetadataCatalog.get(dataset_name).evaluator_type
83         if evaluator_type in ["coco", "coco_panoptic_seg"]:
84             evaluator_list.append(COCOEvaluator(dataset_name,
84               → output_dir=output_folder))
85         if len(evaluator_list) == 0:
86             raise NotImplementedError(
86               "no Evaluator for the dataset {} with the type
86               → {}".format(dataset_name, evaluator_type))

```

```

88         )
89     if len(evaluator_list) == 1:
90         return evaluator_list[0]
91     return DatasetEvaluators(evaluator_list)

```

Code 6: Customize DefaultTrainer

1.4.6 Train the Model

The training is now straightforward, unlike other models. We will train our model with just a few lines of code as implemented in the [Code 7](#).

```

1 resume = True
2 trainer = Trainer(cfg)
3 trainer.resume_or_load()
4 trainer.train()

```

Code 7: Train the model

1.4.7 Prediction and Visualization

Visualization helps present the results of object detection and evaluate them. The completed model is saved in the output directory. We will load the final model's weights, read the images from the test instance one by one, run the predictor, and save the output images with masks and bounding boxes. All things considered in the process mentioned in [Code 8](#).

```

1 # Set the path to the model we just trained
2 cfg.MODEL.WEIGHTS = os.path.join(cfg.OUTPUT_DIR, "/content/output/model_final.pth")
3 cfg.MODEL.ROI_HEADS.SCORE_THRESH_TEST = 0.5    # set a custom testing threshold
4 predictor = DefaultPredictor(cfg)
5
6 dataset_dicts = DatasetCatalog.get("decay_test")
7 for d in random.sample(dataset_dicts, 3):
8     im = cv2.imread(d["file_name"])
9     outputs = predictor(im)
10    v = Visualizer(im[:, :, ::-1],
11                    scale=1, instance_mode=ColorMode.IMAGE_BW
12    )
13    out = v.draw_instance_predictions(outputs["instances"].to("cpu"))
14    im = Image.fromarray(out.get_image()[:, :, ::-1])
15    cv2.imshow(out.get_image()[:, :, ::-1])

```

Code 8: Predict and visualize results

1.5 Model Assessment

Lastly and most importantly, we built various models with different hyperparameters. However, we need to interpret the results extracted from Tensorboard and the prediction process.

Learning rate, Batch Size, RoIHead Batch Size, and DataLoader.Num_Workers are the hyperparameters we will modify before model training to use the results for inference.

- **Learning Rate:**

The learning rate determines how fast the model adapts to the problem. Lower learning rates necessitate more training epochs due to the more minor changes in the weights with each update, whereas more significant learning rates produce quick changes and necessitate fewer training epochs.

A high learning rate can cause the model to converge too rapidly on a suboptimal solution, whereas a low learning rate can cause the process to slow.

Training deep learning neural networks is difficult because we must choose a learning rate appropriately. It could be the model's most crucial hyperparameter.

- **Batch Size:**

In Detectron2 the batch size is defined as *IMS_PER_BATCH* which will define the number of samples propagated via the network in one iteration.

- **RoIHead Batch Size:**

It is a parameter that is used to sample a subset of proposals coming out of RPN to calculate class and regression loss during training.

- **Num_Workers:**

It instructs the data loader instance on the number of sub-processes to use for data loading. If *Num_Workers* is set to zero (the default), the GPU must wait for the CPU to load data. In theory, the bigger the *Num_Workers*, the more effectively the CPU loads data and the less time the GPU needs to wait.

In the upcoming subsections, we will visualize some essential key metrics to improve the results and decide which hyperparameters to choose.

These are the metrics we will follow along this project:

- Learning Rate
- Fast R-CNN Accuracy
- Mask R-CNN Accuracy
- Mask R-CNN False Negative
- Mask R-CNN False Positive
- Total Loss: The total loss is a combination of **loss_box_reg**, **loss_cls**, **loss_mask**, **loss_rpn_cls** and **loss_rpn_loc**.
- Bbox AP
- Segmentation AP

What is AP? Average Precision (AP.) is the mean of the areas under the precision-recall curve for each object class.

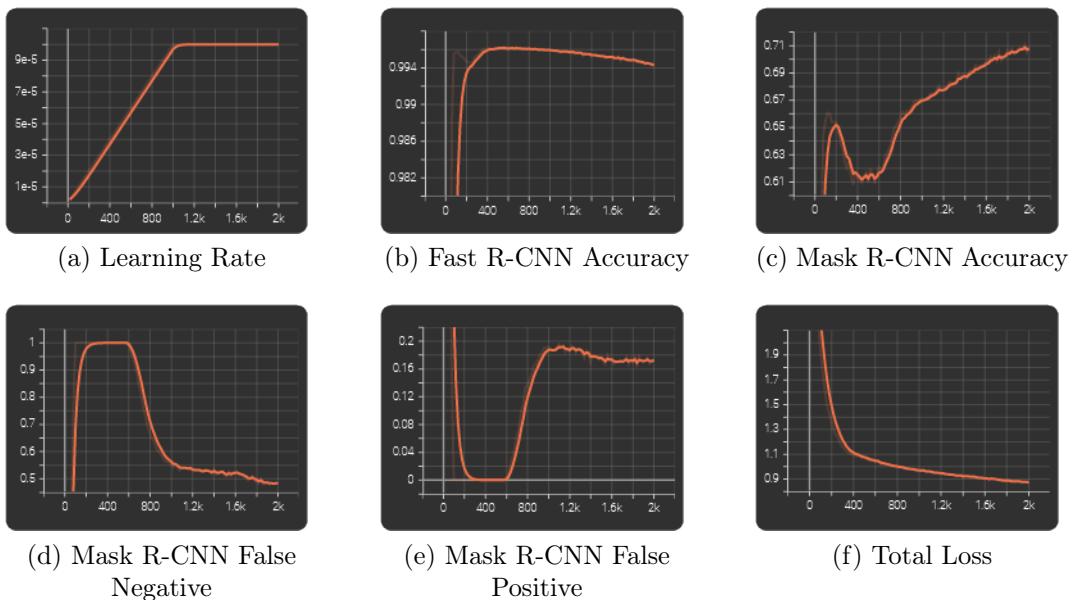
1.5.1 Initial Results

1. First Configuration Results:

Learning Rate	10^{-4}
Batch Size	128
RoIHead Batch Size	512
Num_Workers	2

Table 4.3: 1st Training Results

With this configuration, we attempted to start with a low learning rate, a number of workers equal to 2, and ran the training with the default RoIHead batch size (Results in Figure 4.10 & Predictions in Figure 4.11).

