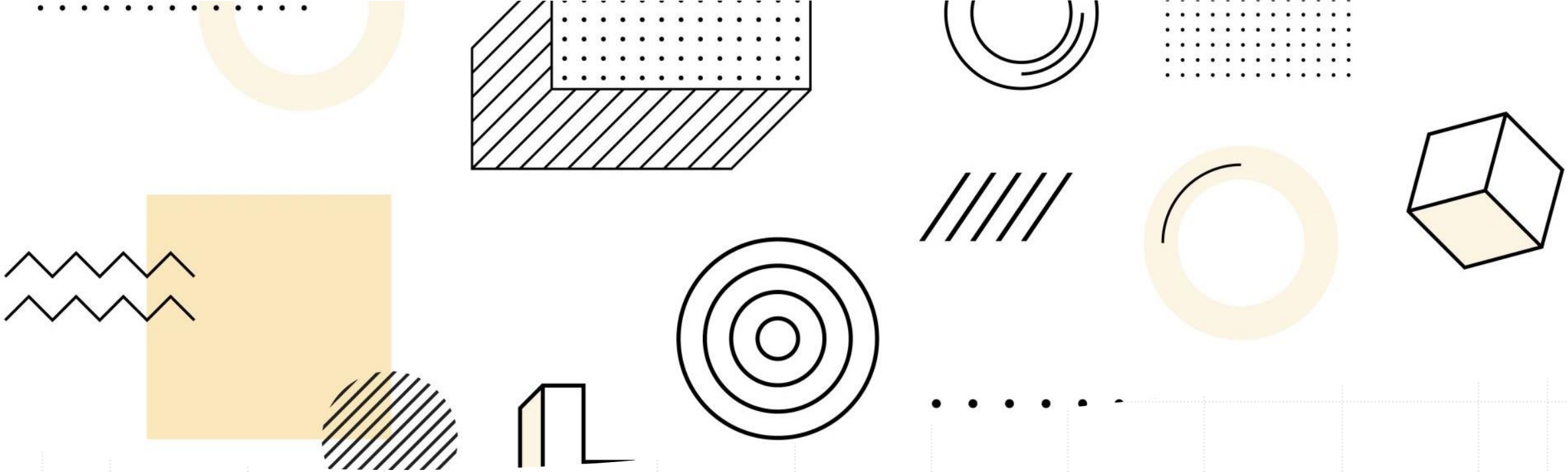


# ENCODiT

## Error flagging and Neutralisation using Conformal Out-of-Distribution Detection in Time-Series Data


**Group 14**

1. Kartik Anant Kulkarni (210493)
2. Rishi Agarwal (210849)
3. Emaad Ahmed (210369)
4. Dhruva Singh Sachan (210343)

# Outline

Problem Statement

Proposed Solution

Verification and Experimentation

Results and Conclusions

Future Work

# Background

- Usage of Learning Enabled Components (LECs) in Cyber-Physical Systems (CPS)
- What is Out Of Distribution (OOD)?
- Safety Insurance in Real-time Applications
- Common Solutions by Cascading Models

# Demerits of Other Approaches

- **Temporal Relationships** in Time-Series Sequences not exploited. E.g. Frozen Camera, Drift Detection etc. cannot be detected by most point-based methods.
- No work has been done on **non-pictorial** data. E.g. GAIT Detection in medical CPS, etc. has not explored.

Frozen Camera

Drift

# Key Contributions of the Paper

## Novel Measure for OOD Detection in Time-Series Data for CPS

- Proposed a **non-conformity measure** that is defined on the window containing information about the sequence of time-series datapoints for enhancing detection of the temporal OODs
- Used a model trained to learn **iD temporal equivariance** via the auxiliary task of **predicting an applied transformation** on windows drawn from training distribution of LEC.
- Used error in this prediction as the non-conformity measure in **ICAD** framework for detection in time-series data for CPS.

## Enhanced detection performance

- Use **Fisher's method** as an ensemble approach for combining predictions from multiple conformal detectors based on the proposed measure

## CODiT

- Compute $n$ **independent** $p$**-values** of the input from the proposed measure in the ICAD framework, and combining these values by Fisher's method leads to the proposed detector CODiT with a **bounded false alarm rate** due to ICAD framework on high-dimensional input space.

