Analysis of the effectiveness of fire region detection using histogram backprojection

Caroline Bridge
CIS*4720: Image Processing and Vision
University of Guelph
Guelph, ON, Canada
bridge@uoguelph.ca

Kevin Sullivan
CIS*4720: Image Processing and Vision
University of Guelph
Guelph, ON, Canada
ksulli06@uoguelph.ca

***Abstract*—Histogram backprojection is a method proposed by Wirth and Zaremba to detect flame regions in an image. The following paper will analyze the success of an implementation of this algorithm.**

***Keywords*—histogram backprojection, flame region detection, analysis, image processing, colour images**

I. INTRODUCTION

Several algorithms have been proposed as a potential solution to detecting fire in still images. The utility of solving this problem is clear: smoke detectors are flawed in that smoke is not always an indicator of fire, and processing an image of fire to detect the existence, severity, and type of fire has clear utilities as a superior fire detection method. In 2010, flame region detection using histogram backprojection was proposed by Wirth and Zaremba as a potential simplistic approach to fire detection using image processing [1]. The following paper will implement and analyze the success of the algorithm proposed to draw insights into the benefits and drawbacks of this particular solution to the problem of processing flame images.

II. ALGORITHM DISCUSSION

A. *YCbCr versus HSV Colour spaces*

When working with colour images, one of the first questions to answer is which colour space will be used in the algorithm. The algorithm proposed by Wirth and Zaremba experiments with both HSV and YCbCr. During early development of the algorithm discussed in this paper, HSV was used to generate the fire-detected image. In the images produced, we found that pieces of the image (in our first trial, it was grass) that were too close to our sample image (which includes hues of yellow) were registering as "fire". This is because the HSV colour space uses Hue, Saturation, and Value, and does not take into account the luminosity of image features. Because luminosity is such a large part of what defines fire in an image, the neglect of this trait led to a large range of false-positives for image features that were close to the sample image.

As such, YCbCr was chosen as an appropriate colour space to detect fire in an image. This colour space accounts for luminosity and eliminates a fair amount of false positives that HSV would otherwise register in our algorithm.

B. *Histogram Backprojection*

Histogram backprojection creates a binary mask of an image using the image's histogram and a sample image. The function replaces the pixels in an image based upon a given pixel's probability to occur given the sample image. These probabilities are represented on a binary mask which visually displays the effectiveness of the backprojection. In terms of fire detection, backprojection is a favoured choice above other methods because it looks at colour information rather than spatial information. The shape of fire is constantly changing and therefore makes detection through spatial information challenging, whereas detection based on the colour of fire is relatively more consistent. This does, however, mean that backprojection will identify objects with similar hues as fire. In order to decrease the amount of false positives returned, a YCbCr colour space should be used when applying backprojection. The emphasis on luminosity creates a more realistic barrier of entry for the binary mask, as it is less likely for an image to have a bright orange non-fire object as opposed to an orange non-fire object.

C. *Constructing the Algorithm*

Starting out, backprojection was the foundation for the algorithm, and several different methods were executed to try to improve the mask produced. HSV was the first colour space used due to its simplicity and adaptability with backprojection. The sample image as shown in Figure 1 used to calculate the backprojection was initially part of the fire in the same image being tested in order to test the effectiveness of backprojection in general.

Fig. 1. A barn fire (left) and its HSV histogram backprojection multiplied onto the original image (right).

The backprojection shown in Figure 1 was able to successfully mask the fire, but masked a lot of the grass as well. At first, it was not clear that the colour space was the problem because fire detected in the image seems nearly perfectly masked. To try and work around this, a higher threshold was placed on the backprojection in order to try and eliminate hues that differed too much from our sample.

Fig. 2. A barn fire (left) and its HSV histogram backprojection multiplied onto the original image with a higher threshold (right).

As depicted in Figure 2, some of the grass has been successfully removed, but so has the fire. It became clear that the yellowish-greens from the grass are too close to the oranges and yellows from the flames on the colour wheel. A greater threshold would not be able to eliminate similarly hued objects from the binary mask. This led to the switching of colour spaces from HSV to YCbCr in order to better separate objects not just by hue, but also by luminosity.

Fig. 3. A barn fire (left) and its YCbCr histogram backprojection multiplied onto the original image (right).

The fire was isolated in the binary mask, which marked the first major breakthrough in the algorithm implementation. It is noticeable, however, that the mask of the fire seems to only accept colours in a very confined range.

