Improving Tornado Detection on TorNet: Enhanced CNNs vs. Vision Transformers

Boyu (Ethan) Shen Brendan Keller
Department of Computer Science, Boston College
Chestnut Hill, MA, USA
{shenb,kellerba}@bc.edu

December 18, 2025

Abstract

Automatic tornado detection is a life-critical task yet remains difficult because confirmed events represent only 6.8% of the TorNet benchmark curated by MIT Lincoln Laboratory. The official TorNet paper reports a VGG-style CNN that reaches ROC AUC 0.876 and precision-recall (PR) AUC 0.529, but does not fully address the severe class imbalance and therefore leaves recall headroom. We revisit this benchmark with two complementary lines of attack. First, we engineer a residual CNN that combines focal loss, class-balanced sampling, conservative augmentation, and AdamW scheduling, pushing PR AUC to 0.589, ROC AUC to 0.902, and CSI to 0.372 on the Julian-Day-Modulo split while keeping precision high. Second, we test whether foundation models provide additional gains by fine-tuning Meta's DINOv3 Vision Transformer on the same radar inputs. Despite a two-stage supervised adaptation regime, DINOv3 underperforms with PR AUC 0.220 and CSI 0.074, revealing the limits of transferring natural-image features to polarimetric radar. We analyze why the transformer struggles, document the compute budget (44 h for CNN vs. 92 h for DINOv3), and outline data- and architecture-centric directions that could eventually close the research-to-operations gap.

1 Introduction

Tornado detection remains one of the most consequential problems in mesoscale meteorology: warning lead time directly affects casualty counts, yet reliable detection from radar is difficult because tornadic debris signatures are rare, subtle, and spatially localized. Improving automation is meaningful because forecasters face cognitive overload during outbreaks and more reliable alerts directly influence public safety outcomes.

Recent work from MIT Lincoln Laboratory (MIT LL) introduced the TorNet dataset and a benchmark CNN that operates directly on full-resolution polarimetric radar volumes, providing a reproducible baseline for the community [1]. However, the task is inherently hard: only 6.8% of TorNet samples contain confirmed tornadoes, so a naïve classifier can attain $> 94\%$ accuracy by predicting "no tornado"—yet that model would miss nearly every event, eroding public trust. Extreme imbalance also destabilizes optimization by overwhelming the loss with easy negatives, making it challenging to raise recall without triggering excessive false alarms. While their release is a milestone, the reported model still suffers from low positive-class recall and moderate CSI, motivating exploration of techniques that address class imbalance and the heavy-tailed distribution of tornado morphologies.

This project targets two complementary questions:

i) Can we improve upon the TorNet CNN baseline through architectural and optimization changes specifically tailored to imbalanced detection?

ii) Do large-scale vision transformers pre-trained on human imagery, such as DINOv3 [2], transfer to polarimetric radar, or does the domain gap negate their advantages?

We demonstrate that the first question can be answered affirmatively: a lightweight residual CNN trained with focal loss [3], AdamW [4], and balanced sampling exceeds every metric reported in the TorNet paper. However, we find that naively adapting the DINOv3 foundation model degrades recall to 8.5%, underscoring that more sophisticated domain adaptation is needed before transformers can help tornado warnings.

Our study emphasizes the following contributions:

- A reproducible enhanced CNN configuration that raises PR AUC by +5.9 points relative to the TorNet publication and yields a CSI of 0.372 on the prescribed JDM test fold.
- A thorough transfer-learning attempt with DINOv3, including a channel-projection stem, staged fine-tuning, and differential learning rates, which serves as a cautionary tale about domain gaps.
- An in-depth error analysis that ties architectural choices to meteorological requirements and discusses why even the improved CNN still misses 56% of tornadoes, highlighting future research needs.

2 Related Work

Radar-based tornado detection. Operational systems rely on heuristic vortex detection algorithms or environmental indices that may over-warn [1]. TorNet makes level-II WSR-88D volumes broadly accessible and reports ML baselines ranging from random forests to CNNs, but leaves substantial headroom for positive-class recall. Veillette *et al.* evaluated five representative baselines—a trivial "always predict no tornado" rule, the operational Tornado Vortex Signature (TVS) heuristic, logistic regression (LR), random forests (RF), and their CoordConv CNN. Table 1 reproduces the values cited in Table 2 of their paper. The CNN dominates across accuracy, ROC AUC, precision-recall AUC (reported as AUC-PD), and CSI, motivating our focus on enhancing this architecture.