# Comparison With Other Approaches

**Table 1: Capabilities of detectors in time-series data for CPS.**

| OOD Detector | False Alarm Rate Guarantees | Temporal OODs | Non-vision Data |
|---|---|---|---|
| VAE [5] | ✓ | ? | ✓ |
| $\beta$-VAE[27] | ✓ | ? | ✓ |
| Memory [36] | ✗ | ? | ✓ |
| Feng et al.'s [10] | ✗ | ✓ | ✗ |
| CODiT (Ours) | ✓ | ✓ | ✓ |

- It is not clear how to directly apply individual point detectors to time-series data with the ICAD guarantees due to the following two reasons:
  - First, even if we apply these detectors to individual datapoints in the time-series window independently, we do not know how to combine detection verdicts on these datapoints for detection on the window.
  - Second, for detection guarantees by ICAD, it is required that all non-conformity scores for $p$-value computation to be IID
- CODiT's approach is not limited to CNN and can be used for other predictive models as well and does not rely on point-based approaches and optical flows.

# Mathematical Formulation

Definition 1 (Schmidt and Roth, 2012). *For a set $X$, a function $f$ is defined to be equivariant with respect to a set of transformations $G$, if there exists the following relationship between any transformation $g \in G$ of the function's input and the corresponding transformation $g'$ of the function's output:*

$$f(g(x)) = g'(f(x)), \forall x \in X. \tag{1}$$

Definition 2 (Jenni and Jin, 2021). *Temporal equivariance of a function $f$ from equation 1 is defined on a set $X$ of windows of consecutive time-series datapoints and with respect to a set $G$ of temporal transformations.*

ICAD

The training dataset $X$ of size $l$ is split into a *proper training set* $X_{\text{tr}} = \{x_j : j = 1, \ldots, m\}$ and a *calibration set* $X_{\text{cal}} = \{x_j : j = m+1, \ldots, l\}$. Proper training set $X_{\text{tr}}$ is used in defining NCM. In the example of reconstruction error by a VAE as the non-conformity score, the VAE trained on $X_{tr}$ is used for computing the error. Calibration set $X_{\text{cal}}$ is a held-out training set that is used for computing $p$-value of an input. $p$-value of an input $x$ is computed by comparing its non-conformity score $\alpha(x)$ with these scores on the calibration datapoints:

$$p\text{-}value(x) = \frac{|\{j = m+1, ..., l : \ \alpha(x) \leq \alpha(x_j)\}| + 1}{l - m + 1}. \tag{2}$$

# Ensembling : Fisher's Method

The same hypothesis can be tested by multiple conformal predictors and an ensemble approach for combining these predictions can be used to improve upon the performance of individual predictors. Fisher's method is one of these approaches for combining multiple conformal predictions or $p$-values of an input from (2). Fisher value of an input $x$ from $n$ $p$-values is computed as follows:

$$\text{fisher-value}(x) = r \sum_{i=0}^{n-1} \frac{(-\log r)^i}{i!}, \text{ where } r = \prod_{k=1}^{n} p_k. \qquad (3)$$

Lemma 2 (Toccaceli and Gammerman, 2017). *If $n$ $p$-values, $p_1, \ldots, p_n$, are independently drawn from a uniform distribution of these values, then $-2\sum_{i=1}^{n} \log p_i$ follows a chi-square distribution with $2n$ degrees of freedom. Thus, the combined p-value is*

$$Pr\left(y \leq -2 \sum_{i=1}^{n} \log p_i\right) = r \sum_{i=0}^{n-1} \frac{(-\log r)^i}{i!},$$

*where $r = \prod_{k=1}^{n} p_k$, $y$ is a random variable following a chi-square distribution with $2n$ degrees of freedom, and the probability is taken over $y$. Moreover, the combined p-value follows the uniform distribution.*

# Algorithm with Guarantee on Correctness

- Theorem 1. The probability of false OOD detection on $Xt,w$ by Algorithm 1 is upper bounded by $\epsilon$, the false alarm rate.
- CrossEntropyLoss ($g$,$M(g(Xt,w))$) is used as the TTPE-NCM.
- Multiple p-values are used as an OOD window is less likely to behave as an iD window under multiple transformations.

**Algorithm 1** CODiT: OOD Detection in Time-Series Data for CPS

**Input:** a window $X_{t,w}$ of time-series data, VAE model $M$ trained on proper training set of the iD windows for LEC, distribution $Q_{G_T}$ over the set $G_T$ of temporal transformations, prediction error function $f$, $n$ sets of calibration set alphas $\{\alpha_j^k : 1 \leq k \leq n, m+1 \leq j \leq l\}$, and desired false alarm rate $\epsilon \in (0, 1)$