Fig. 4. The sample image used for histogram backprojection.

The reason for this seemingly monochromatic masking stems from the sample image initially used. It is visualized in Figure 4 and limits the backprojection probability to a very small colour space. The mediocre result when masking with a sample from the same image raised concerns for future testing with other images.

In order to enhance the sample backprojection image, a selection of 12 images containing fire in different settings both in and out of the testing images were taken and concatenated into one file. Using image samples from outside of the tested images reduced the bias in testing and gave a more general sample colour space of what this algorithm defines as “fire”.

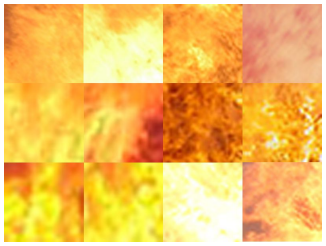

Fig. 5. Enhanced selection of samples defining fire.

The enhanced sample image shown in Figure 5 was hypothesized to be able to expand the scope of the algorithm and ultimately allow more images to be tested. Testing across more images helped aid the algorithm in determining where it was flawed and gave insight into how the implementation could be improved.

Fig. 6. A barn fire (left) and its binary mask backprojection (right) with a diverse sample set.

This algorithm was tested across some images that had an underlying assumption that the fire was more easily detectable. These images have clear colour contrast and similar luminosity throughout. Testing on these kinds of images first exposes how much fire is being masked without other factors impacting the result, enabling more effective analysis in early stages of development. Figure 6 shows a successful masking of the fire, however, the image suffers from multiple regions within the binary mask which produces an eroded appearance. It was clear that expanding the mask to reduce the erosion would improve the algorithm and produce a more monolithic region. The first proposed method was to apply a blur to the original image in order to expand the hues outward and allow the mask to return more data within the fire colour space. The blurring effect did expand the mask, but also caused some masking loss on the edges of the fire. A dilation seemed more natural when dealing with a binary mask, which was implemented and tested next.

Fig. 7. A barn fire (left) the calculated backprojection (middle) and a dilation applied with a 5x5 neighbourhood (right)

Even though the dilation successfully closes the gaps and creates a single enhanced region from the mask, the edges of the flame still lose their original position. The solution is being able to cover the gaps while also keeping the shape of the mask edge. This can be achieved through what is called “closing” [2]. By first dilating the mask and then eroding it, the mask will be filled while also preserving the edges.

Fig. 8. The same image using the closing morphology on the binary mask (middle), and the multiplication (right).

The closing property has created the best rendition of this fire detection so far. Although it works well with this image, not all images of fire require this property, let alone using the same neighbourhood size. Cross-checking the results without the closure property generally yielded better results across multiple images. The metric used for cross-checking will be discussed in our experimental results. A hurdle in implementing this algorithm was focusing on changing/improving too many methods for a single image. Following this realization, an automated testing suite was implemented to test multiple images which greatly decreased bias during implementation. Ultimately, the algorithm was reverted back before dilation was implemented because it was the strongest iteration.

The last changes made involved changing threshold values on the mask and kernel sizes for the histogram and structuring elements.

III. EXPERIMENTAL RESULTS

A. *Algorithm Effectiveness Testing*

Analyzing detection masks can mostly be done visually because the information from a mask can always be presented in the form of an image. However, the effectiveness can still be quantified using a difference function between images. In order to achieve this, a ground truth image of the binary mask is required. The testing algorithm compares both binary masks and through a small calculation confirms how much of outputted image mask matches the ground truth mask.

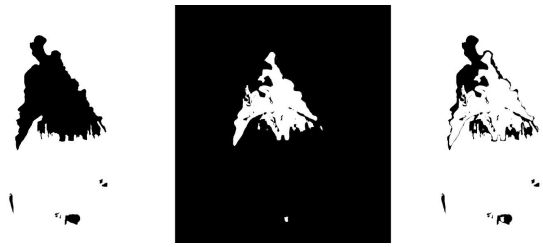

Fig. 9. The complement of the ground truth binary mask (left), the fire detection binary mask using the algorithm (middle), and the difference mask between the left and middle (right).

The complement of the ground truth image is first computed which will help in visually analyzing the effectiveness. The image difference is then computed between the ground truth and the output which is seen in Figure 9 on the right. When this is computed, the white pixels represent equivalence in images while the black pixels represent nonequivalence. It is clear that a calculation can be retrieved from this data to find a level of masking effectiveness.

