Superpixels Segmentation and Interpretable Fuzzy Models for Fire Data Annotation

Pedro Manuel Serro Messias

Thesis to obtain the Master of Science Degree in

Mechanical Engineering

Supervisors:	Prof. Alexandra Bento Moutinho
	MSc. Maria João Santos Lopes de Sousa

Examination Committee

Chairperson: Prof. Carlos Baptista Cardeira
Supervisor: Prof. Alexandra Bento Moutinho
Member of the Committee: Prof. Ricardo Adriano Ribeiro

January 2021

Dedicated to my dear mother

Acknowledgments

The development of this thesis was not possible without the help of some people, who always showed me kindness and support throughout this work. First of all, I would like to thank my two supervisors, professor Alexandra Bento Moutinho and engineer Maria João Sousa, for their guidance, availability, invaluable feedback and, above all, patience, that helped me push through and reach my goals. Their advices were very important for me to stay on the right track towards a good final product.

Secondly, I would like to thank all of my friends for always being there to motivate me and keep pushing through, especially during the quarantine times. A special thank you to my friend Carolina Gonçalves Guerra for doing video calls and online working sessions specially during the period of confinement due to the pandemic. A thank you to my friend Sofia Gonçalves Carreira for organizing a retreat to her parents summer house and allowing me and my friends to relax, spend some time together and work, especially after all those quarantine months. A thank you to my friends Vasco Miguel Campos da Gama Teles Cêpeda, Micael Melo Coelho and Miguel Dias dos Santos for helping proof-read and discuss topics related to this thesis. Furthermore, I would also like to thank my friends Miguel Marques Carreira and Yang Jia Jia. Without my dear group of friends that I had the pleasure to meet during my degree in Instituto Superior Técnico I would not be here today and have one great university experience that I will cherish forever.

Last but not least, I would like to thank my family, especially my mother, for always being there for me in the hardest times throughout my university years and this master thesis.

Resumo

Os incêndios florestais são catástrofes naturais bastante imprevisíveis que conseguem queimar grandes áreas florestais e destruir propriedades. A sua prevenção e deteção permitem diminuir a possibilidade de danos e a rápida mobilização de equipas de emergência. Deste modo, sistemas automáticos capazes de detetar fogos são cada vez mais importantes. O desenvolvimento deste tipo de técnicas necessita de um elevado número de dados de forma a garantirem bons resultados e serem fiáveis em cenários reais. No entanto, o número reduzido e a má qualidade de bases de dados disponíveis na literatura, e a falta de anotações dos mesmos impedem o desenvolvimento de técnicas automáticas. O objetivo deste trabalho, inserido no projeto Eye in the Sky (https://adai.pt/eyeinthesky/), é propor uma arquitetura baseada em segmentação e em modelos linguísticos interpretáveis capazes de gerar anotações relacionadas com incêndios florestais. O método tira proveito das características ricas das cores do fogo em duas etapas diferentes, segmentação e classificação. A primeira está relacionada com gerar os superpixels e agregá-los em regiões com base nas cores do fogo no espaço de cores de YCbCr. Subsequentemente, a classificação de cada região é realizada através de modelos interpretáveis baseados nos espaços de cores HSL e YCbCr, os quais permitem gerar uma segmentação do fogo e anotações semânticas das cores do fogo. Para além disso, este método permite ajustar certos parâmetros de forma a melhorar os resultados. A técnica proposta é avaliada em diferentes contextos reais através de uma base de dados disponível publicamente.

Palavras-chave: deteção de fogo, espaço de cores HSL, espaço de cores YCbCr, superpíxel, modelos linguísticos interpretáveis, anotações de dados de fogo

Abstract

Wildfires are natural disasters that can be quite unpredictable, burning large areas of forests and destroying properties. Fire detection and early prevention enable a faster reaction from emergency teams and to decrease the possibility of fire damage. Therefore, automatic systems capable of detecting fires are increasingly important. Their development requires a high number of data in order to guarantee good performances and be reliable in real scenarios. However, the low number and poor quality of available datasets in the literature, and the lack of annotations hamper the development of such automatic techniques. The objective of this work, developed in the framework of project Eye in the Sky (https://adai.pt/eyeinthesky/), is to propose an architecture based on segmentation and interpretable linguistic models capable of generating wildfire annotations. The proposed approach takes advantage of rich color features representative of fire in two different stages, segmentation and classification. The first one is related to generating superpixels and aggregating these into regions based on the representation of fire colors in the YCbCr color space. Subsequently, the classification of each region is achieved using interpretable rule-base models based on the HSL and YCbCr color spaces, which generates a pixel-wise fire segmentation and the semantic annotations of the fire colors. Furthermore, this method allows certain fine-tunable parameters in order to improve its overall results. The proposed approach is evaluated in different real contexts using a publicly available database.

Keywords: fire detection, HSL color space, YCbCr color space, superpixel, interpretable linguistic models, fire data annotations

Contents

List of Tables

List of Figures

Nomenclature

C_k Cluster center.

D Distance measure to determine the closest cluster center to each pixel.

E Residual error associated with the distance between previous cluster centers and new centers.

H_j^{sp}, S_j^{sp}, L_j^{sp} Mean color of the superpixel of the channels Hue, Saturation and Lightness, respectively.

$H_j^{\mathrm{sp}}, S_j^{\mathrm{sp}}, L_j^{\mathrm{sp}}$ Mean color of the superpixel of the channels hue, saturation and lightness, respectively.

I Sample image.

L, a, b values of CIELAB color space.

N Number of pixels in an image.

N/K Superpixel size.

N_{sp} Number of superpixels.

N_{lab} Maximum color distance in a cluster.

N_{xy} Maximum spatial distance in a cluster.

R_c Classification of each region.

S The distance between each cluster center.

$Y_j^{\mathrm{sp}}, Cb_j^{\mathrm{sp}}, Cr_j^{\mathrm{sp}}$ Mean color of the superpixel of the channels luminance, chrominance blue and chrominance red, respectively.

$\mathcal{C}_{\mathrm{orange}}$ Set of fire pixels displaying orange colors

$\mathcal{C}_{\mathrm{other}}$ Set of fire pixels displaying other colors.

$\mathcal{C}_{\mathrm{red}}$ Set of fire pixels displaying red colors.

$\mathcal{C}_{\mathrm{yellow}}$ Set of fire pixels displaying yellow colors.

$\mathcal{D}$ Color space domain.

$\mathcal{F}$ Set of pixels associated to the fire class.

$\mathcal{N}$ Set of pixels associated to the non-fire class.

$\mu(x)$ Y axis of the membership function.

c Number of channels of the color space.

m Compactness.

sp Mean color of the superpixel.

z_{COA} Output variable of the centroid method in the defuzzification.

Greek symbols

δ Threshold of classification.

Subscripts

i, j, k Computational indexes.

x, y, z Cartesian components.

Superscripts

T Transpose.

Chapter 1

Introduction

Natural disasters like floods, earthquakes and wildfires have a considerable environmental and economic impact. Every year, wildfires destroy hectares of lands, burning forests and villages, leading to the loss of material goods and possibly lives. During the year of 2020, the state of California has registered a significant increase in wildfires, especially when compared to previous years, reporting almost 1700 kha of burned area and more than 30 deaths [1]. In 2018, the same state registered the Ranch Fire (Mendocino Complex) reaching almost 200 kha of burned area [2]. In Portugal, wildfires are very common during the summer and a similar scenario occurred in 2017, where the wildfire of Pedrogão Grande burned around 53 kha and caused more than 60 deaths and 200 injured [3].

In recent years, climate change has been one of the most discussed topics due to the increase of its effects having consequences on the planet Earth. The reduction of greenhouse gases is essential to prevent these consequences. The increase in the average temperature, caused by global warming, can lead to more frequent heat waves and longer dry periods, increasing the risk of wildfire [4].

This chapter provides an overview of the different fire detection techniques present in the literature, followed by the proposed approach in section 1.2. Finally, section 1.3 presents the thesis outline with brief descriptions of the following chapters.

1.1 Fire Detection Topic Overview

Over the years, the increased number of wildfires has raised a need in developing automatic techniques capable of helping in wildfire fighting. Fire detection approaches able to provide earlier fire alerts would enable a faster response from emergency teams and reduce the possibility of fire destruction in large areas and, potentially, material goods. Moreover, further knowledge about the fire in terms of its size and which direction the fire is evolving would reveal to be crucial, specially, in the event of existing urban buildings since it would allow a faster human reaction. Weather forecast can provide to some extent information related to large areas having a higher fire hazard which would enable a better deployment of emergency teams. However, not only are wildfires very unpredictable events but also these weather reports do not give a very precise information about the localization but rather a rough large area.

Accordingly, the scientific community has explored and developed automatic fire event detection techniques to enable faster reaction times and reduce fire destruction. Initially, conventional fire detection approaches were usually based on data collected from ultraviolet or infrared fire sensors. In addition, wireless sensor networks (WSNs) emerge as an alternative to conventional techniques [5, 6]. These consist on a large number of cheaper and smaller sensors able to collect different types of data, e.g., temperature, humidity or carbon monoxide density, and send an alarm whenever a fire is detected. However, those sensors are required to be in close proximity of a fire in order to register reliable data, which can be quite expensive in terms of their deployment and maintenance, specially in large fields [7, 8].

The tecnological advances of cameras in terms of processing capabilities and overall quality of images allowed the progress of computer vision-based techniques. As the name implies, these methods are related to digital images or videos being processed by a machine and utilizing visual features encountered in these type of data, e.g., color, motion or texture, to retrieve important and interesting information. Considering the topic of fire detection, several computer vision approaches take advantage of such characteristics, specially color since fires are usually representative of bright and warm colors.

For these reasons, fire detection approaches are built upon color spaces as these are associated with a crisp representation of colors in different domains. The most well-known one is the RGB color space [9, 10], as it is related to how the human eye perceives color. Chen et al. [9] propose a fire detection technique capable of classifying flame and smoke pixels using the RGB color space and motion features. The latter complements the RGB-based model when it detects similar fire and smoke colors in non-fire objects, which enables the reduction of the number of false alarms. Furthermore, some approaches exploit other color spaces like the HSV [11, 12], for being more interpretable in describing colors, or the YCbCr [13, 14], which allows the separation between luminance (luminosity) and chrominances (color information).

Several classical computer vision techniques utilize visual and motion features of fire and smoke areas to develop better and more reliable classifiers. However, since classifying a fire pixel in terms of colors can be a subjective task, these techniques using rules with fixed fine-tuned values have certain limitations when applied to different scenarios or datasets [15].

Considering the limitations previously outlined with regard to classic computer vision techniques, the scientific community has been applying visual fire attributes (e.g., color or motion features) with intelligent systems methods. Accordingly, one might consider the use of machine learning (ML) or deep learning (DL) techniques, as these have been very recurrent approaches in the last few years [15]. In particular, deep learning methods have been widely used in various applications since they are able to handle high-dimensional data like images or videos [16]. Yin et al. [17] propose a smoke detection method that uses a deep normalization and convolutional neural network (DNCNN) based on ZFNet [18]. The deep neural network allows to extract features from images and classify the smoke dataset. Furthermore, they perform data augmentation to increase the overall number of training samples in the dataset and decrease the possibility of overfitting. In addition to data augmentation techniques, other approaches also incorporate transfer learning methods [19, 20]. The development of computational resources has encouraged the increase in deep learning-based approaches [21]. However, the lack of datasets, not only in terms of quantity but also quality and how relevant and representative their samples are, results in certain limitations in their application to different scenarios and may cause deep learning models to overfit. Moreover, deep learning models are not very interpretable techniques limiting their understanding in terms of the reasons that lead to their results.

Additionally, fire detection approaches use other intelligent system techniques like fuzzy modeling [22–24]. Fuzzy theory can be quite appealing as it allows humans to represent fire colors in a non-crisp way. Considering that humans and computers have different ways of representing colors, where one uses linguistic terms, that are subjective to the person and context, and the other color spaces (vectors), fuzzy models provide a solution to representing color categories using color spaces [25]. This is particularly interesting in classification problems as these models assign a degree of membership instead of the usual binary value, which might enable better interpretations from a user, especially in cases where the results register false positives or false negatives.

The work of Çelik et al. [22] proposes a fire detection technique based on a fuzzy model developed for the YCbCr color space. Taking into consideration that a fire pixel can be represented with a high luminance and chrominance red, and a low chrominance blue, the ability of the YCbCr color space to separate these three attributes is quite useful when compared to other color spaces. Finally, this fuzzy model outputs the possibility of a pixel being a fire pixel. Despite this model registering a detection rate of 99% and a false alarm rate of 4.5%, the available information about the dataset is quite insufficient to fully understand how diverse it is. In fact, there is no information about the number of non-fire images. The authors also propose a smoke detection technique but no evaluation analysis was made.

1.2 Proposed Approach

Wildfires are among the most common natural catastrophes around the world, burning huge areas of land every year and, in the worst case scenario, taking humans lives. Their prevention and early detection are crucial to minimize fire damage, improve the reaction time of firefighters and empower better decisions. For these reasons, fire detection techniques have been proposed by the scientific community as an attempt to create automatic systems.

Many of these fire detection techniques are hampered by the overall quality of datasets and the lack of annotations. The first is related to various limitations concerning the datasets. These limitations are usually associated with the insufficient number of samples, where many can even have some sort of noise, decreasing their overall image quality, or represent different contexts than nature scenery, e.g., fires in buildings or indoor. Moreover, several images come from videos that depending on the capturing rate and the actual video, could be very similar to each other, increasing the number of the dataset but not its significance. Finally, the lack of dataset annotations, i.e., brief key indications that describe different real contexts (e.g., images with fire at long-distance, with firetrucks or with sunsets, which display similar colors to fire) or other valuable features, can be a hindrance for a better comprehension of the existent models in the scientific community, specially in terms of image characteristics that these have been developed and tested on, and in cases of misclassifications.

To address the limitations that datasets face, this work tackles the development of a method for fire data annotations through semantic segmentation, instead of the more common fire detections approaches. In fact, the proposed approach presents a method to generate fire segmentations and labels describing fire colors, which can then be validated by experts related to wildfires to create reliable ground truth data.

At a first stage, the proposed architecture relies on the rich color features representative of fire, namely in the HSL and YCbCr color spaces, which allow an insightful interpretation that, in return, enables the color-based superpixel segmentation. Afterwards, interpretable rule-based linguistic models are employed to classify superpixels in terms of their color attributes to infer which correspond to the fire or non-fire classes. Moreover, these models allow to generate semantic labels describing the fire colors present in the image [26].

The objectives of the proposed fire data annotations approach are:

- Obtain a pixel-wise segmentation of the flame.
- Describe the colors of fire using color categories.

The proposed approach is evaluated against a subset of the Corsican Fire Database, demonstrating excellent segmentation results and their ability to handle a variety of real-contexts, e.g., with fire at long-distances, with firefighters or firetrucks, and in smoke situations. Finally, we demonstrate the different limitations that the several proposed models face and this approach's ability to allow experts to intuitively fine-tune the model output or adjust some parameters, e.g., threshold or number of superpixels, in order to improve the final segmentation and ensure the expert confidence during the annotation process.

The main contributions of the proposed work are:

- Development of a fire data annotation architecture that allows to generate reliable fire segmentations and semantic labels descriptive of the represented colors.
- Implementation of interpretable linguistic models for flame pixel classification, enabling a better comprehensibility and transparency.
- Create datasets with high quality and quantity samples with relevant annotations related to fire and smoke features.

1.3 Thesis Outline

Following this first chapter, this thesis is organized in 6 chapters including the present one that addresses the motivation and objectives of this work.

Chapter 2 introduces all the theoretical concepts of the techniques applied throughout this work. First, with a brief explanation about the different color spaces that the proposed approach utilizes to create the fuzzy models, followed by the superpixels principles and properties, mentioning different algorithms and describing the SLIC superpixel algorithm used for image segmentation in this thesis.

Finally, an explanation of fuzzy logic theory with several examples to demonstrate how fuzzy inference systems works.

Chapter 3 addresses the importance of databases, namely having descriptive annotations and ground truth data, and the different databases used to develop and test the proposed approach.

Chapter 4 delves into the methodology applied to develop the proposed technique. Firstly, the reasons behind choosing the HSL and YCbCr color spaces. Secondly, the procedure to generate and merge the superpixels to reach a close flame-only segmentation, and lastly, the description of the two fuzzy models created using the HSL and YCbCr color features in terms of memberships functions, parameters and rules.

Chapter 5 presents the final performance results, demonstrating the capabilities of the developed method with a set of different examples. Initially, describes the dataset and the performance metrics used to evaluate all the different models. Then, presents all the results and reasons to develop other solutions. And lastly, the limitations and adaptability of the best model solution.

Chapter 6 reasserts all the achievements and conclusions, closing with future work ideas.

Chapter 2

Theoretical Background

This chapter provides the theoretical principles used in the proposed approach. Section 2.1 gives a brief overview of the color spaces HSL and YCbCr and their features since these are used to create the color-based fire segmentation and the interpretable rule-based fuzzy models. Section 2.2 explains the properties and different types of superpixels to then dive into the SLIC superpixel algorithm used to generate superpixels. Finally, section 2.3 describes the fuzzy logic theory, more specifically, how a fuzzy inference system takes an input and outputs a value, using fuzzy sets, membership functions and If-Then rules.

2.1 Color Spaces

As humans and computers have different perceptions of color, color spaces allow the latter to represent colors in different geometric representations. There are many different color spaces defined in $\mathbb{R}^c$, where c represents the dimensions, i.e., the number of channels. The most common ones are usually defined in $\mathbb{R}^3$, composed of three channels.

RGB color space

The most common color space is the RGB color space and it is composed of three channels, R (red), G (green) and B (blue). This color space is very similar to how the human eye perceives color. During the nineteen century, the works of Young and Helmholtz gave birth to the thrichromacy theory [27, 28]. This hypothesis, also known as the Young–Helmholtz theory, declares that the human eye has three types of photoreceptors (cone cells) that are sensitive to how the light hits the retina. As the light hits the retina, the different wavelenghts can be classified into short (blue), middle (green) and long (red). In essence, this theory assumes that any color can be represented by combining three primary colors: red, green and blue. Similarly, the RGB color space exhibits any color by joining different amounts of red, green and blue.

Figure 2.1 outlines the RGB color space, displaying different colors. For instance, the black color corresponds to the origin (Red = Green = Blue = 0) and the white color to the maximum value for all three values (Red = Green = Blue = 255). The range for the RGB color space is [0, 255].

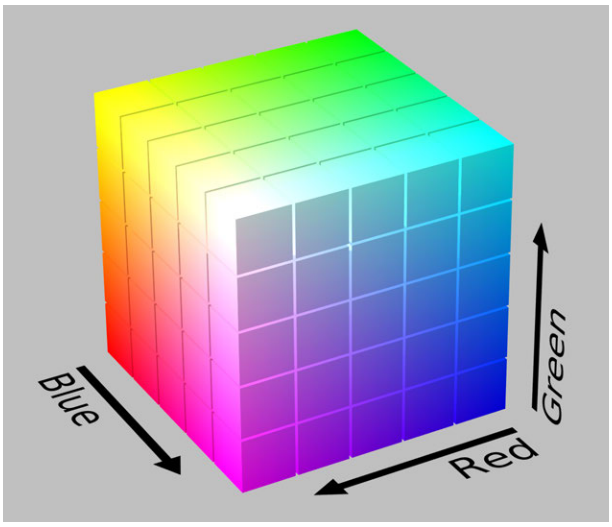


Figure 2.1: RGB color space.[1]

HSL color space

However, the RGB color space is not very suitable for human perception when compared to other color spaces, since it is hard to characterize a color pixel in terms of amount of red, green and blue. Similarly, the HSL color space portrays colors in terms of hue (H), saturation (S) and lightness (L), which is much more perceptive for Humans to comprehend [29].

Hue is related to the dominant wavelength perceived by an observer. It is the attribute of a color, as red, yellow, blue, etc., independent of intensity. Its range is [0, 360^o] [30]. Saturation is associated to the colorfulness of an area relative to its own brightness and it ranges from 0 to 100%. Lightness is related to the brightness of an area relative to the brightness of a similarly illuminated white [30]. For instance, the color black results from a lightness equal to 0 whilst the color white results from a lightness equal to 1. It has the same range as saturation, [0, 100%].

Figure 2.2 displays cylinder representative of the HSL color space. This figure exhibits how each channel, hue, saturation and lightness, varies. Note that this color space is used in the proposed approach because of its simple ability to characterize the different fire colors, specifically, in terms of lightness and saturation, when compared with similar color spaces, for instance, the HSV color space.

YCbCr color space

The YCbCr color space is characterized by the luminance (Y) and the chrominances blue (Cb) and red (Cr). Compared to previous color spaces, the YCbCr is particularly interesting as it allows to separate the luminance (luminosity) from the chrominances (color information) [13]. The luminance is associated with the brightness of the image, whilst chrominances are related to the difference between the blue or red component and a reference value [31].

[1] Image downloaded from wikimedia commons RGB: `https://commons.wikimedia.org/wiki/File:RGB_Cube_Show_lowgamma_cutout_a.png`

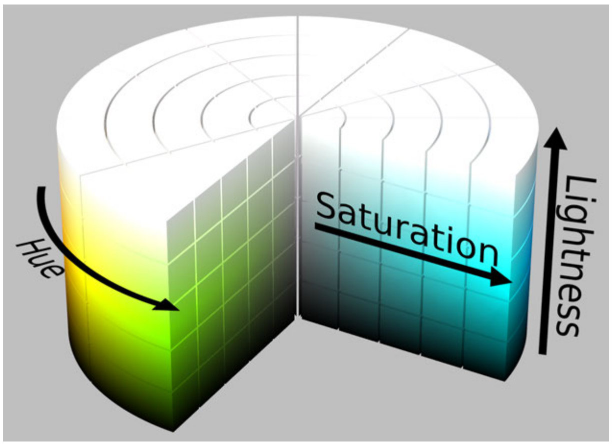


Figure 2.2: HSL color space.[2]

The following equation (2.1) enables the conversion of each color pixel from the RGB color space to the YCbCr color space [31]. Note that this equation converts the RGB image in the range [0, 1] to the YCbCr image in the range where Y is [16, 235], and Cb and Cr are both [16, 240]..

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \tag{2.1}$$

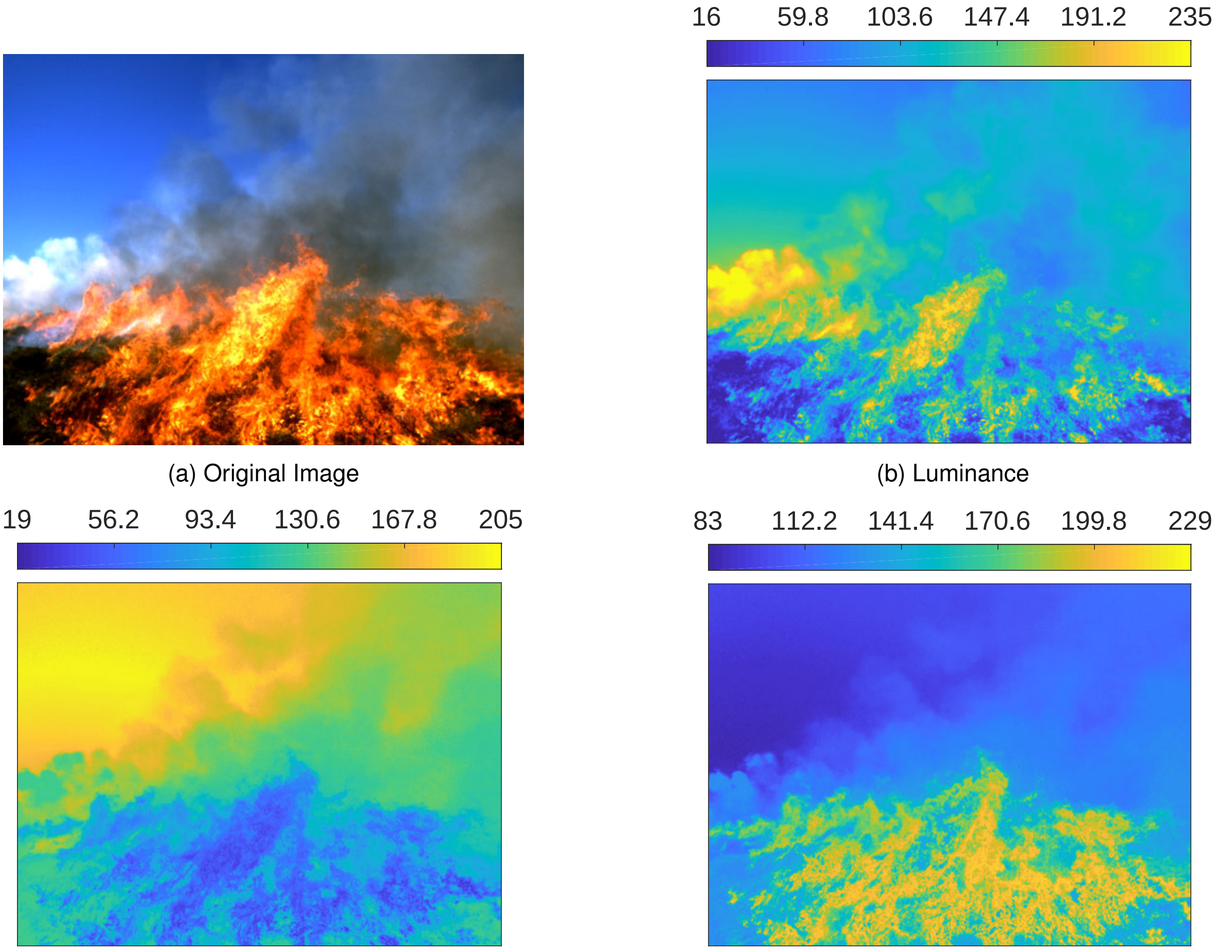
As there is not a simple image to represent the YCbCr color space like the previous color spaces, Fig. 2.3 displays a fire image and each channel to demonstrate the variations of this color space. As expected, the images of chrominances blue (Fig. 2.3c) and red (Fig. 2.3d) exhibit high values in regions in accordance to blue (sky) and red (flame) colors, respectively. In addition, the luminance image (Fig. 2.3b) depicts high values in areas of great luminosity, white clouds and bright yellow regions in the flame.