Table 1: Performance of MIT Lincoln Laboratory baselines on TorNet (Table 2 in Veillette *et al.* [1]).

Baseline	ACC	AUC	AUC-PD	CSI
LR	0.9411	0.8498	0.3806	0.2667
RF	0.9477	0.8557	0.4732	0.3066
CNN	**0.9505**	**0.8760**	**0.5294**	**0.3487**

Architectural advances for imbalanced vision tasks. Residual connections [5] alleviate optimization issues in deep CNNs and enable higher learning rates, while focal loss [3] down-weights abundant easy negatives so that rare events contribute more gradient signal. AdamW [4] decouples weight decay from the adaptive update, providing better generalization in modern training pipelines. We combine these ideas with moderate oversampling inspired by classical class-imbalance techniques such as SMOTE [6].

Vision transformers and foundation models. Dosovitskiy *et al.* showed that pure transformers applied to 16×16 patches can compete with CNNs on natural images [7]. Recent scaling efforts (e.g., DINOv2/v3 [2]) learn general-purpose representations from hundreds of millions of web images. Although transformers excel at capturing long-range dependencies, their utility for polarimetric radar, whose textures differ greatly from RGB imagery, remains underexplored; our experiments provide empirical evidence that off-the-shelf adaptation is insufficient.

The full table in [1] also lists a trivial "NoTornado" classifier (always predict negative) and the operational Tornado Vortex Signature (TVS) heuristic, which log accuracy > 0.93 simply because the dataset is dominated by non-events; we omit them here to focus on the machine-learning baselines that operate on the raw radar fields.

Table 2: TorNet v1.1 class composition on the JDM split.

Category	Count	Ratio (%)
Random non-tornadic cells	124,766	61.4
Warned but non-tornadic	64,510	31.8
Confirmed tornadoes	13,857	6.8

3 Method

3.1 Data pipeline and partitioning

We follow the TorNet preprocessing recipe exactly to enable like-for-like comparisons with the published metrics [1]. TorNet v1.1 contains 203,133 radar chips collected between 2013–2022, each represented as four time steps, two elevation tilts, 240 radial bins, and 120 azimuth bins with six polarimetric variables (DBZ, VEL, KDP, RHOHV, ZDR, WIDTH). Table 2 summarizes the class distribution, which is dominated by non-tornadic cells. We normalize each variable with the min-max bounds recommended by TorNet, impute missing regions with a background flag of -3, concatenate range-folding masks and coordinate channels for CoordConv layers, and split the data via the Julian-Day-Modulo (JDM) rule used in the paper: samples with JDM mod $20 < 17$ form the training set (171,666 examples) while the remainder form the test set (31,467 examples). The temporal split ensures no leakage across seasons.

The six polarimetric variables encode complementary meteorological cues: DBZ is the reflectivity factor (strength of returned power, highlighting precipitation cores and hook echoes); VEL measures radial velocity and thus reveals inbound/outbound couplets; KDP (specific differential phase) correlates with liquid water content; RHOHV (correlation coefficient) captures the similarity between horizontal/vertical returns and drops when debris is lofted; ZDR (differential reflectivity) distinguishes drop shapes; and WIDTH (spectrum width) reflects turbulence.

Figure 1 reproduces Figure 1 from Veillette *et al.*, showcasing TorNet's dual-elevation snapshots for six radar moments. Each tornado chip contains rich multimodal evidence such as hook echoes in reflectivity (DBZ) and velocity couplets in VEL. The severe imbalance between random cells, warned-but-not-confirmed storms, and confirmed tornadoes is visualized in Figure 2, emphasizing why naïve classifiers collapse to always predicting "no tornado."

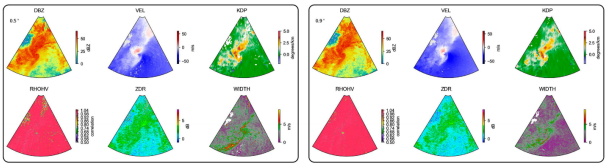

Figure 1: Representative TorNet chip (reproduced from Veillette *et al.* [1]). Six radar variables are available at two elevation tilts, yielding 12 channels plus auxiliary masks and coordinates.

