# Detecting and Labeling Panoramic Tufts Dental Database with Detectron 2 Mask R-CNN

**Mina Pourostad** [1] **Wayne Zhu** [1]

## Abstract

Tooth detection and numbering Dental panoramic radiographs, aka 360-degree x-ray dental images, are commonly analyzed during diagnostics before any targeted treatment can be given. This research is built upon a stream of existing research of region-based convolutional neural networks (R-CNN) automating the detection and numbering of teeth with performance that rivals human experts. Our main contributions are listed as follows: (1) A new dataset consisting of 1000 images—Tufts Dental Database—which includes not only bounding boxes but also unprecedented pixel-level tooth segmentation labeling was analyzed. (2) Comparing the success and failure of training with R-CNN or other models reveals the importance of the relative position of each tooth in prediction. (3) By using Detectron 2's Mask R-CNN model with backbone ResNet 101, we achieve superior accuracy compared to the predecessor's result in almost all areas. (4) We also accomplish tooth segmentation numbering—which to our knowledge has not been done by previous studies.

## 1. Introduction and Problem Definition

The problem of detecting and numbering teeth via dental panoramic radiographs (Figure 1) has numerous medical applications. Any attempt for the diagnosis of dental disease which often precedes dental procedures like surgical planning, postoperative assessment, and dental implants should start with tooth detection and tooth numbering as every finding must be assigned to the related tooth(Pongrácz & Bárdosi, 2006). Modern-day doctors and dentists, however, already spend excessive time on paperwork and other important yet menial tasks like this. Research in this area provides ample economic and social benefits for dentists to rely on and verify computer-aided teeth detection and numbering systems. This study applies the existing machine learning algorithm—Mask R-CNN—on a new dataset, Tufts Dental Database, and achieved higher accuracy than researches before.

The broad task can be further divided into two separate goals: (1) Numbering—predicting which tooth number the radiographed tooth should adopt. (2) Detection—distinguishing the tooth from the background. Subsequently for either numbering or detection, one could perform analysis based on the following labels: (1) tooth segmentation label that shows the tooth's exact boundary, or (2) rectangular bounding box label. Note that tooth numbering is often done using bounding boxes while tooth detection comparison is often pixel-level, but the other way is also possible. Tooth segmentation detection is synonymous with the more general computer vision term binary image segmentation, while tooth segmentation numbering corresponds to multiclass image segmentation. There are two common methods for numbering the tooth, the Universal tooth numbering (UTN) system and Fédération Dentaire Internationale (FDI) system (Figure 2). We used the UTN method for tooth numbering. (Mahdi et al., 2020)

Earlier datasets often didn't include exact object segmentation labels; also, the most advanced model at that time—Faster R-CNN—was only capable of categorizing bounding boxes. It was not until quite recently that object segmentation became incorporated into the latest research. In fact, even papers that use Mask-RCNN in 2021 still only conducted bounding boxes level research due to the constraint of the dataset. Tufts Dental Database's availability in 2022 allowed us to conduct tooth segmentation research. (Prados-Privado et al., 2021) We would also perform bounding box tooth numbering as a direct comparison to other preceding research using Faster R-CNN.

Subsequent sections are divided into the following:

- *Literature Review for R-CNN Algorithms* provides the history and relevance of R-CNN model variations.
- *Literature Review for Tooth Detection and Numbering* report the most closely related studies on tooth

[*]Equal contribution [1]Department of Computer Science, Courant Institute, New York University, New York, United States. Correspondence to: Firstname1 Lastname1 <first1.last1@xxx.edu>.

*Proceedings of the $39^{th}$ International Conference on Machine Learning*, Baltimore, Maryland, USA, PMLR 162, 2022. 

*Figure 1.* Example dental panoramic radiograph

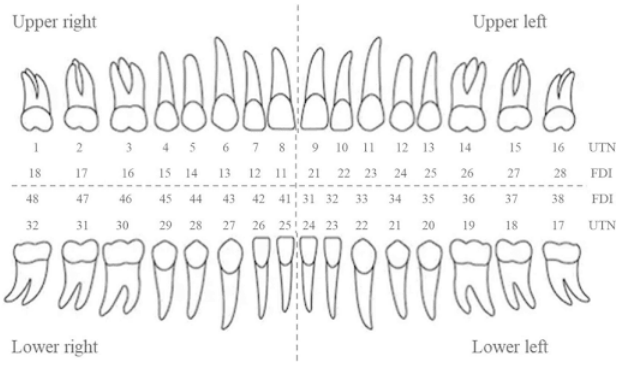


*Figure 2.* TUN and FDI tooth labeling standard

numbering and detection.

- *Challenges and Solutions* focuses on preprocessing and other early attempts with other models.
- *Method and Implementation* explains the design and implementation of the experiments.
- *Experiment Outcome* compares our experiment results to the benchmark and analyzes them.
- *Discussion and Conclusion* summarize the paper.

