**Recommended hyperparameters & training tips**

* num\_epochs: 20–50 depending on dataset size. Start small to iterate.
* batch\_size: 2–8 (depending on GPU). Lower if out-of-memory.
* lr: 0.005 for SGD (with batch size 2–4). If you increase batch size, scale lr.
* weight\_decay: 1e-4 to 5e-4.
* lr\_scheduler: StepLR or ReduceLROnPlateau.
* **Freeze backbone** first 3–5 epochs (for name, p in model.backbone.named\_parameters(): p.requires\_grad = False) then unfreeze.
* **Augmentations**: horizontal flips, small rotations (-10 to 10 deg), color jitter, random crop (careful with keypoints).
* **Loss monitoring**: model returns a dict of losses (classification loss, box regression, keypoint loss). Watch keypoint loss to ensure it decreases.
* **Class imbalance**: if many negatives, consider sampling or loss weighting.
* **Keypoint annotation quality**: keypoint accuracy depends strongly on annotation precision.

**9 — Diagnosing common issues**

* **Model complaining about keypoints shape**: Ensure keypoints is (N, K, 3) and dtype float32.
* **No detections**: lower score\_thresh to debug, check training loss, ensure correct label ids (foreground=1).
* **Incorrect scaling**: ensure transforms for images and targets are consistent (especially resize).
* **Keypoints off**: check whether augmentation flips or affine transforms are applied correctly to keypoints.

**10 — Production & deployment notes**

* Export to TorchScript to serve (but keypoint heads sometimes require extra handling).
* Use mixed precision during inference/training for speed.
* Quantize or prune for embedded systems.
* For a robotics plucker, ensure latency < threshold; you may prefer a lighter backbone (MobileNet based detectors with a lightweight keypoint head) if the real-time requirement is tight.

**11 — Checklist before training**

* Confirm annotation coordinate system — pixel coordinates with origin at top-left.
* Ensure one keypoint per box; if multiple, pad with (0,0,0) if absent.
* Convert all annotations to the same JSON and verify shapes.
* Install required libs: torch, torchvision (matching versions), pycocotools (optional for COCO eval), Pillow, numpy.
* Example installs: pip install torch torchvision pycocotools pillow numpy