Ball action spotting

Team 6:

- María José Millán
- Agustina Ghelfi
- Laila Aborizka

INTRODUCTION TO THE PROBLEM

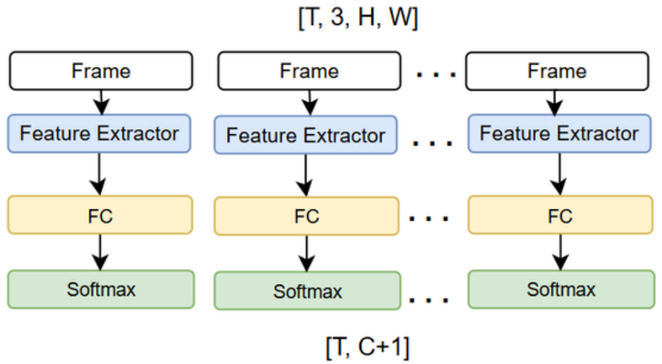
In this work, we address the task of ball action spotting using the SoccerNet dataset, which contains 10.5 hours of annotated 720p football broadcast footage from seven matches. The dataset is split into training, validation, and test sets.

Goal: Precisely detect time instants where specific ball actions occur, across 12 defined categories such as pass, shot, or goal.

Challenges: subtle action boundaries, context dependency, class imbalance.

STARTING POINT: BASELINE MODEL

The baseline model served as the reference point for all experimental comparisons. We trained the baseline model, using AdamW optimizer, learning rate 8e-4, batch size 4 and training for a maximum of 20 epochs and 3 warm-up epochs.



Feature extractor:
RegNet-Y (200MF) with its final FC layer removed – extracts frame-level features.

FC layer:
Maps features to C + 1 outputs (number of classes + 'no-action' category).

To classify each frame into an action category a softmax activation is applied.

Sample FPS: 25
Resolution: 398x224
Clip Length: 50
Temporal Overlap: 90% (train & val) - 0% (test)
Stride: 2

BASELINE

mAP-12 = 6.78
mAP-10 = 8.13

INITIAL EXPERIMENTS

The experiments presented in this section focus on improving the feature extractor. To achieve this, various backbone architectures and temporal modeling strategies were explored to obtain more accurate frame-level action representations. The models in this section were trained with the same configuration parameters as the baseline model, applying early stopping with a patience of 5 epochs and delta of 0.01.

1. EXPERIMENTS WITH EFFICIENTNETB0

EfficientNet-B0 was chosen for experimentation due to its optimized architecture and compound scaling strategy, which were hypothesized to enable the extraction of richer and more detailed features beneficial for capturing complex football actions, potentially leading to improved model accuracy.

1.1. Replace RegNetY with EfficientNetB0.

EfficientNetB0 + FC

mAP-12 = 8.19 (+1.41)
mAP-10 = 9.82 (+1.69)

1.2. Introduce Temporal Convolutional Network (TCN).

EfficientNetB0 + TCN + FC

mAP-12 = 17.46 (+10.68)
mAP-10 = 20.95 (+12.82)

This design allows the model to capture short-range temporal dependencies between frames without downsampling or compressing the sequence over time.

BASELINE

mAP-12 = 6.78
mAP-10 = 8.13

INITIAL EXPERIMENTS

The experiments presented in this section focus on improving the feature extractor. To achieve this, various backbone architectures and temporal modeling strategies were explored to obtain more accurate frame-level action representations. The models in this section were trained with the same configuration parameters as the baseline model, applying early stopping with a patience of 5 epochs and delta of 0.01

2. EXPERIMENTS WITH X3D

X3D's architecture inherently models temporal information, unlike EfficientNet which requires additional mechanisms.
We decided to experiment with X3D as it yielded the best performance across several groups in the previous week.

2.1. Replace RegNetY with X3D_s, keeping only the first 5 blocks.

X3D_s + FC

mAP-12 = 37.31 (+30.53)
mAP-10 = 42.66 (+34.53)

2.2. Introduce Temporal Convolutional Network (TCN).

X3D_s + TCN + FC

mAP-12 = 34.38 (+27.60)
mAP-10 = 40.65 (+32.52)

X3D already incorporates temporal modeling through its 3D filters, which might have made the addition of TCN redundant for this specific task.

