Classifying Disaster Type and Disaster Severity from Satellite Images

Jiaying Yang, Michael Soto, Dulce Torres
University of California Berkeley, 110 Sproul Hall #5800, Berkeley, CA 94720
RECEIVED DATE: December 13, 2024 E-MAIL ADDRESSES:
jiaying_yang@berkeley.edu, michael_soto@berkeley.edu, dulcetorres@berkeley.edu
Youtube Link: https://www.youtube.com/watch?v=qLs6e7XGpfE
Data drive (omitted from gradescope submission):
https://drive.google.com/drive/folders/1dl0yrCIo5m4BkqVnPLOSEpvp1adEiMzA?usp=share_link

Abstract

Two different classification tasks were performed on satellite images consisting of midwest-flooding, socal-fire, and hurricane-matthew disasters. The first task focused on distinguishing fire and flooding disasters while the second task classified hurricane images based on disaster severity. Both a logistic regression model and convolution neural network (CNN) were developed to complete each task. Explored image analysis techniques consisted of RGB , sobel edge detection, gabor filter, and local binary pattern methods. The CNN resulted in the best performance for both classification tasks based on accuracy and f1 metrics. An accuracy of 90% was obtained from the distinction of test flooding and fire images while an f1 score of 0.54 was achieved for the classification of hurricane severity.

Introduction

Digital image processing is a subdiscipline of computer science and engineering that is constantly evolving in both terms of computational speed and accuracy. The application of both supervised and unsupervised learning techniques has specially become resourceful in regards to feature extraction, pattern recognition, object detection and classification methods, all of which are especially relevant in image analysis.[9] Such analysis can be useful in regard to natural disasters, where current methods have focused on the implementation of machine learning techniques to assess damage, predict occurrences, and classify disasters.[10] This work focuses on integrating common applications of machine learning to address natural disasters, and it serves as an exploration of performance when implementing convolutional neural networks and logistic regression models to perform such tasks.

Several methods of analysis have proven useful in the examining of images. In a true color image, there are three distinct values corresponding to each pixel. These values represent the red, blue, and green components of the pixel. Each value contributes to their respective color channels, and combining the channels results in an image with varying complexions. Colors have a profound influence on an image, thus the RGB values can be leveraged to extract prominent image descriptors and objects from a scene.[7] In addition to color analysis, gabor filters, sobel edge detection, and local binary patterns are effective methods in texture and edge analysis. While these methods reveal similar information, they are unique in their mechanisms and strengths. Gabor filters are linear filters that pick up on the frequency content in a given direction for a localized region. The application of gabor filters are especially helpful for spatial location and frequency analysis within images.[8] Sobel edge filtering measures the 2-D spatial gradient

measurement on the image.[6] Furthermore, local binary patterns work by thresholding the environment of a pixel, resulting in a binary value. They are especially robust in gray scale changes.[5]

All of the aforementioned methods of analysis were considered in the classification of satellite images consisting of socal fire, midwest-flooding and hurricane disasters. Essentially, two different classification tasks were performed. The first task focused on distinguishing flooding and fire images through the incorporation of RGB and sobel edge features. The second task focused on classifying the severity of disasters based on rgb, sobel edge, gabor filter, and local binary pattern features. Both a logistic regression model and convolutional neural network were developed for each task, with the latter resulting in the best performance based on accuracy and f1 values.

Data Analysis and Preparation

Data Sampling

Image Data was obtained from the xView2 Challenge data set. This data set consists of satellite images taken of environments after one of the following natural disasters: a hurricane, fire, or flood. More specifically, the hurricane images were of Hurricane Matthew, fire images were obtained from the SoCal region, and flooding images were taken from midwestern floods. Each sample in the data corresponds to a unique satellite picture taken from one of three different disasters. Each image has an associated label corresponding to damage severity. Damage labels fall within a scale of zero to three, where a label of zero signifies the least amount of damage was sustained and three signifies the most damage.

Exploratory Data Analysis

The total number of labeled images in the dataset was 26535. The number of images attributed to hurricanes, fires, and floods were 11151, 8380, and 7004, respectively. To address outliers, particularly images with significantly larger dimensions, a log transformation was applied to the dimensions, normalizing the right-skewed distribution. Missing values were identified and addressed through imputation depending on the extent of missingness and its potential impact on analysis. We apply feature extraction techniques like RGB channel analysis, texture patterns (e.g., LBP), and edge detection (e.g., Sobel filters) to derive structured insights from unstructured satellite image data.