2.2 Superpixels

Image segmentation techniques are related to the partitioning of images into multiple regions which might enable a more insightful representation analysis depending of the method used. Image segmentation is usually used as an intermediate step in several techniques as these allow improvements in computer vision-based approaches.

Superpixel algorithms are also considered an image segmentation technique as their objective is to group pixels together based on their characteristics, thus resulting in an image oversegmentation. This technique is quite interesting as superpixels allow to decrease the difficulty in later operations [33]. In addition, it also reduces the processing time since the number of superpixels is much lower than the number of pixels. The *superpixel* term was first introduced by Ren and Malik [34] by defining the superpixel map and applying the normalized cuts (NCuts) algorithm [35].

[2] Image downloaded from wikimedia commons HSL: `https://commons.wikimedia.org/wiki/File:HSL_color_solid_cylinder_saturation_gray.png`



(a) Original Image

(b) Luminance

(c) Chrominance Blue

(d) Chrominance Red

Figure 2.3: Sample to represent each channel from the YCbCr color. Original image from the Corsican Fire Database [32]

According to Achanta et al. [36], the ideal superpixel approach is usually dependent of a specific application. In other words, many can be well-suited to certain situations and, at the same time, display disadvantages in others. Therefore, it can be hard to specify what establishes an ideal method to generate superpixels. Neverless, the works of Achanta et al. [36] and Wang et al. [37] present the following properties as the most essential and suitable:

1. superpixels should adhere well to image boundaries;
2. ideally, each superpixel should not overlap with multiple objects and should assign a number to every pixel within the superpixel;
3. in a pre-processing phase, superpixels should be efficient and simple to use in order to decrease the need of computational power demand for the overall procedure.

Considering these properties, there are a number of different superpixels algorithms throughout the literature. Based on the classification employed by Achanta et al. [36], these algorithms can usually be divided into two groups, depending on their iterative process. Therefore, they classified the techniques as graph-based or gradient-ascent-based.

Graph-based algorithms

Graph-based methods [35, 38], as the name implies, assign every pixel to a vertex in a graph and links these through edges. The weight of each edge is based on how similar neighboring connected pixels are. The algorithm is optimized by minimizing a cost function concerning the edge weights in order to merge pixels with similar edge weights, generating superpixels. Some algorithms included in this category are the Normalized Cuts (NCuts) [35] and Lazy Random Walks Superpixel [38].

Gradient-ascent-based algorithms

Gradient-ascent methods [36, 39–41] optimize the process of joining pixels into clusters until a certain point where a convergence value is met. Algorithms included in this category are Simple Linear Iterative Clustering (SLIC) [36], Watershed [39], MeanShift [40] and Linear Spectral Clustering (LSC) [41].

2.2.1 Simple Linear Iterative Clustering

The simple linear iterative clustering (SLIC) [36] is a segmentation algorithm based on K-means clustering that creates superpixels within a five-dimensional $[labxy]$ space defined by the L, a, b values of CIELAB color space and the x, y pixel coordinates. In essence, the superpixels are created by grouping pixels that are similar in color and fairly close to each other.

We chose this segmentation algorithm because it is relatively fast to compute and memory efficient when compared to other methods [36]. It is quite easy to use as it only needs two mandatory inputs, the actual image and the number of superpixels to generate. In addition, the SLIC algorithm also allows the change of the superpixels' compactness, which is related with the superpixel overall shape (explained in more detail in chapter 4).

Algorithm description

This approach can be divided into four steps: initialization, assignment, update and post-processing.

The algorithm begins by initializing K cluster centers that are spaced S pixels apart, where K is the desired number of superpixels and S the distance between each cluster center. The clusters centers are then moved to the lowest gradient position in a 3x3 area to prevent creating a center on an image boundary and decrease the chances of choosing a pixel with noise.

In the assignment step, each pixel is linked to the closest cluster center in a search region of 2$S$x2$S$, as depicted in Fig. 2.4. In other words, it calculates the distances between a pixel and every cluster center in a 2$S$x2$S$ area, as supposed to the entire image, assigning the pixel to the nearest cluster center. In fact, this means that the number of distance calculations is substantially lower when compared to conventional k-means clustering where a pixel is compared to all cluster centers within the image, which results in a faster algorithm. For these reasons, this approach requires a distance measure, D, to calculate the closest cluster center for each pixel.

Considering an image with N pixels, each superpixel size is approximately N/K pixels. In order to create fairly identical superpixels, each cluster center is defined at $S = \sqrt{\frac{N}{K}}$. The euclidean distance can not be used within pixels represented in the 5-D $[labxy]$ as it would be dependent on the number of superpixels. A lower number results in larger superpixels where the distance between their centers would outweigh their color similarity, resulting in superpixels that do not retain the image boundaries. Therefore, Achanta et al. [36] use a new distance measure D (equation (2.4)) to allow a nearly equal weigh between the color similarity and distance proximity.

This distance D combines both the color distance (lab) and the spatial distance (xy), where N_{lab} and N_{xy} are their maximum distances in a cluster, respectively, thus resulting in:

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}$$

$$D' = \sqrt{\left(\frac{d_{lab}}{N_{lab}}\right)^2 + \left(\frac{d_{xy}}{N_{xy}}\right)^2} \tag{2.2}$$

where $[l_i, a_i, b_i, x_i, y_i]^\top$ is the pixel i represented in the 5-D space and $C_k = [l_k, a_k, b_k, x_k, y_k]^\top$ the cluster center.

Then, the maximum spatial distance in a cluster N_{xy} is considered equal to S, resulting in $N_{xy} = \sqrt{N/K}$ and, to simplify the process as color distances can be very diverse, N_{lab} is considered equal to a constant m, resulting in the following equation:

$$D' = \sqrt{\left(\frac{d_{lab}}{m}\right)^2 + \left(\frac{d_{xy}}{S}\right)^2} \tag{2.3}$$

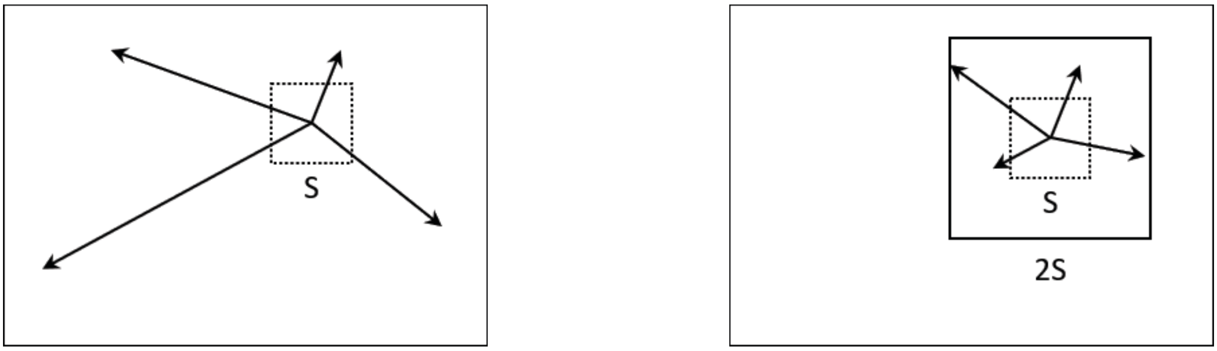
Finally, the distance measure D is defined as:

$$D = \sqrt{{d_{lab}}^2 + \left(\frac{d_{xy}}{S}\right)^2 m^2} \tag{2.4}$$

This variable m allows us to control the compactness of a superpixel, i.e., control the weigh between color similarity and spatial proximity. For a large m value, spatial proximity outweighs color similarity, where the superpixel is more regularly shaped. This variable may take values in the range [1, 20].

After all pixels have been assigned to a cluster center, the update step allocates the mean value of all pixels within a cluster to its center. Finally, the assignment and update steps are repeated iteratively until the residual error E stabilizes, where this error E is associated with the distance between previous cluster centers and new centers.

After the clustering procedure, some pixels may not belong to the same superpixel as their cluster center because this algorithm does not enforce connectivity. Therefore, a post-processing step enforces connectivity and reassigns those pixels to the nearest cluster center using a connected components algorithm.



(a) K-means clustering method searches the entire image

(b) SLIC method searches a limited region

Figure 2.4: Comparing conventional K-means with SLIC. (a) The K-means method computes distances from each cluster center to every pixel in the image, whilst, (b) in SLIC, distances are computed from each cluster center to pixels within a 2$S$x2$S$ area. The smaller dashed square, SxS, represents the expected superpixel size. Images adapted from [36].

2.3 Fuzzy Theory

In everyday life, Humans are faced with situations where the decision-making is very subjective and susceptible to different interpretations from each person thus making it almost impossible to achieve a precise solution. Several complex problems can not be represented using traditional methodologies due to their imprecision, as certain operations can be very ambiguous, and due to the lack of precise information about the problem, where the use of empirical models can be very restrictive or might not even encompass all the possibilities [42].

Fuzzy logic is a soft-computing approach as it allows to connect the human ability to learn from previous mistakes with complex computational problems using mathematical knowledge [43]. In fact, fuzzy modeling is considered interpretable and transparent when compared to other techniques, as it can often use natural language to describe the inputs, outputs and the relationships between them to build fuzzy models.

2.3.1 Fuzzy Logic and Fuzzy Inference

Traditional logic usually states that a variable can only belong exclusively to one class by either taking a true (1) or false value (0). In contrast, fuzzy logic allows variables to have values between 0 and 1 that represents the degree of membership. Considering a classification problem with two classes, in traditional logic, a variable can solely belong to one class, whilst in fuzzy logic, this variable can have a membership value representative of its degree to belonging to that same class.

A fuzzy inference system can be represented by the procedure of taking input variables though a fuzzy model and reaching an output. This process involves memberships functions to define the degrees of membership, If-Then rules to establish relationships between the inputs and outputs, using fuzzy logic operators.

Fuzzy sets

A fuzzy set is set composed of variables having a degree of membership that measures their degree to belonging to a set. As fuzzy sets do not have well-defined boundaries, they are usually described using membership functions [44].

Fig. 2.5 outlines an example of a fuzzy set. This example displays the maximum temperature registered in a specific day, where a temperature higher than 25 degrees Celsius is defined with full degree of membership (color black). In this case, a temperature of 30 degrees Celsius completely belongs to the set, with full degree of membership. A temperature of 10 degrees Celsius does not completely belong to the set, having a zero degree of membership (color white). And finally, a temperature of 20 degrees Celsius is fairly close to the criterion, so it has a partial degree of membership (e.g. 0.5), represented with a color gradient, where this value is decided through a membership function.

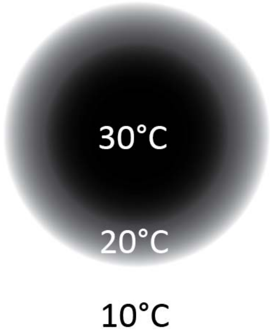


Figure 2.5: Representation of a fuzzy set for the temperature example.

Membership functions

A membership function, as the name implies, is a function that allows to assign membership values (or degree of membership) between 0 and 1 to each point of the fuzzy set [42, 44]. Some common types of membership functions (MF) are: Triangular MF, Trapezoidal MF, Gaussian MF and Sigmoidal MF.

A general membership function is showed in Fig. 2.6. The horizontal axis represents the input variable x and the vertical axis defines the membership value $\mu(x)$. The support of a membership function corresponds to the interval where x will have non-zero membership value ($\mu(x) > 0$). While the core is the interval where x will have full degree of membership ($\mu(x) = 1$).

If-Then rules

Fuzzy models use logic relations and "if-then" rules to establish relationships among the variables defined in the model. The following is the general form of a "if-then" rule:

If antecedent proposition, **then** consequent proposition.

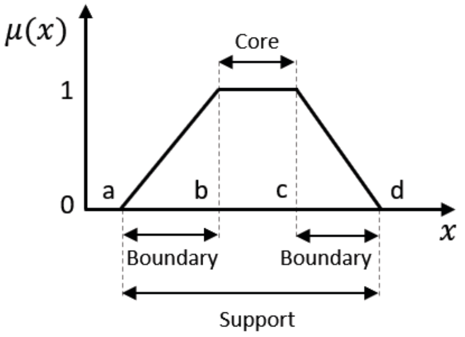


Figure 2.6: Standard membership function representation.

Generally, these rules relate linguistic terms of input variables (antecedent proposition) to linguistic terms of output variables (consequent proposition), where these linguistic terms can be defined by choosing suitable fuzzy sets [42]. In other words, If-Then rules are what allow the linkage between the qualitative values specified in their fuzzy sets, using membership functions, and the input and output numerical values.

From the same example and considering the output ”heating” for the consequent proposition, an “if-then” rule would be:

If temperature is low, then heating is high.

Logical Operators

As previously mentioned, in boolean logic, the result of some logical operation can only be 1 (true) or 0 (false). However, in fuzzy logic, the result of the same operation is a value between 0 and 1, therefore it is necessary to use functions in order to maintain the truth table of those logical operator [45].

The logical operators used are AND, OR and NOT. In fuzzy logic, the operator AND is represented using the function min, i.e., A AND B results in $min(A, B)$. The operator OR is associated with the function max, where A OR B is equal to $max(A, B)$. Finally, the operator NOT as in NOT A becomes $1 - A$ (1 minus A).

2.3.2 Fuzzy Inference Systems

There can be different fuzzy inference systems depending on the structure of the consequent proposition. The most common ones are: Mamdani (Linguistic) fuzzy model and Takagi-Sugeno (TS) fuzzy model:

- **Mamdani fuzzy model** [46, 47] - if X is small and Y is cheap, then Z is good
  Both the antecedent and consequent are usually fuzzy linguistic terms. These models are generally more intuitive and well-suited to human input.

- **Takagi-Sugeno (TS) fuzzy model** [48] - If X is small and Y is cheap, then $z = f(x, y)$
  The consequent is a polynomial function in respect to the antecedent variables. Although these

models are considered to be more computationally efficient, choosing the parameters for the polynomial function can be a hard and not very straightforward when compared to the Mamdani fuzzy model.

A fuzzy inference system is comprised of five parts: fuzzification of the input variables, application of the fuzzy operator, implication from antecedent to the consequent, aggregation of the consequents across the rules and defuzzification.

The example, ”Movie Ratings”, is created to illustrate the complete procedure of a fuzzy inference system. The diagram in Fig. 2.7 outlines the general idea of how the logic behind a fuzzy inference system works. This example has two inputs, plot and actors, three If-Then rules and one output, rating.

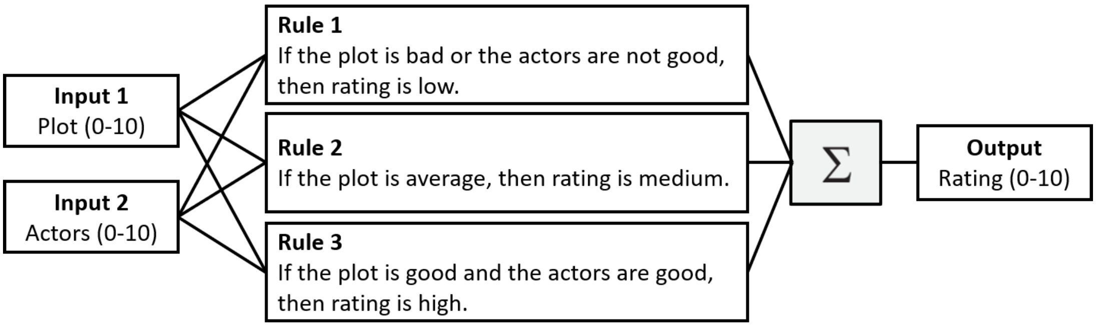

Input 1
Plot (0-10)
Input 2
Actors (0-10)
Rule 1
If the plot is bad or the actors are not good, then rating is low.
Rule 2
If the plot is average, then rating is medium.
Rule 3
If the plot is good and the actors are good, then rating is high.
Σ
Output
Rating (0-10)

Figure 2.7: Logic diagram of ”Movie Ratings” example.

Fuzzification of the input variables

The first step is to take the numerical values of the input variables and determine the equivalent degrees of membership using membership functions. The output is a value describing the degree of membership in the linguistic set between 0 and 1.

The example, ”Movie Ratings”, has three If-Then rules with different fuzzy linguistic sets: plot is good, plot is average, actors are good, actors are not good, etc.. The input variables, plot and actors, need to be fuzzified according to these linguistic sets. The following Fig. 2.8 describes the membership function for the linguistic set ”plot is bad” and shows that, when the input variable plot is equal to three, the membership value for the fuzzy set ”plot is bad” is $\mu = 0.55$.

Application of the fuzzy operator

Taking into consideration the example, where a fuzzy inference system has more than one input variable, the antecedent of the first If-then rule is defined with two fuzzy sets, ”plot is bad” and ”actors are not good”, where a fuzzy operator selects one of the two membership values, representing the result of the antecedent.

The following Fig. 2.9 exhibits the OR operator evaluating the antecedent of the first rule. The two statements (plot is bad and actors are not good) of the antecedent have the membership values 0.55 and 0.0, respectively.

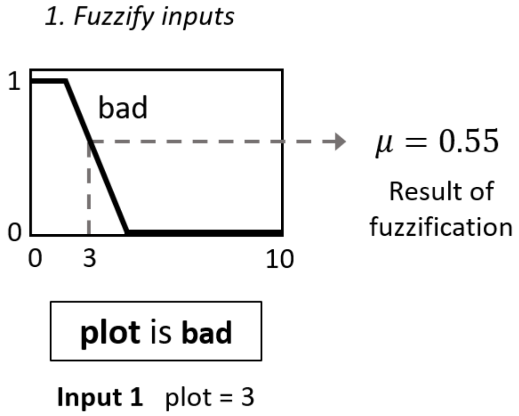


Figure 2.8: Fuzzifying the input variable ”plot” in ”Movie Ratings” example.

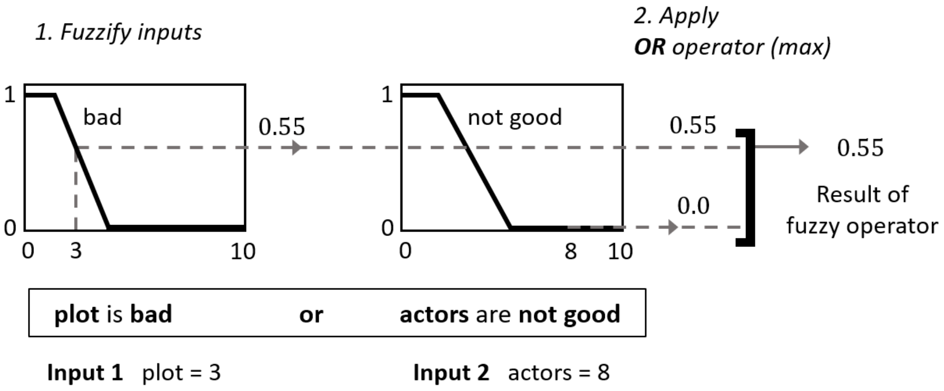


Figure 2.9: Applying fuzzy operator in ”Movie Ratings” example.

As mentioned in logical operators section, the logical OR operator can be expressed using the function max (maximum) and it selects the maximum of the two values, 0.55.

Apply the implication method

The implication method is related to reshaping the consequent fuzzy set based on the result from the previous step and it is applied to every single rule.

Every rule has a weight (number between 0 and 1) assigned to them, which is related to the impact of that rule. In other words, having a lower number decreases its impact relatively to the others. Throughout this work, the weight of every rule is 1, meaning that there is no changes in their impact.

As one can see in Fig. 2.10, the input of the implication method is the number resulted from the application of the fuzzy operator. The output of the implication method is a fuzzy set. In order to obtain the final result, it is necessary to truncate the output fuzzy set using the min function. Figure 2.10 is a visual representation of the implication method, where it shows the min function being implemented.

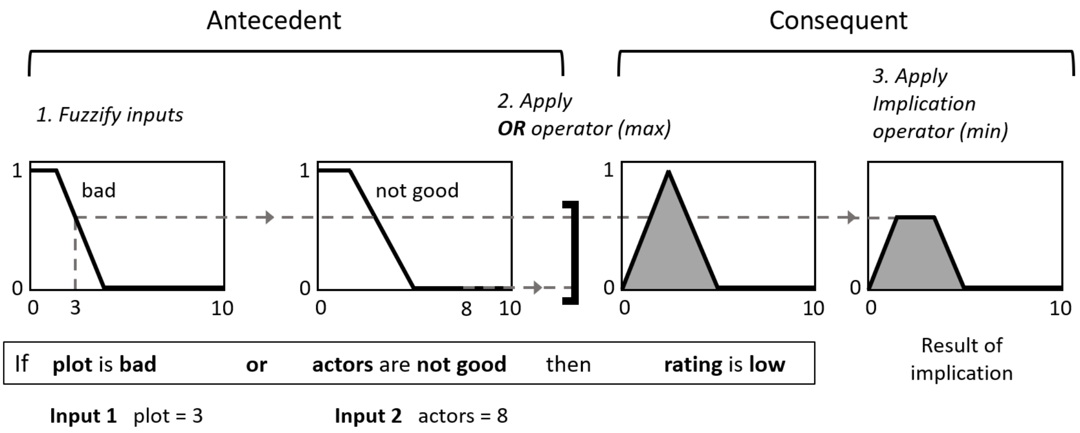


Figure 2.10: Applying implication method in ”Movie Ratings” example.

Apply aggregation method

As the implication method results in an output fuzzy set for each rule, the procedure of the aggregation method is to combine these outputs into a single fuzzy set.

The aggregation operation can be accomplished by applying one of the three functions: max, sum and $probabilistic\ OR$. Usually, the function max is the chosen one since it is more straightforward and simple to use.

Considering the ”Movie Ratings” example, Fig. 2.11 exhibits all the three fuzzy sets, derived from applying the implication method to each rule, being aggregated into a single fuzzy set using the function max. The final fuzzy set represents the output variable ”rating”.

Defuzzification

Finally, the defuzzification method transforms the resulted output fuzzy set from the aggregation method to a single crisp value. From the available defuzzification methods, the centroid method, which returns the center of area under the curve, is the most popular and simple, and, therefore, the only method used in this thesis.

The centroid method can be represented by the following equation:

$$z_{COA} = \frac{\int_z \mu_C(z) z dz}{\int_z \mu_C(z) dz} \tag{2.5}$$

where z_{COA} is the output variable and $\mu_C(z)$ is the membership function of the aggregated fuzzy set C.

Fig. 2.12 outlines the result for the given example, ”Movie Ratings”. For the input variables, plot and actors, equal to three and eight, respectively, the fuzzy system outputs a value of 3.5, which, according to the output's membership functions, is considered a low rating.

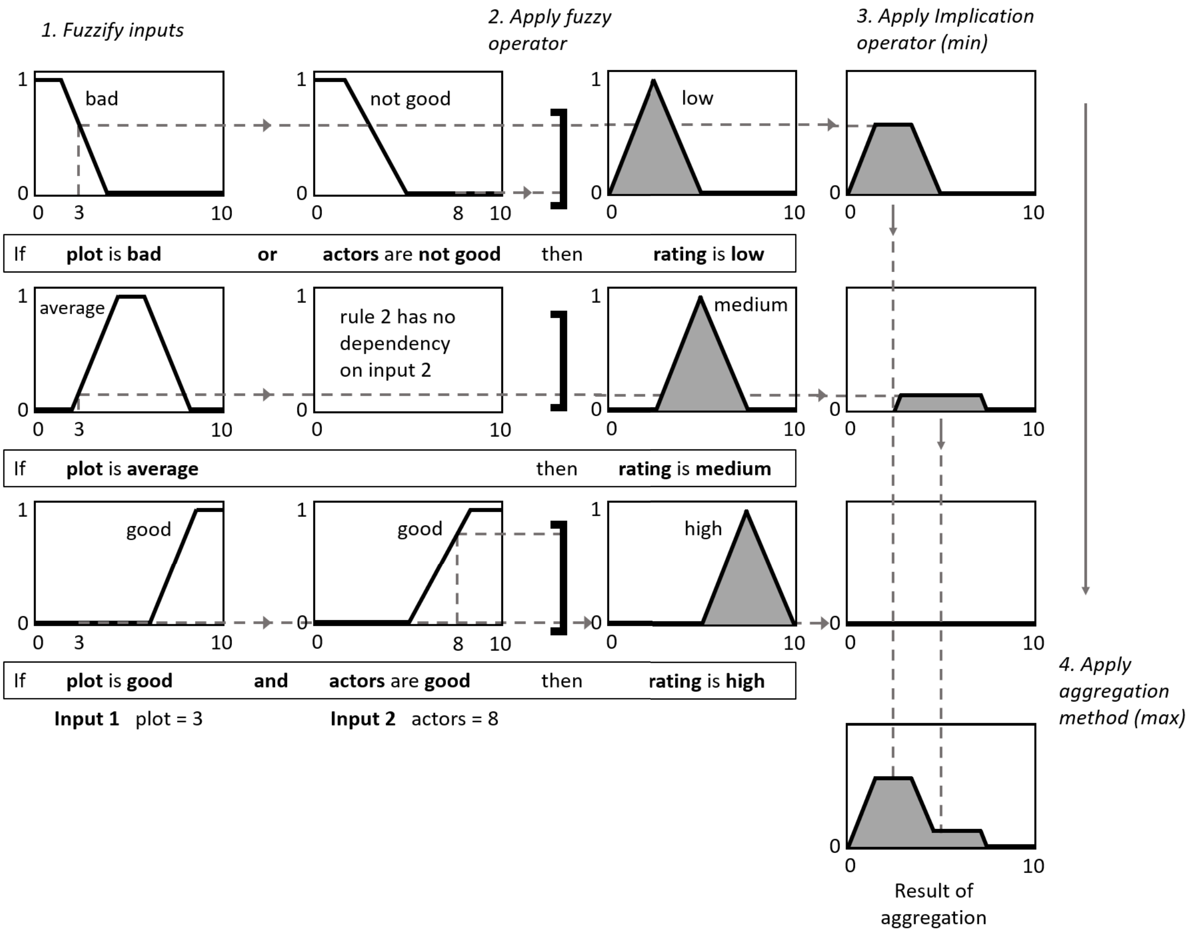

1. Fuzzify inputs
2. Apply fuzzy operator
3. Apply Implication operator (min)
bad
not good
low
If plot is bad or actors are not good then rating is low
average
rule 2 has no dependency on input 2
medium
If plot is average then rating is medium
good
good
high
If plot is good and actors are good then rating is high
Input 1 plot = 3
Input 2 actors = 8
4. Apply aggregation method (max)
Result of aggregation
0
1
3
8
10

Figure 2.11: Applying aggregation method in ”Movie Ratings” example.

