

## 9) Real-time Pipeline Report (即時系統流程報告)

Date: 2025-12-24

### Objective

Describe the real-time FER inference pipeline used in the demo and how it achieves stable predictions.

### Main code

- Real-time demo: ``demo/realtime_demo.py``

### Pipeline stages (high level)

1. Frame capture (webcam/video)
2. Face detection
3. Face crop + resize to model input
4. Preprocessing (including CLAHE when enabled)
5. FER model inference (logits/probabilities)
6. Temporal smoothing / stabilization
7. Visualization + CSV logging

### Stabilization strategy

Implemented in ``demo/realtime_demo.py``:

- Exponential Moving Average (EMA) smoothing on probabilities
- Hysteresis / thresholding to prevent rapid label flips
- Voting window to stabilize final label

### Logging

- Per-frame CSV logs are written to the demo output directory (``demo/outputs/``).
- These logs support post-run analysis (FPS, flip-rate, confidence distribution).

TBD note (requires a real demo run):

- Performance KPIs (FPS/latency/flip-rate) should be computed from an actual `demo/outputs/\*.csv` log produced on the target machine.

### **Deployment notes**

- The pipeline is designed to run on Windows and should be tested on the target hardware to confirm throughput and latency.

### **Next steps**

- Add a small analysis script to compute:

- average FPS
- median per-frame latency
- label flip-rate per minute
- time-in-state for each expression

from the CSV log.