BMC Oral Health

RESEARCH

Open Access

Automatic detection of developmental stages of molar teeth with deep learning

Ertuğrul Furkan Savaştaer[1], Berrin Çelik[2] and Mahmut Emin Çelik[1,3*]

Abstract

Background The aim was to fully automate molar teeth developmental staging and to comprehensively analyze a wide range of deep learning models' performances for molar tooth germ detection on panoramic radiographs.

Methods The dataset consisted of 210 panoramic radiographies. The data were obtained from patients aged between 5 and 25 years. The stages of development of molar teeth were divided into 4 classes such as M1, M2, M3 and M4. 9 different convolutional neural network models, which were Cascade R-CNN, YOLOv3, Hybrid Task Cascade(HTC), DetectorRS, SSD, EfficientNet, NAS-FPN, Deformable DETR and Probabilistic Anchor Assignment(PAA), were used for automatic detection of these classes. Performances were evaluated by mAP for detection localization performance and confusion matrices, giving metrics of accuracy, precision, recall and F1-scores for classification part.

Results Localization performance of the models varied between 0.70 and 0.86 while average accuracy for all classes was between 0.71 and 0.82. The Deformable DETR model provided the best performance with mAP, accuracy, recall and F1-score as 0.86, 0.82, 0.86 and 0.86 respectively.

Conclusions Molar teeth were automatically detected and categorized by modern artificial intelligence techniques. Findings demonstrated that detection and classification ability of deep learning models were promising for molar teeth development staging. Automated systems have a potential to alleviate the burden and assist dentists.

Trial registration This is retrospectively registered with the number 2023–1216 by the university ethical committee.

Keywords Tooth germ, Detection, Staging, Artificial intelligence, Deep learning, Panoramic, Dentistry

Background

In the evaluation of growth and development in children, the morphology of oral structures, especially the development of permanent tooth germs, are observed as maturation indicators [1]. Timely identification of these structures not only helps to assess dental age and growth, but also provides an important basis for individualized treatment planning [2]. The developmental stages of tooth germs can be evaluated from panoramic radiographs or calculated from the eruption age by intraoral examination of permanent teeth [3]. Evaluation of permanent tooth germs in developmental staging is frequently used in tooth age estimation because it can be applied in a wide age range and is less affected by environmental factors [4].

Age estimation also becomes very important in forensic medicine when the patient's real age cannot be proven, and identification information is not available [5]. In dental age estimation, the manual estimation of developmental stages is disadvantageous due to the variability in the classification of observers [6]. The use of automatic age classification methods has been proposed

*Correspondence:
Mahmut Emin Çelik
mahmutemincelik@gazi.edu.tr
[1] Electrical Electronics Engineering Department, Faculty of Engineering, Gazi University, Ankara, Turkey
[2] Oral and Maxillofacial Radiology Department, Faculty of Dentistry, Ankara Yıldırım Beyazıt University, Ankara, Turkey
[3] Biomedical Calibration and Research Center, Gazi University, Ankara, Turkey

in recent years to reduce the variability in stages [7]. In a pilot study on automated staging of third molar germs for tooth age estimation, De tobel et al. reported an overall performance similar to that of staging by human observers [8].

Deep learning methods, particularly convolutional neural networks (CNNs), have demonstrated remarkable capabilities in pattern recognition, enabling the automated identification, localization, and segmentation of a wide range of cases in dentistry [9–16]. Furthermore, deep learning algorithms can adapt to variations in image acquisition techniques and patient demographics, contributing to a more versatile diagnostic toolkit. These algorithms hold the promise of expediting diagnoses, enabling early intervention, and facilitating a more patient-centric approach to dental care. Leveraging this potential, researchers are exploring the integration of deep learning algorithms into dental radiology for the precise identification of tooth germs [8, 17–19].

Developmental staging of all molars was evaluated to automate the processes for age determination. The molars in panoramic images were automatically detected and the developmental stage was classified. The aim of this study is to evaluate the performance of different deep learning models for the detection of permanent molar germ development stages in pediatric and adult panoramic radiographs. 9 different deep learning models were analyzed, and model performances were evaluated using well known detection metric mean average precision (mAP), accuracy, precision, recall and F1-score metrics.

Methods

Data preparation

The retrospective work was approved by the University Ethics Committee (no: 2023–1216). Patients between the ages of 5 and 25, who had panoramic radiographs taken for different reasons between 2022 and 2023 from the Radiology archives of Ankara Yıldırım Beyazıt University Faculty of Dentistry, were included in the study. Systemic and congenital diseases, cleft lip and palate, multiple tooth eruption disorders and delayed eruption, any cyst or tumor in the jaw region, tooth agenesis and a history of endodontic treatment on the relevant permanent molars and images with artifacts that would affect the evaluation were determined as exclusion criteria.

The developmental stages of permanent molars were modified according to Haavikko's classification and divided into 4 classes (M1-4), as shown in Fig. 1 [20].

In accordance with this classification, the developmental stages of all permanent molars were evaluated using panoramic radiographs. In case of differences in the formation of each root (for example, if the mesial root was formed early and the other roots were formed late), the stage with the slowest root formation was used. For panoramic radiographs used for evaluation, dental panoramic device (Planmeca, Helsinki, Finland) was used with 60–70 kVp, 5–12.5 mA, exposure time of 13.8–16 s.

The dataset included 210 panoramic radiographs. were collected. Nine state-of-the-art deep learning models were trained with the train data and the models were checked every five epochs with the validation data. It was ensured that all classes were balanced. Instead of relying on a single train-test split, k-fold cross-validation was applied (k = 6) to repeatedly train and evaluate models on different partitions of the dataset. This leads to a more robust estimate of the model's performance.

k-fold cross-validation is a robust resampling technique used to assess the performance of machine learning models while mitigating overfitting. The dataset is divided into k equally sized subsets (folds), where the model is trained on k-1 folds and tested on the remaining fold. This process is repeated k times, ensuring that each fold serves as the test set once. The final performance metric is obtained by averaging the results across all iterations, providing a more reliable estimate of the model's generalizability.