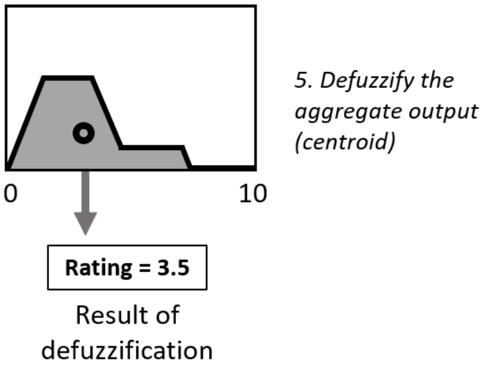

5. Defuzzify the aggregate output (centroid)
0
10
Rating = 3.5
Result of defuzzification

Figure 2.12: Applying centroid method for defuzzification in ”Movie Ratings” example.

Chapter 3

Databases

The following chapter is organized as follows. Section 3.1 begins by presenting in more detail the limitations in present databases and the difficulties in building one. Then, section 3.2 presents the Corsican Fire Database and its characteristics as it is the one used for developing and testing the proposed method. Finally, section 3.3 focuses on the image database being developed between IDMEC and ADAI, and the different annotations created to categorized such dataset.

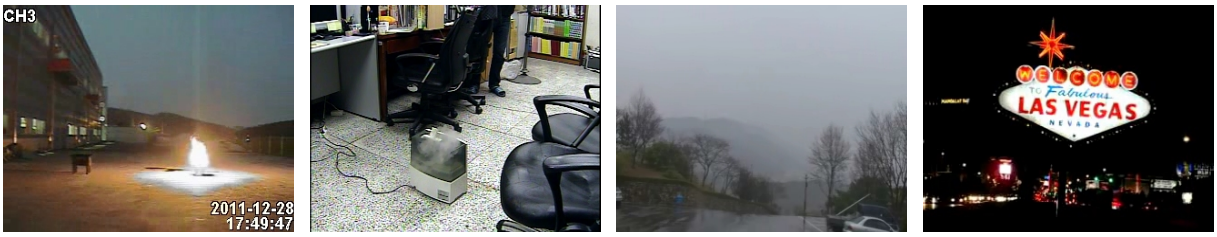
3.1 Importance of Databases

The creation of a meaningful and relevant dataset for wildfire detection and segmentation can be quite the challenge. There are a lot of different real-world scenarios that need to be considered, and creating annotations and ground truth data can usually take long hours. Images with fire and smoke are important but do not encompass all the real-world situations. For a balanced dataset, it is imperative to include samples with firefighters or firetrucks in the field of operations, with sunsets or clouds, as these display similar colors to fire and smoke, and with areas of interest (flame or smoke) at long distances. Additionally, image samples with no fire or smoke regions should also be considered to take into account all the possible real contexts.

In terms of annotations, there is a significant value for researchers having further information beyond knowing if a sample has fire or smoke. In fact, a deep neural network trained in unbalanced dataset, where, for example, all image data have an area of interest centered, will originate a model that can not generalize to other samples where this area of interest is not centered. Therefore, having prior knowledge about the characteristics of a dataset can help researchers improve fire detection approaches. In essence, data annotations are paramount for improving algorithms and embedding relevant information. Taking into consideration how similar sunsets and clouds can be to flame and smoke colors, it can be expected that some algorithms will sometimes misclassify these situations as fire or smoke, respectively. However, having annotations indicating the presence of a sunset or clouds can help the user understand those misclassifications and the reasons behind them, thus allowing to create a more robust algorithm to succeed in these scenarios. For these reasons, some relevant annotations are depicted in

Table 3.1 which includes, e.g., clouds, sunsets, samples captured from the ground level or at night.

Many of the large-scale datasets that we can encounter throughout the scientific community are usually the combination of multiple smaller datasets or have samples based on video frames [11, 49]. However, their samples are usually independent of the scenario they were captured on, in other words, some datasets may include samples with fire or smoke in buildings or in urban contexts (Fig. 3.1a and Fig. 3.1b), or non-fire objects with similar colors (Fig. 3.1c and Fig. 3.1d), which are not adequate for wildfire detection. Moreover, datasets heavily based on video frames [23, 50] result in very similar samples.



(a) Flame (b) Smoke (c) Non-smoke scene (d) Non-fire scene

Figure 3.1: Samples of KMU Fire and Smoke Database [23] to exemplify urban scenarios with fire, smoke, non-smoke and non-fire regions.

3.2 Corsican Fire Database

The Corsican Fire Database (CFDB) is an online database of wildfire images and sequences that allows the evaluation and comparison of computer vision algorithms related to wildfire detection [32]. This database enables users to upload their own images and image sequences, that can be categorized with different annotations, in order to create an evolving dataset. However, as of December of 2020, despite the authors intention, its initial data remains unchanged, attesting the difficulty in developing these repositories. The dataset contains 500 images in the visible region of the spectrum electromagnetic, 100 pairs of visible and near infrared images, and 5 multi-modal sequences acquired in the visible and near infrared areas. Note that this database only contains images with fire. Furthermore, every image is associated with a ground truth (binary) image manually created, where the white pixels indicate the fire pixels, a semantic label in both english and french indicating the dominant color of the fire (e.g., rouge/red), a percentage of fire pixels in the image, a percentage of fire pixels covered by smoke and the level of texture of the fire area, etc.. For the purpose of this work, the data used herein consists of samples from the 500 visible subset (Fig. 3.2).

A reduced dataset of the Corsican Fire Database was used for the development and testing of the proposed algorithm (explained in more detail in section 5.1).

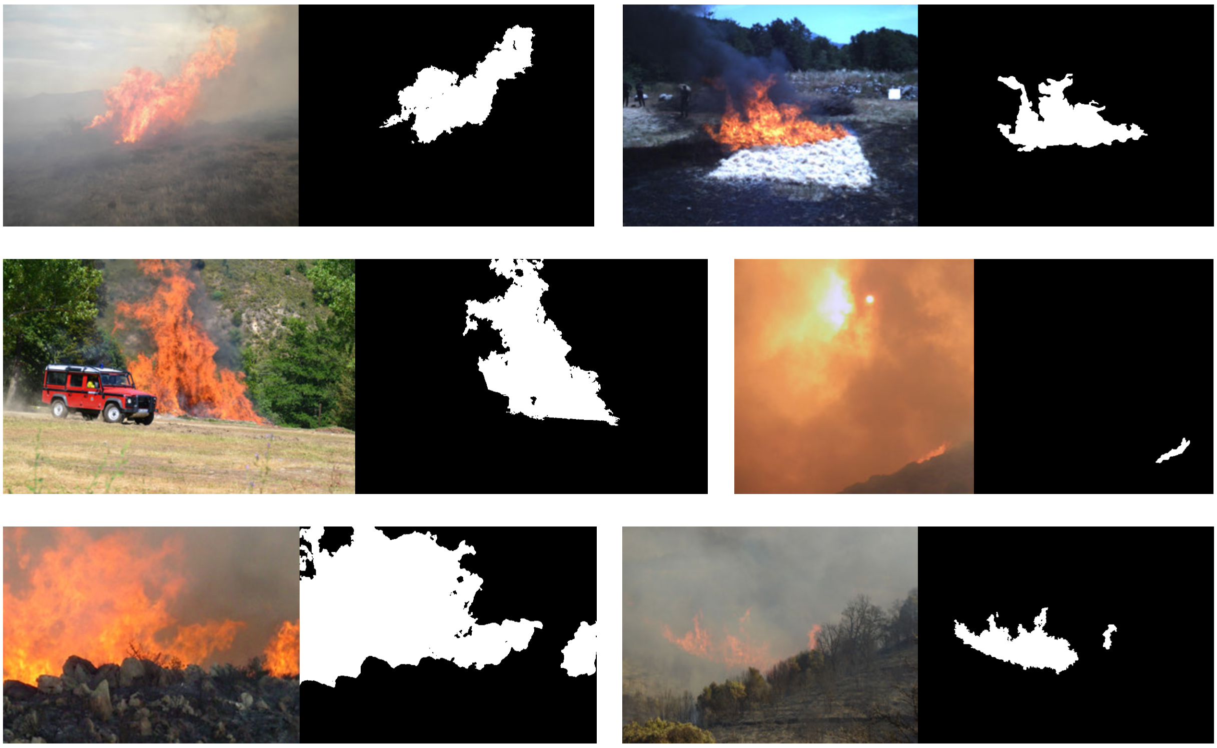


Figure 3.2: Side-by-side samples of fire images (colored) and respective ground truths (binary) of the Corsican Fire Database [32].

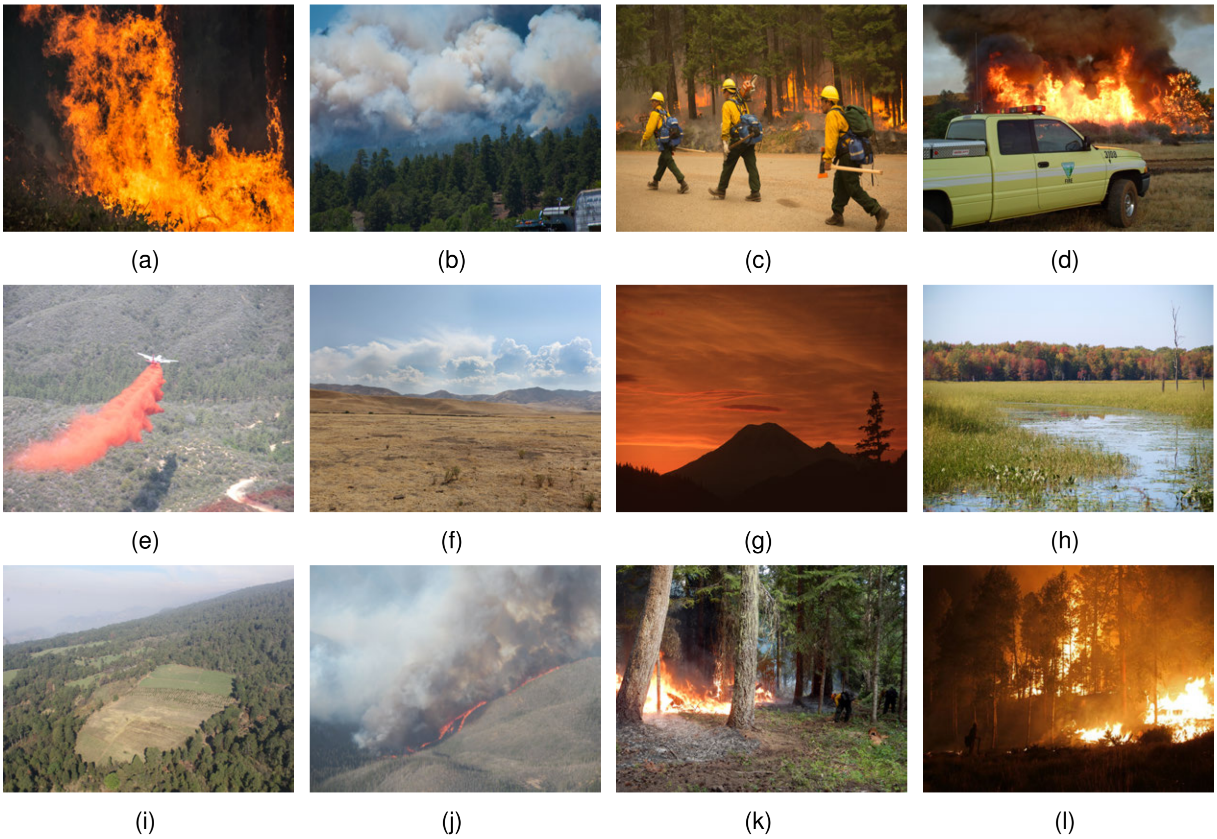
3.3 Fire and Smoke Dataset

This image dataset is part of a larger evolving image database being developed in ongoing research between IDMEC and ADAI towards intelligent wildfire detection and monitoring systems, comprising both thermal and visible range images. For the purposes of this study, the data employed herein concerns only visible range images, which had been gathered and curated prior to this work. The original image sources were released by the United States Forestry Service and are available under public domain. This dataset contains 616 images varying from different real situations to containing different elements (Fig. 3.3).

Considering the importance of annotations in fire and smoke datasets, all the 616 images were classified according to different labels in order to improve the relevance of such database. For a simple and efficient classification, every label presented in Table 3.1 is assigned with a boolean value, depending

Table 3.1: Description of the labels characterizing the Fire and Smoke Dataset. Separated into two groups, Element, related to the presence of wildland-urban elements, and Context, which considers the situation a sample was captured. N_{sample} is the number of images that each label possess.

Element	Label	Flame	Smoke	Firefighters	Firetrucks	Aerial vehicles	Clouds	Sun/Sunset
	N_{sample}	343	524	104	31	39	113	63
Context	Label	Ground level		Aerial view		Long distance	Daylight	Night
	N_{sample}	428		188		201	576	39



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Figure 3.3: Samples of fire images of the Fire and Smoke Database displaying its diversity by including firefighters, firetrucks and other wildland-urban-interface elements, and different conditions, e.g., day, sunset, night, with smoke or fire at long-distances.

on their context, e.g., sunset, night, with fire at long distance, and presence of wildland-urban elements, e.g., fire, smoke, firefighters. Each sample in Fig. 3.3 can be characterized by at least one of the following labels. Some labels like *Flame* and *Smoke* are quite self-explanatory, corresponding to whether an image has fire (Fig. 3.3a) or smoke (Fig. 3.3b), respectively. Others are associated with the presence of non-fire elements with similar fire and smoke colors like firefighters (Fig. 3.3c), firetrucks (Fig. 3.3d) and clouds (Fig. 3.3f). The *Sun/Sunset* label is associated to images with sunsets (Fig. 3.3g) or similar fire colors usually created by the different exposures of sunlight (Fig. 3.4). In addition, the label *Aerial vehicles* is related to samples with airplanes or helicopters as most of them spray a red flame retardant to contain and control wildfires (Fig. 3.3e). Finally, *Aerial view* and *Ground level* labels take into consideration how the image was captured, either in a watchtower or in an aerial vehicle (Fig. 3.3i), or at ground level (Fig. 3.3h), respectively. *Long distance* corresponds to images with a flame or smoke region really far away (Fig. 3.3j). Lastly, the labels *Daylight* and *Night* are related to whether the image was captured during the day (Fig. 3.3k) or at night (Fig. 3.3l), respectively.

In addition to the information presented in table 3.1, it may be interesting to mention that there are 23 images solely with fire, 204 solely with smoke, 320 with both fire and smoke and 58 with no wildland-urban elements, that includes fire, smoke, firefighters, firetrucks and aerial vehicles.

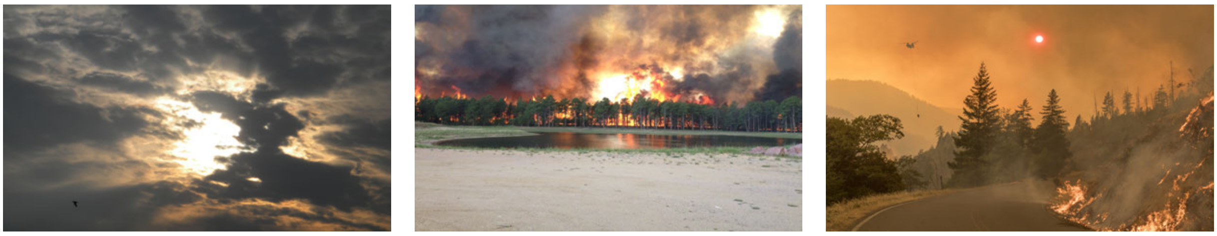


Figure 3.4: Samples with the *Sun/Sunset* label to exemplify other possible situations.

Chapter 4

Methodology

The proposed architecture, outlined in Fig. 4.1, is comprised of three core parts: *i)* Color Feature Engineering (Section 4.2), *ii)* Color-based Superpixel Segmentation (Section 4.3), and *iii)* Interpretable Rule-base (Section 4.4). The first part converts the image data to the HSL and YCbCr color spaces. The second part uses a purpose-built segmentation method that generates superpixels and merges them into regions according to statistical color-based features. Then, the interpretable rule-base employs linguistic models to classify the merged regions, yielding the pixel-wise segmentation of fire and the corresponding color labels for each region. Subsequently, the results can be reviewed by experts to validate and fine-tune new ground truths for fire image data [26].

The following chapter is dedicated to the method developed for the proposed approach. This chapter starts with section 4.1 outlining the problem description where it defines the two stages the proposed approach has and their objectives. Section 4.2 explains the reasons to choose the HSL and YCbCr color spaces for the fire color detection topic. Then, section 4.3 describes the superpixels algorithm and what parameters were used, and the proposed procedure that merges the superpixels together based on the YCbCr color features. Finally, section 4.4 explains the interpretable rule-based models used to classify the different regions according to the possibility of being fire-colored and the semantic color label representative of that fire color.

4.1 Problem Description

The problem tackled by the proposed architecture can be under a two-stage approach: *i) segmentation* and *ii) classification*. The first one is relative to the segmentation of an image that is partitioning into several superpixels and, subsequently, regions (see section 4.3.2). This is achieved using the SLIC algorithm and taking advantage of the HSL and YCbCr color features. The second addresses the classification of the different segmented regions by assigning different categories according to their similarity to fire color attributes [26]. In essence, the objectives of this work are twofold: *i)* pixel-wise segmentation of fire and *ii)* description of the fire color category.

Consider a preset color space, $\mathcal{D}$, that can be defined as $\mathcal{D} \in \mathbb{R}^c$, where c represents the number of

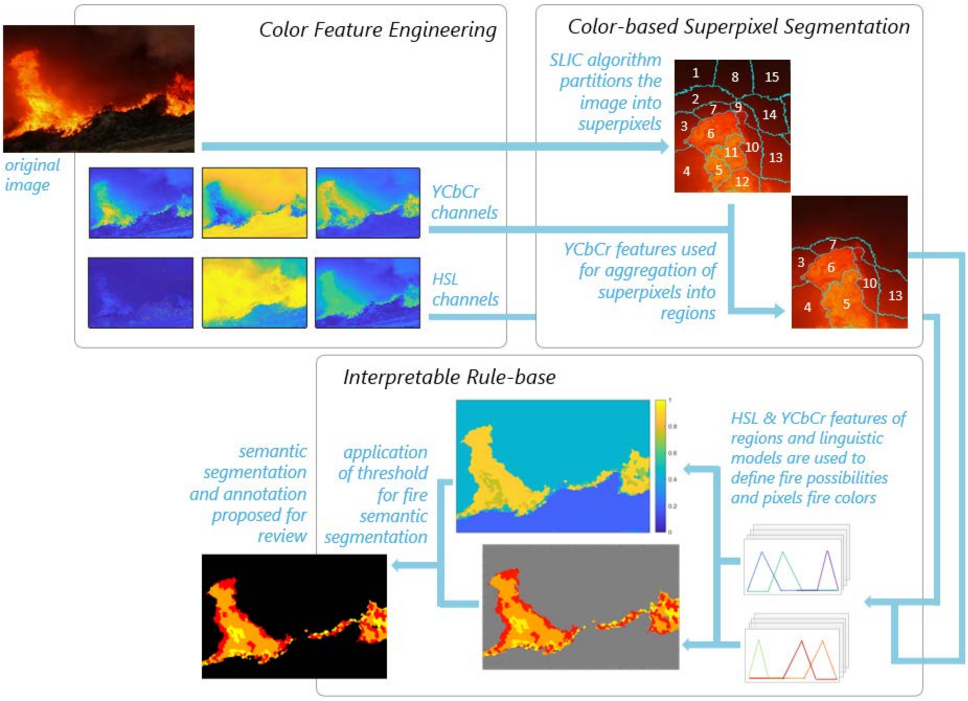

Color Feature Engineering
original image
YCbCr channels
HSL channels
Color-based Superpixel Segmentation
SLIC algorithm partitions the image into superpixels
YCbCr features used for aggregation of superpixels into regions
Interpretable Rule-base
HSL & YCbCr features of regions and linguistic models are used to define fire possibilities and pixels fire colors
application of threshold for fire semantic segmentation
semantic segmentation and annotation proposed for review

Figure 4.1: Visual representation of Fire Data Annotation approach.

channels. A sample image, I, encoded in a $\mathcal{D}$ color space domain, is composed of multiples pixels with every pixel x also being defined as vectors in $\mathbb{R}^c$.

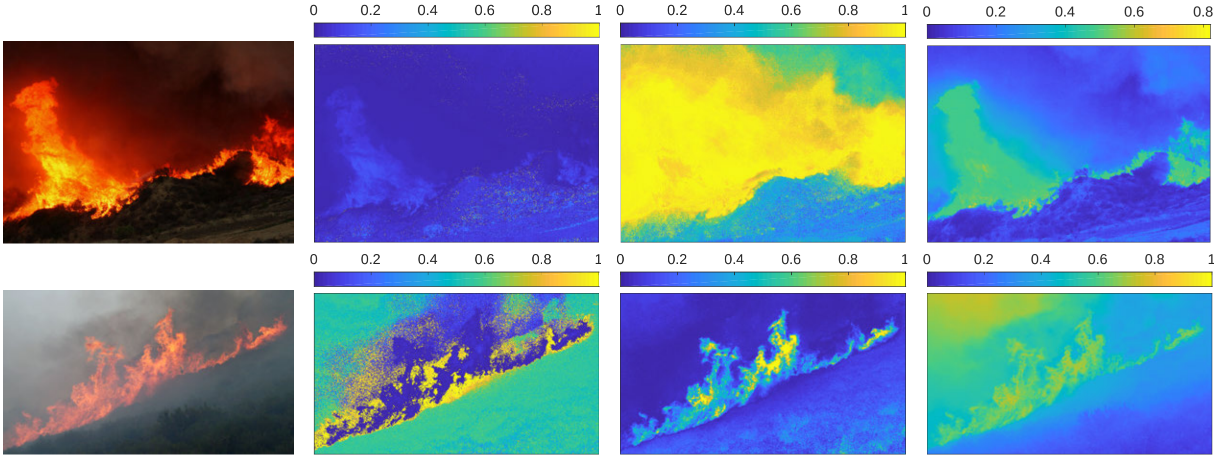
The first objective addresses the pixel-wise segmentation of the flame in an image by representing every pixel as part of two different classes, $\mathcal{F}$ and $\mathcal{N}$. Let $\mathcal{F}$ represent the set of pixels associated to the fire class, i.e., pixels exhibiting fire colors, and $\mathcal{N}$ the pixels belonging to the non-fire class, in other words, pixels that do not display fire colors [26]. Accordingly, the *ground truth* defines the expert validated data, where both classes are, in this case, binary and mutually exclusive, i.e., $x \in \mathcal{F}$ or $x \in \mathcal{N}$.

Furthermore, the second objective tackles the description of fire colors where four different categories, namely red, orange, yellow and other, were modeled in order to create annotations relatively to both the number of fire pixels and the number of pixels belonging to each category, enabling further improvements in the overall dataset. Note that only pixels belonging to the fire class, $x \in \mathcal{F}$, are subsequently part of one of the four categories $\{\mathcal{C}_{\text{red}}, \mathcal{C}_{\text{orange}}, \mathcal{C}_{\text{yellow}}, \mathcal{C}_{\text{other}}\}$. In contrast to the first objective, defining pixels to a color category in a crisp way can be very challenging.