**Output:** "1" if $X_{t,w}$ is detected as OOD; "0" otherwise

**for** $k \leftarrow 1, \ldots, n$ **do**

  $g \sim Q_{G_T}$

  $\hat{g} \leftarrow M(g(X_{t,w}))$

  $\alpha \leftarrow f(g, \hat{g})$

  $p_k \leftarrow \frac{|\{j=m+1,\ldots,l:\ \alpha \leq \alpha_j^k\}|+1}{l-m+1}$

**end for**

$r \leftarrow \prod_{k=1}^{n} p_k$

**if** $r \sum_{i=0}^{n-1} \frac{(-\log r)^i}{i!} < \epsilon$ **then return** 1 **else return** 0

# Results

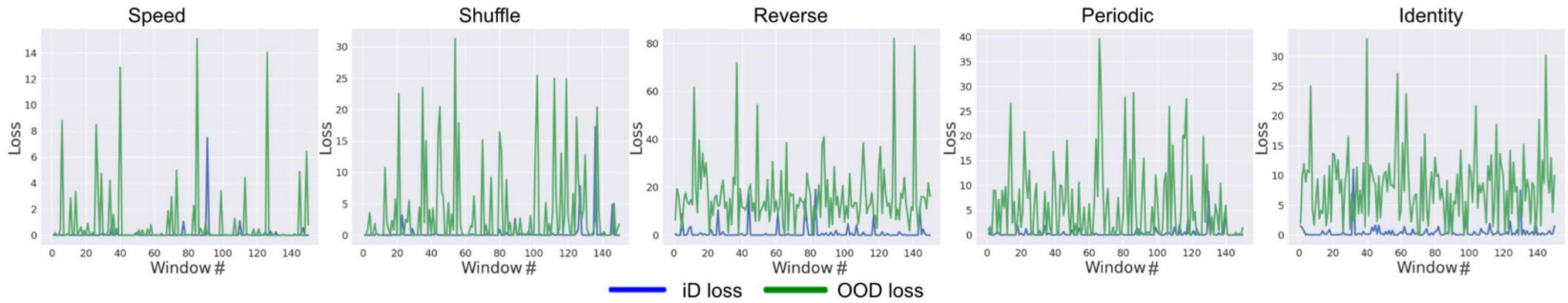



**Figure 4: Higher values of TTPE-NCM = CrossEntropyLoss$(g, M(g(X_{t,w})))$ on OOD windows than on the test iD windows of the drift dataset. This shows that $G_T$-equivariance learned on the windows drawn from the training distribution of LEC is less likely to generalize on the windows drawn from OOD.**

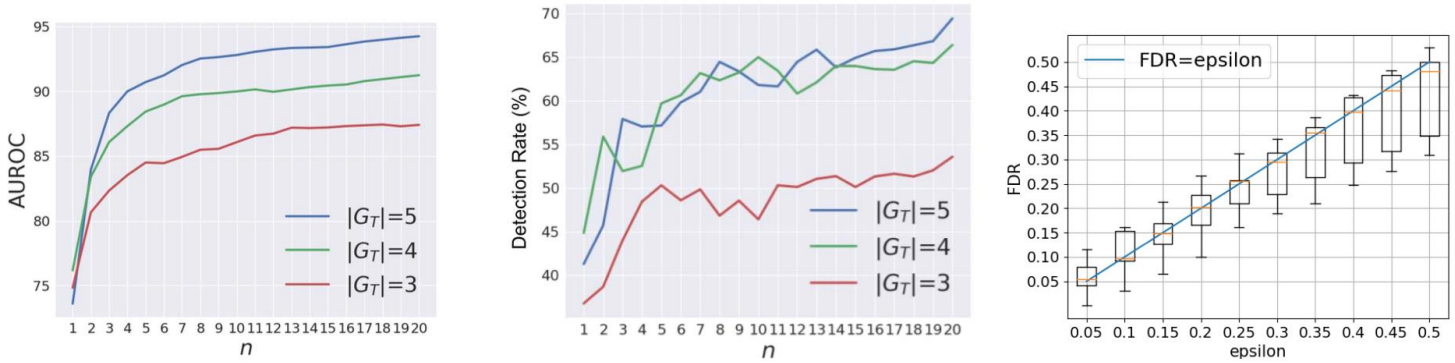
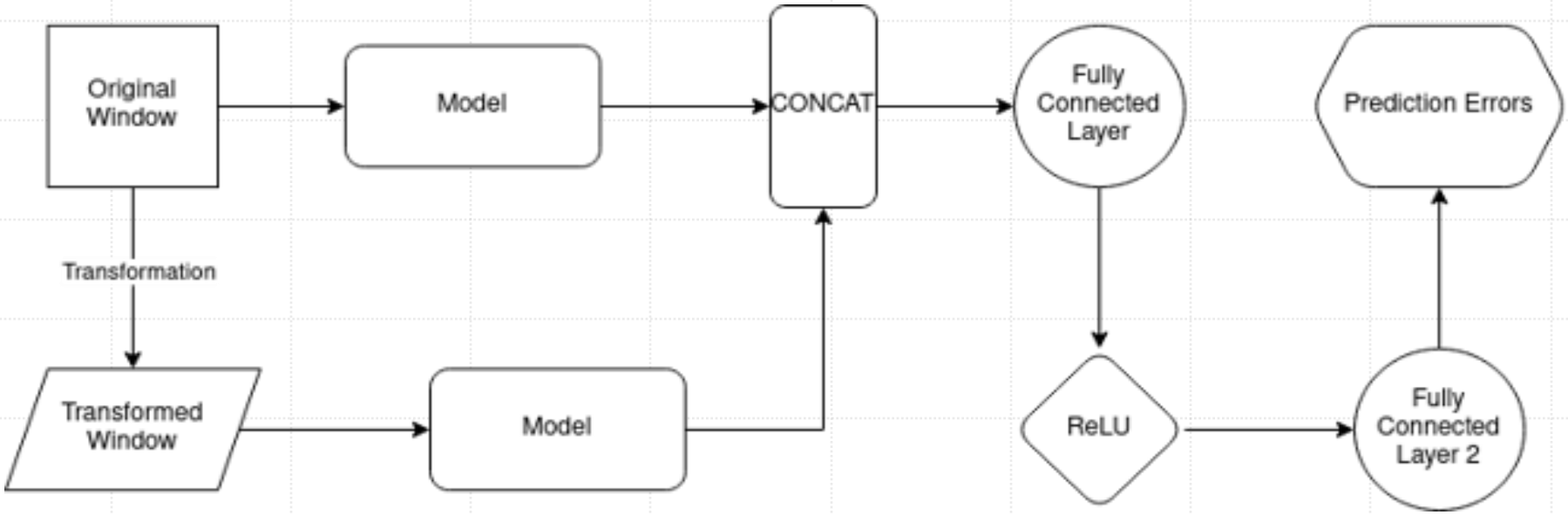


