CHARLOTTE
THE GRADUATE SCHOOL

Final Project Presentation

Incremental Deep Learning for Fire, Smoke and Haze Detection

Sravani Kuncham & Farah Naz

04/29/2025

UNIVERSITY OF NORTH CAROLINA CHARLOTTE | THE GRADUATE SCHOOL

Problem Statement and Solution

Problem Statement:

- In real-world environmental monitoring (e.g., forest fires), **new threats** like haze may emerge **after** a model has already been trained.

- Retraining on all original + new data is often **impractical** due to:
  - Lack of access to old data
  - High compute requirements

Solution:

- The goal is to add new classes (like **Haze**) without **forgetting what the model already knows** (Fire, Smoke, Nofire, Smokefire).

Key Challenges:

Problem	What happened	impact
Catastrophic Forgetting	When we added the Haze class, the model forgot some old classes like Fire and Smokefire	Recall dropped: Fire (0.72), Smokefire (0.32),
Visual Overlap Between Classes	Smoke, Smokefire, and Fire images looked similar, confusing the model,	Smokefire F1 score was very low (~0.32–0.41),
Class Imbalance and Variability	Haze and Nofire were easy and clear to detect, but Smokefire images were fewer/mixed.,	High Haze F1 (0.99), poor Smokefire learning,
Performance Trade-off (LwF vs Full Retraining)	LwF saved old knowledge but slightly reduced overall performance compared to full retraining.,	LwF Accuracy (73%) vs No-LwF (78%)

Motivation:

- Forest fires are dangerous — detecting them early can save lives and nature.
- New problems like haze also need to be recognized by the system.
- But real-world AI systems can't be retrained every time something new appears — it’s too slow and expensive.
- You may not always have access to old training data (due to privacy, size, or loss).
- That’s why Learning without Forgetting (LwF) is important — it helps the model learn new classes like Haze while still remembering what it already knows.

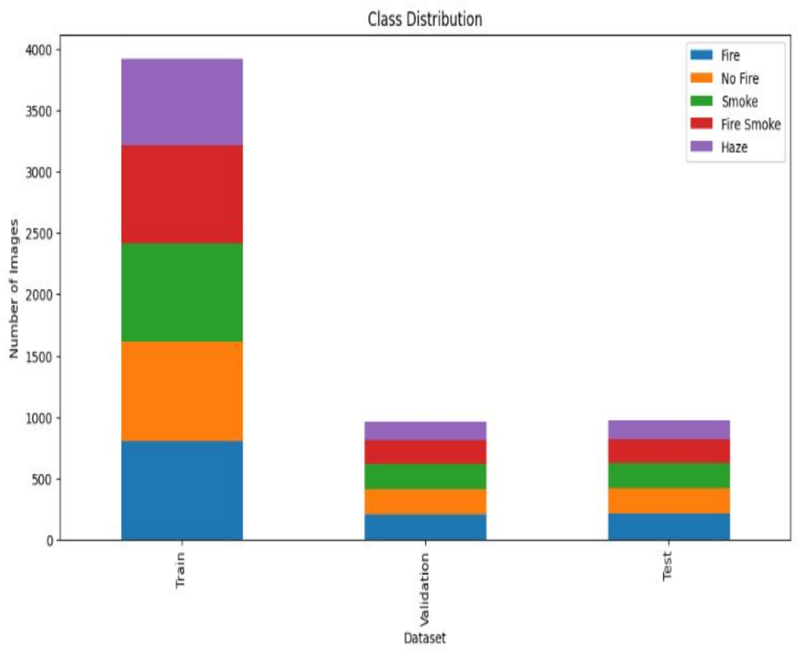
UNIVERSITY OF NORTH CAROLINA CHARLOTTE | THE GRADUATE SCHOOL

Related Works

Paper 1	Paper 2	Paper 3
An improved forest fire detection method using Detectron2 Approach: Uses Detectron2 + Deep Learning for accurate segmentation Strengths: High fire detection accuracy Suitable for real-time monitoring Limitations: Requires large labeled datasets Computationally expensive	***Forest Fire Detection using CNN and Transfer Learning*** Approach: Combines CNNs with transfer learning Strengths: Can be trained on smaller datasets Reduces training time using pre-trained models Limitations: Needs careful hyperparameter tuning Sensitive to dataset quality	***A Deep Learning Approach for Real-Time Wildfire Detection*** Approach: Deep learning model with real-time image processing Strengths: Enables early fire detection Improves situational awareness Limitations: Needs high-performance computing Limited by real-time data availability

Dataset Summary:

- **Total Classes:** 5
  ['fire', 'nofire', 'smoke', 'smokefire', 'haze']
- **Dataset Structure:**
  - **Train set:** ~70% of total images
  - **Validation set:** ~15%
  - **Test set:** ~15%
- **Class Balance:**
  - Most classes have **~800 images**
  - Slightly fewer for haze (~702), but still balanced overall

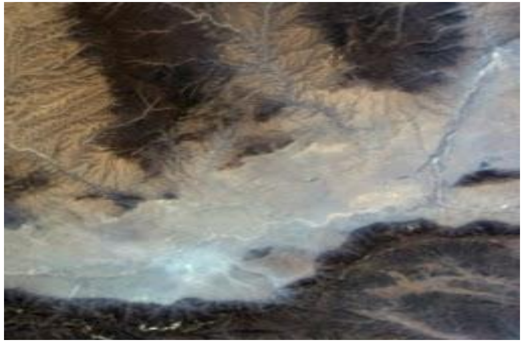


Dataset Summary:

	Fire	No Fire	Smoke	Fire smoke	Haze	Total
Train	800	810	800	810	702	3922
Validation	207	200	203	200	115	960
Test	210	210	200	200	151	971

Fire

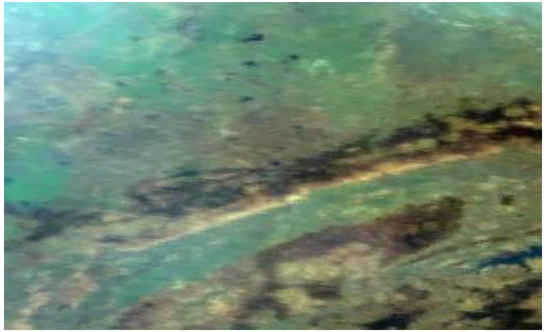
Smoke



Haze

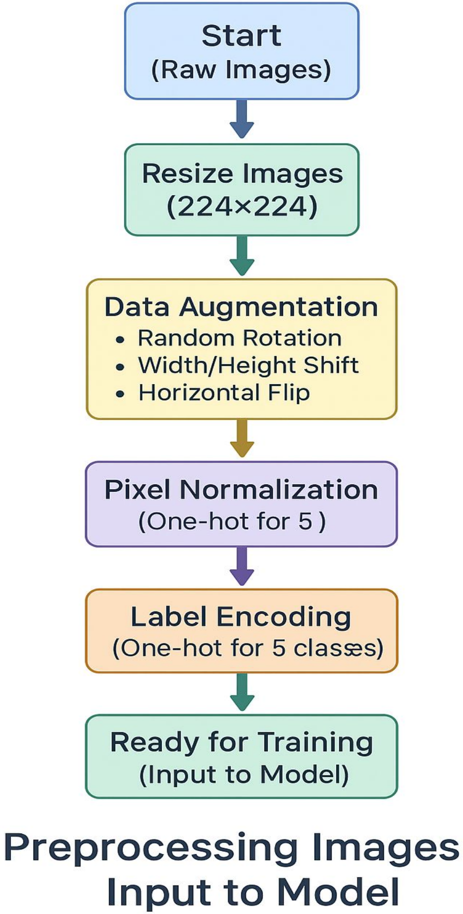
Smoke Fire

No Fire



Haze

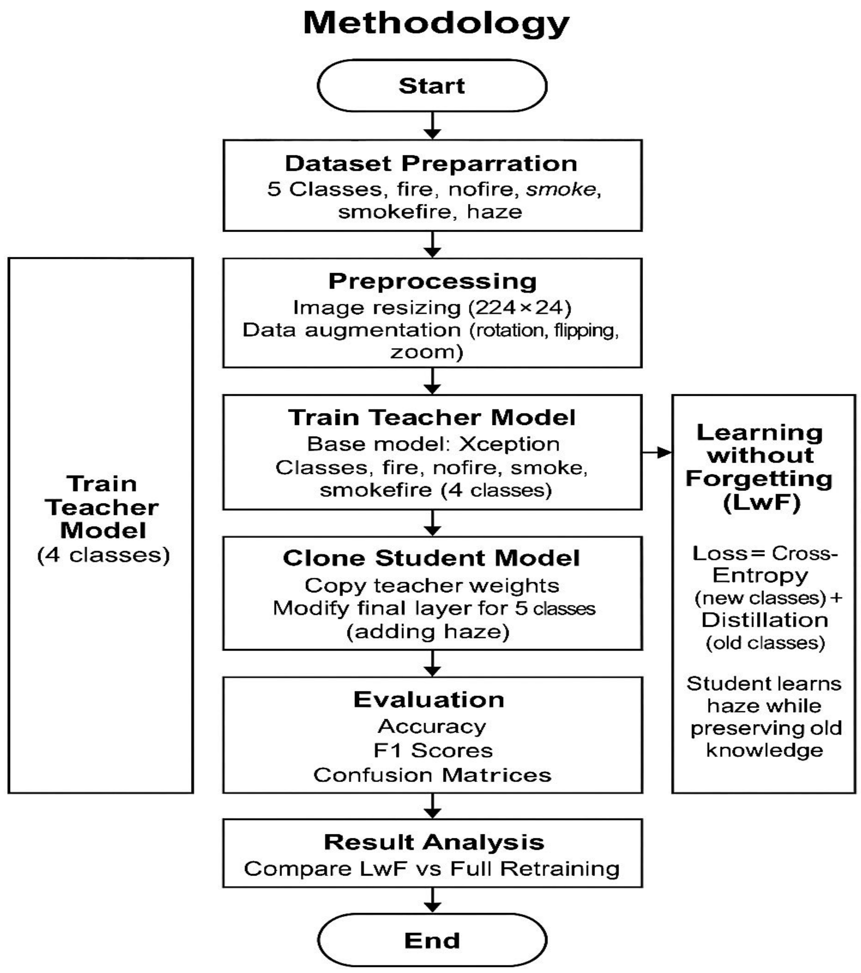
Preprocessing

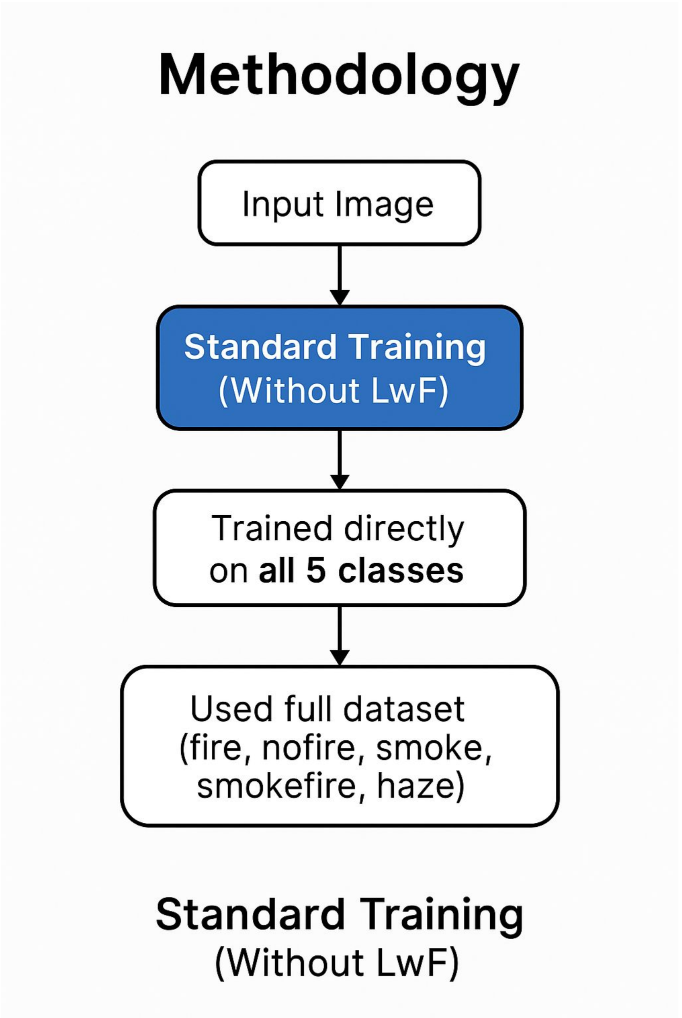


Methodology

Two Experiments Conducted:

- **Standard Training (Without LwF)**
  - Trained directly on **all 5 classes**
  - Used full dataset (fire, nofire, smoke, smokefire, haze)
- **LwF Training**
  - **Teacher model**: trained only on **4 classes** (fire, nofire, smoke, smokefire)
  - **Student model**: trained on new class (haze) + knowledge distilled from teacher
  - Combined two losses:
    - **Distillation loss** (match teacher's old predictions)
    - **Classification loss** (for haze)


Methodology
Start
Dataset Preparration
5 Classes, fire, nofire, smoke, smokefire, haze
Preprocessing
Image resizing (224×24)