Figure 4.10: 1st Tensorboard Results of the Configuration described in Table 4.3

P.S. The number 0 shown in the predictions figures is the class decay.

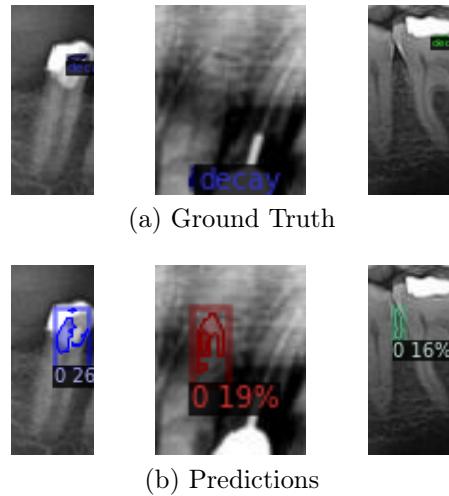


Figure 4.11: Comparaison between Ground Truth and Predictions of the 1st model

2. Second Configuration Results:

Learning Rate	10^{-4}
Batch Size	128
RoIHead Batch Size	64
Num_Workers	2

Table 4.4: 2nd Training Results

With this configuration, we went for a higher number of workers and trained our model with smaller RoIHead batch size (Results in [Figure 4.12](#) & Predictions in [Figure 4.13](#)).

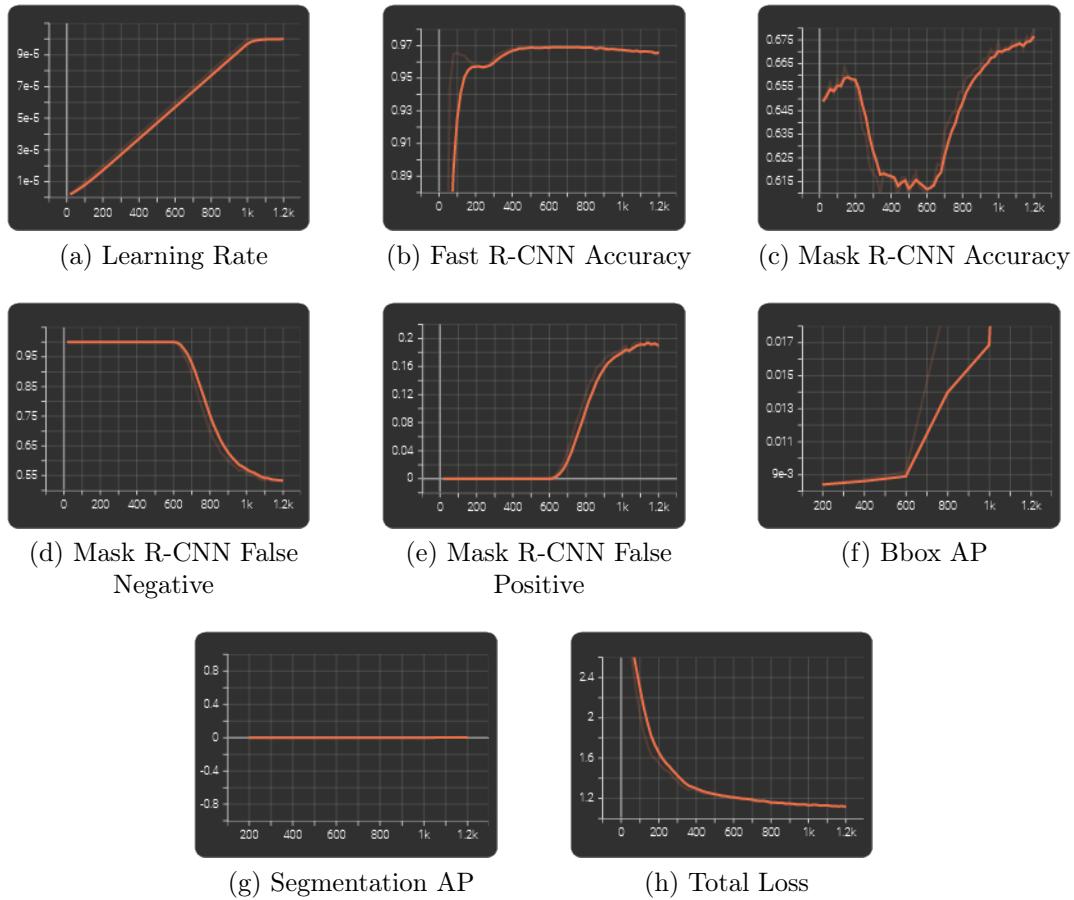


Figure 4.12: 2nd Tensorboard Results of the Configuration described in Table 4.4

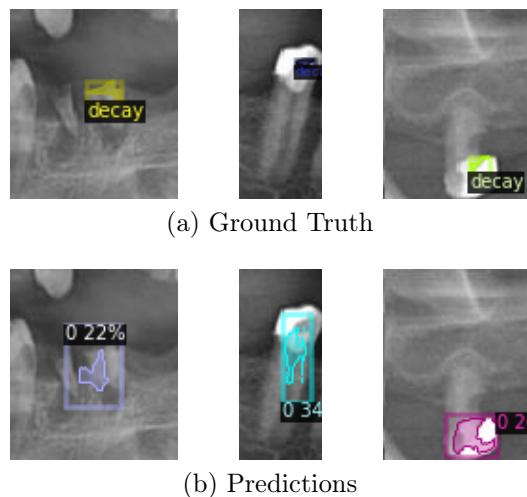


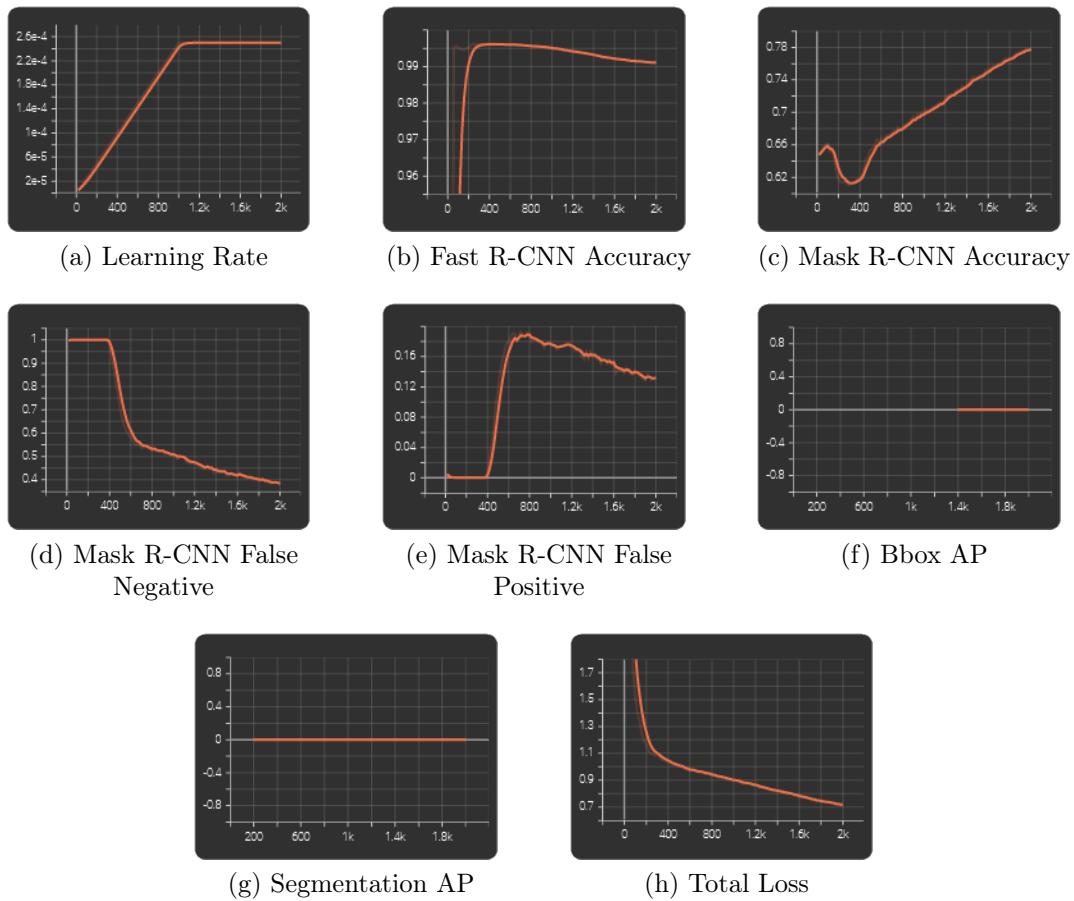
Figure 4.13: Comparaison between Ground Truth and Predictions of the 2nd model

3. *Third Configuration Results:*

Learning Rate	25×10^{-5}
Batch Size	128
RoIHead Batch Size	512
Num _ Workers	2

