

VectorPredictor — Onchocerciasis (*Simulium*) data requirements

Supplement to the main VectorPredictor data SOP (malaria *Anopheles* workflows). Use this document when **Specimen scanned** is set to *Simulium* blackflies.

1. Scope

This profile screens desiccated *Simulium* blackfly spectra for signals consistent with *Onchocerca* infection, using the production Keras **cm24_tf** raw-absorbance CNN in-process (TensorFlow; no Julia on the server).

It does **not** use the malaria DeepAge model or the 1625-point **.dat** mosquito pipeline.

2. Specimen scanned

Before upload or batch prediction:

1. Open **Metadata**, **Upload spectra**, or **Batch predict**.
2. Under **Specimen scanned**, choose *Simulium* blackflies — **Onchocerca infection screening**.
3. Click **Use this specimen type**.

That selects the Keras cm24_tf infection-screening model. Do not use the *Anopheles* age/species model on blackfly spectra.

3. Recommended file formats (in order)

Priority	Format	Notes
1	.dpt	Bruker OPUS export: two columns, wavenumber (cm-1) and absorbance, one spectrum per file. Example: example_spectra/sample_oncho_spectrum.dpt.
2	.mzz	OPUS compressed spectrum (same logical content as .dpt).

Priority	Format	Notes
3	Directory	Folder of <code>.dpt</code> / <code>.mzz</code> / <code>.csv</code> files; all readable spectra are scored.
4	Wide CSV	One row per fly; SampleID plus 2332 absorbance columns on the cm24 wavenumber grid (see §5).

Malaria-style .dat files (1625-point rows with AA metadata) are **not** aligned to the Oncho cm24 grid. They may be converted via the Python adapter but often fail wavenumber checks; prefer native Oncho formats above.

Binary OPUS measurement blocks (`.0`, `.1`, ...) should be exported to `.dpt` or `.mzz` before upload.

4. Spectral requirements (cm24 bundle)

- **Instrument context:** ATR mid-infrared spectra from the Oncho MIRS-AI programme (Bruker-class workflows).
- **Wavenumber grid: 2332** fixed columns from ~ 3699.4 to ~ 700.3 cm⁻¹ (bundle `wavenumber_columns` in `oncho_model_bundle/cm24_tf/bundle.json`).
- **Preprocessing: Raw absorbance** (no Savitzky–Golay derivatives for the production cm24 UI model).
- **Sample state:** Desiccated *Simulium* (whole fly or programme-standard aliquots, per field SOP).
- **Ground truth (training):** Infection labels from PCR/dissection (L1/L2/L3) in partner labs — not required for upload-only prediction.

5. Wide CSV layout (optional)

If you export a wide matrix instead of `.dpt`:

```
SampleID,3699.3986,3697.3388,...,700.3095
fly_001,0.018,0.017,...
```

- First column: **SampleID** (unique per spectrum).
- Remaining **2332** headers must match the cm24 bundle wavenumbers **exactly** (string decimals as in `bundle.json`).
- One row = one blackfly spectrum.

6. Example data

File	Purpose
<code>example_spectra/sample_oncho_spectrum.dpt</code>	Simple point-to-point Bruker-style example (~1700 points; mapped sparsely onto the cm24 training grid).
<code>oncho_model_bundle/cm24_tf/</code>	Production Keras model (<code>bundle.json</code> , <code>model.keras</code> , <code>model.weights.h5</code>) — do not edit.

Quick test: Select the Oncho profile → upload `sample_oncho_spectrum.dpt` → run prediction (requires TensorFlow and the `cm24_tf` bundle on the server). Spectra are mapped onto the sparse cm24 training grid (exact 4-decimal wavenumber labels; other columns zero), not densely interpolated.

7. Metadata and batch workflow

The same VectorPredictor workflow applies (locations, metadata, upload, predict). Record:

- Collection site and date
- Trapping method (e.g. human landing catch, sticky trap, host decoy)
- Species complex / site code if known

Batch uploads: multiple `.dpt` files or one directory in a single batch job.

8. Server prerequisites

Oncho predictions require:

- TensorFlow/Keras (Python 3.12)
- Keras `cm24_tf` bundle at `ONCHO_TF_BUNDLE_DIR` (default: `oncho_model_bundle/cm24_tf` in this repo)

See `ENVIRONMENT_SETUP.md` for configuration.

9. Outputs

Per spectrum, the platform returns infection-screening fields (e.g. probability, screened positive/negative per bundle threshold), not mosquito age or species.

10. Quality checks

- Confirm spectra are not empty and absorbance values are numeric.
- Avoid corrupted OPUS exports (truncated `.dpt` files).
- If prediction fails with “missing wavenumber columns”, the file is not on the cm24 grid — re-export from OPUS or use `.dpt` from the Oncho pipeline.

Version: 2026-08 — aligns with Keras cm24_tf raw CNN bundle in VectorPredictor.