Each image is represented by a tensor of discrete quantitative dimensions (pixel height, pixel width, number of rgb channels) as this is a common way to represent images for channel processing and analysis.[2,3] Damage labels are categorically quantitative and increase from 0 to 3, ranging from no damage to complete environmental destruction. Image dimensions were analyzed through kernel density estimation. The dimensions were then log scaled for outliers, as there is a right-skewed distribution of the untransformed images; some of the images were of significantly larger dimensions i.e. image area > 100000. These distributions are shown in

Figure 1. The peaks for flood image dimensions are right-shifted to those of hurricanes, and left-shifted when compared to those of fires.

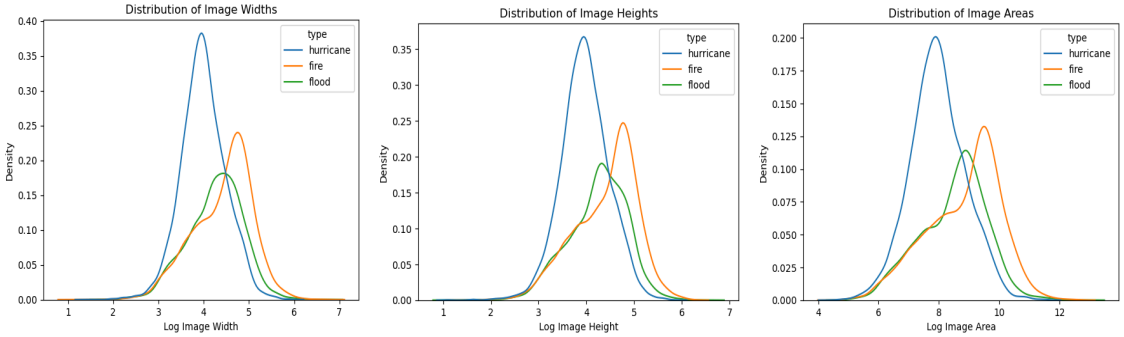

Figure 1 Image Width, Image Area, and Image Height (Log Transformed) distribution for each disaster type.

The distributions for the damage labels can be seen in **Figure 2.** There is a notable difference in the number of labels corresponding to damage level 0 vs damage level 1-3 for the fire and flood images. Additionally, there are a large number of labels that correspond to a severity level of 1 for the hurricane images.

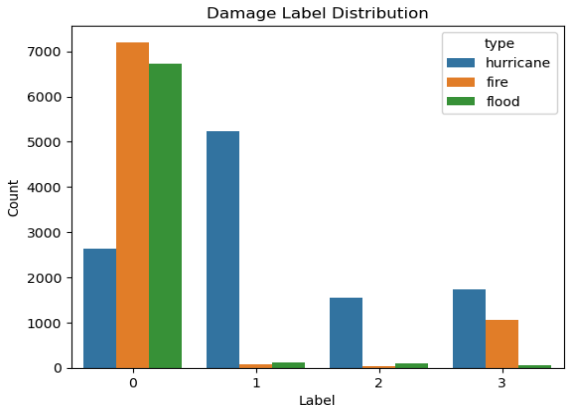

Figure 2 The distribution of damage labels across disaster types

A question of focus was whether or not RGB analysis would allow for the distinction between disaster types, especially between midwest-flooding and socal-fires. In order to explore this question, the given images were separated by disaster type. The average individual RGB scores were then calculated for each row of pixels in a given image. This resulted in a data frame for each disaster with the average red, blue, and green scores for each distinct image in the given data set. As shown in **Figure 3**, color distributions for the socal-fire disaster were concentrated at slightly higher values across all RGB components when compared to the midwest-flooding disaster. Considering this pattern, it was believed RGB component scores could be a candidate for distinguishing flooding and fire disasters. In addition to RGB analysis, the distributions of

sobel edge scores were also examined. The average sobel edge score was determined for each image. The distributions for the resulting values are shown in **Figure 4**.

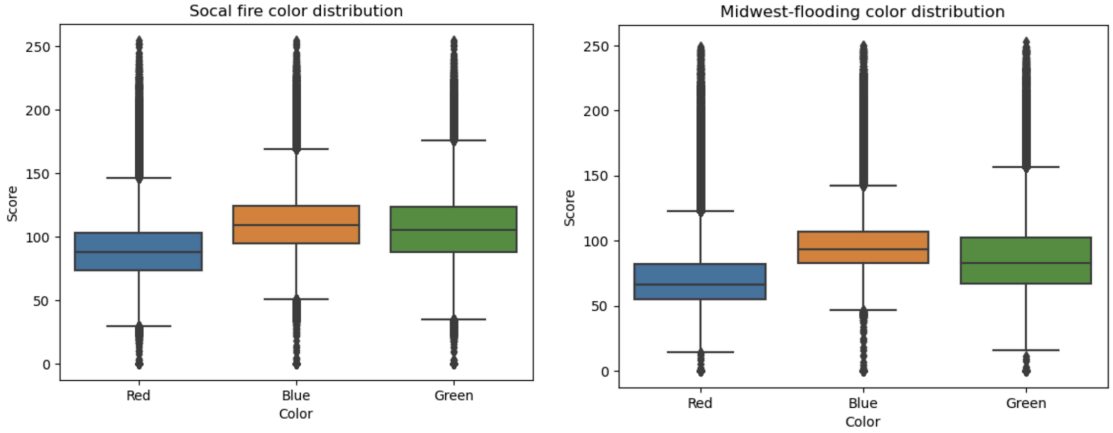

Figure 3 Distribution of average red, blue, and green color values for the Socal fire and midwest-flooding disasters.

