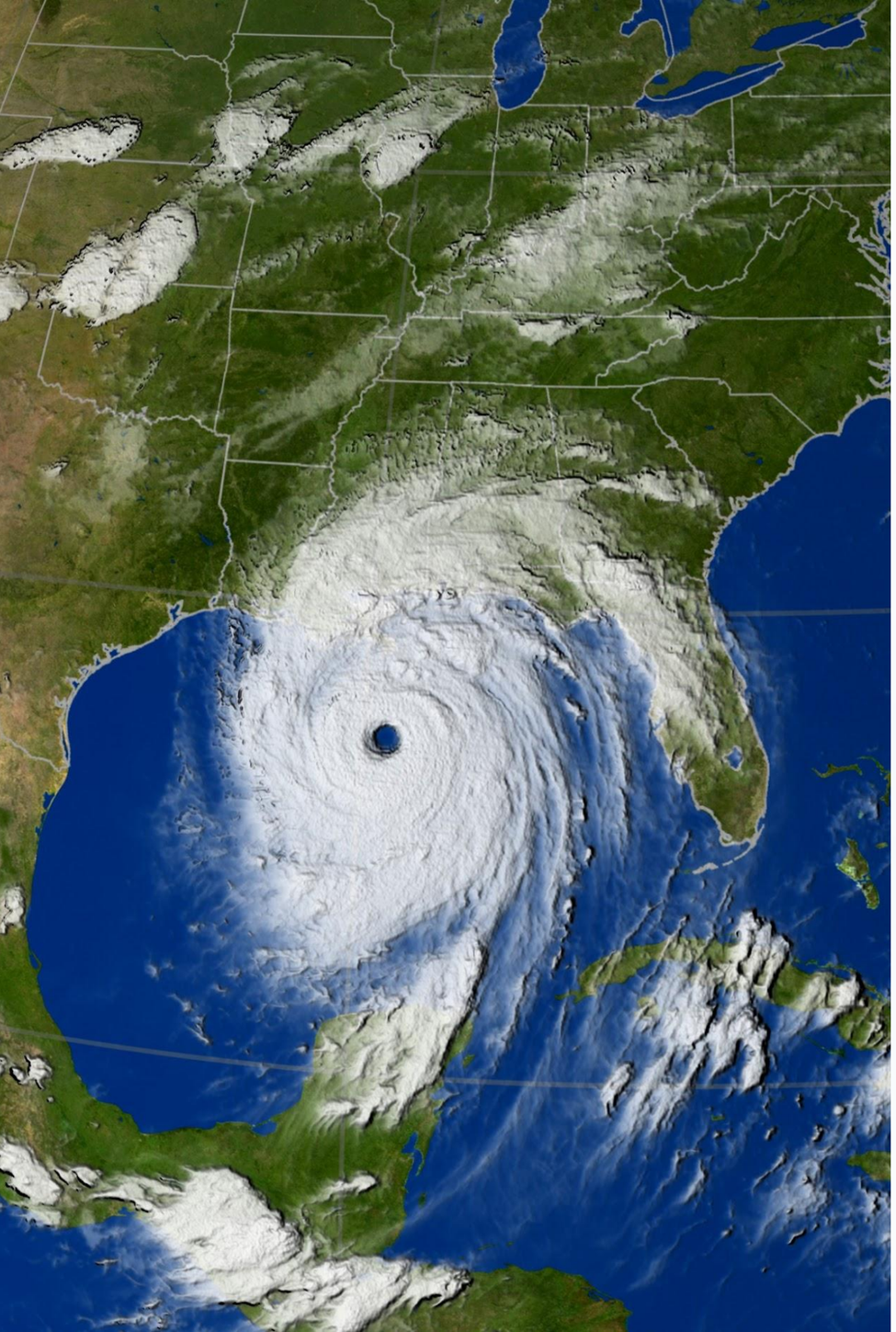


Extreme Weather Image Classification

Group 21 (The Standard Deviants) - Aryan Thodupunuri, Athena Vo

DS 4002, Section 2

December 3, 2025

Outline

- Problem & Motivation
- Data Overview
- Preprocessing
- Model Approach
- Results
- Next Steps

Problem & Motivation:

Automating Extreme Weather Detection

Motivation

- Enabling faster triage and monitoring in critical situations.
- Providing timely alerts for emergency services and infrastructure management.
- Reducing manual effort and potential for human error in identification.

Research Question

Can Convolutional Neural Networks (CNNs) accurately distinguish severe weather from ordinary weather using satellite imagery?

Hypothesis

Extreme weather should have stronger visual patterns than normal conditions, so transfer-learning models will outperform a baseline CNN and exceed 90% accuracy / 0.88 F1-score.