An effectiveness measure E_M is obtained by:

$$E_M = ((B_{GT} - B_{DM}) / B_{GT}) * 100$$

where B_{GT} is the number of black pixels in the ground truth image and B_{DM} is the number of black pixels in the

difference mask. This measurement will return a percentage that correlates to the correctness of the mask with the ground truth. To visualize this measurement, a resulting mask that is completely white would have 100% effectiveness and show no difference (black pixels) between the compared masks.

B. *Summary of Test Data*

TABLE I. EXPERIMENTAL RESULTS

IMAGE	CATEGORY	METRIC PRECISION
output_s1	simple	59.28%
output_s2	simple	64.52%
ouput_s3	simple	77.88%
output_s4	simple	38.25%
output_c1	complex	24.97%
output_c2	complex	36.98%
output_c3	complex	19.25%
output_c4	complex	30.16%

C. *Flames in Simple Scenes*

Testing images categorized as "simple" are images in which fire is the only yellow-red-orange and luminant feature in the image. The first image is the simple barn fire discussed earlier in the paper. Utilizing the final version of the algorithm, the devised metric demonstrated in figure 10 returns 59.28% precision when compared against the ground truth image. The algorithm gathers most of the fire, with some gaps in the binary mask. The algorithm picks up small details in this image, but fails in gathering the darker red parts of the fire that are not as likely to appear in the sample space.

Fig. 10. Output_s1 a burning barn (left), its binary mask (middle), and the multiplication (right).

The next two simple images follow a similar pattern where the fire is the focus of the image, and the most luminant feature. The second and third simple images are that of a forest fire, where the luminosity of the fire shrouds the other image aspects in darkness, making the extraction of the fire easy. The resulting metric illustrated in figure 11 returns 64.52%, where, again, the gaps in the difference between the algorithm output and the ground truth yield the missing pieces. The metric illustrated in figure 12 returns 77.88%; upon analysis, the weakness demonstrated in the algorithm here is the tendency to pick up the fiery smoke as well as all the luminant fire.

Fig. 11. Output_s2: a forest fire with a significant amount of darkness (left), its binary mask (middle), and the multiplication (right).

Fig. 12. Output_s3: a forest fire with a fair amount of darkness (left), its binary mask (middle), and the multiplication (right).

The final simple test image features several fires in a dry grassy field. Considering luminosity, the algorithm picks up a lot of bright blades of grass, yielding a metric of 38.25%. While the algorithm indeed succeeds in detecting the flames in the image, the detection of grass is a clear false positive.

Fig. 12. Output_s4: a dry burning field (left), its binary mask (middle), and the multiplication (right).

Ultimately, the algorithm performs fairly well on the simple fire images, reporting the majority of the fire that can be visually perceived in the image as well as the majority of the ground truth images used to calculate the metric.

D. *Flames in Complex Scenes*

Testing images categorized as "complex" are images in which fire is one of several yellow-red-orange and luminant entities in the image. The first complex image, seen in figure 14, features brightened parts of the image due to the sun as well as a bright red fire truck reflecting some light. Other than the fire, the rest of the image is fairly dull, so it was hypothesized that if anything, the algorithm would report the illuminated features and the fire truck falsely as fire. Indeed, figure 13 demonstrates that the algorithm

picked up the sun-illuminated features, the fire, and some reflection off of the fire truck. The algorithm is sophisticated enough however to ignore the redness of the firetruck as a whole. Ultimately the metric returns a value of 24.97%, where the fire is detected in its entirety, but the sun strongly impacts the accuracy of the result.

Fig. 14. Output_c1 a burning house and red fire truck (left), its binary mask (middle), and the multiplication (right).

The second complex image seen in figure 15 features a building on fire in the day, with several illuminated features in addition to the fire at the top of the image. It was anticipated that the algorithm may have picked up the bright red signs that are illuminated as well as parts of the red building. The algorithm performs with a metric value of 36.98%, the most accurate among the complex images, and picks up the majority of the fire space as well as the anticipated red-illuminated features.

Fig. 15. Output_c2 a red burning building (left) , its binary mask (middle), and the multiplication (right).

The third complex image, seen in figure 16, features a dull burning building with a few illuminated features. It was anticipated that the algorithm would perform fairly well in this image with minor inaccuracies surrounding the light. On the contrary, the algorithm performs less optimally, producing a metric of 19.25%, and as shown in figure 16, only picking up a few aspects of the fire in the image. This is due to the darkness of the flame not falling within the sample image used for the histogram backprojection, as well as the fire not being as luminous as previous images.