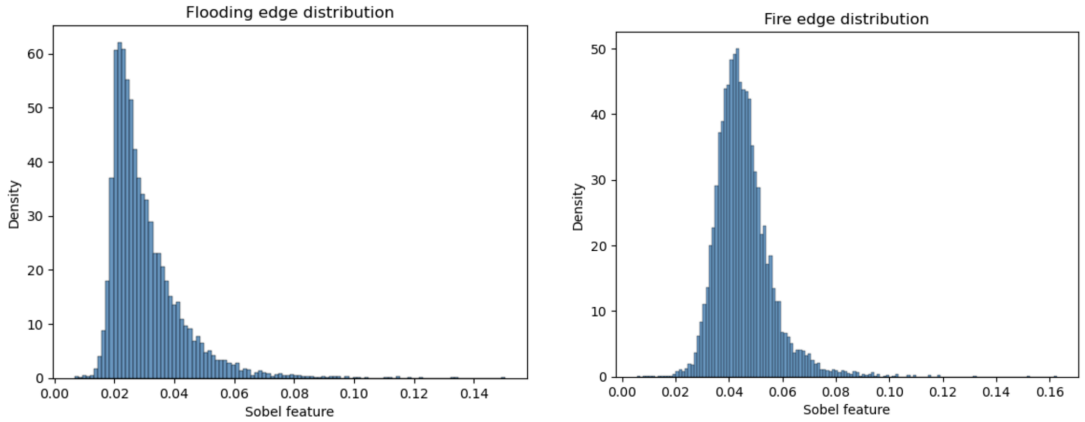

Figure 4 Distribution of sobel edge detection values for flooding and fire satellite images.

Various untransformed image distribution metrics were plotted for the hurricane images to see if there was a fundamental difference between damage levels based on these metrics alone. In every instance, there is a left-shift of the mean peak density value for a lower damage level image, except for the local binary pattern where the damage level 3 images are clearly right-skewed as opposed to the damage level 1 images. These distributions can be seen in **Figure**

