# **FireNet**: A Specialized **LightWeight** Fire & Smoke **Detection Model** For Real-Time **IoT Applications**

Angel Ayala <aaam@ecomp.poli.br>

# Motivation

- Fires cause a huge losses and can occurs a numerous times.
- A precise, fast and portable solution is required.
- The current Deep Learning methods for computer vision have a poor trade-off between performance and model size.
- A lightweight neural network, trained from scratch, could be deployable in a Raspberry Pi.

# Related Work

- Handcraft feature extraction
    - Motion
    - Color
    - Present problems with fire-like color objects
- Deep Learning Approach
    - Fine-tuned DLNN (AlexNet, GoogleNet, VGG16, ResNet)
    - Large on-disk size

# Dataset

**The training dataset** is sampled from other datasets used in **previous works** and complemented with images obtained from **Google Images** and **Flickr**.
Its compile **1124 images with fire** and, **1301 with no fire.**

**Testing dataset** is composed of:
- 46 videos **with fire (19064 frames)**
- 16 videos **(6747 frames)** and **160 images with no fire**

and are sorted randomly

Fig. 1. Few images from our training dataset.

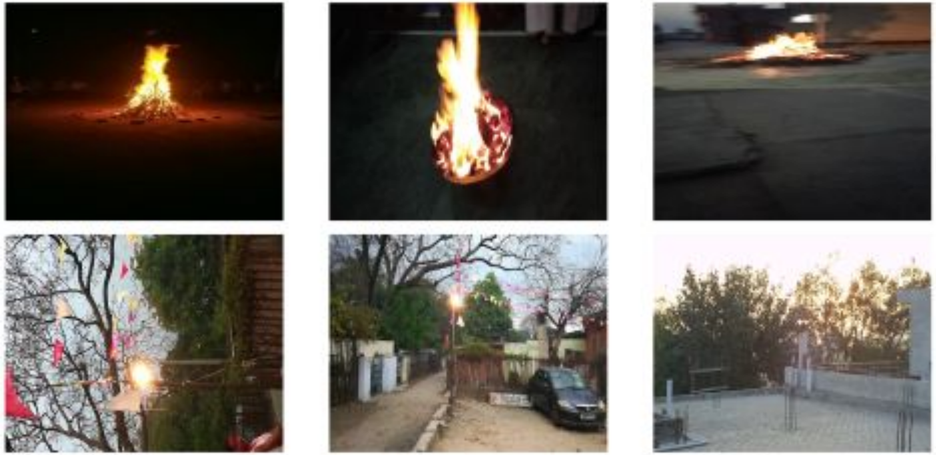

Fig. 2. Few images from our test dataset.

# Proposed approach

System suitable as an **indoor fire detection unit**.

**Lightweight Neural Network** with 646818 params and on-disk size ~ 7.45MB

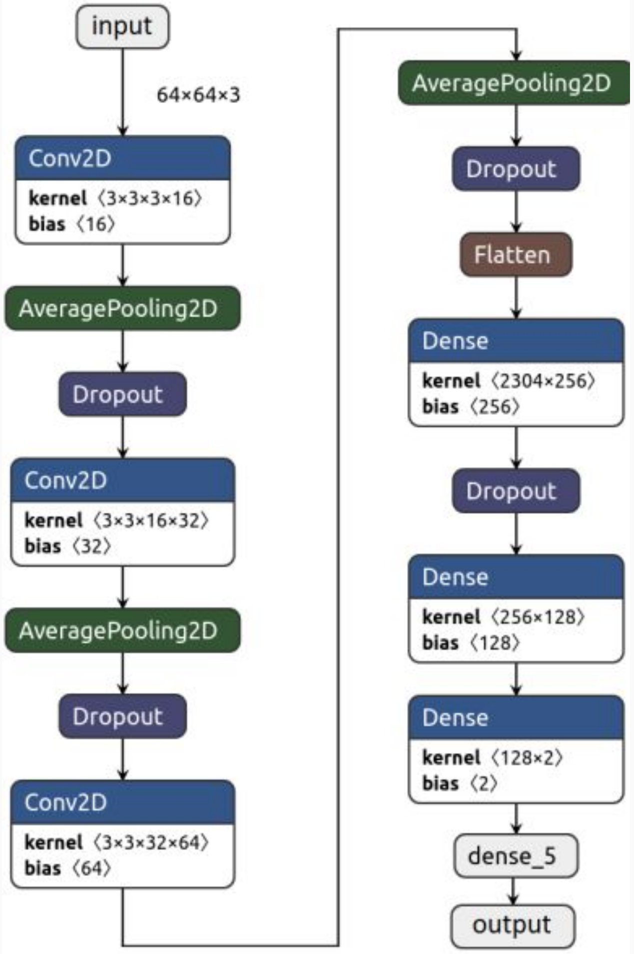


Fig. 3. Architecture of the proposed neural network.

