Adherent Raindrop Detection and Removal in Video

Shaodi You[†] Robby T. Tan[‡] Rei Kawakami[†] Katsushi Ikeuchi[†]
[†]The University of Tokyo
[‡]Utrecht University
{yousd, rei, ki}@cvl.iis.u-tokyo.ac.jp
R.T.Tan@uu.nl

Abstract

Raindrops adhered to a windscreen or window glass can significantly degrade the visibility of a scene. Detecting and removing raindrops will, therefore, benefit many computer vision applications, particularly outdoor surveillance systems and intelligent vehicle systems. In this paper, a method that automatically detects and removes adherent raindrops is introduced. The core idea is to exploit the local spatio-temporal derivatives of raindrops. First, it detects raindrops based on the motion and the intensity temporal derivatives of the input video. Second, relying on an analysis that some areas of a raindrop completely occludes the scene, yet the remaining areas occludes only partially, the method removes the two types of areas separately. For partially occluding areas, it restores them by retrieving as much as possible information of the scene, namely, by solving a blending function on the detected partially occluding areas using the temporal intensity change. For completely occluding areas, it recovers them by using a video completion technique. Experimental results using various real videos show the effectiveness of the proposed method.

1. Introduction

Outdoor vision systems are employed for various tasks such as navigation, data collection and surveillance. Unfortunately, they can be adversely affected by bad weather conditions such as rain, haze and snow. In a rainy day, raindrops inevitably adhered to windscreens, camera lenses, or protecting shields. These adherent raindrops occlude and deform some image areas, making the performances of many algorithms in the vision systems (such as feature detection, tracking, stereo correspondence, etc.) significantly degraded. This problem occurs particularly for vision systems that use a hand-held camera or a top-mounted vehicle sensor where no wipers can be used.

Identifying adherent raindrops from images can be problematic, due to a few reasons as shown in Fig. 1. Foremost, adherent raindrops have various shapes. Unlike opaque objects, they are transparent, making their appearance and thus intensity values vary depending on the environment. They suffer from out-of-focus blur due to their proximity to the camera. Moreover, most raindrops generate glare.

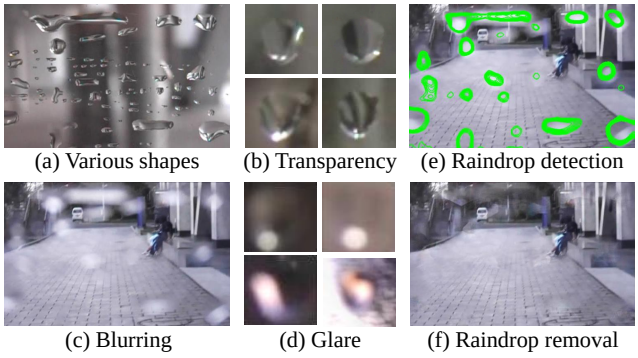


Figure 1. (a-e) The various appearances of raindrops. (e-f) The detection and removal result by our method.

To address the problems, we analyze the appearance of adherent raindrops from their local spatio-temporal derivatives. First, a clear, unblurred adherent raindrop works like a fish-eye lens and significantly contracts the image of a scene. Consequently, the motion inside raindrops is distinctively slower than the motion of non-raindrops. Second, unlike clear raindrops, blurred raindrops are mixtures of rays originated from the points in the entire scene. Because of this, the intensity temporal derivative of blurred raindrops is significantly smaller than that of non-raindrops. These two clues are the key idea of our detection method, and can be processed on a pixel basis, making the method generally applicable to handle any shape and size of raindrops. Fig. 1.e shows a result of our detection method.

By further analyzing the image formation of raindrops, we found that some area of a raindrop completely occludes the scene behind, however the rest occludes only partially. For partially occluding areas, we restore them by retrieving as much as possible information of the scene, namely, by solving a blending function on the detected areas using the intensity change over time. For completely occluding areas, we recover them by using a video completion technique. Fig. 1.f shows a result of our removal method.