Labeling of the molars was performed by LabelImg. It was performed by manually selecting the permanent molars on the right and left in both jaws. A ground truth consisting of localization and class information was created. Before labeling, a calibration session was conducted on 20 panoramic radiographs that were not included in

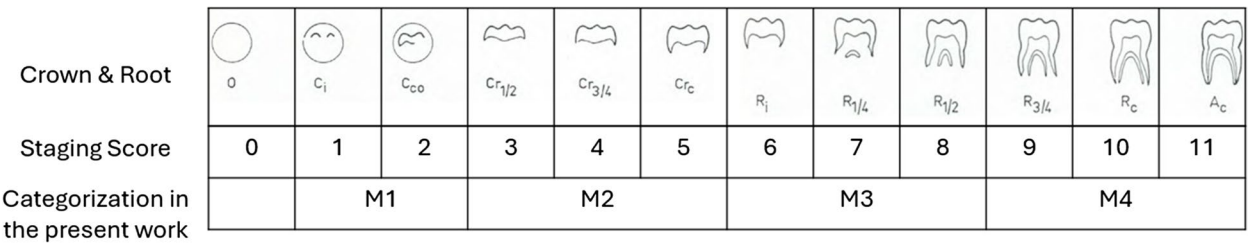

Fig. 1 Stages of tooth formation for assessing the development of molar teeth according to Haavikko et al. [20] and the categorization used in the present work

the study. All evaluations and labeling were performed twice by an oral and maxillofacial radiologist (10 years of experience, B.Ç.). Images that could not be matched in the evaluation were not included in the study.

Google Colab was used for training the models. The data set was saved in Google Drive and accessed via Google Colab. The trained models were then tested with the test data and then performance metrics were calculated in the end. Additionally, the best model performance was also tested by an external data that was publicly available [21]. Testing with an external dataset in a deep learning-based application provides a crucial assessment of the model's generalization ability, ensuring it performs well on unseen data from various centers. It enhances robustness by evaluating the model's ability to handle variations in imaging conditions, noise levels, and demographic differences. Additionally, external validation improves real-world applicability by simulating deployment conditions, particularly when data originates from different institutions or sources. This process also helps detect potential biases that may have arisen from training on a limited or homogeneous dataset. Furthermore, it enables objective performance benchmarking by allowing comparisons with other models using standardized datasets. In regulated fields such as healthcare, external validation is often a necessary step for credibility and approval. Finally, testing on an external dataset helps identify weaknesses in the model, guiding further refinements to improve accuracy and reliability.

Brief summary on CNNs

Cascade R-CNN with ResNet101

Cascade R-CNN is an extension of the popular Faster R-CNN framework for object detection. Its main goal is to enhance detection accuracy by employing a cascaded structure comprising multiple detector stages. The fundamental concept behind Cascade R-CNN involves iteratively refining bounding box predictions and minimizing false positives at each stage. This is accomplished by incorporating a sequence of classifier cascades, with each cascade stage being progressively more discerning and precise than its predecessor [22]. In this study we employ a deep neural network called ResNet101 as Cascade R-CNN's backbone architecture. ResNet101 (Residual Network) is a deep convolutional neural network architecture that has 101 layers.

YOLOv3 with DarkNet53

YOLO (You Only Look Once), is a highly regarded algorithm for real-time object detection. Renowned for its exceptional speed and accuracy, YOLOv3 represents an advancement over the original YOLO algorithm, delivering substantial enhancements in detection performance. YOLOv3 follows the single-shot detection approach, meaning it performs object detection directly on the entire image in one pass, rather than using a two-stage region proposal process. This makes it faster compared to other object detection methods [23]. In this study we employ a deep neural network called Darknet-53 as YOLOv3's backbone architecture. Darknet-53 is a variant of the Darknet architecture and consists of 53 convolutional layers, enabling it to extract rich and high-level features from input images.

Hybrid Task Cascade (HTC) with ResNeXt101

Hybrid Task Cascade (HTC) is an advanced framework for object detection that extends the Cascade R-CNN architecture. It strives for outstanding performance by tackling issues like precise localization, handling objects at different scales, and minimizing false positives. HTC has proven its success by achieving rank 1st in the COCO 2018 Challenge at object detection task [24]. In this study we employ a deep neural network called ResNeXt101 as HTC's backbone architecture. ResNeXt101 is an extension of the ResNet architecture that incorporates cardinality, allowing for parallel pathways within each block.

DetectoRS with ResNet50

DetectoRS (Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution) is an advanced object detection algorithm that aims to improve the accuracy and robustness of object detectors. It addresses the challenges posed by scale variation, occlusion, and object layout diversity. DetectoRS has demonstrated state-of-the-art performance on various benchmark datasets for object detection tasks. By incorporating the Recursive Feature Pyramid, Switchable Atrous Convolution, and object context aggregation, DetectoRS enhances the accuracy, adaptability, and robustness of object detectors [25]. In this study we employ a deep neural network called ResNet50 as DetectoRS's backbone architecture. ResNet50 (Residual Network) is a deep convolutional neural network architecture that has 50 layers.

SSD with VGG16

SSD (Single Shot MultiBox Detector) is a popular object detection algorithm known for its simplicity and efficiency. It provides real-time object detection capabilities by performing object localization and classification in a single forward pass of a deep neural network. SSD offers a good balance between speed and accuracy in real-time object detection tasks. In addition, SSD achieved higher mAP value in VOC2007 test images compared to Faster R-CNN [26]. In this study we employ a deep neural network called VGG16 as SSD's backbone architecture.

VGG16 is a convolutional neural network (CNN) architecture that was introduced by the Visual Geometry Group (VGG) at the University of Oxford that has 16 layers.

EfficientNet

EfficientNet is a family of convolutional neural network models that have gained attention for their remarkable performance and efficiency. These models have achieved state-of-the-art results on various computer vision tasks while maintaining a high level of computational efficiency. EfficientNet models have demonstrated superior accuracy on tasks such as image classification, object detection, and semantic segmentation. They have achieved state-of-the-art results on benchmark datasets like ImageNet, while maintaining computational efficiency, making them highly valuable in both research and practical applications [27].

NAS-FPN with ResNet50

This model is developed by Ghiasi et al. NAS-FPN, which stands for Neural Architecture Search Feature Pyramid Network, is an advanced architecture designed for object detection tasks. It combines two key components: Neural Architecture Search (NAS) and Feature Pyramid Network (FPN). Neural Architecture Search is a technique that automates the design process of neural network architectures. Instead of manually designing architectures, NAS-FPN leverages a search algorithm to explore and discover optimal network architectures specifically tailored for feature pyramid networks.

Feature Pyramid Network (FPN) is a widely used architecture for multi-scale feature extraction in object detection. It enhances the detection performance by fusing features from different scales, enabling the network to detect objects of varying sizes and maintain spatial information [28]. In this study we employ a deep neural network called ResNet50 as NAS-FPN's backbone architecture. ResNet50 (Residual Network) is a deep convolutional neural network architecture that has 50 layers.

Deformable DETR with ResNet50

Deformable DETR is an advanced object detection algorithm that builds upon the DETR (Detection Transformer) framework. It introduces deformable attention mechanisms to enhance the model's ability to handle objects with complex shapes and appearances. Deformable DETR has shown promising results in various computer vision tasks, including object detection and panoptic segmentation. It improves the model's ability to handle objects with complex shapes, occlusions, and variations in scale, resulting in more accurate and robust detections [29]. In this study we employ a deep neural network called ResNet50 as Deformable DETR's backbone architecture. ResNet50 (Residual Network) is a deep convolutional neural network architecture that has 50 layers.

