

Are there water drops in the camera lens?

Carlo Wiese

I. INTRODUCTION

Water drops adsorbed by the lens refract the light that enters the camera lens, causing image disturbance. As a result, image-based recognition algorithms fail to function normally under such conditions. Therefore, it is necessary to determine in advance whether water drops are adsorbed to prevent the misrecognition of these types of algorithms. However, droplets adsorbed by camera lenses have various sizes and shapes, making them difficult to detect.

The task is to develop and deliver an algorithm capable of detecting whether or not there are droplets on a camera lens, using images taken with that camera only.

II. DATA

The provided dataset consists of 1006 images where 503 are clear and the other 503 show droplets. There are also 30 images reserved for the final evaluation, with 15 of these images being clear and the other 15 covered by droplets.

The preliminary analysis shown in Figure 1 demonstrates that taking only the blue channel from the provided images reduces the presence of clouds. As a blue sky and a white cloud both share a high pixel value in the blue channel, taking only the blue channel will suppress most clouds. Clouds, or in other words, white areas in the middle of a blue sky, resemble the light refracted by the camera lenses due to droplets, which would only make our task harder.

Decision 1. I decided to work only with the blue channel of the input images for this task.

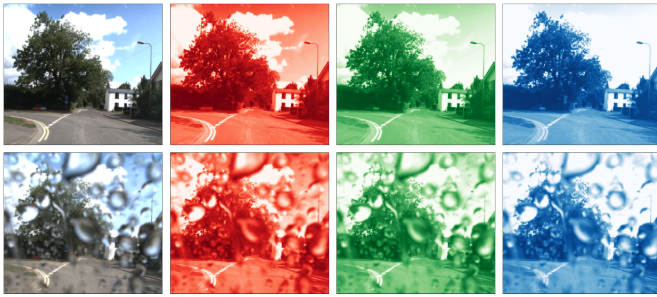


Fig. 1. RGB input with its 3 color channels

Further analysis showed too much contrast between the ground and the sky, meaning we can either focus on the sky or the ground for droplet detection. For such reason, gamma correction was performed on the images to lighten the contrast. The results of this operation are shown in Figure 2.

Decision 2. I decided to apply gamma correction to all input images for this task.

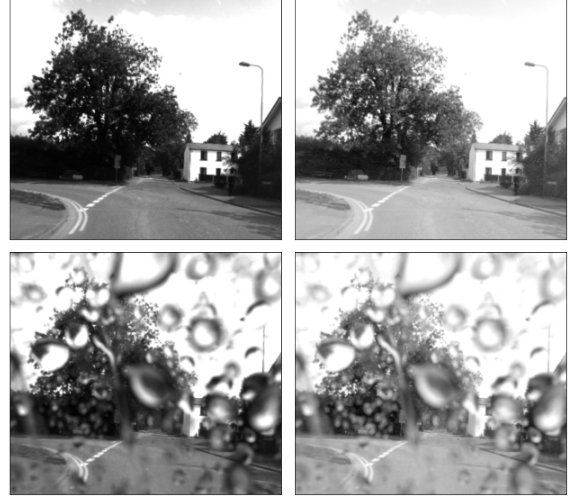


Fig. 2. Gamma correction

At the end, the pixel values of the blue channel of an input image after applying gamma correction will look as the 3D graph in Figure 3. Here we can observe the high-frequency behavior of the droplet textures. High-frequency textures stand out in an image creating contrast, which is why we can perceive droplets so easily. As evidenced by the undulatory behavior of the pixel values on the droplet image, this seems to be the biggest difference between clear and droplet images.

Decision 3. I decided to extract features related to pixel frequency for my proposed approach.

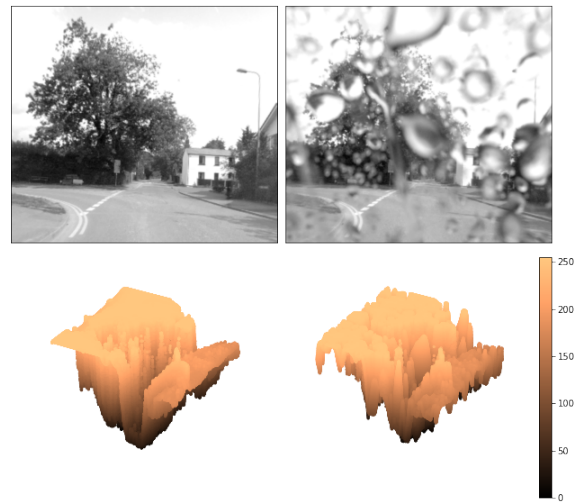


Fig. 3. Pixel values of the preprocessed input plotted as a 3D graph

III. SOLUTION

A. Proposed approach

During the preliminary analysis, I realized the high-frequency behavior of the droplet textures. The best way to extract features related to frequencies is using the Fast Fourier Transform (FFT), and Gist features do just that in [Oliva and Torralba, 2001]. Therefore, this approach extracts Gist features from images and classifies these features into clear or droplet images using a Support Vector Machine (SVM).

