Inter-IIT Tech Meet 10.0

Bosch's Model Extraction Attack For Video Classification

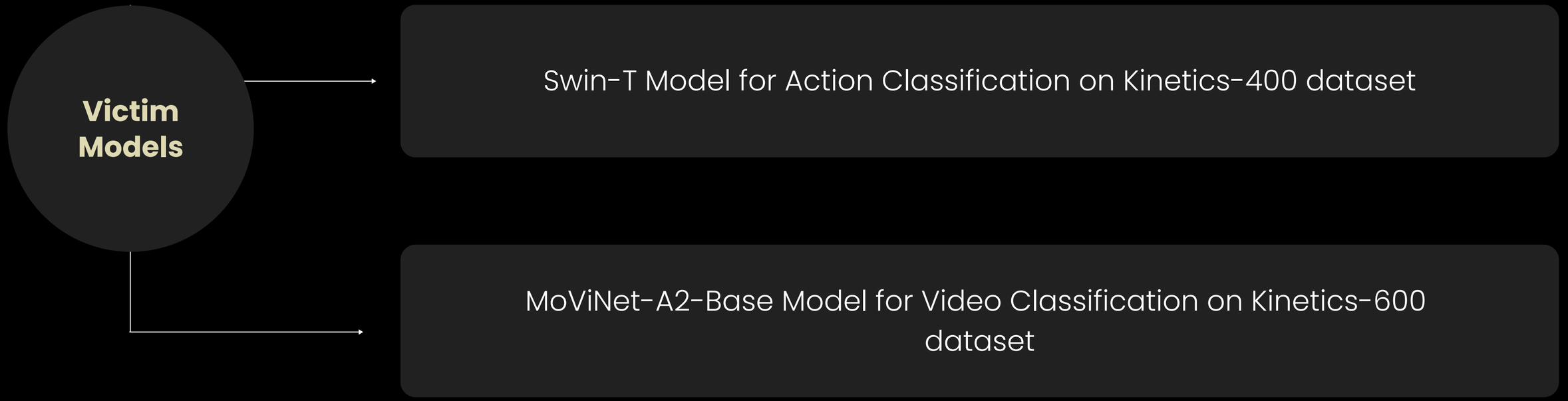
Team 11

Table of Contents

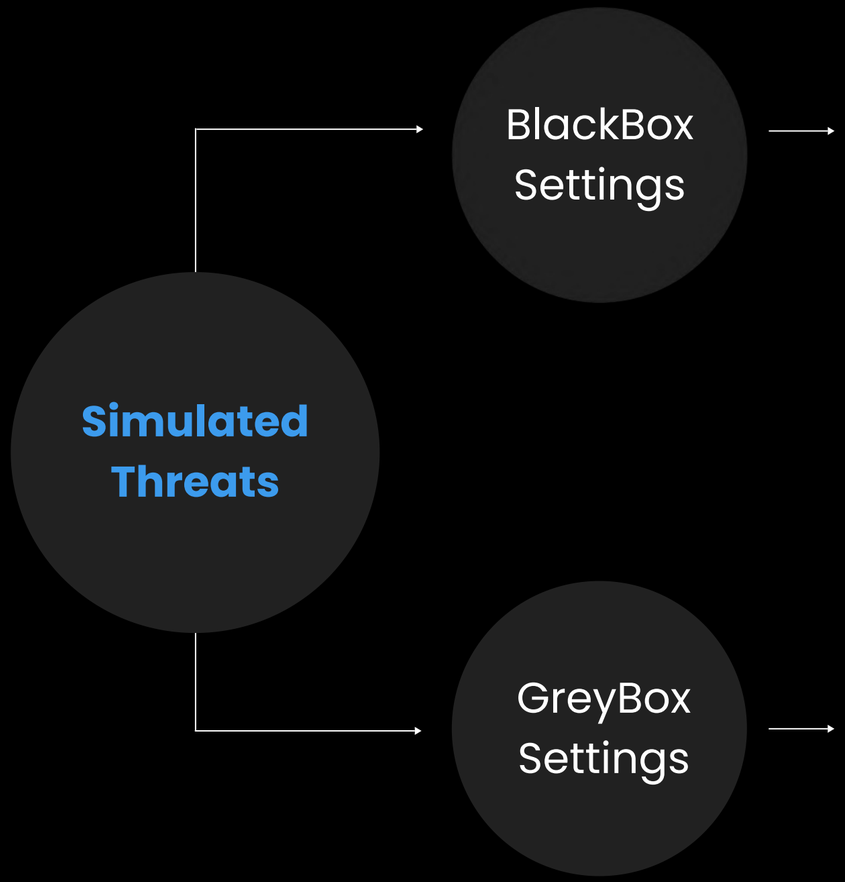
The Problem Statement

Motivation

- Deep learning is intangible for most global companies.
- With such reliance on deep learning, the vulnerability of such techniques must be addressed.
- Hope to help Bosch safeguard their models from malicious threats.