5. Local Binary Patterns provide separable distributions for severity levels 1 and 3.

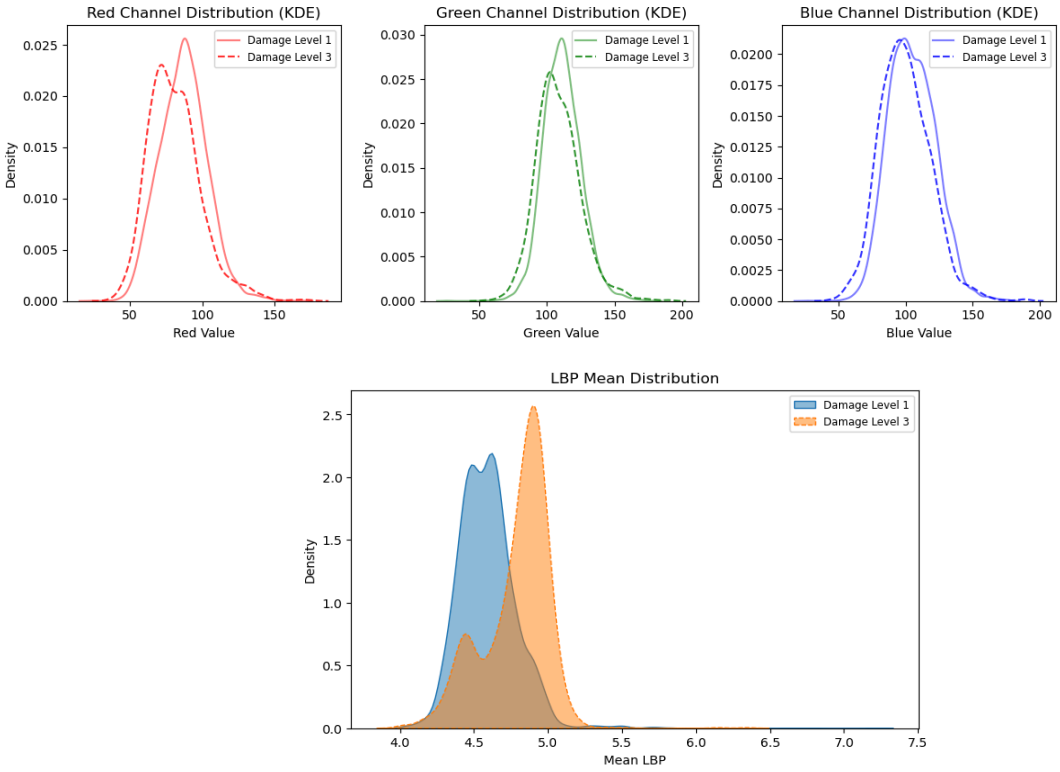

Figure 5 Mean LBP of hurricane images classified as level 1 and level 3 severity

Methodology

Feature Engineering

Initially, a multitude of features were considered for training a Logistic Regression model focused on the classification of damage severity. Images were first split into each of the three separate rgb values, and the mean channel value was calculated across every pixel for each of the images. For edge and texture detection, there were three filtering methods performed. Firstly, Sobel edges were calculated across all pixels for each of the images. Then each image was fitted with a gabor kernel consisting of a theta orientation of 0 degrees, Gaussian envelope standard deviation of 1, and a frequency of 0.1. Lastly, a local binary pattern was calculated for each of the images, with a neighborhood radius of 3. Due to the right-skewed nature of mean local binary pattern values observed during initial EDA,, the log transformed value of each mean local binary pattern was calculated before model training for the hurricane damage labels.

Features of focus for the logistic regression model aimed to predict disaster type were RGB components and sobel edge features. These features were selected based on the varying distributions for flooding and fire images.

Feature engineering outside of EDA analysis was unnecessary as the same data frames were utilized for the training of the logistic regression model. This data frame consisted of three distinct columns corresponding to the average RGB component and sobel edge scores.

Preparation of input data prior to CNN training consisted of resizing and normalizing the images. Normalizing was performed based on the RGB scale, and it resulted in pixel values ranging from zero to one. All images were resized into a 38x38 frame.

Model Selection and Implementation

Initial approaches for classifying flood and fire images consisted of logistic regression. This model was believed to be an appropriate method considering the task at hand was not a regression analysis, but instead a classification problem. In order to identify the optimal parameters of the model, 5-fold cross validation was implemented. Different hyperparameters were assessed consisting of varying penalty and solver algorithms. The L1 regression and liblinear solver resulted in the best accuracy of 84%. In order to achieve further improvements, consideration was given to multicollinearity. As shown in **Figure 6**, there was a lack of linearly independent features in the utilized data set.

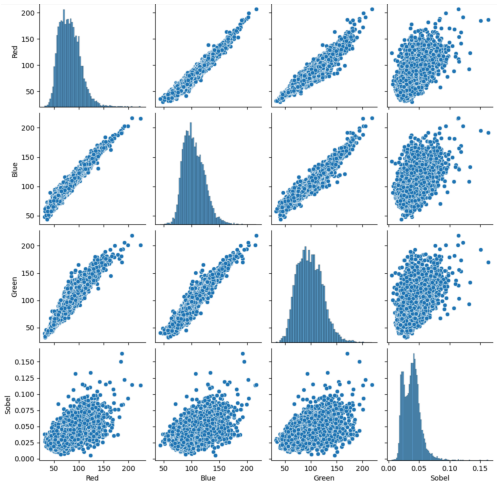

Figure 6 Correlation between the RGB and sobel features of the fire and flood images.

Thus, PCA was performed to improve the interpretation of variable contribution within the model, thus enhancing accuracy. However, implementation of a PCA transformed data set significantly reduced accuracy to 53%.

Evident in gauged performance metric, the current logistic regression model was not sufficient in achieving maximized results. Since only the mean values were considered during feature extraction, it was suspected the current data frame did not contain enough features to provide the model with a sufficient amount of information to distinguish between different disaster types. Thus, the methodology approach transitioned to a convolutional neural network as

a way to retain image information while also leveraging the image classification performance of a CNN.

The initial framework of the model was an 8 layer network consisting of convolution, maxpooling, flattening, activation, and softmax layers. Several changes were incorporated to the original CNN model to further improve performance. Modified parameters were based on the gauged metric of performance during training: accuracy. This was an appropriate metric as there was a relatively even representation of fire and flood images within the training data set. Considerations focused on reducing the aspect of overfitting, thus improving model performance on unseen data. Thus, extensive testing was performed to understand how different dropout rates and model complexity affected accuracy. This was performed across several 5-fold cross validation splits. Evident in earlier models, it was clear that validation accuracy would decrease after a certain amount of epochs. Thus, early stopping was implemented to address the decay in validation performance. However, there continued to be a clear distinction between training and validation accuracy. This is shown in **Figure 9** where training and validation accuracies are capped at 94% and 90%, respectively. In order to address these patterns, different dropout rates ranging from 0.3 to 0.7 were tested. An additional dropout layer after the first convolutional layer was also implemented. Modifications in dropout rate well as the inclusion of an additional dropout layer all resulted in significantly lower accuracy values of around 84%.