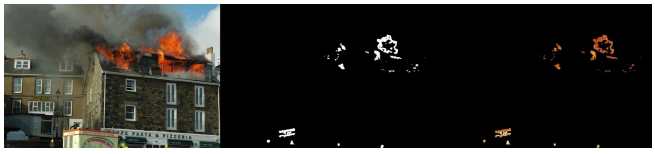

Fig. 16. Output_c3 a dull coloured burning house (left), its binary mask (middle), and the multiplication (right).

The final image seen in figure 17 features an orange building on fire with a dull background. It was hypothesized that the image would pick up much of the orange building and the fire. Indeed, the image performs with an accuracy of 30.16%, picking up less of the building than what was anticipated and all of the image's fire. This again reinforces the algorithm's reliance on colour and luminance to detect fire.

Fig. 17. Output_c3 a bright orange burning building (left), its binary mask (middle), and the multiplication (right).

Ultimately, testing with the complex images is consistent with previous findings, where the algorithm accurately determines there is fire in the image, but when introduced to sun rays that illuminate yellow-red-orange features of the image, the algorithm incorrectly masks these aspects as fire. This is a clear drawback to the algorithm implemented, and further testing regarding this trait of the algorithm will be discussed in the following sections.

E. *False Positives*

To test the algorithm fully, some images that were anticipated to return false positives were added to the testing suite. The first of these images, seen in figure 18, is a very luminant photo with wheat that matched some of the yellowed used in the sample image. It was anticipated, based on previous tasting that this would return a large quantity of false positive throughout the image. Indeed, the algorithm reports this image as a false-positive in detecting fire, as it is recognizing the sun-illuminated wheat as fire. As can be seen in figure 17, the only feature picked up in this image is the golden wheat, demonstrating the significance of the sample space in the performance of this algorithm.

Fig. 18. Output_fp-1 a bright wheat field (left), its binary mask (middle), and the multiplication (right).

Figure 19 depicts several sunflowers of various yellow-red-orange hues. The algorithm was anticipated to fail fire detection in this image, as all the flowers are brightly illuminated, and nearly all fall within the sample's range. Figure 18 shows that the strongest detection occurs in the yellow flowers, which is congruent with previous tests. This is due to the prevalence of yellow in the sample space, though it is noted that each flower at different points are masked based on which hue they depict.

Fig. 19. Ouput_fp-2 several sunflowers (left), its binary mask (middle), and the multiplication (right).

Both images shown in figure 20 and figure 21 use the concept of a "beach" scene to test the algorithms performance. Given that a beach contains the yellow of the sand, skin tones, and sun rays, it was hypothesized that the algorithm would perform poorly when used on these images. The image in figure 20 produces a mask that seems to pick up more of the faces of the people on the beach than the sand (fairly dull when compared to the sun reflecting off skin tones) but ultimately performs fairly well when masking the image that does not contain fire. The image in figure 21 on the other hand produces a mask that considers the majority of the image (illuminated sand and palm leaves) as "fire" and performs as poorly as expected. Ultimately, these results show that while an illuminated sandy scene such as demonstrated in figure 21 perform poorly in the algorithm, a scene with duller sand and illuminated faces performs better than expected.

Fig. 20. Ouput_fp-3: a beach with many people illuminated by the sun (left), its binary mask (middle), and the multiplication (right).

Fig. 21. Ouput_fp-4: a beach with a lot of sand illuminated by the sun (left), its binary mask (middle), and the multiplication (right).

The final two images (seen in figure 22 and 23) used to test false-positives for the algorithm were a deep red sunset and a bright snow scene. It was anticipated that in the first image the redness of the scene would trigger some significant false positives if the luminant threshold was flawed in the algorithm, and that the brightness of the second image would return false positives if the hue and sample space were flawed. On the contrary, both of these images perform well when put through the algorithm, producing masks that barely pick up anything at all in both images. This demonstrates that the core functionality of the algorithm is present and that many of the "challenging" cases test the limits of the algorithm appropriately, while the algorithm still passes fairly simple false-positive cases.

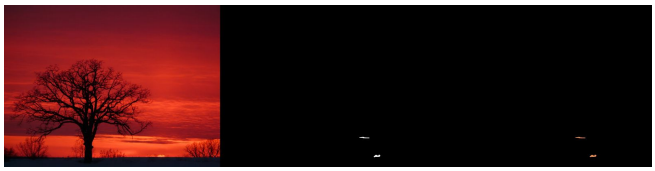

Fig. 22. Ouput_fp-5: a deep red sunset (left), its binary mask (middle), and the multiplication (right).

