**Report – Selecting an Optimal Temporal Window for Transformer-based Action Recognition**

*(PhD thesis supporting document)*

**1 Background and Problem Statement**

Transformer architectures have become a competitive choice for frame-level action recognition in combat-sports video. A key design hyper-parameter is the **temporal window length** (sequence length *T*) on which the model attends. If *T* is too short the model loses motion context; if too long, memory and latency escalate and many windows are dominated by padding. This study establishes an evidence-based window size for a boxing/kick-boxing dataset labelled with YOLO detectors.

**2 Dataset and Pre-processing**

| **Clip** | **Rows** | **FPS** | **Classes used** | **CSV header** |
| --- | --- | --- | --- | --- |
| yolo\_predictions\_1.csv | 10 874 | 30 fps | 1 (high-guard), 2 (kick-knee), 5 (punch) | frame,x1,y1,x2,y2,confidence,class\_id |
| yolo\_predictions\_4.csv | 4 117 | 30 fps | 1, 5 | same |

*Non-action rows* such as the background “person” track (class\_id = 4) were **filtered out** to prevent extreme run-length outliers (e.g. 451-frame person tracks).

Each clip was sorted by class\_id, frame; for each class we computed contiguous-frame run IDs  
run\_id = (frame.diff() != 1).cumsum() and aggregated segment lengths.

**3 Exploratory Statistics**

| **Clip** | **Class** | **Count** | **Median** | **P90** | **P95** | **Max** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | high-guard | 21 | 2 f | 8 f | 11 f | 14 f |
|  | kick-knee | 12 | 3 - 4 f | 7 f | 7 - 8 f | 8 f |
|  | **punch** | 31 | 7 f | 17 f | 19 f | 23 f |
| 4 | high-guard | 13 | 12 f | 15 f | 15 - 16 f | **16 f** |
|  | punch | 13 | 9 f | 11 f | 11 f | 11 f |

Across **all** clips the global 95-percentile segment length is **≤ 19 frames** and the absolute maximum is **23 frames** (0.77 s at 30 fps).

**4 Heuristic for Window Selection**

We adopt

Twindow=2⌈log⁡2(1.5×P95)⌉=32 frames,T\_\text{window}=2^{\lceil\log\_2(1.5\times P95)\rceil}=32\text{ frames},

where the ×1.5 safety factor ensures complete coverage of > 95 % of actions while aligning with power-of-two GPU kernels.

A **stride** of T/2=16T/2 = 16 frames maintains 50 % overlap, giving each frame at least two independent contextual views and permitting max/mean fusion of overlapping logits without excessive compute.

**5 Recommended Model Configuration**

| **Hyper-parameter** | **Value** | **Rationale** |
| --- | --- | --- |
| Sequence length *T* | **32** | Fully contains the longest punch instance (23 f) with margin. |
| Stride | **16** | 50 % overlap ⇒ dense coverage & linear inference cost. |
| Embedding | 1-D Conv patch → d\_model = 64 | Allows flexible up/down-sampling of input length. |
| Positional encoding | **Relative** or learned | Avoids re-interpolation if *T* changes. |
| Classification head | Adaptive max-pool → FC | Produces window-level logits independent of *T*. |

Memory use:  
32 frames × 64 d × FP32 ≈ 0.2 MB/token; batch sizes > 32 fit on a 24 GB card.

**6 Real-time Deployment Variants**

* **Streaming causal transformer.**  
  Maintain past key/value cache; update with one new frame. Preserves 32-frame context at **O(1)** extra FLOPs per step.
* **Light GRU-hybrid.**  
  3-layer 1-D CNN + bidirectional GRU over 16 frames for ultra-low-power devices.
* **Single-frame (“row-by-row”) mode** is discouraged; removing context degrades punch-vs-guard discrimination and increases false positives.

**7 Validation Plan**

1. Grid-search mini-runs (3–5 epochs) with T∈{16,24,32}T\in\{16,24,32\}; monitor event-level mAP.
2. Fine-tune sigmoid threshold (currently 0.8) on the validation set via PR-curve analysis.
3. Confirm no unseen clip exceeds 24 frames per action; otherwise re-evaluate *T*.

**8 Conclusions**

* Empirical segment analysis supports **32-frame windows with 16-frame stride** as a robust trade-off for combat-sports action recognition.
* Filtering non-action tracks is essential; person-presence outliers skew raw statistics by an order of magnitude.
* Maintaining a causal cache affords real-time inference without the accuracy drop inherent in single-frame models.

These findings will guide the forthcoming full-scale training experiments and form the baseline for ablation studies in Chapter 5 of the thesis.

**Appendix A – Script for Segment Statistics**

python check\_sequences\_analysis.py <csv>

Key excerpt:

df = pd.read\_csv(csv\_path)

df = df[df["class\_id"].isin([1,2,5])]

run\_id = (df["frame"].diff() != 1).cumsum()

seg\_lens = df.groupby(["class\_id", run\_id]).size()

*Prepared 18 June 2025*