Additional improvements consisted of fine tuning class weights and model architecture. Although the representation of both fire and flood disasters was relatively even, it was recognized there were 1376 more fire images within the 15384 training images. The inclusion of class weights during training were explored across several different models. Significance improvements were not observed when gauging accuracy on respective validation sets as accuracies remained within 90%. Further adjustments consisted of increasing the number of filters within the convolutional layers of the CNN in ascending order. This was done intentionally to enhance the model's ability to capture intricate details within the deeper levels of the CNN. Improvements in accuracy were not observed. The final model consisted of a single dropout layer with a dropout rate of 0.5 prior to the output layer. A sigmoid activation is also implemented in the output layer to accommodate the binary classification task. Additionally, binary cross entropy was the utilized loss method. The model summary for the CNN used in the classification of fire and flood images is shown in **Figure 7**.

..

Layer (type)	Output Shape	Param #
conv2d_20 (Conv2D)	(None, 36, 36, 32)	896
max_pooling2d_15 (MaxPooling2D)	(None, 18, 18, 32)	0
conv2d_21 (Conv2D)	(None, 16, 16, 64)	18,496
max_pooling2d_16 (MaxPooling2D)	(None, 8, 8, 64)	0
conv2d_22 (Conv2D)	(None, 6, 6, 120)	69,240
max_pooling2d_17 (MaxPooling2D)	(None, 3, 3, 120)	0
conv2d_23 (Conv2D)	(None, 1, 1, 200)	216,200
flatten_5 (Flatten)	(None, 200)	0
dense_10 (Dense)	(None, 64)	12,864
dropout_6 (Dropout)	(None, 64)	0
dense_11 (Dense)	(None, 2)	130

Figure 7 Model summary for the final CNN network that classified between fire and flood disasters.

An obstacle in model training specific to the hurricane damage classification was the large number of images corresponding to a severity level of 1. This would likely lead to overfitting on the basis of class imbalance. Undersampling of the majority class was performed, as this is a common technique to conserve time and computational resources.[4] Specifically, 2000 random samples corresponding to each disaster severity level were withdrawn from the image dataset without replacement. After class balancing was implemented, the training and test datasets were split into 5-fold cross-validation samples. Each cross-validation split was implemented with a 20% validation to training ratio. For each iteration of the split, both the multilogistic regression and CNN model were reinitiated and trained.

The multinomial logistic regression model was prepared with grid search parameterization. Max iterations were capped at 1000 and 10000 epochs. In addition, LBFGS, Newton conjugate gradient, and stochastic average gradient optimization methods were parameterized in order to find the model with the best fit. Ridge regression was incorporated into the model as a regularization method.

The CNN was fitted with resized images of 38 x 38 x 3 dimensions. Additionally, each image underwent scaling such that all pixels contained values between 0 and 1. Layer architecture consisted of three convolutional layers increasing from a total of 32 to 128 neurons, each followed by a max pooling layer. Hidden activation layers incorporated the ReLU activation function, while the final output layer consisted of a softmax activation function and four output neurons, one corresponding to each classification category.

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_21 (Conv2D)          (None, 38, 38, 32)        896

 max_pooling2d_14 (MaxPoolin  (None, 19, 19, 32)       0
 g2D)

 conv2d_22 (Conv2D)          (None, 17, 17, 64)        18496

 max_pooling2d_15 (MaxPoolin  (None, 8, 8, 64)         0
 g2D)

 conv2d_23 (Conv2D)          (None, 6, 6, 128)         73856

 flatten_7 (Flatten)         (None, 4608)              0

 dense_14 (Dense)            (None, 128)               589952

 dense_15 (Dense)            (None, 4)                 516

=================================================================
Total params: 683,716
Trainable params: 683,716
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8 Model architecture for the hurricane damage classification CNN

Results

Classification of Fire and Flood Disasters

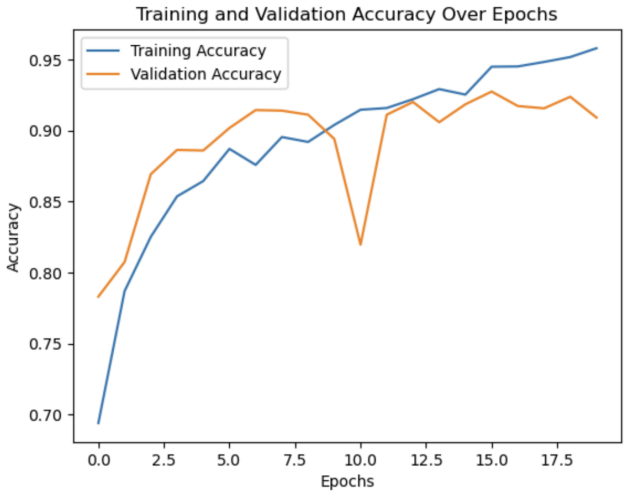

Figure 9 Training and validation accuracy for one of the first binary classification models.