## 2. Literature Review for R-CNN Algorithms

Because the two of us are both new to machine learning and computer vision, extensive background research has been conducted to determine which model and technique to choose. People familiar with computer vision would know most of the concepts in this section already. Object detection generic model typically consists of roughly 3 phases in a pipeline as in Figure 14: Region proposal generator determines where on the image the object is located. Feature extraction converts the image into feature vectors. The classification part assigns a label to the proposed object.(Gad, 2020)

The most popular models in object recognition and tooth detection and numbering, in particular, are all R-CNN-based models, so it becomes necessary to describe the history of the development of R-CNN here. In 2014, researchers at UC Berkeley developed a deep neural network called Region-based Convolutional Neural Network (R-CNN) that can detect 80 different types of objects. Compared to the generic pipeline of object detection techniques, the main contribution of R-CNN is just extracting the features based on a convolutional neural network (CNN). (Gad, 2020; Girshick et al., 2014a) The network achieves a milestone in accuracy but not speed. The evolution of R-CNN to Fast R-CNN to Faster R-CNN to Mask R-CNN can be seen in Figure 17.

Then in 2015, Fast R-CNN is developed. It uses the ROI Pooling layer—a max pooling technique to reduce the amount of information in the feature vector from each regional proposal—to accelerate the model calculation. ROI pooling converts all proposals to fixed shapes. (Girshick et al., 2014b; Girshick, 2015a; Girshick et al., 2014b) Thus, we will have the same feature map for all the proposals. So we can pass the image to CNN one time instead of one time for each proposal. It's more accurate and requires little caching compared to R-CNN. However, fast R-CNN does not improve the time-consuming selective search algorithm of region proposals. (Gad, 2020)

Faster RCNN improves upon Fast R-CNN with a Region Proposal Network (RPN) that is fully convolutional that implements the neural network with attention. It also shares its network with the downstream detection part, so it costs very little additional time. It then uses reference anchor boxes to match regions and detect objects of different scales. Faster R-CNN is basically a combination of RPN and Fast R-CNN, and it allows for faster region proposals. The regional proposal network of R-CNN is pivotal when it comes to tooth detection since teeth numbering is highly positional. The subpar performance of models without this network would be illustrated later. For this reason, most background research about teeth detection and numbering was done using Faster R-CNN. (Gad, 2020; Girshick, 2015b)

Mask R-CNN takes Faster R-CNN a step further by not only outputting the bounding box but also the exact pixel(Farooq, 2015). The researchers add a branch that outputs a binary mask that determines whether each pixel is part of an object. Note that this new branch occurs in parallel with the existing branch that predicts the bounding boxes, so the extra time consumption is minimum.(He et al., 2017)

As for the backbone, the classical R-CNN used ResNet of depth 50 or 101 layers. They can be chosen in combination with a more effective generic feature extractor which is called Feature Pyramid Network (FPN)(Lin et al., 2017) FPN uses a top-down architecture with lateral connections

to build a large-scale semantic feature map. The outcome is a feature pyramid with rich semantics at all levels and is built quickly(Tsang, 2019).This improves both accuracy and speed.

Facebook Detectron2's implementations of RCNN algorithms are used in this study. Detectron2 is a research platform and a production library for object detection providing faster, and more accurate models in Pytorch. A few important components and features of notes are: the model zoos mean different kinds of pre-trained models and parameters are available. Modular architecture means different models can be easily imported and extended when needed. It's completely open-source, and its accuracy and speed dwarf its competitors(Wu et al., 2019). For example, for our model of interest—Mask R-CNN, it's clear that Detectron2's implementation training throughput and accuracy far surpass the original as shown in Figure 15 Figure 16.

## 3. Literature Review for Tooth Detection and Numbering

Quite a few comprehensive researches have been done applying the above-mentioned high-performance R-CNN variants—especially Faster R-CNN—on dental panoramic data. Only one of the four groups listed here—Tufts Dental Database researchers—conducted detection with pixel-level segmentation because that label is not available in other older datasets to our knowledge. Instead, older researchers did both the numbering and detection using bounding boxes. Tufts Dental Database researchers did not number the teeth, which we did both with bounding boxes and pixel-level segmentation labels. The direct comparison of the most important differences between past studies and our study are listed in our own Figure 3 below. The paragraphs highlight and explain them in a detailed but perhaps verbose way.

Group 1 (Mask R-CNN): Researchers from Spain took 8000 panoramic radiographs and conducted tooth detection and numbering with *Mask R-CNN with ResNet 101*(Prados-Privado et al., 2021). The team manually and extensively filtered out images of children, with no teeth, with implants, and with poor definitions and ended up using 1217 samples for training and validation(no separate test set). The best result is documented with an accuracy of 93.83% for tooth numbering and 99.24% for tooth detection. This paper, however, suffers from a few drawbacks. Although they used Mask R-CNN just like ours, they only output bounding boxes, not pixel-level object segmentation. It's perplexing why Mask RCNN is chosen since image segmentation is generated but is not examined in the final result. Perhaps, that is due to a lack of benchmark. Also, the intersection over union (IoU) threshold that is required to determine whether 2 tooth number labels overlapped sufficiently is missing. Also, it is unlikely that real-world images would look like the training and validation since 80% of all, especially those non-typical radiographs, are filtered out manually.