Data augmentation (rotation, flipping, zoom)
Train Teacher Model
Base model: Xception
Classes, fire, nofire, smoke, smokefire (4 classes)
Clone Student Model
Copy teacher weights
Modify final layer for 5 classes (adding haze)
Evaluation
Accuracy
F1 Scores
Confusion Matrices
Result Analysis
Compare LwF vs Full Retraining
End
Train Teacher Model (4 classes)
Learning without Forgetting (LwF)
Loss= Cross-Entropy (new classes) + Distillation (old classes)
Student learns haze while preserving old knowledge


Methodology
Input Image
Standard Training (Without LwF)
Trained directly on all 5 classes
Used full dataset (fire, nofire, smoke, smokefire, haze)
Standard Training (Without LwF)

Model Architecture:

- Started with **pre-trained CNN models**:
  - **Xception** (main model)
  - **VGG16** (also tested)

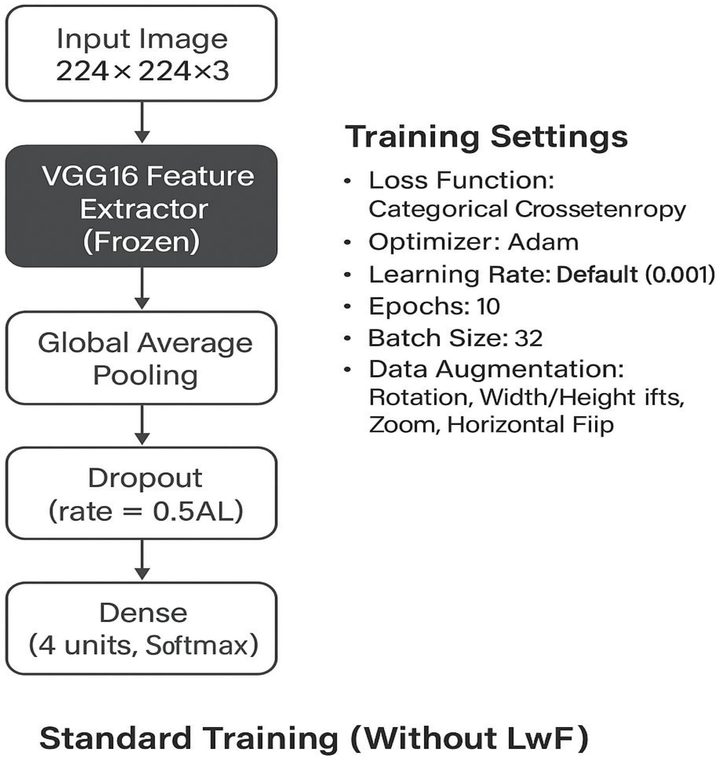
- Modified the final layers to fit **5 classes**

- Added **Dropout layer** and Dense softmax output

VGG16 (Baseline Model)

Method:

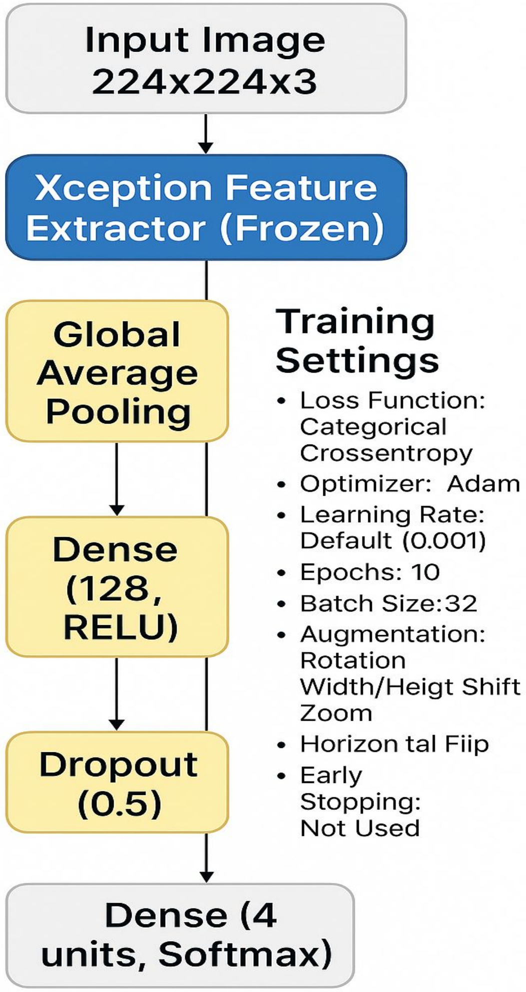
- Used VGG16 model pretrained on ImageNet.
- Feature Extraction: All convolutional layers frozen(trainable = False).



Xception (Baseline model)

Method:

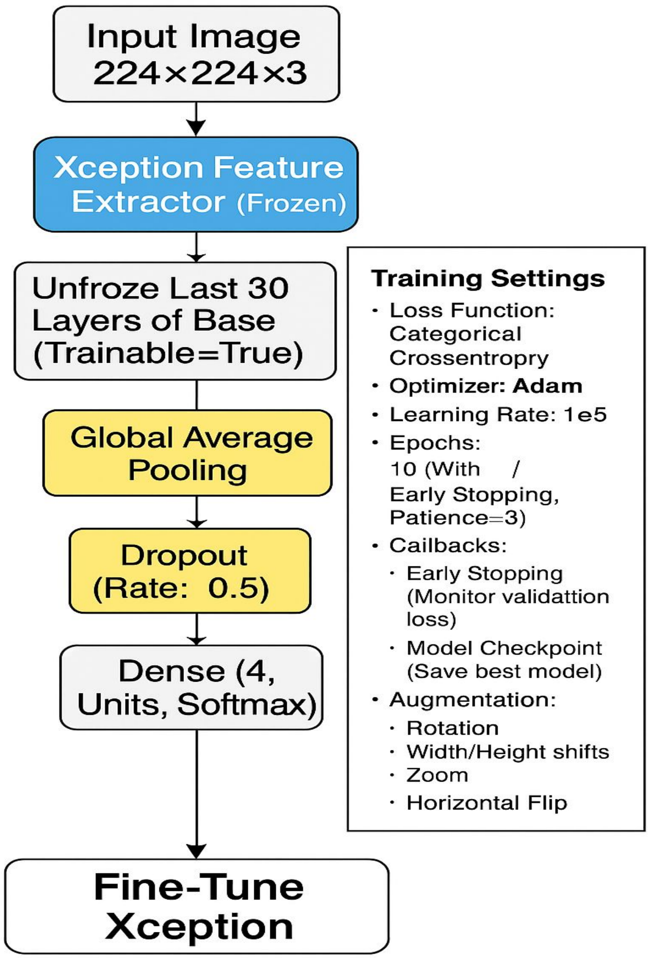
- Used Xception model pretrained on ImageNet.
- Feature Extraction mode: Convolutional layers frozen(trainable = False).



Fine Tuned Xception

Method:

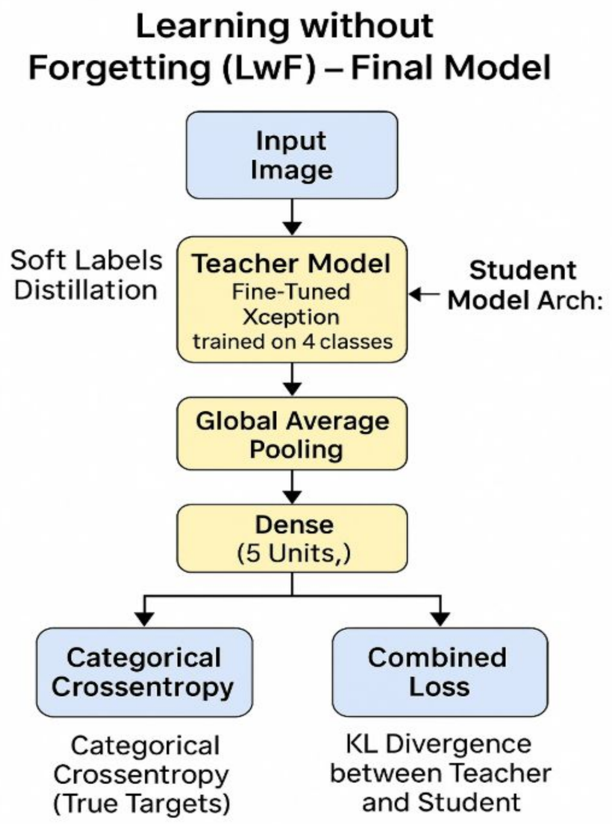
- Fine Tuned the Xception model after initial feature extraction.
- Unfroze the last 30 layers of the Xception base (trainable=True)
- Continued trainable to adapt better to fire /smoke/smokefire variations.