2.3. Increase model complexity: X3D_m.

X3D_m + FC

mAP-12 = 34.81 (+28.03)
mAP-10 = 39.96 (+31.83)

2.4. Reduce the number of blocks: keep only the first 4 blocks.

X3D_s + FC

mAP-12 = 32.01 (+25.23)
mAP-10 = 38.16 (+30.03)

BASELINE

mAP-12 = 6.78
mAP-10 = 8.13

INITIAL EXPERIMENTS

The experiments presented in this section focus on improving the feature extractor. To achieve this, various backbone architectures and temporal modeling strategies were explored to obtain more accurate frame-level action representations. The models in this section were trained with the same configuration parameters as the baseline model, applying early stopping with a patience of 5 epochs and delta of 0.01

3. EXPERIMENTS WITH TRANSFORMER ENCODER

The Transformer architecture leverages Positional Encoding to incorporate temporal order into input sequences.

Transformer uses self-attention to capture long-range dependencies across time without reducing spatial or temporal resolution. This allows the model to refine features at the frame level.

3.1. Replace RegNetY with I3D and introduce a Transformer Encoder architecture.

I3D + Transf Enc + FC

mAP-12 = 27.80 (+21.02)
mAP-10 = 26.39 (+18.26)

3.2. Replace RegNetY with X3D_s and introduce a Transformer Encoder architecture.

X3D_s + Transf Enc + FC

mAP-12 = 34.56 (+27.78)
mAP-10 = 36.01 (+27.88)

Although X3D combined with a Transformer outperforms I3D, this gain is relatively modest compared to earlier experiments. This suggests that while X3D preserves temporal resolution it may still lack the fine-grained temporal modeling capabilities required for precise ball action spotting. Additionally, X3D is significantly lighter than I3D, which may contribute to its relative improvement.

BASELINE

mAP-12 = 6.78
mAP-10 = 8.13

SGP (SCALABLE-GRANULARITY PERCEPTION)

In ball action spotting, capturing both precise frame-level cues and short-term temporal context is essential for accurate event localization. Inspired by TriDet we implemented SGP that is a temporal module that processes video features at multiple granularities, enhancing temporal discriminability at multiple scales. This structure has two principal branches:

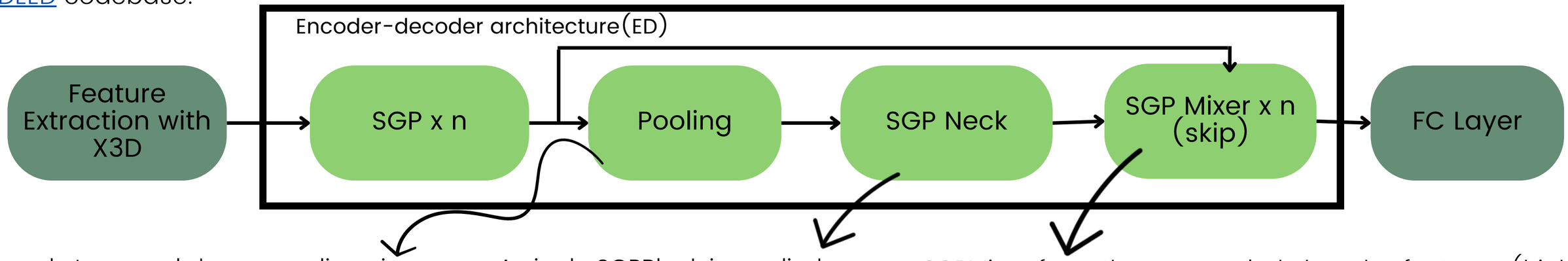
- **Instant-Level Perception (frame-wise)**: focuses on modeling fine-grained, frame-specific variations by subtracting the temporal mean from each frame, effectively removing redundant global information.