The contributions of the paper are threefold: (1) Adher-

ent raindrops are theoretically modeled and analyzed using the derivative properties with few parameters, enabling the method to be applied to general video cameras, *e.g.*, hand-held and vehicle-mounted cameras. (2) A novel pixel-based detection method is introduced. (3) A relatively fast adherent raindrop removal method is proposed. It utilizes not only a video completion technique, but also the information behind some blurred areas of raindrops.

The rest of the paper is organized as follows. Sec. 2 discusses the related work on raindrop detection and removal. Sec. 3 explains the modeling of the spatial and temporal derivative properties on raindrop images. The detailed methodology of the raindrop detection is described in Sec. 4, followed by the detailed methodology of the raindrop removal in Sec. 5. Sec. 6 shows the quantitative experiments and results. Sec. 7 concludes the paper.

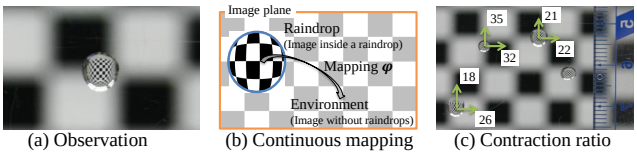
2. Related Work

Removing the influence of haze, mist, to some extent fog (e.g., [20, 2, 7]), rain and snow (e.g., [1, 5]) have been well exploited. Dealing with rain, Garg and Nayar first model it [3], and then detect and remove it [4, 5]. Later, Barnum *et al*. [1] propose a method to detect and remove both rain and snow. Unlike the previous methods that use a video, Kang *et al*. [10] introduce a detection and removal method using a single image. Unfortunately, applying these methods to handle adherent raindrops is rather not possible, since the physics and appearance of falling raindrops are significantly different from those of adherent raindrops.

Methods for detecting adherent raindrops caused by sparse rain have been proposed. Roser *et al*. attempt to model the shape of adherent raindrops by a sphere crown [14], and later, Bezier curves [15]. However, the models are insufficient, since a sphere crown and Bezier curves can cover only a small portion of possible raindrop shapes (Fig. 1.a). Kurihata *et al*. [11] directly collect image templates of many raindrops and calculate their principle components. However, as shown in Figs. 1.a-d, collecting and aligning training images for all various shapes, environment, illumination and blurring are considerably challenging. Yamashita *et al*. propose a detection and removal method for videos taken by stereo [23] and pan-tilt [22] cameras. The methods utilize specific constraints from those cameras and are thus inapplicable for a single camera. Hara *et al*. [6] propose a method to remove glare caused by adherent raindrops by using a specifically designed optical shutter.

As for raindrop removal, Roser and Geiger [14] address it using image registration, and Yamashita *et al*. [23, 22] utilize position and motion constraints from specific cameras.

Video completion has been intensively exploited by computer vision researchers. However, only those methods work with large spatio-temporal missing areas can be used to remove detected adherent raindrops. Wexler *et al*. [21] propose an exemplar based inpainting method by assuming the missing data reappears somewhere else in the video. Jia *et al*. [9] exploit video completion by separating static background and moving foreground, and later [8] exploit video completion under cyclic motion. Sapiro and Bertalmio [16] complete the video under constrained camera motion. Shiratori *et al*. [17] and Liu *et al*. [13] first calculate the motion of the missing areas, and then complete the video according to the motion. Unfortunately, outdoor environments are too complex to satisfy static background, cyclic motion, constrained camera motion, etc. Therefore, we consider using cues from our adherent raindrop modeling to help the removal.



(a) Observation (b) Continuous mapping (c) Contraction ratio

Figure 2. (a) A raindrop is a contracted image of the environment. (b) On the image plane, there is a smooth mapping φ starting from the raindrop into the environment. (c) The contraction ratios from the environment to a raindrop are significant.