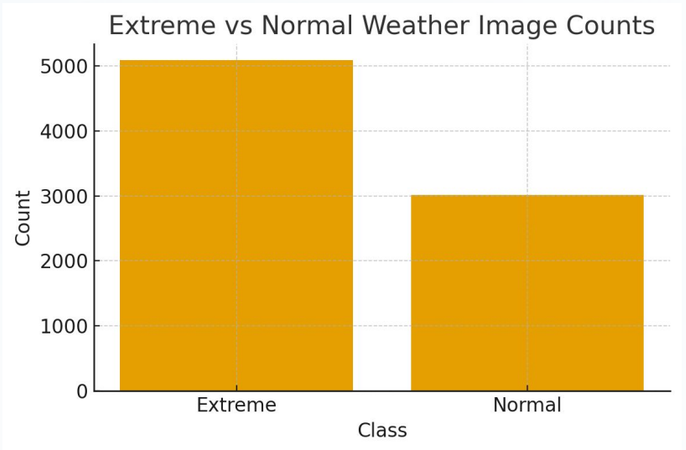
Dataset Overview

Key Dataset Details

- **Source:** Harvard Dataverse
- **Authors:** Ye Li and Mostafa Momen
- **Original Size:** 9,081 satellite images
- **Cleaned Size:** 8,099 satellite images (used for training)
- **Format:** JPEG image files
- **Coverage:** Global satellite imagery of diverse weather events
- **License:** CC0 1.0
- **Categorization:**
  - Original 5 categories consolidated into 2 classes for this study
  - **Extreme:** Dust Storms, Tropical Cyclones, Wildfires
  - **Normal:** Convective Cell Clouds, Convective Roll Clouds

Class Distribution

Extreme Weather Images	5,085
Normal Weather Images	3,014
Total Images	8,099



Bar plot of class distribution from `counts_cleaned.csv`.

Preprocessing Pipeline

1

Category Consolidation: 5 Classes → 2 Classes

Consolidated 5 original weather categories into 2 classes: **Extreme** (Dust Storms, Tropical Cyclones, Wildfires) and **Normal** (Convective Cell Clouds, Convective Roll Clouds).

2

Deduplication and Standardization

Removed duplicates and standardized all images to 224×224 pixels with pixel values normalized to [0, 1].

3

Stratified Train-Validation-Test Split

Applied 80/10/10 train-validation-test split while preserving class balance across all partitions.

4

Stratified Data Split

8,099 images ready for model training and evaluation.

Analysis Plan & Justification

Baseline CNN: A Strong Foundation

Designed a simple 3-block CNN to set an initial benchmark for the classification task. This model helped us understand which visual features were learnable without transfer learning.

MobileNetV2: Boosting Performance

Introduced a pre-trained MobileNetV2 to capture richer texture and structure in weather imagery. This model served as our first step toward more advanced feature representations.

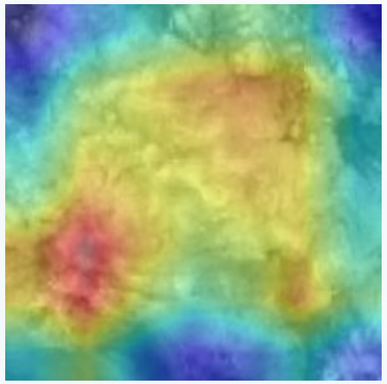
EfficientNetV2: Peak Performance

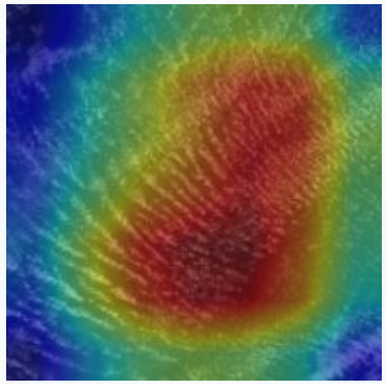
Explored EfficientNetV2-S to evaluate whether a newer, optimized architecture could better model subtle weather patterns. This approach allowed us to test scalability and architectural efficiency.

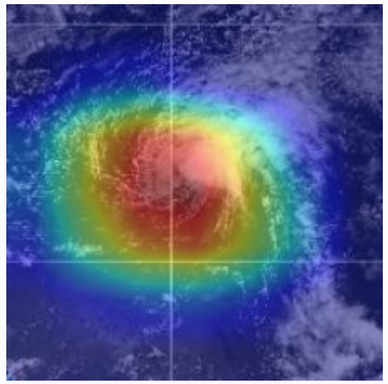
Our journey from a foundational CNN to advanced transfer learning with EfficientNetV2 demonstrates a systematic approach to achieving peak performance in complex weather classification tasks. Each iteration built upon the last, progressively refining our model's ability to accurately identify extreme and normal weather patterns.