4.2 Seeing Fire Across Color Spaces

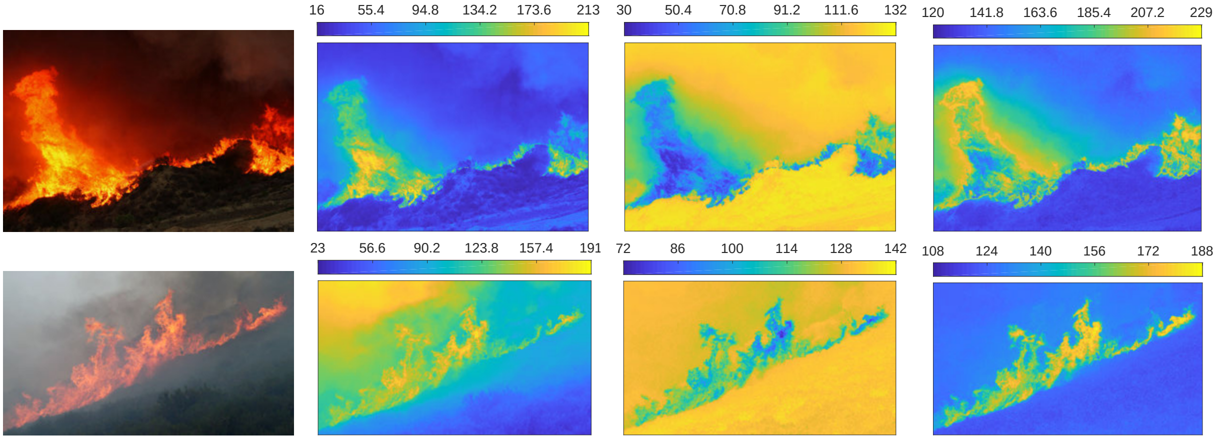
As the objective is to segment the flame based on color, choosing relevant color spaces can be very helpful when defining parameters that correspond to fire colors, thus, leading to better results. This is a particularly important step for images with similar fire colors in non-fire regions and when there is smoke over the flame, decreasing the perception of the fire colors even for human annotation.

For these reasons, the color spaces used are the HSL and the YCbCr. The HSL color space is easy to use and works well in scenarios with a high contrast between the flame and the background. The saturation and lightness channels are more intuitive in defining fire colors and separating them from dark smoke (high saturation) and clouds (low lightness). Moreover, the hue channel makes it easier to specify the range of colors for the linguistic terms (e.g., red, orange, yellow). However, this color space is more challenging when it comes to images with smoke in the scene or regions with similar colors to fire colors (Fig. 4.2). In these situations, the YCbCr color space allows for an easier flame segmentation (Fig. 4.3) because of its ability to separate the colors, but it is less interpretable than the HSL making its development more complex. Color-based features derived from these datasets are employed in the two stages of the proposed approach, namely in the segmentation and classification parts.



(a) Original Image (b) Hue (c) Saturation (d) Lightness

Figure 4.2: Fire Features in HSL. Visual display of each HSL color channel for two different images.



(a) Original Image (b) Luminance (c) Chrominance Blue (d) Chrominance Red

Figure 4.3: Fire Features in YCbCr. Visual display of each YCbCr color channel for two different images.

4.3 Color-based Superpixel Segmentation

The use of superpixels allows to segment an image by clustering pixels based on their color and proximity. This technique can be very useful since superpixels can carry more information than pixels [33], adhere better to the image boundaries and reduce the complexity of several image processing operations [36], thus decreasing processing times.

4.3.1 Superpixel Algorithm

The superpixels are generated using the simple linear iterative clustering (SLIC) algorithm (see section 2.2.1) that allows the specification of both the desired *number of superpixels*, N_{sp}, and their *compactness*, m. The number of superpixels drives the granularity of the image partitioning, being higher with the increase in N_{sp}. In turn, the compactness controls the shape of each element, with higher values creating more regularly shaped superpixels (square like), and lower values creating more irregular shapes, which adhere better to intricate image boundaries.

Number of superpixels, N_{sp} - This number must ensure that the algorithm can achieve a fine-grained segmentation, which drives the quality of the segmentation. Since, by nature, flame shapes are very irregular and the image data might contain regions of interest that are captured at long distances, if N_{sp} is too small the image partitioning results in larger superpixels that do not adhere exclusively to the flames. This behavior is illustrated in Fig. 4.4, where in the lower left corner of the samples presented we can distinctly observe that the superpixels can capture the flames but also aggregate other information nearby. This would inherently degrade the quality of the segmentation, but more importantly, it could prevent an accurate semantic segmentation because a misleading mean color value of the superpixel could result in its misclassification. However, selecting higher values of N_{sp} results in a larger number of increasingly small superpixels, as depicted in Fig. 4.4, which are harder to merge using the mean color statistics as these capture less context information. The value established by default in our algorithm is 1000 as it is considered an adequate trade-off between these factors.

Compactness, m - Since this parameter controls the shape of the superpixels, it is particularly relevant when segmenting irregular shapes like fire. The influence of varying this parameter can be observed notably in Fig. 4.4, by comparing the samples on the upper right corner of top and bottom row images. The effect of enforcing a higher compactness (depicted on the bottom row) could result in less fine-grained semantic segmentation for both fire and fire colors. For this reason, the value of m was established as 1, because it is the lowest value possible, making superpixels adhere better to irregular boundaries.

Furthermore, the superpixels are defined as vectors in $\mathbb{R}^3$ in both HSL and YCbCr color spaces, where each entry corresponds to the mean color of each channel (H_j^{sp}, S_j^{sp}, L_j^{sp} or Y_j^{sp}, Cb_j^{sp}, Cr_j^{sp}) of the superpixel sp, with $j \in [1, N_{\mathrm{sp}}]$.

4.3.2 Merging Superpixels

We propose a procedure that merges similar superpixels into regions. This method combines neighboring superpixels that register a similar color shade. This is performed using the YCbCr color features as these allow the separation of fire from other instances like smoke. Adjacent superpixels, j and i, are compared based on their mean color ($Y_{j,i}^{\mathrm{sp}}$, $Cb_{j,i}^{\mathrm{sp}}$, $Cr_{j,i}^{\mathrm{sp}}$) and merged if each entry of the pairwise

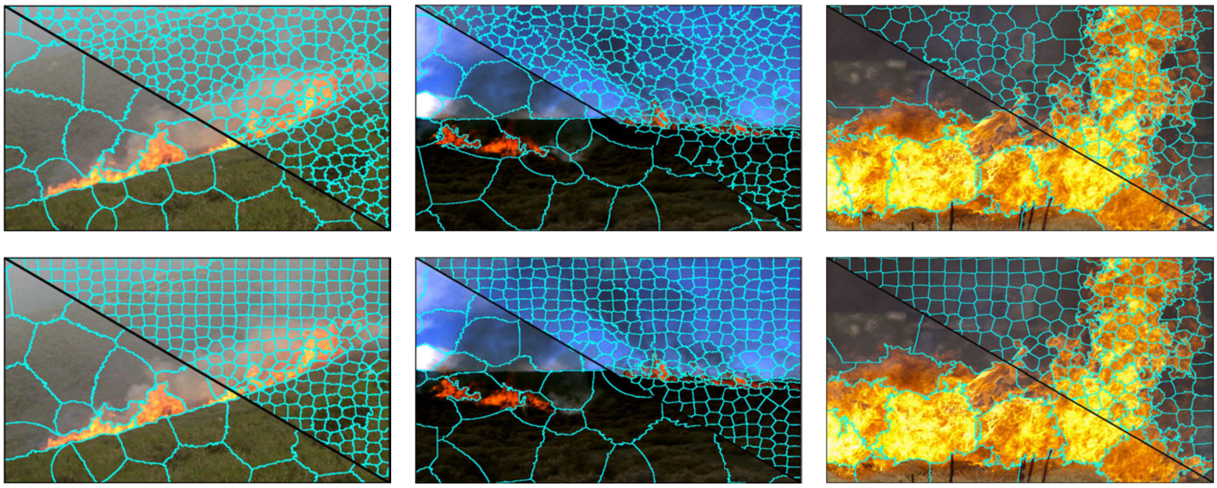


Figure 4.4: Visual comparison between different N_{sp} and C. The m in the lower left corner of each image is 100 and 2000 in the upper right corner. Images in the top row have a m = 1 and images in the bottom row have m = 20.

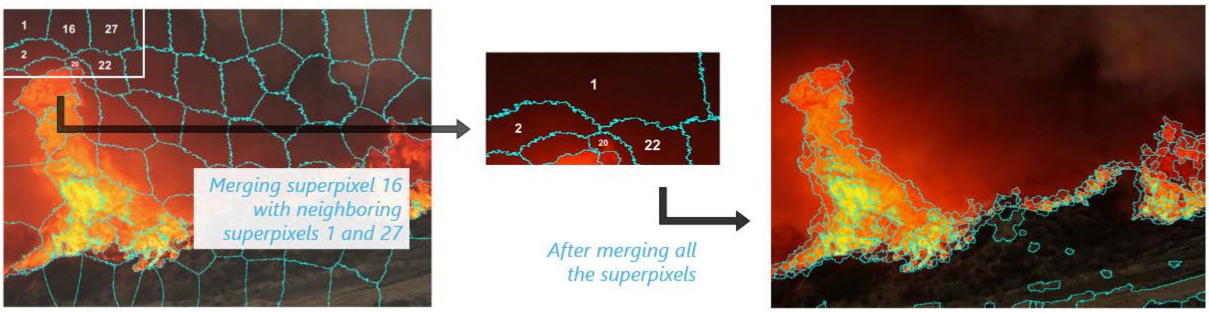
difference is lower or equal to a threshold $\{0.034, 0.1, 0.03\}$, as follows:

$$\begin{aligned} \mid Y_j^{\mathrm{sp}} - Y_i^{\mathrm{sp}} \mid &\leq 0.034 \\ \mid Cb_j^{\mathrm{sp}} - Cb_i^{\mathrm{sp}} \mid &\leq 0.1 \\ \mid Cr_j^{\mathrm{sp}} - Cr_i^{\mathrm{sp}} \mid &\leq 0.03 \end{aligned} \tag{4.1}$$

Figure 4.5 demonstrates how adjacent superpixels would merge together if all the conditions are met. Superpixel 16 is compared with superpixels 1 and 27, and, since all the three conditions in 4.1 are satisfied, the three superpixels are merged together creating region number 1. In the end, this process results in an image similar to the far right one (Fig. 4.5b).

The threshold values used to merge the superpixels were fine-tuned until the overall flame segmentation was quite acceptable and no merged fire region was overlapping with non-fire regions. This procedure was attained by analyzing multiple different scenarios.

Such procedure was implemented in an attempt to achieve the best possible flame-only segmentation, as depicted in Fig. 4.5b, while maintaining a fast processing time.



(a) Image with superpixels

(b) Final result after merging all superpixels

Figure 4.5: Zoomed area shows how similar superpixels 1, 16 and 27 merge, creating region number 1. Merging process results in a region-defined image.

4.4 Interpretable Rule-based Models

Humans describe colors using linguistic terms like red, green or pink, but color perception might differ from person to person [51]. Likewise, image retrieval for search-based analyses is also based on key categorical terms, namely color. Considering that relevant fire characteristics are related to their color, the annotation of these attributes is fundamental towards creating large-scale datasets with relevant information that can be used in a wide range of fire detection scenarios.

4.4.1 Mamdani-type Fuzzy Inference Systems

In this work, we propose interpretable linguistic models, designed for fire segmentation and classification of the regions yielded from the previous step. The rule-based architecture is build with Mamdani-type fuzzy inference systems [47] that describe the rules of the knowledge base with linguistic terms. This architecture incorporates uncertainty and is able to bridge the gap between semantic description of colors and its numerical parametrization.

The concept of the rules relies on the association of the linguistic terms between both the modeling of the color-based features and the categorization of colors, with the underlying range of values. Our approach leverages two complementary models, developed for the HSL and YCbCr color spaces and outlined in Tables 4.1 and 4.2. Both models are defined with three inputs, corresponding to the mean color of each channel for every region. The two models output a fire possibility per region, that is leveraged to perform the classification of the merged superpixels and achieve the pixel-wise segmentation of fire in the images. In addition, the HSL model is able to describe fire color categories to perform semantic segmentation of the colors in the image. The proposed architecture may integrate both models in the data annotation approach (Fig.4.1), combining the HSL and YCbCr using a weighted average or maximum operators, to generate a segmentation of fire. In parallel, the HSL model generates fire color categories that can achieve the final semantic layers that also describe the color of the fire [26].

HSL model

The HSL fuzzy model uses linguistic terms to describe levels of hue, saturation and lightness to model the fire possibility as low or high, and the fire color category, which classifies a region to a corresponding color subset $\{\mathcal{C}_{\mathrm{red}}, \mathcal{C}_{\mathrm{orange}}, \mathcal{C}_{\mathrm{yellow}}, \mathcal{C}_{\mathrm{other}}\}$. The diagram in Figure 4.6 displays an overall visual representation of the respective model.

The interpretable rule-base for the HSL color space is built with simple and intuitive triangular and trapezoidal parametric functions composed of three and four parameters, respectively. Additionally, Table 4.1 outlines the different linguistic terms and respective parameters that each membership function can take. Finally, the inputs and outputs can usually be defined with multiple membership functions as depicted in Fig. 4.7.

As explained in section 2.3.1, a ”If-Then” rule establishes a relationship between the membership functions of inputs and outputs, using their respective linguistic terms. The knowledge-base comprises

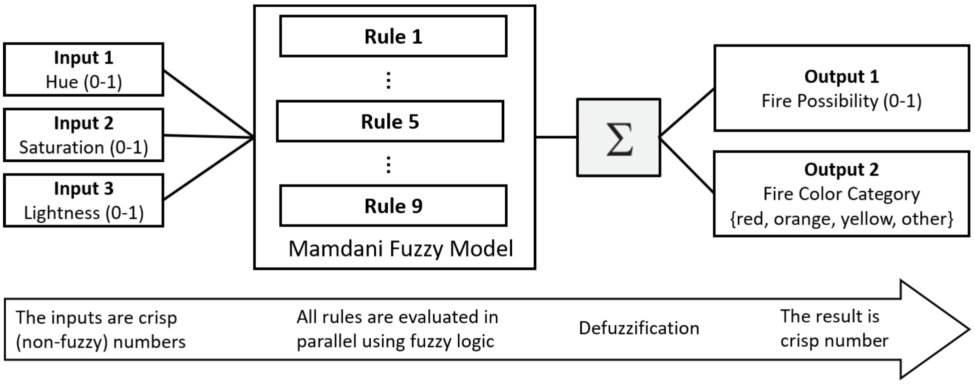
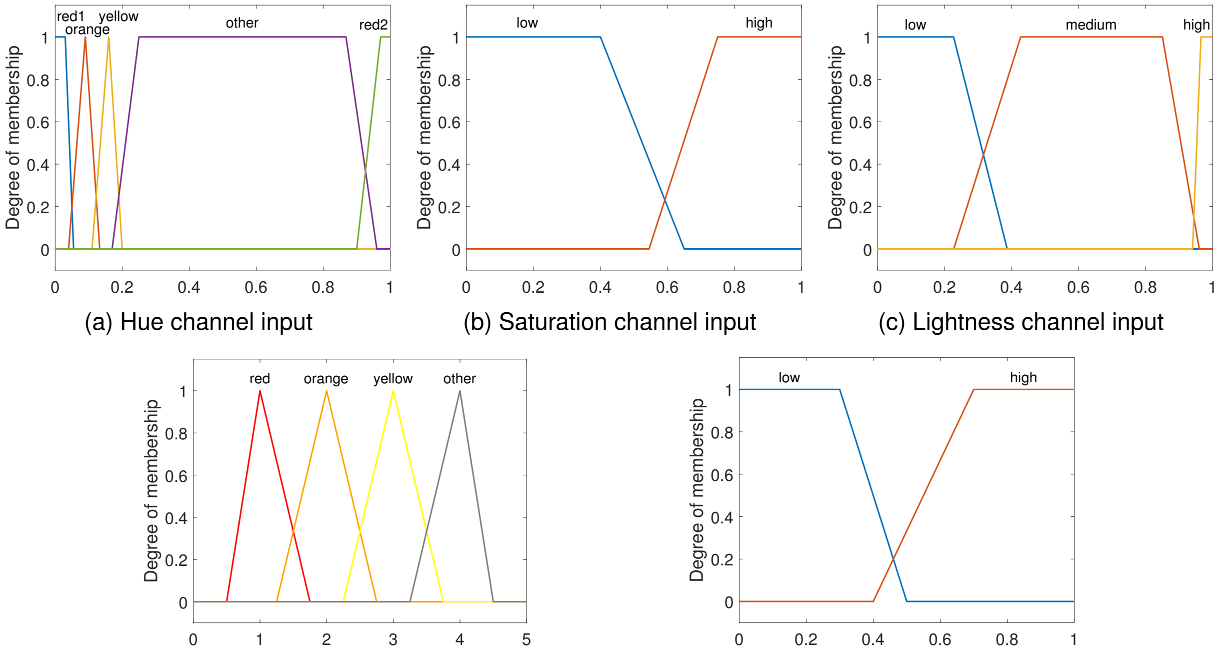


Figure 4.6: Diagram of the fuzzy model in HSL color space: a 3 inputs, 2 outputs and 9 rule system.

Table 4.1: Interpretable Linguistic HSL Model. Description of the HSL ruled-based system designed for fire data annotation. The model has three inputs, hue, saturation and lightness, and two outputs, fire possibility and fire color category. Each input and output is defined with different linguistic terms and parameters, forming membership functions.

	Input			Output		
Model	Variable	Linguistic terms	Parameters	Variable	Linguistic terms	Parameters
HSL	Hue	red1, orange, yellow,	$[0, 0, 0.03, 0.055]$; $[0.04, 0.09, 0.133]$; $[0.11, 0.16, 0.2]$;	fire possibility	low, high	$[0, 0, 0.3, 0.5]$; $[0.4, 0.7, 1, 1]$;
		other, red2	$[0.17, 0.25, 0.87, 0.96]$; $[0.9, 0.97, 1, 1]$;			
	Saturation	low, high	$[0, 0, 0.4, 0.65]$; $[0.545, 0.75, 1, 1]$;	fire color	red, orange,	$[0.5, 1, 1.75]$; $[1.25, 2, 2.75]$;
	Lightness	low, medium, high	$[0, 0, 0.23, 0.39]$; $[0.23, 0.427, 0.85, 0.96]$; $[0.94, 0.965, 1, 1]$;		yellow, other	$[2.25, 3, 3.75]$; $[3.25, 4, 4.5]$;



(a) Hue channel input

(b) Saturation channel input

(c) Lightness channel input

(d) Fire color category output

(e) Fire color possibility output

Figure 4.7: Visual representation of the membership functions for the inputs, hue, saturation and lightness, and outputs, fire color category and fire color possibility, for the HSL model.

nine If-Then rules, outlined in Fig 4.8. The nodes (rectangles) represent inputs or outputs, where each arrow reflects the linguistic terms that each respective node may take. For example, a ”If-Then” rule of the HSL fuzzy model can be expressed as follows:

If Lightness = high and Saturation = high, **then** Fire color = other and Fire possibility = low.

Note that it was not necessary to create a rule for every possible combination as it would induce redundancy.

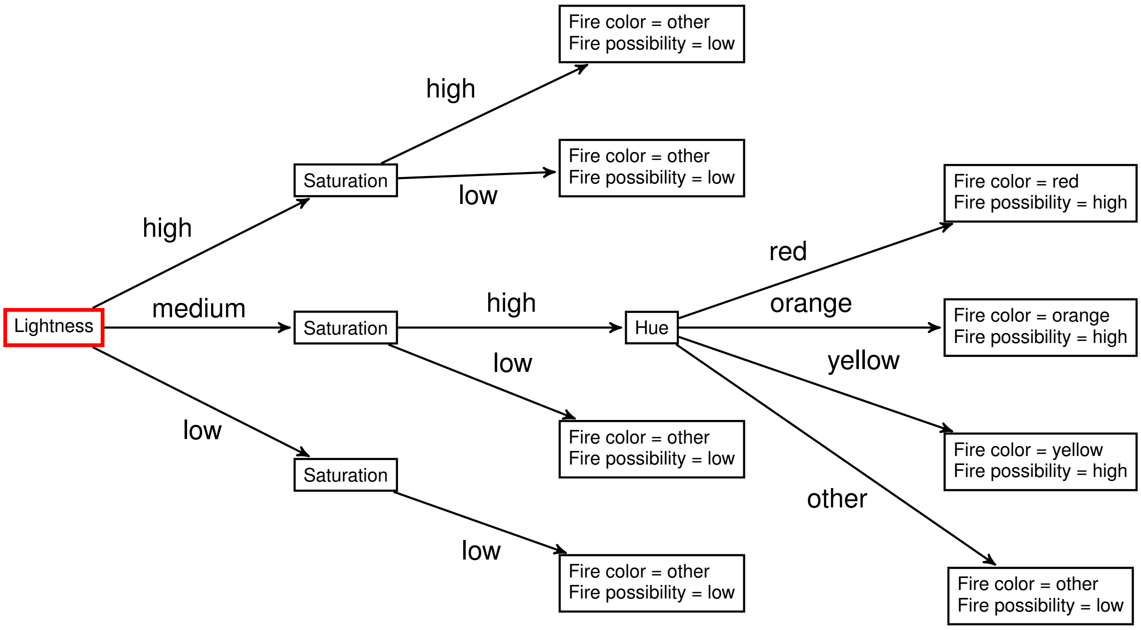


Figure 4.8: Graph representation of the nine If-Then rules for HSL model. Each node (rectangle) represents an input or output just like each arrow represents the respective value a node may take. For an easier analysis, the red rectangle indicates the starting point.

YCbCr model

The YCbCr fuzzy model employs terms describing degrees of luminance, chrominance blue and chrominance red to model the fire color possibility in terms of low, medium, medium high and high. The diagram depicted in Figure 4.9 displays an overall visual description of the YCbCr model.

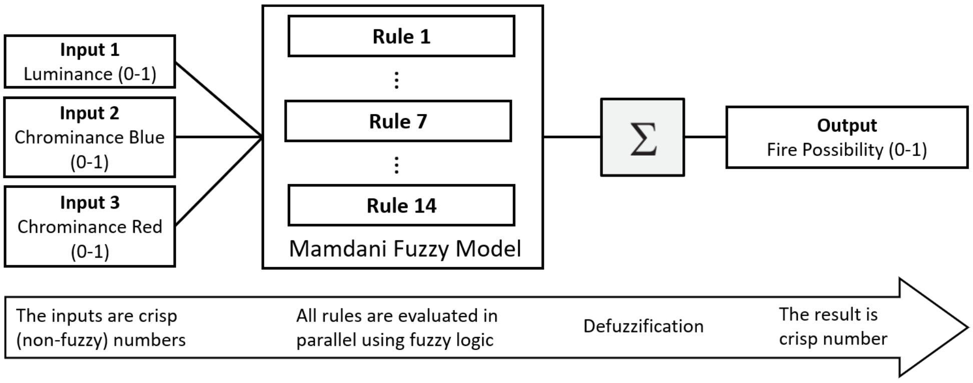
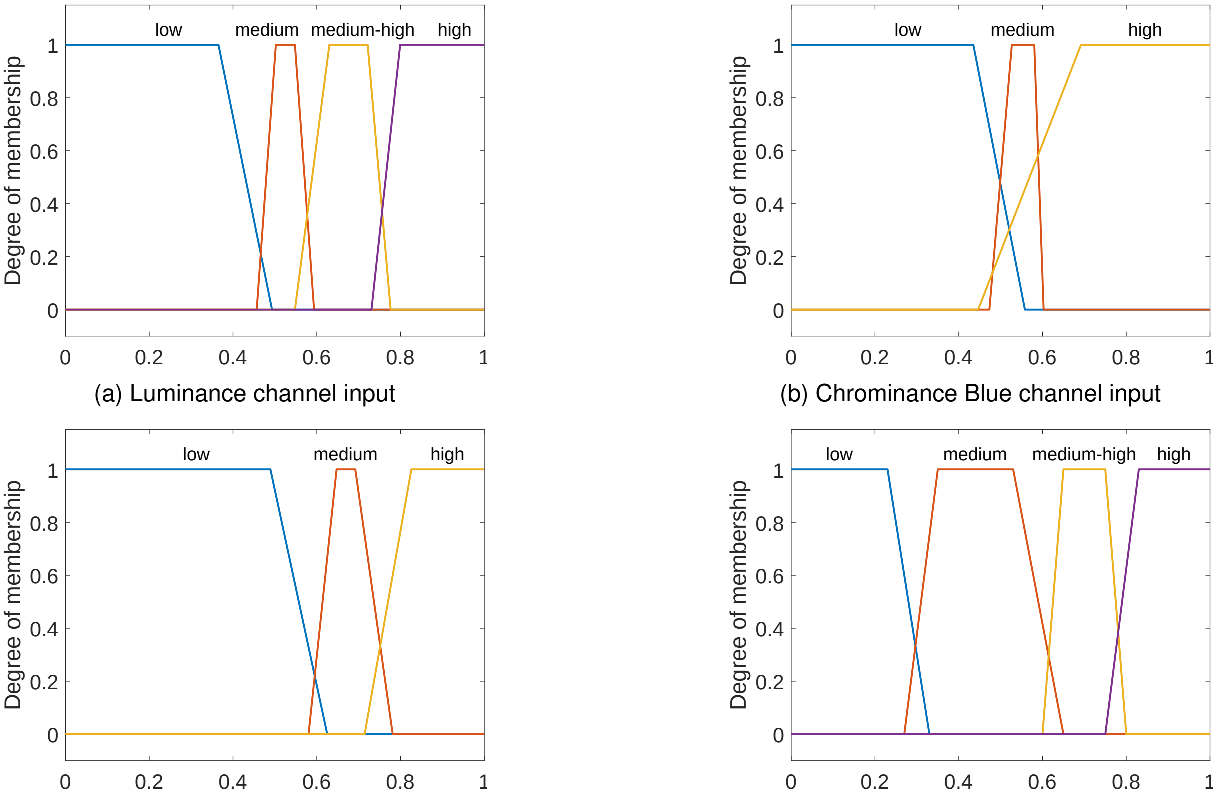


Figure 4.9: Diagram of the fuzzy model in YCbCr colour space: a 3 inputs, 1 output, 14 rule system.

Similar to the HSL model, the YCbCr fuzzy model is developed using simple and intuitive triangular and trapezoidal membership functions composed of three and four parameters, respectively. Additionally, Table 4.2 addresses the different linguistic terms and respective parameters that each membership function can take. Finally, the inputs and outputs can usually be defined with multiple membership functions as demonstrated in Fig. 4.10.