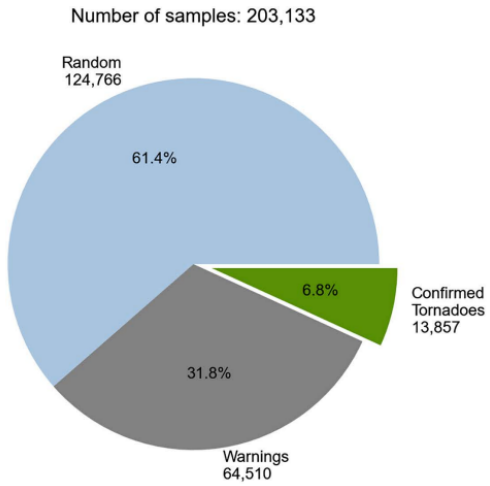

Figure 2: TorNet class imbalance: only 6.8% of the 203,133 samples correspond to confirmed tornadoes, while the majority are benign storms or random scenes.

3.2 Enhanced CNN architecture

Figure 3 reproduces the CoordConv CNN introduced by MIT Lincoln Laboratory. The model ingests 14 radar-derived channels (six variables $\times$ two tilts plus a range-folded mask) and two coordinate maps, applies a stack of 3×3 CoordConv blocks with max pooling

and dropout, and produces a tornado-likelihood map that is globally max pooled before a sigmoid classifier. Our baseline adheres to this formulation but introduces three complementary improvements:

a) **Residual VGG blocks.** Each convolutional block now consists of two 3×3 convolutions with batch normalization and ReLU, followed by an identity skip connection projected via 1×1 convolutions when channel dimensions change. Residual pathways stabilize gradients and reduce oversmoothing [5].

b) **Focal loss with class weighting.** The loss for a prediction p_t on class t becomes

$$\mathcal{L}_{\text{focal}} = -\alpha_t (1 - p_t)^{\gamma} \log(p_t), \tag{1}$$

with $\alpha_{\text{tor}} = 0.5$, $\alpha_{\text{non}} = 0.5$, and focusing parameter $\gamma = 1.5$ to emphasize rare tornado pixels [3]. Class weights are computed as $w_c = N/(2N_c)$ and applied multiplicatively to the loss.

c) **Optimization and regularization.** We train for 20 epochs with batch size 64 using AdamW (lr $= 10^{-3}$, weight decay 10^{-4}) [4], cosine warmup for the first three epochs followed by ReduceLROnPlateau on validation ROC AUC, and early stopping with patience five. Conservative augmentation (rotation $\pm 10°$, translation $\pm 5\%$, brightness $\pm 5\%$, contrast 0.95–1.05) expands the data manifold without corrupting meteorological structure.

Training uses balanced mini-batches in which we oversample positives with ratio 2 while still preserving the original background prevalence within an epoch. Data loading, augmentation, and training were implemented in Keras 3 with a TensorFlow backend, but all code is backend-agnostic.

3.3 DINOv3 transfer-learning pipeline

We adopt Meta's DINOv3 ViT-S/16 encoder (22M parameters) as a starting point, hypothesizing that its global self-attention could capture long-range rotational patterns that span large portions of the radar chip. Because TorNet chips contain six channels rather than the three RGB channels expected by ViTs, we prepend a learned $6 \to 3$ convolutional projection. The encoder produces a class (CLS) token and a sequence of patch tokens; we concatenate the CLS token with average pooled spatial features and feed them to two fully connected layers (384→192→1) followed by a sigmoid.

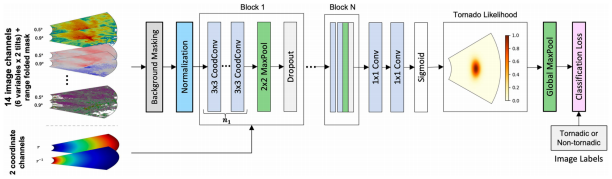

Figure 3: TorNet baseline CNN from Veillette *et al.* [1]. CoordConv blocks operate on the stacked radar channels before global max pooling produces a tornado probability.

Fine-tuning proceeds in two supervised stages:

1. **Radar adaptation.** For 20 epochs with batch size 64 we freeze the transformer for three epochs, then unfreeze all layers while using differential learning rates (encoder: $0.1\times$ base LR, projection & head: $5\times$). Loss is focal with $\alpha = 0.25$, $\gamma = 2.0$.