Probabilistic Anchor Assignment (PAA) with ResNet101

Probabilistic Anchor Assignment (PAA) is a technique used in object detection algorithms that rely on anchor-based methods. Anchors are predefined bounding boxes of different sizes and shapes that act as reference points for detecting objects in an image. In traditional anchor-based methods, each anchor box is assigned a positive or negative label based on a fixed overlap threshold with the ground-truth object. PAA introduces a probabilistic approach to anchor assignment. Instead of binary labels, it assigns a probability score to each anchor, indicating the likelihood of it being associated with an object. This assignment considers the degree of overlap between the anchor box and the ground-truth object [30]. In this study we employ a deep neural network called ResNet101 as PAA's backbone architecture. ResNet101 (Residual Network) is a deep convolutional neural network architecture that has 101 layers.

Transfer learning

Transfer Learning is a technique in which a previously trained model is adapted to solve a new problem with a new data set. For this task, a saved file of the previously trained version of the model is used. The model is then retrained with new data and adapted to perform a new task. In this way, instead of training the model from scratch, we reuse features such as weight learnt from an existing model and fine-tune it for a new task. This reduces the training time of the model. In addition, high performance rates are achieved despite being trained with little data. All 9 models used in this study were pre-trained with the COCO 2017 dataset.

Preprocessing

Normalization

The "Normalize" operation performs pixel-wise normalization on the image, which is a crucial preprocessing step in deep learning models. This method adjusts input images pixels values to make training more stable and faster. This process was applied in all models used in this study.

Padding

The "Pad" is a method that adds extra pixels to the image so that all images used to train the model can be divided by a fixed value. In this study, the fixed value is 32. By default the value of the added pixels are "0" (black). For example, in order to divide both dimensions of a

1900 × 1050 pixel image by 32, the "Pad" method adds 20 pixels to the 1900 pixel axis and 30 pixels to the 1050 axis, making both axes divisible by 32. In order for the convolution layers of models such as YOLO and R-CNN to work correctly, the images used for training must be divisible by a fixed number, usually 16 or 32. Therefore, "Pad" is an important preprocessing operation. Except for the "SSD" model, this process is performed in all other models used.

Loading annotations

The"LoadAnnotations" is the most important preprocessing stage that loads ground truth labels into the trained model. In this study, data such as bounding box (bbox) and class labels were loaded into the model with this process. This process was also applied to all models used.

Data augmentation

Resizing

"Resize" data augmentation technique was used in all models tested in this study. Although the "img_scale" parameter varies from model to model, "keep_ratio" was selected as "True" in all models. In this way, all images were resized while preserving the aspect ratios of the original image. In general, the use of the "Resize" augmentation technique prevents the models from being trained with only one dimension of data and prevents "over fitting". In this way, it is ensured that the models can detect images of different sizes.

Random flipping

"RandomFlip" is one of the most used data augmentation method that rotates the images and bounding boxes in the training set on the horizontal axis at a specified rate. In this study, the rotation rate is set to 0.5 for all models used. With the "RandomFlip" method, the models were trained at different angles to test the models in realistic scenarios. In addition, overfitting was prevented by changing the orientations of images and objects.

Random cropping

RandomCrop is a data augmentation technique that randomly selects and crops a portion of the image, ensuring that labelled objects remain visible. Like RandomFlip, it prevents overfitting by preventing the model from memorising the location of objects. It also focuses the model on different parts of the image, making it easier to detect small objects. This data augmentation method was used in Deformable DETR, EfficientNet, NAS-FPN, SSD and YOLOv3 models.

Photo metric distortion

PhotoMetricDistortion is a data augmentation technique that randomly changes the brightness, contrast, saturation and hue of an image to simulate different lighting conditions. This method increases the robustness of the model against different lighting conditions by randomly changing the brightness and contrast of the data. It also prevents overfitting by introducing colour variations. In this study, the method was used only in SSD and YOLOv3 models.

Evaluation metrics

When using deep learning models, certain performance metrics are used to show that these models have been successfully trained and tested. In this study, accuracy, precision, recall and F1-score metrics are used to compare the performance of the models used.

Accuracy refers to the measurement of how well a model correctly predicts or classifies instances from a given dataset. Accuracy is calculated by dividing the number of correctly classified instances (true positives and true negatives) by the total number of instances in the dataset. Precision is a performance metric that measures the accuracy of positive predictions made by a model. It focuses on the proportion of correctly predicted positive instances out of all the instances predicted as positive. It calculates the ratio of true positive predictions to the total number of positive predictions made by the model. Recall, also known as sensitivity or true positive rate is a performance metric that measures the ability of a model to correctly identify positive instances from the total number of actual positive instances in a dataset. Recall calculates the proportion of true positive predictions made by the model out of all the actual positive instances in the dataset. The F1-score is a metric that combines precision and recall into a single measure. It provides a balanced assessment of a model's performance by considering both the ability to correctly identify positive instances (recall) and the accuracy of positive predictions (precision).

Object detection is a computer vision task that involves identifying and locating objects of interest within an image or a video. In the context of deep learning, object detection often involves using neural networks to predict bounding boxes around objects and assign class labels to those objects. Evaluation of object detection is commonly performed by Average Precision (AP) in computer vision research. It calculates the area under the precision-recall curve for different confidence thresholds. It indicates a deep learning model's ability to accurately localize objects interested. It is also related to Intersection Over Union (IOU) which measures the overlap of predicted

Table 1 Equations for evaluation metrics used

Average Precision
$AP_{threshold} = \int_0^1 p(x)dx$
Mean Average Precision for n-classes
$mAP_{threshold} = \frac{1}{n}\sum_{i=1}^{n} AP_i$
Accuracy
$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
Precision
$Precision = \frac{TP}{TP+FP}$
Recall (Sensitivity)
$Recall = \frac{TP}{TP+FN}$
F1-score
$F1Score = 2 * \frac{Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$
Intersection over Union (IoU)
$IoU = \frac{area(groundtruth \cap predicted)}{area(groundtruth \cup predicted)}$

Table 2 Model results

Models	Accuracy	mAP	Recall	F1-score
Cascade R-CNN	0.79	0.83	0.85	0.84
YOLOv3	0.72	0.82	0.78	0.80
HTC	0.79	0.84	0.85	0.84
DetectoRS	0.76	0.80	0.83	0.81
SSD	0.58	0.66	0.65	0.65
EfficientNet	0.69	0.68	0.79	0.73
NAS-FPN	0.72	0.75	0.79	0.76
Deformable DETR	0.81	0.86	0.85	0.86
PAA	0.68	0.68	0.79	0.73

bounding box to the ground truth. Mean Average Precision (mAP) is the average AP across multiple classes. Table 1 summarizes each formula for the evaluation metrics applied in this work.