 Table 4.5: 3rd Training Results

The adjustments of our hyperparameters in this part are setting higher learning rate, higher batch sizes, 128 for the primary batch size and 512 for the RoIHead batch size (Results in Figure 4.14 & Predictions in Figure 4.15).


 Figure 4.14: 3rd Tensorboard Results of the Configuration described in Table 4.5

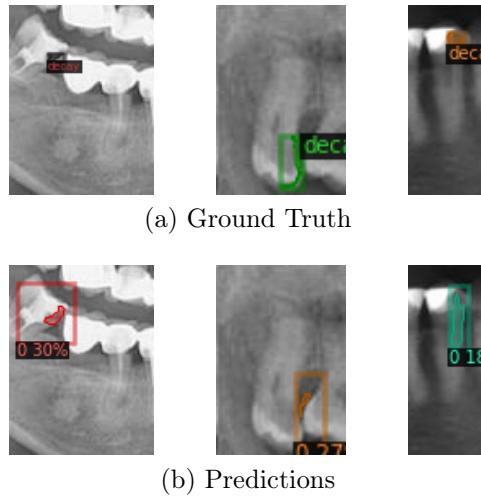


Figure 4.15: Comparaison between Ground Truth and Predictions of the 3rd model

1.5.2 Intermediate Results

1. *Fourth Configuration Results*

Learning Rate	25×10^{-5}
Batch Size	32
RoIHead Batch Size	16
Num_Workers	2

Table 4.6: 4th Training Results

During this training, we set the batch size to 32 and we tried to lower the RoIHead batch size (Results in [Figure 4.16](#) & Predictions in [Figure 4.17](#)).

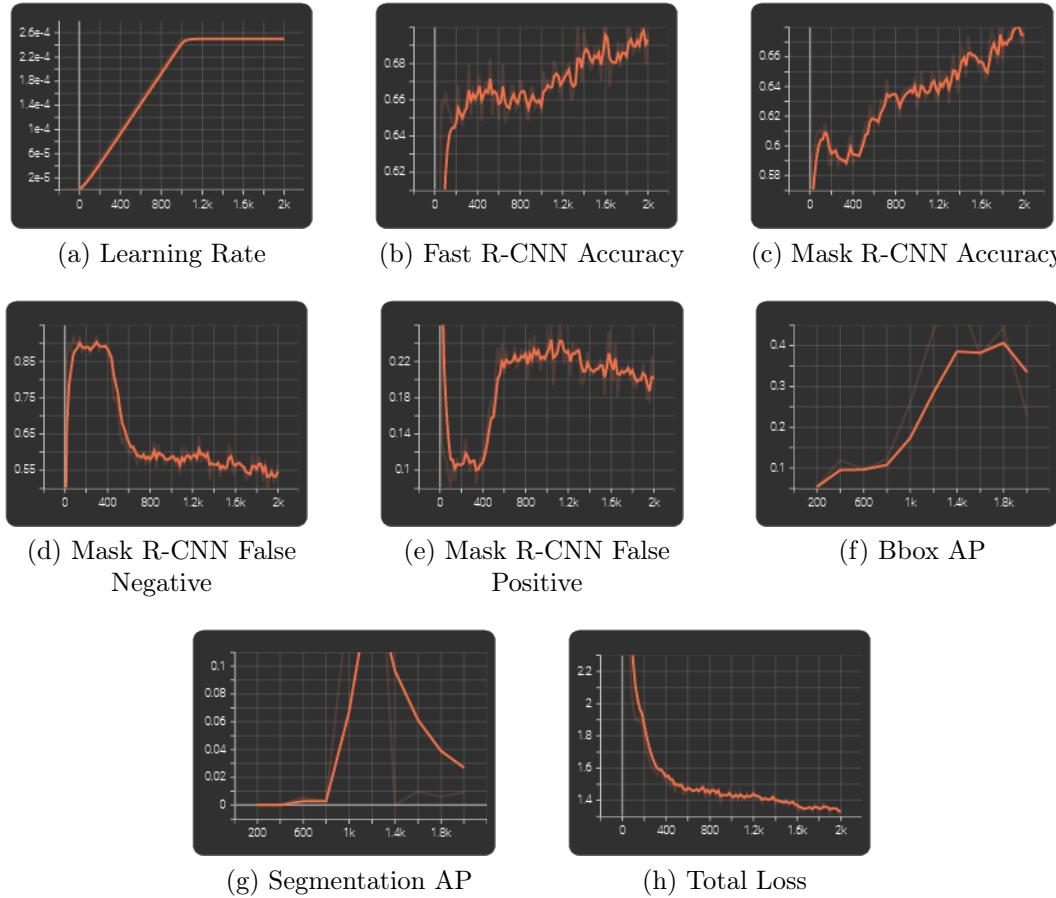


Figure 4.16: 4th Tensorboard Results of the Configuration described in Table 4.6

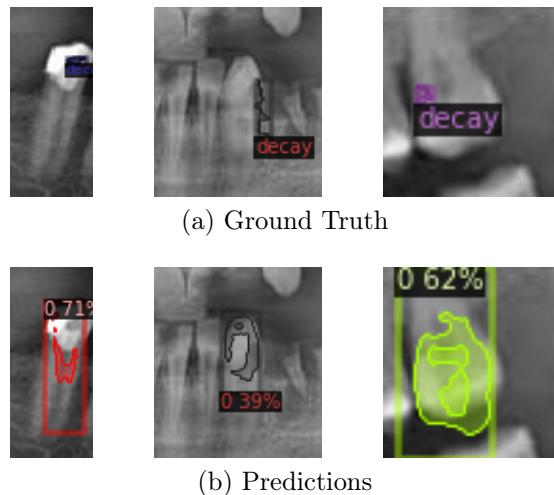


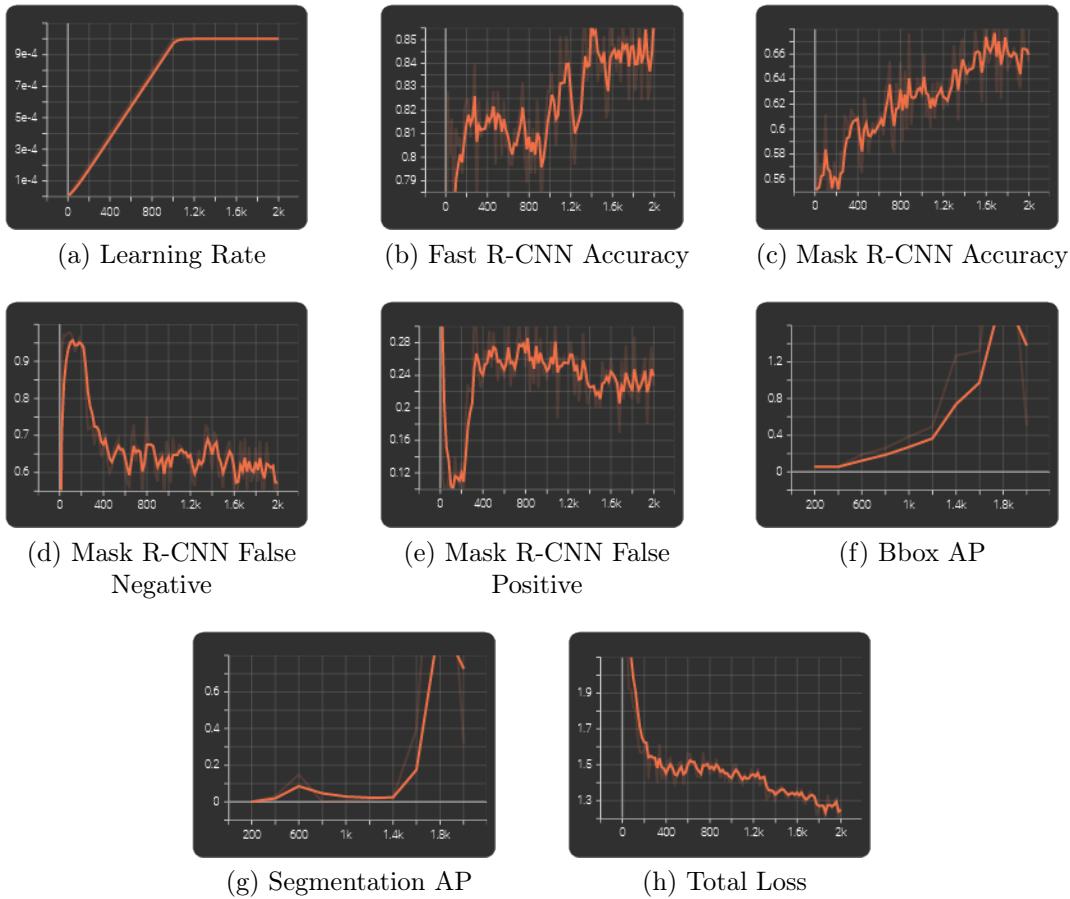