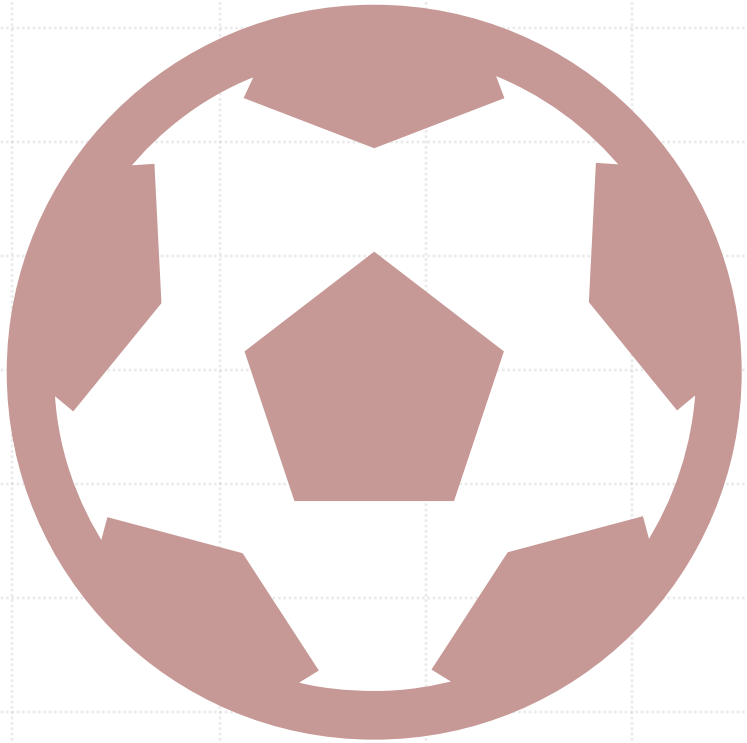

**Figure 5: AUROC vs. $n$ (left), TNR (with detection threshold at 95% TPR) vs. $n$ (center) shows that the performance of CODiT increases with the increase in the number $n$ of $p$-values used in the fisher-value for detection. False Alarm Rate (FDR) of CODiT is empirically bounded by $\epsilon$ on average (right). The yellow line in the box plot indicates the median.**

# General Architecture



# Verification and Experimentation

- We ran their algorithm for multiple random seeds and verified the results mentioned in the paper for the datasets given in the paper.
- Then, we extended the algorithm to the novel use case for ball trajectory classification, generated a dataset and implemented the algorithm.
- We modified the LeNet CNN model to get best performance of 92.314 AUROC for a window size of 36 on our custom dataset, which is close to the max accuracy.
- Transformations used on our dataset are- [Dilation, Erosion, High_Pass, Low_Pass, High_Low, Low_high].
- Further, to improve its real-time usage and applicability, we substituted the overall pipeline model with a LSTM model and identified the potential of efficient capture of temporality they offer.



