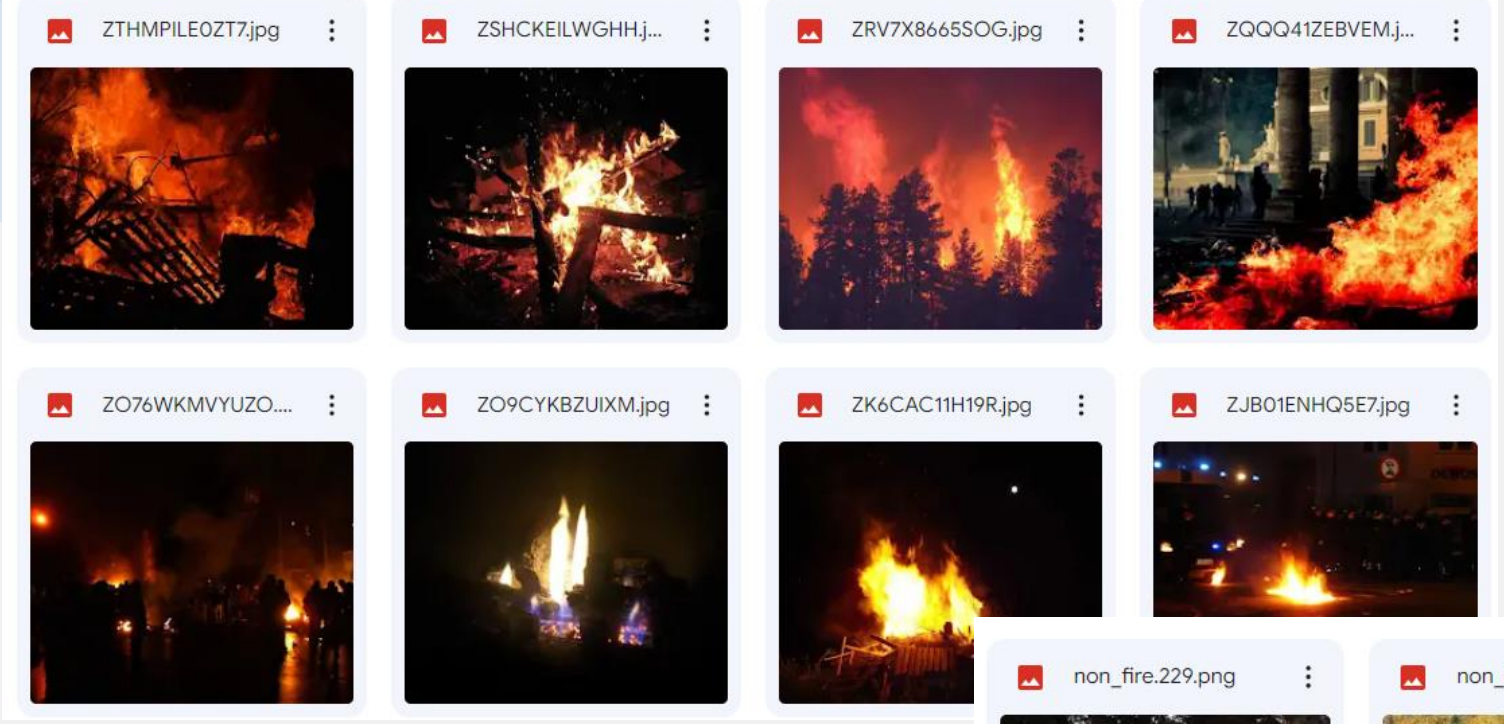
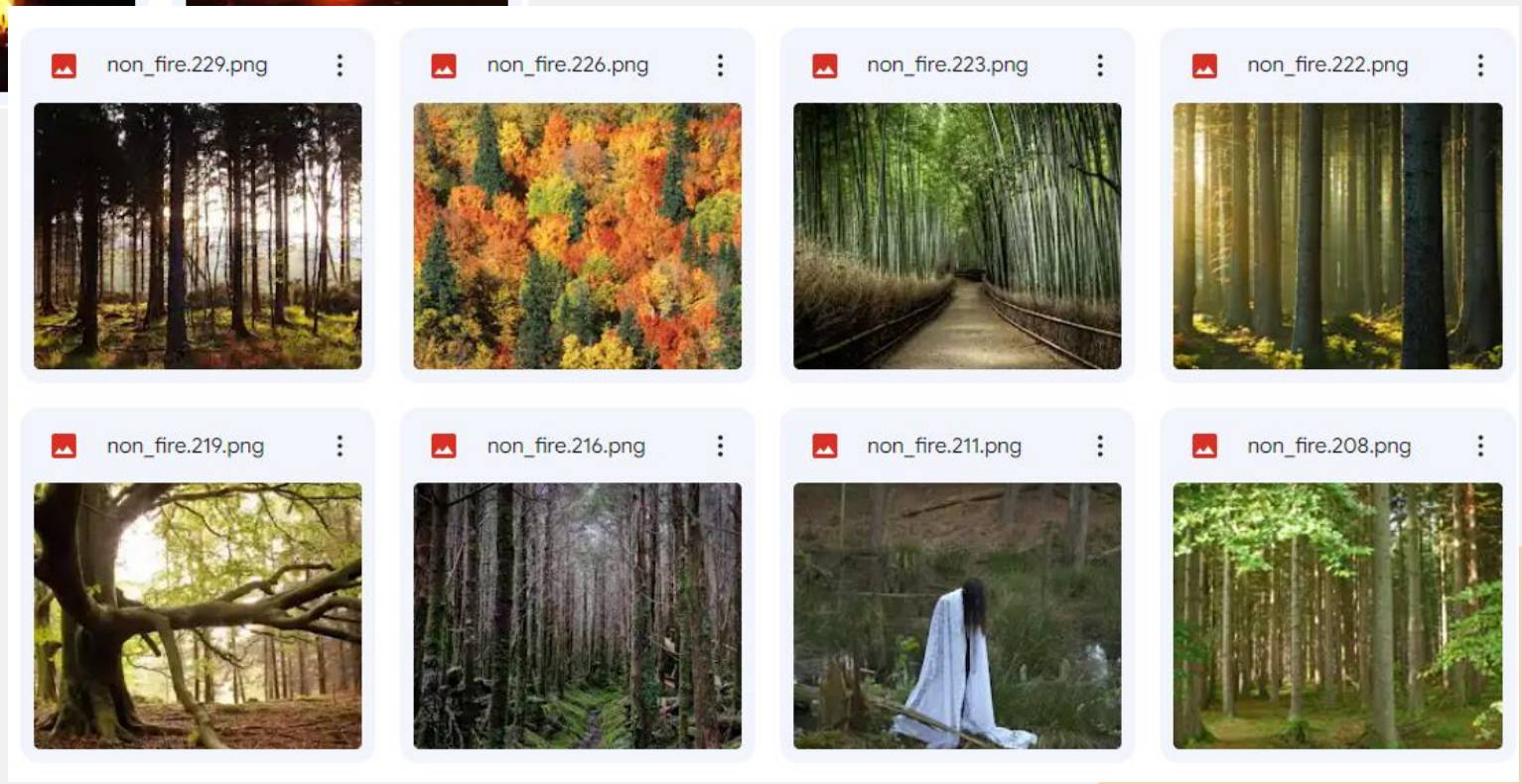
IMAGE FIRE DETECTION USING CNN CLASSIFICATION