Figure 4.17: Comparaison between Ground Truth and Predictions of the 4th model

2. Fifth Configuration Results:

Learning Rate	10^{-3}
Batch Size	8
RoIHead Batch Size	16
Num_Workers	2

 Table 4.7: 5th Training Results

During this training, we alter our hyperparameters to establish a greater learning rate and smaller batch sizes, 8 for the primary and 16 for the RoIHead batch sizes (Results in [Figure 4.18](#) & Predictions in [Figure 4.19](#)).


 Figure 4.18: 5th Tensorboard Results of the Configuration described in Table 4.7

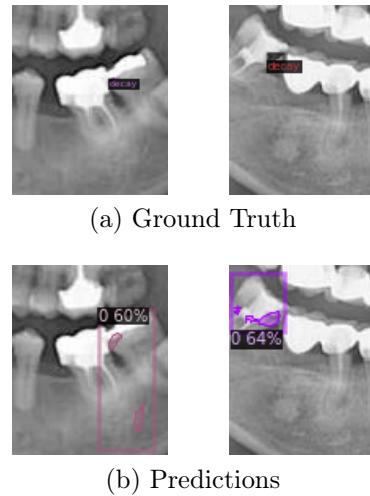


Figure 4.19: Comparaison between Ground Truth and Predictions of the 5th model

3. Sixth Configuration Results:

Learning Rate	10^{-2}
Batch Size	128
RoIHead Batch Size	16
Num_Workers	2

Table 4.8: 6th Training Results

We chose a greater parameter for the learning rate and trained our model with a batch size equal to 128 in this configuration (see [Figure 4.20](#) & Predictions in [Figure 4.21](#)).