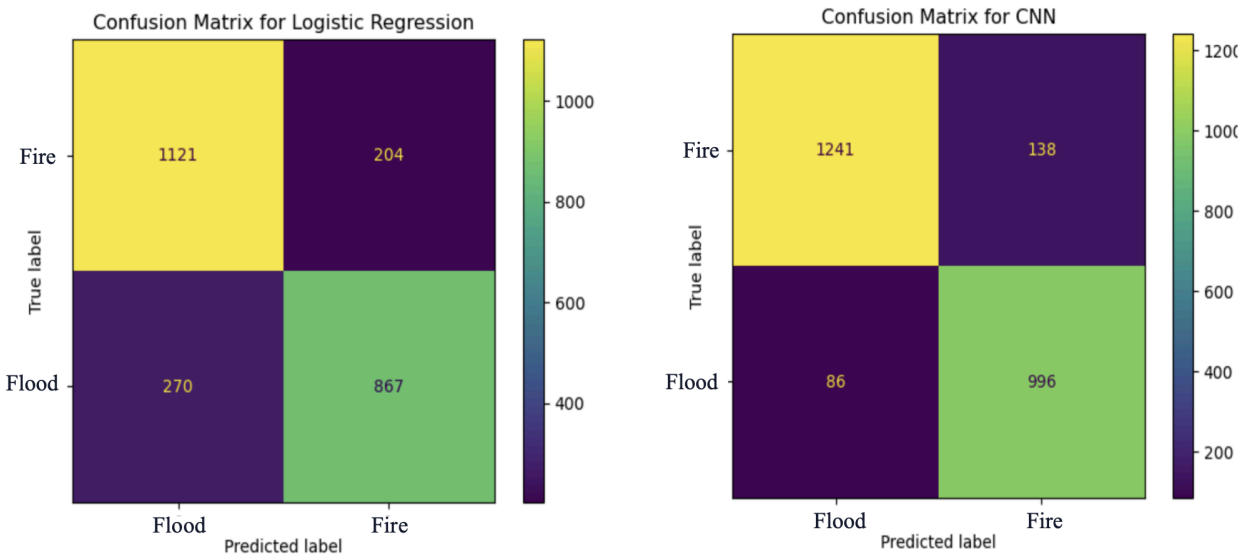

Figure 10 Confusion matrices for the CNN and logistic regression models that predict the classification of fire and flood images.

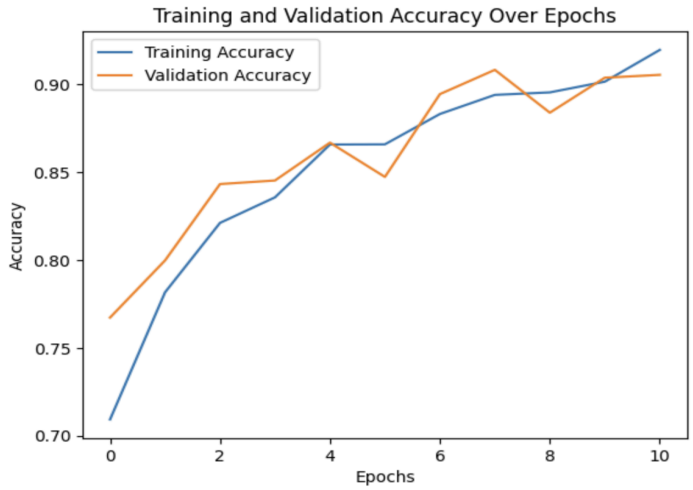

Figure 11 Training and validation accuracy for the final binary classification model.

Hurricane Damage Level Classification

Across each 5-fold cross validation split the average accuracy and f1 score for the multiclass logistic regression model were 44.5% and 0.441, respectively. The average cross validation and accuracy and f1 scores for the CNN trained model were higher at 64.3% and 0.640, respectively. It can be seen in **Figure 12** that accuracy, precision, and recall generally

trended upwards over the course of 30 epochs, with the exception of precision for the validation set. The training loss trended downwards, while the validation loss exhibits a clear minimum around 16 epochs.