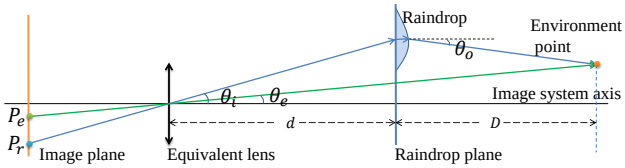


Figure 3. The refraction model of two points on an image plane (P_e and P_r) that are originated from the same point in the environment. There are two refractions on the light path passing a raindrop. The camera lens cover or protecting shield is assumed to be a thin plane and thus can be neglected.

3. Raindrop Modeling

Unlike the previous methods [15, 11, 23, 22, 6], which try to model each raindrop as a unit object, we model raindrops locally from the derivative properties that have only few parameters.

3.1. Spatial Derivative of Clear Raindrop

As shown in Fig. 2.a, the appearance of each raindrop is a contracted image of the environment, as if it is taken from a fish-eye-lens camera. The numeric values indicated in Fig. 2.c are the contraction ratios between the original image and the image inside the raindrops calculated from the black and white patterns. The contraction ratio is around 20 to 30, meaning that the motion observed inside the raindrops will be $1/30$ to $1/20$ slower than the other areas in the image.

Let us denote a point inside a raindrop on the image

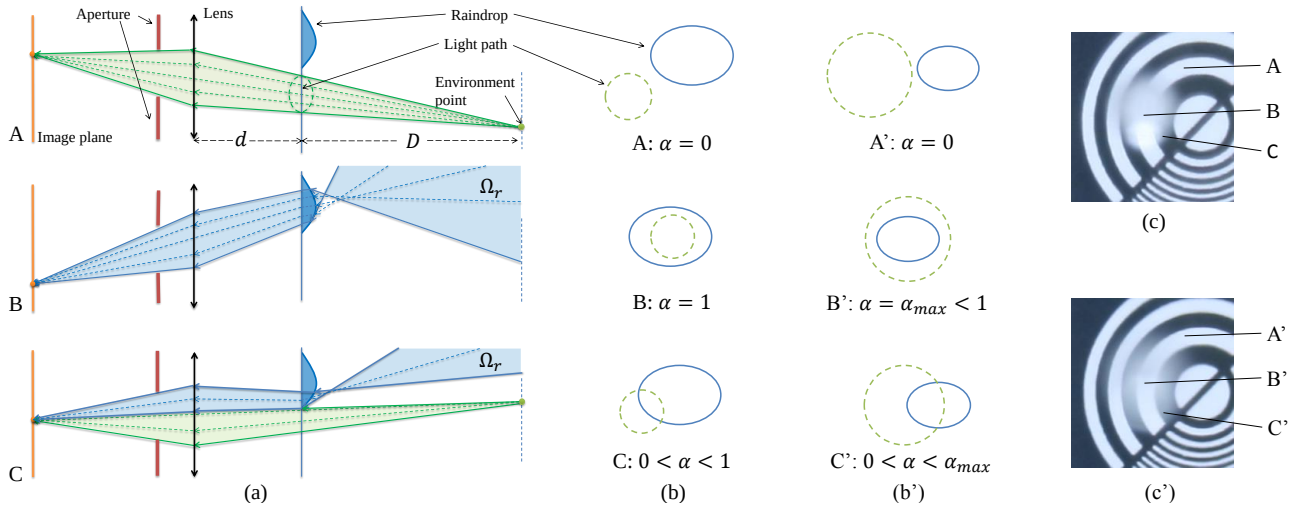


Figure 4. The appearance and model of pixels on an image plane collecting light from A: environment, B: raindrop, C: both. (a) The light path model. Green light: the light coming from environment point; Blue light: the light refracted by a raindrop. (b) The raindrop plane cut the section of model in (a) when a raindrop is big. Green circle: the area of light collected. Blue circle: the raindrop. α: percentage of light collected from the raindrop. (b') A raindrop plane cut the section when it is small. (c) The appearance of the 3 situations in (b). (c') The appearance of the 3 situations in (b').

plane as (u, v), and its corresponding original pixel as (x, y). As shown in Fig. 2.b, there is a 2D to 2D mapping $\boldsymbol{\varphi}$ from (u, v) to (x, y):

$$(x, y) = \boldsymbol{\varphi}(u, v) = (\varphi^1(u, v), \varphi^2(u, v)). \tag{1}$$