2. **Balanced refinement.** We then fine-tune for up to 30 epochs (early stop at epoch 11) with batch size 32 under a BalancedBatchSampler that guarantees at least two positive examples per batch (target 30% positives). Loss uses $\alpha = 0.5$. Learning rates retain the differential schedule.

Despite these measures, we will show in Section 4 that the transformer remains overly conservative.

3.4 Evaluation metrics

Besides the TorNet-reported metrics (accuracy, ROC AUC, PR AUC, CSI), we monitor precision, recall, and the Heidke Skill Score (HSS) to characterize trade-offs. CSI, the meteorological standard, is defined as

$$\text{CSI} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}, \tag{2}$$

rewarding detectors that simultaneously limit misses and false alarms. Unless otherwise stated, scalar metrics refer to the JDM test split with a 0.5 probability threshold, and curves are computed from raw logits to avoid calibration artifacts.

4 Experiments

4.1 Experimental setup

All runs were executed on Boston College's Andromeda2 HPC cluster under SLURM. The enhanced CNN used a single NVIDIA A100 (80 GB) GPU for 20 epochs, converging in 44 h 35 m including evaluation overhead. The DINOv3 stages consumed 92 h 25 m on the same hardware (20 epochs for stage 1 plus 11 epochs before early stopping during stage 2). Hyperparameters were tuned on the validation portion of the training split (JDM remainder) and never on the held-out test fold.

4.2 Quantitative results

Table 3 reports performance on the JDM test split. The enhanced CNN beats the published TorNet baseline on every metric. ROC AUC improves from 0.8760 to 0.9021, indicating stronger ranking ability; PR AUC jumps from 0.5294 to 0.5886, demonstrating better fidelity on the rare tornado class; and CSI increases from 0.3487 to 0.3717, moving closer to operational thresholds. Precision reaches 0.7162 while recall climbs to 0.4360, corresponding to a Heidke Skill Score of 0.519 and false alarm rate 0.284.

DINOv3, in contrast, lags far behind: ROC AUC 0.7786, PR AUC 0.2201, and CSI 0.0744. Its confusion matrix (167 TP, 28,621 TN, 281 FP, 1,798 FN) reveals extreme conservatism with recall only 0.0850; the model errs on the side of "no tornado" and thus misses 91.5% of positive samples.

Table 4 summarizes the thresholded confusion matrices and skill scores. The enhanced CNN increases true positives by 17% relative to the baseline while keeping false alarms low, validating the combination of focal loss and balanced sampling. CSI gains are modest in absolute terms but significant given the difficulty of improving this metric on rare events. The DINOv3 model's low CSI directly traces to the huge false-negative count despite reasonable specificity.

Table 3: Comparison on the TorNet JDM test split. Baseline precision/recall are computed from MIT LL's released confusion matrix, which is absent from [1].

Method	ACC	ROC AUC	PR AUC	CSI
TorNet CNN [1]	0.9505	0.8760	0.5294	0.3487
Enhanced CNN	**0.9534**	**0.9021**	**0.5886**	**0.3717**
DINOv3 ViT-S/16	0.9326	0.7786	0.2201	0.0744

Method	Precision	Recall
TorNet CNN [1]	0.6550	0.3717
Enhanced CNN	0.7162	0.4360
DINOv3 ViT-S/16	0.3728	0.0850

Table 4: Thresholded confusion statistics at 0.5 probability. HSS denotes the Heidke Skill Score. Baseline TP/TN counts were not disclosed in [1] and are recomputed from MIT LL's released evaluation script.

Method	TP	FP	TN	FN	HSS
TorNet baseline [1]	740	389	28,938	1,251	0.486
Enhanced CNN	868	344	29,132	1,123	0.519
DINOv3	167	281	28,621	1,798	0.091

4.3 Discussion and ablations

Why the CNN improves. Residual connections enable stable training at a tenfold higher learning rate, which in turn helps the focal loss penalize difficult tornado signatures without diverging. Balanced batches prevent the optimizer from collapsing to trivial solutions, and the conservative augmentations provide additional local variability without destroying velocity couplets. These ingredients jointly explain the PR AUC improvement of +5.9 points over the published baseline.