Table 4.2: Interpretable Linguistic YCbCr Model. Description of the YCbCr ruled-based system. The model has three inputs, luminance, chrominance blue and chrominance red, and, contrarily to the previous one, only one output, fire possibility. Each input and output is also defined with different linguistic terms and parameters to form membership functions.

	Input			Output		
Model	Variable	Linguistic terms	Parameters	Variable	Linguistic terms	Parameters
	Luminance	low, medium,	$[0, 0, 0.365, 0.49]$; $[0.457, 0.5, 0.548, 0.594]$;			
YCbCr		medium high, high	$[0.548, 0.63, 0.72, 0.776]$; $[0.73, 0.8, 1, 1]$;	fire possibility	low, medium,	$[0, 0, 0.23, 0.33]$; $[0.27, 0.35, 0.53, 0.65]$;
	Chrominance Blue	low, medium, high	$[0, 0, 0.435, 0.56]$; $[0.47, 0.527, 0.58, 0.6]$; $[0.446, 0.69, 1, 1]$;		medium high, high	$[0.6, 0.65, 0.75, 0.8]$; $[0.75, 0.83, 1, 1]$;
	Chrominance Red	low, medium, high	$[0, 0, 0.49, 0.625]$; $[0.58, 0.647, 0.69, 0.78]$; $[0.71, 0.826, 1, 1]$;			



(a) Luminance channel input

(b) Chrominance Blue channel input

(c) Chrominance Red channel input

(d) Fire color possibility output

Figure 4.10: Visual representation of the membership functions for the inputs, luminance, chrominance blue and chrominance red, and outputs, fire color possibility, for the YCbCr model.

This fuzzy model comprises fourteen If-Then rules, outlined in Fig 4.11. Similarly, the nodes (rectangles) correspond to an input or output, where each arrow reflects the linguistic terms that each respective node may take. For this case, an If-Then rule can expressed as follows:

> **If** Chrominance Blue is low and Luminance is medium and Chrominance Red is high,
> **then** Fire possibility is high.

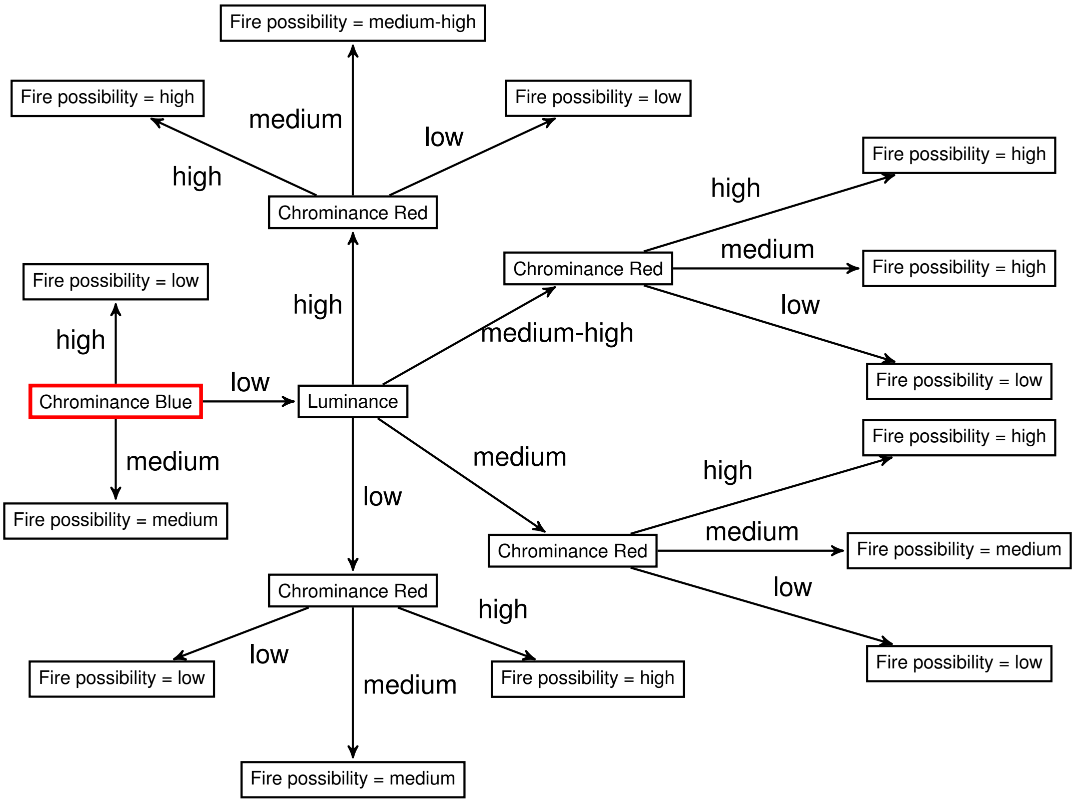


Figure 4.11: Graph representation of the fourteen If-Then rules for YCbCr model. Each node (rectangle) represents an input or output just like each arrow represents the respective value a node may take. For an easier analysis, the red rectangle indicates the starting point.

Chapter 5

Results

The following chapter presents multiple models developed based on the proposed architecture and their respective results evaluated using performance metrics on a smaller dataset from the Corsican Fire Database. Section 5.1 outlines the dataset used for evaluating the performance of the baseline methods and obtain their overall results. Taking into consideration this dataset, 5.2 describes the performance metrics used to assess and compare the different methods. Finally, section 5.3 explains the different baseline approaches, analyzing their differences and their limitations. Moreover, it demonstrates different situations where changing and adjusting key parameters, e.g., number of superpixels or threshold, enables the increase or decrease in the overall performance.

5.1 Fire Image Dataset

The development and testing of the proposed approach was achieved using a smaller dataset (Fig. 5.1) of the Corsican Fire Database (explained in section 3.2). This dataset is composed of 207 images from the 500 samples in the visible region. The reasons behind removing part of the images was mostly due to samples having low resolution or noise (Fig. 5.2a), decreasing the overall quality of the dataset and hampering the results. Moreover, samples with fires created in a controlled environment (Fig. 5.2b), exhibiting cameras and other objects, were also excluded because these do not represent a real-world wildfire scenario. In essence, the objective was to create a dataset capable of representing different wildfire sceneries that emergency teams may encounter.

The performance of the algorithm was fully tested using the 207 images. In an initial stage, the development of each fuzzy model, specially the choice and fine-tuning of parameters, were attained using 50 images for an easier and faster evaluation analysis. Furthermore, the ground truth data considered for this work is the ground truth (binary) images available in the Corsican Fire Database.

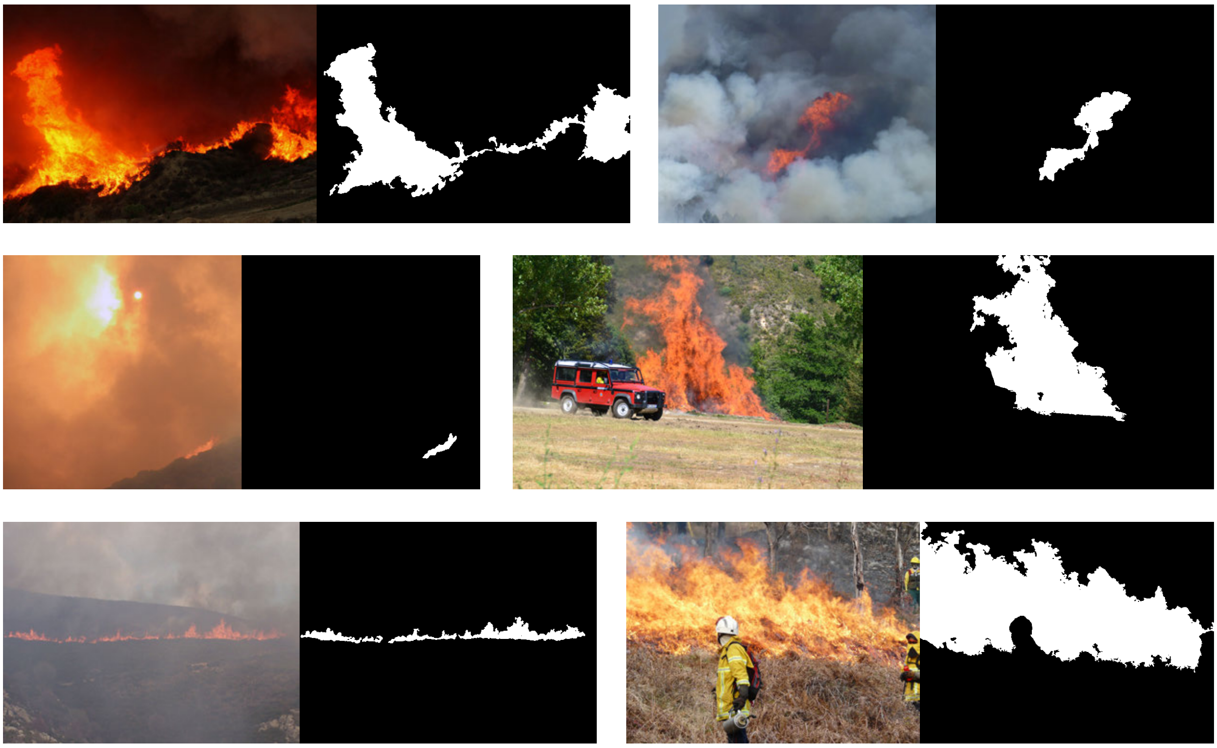


Figure 5.1: Fire Image Dataset. Side-by-side samples of fire images (colored) and respective ground truths (binary), including firefighters and firetrucks, and varying visibility conditions, e.g., day, sunset, night and with smoke.

(a) Low quality image

(b) Controlled environment

Figure 5.2: Samples of low quality and with fires created in a controlled environment.

5.2 Evaluating the Segmentation Models

Considering the objectives of the proposed fire annotation approach (see section 4.1), the performance analysis focus in evaluating the developed architecture with several baseline techniques. This work presents the assessment and comparison of four different methods. The first two techniques are based solely on the HSL and YCbCr models, whilst the others combine both models to incorporate their features and achieve better results. Their difference relies on how the classification of each region is obtained, as one uses a weighted average, while the other is based on the max value operator. For these reasons, it is imperative to use metrics that enable the evaluation of the different segmentation algorithms and the comparison between the segmented images with the ground truth data.

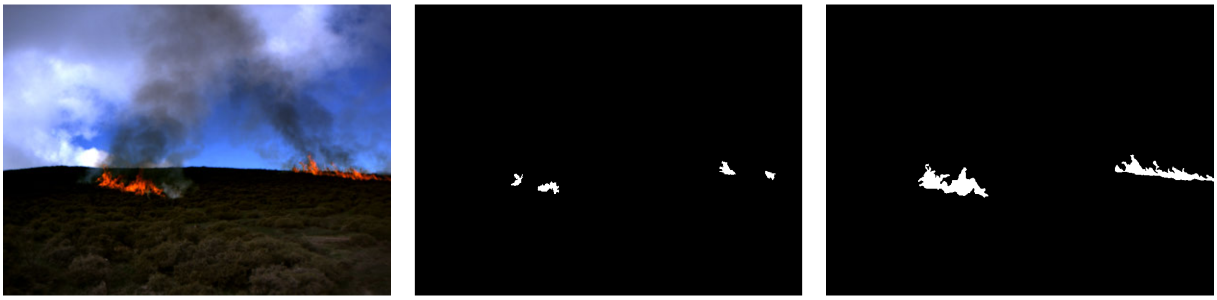
5.2.1 Performance Metrics

The following metrics allow the assessment of the different models in several important aspects. Considering our work is based on a binary classification, the results are often defined based on true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). The terms ”true” and ”false” are associated to whether the prediction model was able to correctly classify the positives and negatives, respectively. In other words, considering a non-fire pixel, a false positive would state that this pixel is a fire pixel (positive) while, in reality, it is not (false). The performance metrics used are accuracy, intersection over union (IoU), precision and recall, which are presented below:

- **Accuracy** evaluates the pixels correctly recognized. It is defined as the ratio between correctly classified pixels and all the available pixels.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.1}$$

Note that this metric can sometimes lead to misleading results when the segmentation is small within the image. As one can see, despite Figure 5.3a registering an accuracy value extremely high, equal to 0.9887, the predicted segmentation (Fig. 5.3b) is not that similar to the ground truth (Fig. 5.3c). This happens because the accuracy takes into account the non-fire pixels that were correctly classified, namely, the true negatives. Nevertheless, this metric is still important as it is one of the most used and versatile metrics in the scientific community, providing an overall result to the dataset and allowing the comparison of results between different approaches.



(a) Original image

(b) Binary Output

(c) Ground truth

Figure 5.3: Scenario where accuracy is extremely high even though the binary output it is not very similar to the ground truth image. Note that the white areas represent the fire-colored pixels, whilst the black regions the non-fire-colored pixels.

- **Intersection over Union (IoU)** [52] - also known as the Jaccard similarity coefficient, measures the overlap between the ground truth segmentation and the predicted segmentation. It is defined as the ratio between the intersection and reunion of the ground truth and the prediction.

$$IoU = \frac{Binary\ Output\ \cap\ Ground\ truth}{Binary\ Output\ \cup\ Ground\ truth} = \frac{TP}{TP + FP + FN} \tag{5.2}$$

- **Precision** measures the pixels that were correctly classified as fire over all the pixels that the model classified as fire. Therefore, this metric is defined by the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{5.3}$$

- **Recall** measures the fire pixels that were correctly classified as fire over all the pixels that are indeed fire pixels.

$$Recall = \frac{TP}{TP + FN} \tag{5.4}$$

5.3 Results Evaluation

The pixel-wise segmentation, stated as the first objective, is defined by classifying each region (superpixels merged together) with the fire color possibility output from the HSL (Fig. 4.7e) and YCbCr (Fig. 4.10d) models (explained in section 4.4.1). As the name implies, this output specifies the degree of membership of each region regarding the possibility of being fire-colored. The threshold, represented as δ, indicates the starting point where each region classification, R_c, is consider mutually exclusive, i.e., $R_c \in \mathcal{F}$ or $R_c \in \mathcal{N}$. In other words, values higher than the threshold are associated with the fire class, $\mathcal{F}$, whilst values less or equal are affiliated to the non-fire class, $\mathcal{N}$, which, in the end, results in the binary image. In an initial stage, the threshold used to obtain the various results was 0.5, which could be then replaced by other values depending on registering positive changes in the overall results. Note that, usually, a binary image represents the set of pixels belonging to the fire class as white and the pixels related to the non-fire class as black.

The following sections represent the different baseline models created to evaluate the performance of the developed architecture.

5.3.1 HSL Model vs. YCbCr Model

Initially, the first baseline approach created to classify each region with a degree of membership relatively to the fire color possibility was based on the HSL color space because of its simplicity and easier interpretability to define the different shades of fire colors. The overall results presented in Table 5.1 demonstrate how, despite registering an extremely high accuracy of 93.39%, the overall predictions were not quite identical to the expert ground truth data as the intersection over union (IoU) can demonstrate. On the one hand, a lower recall value, 58.97%, means that the number of false negatives is considerable high, i.e., the classification model did not correctly classified a lot of fire pixels. On the other hand, a higher precision value, 93.50%, conveys that the false positives are low, indicative of the model not classifying non-fire pixels as fire pixels.

As one can see, Figure 5.4 displays three scenarios where the model using solely the HSL color space exhibits a poor segmeted image. Considering that the intersection over union executes a pixel-wise comparison between the segmented image and the ground truth, it is to be expected that this metric registers very poor results (Fig. 5.4c). Furthermore, one can notice that the HSL model struggles in classifying regions with high smoke levels above the flame pixels. In fact, this limitation is clearly

Table 5.1: Overall results of the HSL and YCbCr models.

Model		δ	Accuracy	IoU	Precision	Recall
HSL	HSL model	0.5	93.39	62.49	92.50	65.77
YCbCr	YCbCr model	0.5	91.58	56.68	93.50	58.97

demonstrated in the last row sample (Fig. 5.4), as it shows some areas with smoke, which were misclassified, and others with a more vivid fire color, correctly classified.

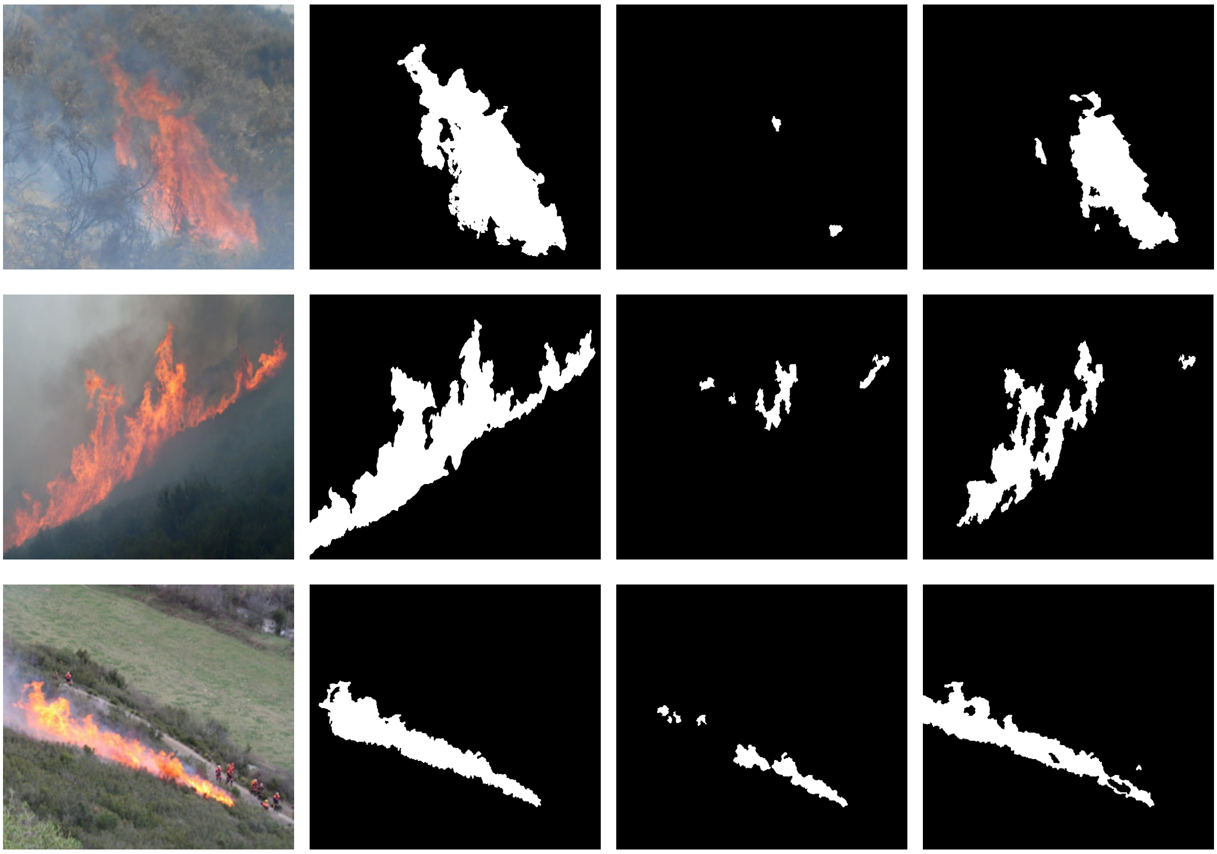
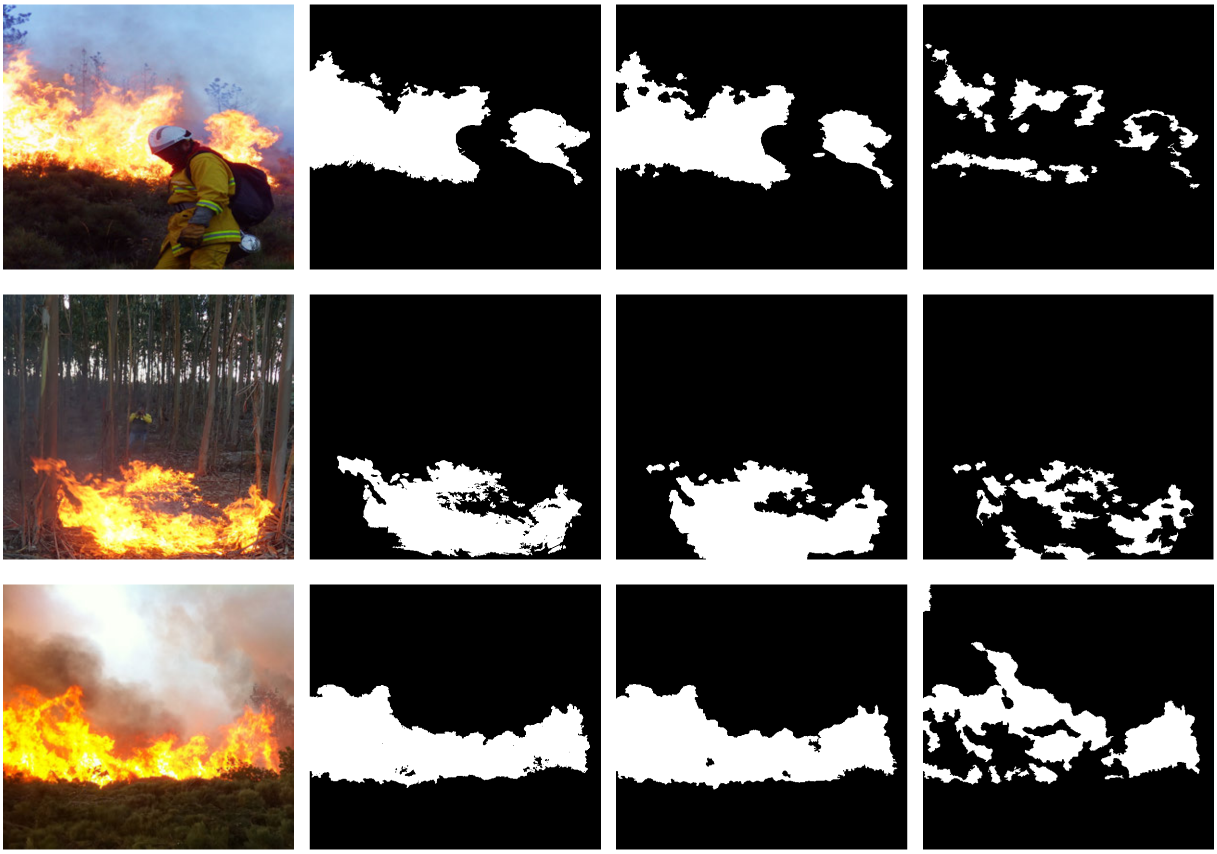

(a) Original Image (b) Ground truth (c) Output - HSL model (d) Output - YCbCr model

Figure 5.4: Samples displaying scenarios with smoke above fire pixels where the HSL model registers worse results than the YCbCr model.

For these reasons, the YCbCr model was created to mitigate such limitation. Considering its ability to separate the luminance from the chrominances blue and red, there is a slightly advantage in using this color space, specially, in high smoke areas, as outlined in Fig. 5.4d. There is a considerably improvement over the results registered from the HSL model. However, as depicted in Table 5.1, the overall results from the YCbCr model are outperformed by the HSL model. Figure 5.5 outlines samples where such observation is confirmed.

As one can see, both models demonstrate different limitations. In areas with high smoke levels above the fire, it is clear that the YCbCr model registers good results in comparison to the HSL. On the other hand, in regions with a bright yellow or a slightly white color (high luminance), the latter exhibits better results (Fig. 5.5c and Fig. 5.5d). Therefore, the next step is to utilize both models and create a multi-model able to incorporate their qualities.



(a) Original Image (b) Ground truth (c) Output - HSL model (d) Output - YCbCr model

Figure 5.5: Samples displaying scenarios where the HSL model registers better results than the YCbCr model.

5.3.2 Combination of Both Models

Considering the previous results, the next baseline approaches are based on combining the HSL and YCbCr models. Since they have registered opposite limitations, the idea is to achieve a multi-purpose model capable of utilizing their qualities and complement each other. There are two proposed approaches with one using a weighted average, with multiple weighted values, and the other selecting the maximum value between the models.

The Weighted model takes into consideration the output fire color possibility from the YCbCr and HSL models and combines these using a weighted average. Taking into consideration the previous observations, we can see that the difference between the results of the HSL and YCbCr models in scenarios outlined in Fig. 5.4 are much worse in comparison to the results depicted in Fig. 5.5. In other words, the YCbCr model is more valuable even though it registers a worse overall performance (see Table 5.2). For these reasons, the weighted model uses a higher weighted value for the YCbCr model. Table 5.2 presents the overall results of all the baseline approaches. Considering the Weighted model, multiple weighted values were used in order to comprehend the degree of importance of each fuzzy model. Even though their results are very similar, we can see a slightly decrease in the overall performance as the weighted value of the YCbCr increases. In particular, the precision metric actually increases as the weighting value of the HSL model decreases, meaning a reduction in the false positives number.

Table 5.2: Overall results of all the models.