Precision-Recall curves providing the area under curve (AUC) includes legends that are Common Objects in Context (COCO) metrics. These are C75, C50, Loc, Sim, Oth, and FN. While C75 and C50 stand for AUC for Intersection over Union (IoU) of 0.75 and 0.5 respectively, Loc refers to AUC with ignoring localization errors. Sim, Oth and FN indicate AUC while removing super-category class confusions, class confusions and all remaining errors respectively.

Results

Deep learning models were implemented for molar teeth germ detection. Table 2 presented results on average, i.e. metrics were presented as an average of all 4 classes for each fold considering k-fold cross validation (k = 6). The evaluation of various object detection models based on accuracy, mean Average Precision (mAP), recall, and F1-score reveals that Deformable DETR outperforms all other models, achieving the highest accuracy (0.81), mAP (0.86), and F1-score (0.86), indicating superior detection and classification capabilities. HTC and Cascade R-CNN follow closely, both attaining an accuracy of 0.79, with mAP values of 0.84 and 0.83, respectively, demonstrating robust performance across different detection tasks. DetectoRS and YOLOv3 exhibit moderate performance, with YOLOv3 achieving a competitive mAP (0.82) but lower recall (0.78), making it a viable choice for real-time applications. NAS-FPN presents a balanced performance, though with a lower mAP (0.75). In contrast, PAA and EfficientNet yield lower accuracy (0.68 and 0.69, respectively) and mAP (both 0.68), though their recall values suggest reasonable object detection capabilities. Finally, SSD performs the weakest, with the lowest accuracy (0.58) and F1-score (0.65), indicating limited suitability for high-precision detection tasks. These findings suggest that Deformable DETR is the most effective model for high-accuracy object detection, while YOLOv3 remains a suitable choice for applications requiring real-time inference.

Table 3 demonstrated class-wise evaluation metrics for model, still average of six-folds. The performance of object detection models varied across the four classes, highlighting differences in detection effectiveness for each category. For class M1, Deformable DETR achieved the highest F1-score (0.86), followed closely by HTC (0.85) and Cascade R-CNN (0.84), indicating strong detection capabilities in this category. In contrast, SSD and PAA exhibited the lowest F1-scores (0.53 and 0.65, respectively), suggesting weaker performance in M1. For class M2, Cascade R-CNN, YOLOv3, and Deformable DETR demonstrated high precision (0.86–0.88), though recall varied, with Deformable DETR achieving the highest F1-score (0.79). HTC and Cascade R-CNN followed closely, both attaining an F1-score of 0.77. SSD again underperformed, with the lowest F1-score (0.65), indicating difficulties in detecting objects of class M2. For class M3, Cascade R-CNN, HTC, and Deformable DETR achieved the highest precision (0.83–0.85), though their recall values varied. Deformable DETR led with the highest F1-score (0.80), while Cascade R-CNN and HTC followed at 0.78 and 0.77, respectively. SSD had the lowest F1-score (0.58), highlighting its difficulty in detecting objects from M3. For class M4, most models performed significantly better, with Deformable DETR, Cascade R-CNN, and HTC attaining the highest F1-scores (0.92–0.94). Cascade R-CNN and HTC both exhibited strong recall (0.99), indicating high sensitivity in detecting M4 objects. SSD and EfficientNet had the lowest F1-scores (0.82), showing comparatively weaker performance in

Table 3 Models class-wise results

Models	Precision				Recall				F1-Score			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
Cascade R-CNN	0.72	0.86	0.83	0.89	1.00	0.69	0.74	0.99	0.84	0.77	0.78	0.94
YOLOv3	0.71	0.86	0.76	0.89	0.83	0.64	0.72	0.90	0.77	0.73	0.74	0.89
HTC	0.76	0.79	0.83	0.90	0.97	0.70	0.72	0.99	0.85	0.77	0.77	0.94
DetectoRS	0.71	0.84	0.80	0.87	0.96	0.65	0.73	0.98	0.82	0.73	0.76	0.92
SSD	0.63	0.68	0.61	0.73	0.45	0.63	0.57	0.93	0.53	0.65	0.58	0.82
EfficientNet	0.56	0.73	0.69	0.75	0.89	0.66	0.65	0.90	0.70	0.69	0.67	0.82
NAS-FPN	0.60	0.74	0.76	0.90	0.94	0.69	0.60	0.90	0.73	0.71	0.67	0.90
Deformable DETR	0.78	0.88	0.85	0.94	0.97	0.72	0.76	0.96	0.86	0.79	0.80	0.92
PAA	0.51	0.75	0.72	0.75	0.90	0.65	0.69	0.93	0.65	0.69	0.70	0.83

this class. Overall, Deformable DETR, Cascade R-CNN, and HTC demonstrated the most consistent and reliable performance across all four classes.

Next, the best performing model from the previous step was tested by an external public dataset to provide validity in real-world conditions using data originating from different sources. 30 panoramic radiographs were used for testing. Table 4 demonstrates that Deformable DETR achieves an accuracy of 0.69, indicating the proportion of correctly classified instances. The mean Average Precision (mAP) is 0.72, reflecting the model's precision-recall trade-off across different thresholds. The recall value of 0.82 suggests that the model effectively identifies relevant instances, while the F1-score of 0.77 represents a balanced measure of precision and recall, showing reliable overall performance. The second part provides a more detailed breakdown of Deformable DETR's performance across four classes based on Precision, Recall, and

Table 4 Comparisons between the studies in the literature on germ detection and this study

Author	Task	Type of Image	Model	Number of Classes	Data size	Metrics
Çalışkan et. al	Object Detection	Panoramic	Faster R-CNN	1	74	Accuracy: 0.8372, Sensitivity: 0.4545 Specificity: 0.9688 Precision: 0.8333
De Tobel et. al	Classification	Panoramic	AlexNet	10	400	Mean accuracy: 0.51, Mean absolute difference: 0.6, Mean linearly weighted kappa: 0.82
Merdietio et. al	Classification	Panoramic	DenseNet201	10	400	Accuracy: 0.61 mean absolute difference: 0.53 linear Cohen's kappa coefficient: 0.84
Banar et. al	Segmentation	Panoramic	U-Net like CNN model	10	400	Dice score: 93% Accuracy: 54% mean absolute error: 0.69 linear Cohen's kappa coefficient: 0.79
Kaya et. al	Object Detection	Panoramic	YOLOv4	1	4518	Average Precision: 94.16% Precision: 0.89 Recall: 0.91 F1-score: 0.90
This work	Object Detection	Panoramic	Cascade R-CNN YOLOv3 HTC DetectoRS SSD EfficientNet NAS-FPN Deformable DETR PAA	4	210	Avg. Accuracy: 0.81 Average Precision: 0.86 Average Recall: 0.85 Average F1-score: 0.86