The contraction ratio $\mathcal{E}_\varphi$ is the derivative of $\boldsymbol{\varphi}$ with respect to u and v in the direction $(\delta u, \delta v)$:

$$\mathcal{E}_\varphi(u, v, \delta u, \delta v) = \lim_{(\delta u, \delta v) \to 0} \frac{\|\boldsymbol{\varphi}(u + \delta u, v + \delta v) - \boldsymbol{\varphi}(u, v)\|}{\|(u + \delta u, v + \delta v) - (u, v)\|}. \tag{2}$$

Using the imaging model in Fig. 3, in outdoor environment, we can prove that:

$$\mathcal{E}_\varphi > 10 \gg 1. \tag{3}$$

The proof is provided in the supplementary material.

Considering the refraction model in Fig. 3, to know the function $\boldsymbol{\varphi}$, we need: (1) the position and shape information of the raindrop, (2) the camera inner parameters, and (3) the background depth information. Fortunately, our analysis looks into the spatial derivative properties, and therefore can avoid obtaining $\boldsymbol{\varphi}$ explicitly.

3.2. Temporal Derivative of Blurred Raindrop

When a camera is focused on the environment scene, raindrops will be blurred. The image intensity of blurred pixels is a mixture of rays originated a point that coincides with the line of sight (the green line in Fig.3), and the collection of rays emitted from different points in the entire scene, as shown in Figs. 4.c and c'. Thus, recovering the original image is equivalent to extracting the rays from the point that coincides with the line of sight.

Let us model the image intensity of blurred pixels using a blending function. We denote the light intensity collected by pixel (x, y) as $I(x, y)$, the light intensity formed by an environment point that intersects with the line of sight without being through a raindrop as $I_e(x, y)$, and the light intensity reached (x, y) through a raindrop as $I_r(x, y)$. Then, pixel (x, y) collecting light from both the raindrop and the environment can be described as:

$$I(x, y) = (1 - \alpha) I_e(x, y) + \alpha I_r(x, y), \tag{4}$$

where α denotes the proportion of the light path covered by a raindrop, as depicted in Figs. 4.b and b'. In Fig. 4.a, A to C illustrate the three situations. In A, light rays emitted from an environment point are all collected at (x, y), thus $I(x, y) = I_e(x, y)$. In B, the pixel (x, y) collects light rays passing through a raindrop, and therefore $I(x, y) = I_r(x, y)$. In C, only some of the light rays pass through a raindrop. The images of those three cases are shown in Figs. 4.c and c'.

In consecutive frames, we observed that the intensity of blurred pixels (case B and C) does not change as distinctive as that of environment pixels (case A). To analyze this property more carefully, let us look into the intensity temporal derivatives of blurred pixels. Referring to Figs. 4.a B and C, light collected from raindrop is actually refracted from a large area in the environment. We refer to the area as $\Omega_r(x, y)$. At time t, we expand $I_r(x, y)$ in Eq. (4) as:

$$I_r(x, y, t) = \sum_{(z,w) \in \Omega_r(x,y)} W(z, w) I_e(z, w, t), \tag{5}$$

where $W(z,w)$ is the weight coefficient determined by the raindrop geometry. $W(z,w)$ and $\Omega_r(x,y)$ can be considered to be constant in a short time period.

If we take the difference of intensity between time t_1 and t_2 in Eq. (5), and consider the triangle inequality, we have:

$$|I_r(x,y,t_1) - I_r(x,y,t_2)| \leq \sum_{(z,w)\in\Omega_r(x,y)} W(z,w)|I_e(z,w,t_1) - I_e(z,w,t_2)|. \quad (6)$$

Here, by considering Eq. (3), we know that the area ratio is more than a hundred, namely, $\mathcal{E}_\varphi^2 > 100 \gg 1$ (the proof is provided in the supplementary material), and thus, we can consider $\Omega_r(x,y)$ to be a sufficiently large area. Moreover, we may assume $|I_e(x,y,t_1) - I_e(x,y,t_2)|$ obeys an independent Gaussian distribution for any point (x,y). According to the law of large number, we have:

$$E|I_r(x,y,t_1) - I_r(x,y,t_2)| \ll E|I_e(x,y,t_1) - I_e(x,y,t_2)|, \quad (7)$$

where E denotes the expectation. By taking into account Eq. (7) with Eq. (4), the temporal difference for $I(x,y,t)$ will be small when α is large, and vice versa.