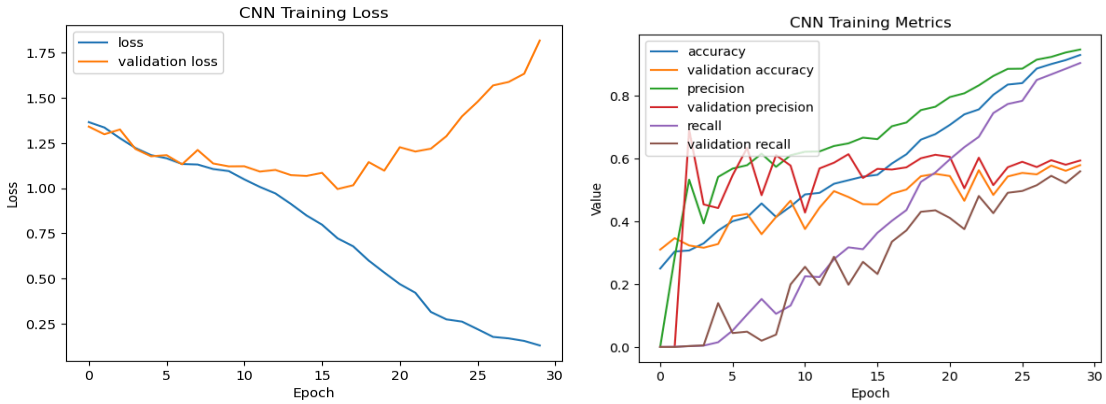

Figure 12 Loss, accuracy, precision, and recall during CNN training for hurricane damage classification

Both of the models had the highest number of true positive labels when predicting level 3 damage, as indicated by the confusion matrices in **Figure 13**. The highest number of misclassifications occurred for a true label of 2 and predicted label of 1 for the logistic regression model. For the CNN model, the highest number of misclassifications occurred for a true label of 1 and a predicted label 2.

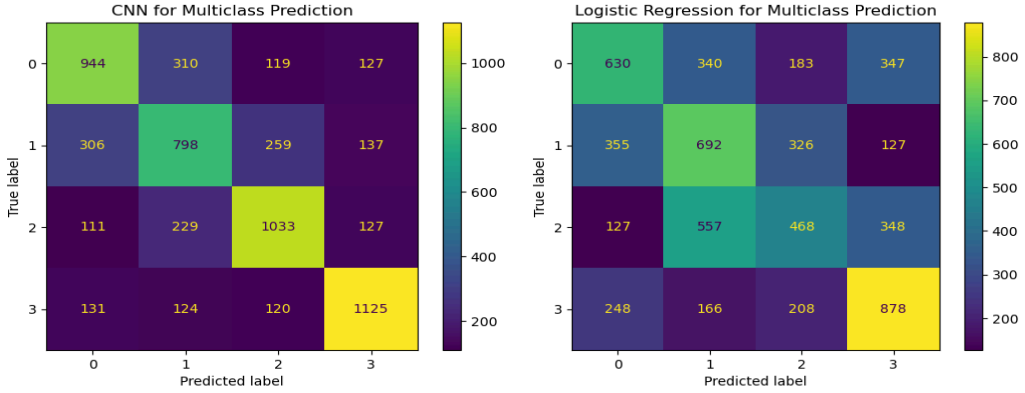

Figure 13 Confusion matrix of true and predicted hurricane damage levels for the convolutional neural network model (left) multinomial logistic regression model (right)

After training, the CNN model was run over 15 epochs on a separate test dataset of satellite images, yielding an f1 score of 0.546. While this score is lower than the validation sets, it is well above the score of a random classifier.

Discussion

Implementation of the CNN for the classification of fire and flood disasters resulted in a test accuracy of 90%. Analysis of performance based on accuracy suggested the CNN model was the most successful in performing the classification task relative to the logistic regression model. This is evident through the true positive values along the diagonal of the confusion matrix shown in **Figure 10**. Although extensive training was performed with numerous CNN frameworks and hyperparameters, accuracy remained relatively consistent between the initial and revised CNN architectures. It was evident in earlier models that overfitting was prominent with training accuracies significantly out performing validation accuracies. This is shown in **Figure 9**. Thus, focus was taken to mitigate the issue of overfitting to the train images. The training and validation accuracy over several epochs for the final model is shown in **Figure 11**. Although the gap between train and validation accuracy was reduced, the point of convergence did not occur at an appropriate accuracy. Since there are many stakeholders at risk when it comes to the misclassification of a disaster, it is especially important to develop a computer vision model that is able to successfully classify unseen data. A successful model in this case would be a CNN with significantly improved generalization performance. In order to reduce those affected by an improper classification as well as save resources, further research would need to be explored in the binary classification of flooding and fire images. Exploration could begin with a different preparation of data. A more sophisticated method would want to be deployed as a way to prevent image distortion during the feature engineering process through the incorporation of minimal padding. Since computer vision tasks rely heavily on large data sets to prevent overfitting, data augmentation would also be explored. Further modifications in model architecture would also be implemented in order to find the most optimal framework and set of hyperparameters.