Fig. 23. Ouput_fp-6: a bright sun-lit snow scene with dogs (left), its binary mask (middle), and the multiplication (right).

F. *False Negatives*

The final test that the algorithm was put through was to see if it would incorrectly produce false negatives. The first of these images, seen in figure 24 was a fire in the daylight on water. This image was anticipated to produce a false negative output if the algorithm was unable to differentiate between the luminosity of the water and the fire. On the contrary, the algorithm performs extremely well on this image and successfully masks the fire.

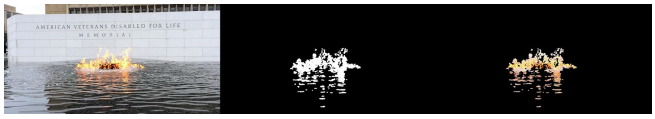

Fig. 24. Ouput_ch-1: fire on water in the day (left), its binary mask (middle), and the multiplication (right).

The second test image seen in figure 25 was anticipated to test the algorithm's detection of fire when the fire is illuminating skin in a dark environment. It was hypothesized that the algorithm would mask the skin more than anything, as the fire is too bright white to be picked up by the sample space. Our hypothesis proved correct, through the algorithm masks the edge of the fire successfully, but also picks up a lot of the skin in the picture that falls within the sample and the luminosity threshold. The third image shown in figure 26 produces similar results where a flame on a black background is difficult to mask, as the fire is *too* luminant to distinguish, though the algorithm does catch the edges of the flame. We identify this image as a true false negative among our false negative tests

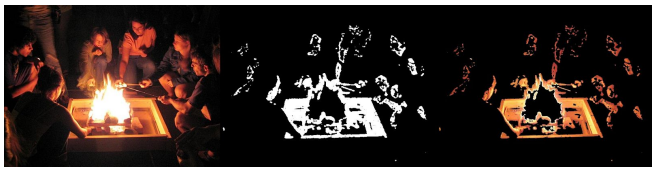

Fig. 25. Ouput_ch-2: campfire in the dark illuminating people (left), its binary mask (middle), and the multiplication (right).

Fig. 26. Ouput_ch-3: flame from a lighter in the dark (left), its binary mask (middle), and the multiplication (right).

The final test image shown in figure 27 is a campfire in the daylight. The campfire is one of many luminant objects in the image, and it was hypothesized that the brightness of the sun would be picked up more so than the flame. The algorithm masks most of the sun rays and an illuminated orange towel as well as the fire, producing a fair amount of false negatives, but an interesting effect.

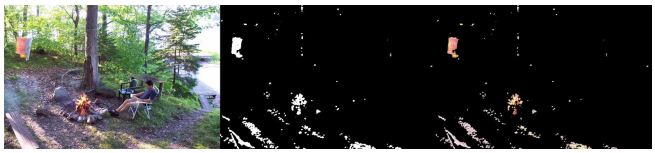

Fig. 27. Ouput_ch-4: campfire in the day with a lot of sunlight (left), its binary mask (middle), and the multiplication (right).

Ultimately, the algorithm is fairly successful in eliminating false negatives, where more often than not it picks up the fire despite interferences, but as shown in figure 26, it still fails in some cases.

IV. CONCLUSION

The algorithm discussed in this paper ultimately performed fairly well in the detection of fire. While the algorithm has clear deficiencies when it comes to hue and luminant-similar aspects of images, when presented with true fire in an image the algorithm has an equitable success rate. The simplicity of the algorithm makes it easy to understand, implement, and reduces the computational power needed to create the masked images. However, the algorithm's dependency on still images eliminates an important aspect of fire that would assist in its detection — movement. Furthermore, the algorithm's use of a sample eliminates examples of fire that do not fall within the samples used such as blue flames and the white core seen in some of the examples. While these are notable drawbacks that can be improved upon, based upon the metric used the algorithm succeeds more often than not at detecting fire in the first two sets of images. As such, the histogram backprojection approach to fire detection can be defined as a basic, and accurate version of a fire detection algorithm.

REFERENCES

[1] Wirth, M.A. Zaremba, R., "Flame region detection based on histogram backprojection", Canadian Conference on Computer and Robot Vision, IEEE, pp.167-174 (2010).

[2] Nikitenko, D., "Binary Image Morphology", Lecture, University of Guelph, (2019).