Grad-CAM Explainability: Unveiling Model Decisions

To build trust and provide crucial insights into our MobileNetV2 model's decision-making process, we employed Grad-CAM (Gradient-weighted Class Activation Mapping). This technique generates heat maps that highlight the regions in an image that were most influential in the model's prediction.







Visualizing Evidence: Grad-CAM overlays on the last convolutional layer effectively visualize the areas the model "attends" to when making its classification.

Validation and Trust: These visualizations help us validate that the model is focusing on relevant meteorological structures and features (e.g., swirling clouds in a hurricane, heavy snowfall, clear skies) rather than spurious background elements. This interpretability is vital for ensuring the model's reliability and building confidence in its predictions.

Tricky Analysis Decisions

Crucial analytical choices significantly impact model performance and interpretation. Understanding these decisions is key to developing robust and reliable classification systems.

1

Class Collapsing (5 to 2 Classes)

The Challenge: Grouping 5 original weather classes into 2 ("Extreme" vs. "Normal") required careful consideration. While categories like dust storms are clearly "extreme," borderline phenomena like convective cells posed a dilemma.

Impact: This labeling choice directly influenced model difficulty and the definition of "extreme" weather.

2

Image Resizing (to 224x224)

The Challenge: Resizing images to 224×224 pixels risked losing critical meteorological details. Tropical cyclones could lose fine spiral structures, and roll clouds might disappear entirely.

Impact: A trade-off between preserving resolution (for feature fidelity) and model efficiency.

3

Dataset Deduplication

The Challenge: Our dataset contained many implicit duplicates. Removing too many could eliminate valid unique samples, while keeping too many risked data leakage and inflated accuracy.

Impact: We used filename-based deduplication for speed, knowing content-hashing would be more accurate but too heavy for our timeline.

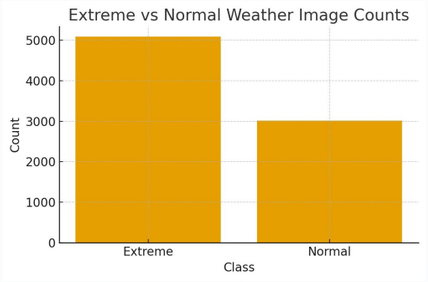
Bias and Uncertainty Validation

Class Distribution Bias

Issue: Imbalanced distribution of extreme vs. normal images.

Risk: Model might over-predict the majority class.

Correction: Used stratified splitting (80/10/10) to maintain proportional representation across train, validation, and test sets.

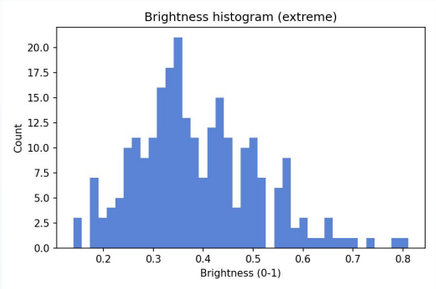


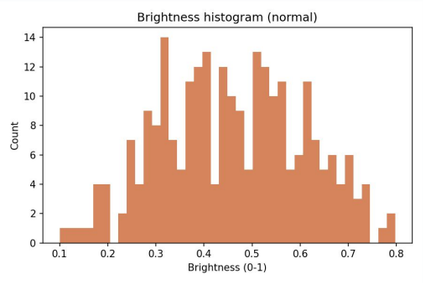
Acquisition Bias (Visual Differences)

Issue: Extreme events often have higher visual contrast/texture.

Risk: Model might learn contrast instead of true meteorological patterns.

Correction: Normalized images for consistent brightness and cross-validated with multiple architectures.





Source Bias (Satellite Archive)

Issue: All images originated from a single satellite archive.

Risk: Model might not generalize to other satellite data or sensors.

Correction: Identified as a limitation and used augmentations to enhance visual diversity.

Addressing potential biases and validating uncertainty are crucial steps to building a reliable and generalizable weather classification model. Our rigorous approach ensures the model's performance is not artificially inflated but reflective of its true predictive power across diverse scenarios.

Results and Conclusions

Conclusion: ✔ HYPOTHESIS CONFIRMED

All three models surpassed the performance targets, with transfer-learning architectures reaching 99% accuracy.