Similarly to the classification of fire and flood disasters, the CNN greatly improved accuracy for hurricane damage assessment. In addition, the F1 score was significantly better. Each metric saw an improvement of approximately 20%. The lower accuracy in logistic regression classification was likely due to limited feature transformation when training the model. In the future, it would be beneficial to train a model on more localized pixel by pixel image features, as opposed to taking mean values across all pixels. For the CNN, the validation loss clearly exhibits a minimum value, thus it can be beneficial to run the model for lower epochs: in this case between 15 and 20 epochs, as opposed to 30. This decrease in training duration led to less overfitting of the CNN on the testing dataset. Future considerations for model adjustment include tuning the number of neurons per convolutional layer, and increasing layer depth, provided computational resources are sufficient.

Conclusion

The classification between flood and fire images as well as the multi-class classification of hurricane images was performed. Both a logistic regression model and CNN were utilized, with the latter method resulting in the best performance. Performance for the binary classification was gauged through accuracy, with the final model resulting in 90%. Model

performance for the classification of different damage labels was analysed through F1 values. An optimal F1 score of 0.54 was achieved. The societal repercussions of building a widespread computer vision model for the purpose of classification can be vast, possibly impacting millions of lives in the future. The cost of misclassifying or underestimating the severity of a new or ongoing disaster could lead to loss of monetary wealth and life. Ethically, it might also be a challenge to implement such a model, as collecting data for this purpose would involve widespread property surveillance. Despite these concerns, an accurate, unbiased classifier potentially provides a net benefit for society, allowing for quicker emergency personnel response times as well as proper budgeting and dispersal of resources to those affected. Ultimately, this analysis of satellite images has highlighted the strength of Convolutional Neural Networks when compared to more traditional regressive models for the purpose of classifying fires, floods, and hurricanes. With enough computational resources, the setup of CNN architecture can drastically cut down on implementation time, where minimal initial preprocessing and data analysis is desired. Furthermore, the ability of CNNs to recognize spatial patterns within image subregions can lead to increased model accuracy, provided sufficient training. The next step for future researchers taking a neural network approach to disaster classification based on satellite images would be to fine tune CNN parameters for a similar dataset or combine a CNN based model with novel techniques, such as attention mechanisms. Lastly, there are various other disasters that should be studied with computer vision models such as tornadoes, tsunamis, blizzards, and earthquakes, as these natural occurrences can often be equally as devastating.

References

1. Baskar, A., Rajappa, M., Vasudevan, S. K., & Murugesh, T. S. (2023). *Digital Image Processing* (First edition., Vol. 1). CRC Press. https://doi.org/10.1201/9781003217428

2. Duan, Y., Wang, N., Zhang, Y., & Song, C. (2024). Tensor-Based Sparse Representation for Hyperspectral Image Reconstruction Using RGB Inputs. *Mathematics*, *12*(5), 708-. https://doi.org/10.3390/math12050708

3. Qureshi, M. A., Deriche, M., & Beghdadi, A. (2016). Quantifying blur in colour images using higher order singular values. *Electronics Letters*, *52*(21), 1755–1757. https://doi.org/10.1049/el.2016.1792

4. Abhishek, K., & Abdelaziz, M. (2023). *Machine Learning for Imbalanced Data : Tackle Imbalanced Datasets Using Machine Learning and Deep Learning Techniques / Kumar Abhishek and Mounir Abdelaziz.* (First edition.). Packt Publishing Ltd.

5. Matti Pietikäinen (2010) Local Binary Patterns. Scholarpedia, 5(3):9775.

6. D. Ai, G. Jiang, S.K. Lam, C. Li (2023). Computer vision framework for crack detection of civil infrastructure—A review

7. Kumar, Tarun, and Karun Verma. "A Theory Based on Conversion of RGB image to Gray image." *International Journal of Computer Applications* 7.2 (2010): 7-10.

8. “Gabor Filter.” *Wikipedia*, Wikimedia Foundation, 26 Apr. 2024, en.wikipedia.org/wiki/Gabor_filter#:~:text=The%20Gabor%20filter%20is%20a,point%20or%20region%20of%20analysis.

9. Ayub Khan A, Laghari AA, Ahmed Awan S. Machine Learning in Computer Vision: A Review. EAI Endorsed Scal Inf Syst [Internet]. 2021 Apr. 21 [cited 2024 Dec. 14];8(32):e4.

10. B. Yogameena, C. Nagananthini, Computer vision based crowd disaster avoidance system: A survey,International Journal of Disaster Risk Reduction, Volume 22, 2017, Pages 95-129, ISSN 2212-4209, https://doi.org/10.1016/j.ijdrr.2017.02.021.