Table 5 Testing with external public dataset

	Accuracy				**mAP**				**Recall**				**F1-score**			
Deformable DETR	0,69				0,72				0,82				0,77			
Models	Precision				Recall				F1-Score							
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4				
Deformable DETR	0,52	0,84	0,93	0,6	0,94	0,77	0,56	1,00	0,67	0,80	0,70	0,75				

F1-score. Precision values range from 0.52 (M1) to 0.93 (M3), showing that M3 has the highest precision. Recall scores vary from 0.56 (M3) to 1.00 (M4), meaning M4 captures all relevant instances. The F1-score, which balances precision and recall, fluctuates between 0.67 (M1) and 0.80 (M2), indicating that M2 performs best in maintaining both precision and recall. Table 5.

Detection predictions performed by the best and the worst performing models were demonstrated in Figs. 2, 3 and 4. Three different image samples were randomly chosen. In each figure, it includes the ground truth of the image, prediction result from the best performing model and the worst performing model, SSD, were sequentially positioned with figure labels A, B and C respectively.

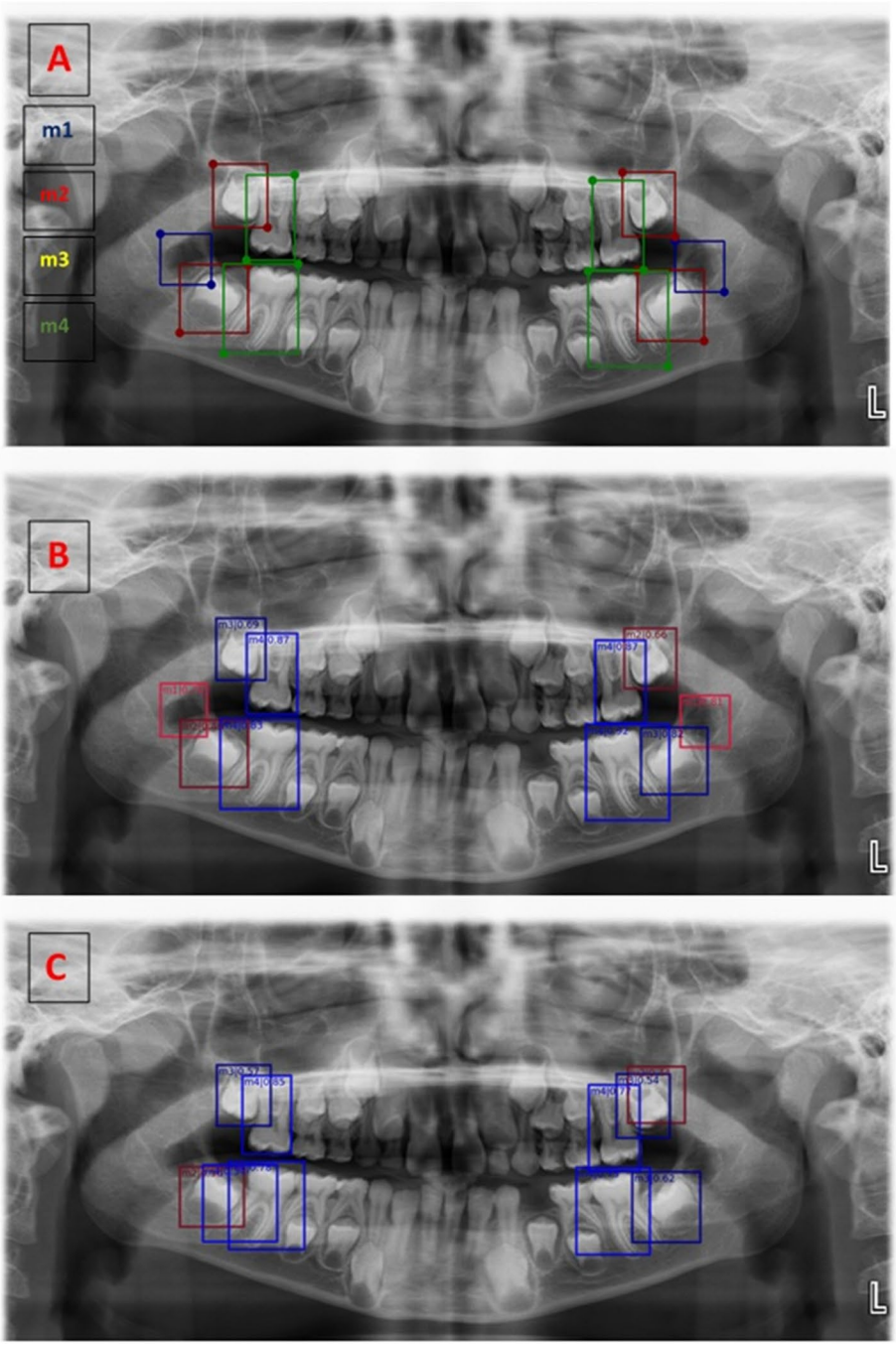

Fig. 2 For the selected first image, from up to bottom; A. Ground truth, B. predicted result from Deformable DETR with ResNet50 and C. predicted result from SSD