The Problem Settings and Constraints



BlackBox Setting

- Usage of any relevant data set available is not allowed and usage of synthetic or generated data without using the Kinetics series dataset is allowed
- We do not have access to the architecture of the victim model or the data it is trained on.
- We do not have access to the probabilities and class names of the output classes from the victim model.

GreyBox Setting

- Usage of 5% of original data (balanced representation of classes) is allowed for querying in addition to synthetic data.
- We do not have access to the architecture of the victim model and the data it was trained on.
- We do not have access to the probabilities of the output classes and the class names from the victim model.

BlackBox Setting

Roar of the Random!

Workflow

The Journey

1. **Rudimentary Methods**
   Understanding the power of psychedelic animations

2. **Going Open-Source**
   Leveraging the usefulness of open-source images

3. **Balancing the data**
   The Proposed mechanism to balance the data

4. **Results**
   Time for results!

Psychedelic Animations

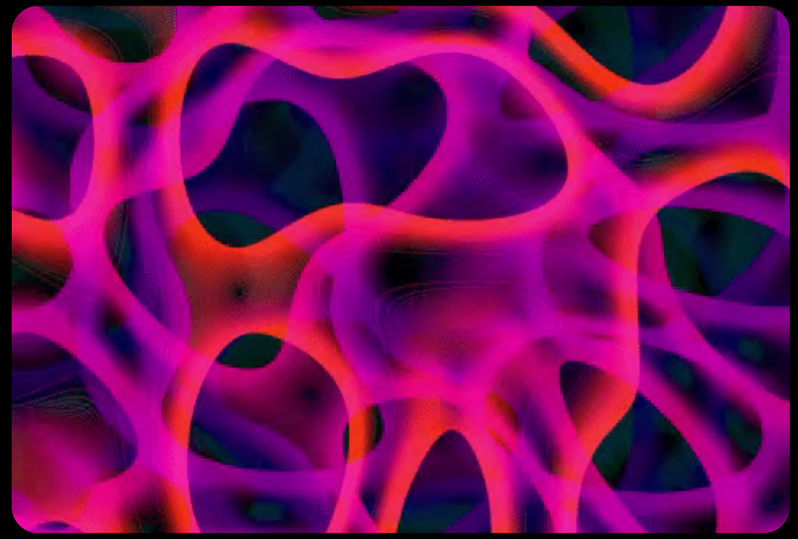
Using Animations as Queries for the Victim model

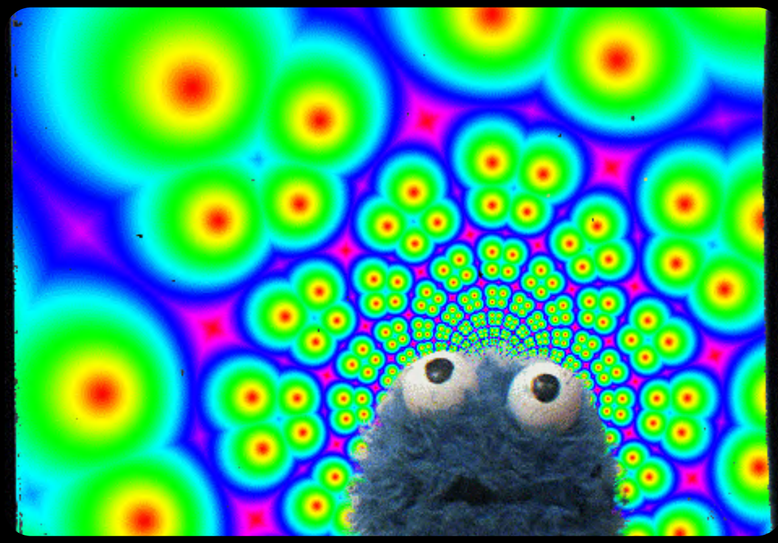
What we did

- Generated custom animations using Python libraries.
- Queried the victim model with animations to create the attack dataset.
- Trained and validated the threat model on the generated dataset.