# RoboCup Middle Size League

- International competition in which a team of five fully autonomous robots play with a FIFA-sized soccer ball
- Involves CPS spanning modules with LEC:
  - Swarm Robotics
  - Perception Systems
  - Localization Challenges
  - Decision Making
  - Path Planning

# Challenge

Feedback Based Mechanisms might lead to erroneous outputs and unpredictable results

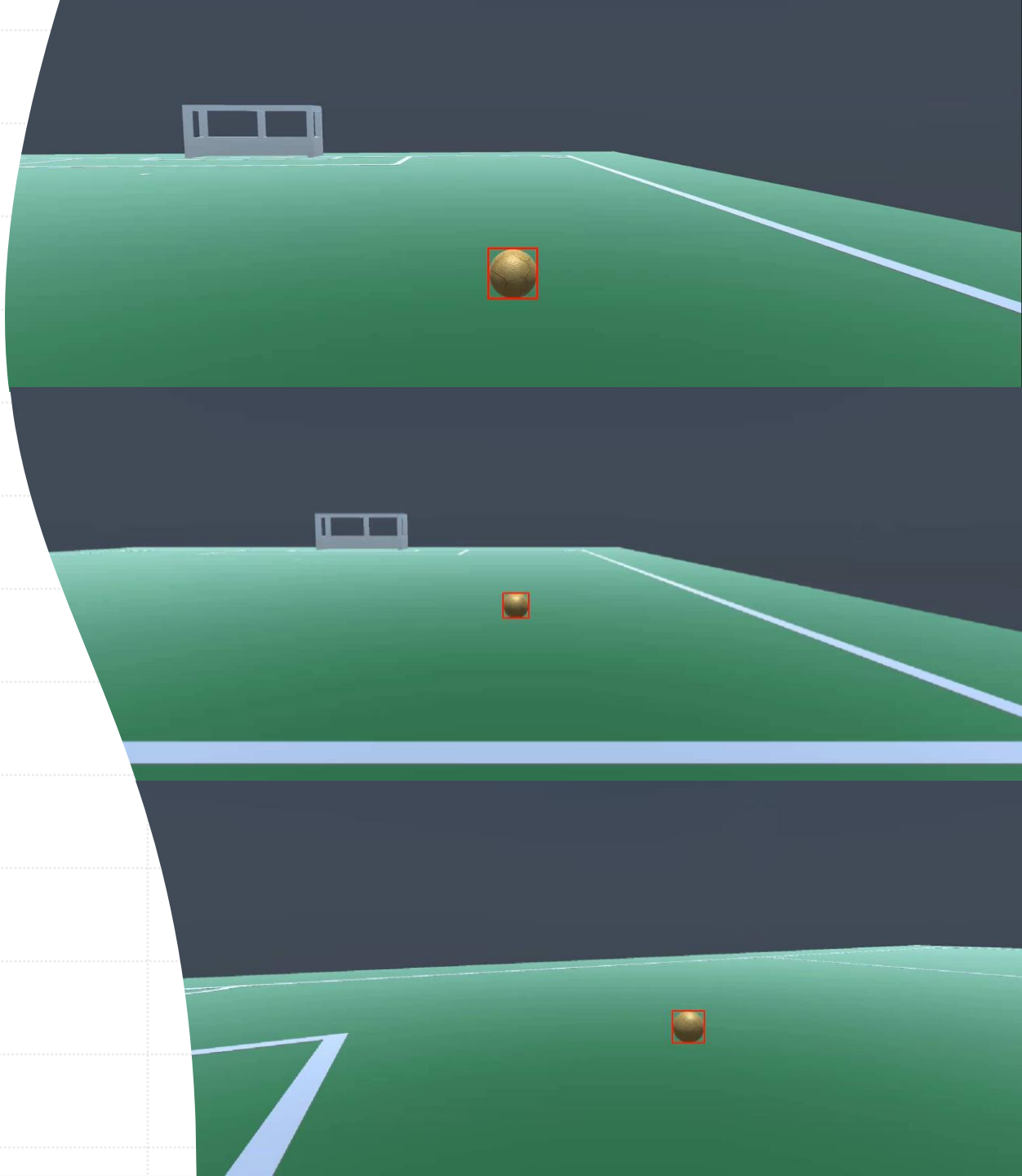
Methods to detect and handle hardware and sensor failures