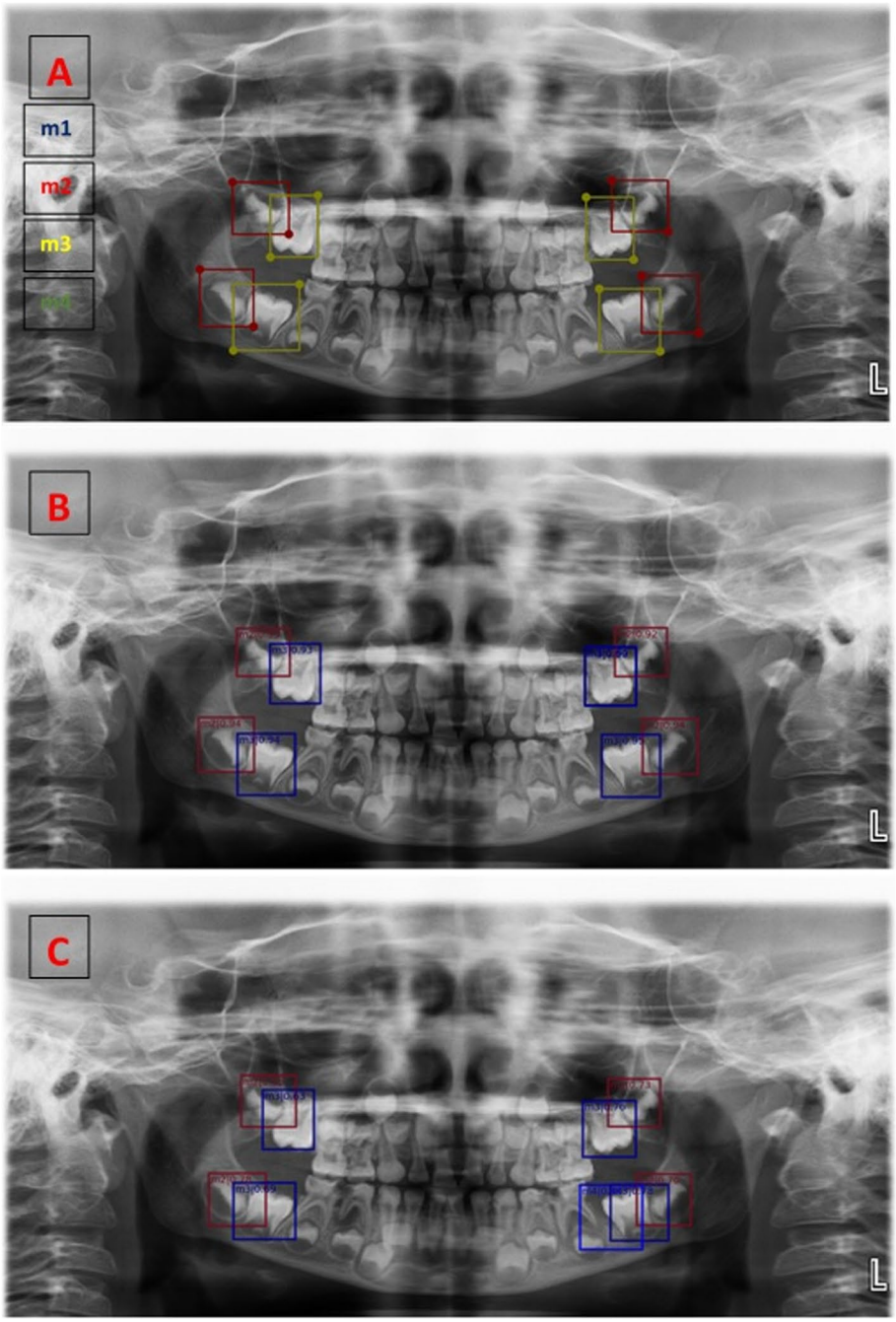

Fig. 3 For the selected second image, from up to bottom; A. Ground truth, B. predicted result from Deformable DETR with ResNet50 and C. predicted result from SSD

Figure 5 shows the loss curve obtained from the Deformable DETR model at 100 epochs. The loss curve indicates that the training and validation loss values decrease regularly as the number of epochs increases, resulting in successful training with sufficient number of epochs.

Figure 6 presents precision-recall curve. In this graph, the AUC (Area Under Curve) metric is 0.918. There are no errors related to super category false positives and class confusions. When background confusions are removed, the AUC will be 1. In general, the errors were due to location and background errors.

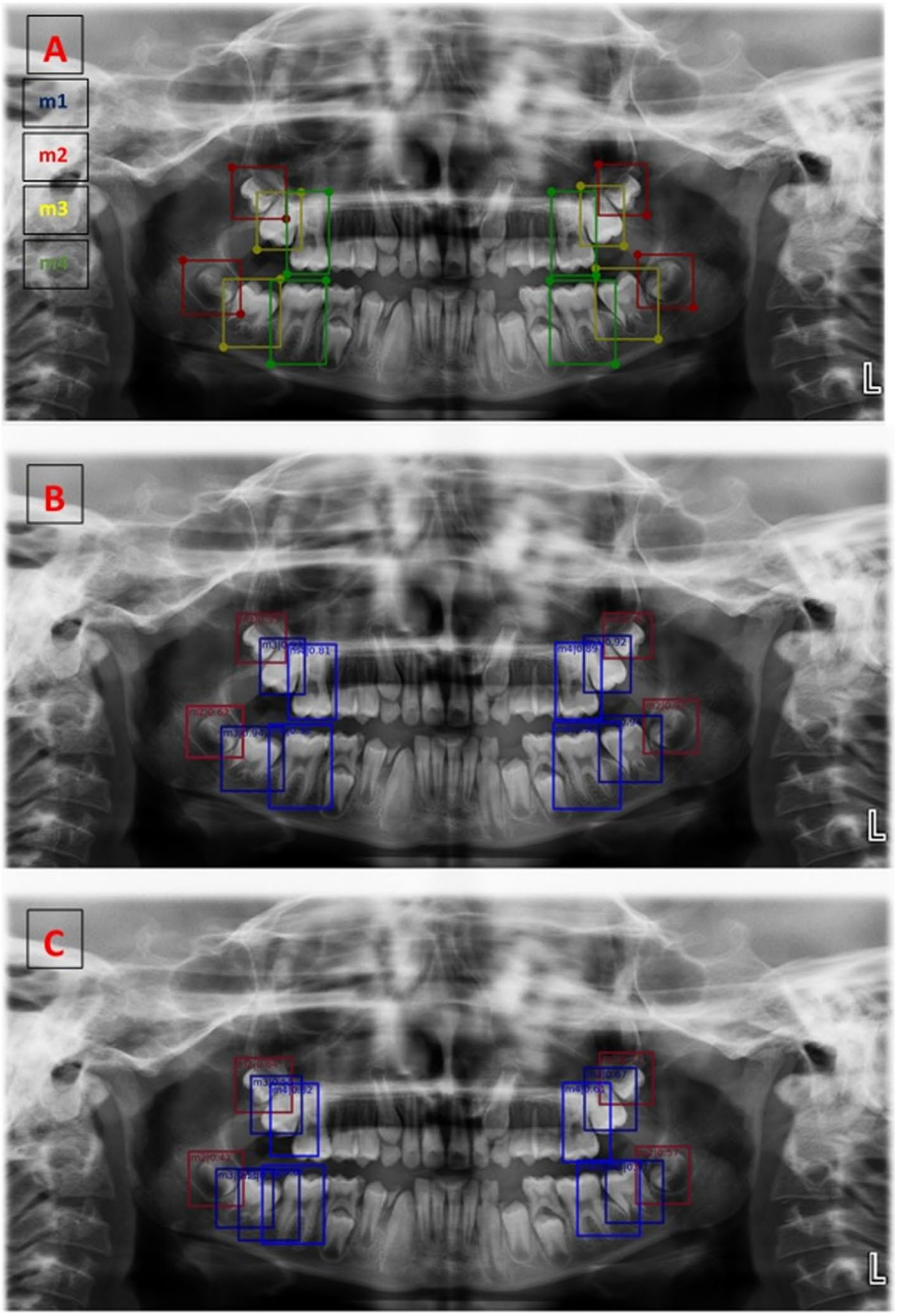

Fig. 4 For the selected third image, from up to bottom; A. Ground truth, B. predicted result from Deformable DETR with ResNet50 and C. predicted result from SSD