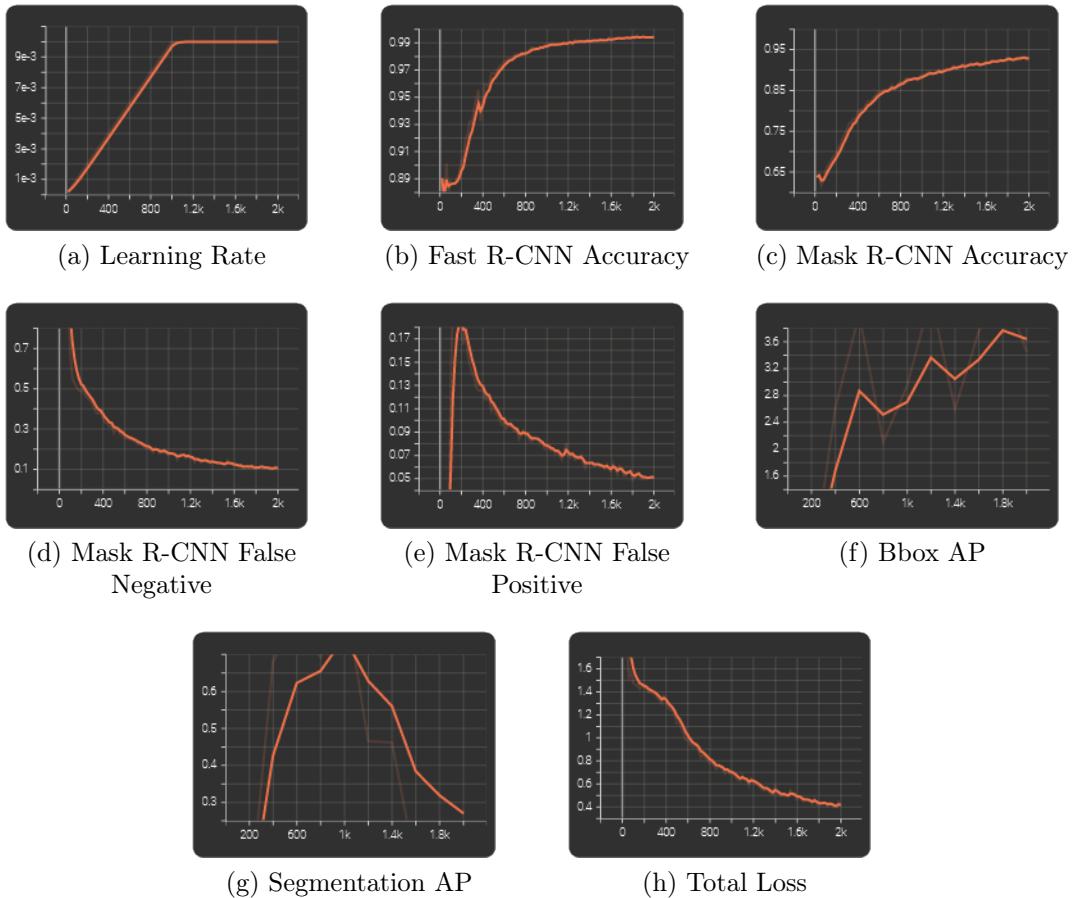


Figure 4.20: 6th Tensorboard Results of the Configuration described in Table 4.8

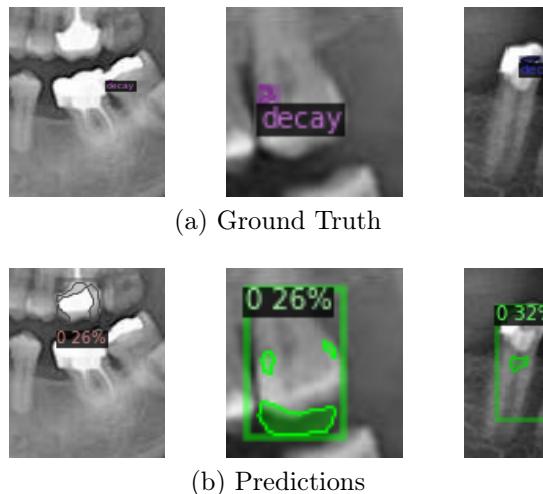


Figure 4.21: Comparaison between Ground Truth and Predictions of the 6th model

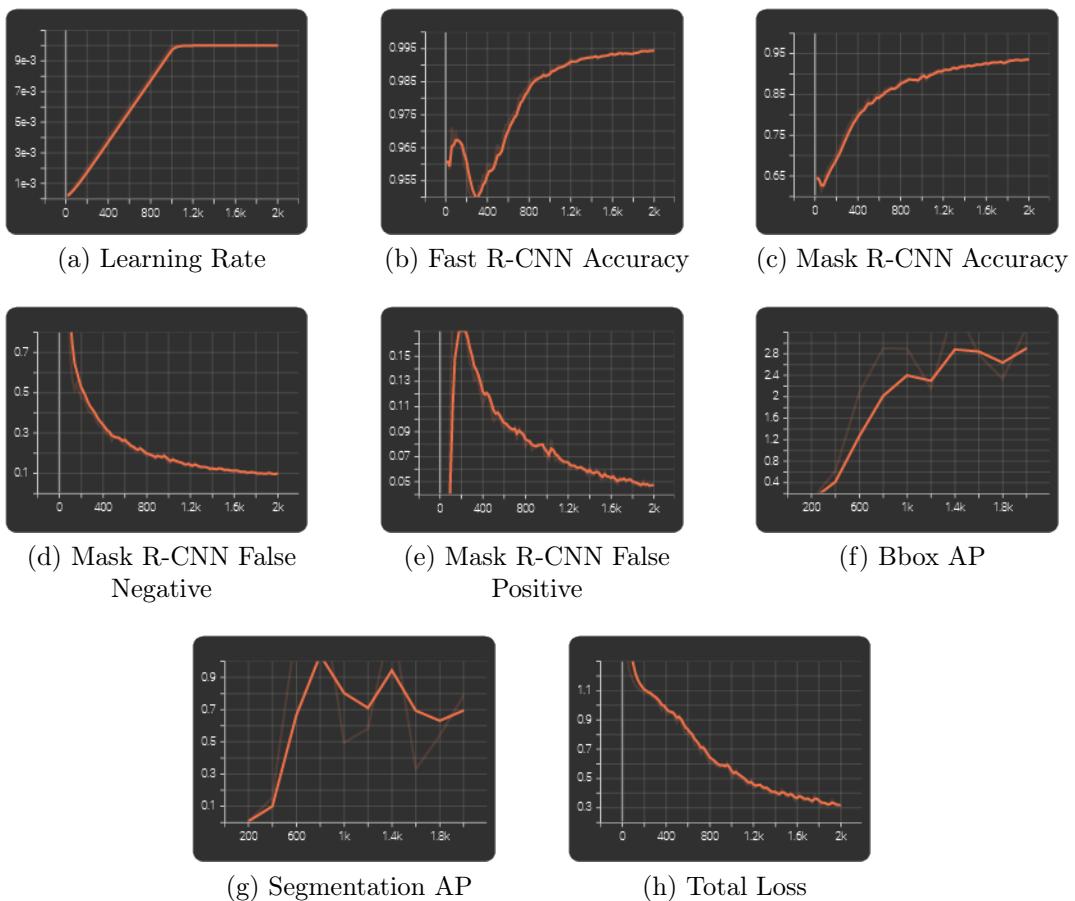
1.5.3 Final Results

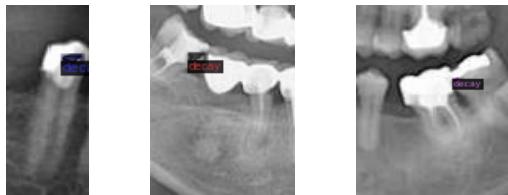
1. *Seventh Configuration Results:*

Learning Rate	10^{-2}
Batch Size	128
RoIHead Batch Size	64
Num_Workers	2

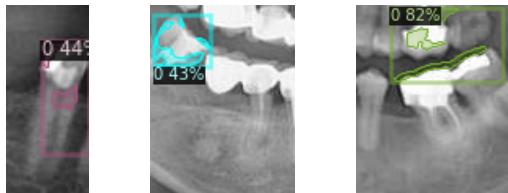
 Table 4.9: 7th Training Results

We attempted to start with the same learning rate and execute the training with a higher RoIHead batch size (Results in Figure 4.22 & Predictions in Figure 4.23).


 Figure 4.22: 7th Tensorboard Results of the Configuration described in Table 4.9



(a) Ground Truth