# Proposed approach

**Raspberry Pi 3B unit** for fire and smoke detection

**Twilio** for SMS/MMS Fire detection **notification**

Amazon Web Service's Simple Storage Service **(AWS S3)** for images or clip **storage when the alarm is triggered**

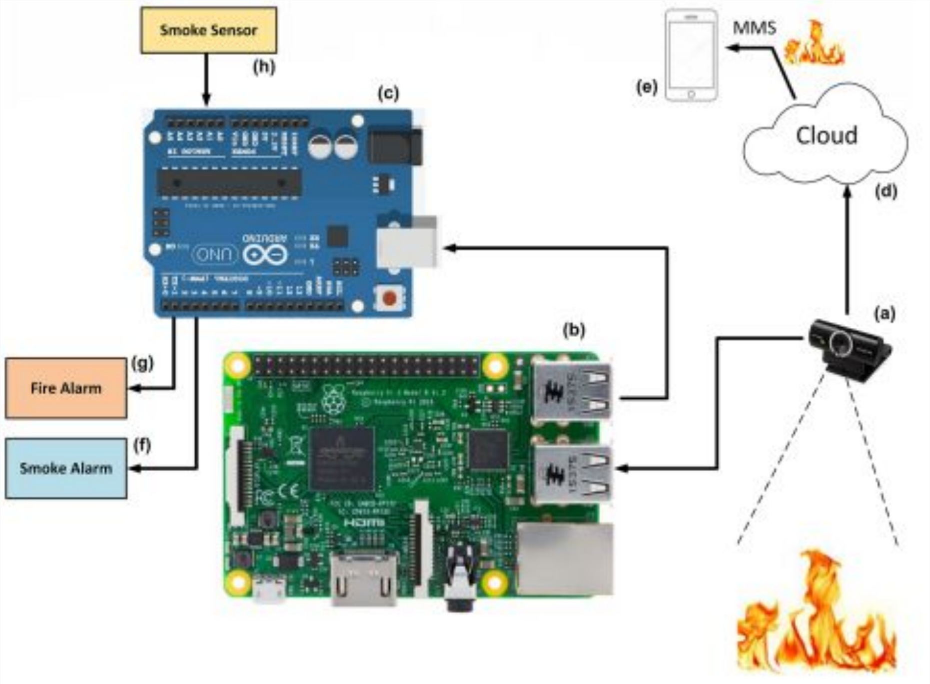


Fig. 4. Overview of the complete unit: (a) Camera (b) Raspberry Pi 3B (c) Microcontroller (d) Cloud storage and SMS/MMS service (Amazon S3 and Twilio) (e) End-user device for receiving fire alert (visual and textual) (f) Smoke alarm (g) Fire alarm with a different sound than smoke alarm (h) Smoke sensor for sensing smoke and thus, aiding in fire-smoke differentiation.

# Results

From the dataset, **70% was used for training** and **30% for validation**.

**The results** obtained were performed against a **dataset used in previous work** and against the **compiled testing dataset**.

TABLE I
TEST PERFORMANCE OF 'FIRENET'ON OUR REAL-WORLD TEST DATASET

| Metrices | Our dataset (%) | Foggia's dataset (%) |
|---|---|---|
| Accuracy | **93.91** | **96.53** |
| False Positives | 1.95 | 1.23 |
| False Negatives | 4.13 | 2.25 |
| Recall | 94 | 97.46 |
| Precision | 97 | 95.54 |
| F-measure | 95 | 96.49 |

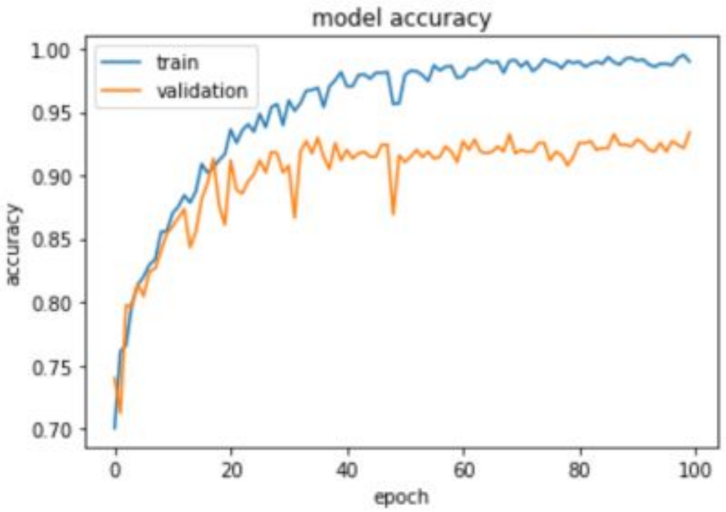


Fig. 5. Training and validation curves for model accuracy.

# Problems found

- It doesn’t present any **metrics for computational cost**.

**- Mislabeled images** on the training dataset result in **bad response** for the output classification.

# Proposed work

- Choose a better dataset for fire classification.
- Improve the Neural Network lightweight model.
- Add at least one computational cost measure.
- (optional) Extent the work to fire detection with a bounding box.

# Thanks for your attention

# **FireNet**: A Specialized **LightWeight** Fire & Smoke **Detection Model** For Real-Time **IoT Applications**

Angel Ayala <aaam@ecomp.poli.br>