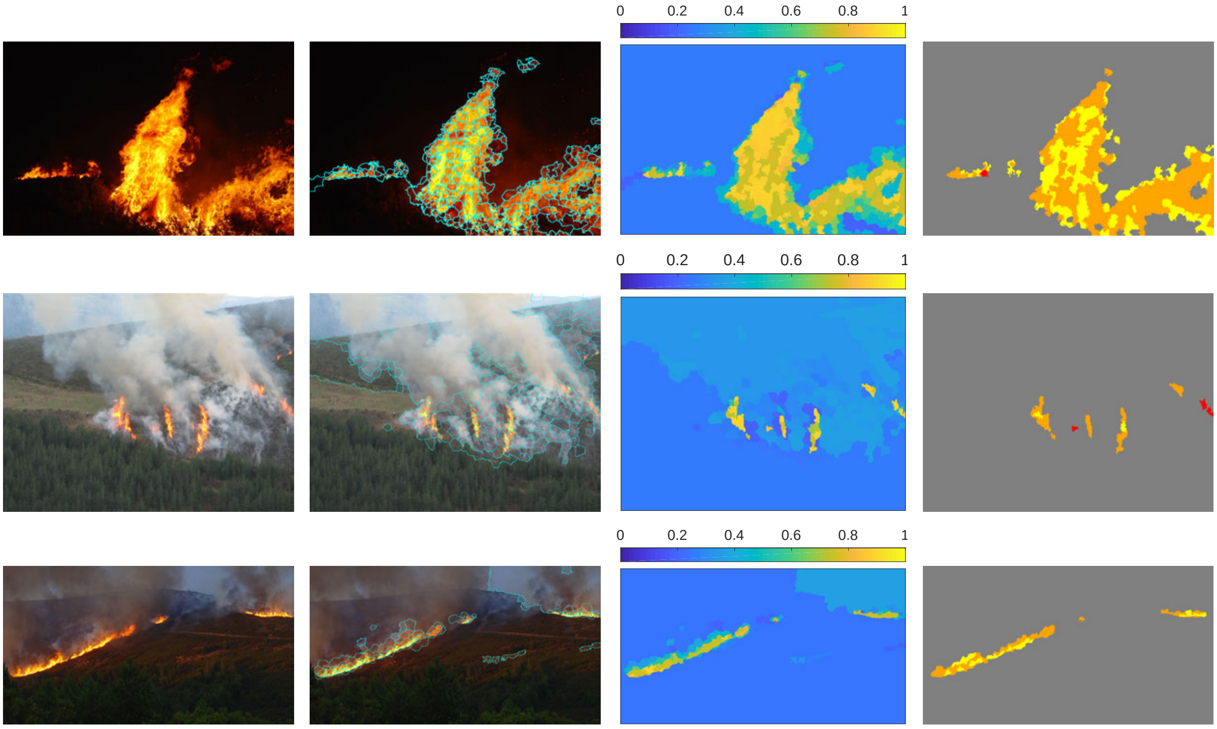
Model		δ	Accuracy	IoU	Precision	Recall
HSL	HSL model	0.5	93.39	62.49	92.50	65.77
YCbCr	YCbCr model	0.5	91.58	56.68	93.50	58.97
Weighted	0.4 HSL + 0.6 YCbCr	0.5	93.16	61.52	94.24	63.83
	0.3 HSL + 0.7 YCbCr	0.5	93.01	61.00	94.27	63.19
	0.2 HSL + 0.8 YCbCr	0.5	92.81	60.13	94.62	62.32
Max Value	max(HSL,YCbCr)	0.5	93.47	66.51	**92.74**	71.23
		0.4	**94.04**	**73.53**	88.57	**83.14**

Despite creating the Weighted model as an alternative to obtain better results when compared to using solely one model, one can see that such expectation is not achieved. From table 5.2, even the Weighted model using a weighting of 40% and 60% for the HSL and YCbCr fuzzy models does not register better results when compared to the HSL, specially in the IoU performance metric.

Consequently, instead of trying to encounter a better weighted values combination for the weighted average, the next baseline approach uses the maximum output between the HSL and YCbCr models to classify each region. As we can see from table 5.2, the Max Value model achieves better overall results than all of the previous ones. Moreover, by changing the threshold value for 0.4, one can observe slight improvements in this model's performance. Solely the precision metric registered a lower result for this threshold. Further explanations and examples about the worth of such parameter are explained in section 5.3.4. Finally, as the Max Value method registers the best performance, the following analysis and results are relative to this model.

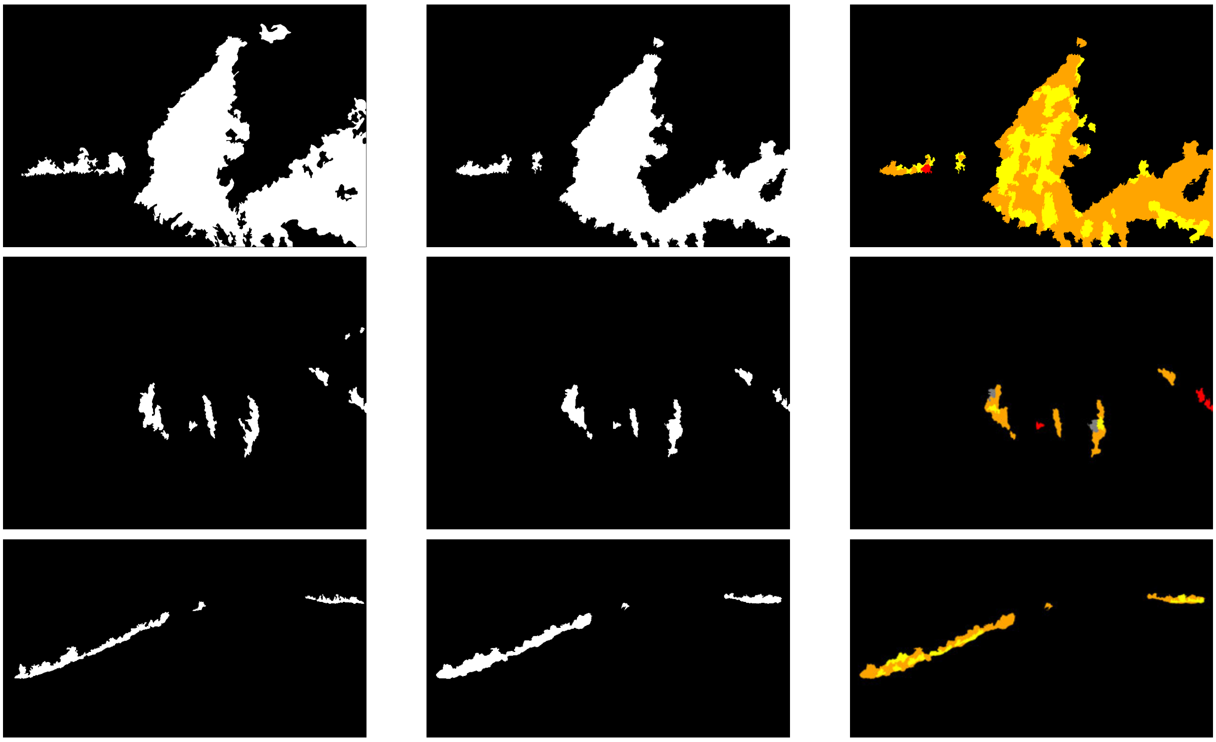
Max Value model

Considering the second objective, the description of the fire color category is achieved by assigning different semantic labels to the segmentation regions belonging to the fire class $\mathcal{F}$. The three samples depicted in Fig. 5.6 and Fig. 5.7 represent different scenarios that the Fire Image Dataset (section 5.1) encompasses and showcase all the intermediate steps of the proposed architecture. In Figure 5.6, the first image represents the original sample in the common RGB color space. The second one exhibits the result after merging all the similar superpixels together to achieve a separation between the flame and the surrounding scenery. Subsequently, the fuzzy models generate the classification of all regions according to the fire color possibility output, where warm colors represent a higher value and cold colors a lower value (Fig. 5.6c). Moreover, the HSL linguistic model also classifies the fire colors as per red, orange, yellow and other, where the latter is represented using the grey color (Fig. 5.6d). Finally, Figure 5.7c outlines the intersection between the binary output (Fig. 5.7b) and the fire colors fuzzy output (Fig. 5.6d).



(a) Original Image (b) Final Merging (c) Fire Color Possibility fuzzy output (d) Fire Colors fuzzy output

Figure 5.6: Samples representing real-world scenarios to exemplify the results at each stage of the proposed approach.



(a) Ground truth (b) Binary Output δ = 0.5 (c) Output

Figure 5.7: Display of the binary output image and output generated by the Max Value model.

As one can see, the binary output image (Fig. 5.7b) is generated by assigning a threshold to the Fire color possibility fuzzy output image which allocates each pixel to the respective fire or non-fire class, thus resulting in a pixel-wise segmentation of the flame. The better the segmentation the closest the binary output is to the ground truth. Furthermore, the output image, depicted in Fig. 5.7c, enables the automatic creation of fire data annotations in terms of expert annotated data (ground truths) (Fig. 5.7b) with information representing the four color categories specific of fire colors (Fig. 5.7c).

5.3.3 Limitations

In a relevant and complete fire image database, it is imperative to include scenery or objects that resemble fire colors as these represent real-world scenarios. Although the proposed architecture registers significant results and allows to create rich annotations, the approach still faces some limitations especially in situations with similar fire colors in non-fire regions, e.g., sunsets, firefighters or firetrucks. Note that this method is a fire color classifier. In other words, as outlined in Fig. 5.8, it is expected that this method classifies scenarios and objects with similar fire colors with a high fire color possibility. Such limitations can be improved by either some type of expert input or with algorithms able to identify this kind of scenes and objects.

5.3.4 Adaptability of the Proposed Approach

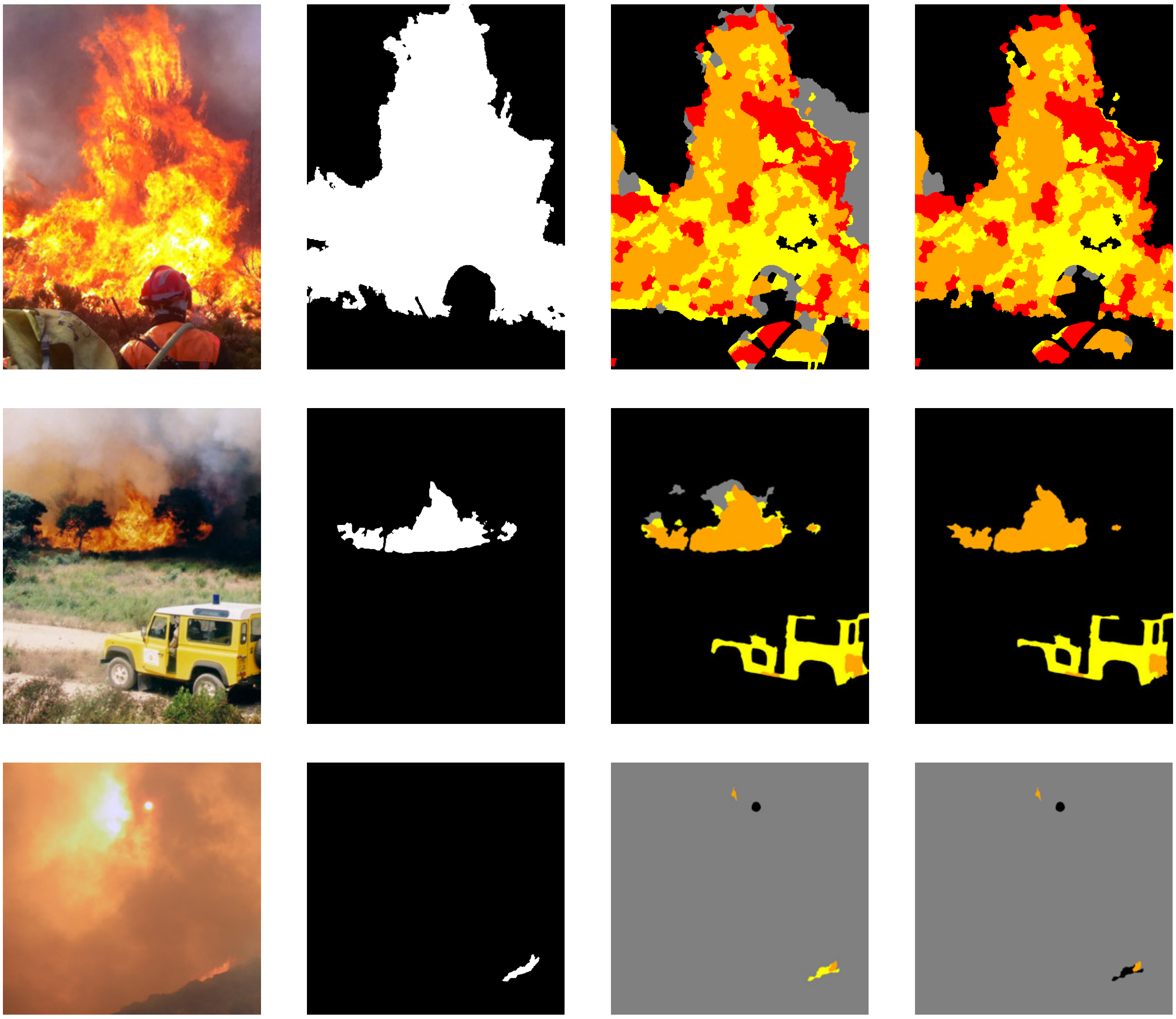
The proposed approach enables easy adjustments of some key parameters associated with the intermediate steps from the architecture. Several parameters are the number of superpixels, which can affect the overall result of merging all the superpixels, the threshold, allowing regions with a lower fire possibility to be included, and the fuzzy model parameters that represent fire colors in each color space. The latter is not as straightforward as the two previous ones, since fine-tuning these parameters requires a deeper analysis to understand if the alterations make sense or not.

Number of superpixels

The number of superpixels N_{sp} can be a decisive parameter as it increases or decreases the allowed number to generate superpixels, impacting the procedure of merging the superpixels which can improve or damage the final output. Figure 5.9 demonstrates a situation where increasing the N_{sp} results in a better and worse final segmentation, respectively.

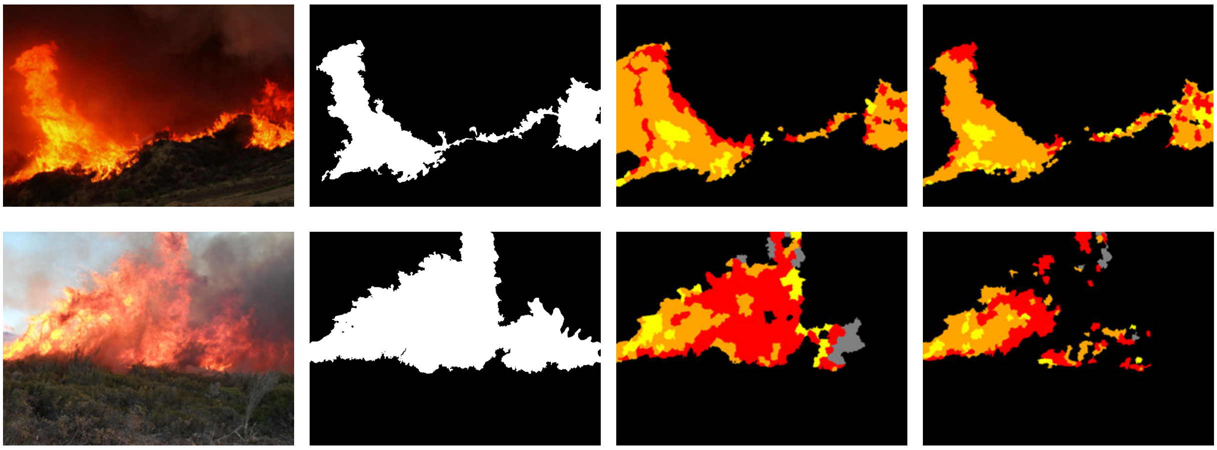
Threshold

Similarly, the threshold value can also change the overall segmentation. In images with high levels of smoke, the modification of such parameter usually enhances the final semantic segmentation (Fig. 5.10c). However, since these images usually register an overall high saturation, the proposed approach classifies most of the improvements to the other class, represented by the grey color. Moreover, the same observations are registered in regions similar to fire colors. Note that the Output (Fig. 5.7c) is



(a) Original Image (b) Ground truth (c) Output - δ = 0.4 (d) Output - δ = 0.5

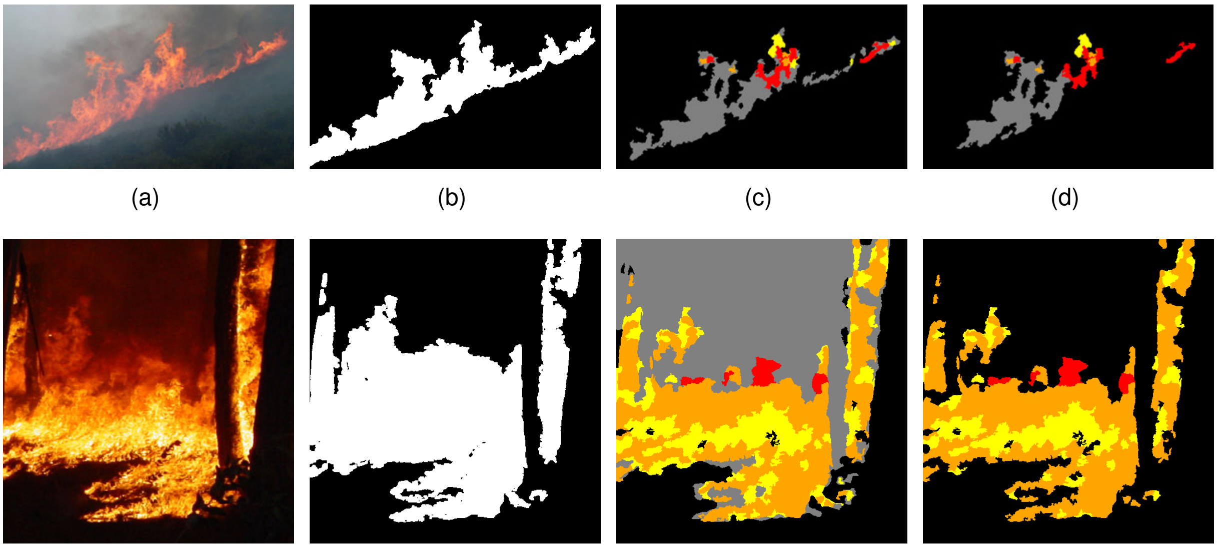
Figure 5.8: Samples where the proposed approaches registers some limitations in the classification.



(a) Original Image (b) Ground truth (c) Output - N_{sp} = 1000 (d) Output - N_{sp} = 2000

Figure 5.9: Samples displaying the number of superpixels influencing the final semantic segmentation.

the intersection between the binary output image (Fig. 5.7b) with the fire colors fuzzy output (Fig. 5.6d). For that reason, using a threshold of 0.4 results in a worse segmented image (Fig. 5.10g).



(a) (b) (c) (d)

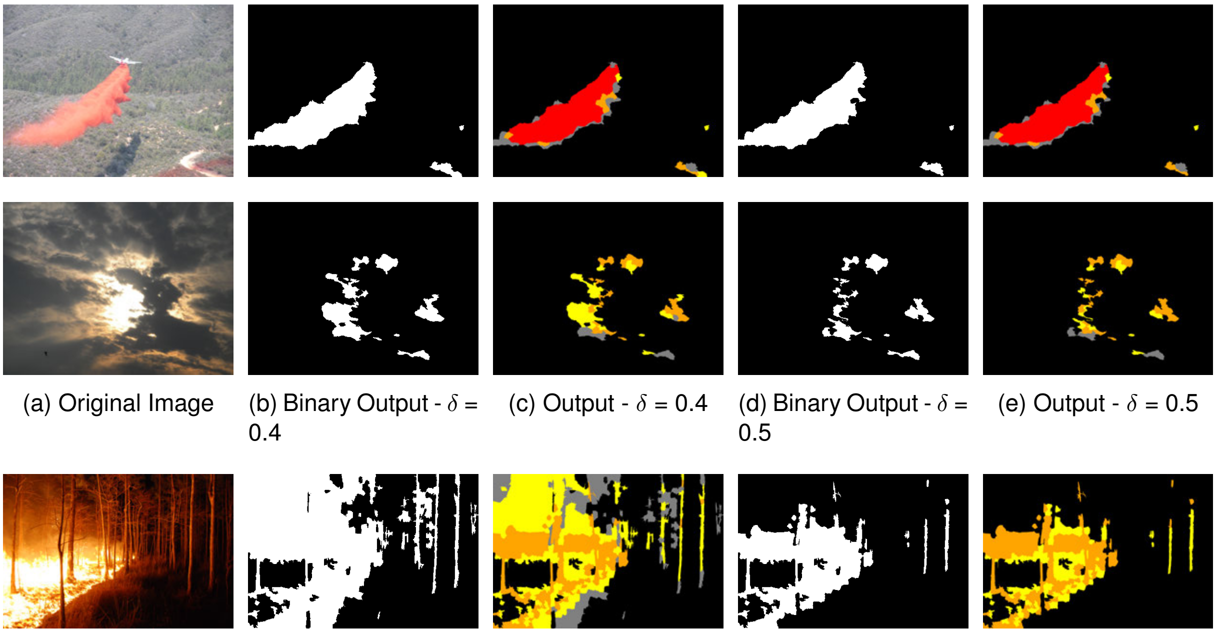
(e) Original Image (f) Ground truth (g) Output - δ = 0.4 (h) Output - δ = 0.5

Figure 5.10: Samples displaying how the threshold affects the final semantic segmentation.

As one can see, this work proposes a technique that is quite interpretable and transparent, enabling the fine-tuning of certain parameters to obtain better segmentations. In fact, by being based on interpretable rule-base models, the proposed approach can enable the expert to validate and improve such results, and generate ground truth data with prominent annotations concerning the fire colors.

5.3.5 Results with the Fire and Smoke Dataset

Even though the Fire and Smoke Dataset (section 3.3) does not provide the ground truth data, this dataset offers other important scenarios that are interesting to analyze. As outlined in Fig. 5.11, the three samples are representative of complex scenes fairly common in a wildfire context. Although images in Fig. 5.11a exhibit similar color shades to the ones displayed in section 5.3.3, these represent different scenarios, namely, an airplane spraying a red flame retartant and the sun, instead of the usually sunset. As it was expected, the proposed approach classified non-fire regions with a high fire possibility as these present similar colors to fire colors. Moreover, changing the threshold value does not improve whatsoever the final segmentation (Fig. 5.11c and 5.11e). Considering that Figure 5.11f displays a wildfire in a night setting, one can see that at a lower threshold the classifier assigns non-fire areas to the fire class $\mathcal{F}$, specially regions close to flame (Fig. 5.11h). In fact, increasing the threshold value, as depicted in Fig. 5.11j, improves the output segmentation.



(a) Original Image

(b) Binary Output - δ = 0.4

(c) Output - δ = 0.4

(d) Binary Output - δ = 0.5

(e) Output - δ = 0.5

(f) Original Image

(g) Binary Output - δ = 0.4

(h) Output - δ = 0.4

(i) Binary Output - δ = 0.5

(j) Output - δ = 0.5

Figure 5.11: Output results for scenarios with fire and non-fire regions from the Fire and Smoke Dataset (section 3.3).

Chapter 6

Conclusions

6.1 Achievements

The two objectives outlined for this work were successfully achieved. Regarding the first objective, the pixel-wise segmentation of the fire can be obtained by specifying a threshold to the image containing the fire color possibilities of each region. The second objective, description of the fire color category, can then be generated by intersecting the pixel-wise segmentation of the fire with the output fire color category of the HSL fuzzy model. The proposed method, the Max Value model, gives the best overall performance of all the baseline approaches. It can cope with a series of different scenarios as it was previously demonstrated. Where previously baseline models had some step backs, the Max Value model was able to overcome those by combining the two fuzzy models. Considering that the proposed approach detects fire colors, it is to be expected that in certain situations it would detect similar fire colors in non-fire objects. However, the interpretability and adaptability of the proposed architecture comes into play, allowing the expert to adjust some parameters to validate and generate reliable ground truth data and respective annotations. The most common parameter that can usually be changed in other fire detection techniques is the classification threshold. As one could see in section 5.3.2, changing the threshold value resulted in better overall results. This parameter is particularly interesting in regions where the fire has smoke over it, since these areas usually output a lower possibility of being fire-colored as the smoke decreases the pixel's saturation. Therefore, decreasing the threshold value enables these regions to be considered. Another parameter, particular to the proposed approach, is the number of superpixels. As it was demonstrated, the number of superpixels can sometimes increase or decrease the overall performance. Finally, the fuzzy model parameters could also be changed in the situation that the previous ones mentioned would not increase the results. However, these are not very straighforward like the threshold or number of superpixels, since the fuzzy model parameters are related to how the HSL and YCbCr color spaces perceive fire colors.

In conclusion, the proposed architecture contributes ot fire detection research, as even though it is not a fire detection approach, it allows to propose automatic fire data annotations for expert validation towards the creation of high-quality large-scale datasets.

The proposed fire data annotation architecture has been submitted to the CVPR conference on the 16th of November 2020 [26].

6.2 Future Work

For future work, at an initial stage, it would be interesting to improve the proposed approach in the outlined limitations. The use of texture feature would be relevant as fire regions have different textures than firefighters or firetrucks. Some examples of texture feature extractors would be the local binary pattern (LBP) or gray level co-occurrence matrices (GLCM). This could be a simple and easy solution towards images containing objects with similar fire colors. The shape feature could be another interesting method to help discerning fire from firefighters and firetrucks with similar fire colors, as fire has a very irregular shape and firefighters and firetrucks usually have a fairly regular shape.

As the next objectives, the development of a similar architecture for smoke colors in order to create smoke data annotations and provide help and improvements in smoke detection techniques as these are particularly important to enable faster fire alarms. It would be interesting to experiment other color spaces to have better a definition of smoke colors and other similar smoke colors in non-smoke objects.

Bibliography

[1] State of California. URL: `https://fire.ca.gov/incidents/2020/`, 2020. Accessed: 2020-12-10.

[2] State of California. Ranch Fire (Mendocino Complex) Information. URL: `https://fire.ca.gov/incidents/2018/7/27/ranch-fire-mendocino-complex/`, 2018. Accessed: 2020-12-10.

