```
# Import necessary libraries
import cv2
import numpy as np

# Load the video file
cap = cv2.VideoCapture('forest_fire_video.mp4')

# Define the color range for the fire
lower_red = np.array([0, 50, 50])
upper_red = np.array([10, 255, 255])

# Define the kernel for erosion and dilation
kernel = np.ones((5, 5), np.uint8)

while True:
    # Read a frame from the video
    ret, frame = cap.read()

    # Convert the frame to HSV color space
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

    # Apply a mask to extract the fire pixels
    mask = cv2.inRange(hsv, lower_red, upper_red)

    # Apply erosion and dilation to remove noise
    mask = cv2.erode(mask, kernel, iterations=1)
    mask = cv2.dilate(mask, kernel, iterations=1)

    # Find the contours of the fire
    contours, hierarchy = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

    # Draw a bounding box around the fire
    for contour in contours:
        x, y, w, h = cv2.boundingRect(contour)
        cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 0, 255), 2)

    # Display the frame with the bounding box
    cv2.imshow('frame', frame)

    # Exit if the user presses 'q'
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release the video file and close all windows
cap.release()
cv2.destroyAllWindows()
```