- **Temporal Window Perception (context-aware)**: To capture the temporal evolution of actions, this branch applies 1D convolutions with multiple dilation rates (1, 2, 3) over the temporal axis. The resulting features are concatenated and fused enabling the model to recognize temporal dynamics such as motion trajectories.

X3D + SGP + FC	X3D + TCN + SGP + FC	BASELINE
mAP-12 = 36.15 (+29.37) mAP-10 = 42.47 (+34.34)	mAP-12 = 36.19(+29.41) mAP-10 = 42.06(+33.93)	mAP-12 = 6.78 mAP-10 = 8.13

SGP-MIXER ENCODER-DECODER LAYER

Although SGP improves frame-to-frame discriminability within a sequence, it doesn't capture the interaction between temporal scales well. To address this, we propose an extension of SGP within an encoder-decoder architecture, which blends information from different temporal scales while improving frame discriminability. For the implementation, we integrated components from the TDEED codebase.



We apply temporal downsampling via AdaptiveMaxPool1d to progressively reduce the temporal resolution and capture long-range context. Intermediate features at each level are stored for later fusion in the decoder through skip connections.

A single SGPBlock is applied at the network bottleneck to refine compressed features and preserve temporal discriminability.

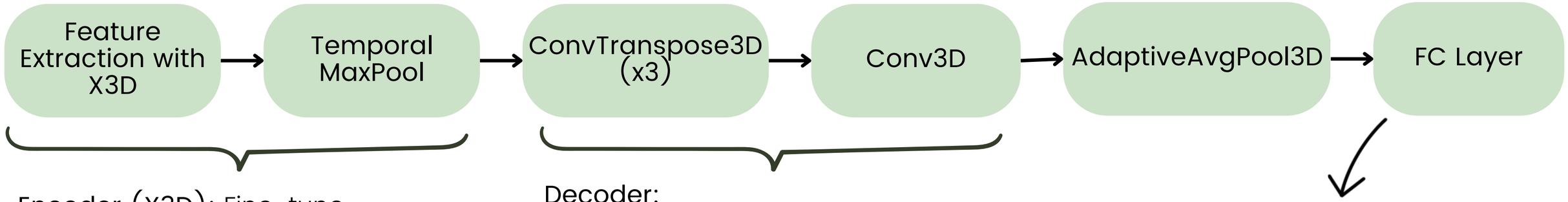
SGPMixer fuse, the upsampled decoder features (high-level, but temporally coarse) and the skipped encoder features (low-level, but temporally fine). It applies:

- Separate instant-level and window-level branches to both inputs.
- A concatenation + linear projection.
- Final refinement via a lightweight MLP block.

X3D_s + ED + FC BEST!	Focal loss X3D + ED + FC	X3D_m + ED + FC	BASELINE
mAP-12 = 42.96 (+36.18) mAP-10 = 46.10 (+37.97)	mAP-12 = 41.37 (+34.59) mAP-10 = 43.11 (+34.98)	mAP-12 = 39.08 (+32.30) mAP-10 = 44.17 (+36.04)	mAP-12 = 6.78 mAP-10 = 8.13

X3D ENCODER-DECODER ARCHITECTURE

The goal is to check how temporal aggregation can affect (positively or negatively) the spotting performance.
To study this, we implemented an encoder-decoder architecture with upsampling, where temporal resolution is first reduced and then recovered using transposed convolutions. Classification is performed on the temporally reconstructed signal.



Encoder (X3D): Fine-tune pretrained X3D-S backbone with last block removed, followed by MaxPool3D for temporal downsampling (factor of 8).

Decoder:

- 3 stacked ConvTranspose3D layers progressively upsample in the temporal dimension (6 → 11 → 21 → 41 frames).
- Final Conv3D aligns channel dimensions with encoder output.
- Optional trilinear interpolation adjusts to target 50-frame length.

FC Layer: Linear layer maps each timestep to output classes. Prediction is per frame.