Robust Ball trajectory prediction,

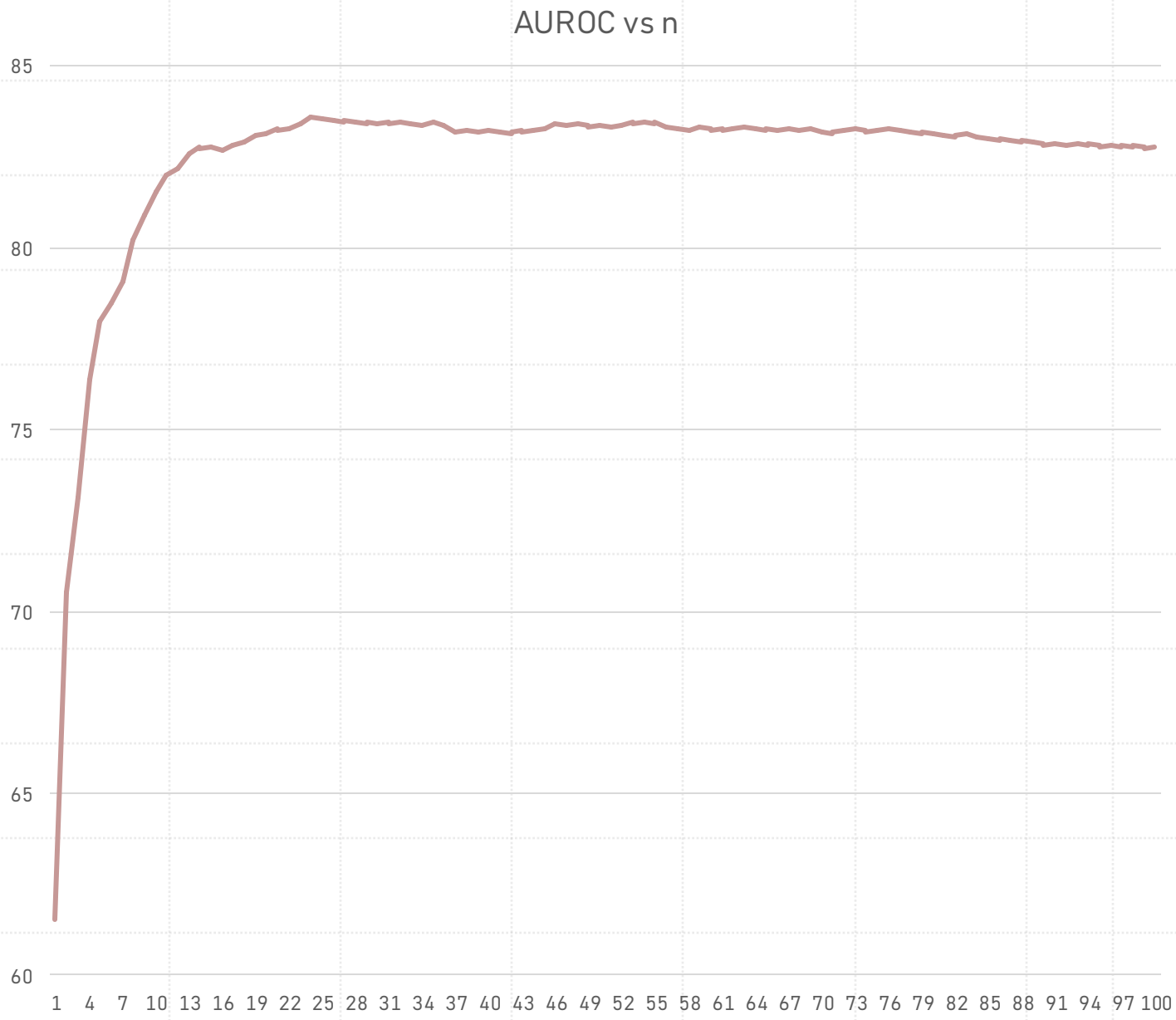
- Kicked ball can have an aerial or grounded trajectory in real life
- On ground, ball might show non-linear trajectory
- Estimation of ball trajectory is further used in decision making and game strategization

# Using CODiT with Ball Trajectory Model

- Why OOD detection in Ball Trajectory
- Dataset Creation
  - Made use of Unity Game Engine for creating a simulation
  - Generated 60 frame video clip dataset with different kicking speeds and camera angles
  - Test Dataset consisted of ball in a curved trajectory
- Further Image Processing,
  - Dataset was further processed to mask and detect the soccer ball in video
  - Bounding box was then used to extract the x, y and area time series data