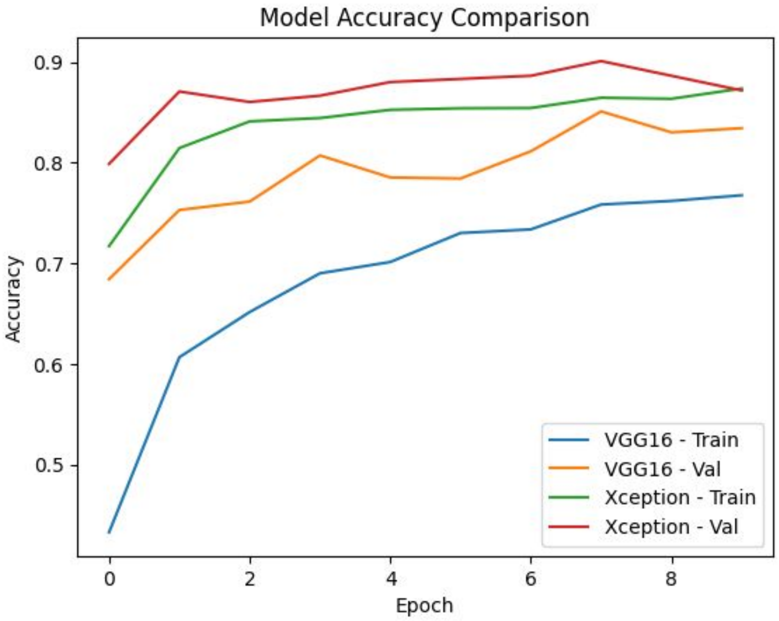
Learning without Forgetting (LWF) - Final model

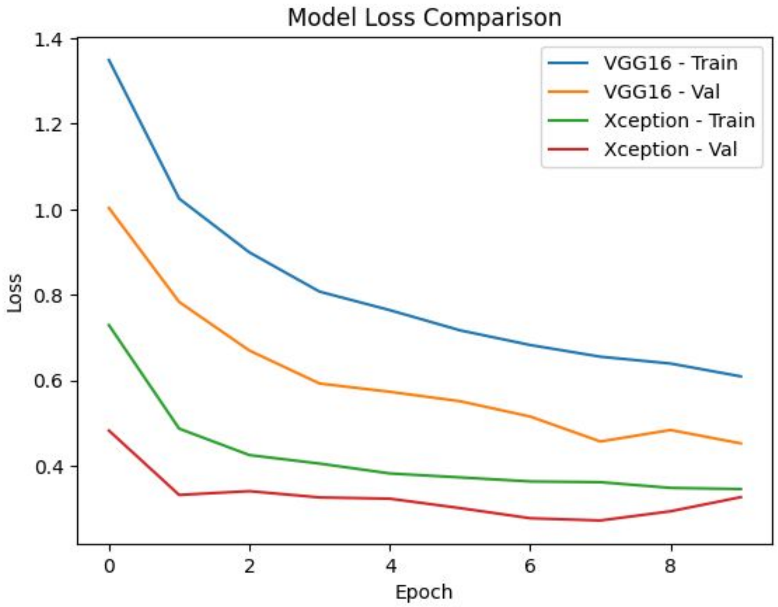
Method:

- Added Haze class to the dataset(making it 5 classes: fire, nofire, smoke, smokefire,haze)
- Used Learning without forgetting (LWF):
  - Teacher model : Fine tuned Xception trained on 4 classes.
  - Student model: new model trained on 5 classes using both:
    - Hard Labels(true class labels)
    - Soft labels (Teacher's knowledge via KL-divergence distillation)
- Student model Architecture:
  - Xception backbone (pretrained on ImageNet)
  - Global Average pooling
  - Dense(128, ReLU)
  - Dropout(0.5)
  - Dense(5, Softmax)
- Loss Function:
  - Combined Loss:
    - Categorical crossentropy (True targets)
    - KL Divergence between softened teacher and student predictions.
  - Alpha:0.5 (balance between hard loss and soft loss)
  - Temperature : 2.0 (to soften logits)
- Training Details:
  - Optimizer: Adam
  - Learning Rate : 1e-5
  - Epochs :10 , Batch Size:32
  - Call backs : Early Stopping, Model Checkpoint
  - Augmentation : Rotation, width/height shifts, zoom, horizontal flip



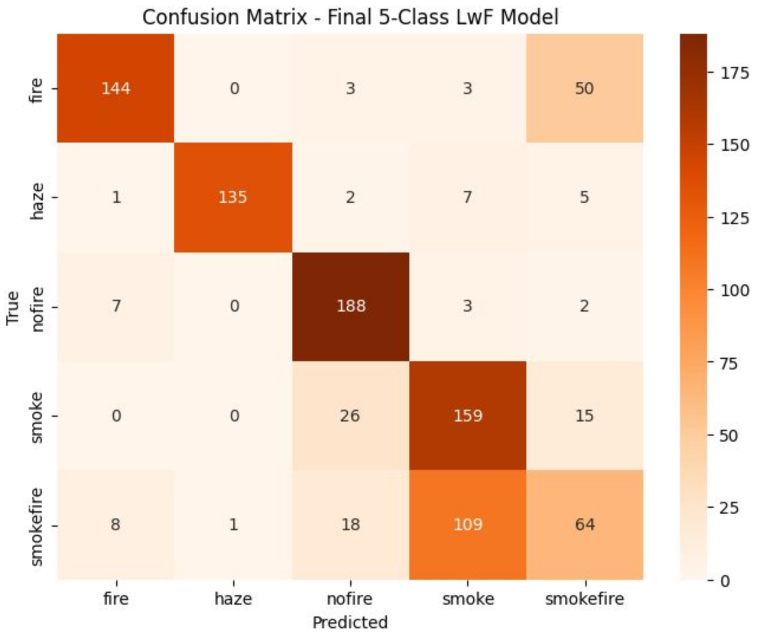
Model Architecture Comparison: VGG16 vs. Xception