Group 2 (Extensive preprocessing and postprocessing): This international team used Faster R-CNN with the classical VGG-16 CNN backbone to achieve tooth detection and numbering at 99% recall and precision on 1352 proprietary dental images (Tuzoff et al., 2019). They conducted heavy post-processing by sorting and counting any missed tooth and trying out every combination of tooth numbers for the highest total confidence score. Images were also augmented but conserved neighboring content. Image augmentation alone increases the precision by 2 percent but including additional context around the cropped teeth increases the precision by 6 percent. This reveals the importance of relative tooth position in tooth numbering. IoU intersection threshold was set at "substantial" with no clearly defined numerical value, which makes direct comparison and benchmarking difficult. Then, they applied logistics filtering. Finally, the group achieved both tooth detection and tooth numbering precision at 98 percent and recall above 99 percent. Molars were mainly misclassified by both machines and experts because some people have fewer or no wisdom teeth.

Group 3 (Extensive algorithmic post-processing): Another group from China used Faster R-CNN to conduct teeth detection and numbering with a highlight on the performance difference before and after postprocessing(Chen et al., 2019). Before post-processing, the tooth detection F1 score is 0.941, and the tooth numbering F1 score is 0.747. This will serve as a good benchmark for us. They propose 3 postprocessing techniques: (1) A filtering algorithm that deletes overlapping boxes associated with the same tooth. (2) A neural network that detects missing teeth. (3) A rule-based module that identifies counterintuitive predictions. An output measurement is based on the threshold intersection over union(Iou) of 0.5, and a dramatic increase of F1 Score after postprocessing is observed like previous research—see Figure 18. This highlights the importance of post-processing in achieving a production-level system.

Group 4 (Tufts Dental Database): Tufts Dental Database (TDD) was impressive in several ways: It came out in 2022 and provided the most comprehensively labeled sets of panoramic radiographs compared to its predecessors including not only bounding boxes but also precise tooth segmentation(Panetta et al., 2021). Tufts researchers highlighted and compared a few image enhancement techniques, which could be used during preprocessing of other future research. TDD researchers also divided data into training and validation sets only and perform Mask R-CNN pixel-level tooth detection on FPN ResNet50, which would serve as a benchmark for our experiment with Mask R-CNN+Resnet101+FPN later.(See Figure 19)

| | Model Used | Numbering with Bbox | Numbering with Tooth Segmentation | Detection with Bbox | Detection with Tooth Segmentation | Extensive Preprocessing or Filtering | Extensive Postprocessing | Test set not seen during training | IoU Threshold for Bbox |
|---|---|---|---|---|---|---|---|---|---|
| Group 1 | Mask R-CNN | Yes | No | Yes | No | Yes | No | No | Not Mentioned |
| Group 2 | Faster R-CNN | Yes | No | Yes | No | No | Yes | Yes | "Substantial" |
| Group 3 | Faster R-CNN | Yes | No | Yes | No | No | Yes | Yes | 0.5 |
| Group 4 | Mask R-CNN | No | No | No | Yes | No | No | No | N/A |
| This Paper | Mask R-CNN | Yes | Yes | No | Yes | Yes | No | Yes | 0.5 |

*Figure 3.* Comparison of all group's results

## 4. Challenges and Solutions

### 4.1. Detection Transformer Attempt

Originally, we planned to use a new transformer-based model object recognition model DETR which treats object recognition as a set prediction problem(Carion et al., 2020). It was not used in any dental research previously. The model is completely different from R-CNN and performs both object proposal and classification with a CNN network and a transformer as in Figure 20. However, it soon becomes apparent that DETR is not good at the task at hand—in fact, our result shows consistent low confidence prediction and large overlapping bounding boxes that fail to differentiate between different teeth as in Figure 4. This corresponds to the existing claim by DETR authors that the model is good at predicting large objects of distinct shapes but is not good at predicting small objects. It also shows that the most important aspect of tooth numbering and detection is the tooth's relative position, and the lack of an explicit regional proposal network of DETR likely results in its failure. We left it to the more mathematically-oriented future researchers to further analyze.

### 4.2. Data Preprocessing

TDD included both images for polygon tooth masks (example as in Figure 5) and JSON files describing teeth polygons and teeth bounding boxes vertices for each radiograph image. We started by overlapping the tooth mask images with unlabelled radiographs to establish the ground truth label,

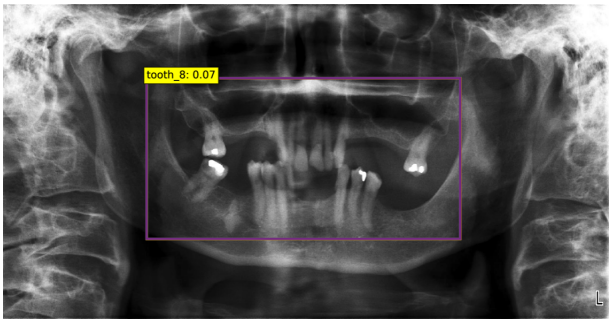


*Figure 4.* Detr's final prediction shows one overlapping huge bounding box with low confidence

but we noticed that, in some cases, the mask is missing some critical tooth that exists on the unlabeled radiograph as in Figure 6. So we had to reconstruct the masks from JSON vertices and created new teeth mask data to establish the actual ground truth. Furthermore, we realized that around 70 out of 1000 images are pediatric dental images, which contain deciduous teeth as in Figure 7. 70 images are too little to make an accurate prediction for those deciduous teeth but too large a number to ignore, so we remove those children's radiographs.