What we observed

- Real-world action datasets like Kinetics require the knowledge of complex temporal and spatial features.
- The generated animations are not complex enough to capture these relations.





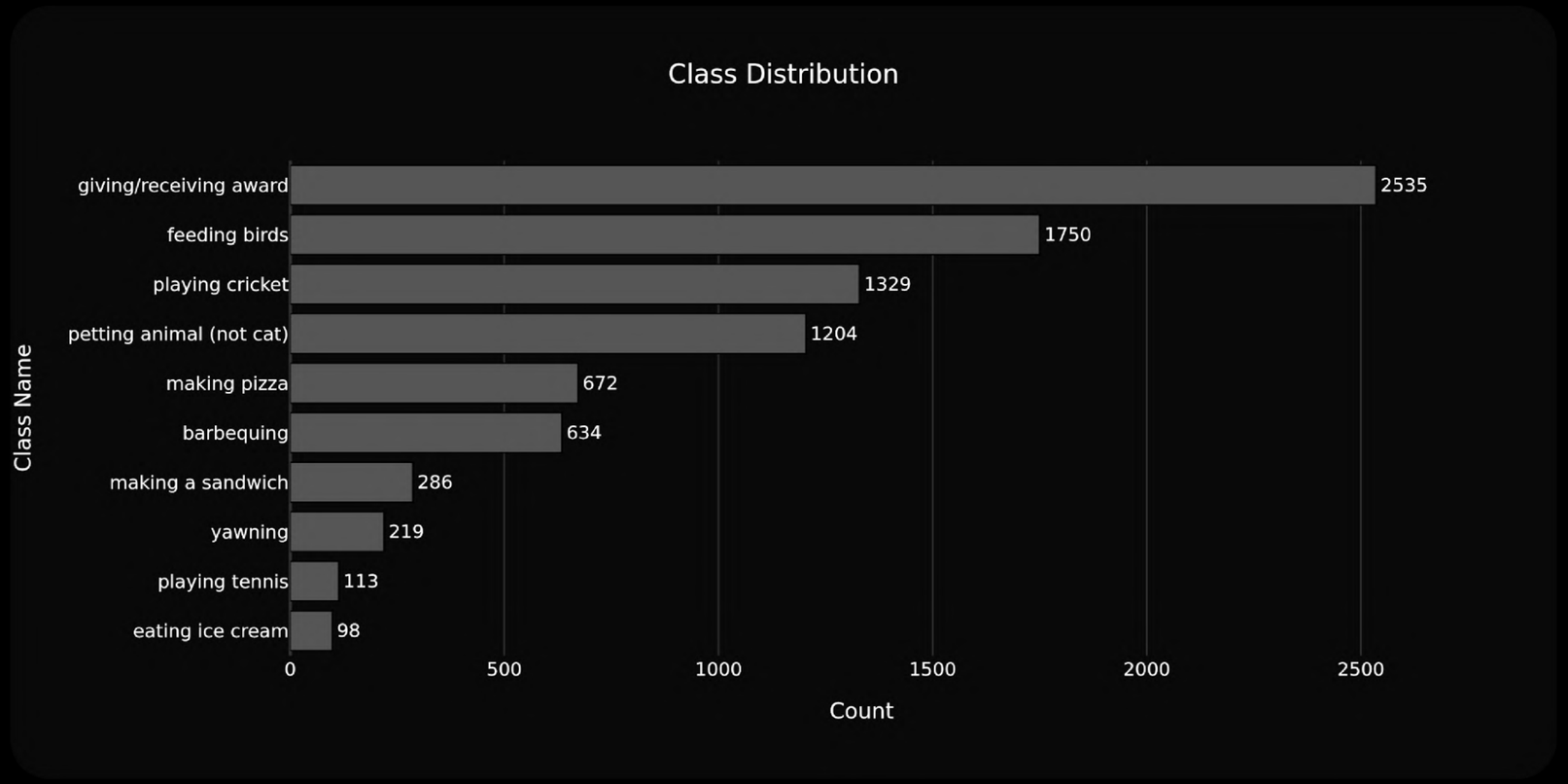
Dataset Generation

We gather image data from various public, OpenSource datasets and stack them to create incoherent videos.