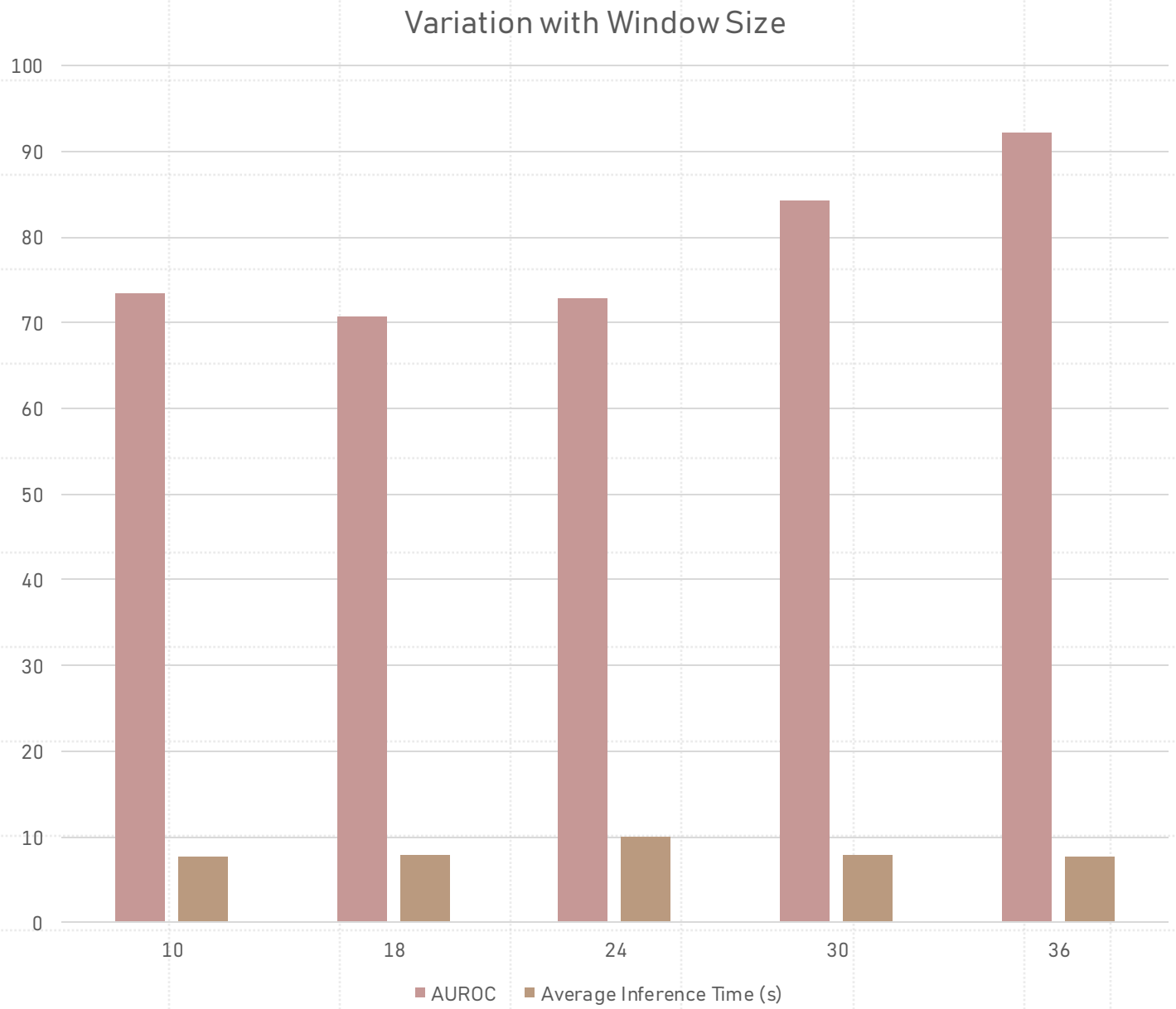
# Variation with Fisher Values for a given Window Size for the custom LeNet5 Model



