**Event Based Eye Movement**

**and Blink/No-Blink Detection**

**1. Dataset Overview**

* **Source**
  + Event-camera recordings of human eye movements (train 80 , validation 10, test 10 splits).
  + Dataset link: <https://www.kaggle.com/competitions/event-based-eye-tracking-ais2024/data>
  + Data stored per subject/session in HDF5 format with event streams and labels.
* **Windows**
  + 100 ms sliding windows (WIN\_SEC = 0.10 s).
  + Each window converted into a 5-bin voxel grid (time bins) of shape (T, 2, H, W).
* **Labels per window**
  + **Blink/No-Blink**
    - Binary (0 = open, 1 = blink).
  + **Retina Centre**
    - 2D real-valued (x, y) location of fixation center, normalized by image width/height.

**2. Preprocessing Pipeline**

1. **Event & Label Loading**
   * Read events dataset from HDF5: columns (x, y, t, polarity).
   * Fallback to label.txt when HDF5 labels missing.
2. **Voxel Encoding**
   * **Time binning**: Split each 100 ms window into NUM\_BINS (e.g. 5) equal-length bins.
   * **Spatial rasterization**: 2 channels (polarity positive/negative) per bin.
   * **Clipping**: Counts clipped to [–3, 3] to limit dynamic range.
   * **Resizing**: Down sample from (640×480) to (160×120) using area interpolation.
3. **Empty-Window Filtering**
   * Discard windows whose total event count sum(abs(voxel)) < MIN\_EVENTS (e.g. 10).
   * Eliminates near-blank samples conveying little information.
4. **Train/Val/Test Split**
   * Batching by subject: last subject of batch reserved for validation.
   * Entire held-out test directory processed unlabeled for inference.

**3. Model Architecture**

* **Spiking Convolutional Neural Network (SNN)**
  1. **3D Convolutions**
     + Three layers of Conv3D + BatchNorm3D + Leaky‐integrate‐and‐fire surrogate activation.
  2. **Pooling**
     + Global average pooling across time & space → features of size 128.
  3. **Auxiliary Count Map**
     + Flattened event‐count map (1, H, W) → H×W, concatenated to features.
  4. **Fully‐Connected Head**
     + Dense layer → ReLU → Dropout(0.5) → two separate heads:
       - **Blink**: linear → 1 logit
       - **Centre**: linear → 2 regression outputs

**4. Imbalance & Loss Handling**

* **Class Imbalance (Blink/No-Blink)**
  + **WeightedSampling**: Compute inverse‐frequency weights w = 1/count[class]; sample balanced batches.
  + **Focal Loss**
    - Down-weights easy negatives; focuses learning on hard/misclassified examples.
    - Hyperparameters:
      * α (class‐balance term) = 0.25
      * γ (focusing term) = 2.0
* **Multi-Task Loss**
  + **Blink loss**: Focal loss on binary logits
  + **Centre loss**: MSE on normalized (x, y)
  + **Total**:
  + Loss = L\_blink + λ\_center · L\_centre
    - Centre weight λ\_center = 0.001
    - Warm-up: blink‐only (λ\_center=0) for first 5 epochs

**5. Optimization & Smoothing Techniques**

1. **Optimizer & Scheduler**
   * **AdamW** (weight decay for L2 regularization = 1 × 10⁻³)
   * **CosineAnnealingWarmRestarts**
     + Initial period T₀=10 epochs, multiplicative factor T\_mult=2, minimum LR 1 × 10⁻⁶.
2. **Stochastic Weight Averaging (SWA)**
   * Start at epoch 20
   * Maintain running average of model parameters → smoother, flatter optima.
3. **Exponential Moving Average (EMA)**
   * Shadow copy of parameters updated each batch:
   * shadow = decay·shadow + (1−decay)·current\_param
   * Applied at end to stabilize final weights.
4. **Early Stopping**
   * Monitor validation loss; patience = 10 epochs without improvement → stop training.
   * Save **best** model checkpoint when validation loss decreases.

**6. Data Augmentation & Regularization**

* **Spatial Jitter**
  + Random integer shifts (±5 px) in width & height; roll both voxel and count map.
* **Random Flips**
  + Horizontal and vertical flips with probability 0.5; adjust normalized centre accordingly.
* **Time‐Bin Masking**
  + Zero‐out one random time bin per window (p = 0.3) to improve temporal robustness.
* **MixUp**
  + Interpolate pairs of examples (inputs & labels) with β‐distribution (α=0.2).
* **Dropout**
  + 50 % after the dense layer to prevent over-fitting.

**7. Training Summary**

* **Epochs**: up to 50
* **Batch Size**: 8
* **Learning Rate**: starts at 1 × 10⁻⁵, *blink-only* warm-up at 0.1× for first 5 epochs
* **Weight Decay**: 1 × 10⁻³
* **Early Stop Patience**: 10
* **SWA Start**: epoch 20

**8. Expected Outcomes**

* **Smoother Learning Curves**
  + SWA + EMA reduce variance; warm restarts encourage flat minima.
* **Balanced Blink Accuracy**
  + Focal loss + weighted sampling ensure both blink & open classes are learned.
* **Robust Centre Regression**
  + MSE weighted lightly avoids dominating blink objective.
* **Generalization**
  + Aggressive augmentation + regularization to prevent over-specialization.