Datasets used:

- Facial expressions
- Indian actors
- Musical instruments
- Human faces dataset.

Obstacle faced



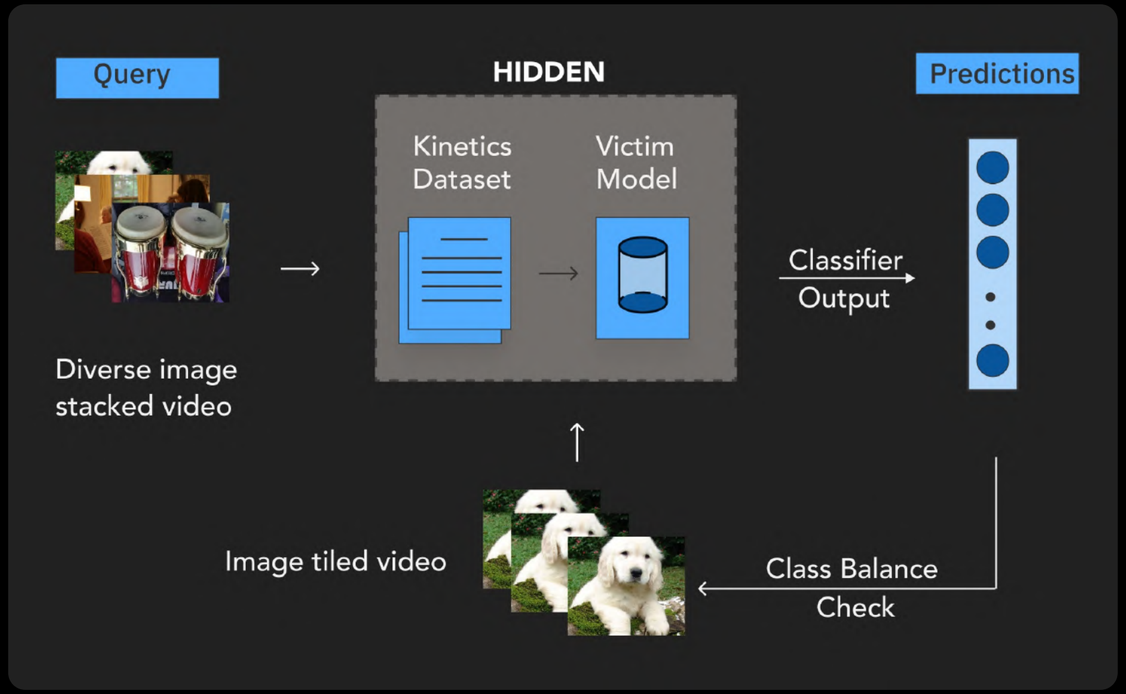
The query-prediction dataset generated has **high class- imbalance.**

How did we solve?

Enhancing class balance through image tile-based video inferencing

What we did

- The image of a specific class is stacked and queried against the victim model.
- Set a heuristic threshold for number of videos per class.
- The data is saved depending on the *videos per class* threshold



EXAMPLES OF
IMAGE-STACKED VIDEOS

The Results

K400		K600
29.17	Score submitted on 19th March	25.36
39.42	Score submitted on 23rd March (Final)	34.51

GreyBox Setting

The original task

Workflow

The Journey

1. Kinetics Dataset

Using 5% Kinetics dataset for querying the victim.

2. Class-wise Error Analysis

Where does our threat model fail to threaten!

3. Results

Time to see Results

Attack Dataset

Using the balanced representation of 5% Kinetics dataset

What we did

- We sample a balanced subset of Kinetics Dataset.
- The sampled dataset is queried against victim model to generate the labels.
- The generated data is used to train the threat model.

Training saturated quickly due to weights initialized from scratch.