### 4.3. Data augmentation

The limited size of the dataset has long been an issue for tooth numbering and detection. Thus, in the first round of training, we used data augmentation—flipping images hori-

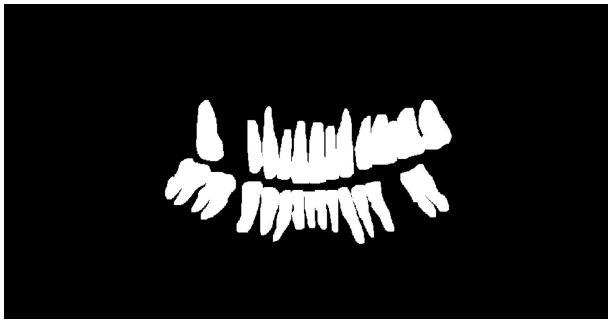

*Figure 5.* An example of tooth mask image corresponding to a radiograph

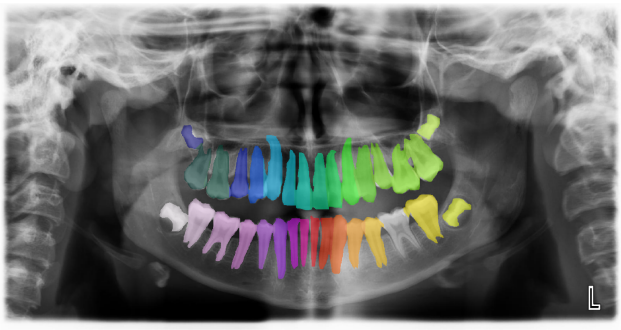

*Figure 6.* An example of missing mask for tooth number 19 in Tufts Dental Database after we overlap tooth mask image with the radiograph

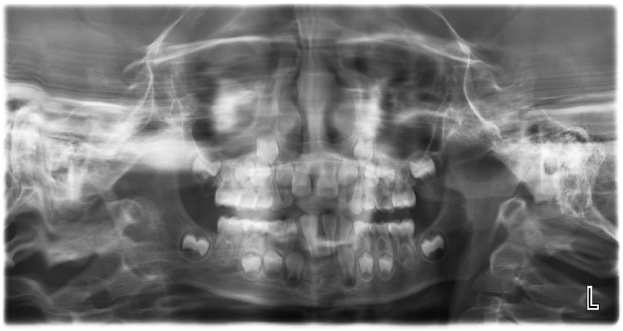

*Figure 7.* an example of a pediatric dental image removed from the dataset

zontally or vertically or cropping—to increase the number of data. We didn't get good results. Recalling from the failure of DETR and the literature review that the relative position of each tooth is extremely crucial in generating the final outcome, the loss of relative position during the data flipping and cropping likely results in lower accuracy and slower convergence. Once we realize this and train without data augmentation, the resulting accuracy started to improve. Future researchers could attempt data augmentation but they should include the neighboring tooth of the cropped tooth and change the bounding box vertices accordingly. This will, however, likely be labor-intensive.

## 5. Method and Implementation

Given previous studies have been done extensively on Faster R-CNN with a smaller ResNet backbone, Facebook Detectron 2 library's models Mask R-CNN with FPN and backbone ResNet 101 was chosen for its extensiveness and accuracy. For the Detectron2 implementation of Mask R-CNN, we converted the TDD's JSON label format to COCO JSON Format—a format commonly used in object detection. After the data preprocessing, 915 out of 1001 initial Tufts Dental Database images are chosen as validly labeled. 915 images were further divided into 735 images for training (80%), 91 for validation(10%), and 89 for testing (10%) which will not be seen at all by the model during training. We trained our data without data augmentation for 50000 iterations (125 epochs) with a batch size of 2 and a learning rate of 0.00025 using RTX4000 GPU.

Google Colab environment was considered originally but its filesystem is not persistent. The training and inference were conducted instead on the Coreweave platform—which provided a Quadro RTX 4000 GPU—with a Jupyter Lab notebook. The Region of Interest head batch size is set at 128, and the number of classes is 32, which is equivalent to the maximum number of teeth an adult can have. The validation set ran for 100 times during training and its incremental accuracy is recorded and will be reported below.

## 6. Experiment Outcome

### 6.1. Training and Validation Result and Analysis

Since Mask R-CNN was used for pixel-level tooth segmentation detection, we see the following Figure 8 for Mask R-CNN's validation accuracy. The accuracy increases at first quickly but then slowly at the end to reach. It reaches 93% in the end. The accuracy is comparable to the Tufts Dental Database's pixel accuracy of 95.08%. This result makes sense considering Tufts researchers ran 150 epochs instead of the 125 epochs.