[3] D. X. Viegas, M. Figueiredo Almeida, L. M. Ribeiro, J. Raposo, M. T. Viegas, R. Oliveira, D. Alves, C. Pinto, H. Jorge, A. Rodrigues, D. Lucas, S. Lopes, and L. F. Silva. O complexo de incêndios de Pedrógão Grande e concelhos limítrofes, iniciado a 17 de junho de 2017. *Technical Report Forest Fire Research Center, Association for the Development of Industrial Aerodynamics, University of Coimbra*, 2017. [The complex of fires of Pedrógão Grande and bordering counties, begun on June 17, 2017 - written in Portuguese].

[4] O. Hoegh-Guldberg, D. Jacob, M. Taylor, M. Bindi, S. Brown, I. Camilloni, A. Diedhiou, R. Djalante, K. Ebi, F. Engelbrecht, J. Guiot, Y. Hijioka, S. Mehrotra, A. Payne, S. I. Seneviratne, A. Thomas, R. Warren, and G. Zhou. Impacts of 1.5ºc global warming on natural and human systems. *Global warming of 1.5℃. An IPCC Special Report on the impacts of global warming of 1.5℃ above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty[V. Masson-Delmotte, P. Zhai, H. O. Pörtner, D. Roberts, J. Skea, P.R. Shukla,A. Pirani, W. Moufouma-Okia, C.Péan, R. Pidcock, S. Connors, J. B. R. Matthews, Y. Chen, X. Zhou, M. I. Gomis, E. Lonnoy, T. Maycock, M. Tignor, T. Waterfield (eds.)]*, 2018. URL `https://www.ipcc.ch/site/assets/uploads/sites/2/2019/06/SR15_Chapter3_Low_Res.pdf`. In Press.

[5] L. Yu, N. Wang, and X. Meng. Real-time forest fire detection with wireless sensor networks. In *Proceedings. 2005 International Conference on Wireless Communications, Networking and Mobile Computing, 2005.*, volume 2, pages 1214–1217. IEEE, 2005. doi: 10.1109/WCNM.2005.1544272.

[6] V. Khanna and R. K. Cheema. Fire Detection Mechanism using Fuzzy Logic. *International Journal of Computer Applications*, 65(12):5–9, 2013.

[7] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002. doi: 10.1016/S1389-1286(01)00302-4.

[8] M. A. M. Vieira, C. N. Coelho, D. C. da Silva, and J. M. da Mata. Survey on wireless sensor network devices. In *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory*

Automation. Proceedings (Cat. No. 03TH8696), volume 1, pages 537–544, 2003. doi: 10.1109/ETFA.2003.1247753.

[9] T. H. Chen, P. H. Wu, and Y. C. Chiou. An early fire-detection method based on image processing. In *2004 International Conference on Image Processing, 2004. ICIP'04.*, volume 3, pages 1707–1710, 2004. doi: 10.1109/ICIP.2004.1421401.

[10] T. Çelik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu. Fire detection using statistical color model in video sequences. *Journal of Visual Communication and Image Representation*, 18:176–185, 2007. doi: 10.1016/j.jvcir.2006.12.003.

[11] A. Chenebert, T. P. Breckon, and A. Gaszczak. A non-temporal texture driven approach to real-time fire detection. In *2011 18th IEEE International Conference on Image Processing*, pages 1741–1744, 2011.

[12] X. Qi and J. Ebert. A computer vision based method for fire detection in color videos. *International Journal of Imaging*, 2(S09):22–34, 2009.

[13] T. Çelik and H. Demirel. Fire detection in video sequences using a generic color model. *Fire Safety Journal*, 44:147–158, 2009. doi: 10.1016/j.firesaf.2008.05.005.

[14] V. Vipin. Image processing based forest fire detection. *International Journal of Emerging Technology and Advanced Engineering*, 2(2):87–95, 2012.

[15] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh. Deep learning vs. traditional computer vision. In *Science and Information Conference*, pages 128–144, 2019. doi: 10.1007/978-3-030-17795-9_10.

[16] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi: 10.1038/nature14539.

[17] Z. Yin, B. Wan, F. Yuan, X. Xia, and J. Shi. A Deep Normalization and Convolutional Neural Network for Image Smoke Detection. *IEEE Access*, 5:18429–18438, 2017. doi: 10.1109/ACCESS.2017.2747399.

[18] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Computer Vision – ECCV 2014*, pages 818–833. Springer, 2014.

[19] M. J. Sousa, A. Moutinho, and M. Almeida. Wildfire detection using transfer learning on augmented datasets. *Expert Systems with Applications*, 142:112975, 2020. doi: 10.1016/j.eswa.2019.112975.

[20] J. Sharma, O.-C. Granmo, M. Goodwin, and J. T. Fidje. Deep convolutional neural networks for fire detection in images. In *Engineering Applications of Neural Networks*, pages 183–193. Springer International Publishing, 2017. doi: 10.1007/978-3-319-65172-9_16.

[21] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8):5455–5516, 2020.

[22] T. Çelik, H. Özkaramanli, and H. Demirel. Fire and smoke detection without sensors: Image processing based approach. In *2007 15th European Signal Processing Conference*, pages 1794–1798, 2007.

[23] B. C. Ko, S. J. Ham, and J. Y. Nam. Modeling and formalization of fuzzy finite automata for detection of irregular fire flames. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(12): 1903–1912, 2011. doi: 10.1109/TCSVT.2011.2157190.

[24] P. Bolourchi and S. Uysal. Forest fire detection in wireless sensor network using fuzzy logic. In *2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks*, pages 83–87, 2013. doi: 10.1109/CICSYN.2013.32.

[25] J. Chamorro-Martínez, J. M. Soto-Hidalgo, P. M. Martínez-Jiménez, and D. Sánchez. Fuzzy color spaces: A conceptual approach to color visionn. *IEEE Transactions on Fuzzy Systems*, 25:1264–1280, 2017. doi: 10.1109/TFUZZ.2016.2612259.

[26] P. Messias, M. J. Sousa, and A. Moutinho. Color-based superpixel semantic segmentation for fire data annotation. pages 1–8, 2021. Submitted to CVPR on the 16th of November 2020.

[27] T. Young. Bakerian lecture: On the theory of light and colours. *Philosophical transactions of the Royal Society of London*, (92):12–48, 1802. doi: 10.1098/rstl.1802.0004.

[28] H. v. Helmholtz. *Handbuch der physiologischen Optik*, volume 2. Leopold Voss, Leipzig, 1860.

[29] L. Busin, N. Vandenbroucke, and L. Macaire. Color spaces and image segmentation. *Advances in imaging and electron physics*, 151, 2008.

[30] M. D. Fairchild. *Color appearance models*. John Wiley & Sons, 2005.

[31] C. A. Poynton. *A Technical Introduction to Digital Video*. John Wiley & Sons, 1996.

[32] T. Toulouse, L. Rossi, A. Campana, T. Celik, and M. A. Akhloufi. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Safety Journal*, 92:188–194, 2017. doi: 10.1016/j.firesaf.2017.06.012.

[33] P. Neubert and P. Protzel. Superpixel benchmark and comparison. In *Proceedings Forum Bildverarbeitung*, pages 1–12, 2012.

[34] X. Ren and J. Malik. Learning a classification model for segmentation. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 1, pages 10–17, 2003. doi: 10.1109/iccv.2003.1238308.

[35] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 2000. doi: 10.1109/34.868688.

[36] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:2274–2282, 2012. doi: 10.1109/TPAMI.2012.120.

[37] M. Wang, X. Liu, Y. Gao, X. Ma, and N. Q. Soomro. Superpixel segmentation: A benchmark. *Signal Processing: Image Communication*, 56:28–39, 2017. doi: 10.1016/j.image.2017.04.007.

[38] J. Shen, Y. Du, W. Wang, and X. Li. Lazy random walks for superpixel segmentation. *IEEE Transactions on Image Processing*, 23:1451–1462, 2014. doi: 10.1109/TIP.2014.2302892.

[39] Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 583–598, 1991. doi: 10.1109/34.87344.

[40] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002. doi: 10.1109/34.1000236.

[41] Z. Li and J. Chen. Superpixel segmentation using Linear Spectral Clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1356–1363, 2015. doi: 10.1109/CVPR.2015.7298741.

[42] R. Babuška. *Fuzzy Modeling for Control*. Springer, 1998.

[43] J.-S. R. Jang, C.-T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, 1997.

[44] L. A. Zadeh. Fuzzy sets. *Information and control*, 8:338–353, 1965. doi: 10.1016/S0019-9958(65)90241-X.

[45] C. Wang. A study of membership functions on mamdani-type fuzzy inference system for industrial decision-making. Master's thesis, Lehigh University, 2015.

[46] L. A. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3:28–44, 1973. doi: 10.1109/TSMC.1973.5408575.

[47] E. H. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Computers*, (12):1182–1191, 1977. doi: 10.1109/TC.1977.1674779.

[48] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15:116–132, 1985. doi: 10.1109/TSMC.1985.6313399.

[49] A. J. Dunnings and T. P. Breckon. Experimentally defined convolutional neural network architecture variants for non-temporal real-time fire detection. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 1558–1562, 2018. doi: 10.1109/ICIP.2018.8451657.

[50] C. R. Steffens, R. N. Rodrigues, and S. S. da Costa Botelho. Non-stationary vfd evaluation kit: Dataset and metrics to fuel video-based fire detection development. In *Robotics*, pages 135–151. 2016. doi: 10.1007/978-3-319-47247-8_9.

[51] J. Chamorro-Martínez and J. M. Keller. Granular modelling of fuzzy color categories. *IEEE Transactions on Fuzzy Systems*, pages 1–1, 2019. doi: 10.1109/tfuzz.2019.2923966.

[52] P. Jaccard. The distribution of the flora in the alpine zone. *New phytologist*, 11(2):37–50, 1912. doi: 10.1111/j.1469-8137.1912.tb05611.x.

Appendix A

Publications

The article below is the scientific article that this work led to and it has been submitted on the 16th of November 2020.

Color-based Superpixel Semantic Segmentation for Fire Data Annotation

Pedro Messias[1] Maria João Sousa[2] Alexandra Moutinho[2]

[1] Instituto Superior Técnico, Universidade de Lisboa
[2] IDMEC, Instituto Superior Técnico, Universidade de Lisboa
{pedro.messias,maria.joao.sousa,alexandra.moutinho}@tecnico.ulisboa.pt

Abstract

Image-based fire detection is a safety-critical task, which requires high-quality datasets to ensure performance guarantees in real scenarios. Automatic fire detection systems are in ever-increasing demand, but the limited number and size of open datasets, and lack of annotations, hinder model development. Solving this issue requires that experts dedicate a significant time to classify and segment fire events in image datasets. Towards building large-scale curated datasets, this paper presents a data annotation method that leverages semantic segmentation based on superpixel aggregation and color features. The approach introduces interpretable linguistic models that generate pixel-wise fire segmentation and annotations, which are explainable through simple fine-tunable rules that can support subsequent annotation validation by fire domain experts. The performance of the proposed algorithm is evaluated for relevant scenarios using a publicly available dataset, namely through the assessment of the segmentation quality and the labeling of fire color categories. The outcomes of this approach pave the way for creating large-scale datasets that can empower future deployments of learning-based architectures in fire detection systems.

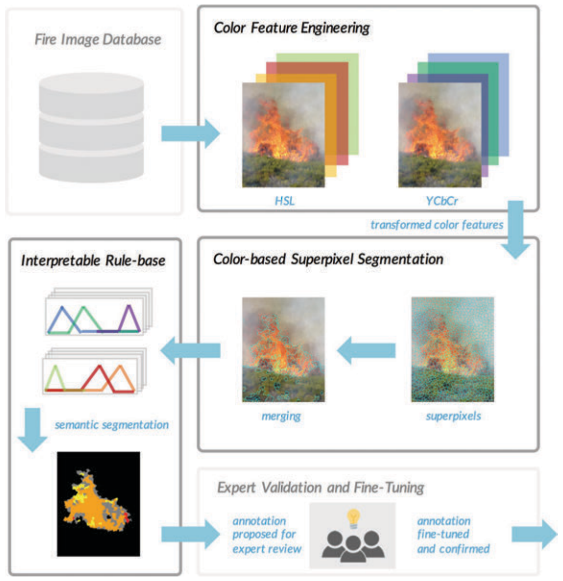

Fire Image Database
Color Feature Engineering
HSL
YCbCr
transformed color features
Interpretable Rule-base
Color-based Superpixel Segmentation
merging
superpixels
semantic segmentation
Expert Validation and Fine-Tuning
annotation proposed for expert review
annotation fine-tuned and confirmed

Figure 1: **Overview of the fire data annotation pipeline.** Our approach exploits color-based features across the HSL and YCbCr color spaces to design statistical variables with significance for fire detection. Fire data annotation is performed with superpixel segmentation in combination with interpretable linguistic models.

1. Introduction

The challenges in facing wildfires are increasing the demand for automatic surveillance systems. However, current systems are typically prone to false alarms in real deployments, thus unreliable to response teams because even false positives hinder the confidence in these solutions. Although several research efforts have attempted to leverage the advances in deep learning in classification and segmentation tasks, the performance and evaluation are hampered by the quality of existing image datasets. Limitations include, e.g., the low number of samples and quality of image data, lack of representativity of real-world situations, or being heavily based on video frames, reducing variability. Moreover, datasets are often missing annotations, limiting both the performance and the information extracted for real operations.

To address this issue, this paper shifts focus from the usual fire detection architectures, to instead target the development of a pipeline for fire data annotation through semantic segmentation. Towards creating large-scale high-quality annotated datasets, this paper presents a method to generate fine-grained fire segmentations along with fire color labels, which can subsequently be validated by fire domain experts to produce reliable ground truth data. The proposed approach, depicted in Fig. 1, leverages the feature rich representations of fire, namely in the HSL and YCbCr color spaces, that allow an insightful interpretation that informs the color-based superpixel segmentation. Subsequently, interpretable rule-based linguistic models are employed for pixel-wise classification, using statistical color attributes to infer which superpixels correspond to fire or not, and generate semantic labels descriptive of the colors represented. Our approach is evaluated against the Corsican Fire Database, demonstrating excellent results in fine-grained segmentations. The proposed solution can cope with a wide array of real-contexts: e.g., with fire at long-distances, with firefighters in the field of operations, and in smoke situations. Where the performance might exhibit some limitations, the interpretability and flexibility of this approach come into play, allowing the expert to intuitively fine-tune the model output to improve the segmentation, and thus guaranteeing the expert confidence in the annotation process. This novel approach extends the state-of-the-art on fine-grained fire semantic segmentation for data annotation, which we believe is an instrumental step towards building large-scale datasets for safety-critical deployments.

2. Related work

Image-based Fire Detection approaches have widely explored color-based techniques [6, 14], and recent works have adopted learning-based methods [3, 9, 13]. Herein, we concentrate on image data from the visible spectrum because its widespread availability allows a broader deployment. Contributions in this domain targeting fire event detection generally design techniques for either flame or smoke [17]. Most center on a single application, e.g., ground operations, watchtower systems, or aerial surveillance [8]. That means methods are considerably different, use distinct types of data sources, thus making these hard to compare generalization-wise. For learning-based approaches, the scarce availability of curated datasets currently presents a significant roadblock for scaling up their applications [2, 13, 16]. Aiming for better generalization and broader context applicability, this work pivots the attention to approaches that enable building large-scale datasets, focusing on flame detection.

Data annotation is paramount for improving algorithm performance and embedding relevant information for this application. However, the tendency in fire detection works has favored computational efficiency [11], rather than segmentation granularity [14, 7], hence being blindsided to its key role. In contrast, our objective is to prioritize fine-grained segmentation to streamline the data annotation process. Although state-of-the-art methods can support this goal, the task of defining ground truths for fire data with meaningful information involves a high level of complexity and nuance.

Describing colors of fire in the wild calls for expert knowledge from fire domain specialists because it involves a deep understanding of the natural and artificial fuels burning [5]. The role of this characterization has utmost importance both in wildfire science and wildfire operations, e.g., as it relates to fire intensity and severity. However, hand-crafted pixel-wise annotation is extremely time consuming, therefore semantic segmentation tools are crucial to simplify this procedure and assist experts in developing ground truths.

Model interpretability. Being the experts involved in the validation of algorithm outputs, the fine-tuning capacity and interpretability of the models are essential to leveraging their relevant inputs. For this reason, in this work, we privilege interpretable ruled-based models that are fine-tunable if and whenever needed. In this way, these systems provide higher flexibility in an early stage than black-box models, which could require extensive manual correction cycles until starting generating high-quality outputs. That problem could discourage expert commitment in the data curation process, which is avoided using our explainable and interpretable approach.

3. Dataset

The **Corsican Fire Database** (CFDB) is a public database of wildfire images with several labels that allows the evaluation and comparison of algorithms related to wildfire detection [15]. As of this writing, despite the authors intention for an evolving dataset, where users can upload new data for categorization, it remains composed of initial data, attesting to the difficulty in developing these repositories. The database comprises 500 images in the visible spectrum, 100 pairs of visible and near-infrared images, and 5 multi-modal sequences with visible and near-infrared pairs. We benchmark our approach using a part of this database.

Fire Image Dataset. A reduced dataset was used with 207 images from the 500 visible subset of the CFDB to create a representative dataset with different scenarios (Fig. 2).

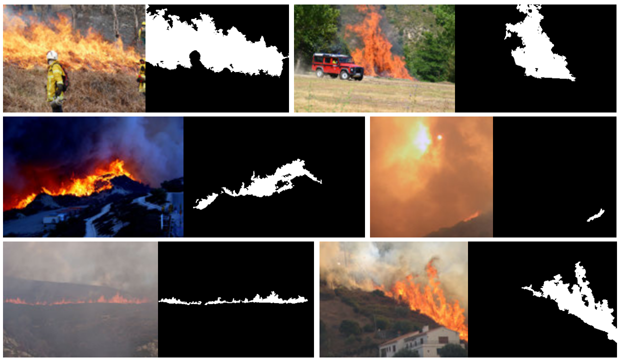


Figure 2: Side-by-side samples of fire images (colored) and respective ground truths (binary) of the reduced Fire Image Dateset used, including firefighting and wildland-urban-interface elements, and varying visibility conditions, e.g., day, sunset, night and with smoke.

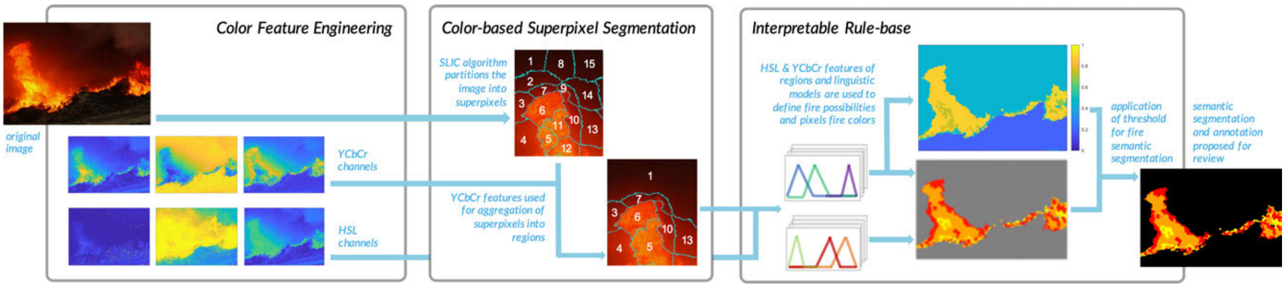

Color Feature Engineering
original image
YCbCr channels
HSL channels
Color-based Superpixel Segmentation
SLIC algorithm partitions the image into superpixels
1
8
15
2
7
9
14
3
6
10
11
13
4
5
12
YCbCr features used for aggregation of superpixels into regions
1
7
3
6
10
13
4
5
Interpretable Rule-base
HSL & YCbCr features of regions and linguistic models are used to define fire possibilities and pixels fire colors
application of threshold for fire semantic segmentation
semantic segmentation and annotation proposed for review

Figure 3: **Fire Data Annotation Pipeline.** The proposed architecture is comprised of three core parts: *i)* Color Feature Engineering (Section 4.1), *ii)* Color-based Superpixel Segmentation (Section 4.2), and *iii)* Interpretable Rule-base (Section 4.3). The first layer transforms the image data to the HSL and YCbCr color spaces, better suited for fire instances. The second layer uses a purpose-built segmentation method that generates superpixels and merges them into regions according to statistical color-based features. Then, the interpretable rule-base employs linguistic models to classify the merged regions, yielding the pixel-wise segmentation of fire and the corresponding color labels for each region. Subsequently, the results can be reviewed by experts to validate and fine-tune new ground truths for fire image data.

Samples were excluded due to low resolution, noise or image artifacts, or pixelization that reduced the granularity of fire features. From these 207 images, 50 were used to develop and test the algorithm while the remaining were used only for testing. Images without fire were not added since important real-world contexts, e.g., sunsets, and presence of firefighting means are already portrayed in this dataset.

4. Fire Data Annotation Pipeline

Data annotation can be addressed using semantic segmentation methods. In this work, we pose this problem under a two-stage approach: *i) segmentation* and *ii) classification*. The first concerns the segmentation of image data that can be handled with, e.g., partitioning algorithms. The second deals with the classification of segmented regions through assignment to categories according to their attributes. The objectives of the proposed fire data annotation pipeline outlined in Fig. 3 are twofold: *i)* pixel-wise segmentation of fire and *ii)* description of the fire color category.

Problem Formulation: Consider a sample image, I, encoded in a preset color space domain defined as $\mathcal{D} \in \mathbb{R}^c$, where c represents the number of channels. Let X represent the space of image pixels and x represent a pixel of the image, $x \in X$. To address the first objective of performing a pixel-wise segmentation of fire in an image, let $\mathcal{F}$ represent the set of pixels belonging to the fire class, and $\mathcal{N}$ consist of the pixels that do not depict fire. The *ground truth* defines the expert validated data, where both classes are, in this case binary and mutually exclusive, i.e. $x \in \mathcal{F}$ or $x \in \mathcal{N}$. Regarding the second objective of describing the color of fire, we model four color categories, namely red, orange, yellow and other. Note that, unlike the first case, sets of pixels belonging to a color subset $\{\mathcal{C}_{\text{red}}, \mathcal{C}_{\text{orange}}, \mathcal{C}_{\text{yellow}}, \mathcal{C}_{\text{other}}\}$ are harder to define in a crisp way.

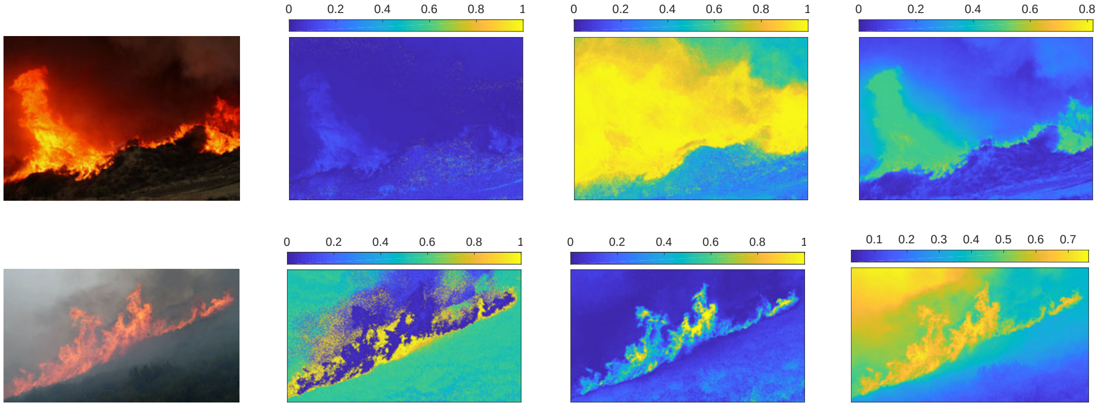
The following sections detail the three layers of the proposed architecture (Section 4.1 - 4.3), and the implementation details are discussed in Section 4.4.

4.1. Color Feature Engineering

Seeing Fire Across Color Spaces. As the objective is to segment the flame based on color, choosing relevant color spaces can be very helpful when defining parameters that correspond to fire colors, thus, leading to better results. This is a particularly important step for images with similar fire colors in non-fire regions and when there is smoke over the flame, decreasing the perception of the fire colors even for human annotation. For these reasons, the color spaces used are the HSL and the YCbCr. The HSL color space is easy to use and works well in scenarios with a high contrast between the flame and the background. The saturation and lightness channels are more intuitive in defining fire colors and separating them from dark smoke (high saturation) and clouds (low lightness). Moreover, the hue channel makes it easier to specify the range of colors for the linguistic terms (e.g., red, orange, yellow). However, this color space is more challenging when it comes to images with smoke in the scene or regions with similar colors to fire colors (Fig. 4). In these situations, the YCbCr color space allows for an easier flame segmentation (Fig. 5) because of its ability to separate the colors, but it is less interpretable than the HSL making its development more complex. Color-based features derived from these datasets are employed in the two stages of the proposed pipeline, namely in the segmentation and classification parts.