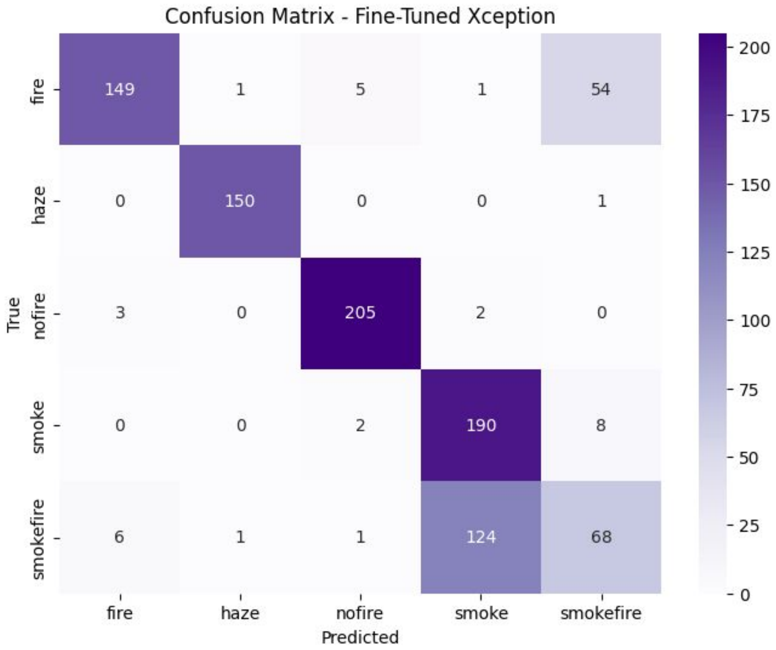




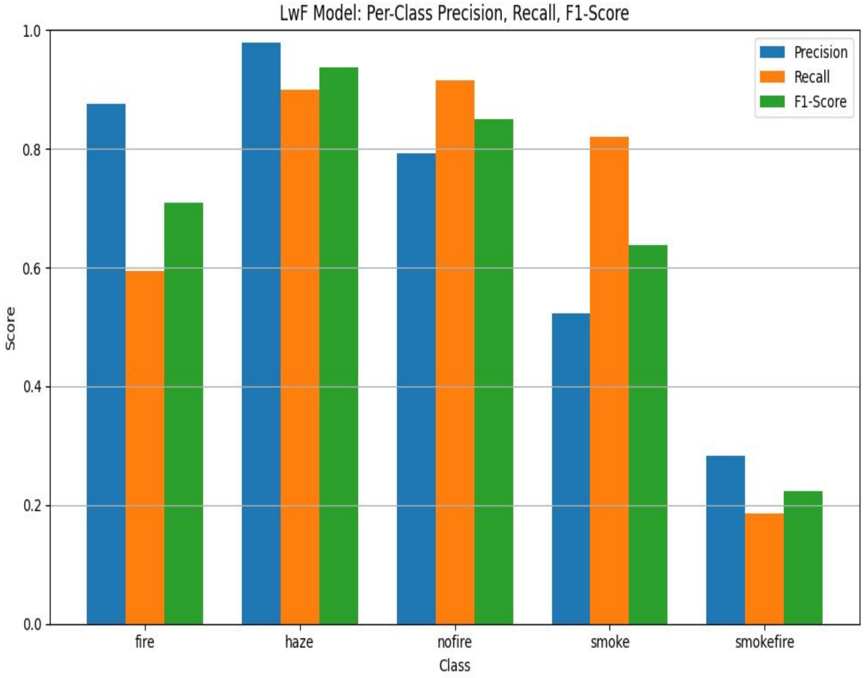
UNIVERSITY OF NORTH CAROLINA CHARLOTTE | THE GRADUATE SCHOOL

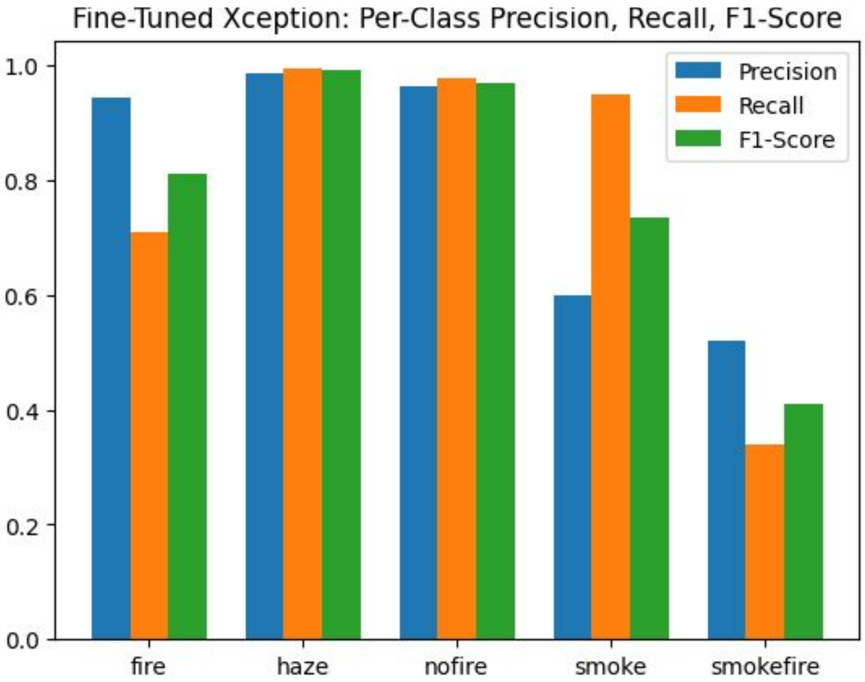
Confusion Matrix Comparison: LwF Model vs Base Model