(b) Predictions

Figure 4.23: Comparaison between Ground Truth and Predictions of the 7th model

2. Eighth Configuration Results:

Learning Rate	10^{-2}
Batch Size	128
RoIHead Batch Size	64
Num_Workers	8

Table 4.10: 8th Training Results

Finally, with the configuration described in [Table 4.10](#), we changed our hyperparameters to get a higher number of data loader workers and keep the same other hyperparameters (Results in [Figure 4.24](#) & Predictions in [Figure 4.25](#)).

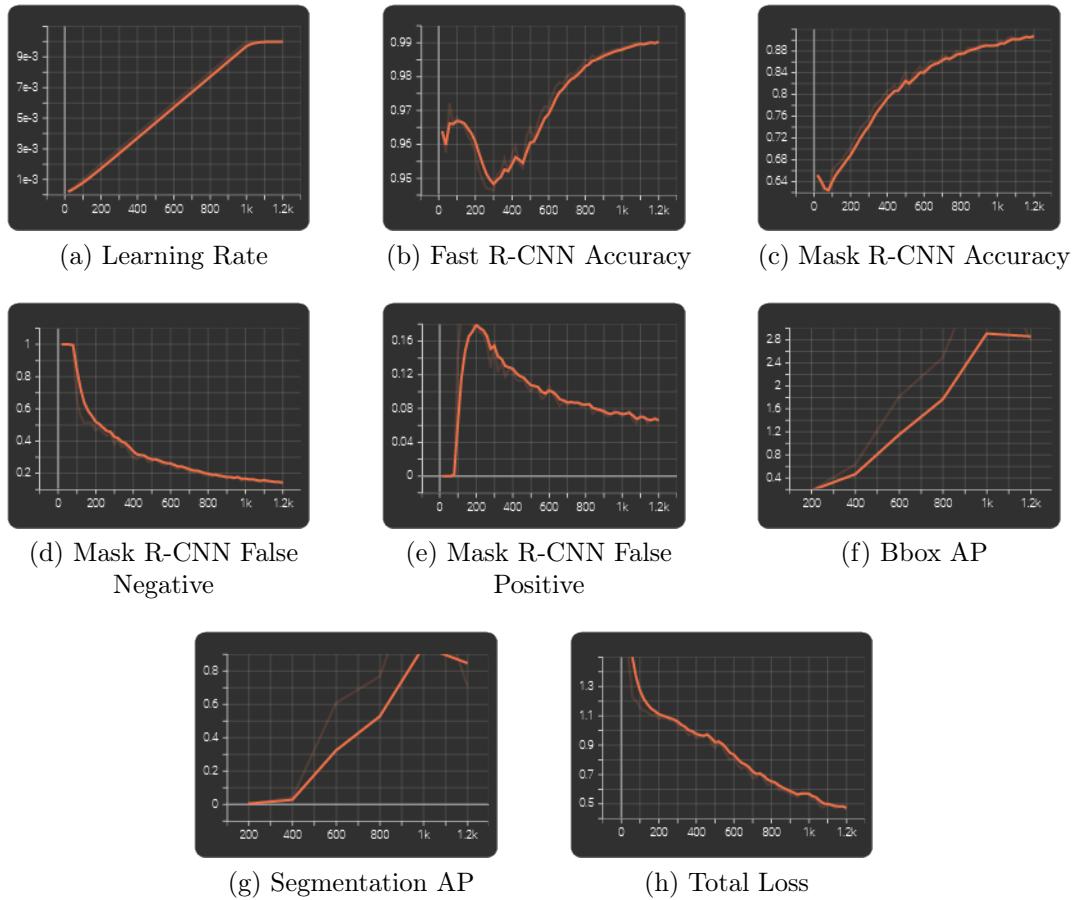


Figure 4.24: 8th Tensorboard Results of the Configuration described in Table 4.10

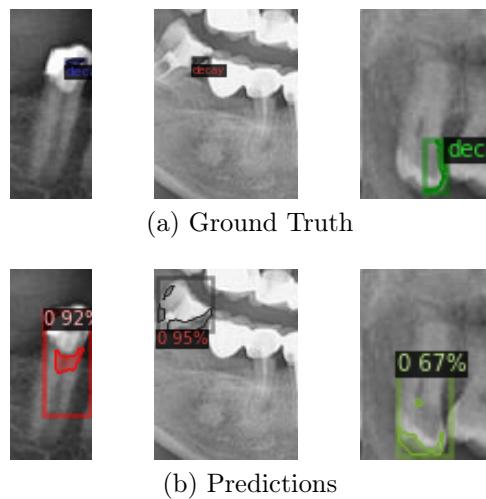


Figure 4.25: Comparison between Ground Truth and Predictions of the 8th model

2 Evaluation

2.1 Results

Which one should we approve for this project?

