7) Alignment & Reproducibility Report（對齊與可重現性報告）

Date: 2025-12-24

Objective

Ensure that training, distillation, and evaluation are reproducible and that data/label alignment is consistent end-to-end.

Alignment artifacts (data)

- Unified manifest validation:

  - `outputs/manifest_validation_all_with_expw.json` (466,284 rows, 0 missing paths, 0 bad labels)

  - `outputs/manifest_validation.json` (374,491 rows, 0 missing paths)

- Cleaning provenance:

  - `Training_data_cleaned/clean_report.json`

Alignment artifacts (distillation)

- Softlabel export artifacts are explicit and indexable:

  - `outputs/softlabels/.../softlabels.npz`

  - `outputs/softlabels/.../softlabels_index.jsonl`

- Teacher vs student decoupling:

  - Student training consumes saved teacher logits (no need to run teacher ONNX at student training time).

Checkpoint / run provenance

- Teacher + student runs save reproducible artifacts:

  - `best.pt` checkpoints

  - `history.json` curves

  - `reliabilitymetrics.json` and `calibration.json` evaluations

- Provenance helpers:

- `scripts/inspect_checkpoint_provenance.py`

  - `_ckpt_provenance_*.txt` snapshots

Windows reproducibility notes

- Windows multiprocessing can be fragile (DataLoader, shared mappings). The project uses safer defaults and PowerShell orchestration for stable runs.

DKD resume incident (and fix)

- Root cause: DKD resumed from KD at a `start_epoch` greater than the configured total `--epochs`, resulting in a no-op training loop and an empty output.

- Fix: compute DKD total epochs as (resume_epoch + 1 + extra_dkd_epochs) inside the runner.

- This was documented in the weekly log and validated by a successful DKD-only rerun producing full artifacts.

Next steps

- Add a single "run manifest" per experiment (args + git commit hash + environment packages) to make reproducing any run one command.