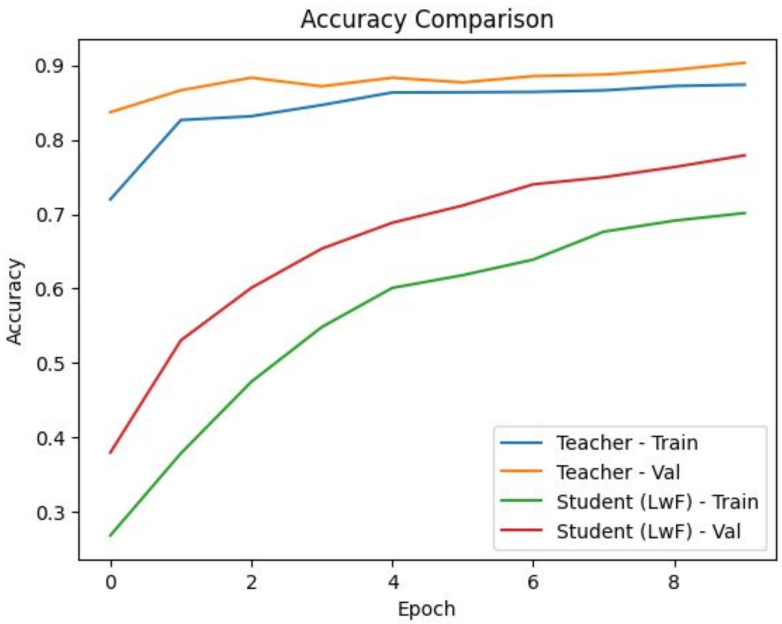


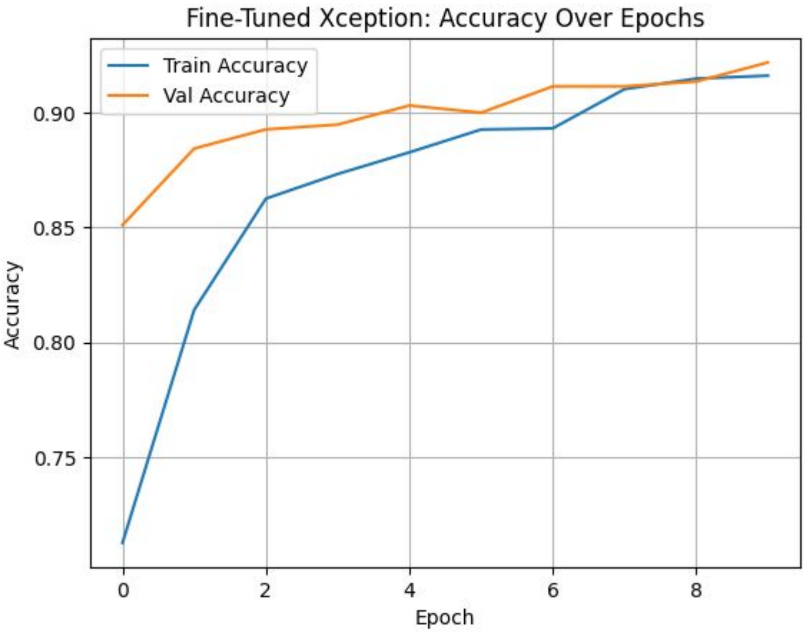
Comparison of LwF Model vs Fine-Tuned Xception: Per-Class Precision, Recall, and F1-Score