Gizem Aleyna Tuzcu – 2200356816

Hasan Malkoç - 2200356826

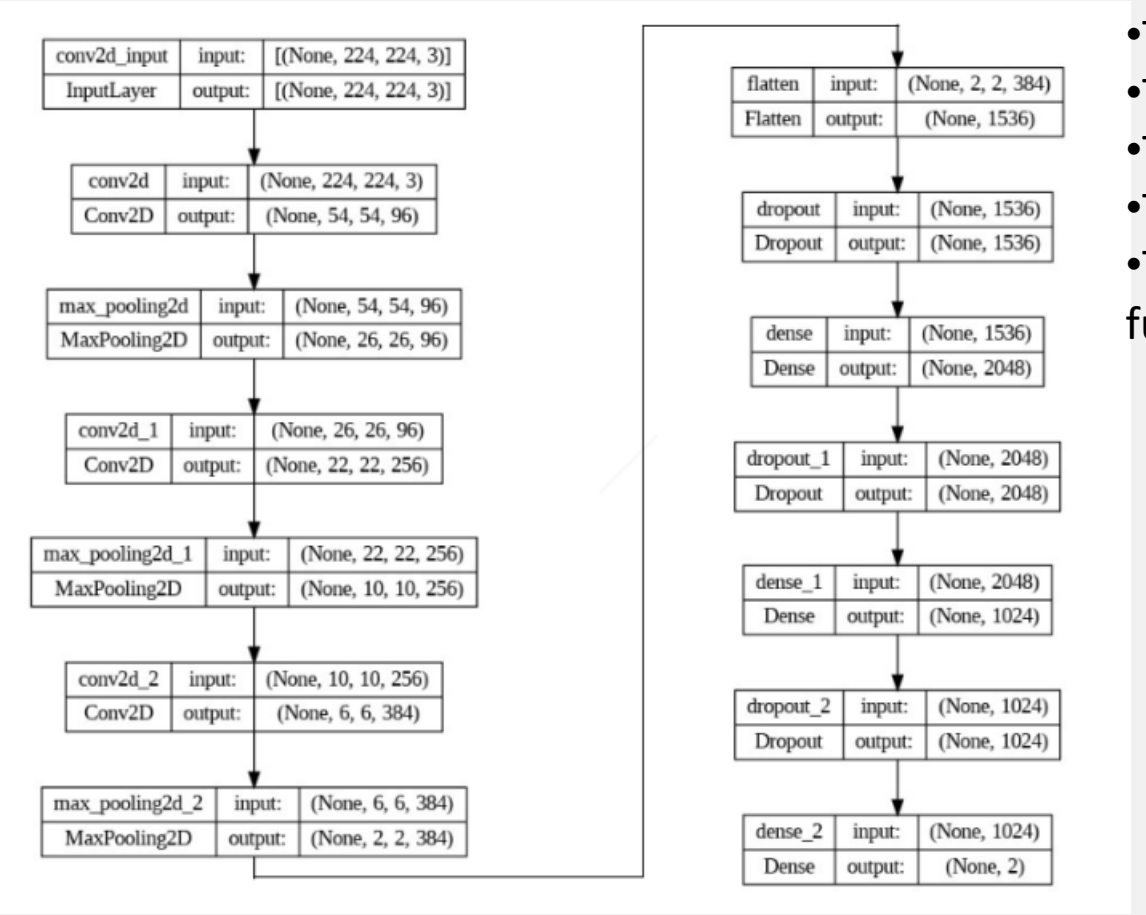


There are 2 labeled image classes «Fire» and «No Fire»



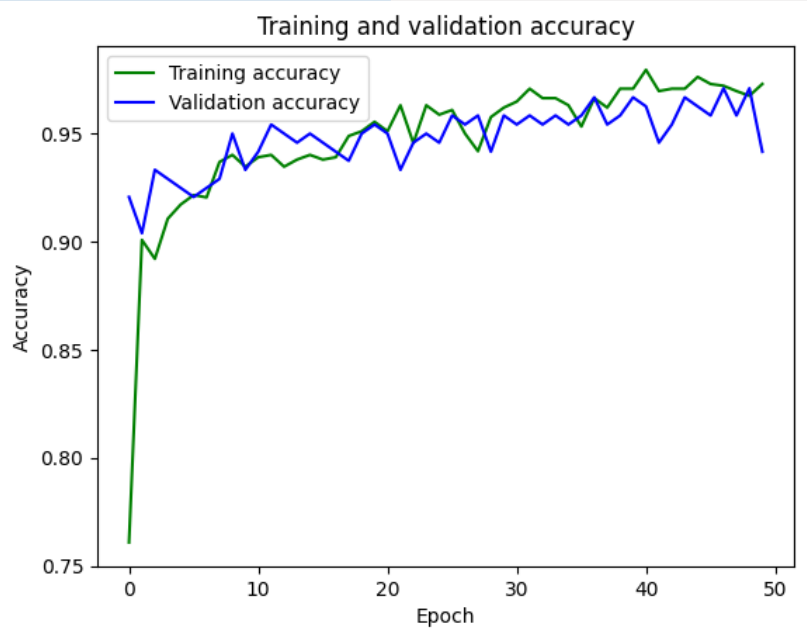
These images are split into Train Test and Validation with 70%, 15%, and 15% proportions