Since the temporal derivative works as a high pass filter, we may also consider Eq. (7) in a frequency domain, where the temporal high frequency component on a raindrop is significantly smaller than those of the environment, described as:

$$\mathcal{I}_r(x,y,\omega) \ll \mathcal{I}_e(x,y,\omega), \omega = \omega_{th}, \omega_{th}+1, \cdots, N \quad (8)$$

where $\mathcal{I}$ is the Fourier transform of sequence $I(x,y,t), t = t_1, t_2, \cdots, N$, and ω_{th} is currently undetermined threshold for high frequency.

3.3. Effects of Glare

As illustrated in Fig. 1.d, a raindrop will refract bright lights from the environment, and generate glare. This phenomenon was not discussed of the derivative properties described in the previous subsections. The reasons are, first, glare is the effect caused by a light source emitting high intensity light, and the spatial derivative introduced in Sec. 3.1 is independent from light intensity. Second, the appearance of glare in videos is temporally smooth, *i.e.*, the intensity monotonically increases until it saturates, and then it monotonically decreases until the glare fades out. The temporal derivatives of this smooth change is still small, and does not affect the analysis we have discussed.

4. Raindrop Detection

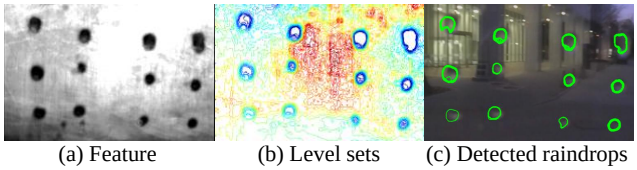
Feature extraction Based on the analysis of motion and the intensity temporal derivative, we generate features for the detection. First, we calculate dense motion, *e.g.*, SIFT-flow [12], as shown in Fig. 5.b. Second, we calculate the intensity temporal change $|I(x,y,t_1) - I(x,y,t_2)|$, as shown in Fig. 6.b.

(a) Image sequence (b) Inter frame SIFT flow (c) Summation of (b) over 100 frames

Figure 5. The accumulated optical flow as a feature.

(a) Image sequence (b) Inter frame intensity change (c) Summation of (b) over 100 frames

Figure 6. The accumulated intensity change as a feature.



(a) Feature (b) Level sets (c) Detected raindrops

Figure 7. The raindrop detection from the accumulated features.

In the examples, the two features are calculated only from two consecutive frames. In fact, they will be more informative if they are accumulated over a few frames. Statistically, the more the number of the frames, the more descriptive the features. However, raindrop positions can shift over a certain period of time. In our observation, with moderate wind, raindrops can be considered static over a few minutes. Considering this, we calculate the features over 100 frames which is about 4 seconds if the frame rate is 24 frames per second. Figs. 5.c and 6.c are examples of the two accumulated features.

Refined detection Having calculated the features, we use level sets [18] to identify raindrops. First, a convolution with Gaussian ($\sigma = 2$ pixels by default) is employed to reduce noise. Then, level sets are calculated, as shown in Fig. 7.b.

The following criteria are applied further for determining raindrop areas:

1. Feature threshold. We normalize the feature values such that the average is 0 and the variance is 1. Since raindrop areas should have smaller feature values, the threshold is set to -0.7 by default.
2. Closure. The level set around a raindrop area must be closed.