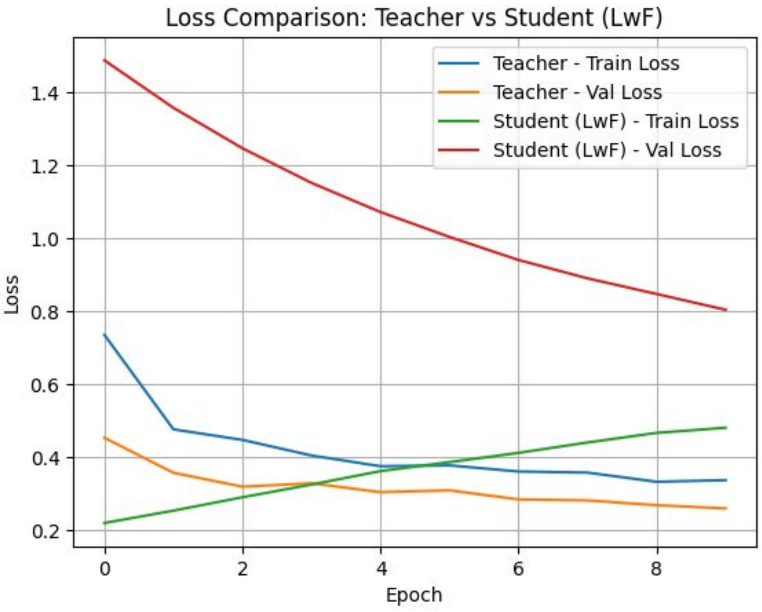


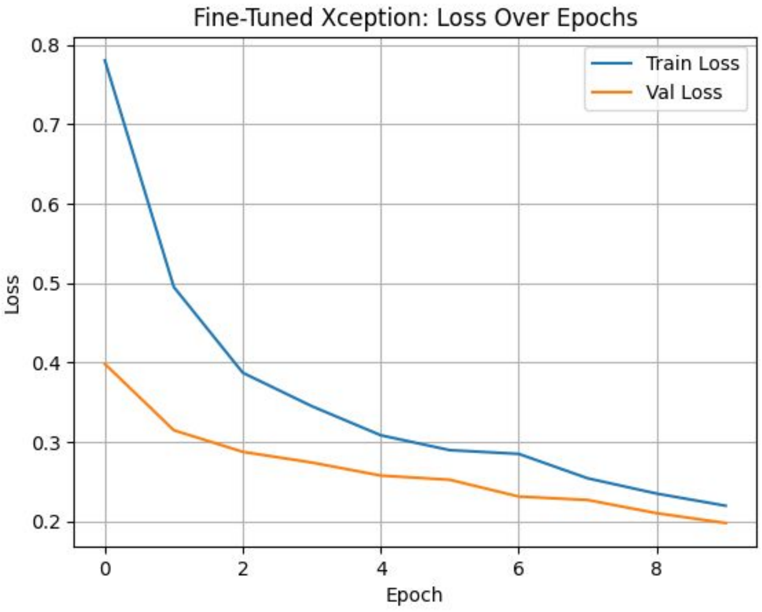
Training vs Validation Accuracy: Teacher vs Student (LwF) and Base Model





Loss Comparison: LwF vs Full Retraining





Results

Class	Precision(B)	Recall(B)	F1(B)	Precision(Lwf)	Recall(Lwf)	F1(Lwf)
Fire	0.94	0.71	0.81	0.88	0.72	0.79
Haze	0.99	0.99	0.99	0.99	0.99	0.99
No fire	0.96	0.98	0.97	0.98	0.97	0.97
Smoke	0.60	0.95	0.74	0.62	0.96	0.76
Smokefire	0.52	0.34	0.41	0.54	0.38	0.45

Note: B shows base model(model without Lwf)

Results

Metric	Baseline Model	Lwf Model
Test Accuracy	78%	75%
Validation Accuracy	90.3%	85.1%
Validation loss	0.2588	0.4574

Key Observations:

Aspect	Observation
Easy classes	Haze and Nofire easily detected
Challenging Class	Smokefire confused with fire/smoke(~45%)
Benefit of Lwf	Prevented forgetting old classes
Accuracy Trade-off	Lwf (75%) vs Full retraining (78%)
Fine-Tuning Impact	Fine tuning+early stoping improved result

Conclusion:

Key point	Observation
Effectiveness of Lwf	Helped add new class without retraining classes.
Knowledge Preservation	The student model was able to learn Haze while preserving the old classes (Fire, Smoke, Nofire, Smokefire).
Baseline vs Lwf	Baseline model had slightly better raw accuracy, but LwF provides better flexibility when old data is missing.
SmokeFire challenges	Smokefire remains a challenging class, needing more advanced feature extraction or additional data.

Future Work:

Future Direction	Description
Improve Smokefire Detection	Use better augmentation or attention mechanisms to help the model focus on mixed features.
Progressive LwF	Keep updating the model as new classes are added over time (real-world adaptability).
Real-Time Monitoring	Extend this work to real-time video fire and haze detection using lightweight CNNs.
Self-Learning System	Explore semi-supervised or self-supervised learning to reduce the need for labeled data.

Team Contributions:

- Prepared the **5-class dataset** (Fire, Nofire, Smoke, Smokefire, Haze)
  - Organized, cleaned, and split into train, validation, and test sets.

- Fine-tuned **Xception** and **VGG16** models for classification.

- Implemented a **custom Learning without Forgetting (LwF)** model in **Keras**, using:
  - Custom train_step() for dual loss (classification + distillation).

- Performed **experiments** and **evaluations**:
  - Trained models with and without LwF.
  - Compared their accuracy, precision, recall, F1-scores.

- Analyzed the results and created full **discussion and observations**.

- Created all **plots**, **graphs**, and **classification reports** used in presentation.

Sources Used:

- **Pre-trained Models:**
  - Xception and VGG16 from tensorflow.keras.applications

- **Learning without Forgetting (LwF):**
  Li & Hoiem, ECCV 2016 : https://arxiv.org/abs/1606.09282

- **Related Work Papers:**
  "An improved forest fire detection with Detectron2"
  - "Forest fire detection using CNN and Transfer Learning"
  - "Real-time wildfire detection using deep learning"

- **Libraries Used:**
  - TensorFlow, Keras, OpenCV, Pandas, Matplotlib

UNIVERSITY OF NORTH CAROLINA CHARLOTTE | THE GRADUATE SCHOOL

References:

Paper 1:
https://www.researchgate.net/publication/367539352_An_Improved_Forest_Fire_Detection_Method_Based_on_the_Detectron2_Model_and_a_Deep_Learning_Approach

Paper 2:
https://www.sciencedirect.com/science/article/pii/S2405844023103355

Paper 3:
https://www.techscience.com/csse/v44n2/48251/html