Based on the previous training results, we approved 8^{th} configuration as the best model for prediction results and the key metrics we registered. According to the training results and [Figure 4.24](#), the selected model achieves 99% of Accuracy in Fast R-CNN and 92% in Mask R-CNN. for Bbox AP, the model reaches a maximum score equal to 70 and 30 for the Segmentation AP. About Total Loss, we accomplished 0.45 as a minimum loss value.

2.2 Next Step

Indeed, The prediction results were close to the ground truth, yet not enough to move to the next step. Hence, we must implement improvements across all previous actions to provide more accurate results.

Conclusion

To summarize, in this chapter, we will describe the chosen training architecture, how it works, and the code implemented during the training process. Afterward, we ran numerous training settings to examine the retrieved data and predictions. We analyzed the authorized model by reviewing the test dataset's key metrics and prediction results.

Conclusion and perspectives

This report describes the work within AI Diagnosis Vision for six months within the framework of a project end of the study to obtain a national diploma of an engineer in computer science at the National School of Engineers of Carthage. In this report, we have detailed the different steps followed to achieve the implementation of our solution. For the realization, we applied the principles of the CRISP-DM method.

Our main goal of this project is to validate a trained model to detect caries in a radiograph and segment them. As mentioned in the first chapter, we must perform specific steps to reach this goal.

After introducing the hosting company and defining our goals and action steps, we began by stating the problems and the dentist's goals. Meanwhile, we studied various research papers, approaches, and related projects that helped us in the rest of the project.

Afterward, we described the collected data and its properties. Apart from this, we developed our tooth segmentation technique after improving the dataset to extract the main features in a radiograph; meanwhile, we updated the annotation files. Following this part, we performed various augmentation techniques to expand the dataset taken from the previous processing.

Finally, not only we discussed why we selected Detectron2 to train our model but also we clarified the chosen architecture and the training process we implemented. Furthermore, we evaluated the approved model we selected among different training configurations.

Despite the work done in this project, we require optimizing every step of this project. To start with, in the data collection step, we have to ensure that we get higher quality radiographs and collect more images for the training. In addition, instead of using enhancement on all the pixels of the input image, we can apply a specific enhancement to particular regions of the image for separating teeth from auxiliary objects (spinal column, prosthesis, bone, contrast noise).

In the gap detection process, detecting a small space between teeth is difficult. To create a more accurate teeth division, we can use a polar coordinate system; in other words, we can convert the jaw into a circular shape with polar detection to expand the tiny cavity between teeth, and after detecting that gap, we reconvert the image to the cartesian coordinate system. Following this process, we may change the algorithm of linking points from the generating borders process.

Altogether, we will improve our model results and predictions to proceed to the deployment step and finalize the work.

References

- [1] Openstax college. 2409. adapted by geeky medics. https://commons.wikimedia.org/wiki/File:2409_Tooth.jpg.
- [2] Richard Verbeet Mathias Rieder. Robot-human-learning for robotic picking processes. *Conference: Hamburg International Conference of LogisticsAt: Hamburg*, 2019.
- [3] Piotr Dollár Ross Girshick Kaiming He, Georgia Gkioxari. Mask R-CNN. *IEEE International Conference on Computer Vision, Facebook AI Research (FAIR)*, 2017.
- [4] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019.
- [5] AI Diagnosis Vision. <https://aidgvision.com>.
- [6] Vos et al., T. years lived with disability (ylds) for 1160 sequelae of 289 diseases and injuries 1990–2010: a systematic analysis for the global burden of disease study2010. *lancet*380(9859). pages 2163–2196, 2013.
- [7] Khalesi M. Gholami L Booshehry M.Z., Fasihinia H. Dental caries diagnostic methods. *DJH2*(1), page 1–12, 2011.
- [8] Amaechi B.T. Emerging technologies for diagnosis of dental caries: the road so far. *J. Appl. Phys.* 105(10), 102047, 2009.
- [9] Ali M.H. Numpang A.D.A. Noor N.M., Khalid N.E.A. Fish bone impaction using adaptive histogram equalization (AHE). second international conference on computer research and development, pp. *IEEE*, page 163–167, 2010.
- [10] Ogawa K. Sakata M. Noise reduction and contrast enhancement for small-dose-x-ray images in wavelet domain. ieee nuclear science symposium con-fference record (nss/mic), pp. *IEEE*, page 2924–2929, 2009.
- [11] Ana Azevedo and Manuel Santos. Kdd, semma and crisp-dm: A parallel overview. pages 182–185, 01 2008.
- [12] Krzyzak A. Li S. Li S., Fevens T. An automatic variational level set segmentation framework for computer aided dental x-rays analysis in clinical environments. *Comput. Med. Imag. Graph.* 30, page 65–74, 2006.
- [13] Y.S.Cho C.-H.Kuo P.-L.Lin, P.-W.Huang. An automatic and effective tooth isolation method for dental radiographs. *OPTO-ELECTRONICS REVIEW* 21(1), pages 126–136, 2012.

-
- [14] O. Nomir and M. Abdel Mottaleb. A system for human identification from x ray dental radiographs. *Pattern Recogn.* 38, page 1295–1305, 2005.
 - [15] Darren Osterloh Serestina Viriri. Unsupervised caries detection in non-standardized periapical dental x-rays. *Computer Vision and Graphics*, pages 329–340, 2018.