X3D Enc + Dec + FC

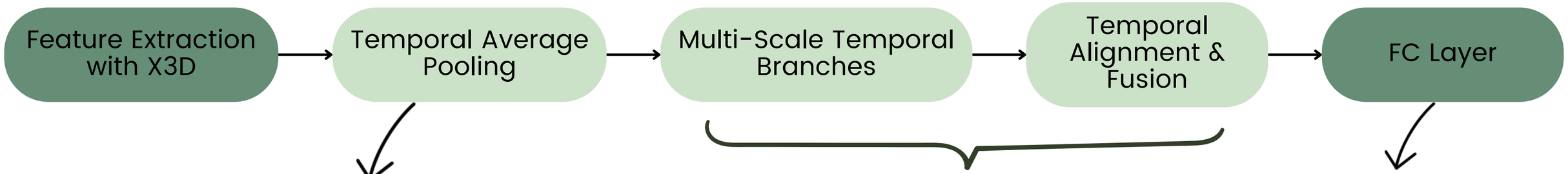
mAP-12 = 32.31 (+25.53)
mAP-10 = 38.09 (+29.96)

BASELINE

mAP-12 = 6.78
mAP-10 = 8.13

TEMPORAL PYRAMID NETWORK

The goal is to assess how combining short- and long-term information influences recognition accuracy so we use a Temporal Pyramid Network that processes frames at multiple temporal scales. Each branch compresses the temporal resolution to capture different context lengths, and the outputs are later up-sampled and fused. Classification is then performed on the 'aggregated' signal.

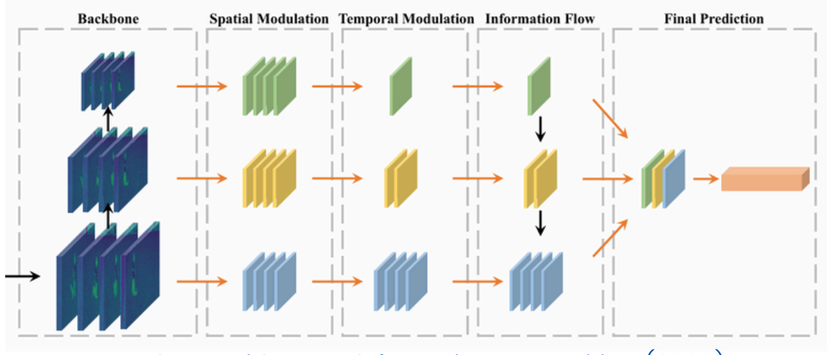


This operation compresses each frame into a single vector by averaging over its spatial features, focusing the model on **what happens over time**, rather than where it happens in the frame.

Temporal Pyramid:

- Several parallel branches down-sample the temporal axis at different strides (1×, 2×, 4×)
- The outputs of these branches are up-sampled back to the original length using nearest-neighbor interpolation and concatenated.

FC Layer: Linear layer maps each timestep to output classes. Prediction is per frame.



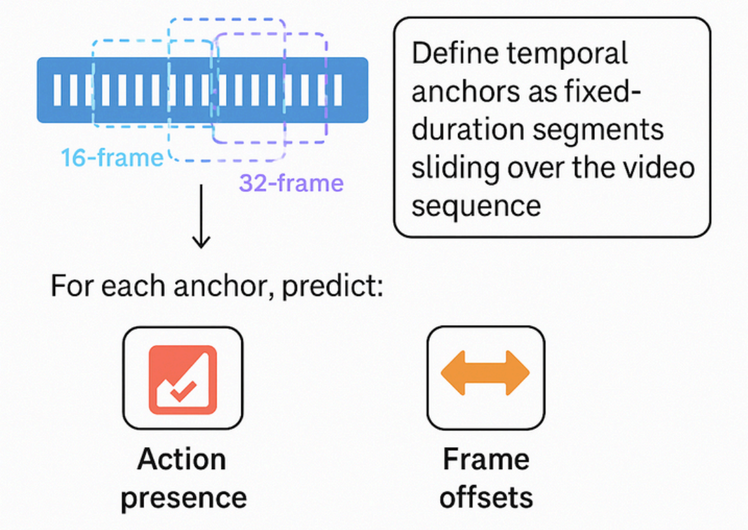
Temporal Pyramid Network for Action Recognition (2020)