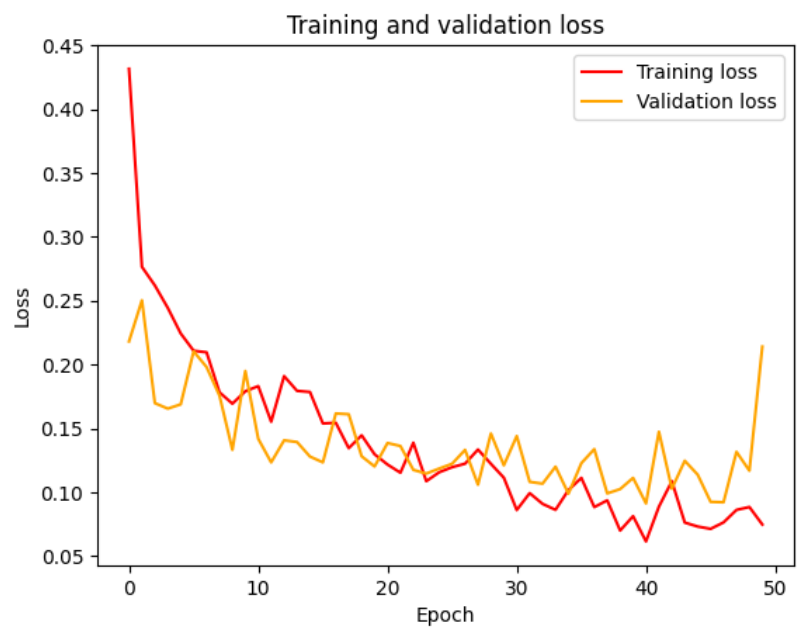
CNN Architecture



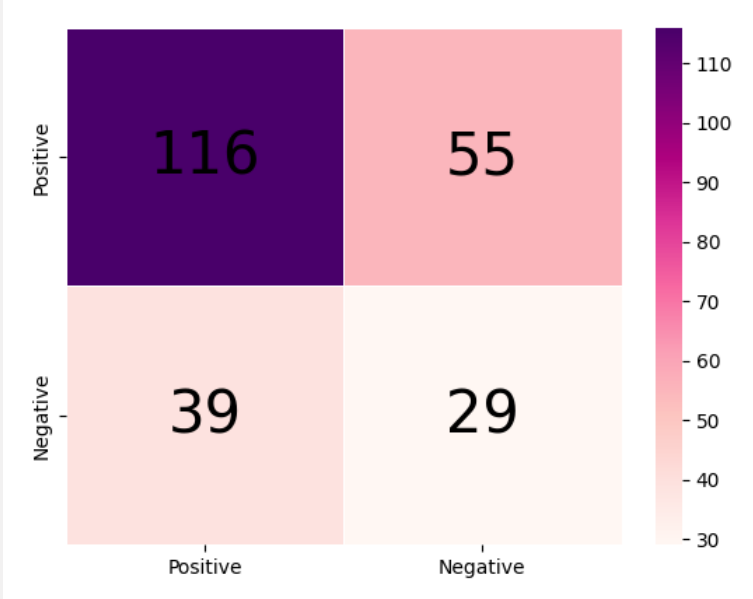
- The input shape is (224, 224, 3)
- The dropout layers have a dropout rate of 0.2, 0.25, and 0.2 respectively.
- The first dense layer has 2048 units with a ReLU activation function.
- The second dense layer has 1024 units with a ReLU activation function.
- The final dense layer has 2 units with a softmax activation function for binary classification.

Batch Size: 32
Epoch: 50
Optimizer: Adam
Learning Rate: 0.0001



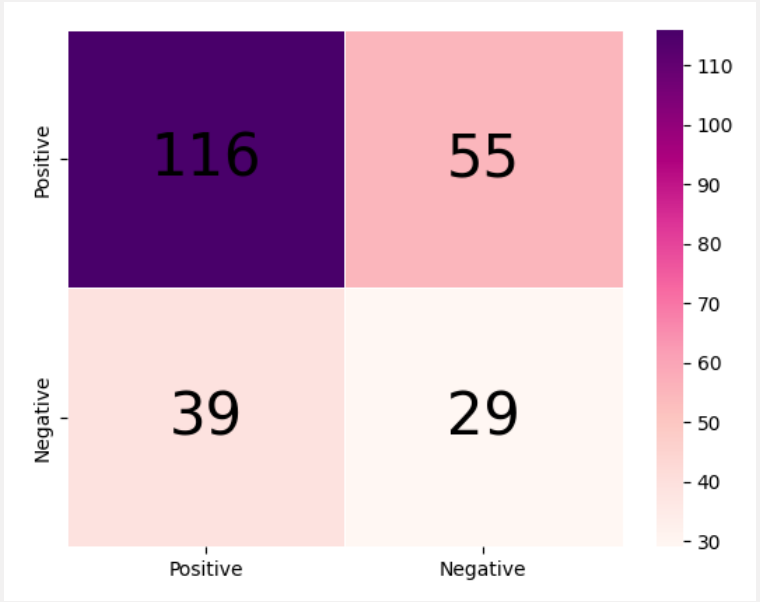


Then applied test images to the model.
Accuracy on fire: 61%



Classification Report

	precision	recall	f1-score	support
0	0.75	0.68	0.71	171
1	0.35	0.43	0.38	68
accuracy	0.61	239		
macro	avg	0.55	0.55	0.55
weighted	avg	0.63	0.61	0.62