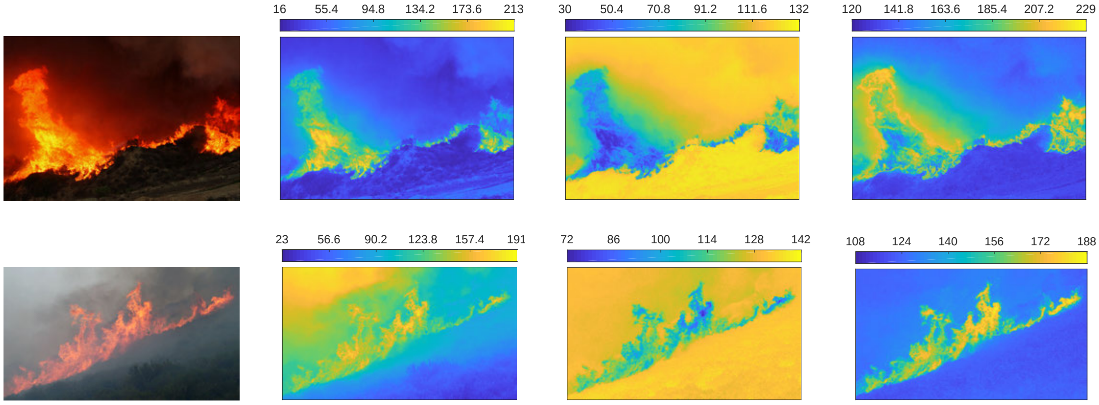
4.2. Color-based Superpixel Segmentation

The use of superpixels allows to segment an image by clustering pixels based on their color and proximity. This technique can be very useful since superpixels can carry more information than pixels [12], adhere better to the image boundaries and reduce the complexity of several image



(a) Original Image (b) Hue (c) Saturation (d) Lightness

Figure 4: **Fire Features in HSL.** Visual display of each HSL color channel for two different images.



(a) Original Image (b) Luminance (c) Chrominance Blue (d) Chrominance Red

Figure 5: **Fire Features in YCbCr.** Visual display of each YCbCr color channel for two different images.

processing operations [1], thus decreasing processing times.

Superpixel Algorithm. The superpixels are generated using the simple linear iterative clustering (SLIC) algorithm [1] that allows the specification of both the desired *number of superpixels*, N_{sp}, and their *compactness*, C. The number of superpixels drives the granularity of the image partitioning, being higher with the increase in N_{sp}. In turn, the compactness controls the shape of each element, with higher values creating more regularly shaped superpixels (square like), and lower values creating more irregular shapes, which adhere better to intricate image boundaries. Furthermore, the superpixels are defined as vectors in $\mathbb{R}^3$ in both HSL and YCbCr color spaces, where each entry corresponds to the mean color of each channel (H_j^{sp}, S_j^{sp}, L_j^{sp} or Y_j^{sp}, Cb_j^{sp}, Cr_j^{sp}) of the superpixel sp, with $j \in [1, N_{\text{sp}}]$.

Merging Superpixels. We propose a procedure that merges similar superpixels into regions. This method combines neighboring superpixels that register a similar color shade. This is performed using the YCbCr color features as these allow the separation of fire from other instances like smoke. Adjacent superpixels, j and i, are compared based on their mean color ($Y_{j,i}^{\text{sp}}$, $Cb_{j,i}^{\text{sp}}$, $Cr_{j,i}^{\text{sp}}$) and merged if each entry of the pairwise difference is lower or equal to a threshold $\{0.034, 0.1, 0.03\}$, as follows:

$$\mid Y_j^{\text{sp}} - Y_i^{\text{sp}} \mid \leq 0.034 \tag{1}$$

$$\mid Cb_j^{\text{sp}} - Cb_i^{\text{sp}} \mid \leq 0.1 \tag{2}$$

$$\mid Cr_j^{\text{sp}} - Cr_i^{\text{sp}} \mid \leq 0.03 \tag{3}$$

To exemplify this procedure, Fig 6 illustrates the steps of the process and the final result of merging the superpixels. Such operation allows the number of regions to decrease and, consequently, the processing time, and eliminate certain superpixels that could be classified as fire colour.

4.3. Interpretable Rule-based Models

Color Perception. Humans describe colors using linguistic terms like red, green or pink, but color perception might

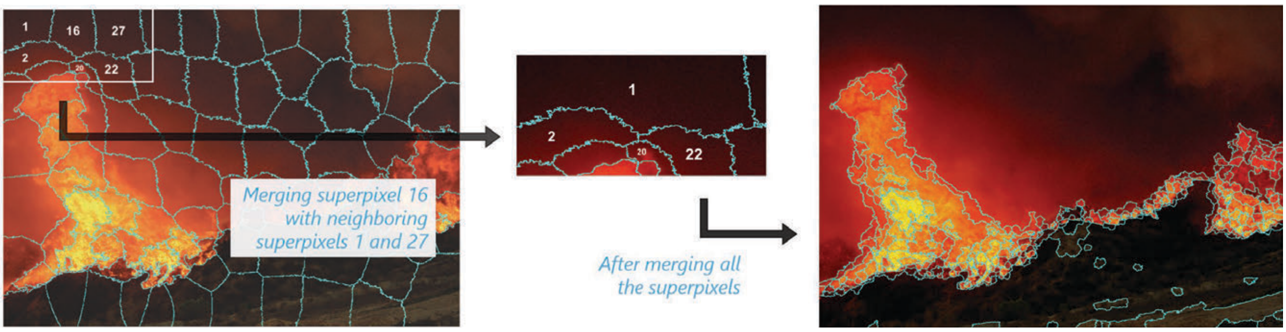

1
16
27
2
20
22
Merging superpixel 16 with neighboring superpixels 1 and 27
1
2
20
22
After merging all the superpixels

Figure 6: Zoomed area shows how similar superpixels 1, 16 and 27 merge, creating region number 1. Merging process results in a region-defined image.

differ from person to person [4]. Likewise, image retrieval for search-based analyses is also based on key categorical terms, namely color. Relevant fire characteristics are related to their color so the annotation of these attributes is fundamental towards creating large-scale datasets with relevant information that can be used in a wide range of fire detection scenarios.

Linguistic Models Architecture. In this work, we propose interpretable linguistic models, designed for fire segmentation and classification of the superpixel regions yielded from the previous step. The rule-based architecture is build with Mamdani-type fuzzy inference systems [10], that describe the rules of the knowledge base with linguistic terms. This architecture incorporates uncertainty and is able to bridge the gap between semantic description of colors and its numerical parametrization. The concept of the rules relies on the association of the linguistic terms between both the modeling of the color-based features and the categorization of colors, with the underlying range of values. Our approach leverages two complementary models, developed for the HSL and YCbCr color spaces and outlined in Table 1. Both models are defined with three inputs, corresponding to the mean color of each channel for every region. The two models output a fire possibility per region, that is leveraged to perform the classification of the merged superpixels and achieve the pixel-wise segmentation of fire in the images. In addition, the HSL model is able to describe fire color categories to perform semantic segmentation of the colors in the image. The proposed architecture may integrate both models in the data annotation pipeline (Fig.3), combining the HSL and YCbCr using a weighted average or maximum operators, to generate a segmentation of fire. In parallel, the HSL model generates fire color categories that can achieve the final semantic layers that also describe the color of the fire.

The **Interpretable Rule-base** is built with simple and intuitive triangular and trapezoidal parametric functions, that model the corresponding linguistic terms for inputs and outputs as follows.

HSL model: uses terms describing levels of hue, saturation and lightness to model the fire possibility, i.e. low or high, and the fire color category, which classifies a region to a

Table 1: **Interpretable Linguistic Models.** Description of the ruled-based systems designed for fire data annotation, using the HSL and YCbCr color-features. The classification models have three inputs and one common output. Each input/output is defined with different linguistic terms and parameters, forming membership functions. There are triangular (3 parameters) and trapezoidal (4 parameters) membership functions. The HSL model has one additional output, fire color, to generate fire color category.

	Input			Output		
Model	Variable	Linguistic terms	Parameters	Variable	Linguistic terms	Output parameters
HSL	Hue	red1, orange, yellow, other, red2	$[0, 0, 0.03, 0.055]$; $[0.04, 0.09, 0.133]$; $[0.11, 0.16, 0.2]$; $[0.17, 0.25, 0.87, 0.96]$; $[0.9, 0.97, 1, 1]$;	fire possibility	low, high	$[0, 0, 0.3, 0.5]$; $[0.4, 0.7, 1, 1]$;
	Saturation	low, high	$[0, 0, 0.4, 0.65]$; $[0.545, 0.75, 1, 1]$;	fire color	red, orange, yellow, other	$[0.5, 1, 1.75]$; $[1.25, 2, 2.75]$; $[2.25, 3, 3.75]$; $[3.25, 4, 4.5]$;
	Lightness	low, medium, high	$[0, 0, 0.23, 0.39]$; $[0.23, 0.427, 0.85, 0.96]$; $[0.94, 0.965, 1, 1]$;			
YCbCr	Luminance	low, medium, medium high, high	$[0, 0, 0.365, 0.49]$; $[0.457, 0.5, 0.548, 0.594]$; $[0.548, 0.63, 0.72, 0.776]$; $[0.73, 0.8, 1, 1]$;	fire possibility	low, medium, medium high, high	$[0, 0, 0.23, 0.33]$; $[0.27, 0.35, 0.53, 0.65]$; $[0.6, 0.65, 0.75, 0.8]$; $[0.75, 0.83, 1, 1]$;
	Chrominance Blue	low, medium, high	$[0, 0, 0.435, 0.56]$; $[0.47, 0.527, 0.58, 0.6]$; $[0.446, 0.69, 1, 1]$;			
	Chrominance Red	low, medium, high	$[0, 0, 0.49, 0.625]$; $[0.58, 0.647, 0.69, 0.78]$; $[0.71, 0.826, 1, 1]$;			

corresponding color subset $\{\mathcal{C}_{\text{red}}, \mathcal{C}_{\text{orange}}, \mathcal{C}_{\text{yellow}}, \mathcal{C}_{\text{other}}\}$.

YCbCr model: employs terms describing degrees of luminance, chrominance blue and chrominance red. The knowledge-base comprises fourteen rules. The outputs of the rules are combined and transformed to a crisp value representing the fire color possibility.

The models generate continuous-valued outputs that are converted to multi-class labels through the application of rounding thresholds. For classification of a region as being fire the decision threshold, δ, is usually applied at 0.5. This value can be fine-tuned in the validation procedure as will be discussed further along.

4.4. Implementation details

The proposed algorithm allows the flexible selection of three parameters. For segmentation (Section 4.2), the number of superpixels, N_{sp}, and the compactness, C, which are defined within the algorithm. For classification (Section 4.3), a threshold is applied to the outputs generated in the semantic classification of fire and the multi-class color categories. These parameters are established in the algorithm but can be easily fine-tuned by experts upon validation.

Parameter selection: Concerning the segmentation part, the selected number of superpixels must ensure that the algorithm can achieve a fine-grained segmentation. The number of superpixels drives the quality of the segmentation in this regard. Since, by nature, flame shapes are very irregular and the image data might contain regions of interest that are captured at long distances, if N_{sp} is too small the image partitioning results in larger superpixels that do not adhere exclusively to the flames. This behavior is illustrated in Fig. 7, where in the lower left corner of the samples presented we can distinctly observe that the superpixels can capture the flames but also aggregate other information nearby. This would inherently degrade the quality of the segmentation, but more importantly, it could prevent an accurate semantic segmentation because a misleading mean color value of the superpixel could result in its misclassification. However, selecting higher values of N_{sp} results in a larger number of increasingly small superpixels, as depicted in Fig. 7, which are harder to classify using the mean color statistics as these capture less context information. The value established by default in our algorithm is 1000 as it is considered an adequate trade-off between these factors. Regarding the compactness, since it controls the shape of the superpixels, it is particularly relevant when segmenting irregular shapes like fire. The influence of varying this parameter can be observed notably in Fig. 7, by comparing the samples on the upper right corner of left and right side images. The effect of enforcing a higher compactness (depicted on the right side) could result in less fine-grained semantic segmentation for both fire and fire colors. For this reason, the value of C was established as 1, because it is the lowest value possible, making superpixels adhere better to irregular boundaries.

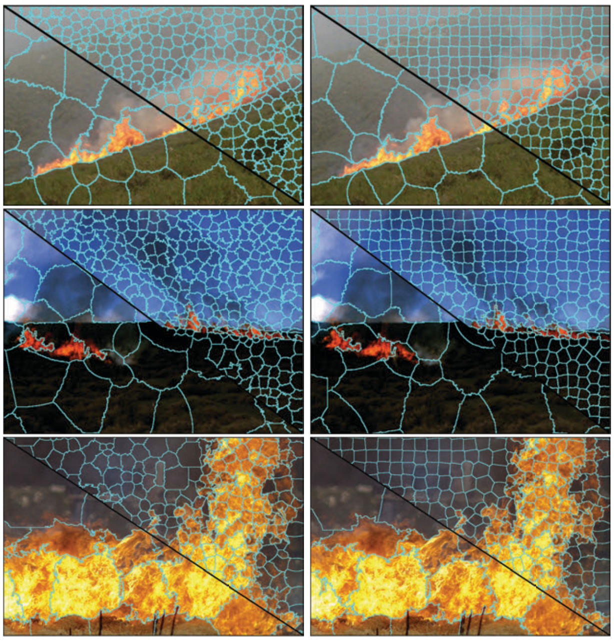


Figure 7: Visual comparison between different N_{sp} and C. The N_{sp} in the lower left corner of each image is 100 and 2000 in the upper right corner. Images on the left have a C equal to 1 and images on the right to 20.

Considering the proposed pipeline integrates two ruled-based models, there are two possible values for the classification of the input in terms of the possibility of corresponding to a color similar to fire colors. The ensemble classier uses as decision function the weighted average or maximum operators, being possible to define the weights and the strategy to apply. The proposed solution is intended to be fine-tunable, so while a threshold on fire classification is set typically at 0.5, this can still be adjusted if required. Our experiments use always a $\delta = 0.5$ threshold for the fire segmentation, except when specified otherwise.

5. Experiments

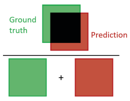
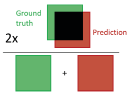
5.1. Performance Evaluation

The Fire Image Dataset (Section 3) is a subset of the Corsican Fire Database (CFDB), which provides pixel-level segmentations annotated by a single expert as reported in [15]. While the definition of the regions corresponding to fire are complex to define and encompass a great degree of uncertainty, herein we consider this unique source our ground truth.

Baseline: To evaluate the proposed architecture we assess several baseline approaches. The first and simpler ones are based on the **HSL model** and the **YCbCr model** separately. We test the integrated approach, with the combination of the two models with a **Weighted** average ensemble classifier, considering several weight distributions. Subsequently, we also present results for the multi-model approach with an ensemble classification based on the **Max Value** operator.

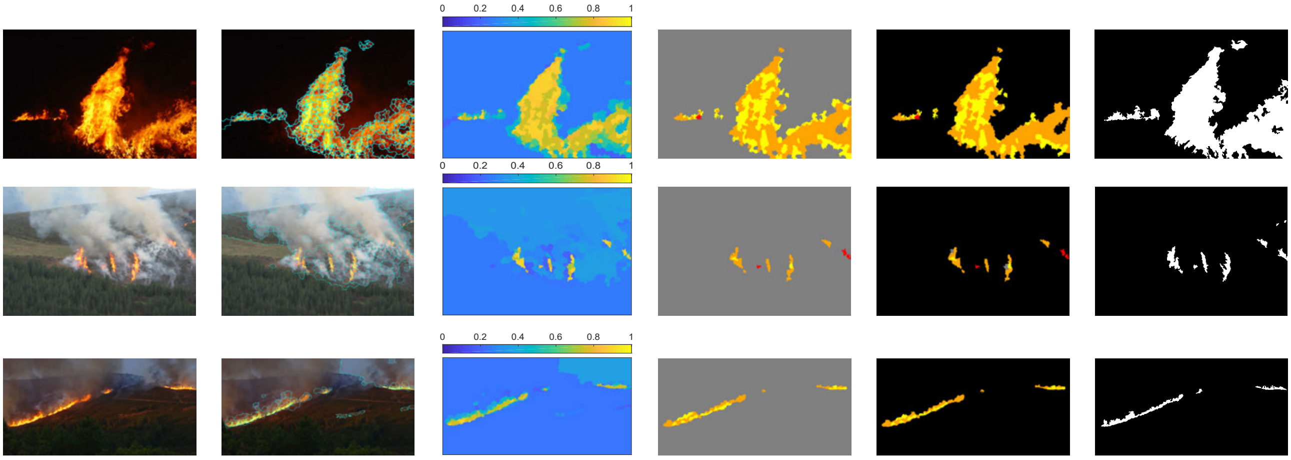
Performance Measures: To evaluate the performance of the segmentation algorithms, we will used a set of different metrics that enable the comparison with the ground truth data. Table 2 summarizes their formulas and a visual intuition of each. For model assessment we will focus on three metrics namely: Accuracy, Intersection over Union (IoU), and Dice coefficient. The metrics consider: true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). The words true and false refer to whether the positives and negatives were correctly classified. For the fire data annotation purpose we are interested in obtaining segmentations with rich information. Considering the outputs will be subsequently validated by experts, the oversegmentation while not ideal is not such a problem, since these can be easily corrected through fine-tuning or by specifying which regions of superpixels were misclassified and annotating the true label. In the following we evaluate cases on interests that showcase the capabilities of the proposed method, along with the identification of the most challenging scenarios that are likely to require the expert annotation.

Table 2: Performance measures for segmentation.

Accuracy	IoU	Dice
$\frac{TP+TN}{TP+FN+FP+TN}$	$\frac{TP}{TP+FP+FN}$	$\frac{2TP}{2TP+FP+FN}$
		

5.2. Results Evaluation

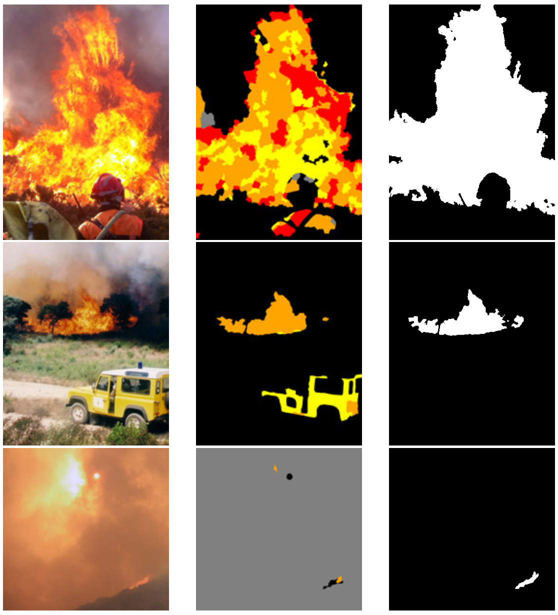
To demonstrate the capabilities of the proposed algorithm, a set of representative examples was selected for analysis. The examples depicted in Fig. 8, showcase the performance of the internal steps of the semantic segmentation for three distinct scenarios. The figure represents the pipeline by the first presenting the original of each image, followed by the merged superpixel segmentation. This step already was designed to target the separation from fire from the surrounding context so it already gives a close and granular partitioning of the image, capturing the fire instances even when these are at long distances an covered by smoke. Next, the ruled-based models generate the classification of the superpixels according to fire possibility, with higher values (brighter in the image), representing the superpixels that will be classified with the fire label. The HSL linguistic model also classifies the fire colors in the image, which relates to fire intensity and severity. By comparison with the original image, it can be observed the superpixel approach combined with the fire color classification is able to identify all possible colors in the first two sample images, and correctly attributes the other color label to the remainder of the scene. By intersecting both the fire classification with the color classification, we can automatically generate fire data annotations that closely approximate the expert annotated data, but with richer information concerning the color characteristics.



(a) Original Image (b) Merged (c) Fire Possibility (d) Fire Color (e) Output (f) Ground truth

Figure 8: Examples for each step in real-world scenarios

Regarding limitations of the color-based approach, as expected scene objects resembling fire colors are more likely to be captured by the propose segmentation. However, such cases as depicted in Fig. 9, like the sunset and firefighting elements are normally likely to require expert annotation, or need to be complemented by other algorithms for detection of other objects in the scene.



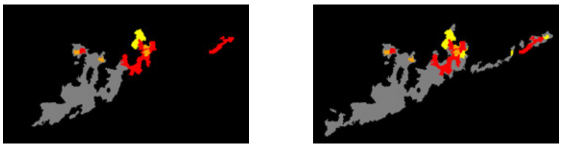
(a) Original (b) Output (c) Ground Truth

Figure 9: Examples of limitations in real-world scenes.

5.3. Understanding the Architecture

Fine-tuning: While most parameters in the architecture are generically preset, the proposed architecture is interpretable and can be fine-tuned according to the more complex scenarios. For instance, for the image represented in Fig. 4 obscured by smoke, the fine-tuning of the classification threshold can improve the final semantic segmentation as is illustrated in Fig. 10.

Ablation Study: The design process described throughout this paper was evaluated with the incremental assembly of its building blocks. Table 3 presents the ablation study with the overall results for each model, including variants of the pipeline proposed for semantic segmentation of fire.



(a) Threshold, $\delta = 0.5$ (b) Threshold, $\delta = 0.4$

Figure 10: Fine-tuning of fire classification threshold for improving the semantic segmentation.

Table 3: **Ablation Study.**

Model		δ	Accuracy	IoU	Dice
HSL	HSL model	0.5	93.39	62.49	73.92
YCbCr	YCbCr model	0.5	91.58	56.68	69.82
Weighted	0.4 HSL + 0.6 YCbCr	0.5	93.16	61.52	73.66
	0.3 HSL + 0.7 YCbCr	0.5	93.01	61.00	73.29
	0.2 HSL + 0.8 YCbCr	0.5	92.81	60.13	72.63
Max Value	max(HSL,YCbCr)	0.5	93.47	66.51	77.59
		0.4	**94.04**	**73.53**	**82.63**

6. Conclusion

This paper introduces a novel pipeline for fire data annotation that enables semantic segmentation of fire and the pixel-wise annotation of relevant color characteristics. The architecture proposed leverages a purpose-built algorithm that combines color-based superpixel segmentation with interpretable rule-based models that allow generating pixel-wise semantic labels of fire and of fire colors in the images. We demonstrate that the proposed approach is able to obtain fine-grained semantic segmentations and discuss the limitation in challenging real-world scenarios. Our solution has key advantages due to its fine-tuning ability and interpretable nature, which enables the close involvement of fire domain experts in the validation of new ground truth data for a broad array of fire detection applications. Our approach to fire data annotation aims to streamline this procedure, towards the creation of high-quality large-scale datasets that can allow robust deployments in safety-critical real-world scenarios.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:2274–2282, 2012. 4

[2] T. Adão, T. M. Pinho, L. Pádua, N. Santos, A. Sousa, J. J. Sousa, and E. Peres. Using virtual scenarios to produce machine learnable environments for wildfire detection and segmentation. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3/W8:9–15, 2019. 2

[3] Fengju Bu and Mohammad Samadi Gharajeh. Intelligent and vision-based fire detection systems: A survey. *Image and Vision Computing*, 91:103803, 2019. 2

[4] Jesús Chamorro-Martínez and James M Keller. Granular modelling of fuzzy color categories. *IEEE Transactions on Fuzzy Systems*, pages 1–1, 2019. 5

[5] John Dehaan, Paul Kirk, and David Icove. *Kirk's fire investigation*. Pearson-Prentice Hall, 05 2011. 2

[6] XF. Han, J.S. Jin, and M.J. Wang et al. Video fire detection based on gaussian mixture model and multi-color features. *Signal, Image and Video Processing*, 11:1419–1425, 2017. 2

[7] H. Harkat, J. Nascimento, and A. Bernardino. Fire segmentation using a DeepLabv3+ architecture. In Lorenzo Bruzzone, Francesca Bovolo, and Emanuele Santi, editors, *Image and Signal Processing for Remote Sensing XXVI*, volume 11533, pages 134 – 145. International Society for Optics and Photonics, SPIE, 2020. 2

[8] Christos Kyrkou and Theocharis Theocharides. Deep-learning-based aerial image classification for emergency response applications using unmanned aerial vehicles. In *CVPR Workshops*, pages 517–525, 2019. 2

[9] Pu Li and Wangda Zhao. Image fire detection algorithms based on convolutional neural networks. *Case Studies in Thermal Engineering*, 19:100625, 2020. 2

[10] Ebrahim H Mamdani. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Computers*, (12):1182–1191, 1977. 5

[11] K. Muhammad, S. Khan, M. Elhoseny, S. Hassan Ahmed, and S. Wook Baik. Efficient fire detection for uncertain surveillance environment. *IEEE Transactions on Industrial Informatics*, 15(5):3113–3122, 2019. 2

[12] Peer Neubert and Peter Protzel. Superpixel benchmark and comparison. In *Proceedings Forum Bildverarbeitung*, pages 1–12, 2012. 3

[13] Maria João Sousa, Alexandra Moutinho, and Miguel Almeida. Wildfire detection using transfer learning on augmented datasets. *Expert Systems with Applications*, 142:112975, 2020. 2

[14] T. Toulouse, L. Rossi, M. Akhloufi, T. Celik, and X. Maldague. Benchmarking of wildland fire colour segmentation algorithms. *IET Image Processing*, 9(12):1064–1072, 2015. 2

[15] Tom Toulouse, Lucile Rossi, Antoine Campana, Turgay Celik, and Moulay A Akhloufi. Computer vision for wildfire research: An evolving image dataset for processing and analysis. *Fire Safety Journal*, 92:188–194, 2017. 2, 6

[16] Qi xing Zhang, Gao hua Lin, Yong ming Zhang, Gao Xu, and Jin jun Wang. Wildland forest fire smoke detection based on faster r-cnn using synthetic smoke images. *Procedia Engineering*, 211:441 – 446, 2018. 2017 8th International Conference on Fire Science and Fire Protection Engineering (ICFSFPE 2017). 2

[17] Zhiqiang Zhou, Yongsheng Shi, Zhifeng Gao, and Sun Li. Wildfire smoke detection based on local extremal region segmentation and surveillance. *Fire Safety Journal*, 85:50 – 58, 2016. 2