 - [16] Ro Y.M.a Yoon, J.H. Enhancement of the contrast in mammographic images using the homomorphic filter method. *IEICE Trans. Inf. Syst.* 85(1), page 298–303, 2002.
 - [17] T. Lindeberg. Feature detection with automatic scale selection. *Int. J. Comput. Vis.* 30(2), page 79–116, 1998.
 - [18] Cotrim-Ferreira F.A.-Ribeiro J.A. Ferreira-Santos R.I. Vellini-Ferreira, F. Mapping of proximal enamel thickness in permanent teeth. *Braz. J. Oral Sci.* 11(4), page 481–485, 2012.
 - [19] J. Oliveira. Caries detection in panoramic dental x-ray images. *Computer Vision and Graphics*, pages 329–340, 2009.
 - [20] Nassar D.E.M.-Fahmy G.-Ammar H.H. Said, E.H. Teeth segmentation in digitized dental x-ray films using mathematical morphology. *IEEE Trans. Inf. Forensics Secur.* 1(2), page 178–189, 2006.
 - [21] Abaza A.-Ross A.-Ammar H. Shah, S. Automatic tooth segmentation using active contour without edges. *Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pages 1–6, 2006.
 - [22] Mohd Rahim-M.S. Rehman-A. Altameem A.-Saba T. Rad, A.E. Evaluation of current dental radiographs segmentation approaches in computer-aided applications. *IETE Tech. Rev.* 30(3), page 210–222, 2013.
 - [23] Lin P.L. Lai, Y.H. Effective segmentation for dental x-ray images using texture-based fuzzy inference system. *Blanc-Talon, J., Bourennane, S., Philips, W., Popescu, D., Scheunders, P. (eds.) ACTIVS 2008. LNCS, vol. 5259*, page 936–947, 2008.
 - [24] et al. Tracy, K.D. Utility and effectiveness of computer-aided diagnosis of dental caries. *Gen. Dent.* 59(2), page 136–144, 2010.
 - [25] B. Dykstra. Interproximal caries detection. *how good are we? Dent. Today* 27(4), page 144–146, 2008.
 - [26] T.Kalpalatha Reddy Megalan Leo. L. Learning compact and discriminative hybrid neural network for dental caries classification. *Microprocessors and Microsystems* (82), 2021.
 - [27] T.Kalpalatha Reddy Megalan Leo. L. Removal of various noises in dental x-ray images using selective median filter. *Int J Innovative Technol Expl Eng (IJITEE)* 8 (12), pages 2278–3075, 2019.
 - [28] Amjad Rehman-Ayman Altameem Tanzila Saba Abdolvahab Ehsani Rad, Mohd Shafry Mohd Rahim. Evaluation of current dental radiographs.
 - [29] J. Oliveira and H. Proenc a. Caries detection in panoramic dental x-ray images. pages 181–182, 2011.

-
- [30] H. Chen. Matching of dental K. Jain. x-ray images for human identification. *Pattern Recognition*, 37(7), 2004.
 - [31] Albumentations library. <https://github.com/albumentations-team/albumentations#pixel-level-transforms>.

Abstract

This report results from a six-month graduation internship within AI Diagnosis Vision. Dental caries is a disease of the hard tissues of the teeth caused by the action of micro-organisms present in plaque on fermentable carbohydrates (principally sugars). As a result, detecting dental caries in its early stages is critical. This project serves two essential purposes: Implementing a custom teeth segmentation process for feature extraction. The dataset contains 237 images with varying types of noise and contrast. Second, presenting a complete case study to detect and segment dental caries in panoramic dental X-ray images. Various processes involved in the dental input image are 1. Preprocessing, 2. Segmentation, 3. Model Construction and 4. Evaluating. A considerable part is given to image preprocessing to segment teeth with reasonable accuracy, and as a consequence, a good research study must be done.

keywords: Caries Detection • Segmentation Process • Panoramic X-Ray • Feature Extraction

الملخص

يلخص هذا التقرير عن فترة تدريب التخرج لمدة ستة أشهر في AI Diagnosis Vision . تسوس الأسنان هو مرض يصيب الأنسجة الصلبة للأسنان بسبب تأثير الكائنات الحية الدقيقة الموجودة في صفات الكربوهيدرات القابلة للتخرّم (السكريات بشكل أساسي). نتيجة لذلك، فإن اكتشاف تسوس الأسنان في مرحلة المبكرة أمر بالغ الأهمية. يخدم هذا المشروع غرضين مهمين: أولاً، تنفيذ عملية مخصصة لتجزئة الأسنان لاستخراج الميزات. تحتوي البيانات على ٢٣٧ صورة بأنواع مختلفة من الضوابط والبيانات الضوئي. ثانياً، دراسة حالة كاملة للكشف عن تسوس الأسنان وتقسيمه في صور الأشعة السينية البانورامية. المراحل المختلفة المتضمنة في تحليل صور الأسنان هي كما يلي: ١. المعالجة المسبقة ، ٢. التجزئة ، ٣. بناء النموذج و ٤. تقييم النموذج. يتم منح جزء كبير من المعالجة المسبقة للصور لقطع الأسنان بدقة ، ولذلك ، يجب إجراء دراسة بحثية جيدة.

الكلمات المفتاحية: كشف تسوس الأسنان • عملية التجزئة • الأشعة السينية البانورامية • استخراج الميزات

Résumé

Ce rapport résulte d'un stage de fin d'études de six mois au sein d'AI Diagnosis Vision. La carie dentaire est une maladie des tissus durs des dents causée par l'action des micro-organismes présents dans la plaque dentaire sur les glucides fermentescibles (principalement les sucres). Par conséquent, il est essentiel de détecter les caries dentaires à leurs premiers stades. Ce projet répond à deux objectifs importants : Premièrement, la mise en œuvre d'un processus personnalisé de segmentation des dents pour l'extraction de caractéristiques. L'ensemble de données contient 237 images avec différents types de bruit et de contraste. Deuxièmement, présenter une étude de cas complète pour détecter et segmenter les caries dentaires dans des images radiographiques panoramiques. Les différents processus impliqués dans l'image dentaire d'entrée sont les suivants : 1. Prétraitement, 2. Segmentation, 3. Construction du modèle et 4. Evaluation. Une part considérable est accordée au prétraitement de l'image pour segmenter les dents avec une précision raisonnable, en conséquence, une bonne étude de recherche doit être faite.

Mots clés: Détection des caries • Processus de segmentation • Radiographie panoramique • Extraction de caractéristiques