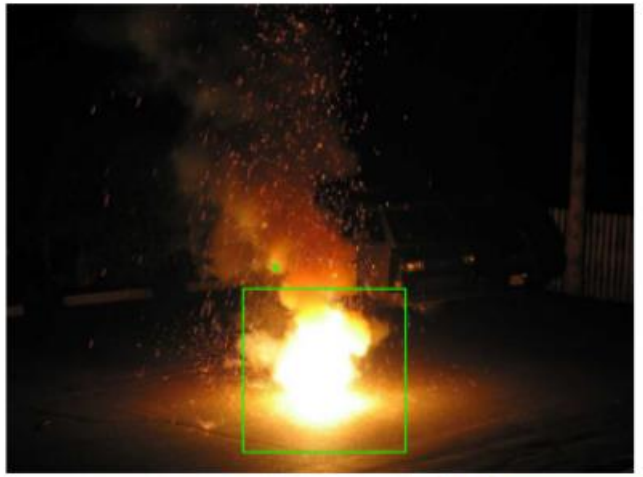
- The model seems to perform better in terms of precision, recall, and F1-score for "Fire" compared to "No Fire". The reason for this result can be the inequal distrubition of the images in the classes. ("No Fire" class has significantly less images than the "Fire" class, almost the third of it.)

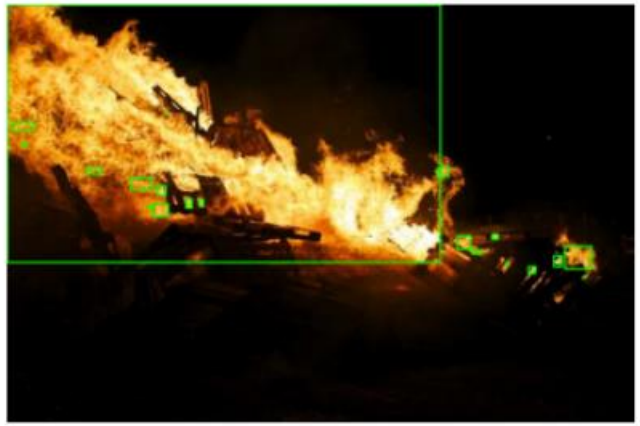
Classification On Images

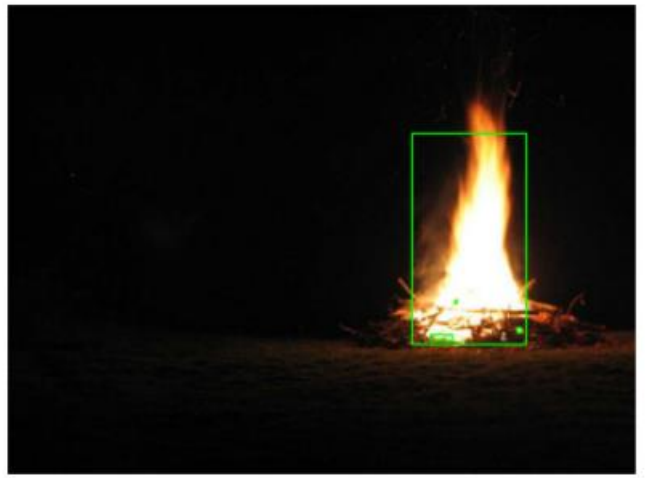
For each image, the model predicted whether it contained fire or not with the percent of the correctness.

The misprediction of an image (third image above) labeled as fire could be attributed to its predominant red and orange colors, which may resemble the visual characteristics typically associated with fire in the model's training data.

Detecting Fire







- Gaussian blur: to reduce noise
- Color Space Conversion: BGR to HSV
  HSV space represent fire colors better
- Thresholding: To range desired colorspace
- Mask Application: To get potential fire regions
- Contouring and Drawing Rectangle