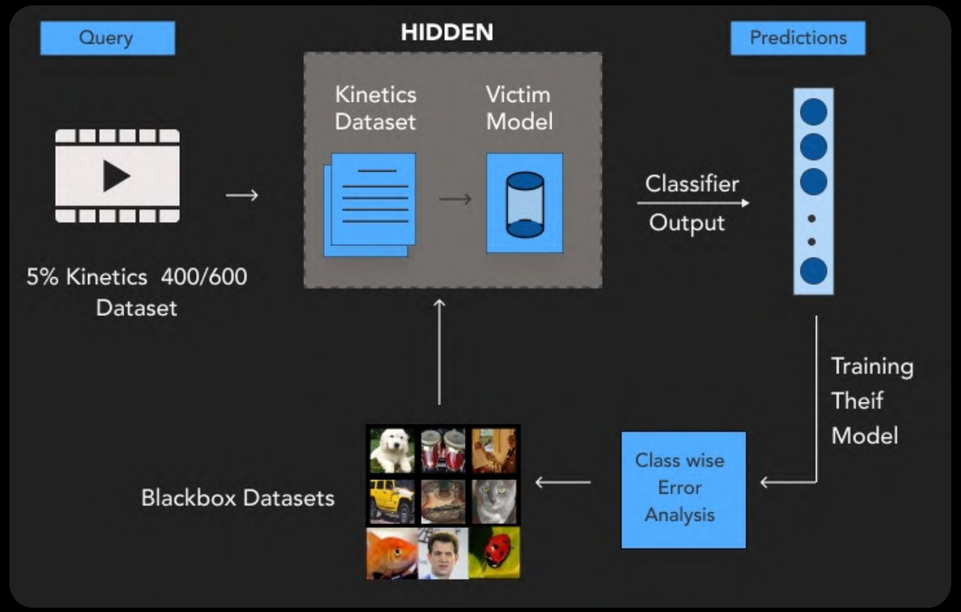
We speculate that using an augmented dataset can improve the results.

How to improve?

Class-wise error analysis

What we did

- After training the threat model we analyze the per class errors.
- The analysis reported high number of erroneous predictions for few classes.
- We augmented these classes with the data generated in the black box section.



Improves the GreyBox accuracy with a minimal number of addition queries

The Results

K400		K600
71.23	Score submitted on 19th March	62.81
77.50	Score submitted on 23rd March (Final)	75.36

Literature Survey

We explore multiple possible threat models to extract the victim model.

Models explored:

- EfficientNetB0+ LSTM
- SlowFast
- Slow - Only (Slow ResNet-50)
- TimeSformer

Final Model chosen:

- Temporal Segment Networks

Explaining the terms

We employed Temporal Segment Networks as our threat model.

Model Architecture -

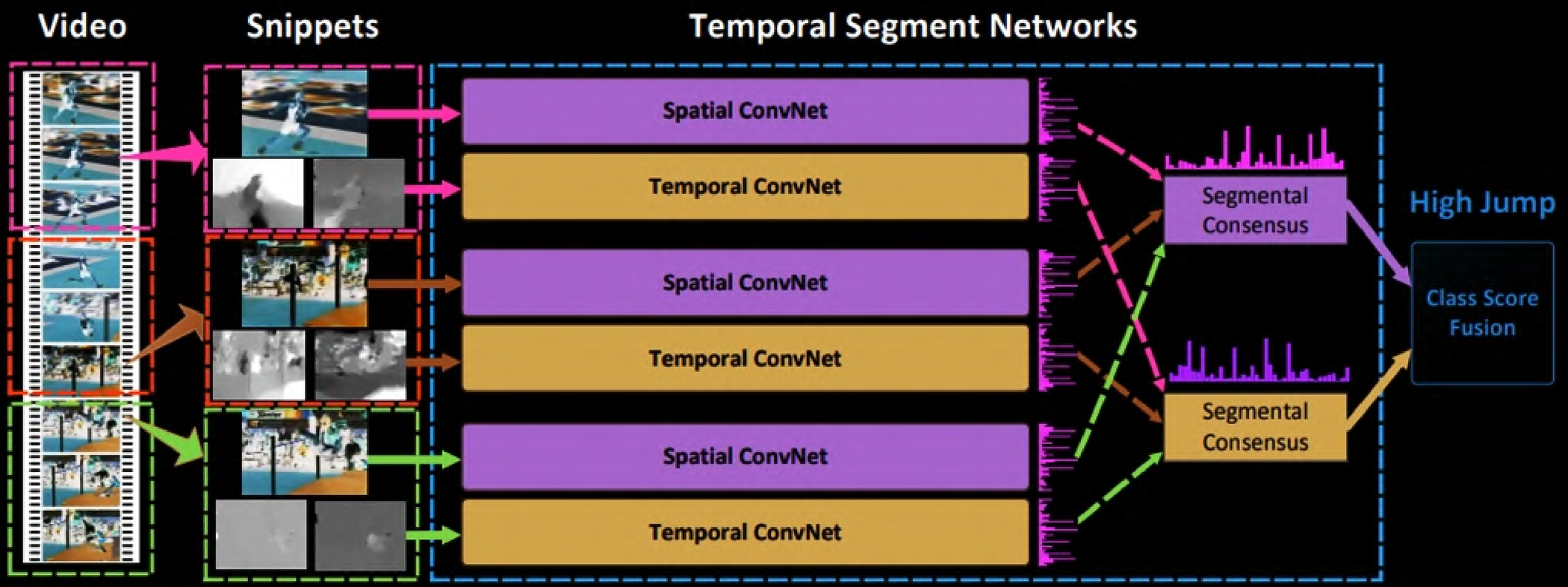
- **EfficientNet:** One of the state-of-the-art models for image classification
- **LSTM (Long Short Term Memory networks):** Special types of RNN, capable of learning long-term dependencies. Work well on a large variety of problems.
- **ResNet:** Uses Residual blocks and skip connections to solve the problem of vanishing gradient in deep CNNs.
- **TimeSformer:** A pure and simple attention-based solution for reaching SOTA on video classification.