We used Mask R-CNN's Faster R-CNN part for bounding

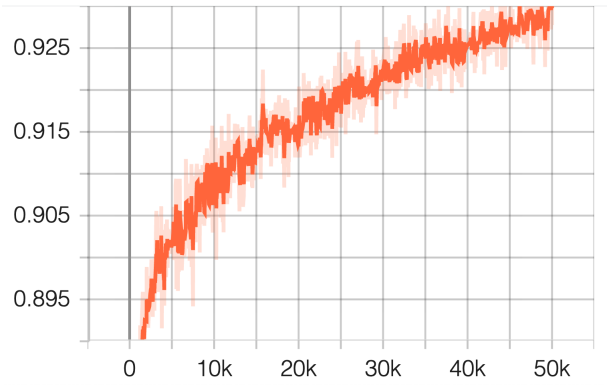


*Figure 8.* Multiclass tooth segmentation aka tooth segmentation detection. X-axis: Num of iterations. Y-axis: Pixel-accuracy

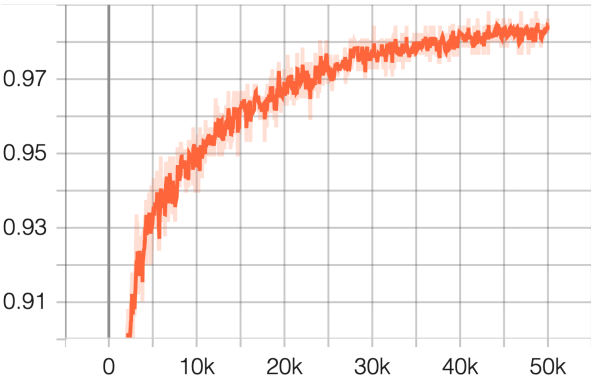


*Figure 9.* Tooth Numbering with bounding boxes. X-axis: Num of iterations. Y-axis: Bounding Box classification Accuracy

box numbering. The following Figure 9 for Faster R-CNN's validation set accuracy shows the same trend as Mask R-CNN and reaches an astonishing 98% accuracy near the end. This suggests some level of overfitting to the validation and training set as this validation accuracy is higher than test accuracy. The numbering accuracy is also higher than the previous detection task, which is supposed to be easier. The IoU threshold, usually 50% for numbering tasks and not applicable to the pixel-level detection task, might contribute to the seemingly high accuracy while it's much harder to get the pixel to align entirely.

If we look at the average precision for numbering prediction with an IoU threshold of 50 as in this Figure 10, we see that the precision peak at around 10K iterations. Then, it didn't increase much or even decreased. This suggests that average recall must have increased to contribute to the overall increase in classification accuracy. This suggests that the quality of positive prediction stays the same while a larger percentage of positive values were identified.

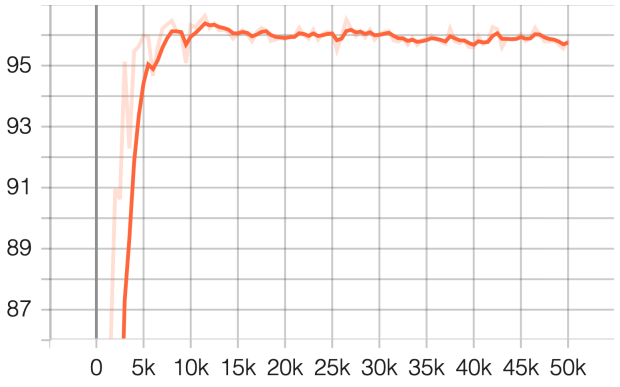


*Figure 10.* Average precision for tooth numbering with IoU 50. X-axis: Num of iterations. Y-axis: Average precision

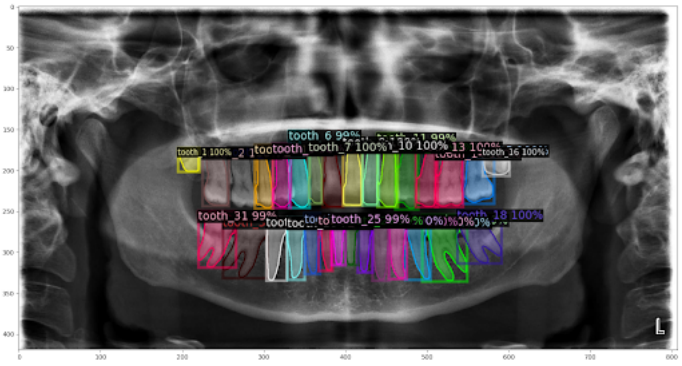

*Figure 11.* Prediction of an individual with all teeth

### 6.2. Testing Result Analysis

The graphic results are reassuring. In Figure 11, we see all the teeth detected and labeled both with bounding boxes and pixel-level segmentation. The confidence score is also very high.

Even for the edge cases in Figure 12 where the person has an incomplete set of teeth, the prediction is still accurate with no extra bounding boxes.

As shown in this comprehensive figure here Figure 13, our study's bounding box numbering average precision surpasses our benchmark, Group 3's, by 0.2 for the same IoU threshold. The prediction for both groups is for data not post-processed. Our average recall for the same threshold 50 is not computed but is definitely not much lower. Even the IoU 50:95 average recall(taking the average recall with IoU threshold 50:95 with an increase of 5 at a time), which tends to be a much lower statistic than the IoU 50 measure, is at 0.7.

For the pixel-level tooth segmentation task, Tufts Dental Database researchers (Group 4)—with whom we shared the

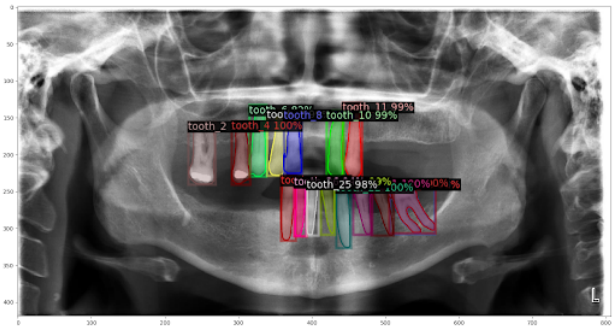


*Figure 12.* Prediction of an individual with fewer teeth

same dataset—outperformed us by a tiny margin. Nonetheless, a key distinction has to be made that their result is obtained from the validation set, not the test set. So in any real-world dataset, our model likely will much outperform theirs.

