# Detection of Dental Abnormalities in Panoramic Radiographs via Convolutional Neural Networks

Group Members: Alexandre Delgado 58964, Tiago Garcia 54938, Mariia Korepanova 64689

## A. Quick rubric (✓ / △ / ✗)

| Criterion | Assessment | Notes |
|---|---|---|
| **Technical soundness** | ✓ | Object-detection framing (YOLOv5) suits anomaly localisation; DENTEX annotations are in Pascal-VOC and COCO formats. |
| **Feasible on free-tier GPUs** | ✓ | 1000 images (≈ 1 GB) × imgsz 640; YOLOv5-s fine-tunes in ≈ 2 h on a Colab T4 with AMP and batch = 8. |
| **Dataset readiness** | △ | DENTEX classes are *quadrant* × *anomaly* (21 labels) and bounding boxes are often small; class imbalance (caries ≫ impacted) must be handled. |
| **Starting-code / transfer-learning plan** | ✓ | Ultralytics YOLOv5 notebook → change `data.yaml`; DiffusionDet weights can be a later comparison. |
| **Evaluation metrics** | ✓ | mAP@[0.5:0.95] per class, plus class-wise precision/recall; add mean IoU for tooth-level segmentation if you later train a mask head. |

(✓ = ready; △ = needs work; ✗ = high risk)

---

## B. Targeted suggestions

1. **Normalise image resolution once**
   *Rescale longest edge to 1024 px, letter-box to 4:3 ratio.*
   This keeps small lesions (> 32 px) detectable while fitting GPU memory.
2. **Class-imbalance mitigation**
   - Use **image-level sampling**: oversample "impacted" and "periapical" radiographs 2 ×.
   - Enable **loss = "cri+focal"** (`--fl_gamma 1.5`) in Ultralytics to emphasise rare boxes.
3. **Transfer-learning recipe (YOLOv5-s)**

| Phase | Layers | Epochs | LR | Time |
|---|---|---|---|---|
| 1 – warm-up | freeze backbone | 20 | 1e-3 | 30 min |
| 2 – fine | unfreeze all | 80 | 3e-4 cosine | 90 min |

Mixed-precision (`--amp`) and `imgsz 640` keep VRAM ≈ 6 GB.

4. **Tooth-number awareness**
   *Option A – two-stage:* detect teeth first (YOLOv8-seg pretrained on Dentures) → crop → anomaly detector.
   *Option B – add **tooth-index** as an extra label attribute* and train a single model; easier but lower recall on tiny lesions. Start with option A only if individual-tooth localisation is critical.

5. **Qualitative outputs**
    - Save **overlay PNGs** and **Grad-CAM heatmaps** for 20 validation images. Dentists can sanity-check false positives quickly.
    - Compute per-image **average false-positive count**; keep ≤ 1 to maintain clinical usability.
6. **Compute budget**

| Task | GPU h |
| --- | --- |
| Data conversion + sanity plots | 0.2 |
| YOLOv5-s full schedule | 2.0 |
| Hyper-param sweep (imgsz 512 vs 640) | 0.8 |
| Inference + CAM | 0.3 |
| **Total** | **≈ 3.3 h** |

Safe under free Colab quota.

---

### Immediate Milestone-1 checklist

1. **Create `data.yaml`** listing train/val splits (patient-wise) and the 4 anomaly classes.
2. Run phase-1 warm-up (20 epochs) to verify GPU fit; log mAP@0.5 and confusion matrix.
3. Push `prepare_data.py`, `train_yolo.sh`, and sample prediction images to Git.

With this streamlined plan you will have a functioning detector in week 1, leaving time to explore DiffusionDet or tooth-instance segmentation later.