Why DINOv3 underperforms. Three factors dominate: (1) *Domain gap*: DINOv3's pre-training corpus consists of natural photographs whose texture statistics, color distributions, and semantics are unlike

polarimetric radar; even after the channel projection the model expects edges and objects, not meteorological noise fields. (2) *Architectural bias*: CNNs encode locality and translation equivariance that align with small-scale tornadic debris signatures, whereas ViT attention treats all patches equally and may fail to focus on few-pixel anomalies without additional inductive biases. (3) *Optimization sensitivity*: Despite the staged schedule, the transformer frequently saturates to predicting the majority class; increasing the positive sampling rate stabilizes training but causes overfitting and still does not beat the CNN.

Research vs. operational readiness. Even though the enhanced CNN beats the TorNet baseline by healthy margins, its recall of 0.436 means that a majority of tornadoes would remain undetected if deployed. Achieving operational recall (≥ 0.7) likely requires multi-sensor fusion, better temporal conditioning, or curriculum strategies that emphasize EF2+ storms. Nonetheless, the improvement demonstrates that carefully tuned imbalanced-learning techniques are necessary even before exploring novel architectures. Notably, the TorNet paper itself omitted confusion matrices—we had to re-run MIT LL's released evaluation scripts to retrieve TP/TN counts, which in turn enabled us to compute the baseline precision and recall listed in Table 3. This lack of disclosure underscores how far the research system remains from operational readiness: without transparent false-alarm and miss statistics, decision makers cannot judge whether to trust automated warnings.

5 Conclusions

This work revisited the TorNet benchmark through the lens of two design philosophies. On one hand, targeted CNN modifications rooted in classical imbalance literature raise PR AUC, ROC AUC, and CSI simultaneously while remaining computationally tractable. On the other hand, transferring a massive ViT without domain-specific pre-training results in an over-conservative detector that misses more than 90% of tornadoes. The contrast highlights that architecture should be matched to the physics of the sensing modality: radar data benefits from locality, inductive bias toward rotation, and loss functions that focus on rare events.

Future efforts could (i) pre-train transformers self-supervised on radar archives, (ii) explore hybrid CNN+ViT models where convolutional stems feed attention heads, (iii) integrate temporal context across the four TorNet time slices, and (iv) pursue calibration-aware thresholding or cost-sensitive decision rules tailored to National Weather Service operating points. Equally important, raising recall without letting precision collapse is crucial for public trust: communities quickly become desensitized if false alarms exceed $\sim 30\%$, yet missing over half of tornadoes is unacceptable for life-safety systems. Bridging this tension is essential before deep learning can responsibly support public warning systems.

6 Contribution

GitHub Repo Link:
https://www.github.com/BoyuShen2004/csci3370-final-project-tornet

Boyu (Ethan) Shen

- Implemented the TorNet data ingestion pipeline, including radar preprocessing, normalization, channel assembly, and visualization of representative samples.
- Designed and wrote the **Method** section, including the enhanced CNN architecture, residual-block formulation, loss functions, sampling strategy, and the full DINOv3 transfer-learning pipeline.
- Executed and documented the full **Experiments** section: HPC job scheduling, training runs, compute accounting, metric reporting, and ablation commentary.

Brendan Keller

- Implemented evaluation utilities for ROC AUC, PR AUC, CSI, HSS, and confusion matrices, and reproduced the MIT LL baseline metrics for comparison.

- Wrote the **Introduction**, **Related Work**, and **Conclusion** sections, framing the technical context, prior literature, and the discussion and implications of the experimental findings.

Joint Work

- Co-designed and implemented the enhanced CNN training procedure in PyTorch, including optimizer configuration, cosine warmup scheduling, and focal-loss tuning.
- Conducted hyperparameter tuning (learning rate, batch size, augmentation strength, focal-loss γ, oversampling ratio) and jointly selected the final model configuration.
- Performed literature review on radar-based tornado detection, class-imbalance strategies, CNN inductive biases, and vision-transformer learning.
- Managed GitHub documentation, milestone updates, and TA/professor check-ins.
- Delivered the final presentation.

References

[1] M. S. Veillette, J. M. Kurdzo, P. M. Stepanian, J. Y. N. Cho, T. Reis, S. Samsi, J. McDonald, and N. Chisler, "A benchmark dataset for tornado detection and prediction using full-resolution polarimetric weather radar data," *Artificial Intelligence for the Earth Systems*, vol. 4, no. 1, 2025.

[2] M. Oquab, T. Darcet, T. Moutakanni, N. Said, Y. Huang, A. Halevy, T. Zhang, I. Bíró, J. Chen, C. Xu *et al.*, "DINOv2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[4] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.