Discussion

Staging the development of molar teeth, especially the third molar, is an age estimation process for infants, adolescents, and adults. Manual staging is a process in which a qualified one examines images, categorization of the development for specific staging, resulting in an approximation of age. As a significant limitation, it is prone to variability within and between experts in staging, which was further reported by De Tobel et al. for third molar [8]. Therefore, to handle this issue, deep learning is a promising tool instead of manual labor. Detection of molar teeth and their staging classification is an essential step towards automated systems for age estimation and charting. Deep learning has been applied to a wide

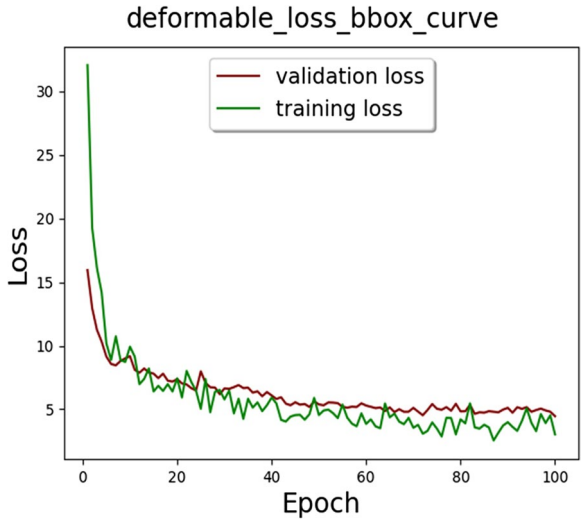

Fig. 5 Loss curve for 100 epochs from Deformable DETR with ResNet50

range of dental problems in recent years [31–34]. There is a limited number of previous studies in the literature on development staging of molar teeth for age estimation.

Çalışkan et. al. used deep learning algorithms to detect submerged primary molars. In their study, they detected submerged teeth in 74 panoramic radiography images using the Faster R-CNN deep learning model. They compared their results with the findings made by 2 dentists. As a result, they obtained accuracy of 0.8372, sensitivity of 0.4545, specificity of 0.9688 and precision of 0.8333 metrics [17].

Tobel et al. classified the lower third molar developmental stages in order to make age estimation [8]. They classified the developmental stages of lower third molars in 400 panoramic radiography images using Adobe Photoshop and MATLAB programs. They used AlexNET deep learning model to classify these images. As a result, Mean accuracy of 0.51, Mean absolute difference of 0.6 and Mean linearly weighted kappa of 0.82 metrics were obtained.

Merdietio et al. performed the classification of the developmental stages of the third molars using the deep learning method [19]. In this study, images of the third molar teeth were obtained manually from 400 panoramic X-ray radiography images. These images are divided into three as bounding box (BB), rough segmentation (RS) and full segmentation (FS). DenseNet201 CNN model was used to classify these images. With this model, accuracy of 0.61, mean absolute difference of 0.53 and linear Cohen's kappa coefficient of 0.84 results were obtained, respectively.

Banar et al. used deep learning to detect the developmental stages of the third molar in their study [6]. The dataset they used consists of 400 panoramic X-ray radiographs. In this study, the development of third molars is divided into 10 stages. A 3-step CNN-based, U-Net like model was used for the detection, segmentation and classification of these stages. In the previous works, detection and segmentation were done manually and only classification was done automatically [19]. In this study, all stages were carried out automatically with the CNN model. With this model, they obtained dice score of 93%,

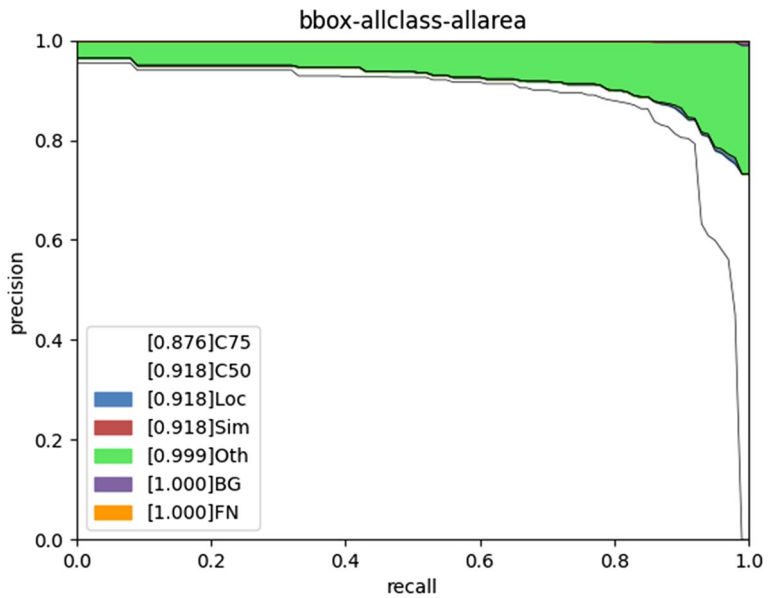

Fig. 6 Precision-Recall curve from Deformable DETR with ResNet50

accuracy of 54%, mean absolute error of 0.69 and linear Cohen's kappa coefficient of 0.79, respectively.

Kaya et al. used a deep learning model to detect tooth germs in panoramic X-ray radiography dataset [18]. In their study, CNN-based YOLOv4 model with CSPDarknet53 backbone is used for tooth germ detection. A single class tooth germ was chosen for all tooth types. It was resulted that the average precision, precision, recall, and F1-score were 94.16%, 0.89, 0.91 and 0.90 respectively for the tooth germ class.

In this study, the developmental stages of molars were divided into 4 classes and these stages were automatically determined by nine deep learning methods. These models are Cascade R-CNN, YOLOv3, Hybrid Task Cascade(HTC), DetectorRS, SSD, EfficientNet, NAS-FPN, Deformable DETR and Probabilistic Anchor Assignment(PAA), respectively. The dataset used consists of 210 panoramic radiography images. Among the models used, the Deformable DETR CNN model gave the best results with the values of 0.82 total accuracy, 0.86 average precision, 0.86 average recall and 0.86 average F1-score, respectively. The two main features that distinguish the current study from other studies are: (i) staging was performed for all molars instead of just a single molar, (ii) the performance of a large number of deep learning models was analyzed in the broadest perspective. The results obtained from literature reviews and from this study are given in Table 4.