Finally, for our unique contribution, we conducted unprecedented pixel-level segmentation numbering of teeth with a precision of 0.93. This pixel-level outcome has an IoU threshold of 50% because it's evaluated tooth by tooth. Though the average precision is similar to the bounding box numbering outcome, the overall recall and precision for the higher threshold are much lower. Indeed, this is especially the case for molars, where there are fewer data and the average precision with higher IoU threshold using segmentation labels is on average 0.05 lower than using bounding box labels.

Because pixel-level segmentation detection is far superior to bounding box tooth detection, the latter was no longer necessary.

## 7. Discussion and Conclusion

Our research builds on top of the previous studies of tooth detection and contributes a few key insights. First, we verify the high quality of the Tufts Dental Database which contains not only bounding boxes but also unprecedented segmentation masks while acknowledging that some of its image masks are incomplete. Secondly, we learn from our failure using DETR and data augmentation and the success of R-CNN that the relative position of each tooth is an extremely crucial piece of information when it comes to tooth detection and prediction. Thirdly, we use Detectron 2's implementation of the state-of-the-art Mask R-CNN+Resnet101+FPN to achieve accuracy and precision surpassing our predecessors. Finally, we train and produce models capable of pixel-level segmentation numbering that is unprecedented to our knowledge. This study provides future researchers with ample guidance on how to prepare and train models for further post-processing in order to achieve enterprise-level accuracy that can be used in actual clinical settings to diagnose tooth abnormalities.

| | Bbox Numbering Avg Precision | Bbox Numbering Avg Recall | Pixel-lv Segmentation Detection F-1 | Pixel-lv Segmentation Numbering Precision | Pixel-lv Segmentation Numbering Recall | Model | Test Set Not Seen During Training |
|---|---|---|---|---|---|---|---|
| Group 3 | 0.715 (IoU 50) | 0.782 (IoU 50) | N/A | N/A | N/A | Faster RCNN Resnet v2 Atrous version | Yes |
| Group 4 | N/A | N/A | 0.923 | N/A | N/A | Mask RCNN ResNet50+FPN | No |
| This Paper | 0.935 (IoU 50) | 0.705 (IoU 50:95 avg) | 0.919 | 0.938 (IoU 50) | 0.644 (IoU 50:95 avg) | Mask RCNN ResNet101+FPN | Yes |

*Figure 13.* Table Comparing Test Result with Benchmark

## References


Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Chen, H., Zhang, K., Lyu, P., Li, H., Zhang, L., Wu, J., and Lee, C.-H. A deep learning approach to automatic teeth detection and numbering based on object detection in dental periapical films. *Scientific reports*, 9(1):1–11, 2019.

Farooq, U. From r-cnn to mask r-cnn. *Medium*, 2015.

Gad, A. F. Faster r-cnn explained for object detection tasks. *Paperspace Blog*, 2020.

Girshick, R. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015a.

Girshick, R. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015b.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014a.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014b.

Gonzalez, S., Arellano, C., and Tapia, J. E. Deepblueberry: Quantification of blueberries in the wild using instance segmentation. *Ieee Access*, 7:105776–105788, 2019.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Mahdi, F. P., Motoki, K., and Kobashi, S. Optimization technique combined with deep learning method for teeth recognition in dental panoramic radiographs. *Scientific Reports*, 10(1):1–12, 2020.

Panetta, K., Rajendran, R., Ramesh, A., Rao, S. P., and Agaian, S. Tufts dental database: A multimodal panoramic x-ray dataset for benchmarking diagnostic systems. *IEEE Journal of Biomedical and Health Informatics*, 26(4): 1650–1659, 2021.

Pongrácz, F. and Bárdosi, Z. Dentition planning with image-based occlusion analysis. *International Journal of Computer Assisted Radiology and Surgery*, 1(3):149–156, 2006.

Prados-Privado, M., García Villalón, J., Blázquez Torres, A., Martínez-Martínez, C. H., and Ivorra, C. A convolutional neural network for automatic tooth numbering in panoramic images. *BioMed Research International*, 2021, 2021.

Tsang, S. Review: Fpn—feature pyramid network (object detection). 2019.