# Dynamic Windows

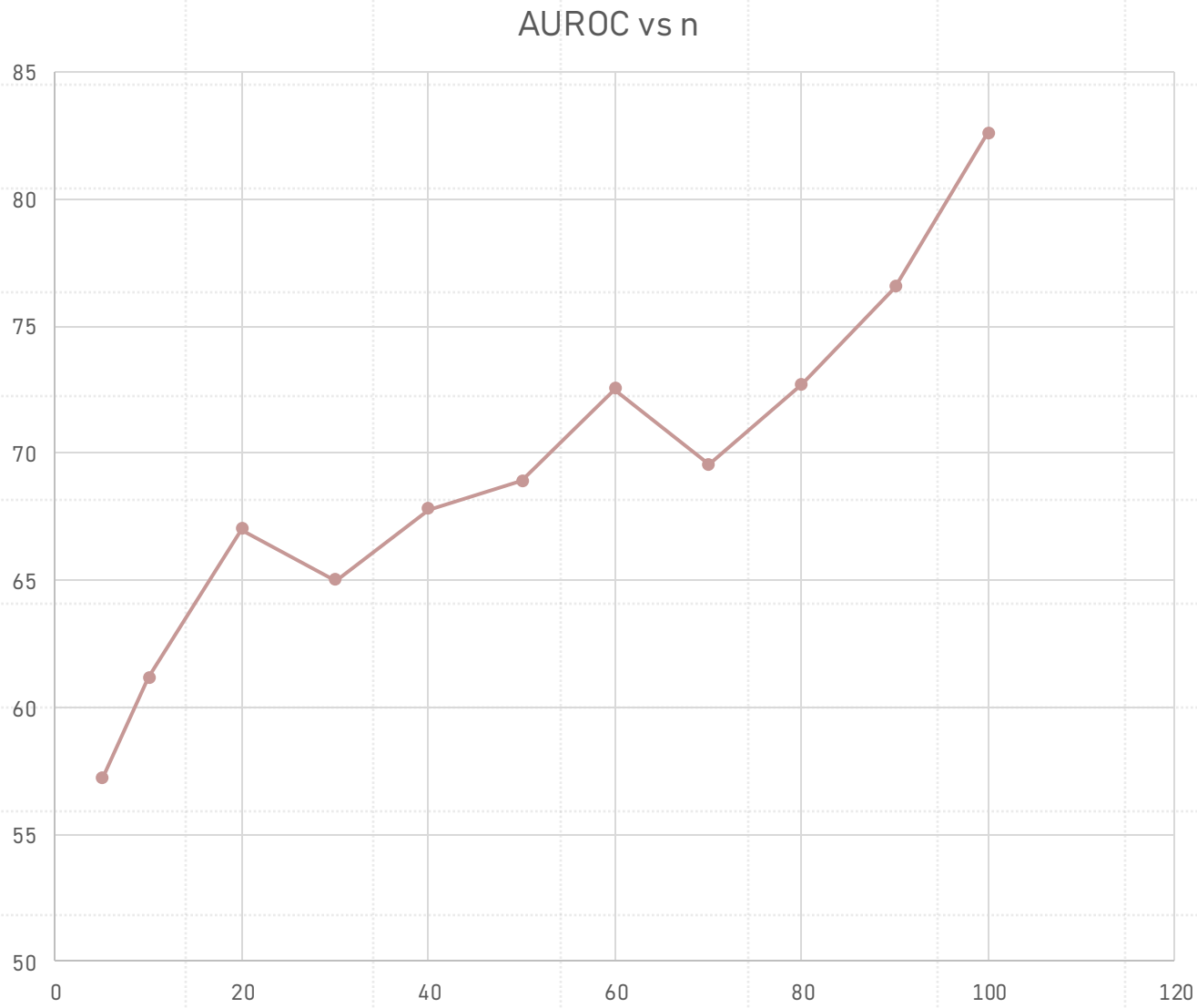
- **Motivation:** Decreasing the window size may improve the CPU Usage and Inference time of the overall pipeline of the decision module.
- We trained the different Lenet models for different window lengths ranging from 10 to 36 size by the method illustrated in the paper.
- Further we plotted the variation of AUROC and Average Inference Time (s) versus the window length for different window lengths.
- We observed that increasing window size usually always improves the output AUROC while the window size did not make any significant difference in inference time.

# Variation with Window Size



# RNN Based Approach

- We used these to verify the claim that the algorithm works on models other than CNNs
- We tried LSTMs, Hybrid models with GRU & CNN, Self-Attention layers.
- While the GRU-CNN gave the highest validation accuracy during training, GRU with a self-attention layer in between gave the highest AUROC score (82.5).



# Merits/ Demerits in the approach

LSTM/GRU capture the temporality of the data continuously rather than quantized window sizes.

We propose using these in conjunction with attention layers to effectively capture temporal data and achieve dynamic window length behaviour, which should optimize performance and inference time significantly

While hybrid models of RNNs and attention layers are known to produce SOTA results on many time series task, it did not give the best results.

The primary reason is, the transformations proposed in the paper, (Erosion, Dilation etc) are convolution-based, on which CNNs perform better than RNNs.

We tried exploring different possible transformations like [shuffling, reversing], but they sub-par results.

# Future Work

Work on improving the LSTM's TNR and AUROC Score to aid in practical applicability.

Use Ensemble-based models to improve post and pre-processing.

Consider usage of other temporal and non-temporal transformations to better learn the equivariance in our particular use case.

# Thank You