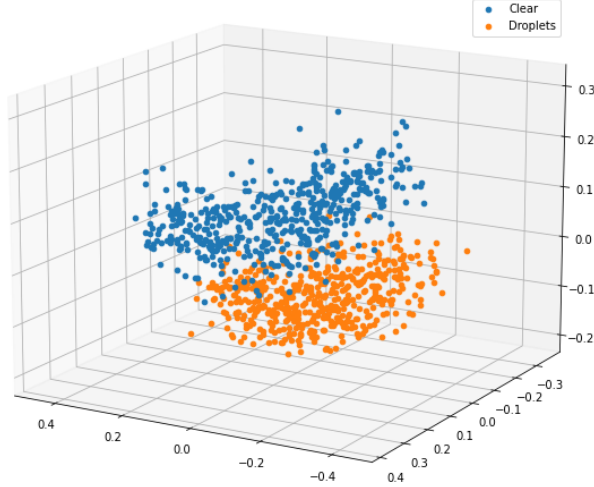


Fig. 4. Labelled feature space for all images in the training set

Figure 4 shows a clear division between these two clouds of feature points, confirming my claim that frequency is very relevant to this task.

B. Alternative approaches

For the sake of comparison, I also opted to train a deep learning model, as they excel at identifying complex features from images. Moreover, with deep learning being so popular, likely, someone has already tried to solve the same problem or similar using this approach. Therefore, I decided to look for research papers on droplet detection and found [Bae et al., 2019], a paper on raindrop detection. This paper provides two models, one for detection and one for classification. However, the authors claimed that their detection model wasn't fit for real-time processing due to its high complexity. This situation left us with their proposed classification model, which I implemented and tested successfully.

As the raindrop detection scheme from [Bae et al., 2019] was unsuitable for our task, I developed a droplet detection scheme. This alternative approach uses foreground extraction, image segmentation, and blob detection to filter out droplets in the image. Then I set a minimum number of raindrops detected as a threshold to decide whether the image is clear or not.

IV. RESULTS

As for the implementation details, the evaluation was performed using single-point precision (FP32) on a laptop with a

1.60 GHz Intel i5-8250U processor and no graphics card. The code is written using Python 3.6.9.

TABLE I
CLASSIFICATION AND COMPUTATIONAL PERFORMANCE RESULTS

Metric	Approach		
	Proposed	Alternative 1	Alternative 2
Recall	1	0.93	1
Precision	1	0.93	1
Computation time	104 \pm 2 ms	406 \pm 22 ms	107 \pm 2 ms
RAM memory used	230 MB	1380 MB	230 MB

A. Proposed Approach

I achieved a perfect score on the training and the evaluation set, making this approach the best so far. Additionally, SVMs provide great generalization, so I'm confident this approach will work well even on images outside our dataset.

Decision 4. As this approach was perfect, I decided to submit the proposed approach as my solution to the task.

B. Alternative approaches

The results of the deep learning model (Alternative 1) were quite good but still failed in some cases of the evaluation set. For instance, when most image contents were light going through trees and blurry objects, the classifier may misclassify a clear image as a droplet image. So, in essence, the light going through trees gets confused with droplet refraction and blurry objects with droplet texture. On the other hand, droplet texture merges with the trees and road when the image is mostly forest and road with little to no light from the sky. As the characteristic droplet refraction becomes less evident, the classifier may perceive droplet images as clear.

As for the other approach (Alternative 2), this proved to be very successful, even achieving a perfect score in the evaluation set. However, it required extensive parameter tuning, making this approach very sensitive to changes in the dataset.

V. CONCLUSIONS

Finally, given how crucial it is to know when images become unusable for other visual algorithms, I would focus on recall. A high recall means fewer misclassifications, while high precision means fewer false alarms. Therefore, a high recall would be my priority to ensure droplets are always detected, which won't be a problem based on my results. The proposed approach shows great performance in terms of classification, generalization capabilities, and computation time.

So far, I'm unaware of the limitations of this approach since it worked flawlessly with all the training and evaluation data. However, images with only small droplets or those with droplets clustered on the camera lens border may require tuning the Gist feature extraction parameters.

REFERENCES

- [Bae et al., 2019] Bae, I.-G., Bang, S., and Kim, W. (2019). Convolutional neural network for raindrop detection. In *2019 16th International Conference on Ubiquitous Robots (UR)*, pages 102–105.
- [Oliva and Torralba, 2001] Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *The International Journal of Computer Vision*, 42:145–175.