Tuzoff, D. V., Tuzova, L. N., Bornstein, M. M., Krasnov, A. S., Kharchenko, M. A., Nikolenko, S. I., Sveshnikov, M. M., and Bednenko, G. B. Tooth detection and numbering in panoramic radiographs using convolutional neural networks. *Dentomaxillofacial Radiology*, 48(4): 20180051, 2019.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

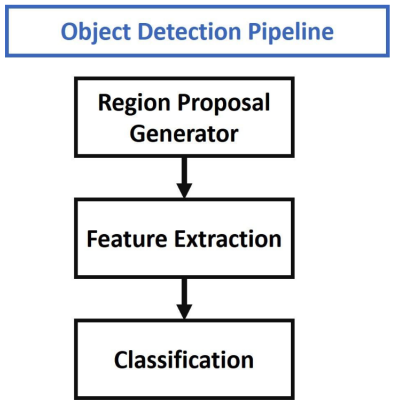


*Figure 14.* RCNN Pipeline
(Gad, 2020)

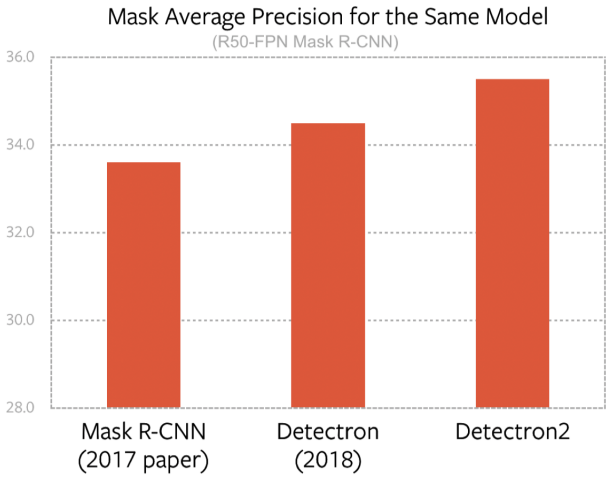


*Figure 15.* Detectron2 Mask R-CNN MAP comparison

## A. You *can* have an appendix here.

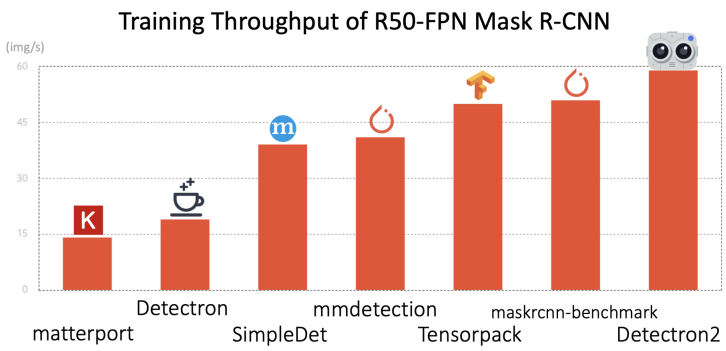


*Figure 16.* Detectron2 Mask R-CNN training throughput comparison

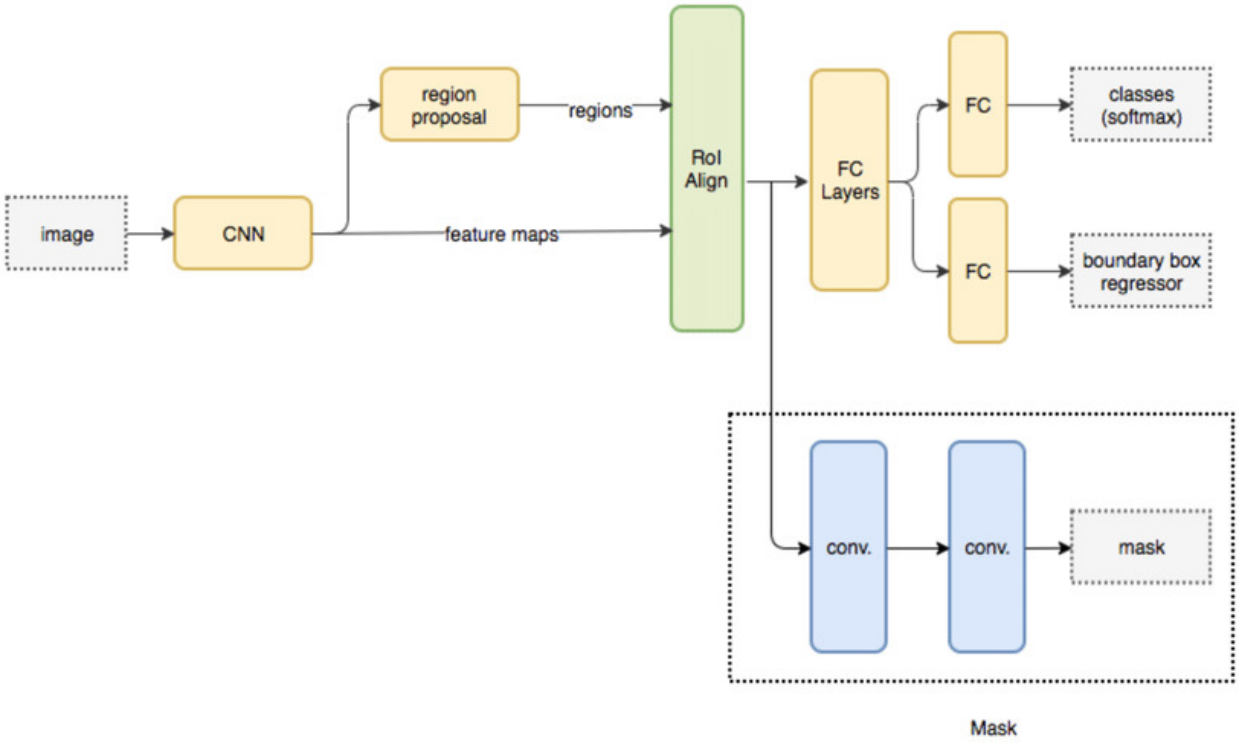


*Figure 17.* Mask RCNN Architecture (Gonzalez et al., 2019)