Although deep learning has achieved significant success across various fields, certain limitations restrict its broader applicability and reliability. A key challenge of this work is the dependence on extensive, multi-centered labeled datasets, as deep learning models often fails due to poor, insufficient and imbalanced data. Moreover, staging of tooth formation for assessing the development of molar teeth according to Haavikko et al. [20] was simplified in the present work by bringing them together in groups of three as shown in Fig. 1. Furthermore, there is a lack of standardized reporting practices, which makes benchmarking efforts difficult to improve future research.

Conclusion

In this study, the developmental stages of permanent molar teeth in panoramic radiography images were divided into 4 classes and the detection of these classes was carried out with nine CNN models. Among these models, the best and most consistent results were obtained from the Deformable DETR model. In the future, it is planned to create an auxiliary system for dentists that detects the developmental stages of all tooth germ with this deep learning model.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12903-025-05827-4.

Additional file 1.

Acknowledgements
None.

Authors' contributions
EFS processed, analyzed and visualized the data. BÇ conceptualized and designed the work, interpreted results, drafted and revised the work. MEÇ designed the work, analyzed results, drafted and revised the work.

Funding
None.

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
The study's procedures received approval from the Ethics Committee of the Gazi University, under the authorization number 2023—1216, in accordance with the Helsinki Declaration of the World Medical Association (2008 Version).

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 21 February 2025 Accepted: 17 March 2025
Published online: 01 April 2025

References

1. Kametani T. Maxillofacial cranial growth. Dental Journal of Iwate Medical University. 1979;4:165–74.
2. Mouradian WE. The face of a child: children's oral health and dental education. J Dent Educ. 2001;65(9):821–31.
3. Kuremoto K, et al. Estimation of dental age based on the developmental stages of permanent teeth in Japanese children and adolescents. Sci Rep. 2022;12(1):3345.
4. Yan J, et al. Assessment of dental age of children aged 3.5 to 16.9 years using Demirjian's method: a meta-analysis based on 26 studies. Plos One. 2013;8(12):e84672.
5. Schmeling A, et al. Age estimation. Forensic Sci Int. 2007;165(2–3):178–81.
6. Banar N, et al. Towards fully automated third molar development staging in panoramic radiographs. Int J Legal Med. 2020;134:1831–41.
7. Ching T, et al. Opportunities and obstacles for deep learning in biology and medicine. J R Soc Interface. 2018;15(141):1–47.
8. De Tobel J, et al. An automated technique to stage lower third molar development on panoramic radiographs for age estimation: a pilot study. J Forensic Odontostomatol. 2017;35(2):42.
9. Mohammad-Rahimi H, et al. Deep learning: a primer for dentists and dental researchers. J Dent. 2023;130:104430.
10. Mesquita GDTB, et al. Artificial intelligence for detecting cephalometric landmarks: a systematic review and meta-analysis. J Digit Imaging. 2023;36(3):1158–79.
11. Huang CX, et al. A review of deep learning in dentistry. Neurocomputing. 2023;554:1–13.
12. Hamd ZY et al. A closer look at the current knowledge and prospects of artificial intelligence integration in dentistry practice: a cross-sectional study. Heliyon. 2023;9(6):1–8.

13. Çelik B, ME Çelik. Root dilaceration using deep learning: a diagnostic approach. Appl Sci Basel. 2023;13(14):1-13.
14. Çelik B, Savaştaer EF, Kaya HI, Çelik ME. The role of deep learning for periapical lesion detection on panoramic radiographs. Dentomaxillofac Radiol. 2023;52(8):20230118.
15. Celik ME. Deep learning based detection tool for impacted mandibular third molar teeth. Diagnostics. 2022;12(4):942.
16. Çelik, B. and M.E. Çelik, *Automated detection of dental restorations using deep learning on panoramic radiographs.* Dentomaxillofacial Radiology, 2022. 51(8).
17. Caliskan S, et al. A pilot study of a deep learning approach to submerged primary tooth classification and detection. Int J Comput Dent. 2021;24(1):1-+.
18. Kaya E, et al. A deep learning approach to permanent tooth germ detection on pediatric panoramic radiographs. Imaging Sci Dent. 2022;52(3):275.
19. Merdietio Boedi R, et al. Effect of lower third molar segmentations on automated tooth development staging using a convolutional neural network. J Forensic Sci. 2020;65(2):481–6.
20. Haavikko K. The formation and the alveolar and clinical eruption of the permanent teeth. An orthopantomographic study. Suom Hammaslaak Toim. 1970;66(3):103–70.
21. Zhang Y, et al. Children's dental panoramic radiographs dataset for caries segmentation and dental disease detection. Scientific Data. 2023;10(1):380.
22. Cai Z, Vasconcelos N. Cascade R-CNN: high quality object detection and instance segmentation. IEEE Trans Pattern Anal Mach Intell. 2019;43(5):1483–98.
23. Redmon J, Farhadi A. Yolov3: An incremental improvement.. arXiv preprint arXiv:1804.02767, 2018.
24. Chen K et al. Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.
25. Qiao S, LC Chen, A Yuille. Detectors: detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021.
26. Liu W et al. Ssd: single shot multibox detector. In: Computer vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer. 2016.
27. Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. in International conference on machine learning. 2019. PMLR.
28. Ghiasi G, Lin TY, Le QV. Nas-fpn: Learning scalable feature pyramid architecture for object detection. arXiv preprint arXiv:1904.07392, 2019.
29. Zhu, X., et al., *Deformable detr: Deformable transformers for end-to-end object detection.* arXiv preprint arXiv:2010.04159, 2020.
30. Kim K, HS Lee. Probabilistic anchor assignment with iou prediction for object detection. In: Computer vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16. Springer. 2020.
31. Çelik ME, Mikaeili M, Çelik B. Improving resolution of panoramic radiographs: super-resolution concept. Dentomaxillofacial Radiology. 2024;53(4):240–7.
32. Broll A, Goldhacker M, Hahnel S, Rosentritt M. Generative deep learning approaches for the design of dental restorations: a narrative review. J Dent. 2024;145:104988.
33. Çelik B, Genç MZ, Çelik ME. Evaluation of root canal filling length on periapical radiograph using artificial intelligence. Oral Radiol. 2024. 1–9.
34. Ong SH, Kim H, Song JS, Shin TJ, Hyun HK, Jang KT, Kim YJ. Fully automated deep learning approach to dental development assessment in panoramic radiographs. BMC Oral Health. 2024;24(1):426.

Publisher's Note