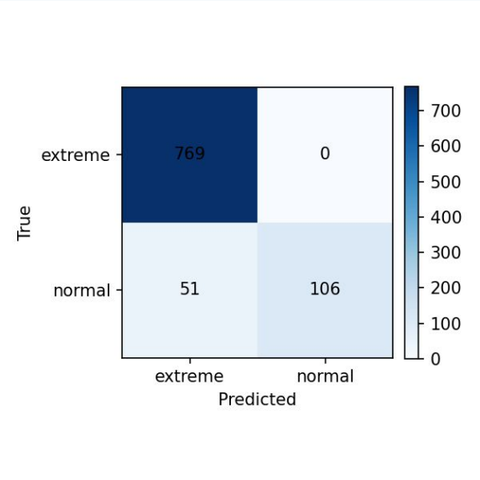
96.8%

99.0%

99.0%

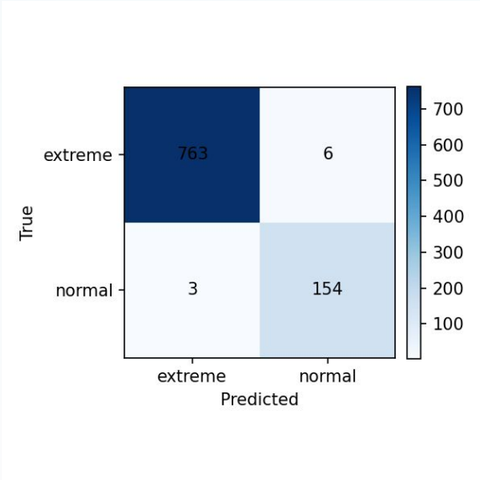
Baseline CNN

Our initial model demonstrated strong foundational performance with an F1-score of 0.98 for 'extreme' and 0.90 for 'normal', proving the viability of the task.



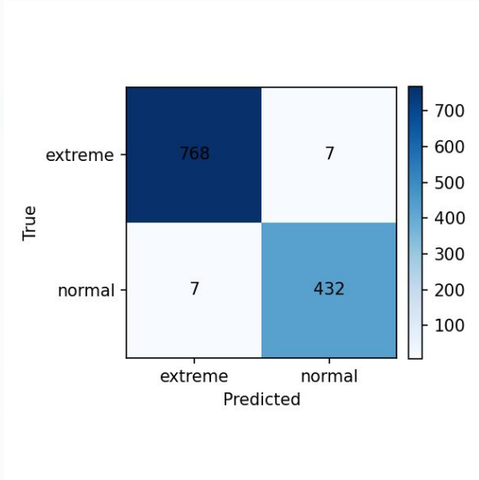
MobileNetV2 Transfer Learning

Leveraging pre-trained weights, MobileNetV2 significantly boosted performance, achieving a macro F1-score of 0.98, with improved recall for the 'normal' class.



EfficientNetV2-S Transfer Learning

The state-of-the-art EfficientNetV2-S model pushed performance to its peak, achieving a macro F1-score of 0.99 and near-perfect precision and recall across both classes.



Detailed classification reports available in `reports/training/`.

Next Steps

Our project is a continuous journey of improvement. The next phase focuses on enhancing data integrity, refining model robustness, and preparing for real-world deployment.

Filename-to-Hash Mapping

Implement a system to add a filename → hash mapping file during the preprocessing stage. This will create a verifiable record for each image and improve data provenance.

Content-Hash Duplicate Detection

Integrate content-hash based duplicate detection to more accurately identify and remove redundant data, ensuring a cleaner and more effective training set.

Threshold Calibration & Monitoring

Rigorously calibrate model thresholds for optimal performance. Establish drift monitoring and data quality assurance checks to maintain model effectiveness in production.

Deployment Benchmarking

Benchmark the latency and size of EfficientNetV2 against MobileNetV2 to determine the most suitable architecture for efficient and scalable deployment.

These strategic next steps are designed to propel our model from an experimental prototype to a production-ready solution, capable of robustly identifying extreme weather events.

References & Resources

References:

[1] Li, Ye, and Mostafa Momen. 2024. "9081 Images Dataset for: Detection of Weather Events in Optical Satellite Data Using Deep Convolutional Neural Networks." Harvard Dataverse. https://doi.org/10.7910/DVN/PUIHVC

[2] Li, Ye, and Mostafa Momen. 2021. "Detection of Weather Events in Optical Satellite Data Using Deep Convolutional Neural Networks." Remote Sensing Letters 12 (12): 1227–37. https://doi.org/10.1080/2150704X.2021.1978581

Resources:

- GitHub Repository: https://github.com/AryanThodupunuri/extreme-weather-classification
- PyTorch Documentation: https://pytorch.org/
- Grad-CAM: https://github.com/jacobgil/pytorch-grad-cam

Acknowledgements:

Special thanks to Professor Loreto Alonzi and the DS 4002 course staff for guidance and feedback throughout this project.

Thank you! Any questions?