Threat Model

We employed Temporal Segment Networks as our threat model.

Model Architecture :

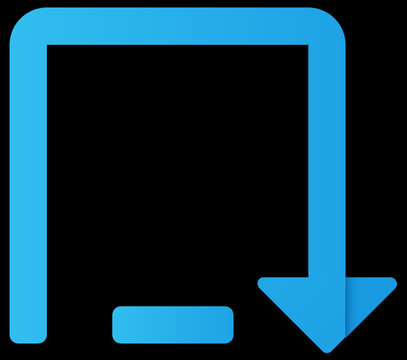
- Composed of spatial stream ConvNets (ResNet) and temporal stream ConvNets (ResNet)
- A sequence of short snippets is sparsely sampled from the entire video rather than working on single frames or frame stacks
- Every individual snippet from this sequence generates its own initial prediction of the action classes.
- The common ground among them will be acquired as the video-level prediction.



Model Architecture

Training Strategy

- We use various augmentations like RandomResized crop, Mixup, and flips to avoid overfitting the model.
- We train the threat model on this augmented dataset using Adam Optimizer with an empirically set learning rate of 7.03e-7.
- We use the StepLR scheduler which decreases the learning rate after every 3000 iterations.
- Max Grad norm of 40 is used to clip the gradient.



Augmentations

To add generalization to the models we augment the dataset with various augmentations

RandomResized Crop
A Crop of frame is taken and the short edge is resized to 256

Video Mixup
The frames of the two videos are superimposed on each other. It's a weighted combination of two different data samples.

Flips
A few randomly selected frames were flipped.

Final Results

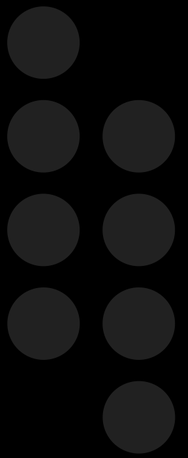
Dataset	Mode	Phase 1 (60 epochs)	Phase 2 (110 epochs)	Benchmark	Queries
K400 (P1)	Black Box Grey Box	29.17 71.23	39.42 77.50	90.8	24320 23925
K600 (P2)	Black Box Grey Box	25.36 62.81	34.51 75.36	89.4	40586 45731

Conclusion And Future Work

A Surprising discovery

Conclusion

Through our work on this problem statement, we present a very direct and dangerous threat to deployed DL models and the attack strategies that could be employed. Even a small subset of training data ridiculously increases the risk of being attacked.



Future Work

Further Architecture Search

Understanding the role that architecture plays in theft of models and utilizing modern ones better

Scaling Laws

Most threat models need not be of the same size as the original model and understanding the parameter count is essential to save compute resources.

Reducing Queries and Active Theft

Active Learning shows great promise in reducing queries for threat models and converges faster

Better Dataset Generation

As they say "Garbage In, Garbage Out"; generating good datasets with equal data of diverse classes is essential. Further work can be done on this.

Utilizing Generative Networks

Training and utilizing generative networks to create better queries would theoretically decrease queries and improve training, though they require time and resources

Greater Hyperparameter Search

Tuning hyperparameters can cause drastic differences in training, especially with constrained time and compute

Thank You!