TPN
mAP-12 = 41.42 (+34.64)
mAP-10 = 42.43 (+34.30)

BASELINE
mAP-12 = 6.78
mAP-10 = 8.13

OTHER INTERESTING IDEAS EXPLORED

1. ANCHOR BASED APPROACHES



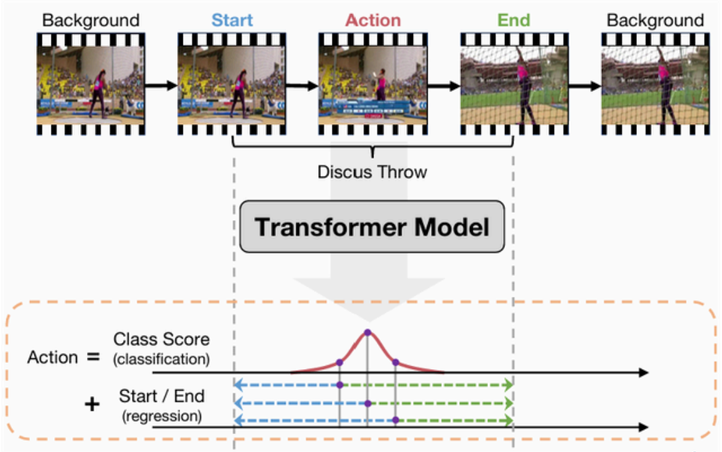
Why? Frame-level anchors provide a systematic way to generate action proposals while leveraging temporal context.

Define temporal anchors (8,16-frame, 32-frame windows)
For each anchor, predict:

- Action presence (whether it contains an action).
- Frame offsets (refining the start, center, end frames for precise boundary adjustment).

Why it didn't work? The implementation faced challenges.

2. ACTION-FORMER



[ActionFormer: Localizing Moments of Actions with Transformers(2022)]

Why? SOTA framework for temporal action localization in videos

Transformer based model to localize action instances in time by

- classifying every moment into action categories.
- estimating their distances to action boundaries.

Why it didn't work? Heavy and lack of time.

FINAL RESULTS & CONCLUSIONS

Class	BASELINE: REGNETY002 AP	PREVIOUS WEEK: X3D (S) + FC AP	THIS WEEK: X3D (S) + ED + FC AP
PASS	29.89	75.27	76.12
DRIVE	21.12	68.38	67.26
HEADER	3.56	53.36	48.83
HIGH PASS	13.47	77.27	71.08
OUT	0.45	15.1	14.4
CROSS	2.04	41.79	57.36
THROW IN	10.24	54.56	64.42
SHOT	0.4	38.87	35.62
BALL PLAYER BLOCK	0.21	12.02	16.84
PLAYER SUCCESSFUL TACKLE	0	0	9.09
FREE KICK	0	21.11	54.55
GOAL	0	0	0
Average Precision 12	6.78	37.31	**42.96**
Average Precision 10	8.13	42.66	**46.10** BEST!

Key Observations

- FREE KICK had a huge boost from **0 to 54.55 AP** in the best model.
- PASS, DRIVE, HEADER, and HIGH PASS consistently show high AP across all the weeks

Temporal modeling is crucial

Architectures that effectively capture short- and long-range temporal dependencies (e.g., TCN, Transformer, SGP, TPN) consistently outperformed those relying solely on spatial feature extraction.

Backbone choice matters

Switching from RegNetY to X3D led to the most significant performance boost, highlighting the importance of choosing architectures that inherently model spatiotemporal information.

Multi-scale temporal approaches dominate

Models that combined fine-grained frame-level analysis with broader temporal context (e.g., SGP, TPN, ED) yielded the best mAP scores, suggesting that there is some dependency on subtle temporal cues for accurate ball action spotting.