| Object | AS* | | | | Human Experts | | | Prior work[26] |
|---|---|---|---|---|---|---|---|---|
| | stage 1** | stage 2** | stage 3** | stage 4** | A | B | C | |
| Test images | 250 | 250 | 250 | 250 | 250 | 250 | 250 | 250 |
| GT* boxes exist | 871 | 871 | 871 | 871 | 871 | 871 | 871 | 871 |
| Box detected | 953 | 868 | 868 | 868 | 869 | 866 | 873 | 822 |
| Detection prec* | 0.900 | 0.988 | 0.988 | 0.988 | 0.993 | 0.991 | 0.995 | 0.838 |
| Detection recall | 0.985 | 0.985 | 0.985 | 0.985 | 0.991 | 0.985 | 0.998 | 0.791 |
| Mean IOU | $0.91 \pm 0.04$ | $0.91 \pm 0.04$ | $0.91 \pm 0.04$ | $0.91 \pm 0.04$ | $0.92 \pm 0.05$ | $0.90 \pm 0.05$ | $0.92 \pm 0.05$ | $0.81 \pm 0.06$ |
| Numbering prec* | 0.715 | 0.797 | 0.897 | 0.917 | 0.938 | 0.930 | 0.975 | 0.771 |
| Numbering recall | 0.782 | 0.794 | 0.894 | 0.914 | 0.936 | 0.924 | 0.977 | 0.728 |

**Table 3.** The precision, recall, and IOU of detected box on test dataset. *AS = our automatic teeth detection and numbering system, GT = ground truth, prec. = precision. **Stage 1: teeth bounding boxes detected by trained faster R-CNN; stage 2: after deleting overlapped boxes; stage 3: after matching with template; stage 4: after predicting missing teeth and matching with template.

*Figure 18.* Group 3's reported result

TABLE IV

SUMMARY OF THE PERFORMANCE OF THE STATE-OF-THE-ART ALGORITHMS IN SEGMENTING TEETH FROM PANORAMIC RADIOGRAPHS. THE UNET ARCHITECTURE PERFORMS MARGINALLY BETTER IN COMPARISON

| Model | VGG19 | | | ResNet18 | | | ResNet50 | | |
|---|---|---|---|---|---|---|---|---|---|
| | PA (%) | IoU (%) | Dice (%) | PA (%) | IoU (%) | Dice (%) | PA (%) | IoU (%) | Dice (%) |
| FPN | 95.11 | 86.39 | 92.30 | 95.17 | 86.37 | 92.24 | 95.08 | 86.42 | 92.29 |
| UNet | **95.22** | **86.67** | 92.46 | 95.11 | 86.42 | 92.27 | 95.13 | 86.49 | 92.36 |
| UNet++ | 95.19 | 86.62 | 92.47 | 95.15 | 86.54 | 92.43 | 95.16 | 86.62 | **92.49** |
| PSPNet | 95.00 | 86.08 | 91.77 | 94.76 | 85.66 | 91.49 | 94.95 | 86.04 | 91.70 |
| DeepLabV3 | - | - | - | 94.91 | 86.02 | 91.87 | 95.00 | 86.21 | 92.00 |
| DeepLabV3+ | - | - | - | 95.13 | 86.41 | 91.80 | 95.07 | 86.33 | 92.29 |

| | PA (%) | IoU (%) | Dice (%) |
|---|---|---|---|
| nnUNet** | 94.91 | 86.11 | 90.86 |

| | 150 epochs | | | 250 epochs | | | 400 epochs | | |
|---|---|---|---|---|---|---|---|---|---|
| | PA (%) | IoU (%) | Dice (%) | PA (%) | IoU (%) | Dice (%) | PA (%) | IoU (%) | Dice (%) |
| CE-Net** | 86.52 | 71.13 | 76.92 | 91.68 | 79.75 | 84.98 | 92.67 | 81.64 | 86.62 |

** No backbone and pretrained weights used.

*Figure 19.* Group 4's reported result

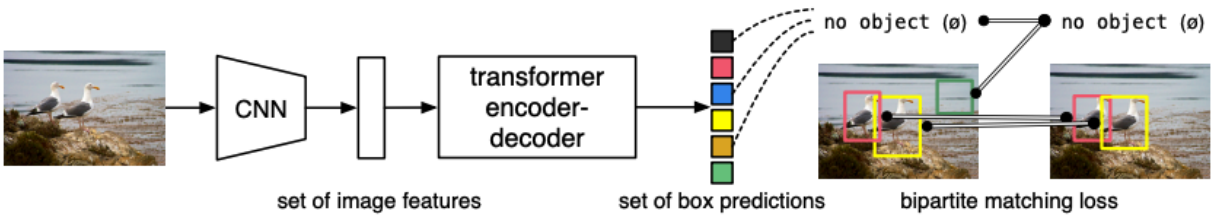


Fig. 1: DETR directly predicts (in parallel) the final set of detections by combining a common CNN with a transformer architecture. During training, bipartite matching uniquely assigns predictions with ground truth boxes. Prediction with no match should yield a "`no object`" ($\varnothing$) class prediction.

*Figure 20.* DETR's architecture