Algorithm 1 Raindrop detection

if (default)
$feature = intensity\ change$, $N = 100$, $\sigma = 2$,
$th1 = -0.7$, $th2 = 200 pixels$, $th3 = 0.05$
end
calculate inter-frame $feature$.
while $(ii + N < sequence\_end)$
sum up the feature for frames ii *to* $ii + N$
reduce noise using convolution of σ-Gaussian
normalize feature to $average = 0$, $variance = 1$
calculate level sets of the feature image.
for all level sets
if (feature $< th1$ & circumference $< th2$
& roundness $> th3$ & closed)
area circle by level set is a raindrop
end
end
$ii = ii + 10$
end

3. Size. The diameter of a single raindrop area is normally smaller than 5 millimeter. Thus, for our camera setting, the threshold for the raindrop circumference is loosely set less than 200 pixels.
4. Roundness: We define the roundness of a contour as $Area/Circumference^2$. A rounder shape would have a bigger roundness value and a perfect circle has the maximum roundness value: $\frac{\pi r^2}{(2\pi r)^2} = \frac{1}{2\pi} = 0.16$. We set the threshold of roundness loosely, *i.e.*, the threshold for raindrop is set greater than 0.05, which is far below the values that normally belong to raindrops.

Fig. 7.c is an example of the final detection result. Our overall raindrop detection algorithm is described in Algorithm. 1.

5. Raindrop Removal

Although the existing methods try to restore the entire areas detected as raindrops by considering them as solid occluders [14, 22] it will be more factual if we can restore the raindrop areas from the source scenes whenever possible. Based on Eq. (4), we know that some area of a raindrop completely occludes the scene behind, however the rest occludes only partially. For partially occluding areas, we restore them by retrieving as much as possible information of the scene, and for completely occluding areas, we recover them by using a video completion technique.

Restoration Blurred image can be recovered by estimating $I_e(x, y)$ in Eq. (4), in the condition that the blending is moderate, *i.e.*, $\alpha(x, y) < 1$.

To do this, we first have to calculate α in Eq. (4). Note that, based on the detection phase, the position and shape of raindrops on the image plane are known. Using the out-of-focus blur model in Fig. 4.a, the diameter ℓ of the equivalent light path area on the image plane is given by:

$$\ell = \frac{(D-d)}{(D-f)}\frac{f^2}{Od}, \tag{9}$$

where f is the focal length. O is the relative aperture size (also called f-stop) which can be found in the camera setting. D can be assumed to be infinite, and d is estimated by experiments (though, it is not a strict parameter and is constant throughout our experiments). The derivation of Eq. (9) can be found in the literature of depth from defocus [19]. Thus, a circle centered at (x, y) with diameter ℓ on the image plane can be drawn, as in Figs. 4.b and b'. $\alpha(x, y)$ is the proportion of the circle that overlaps with the raindrop.

Having obtained α, we recover I_e from the frequency domain. According to Eq. (8), the high frequency component of raindrop I_r is negligible. Thus, for frequency higher than a threshold ω_{th}, we have:

$$\mathcal{I}_e(x, y, \omega) = \frac{1}{1-\alpha(x, y)}\mathcal{I}(x, y, \omega), \quad \omega > \omega_{th}, \tag{10}$$

where $\mathcal{I}(x, y, \omega)$ is the Discrete Cosine Fourier Transform (DCT) of $I(x, y, t)$ on N consecutive frames. ω_{th} is set as $0.05N$ as default. As for the low frequency component, we replace it with the mean of its spatial neighborhood:

$$\mathcal{I}_e(x, y, \omega) = mean(\mathcal{I}(x + \Delta x, y + \Delta y, \omega)),\ \omega \leq \omega_{th}, \tag{11}$$

Algorithm 2 Raindrop removal

if (default)
$N = 100$, $\omega_{th} = 0.05N$, $\Delta x = \Delta y = -1 : 1 pixel$
$th1 = 250$, $th2 = 40$
end
load N continuous frames
Calculate $\alpha(x, y)$ for each pixel $I(x, y, \cdot)$.
if $(max(I(x, y, \cdot)) > th1$ & $\alpha(x, y) > 0)$ $\{(x, y)$ is glare$\}$
for (non-glare pixels and $0 < \alpha(x, y) < 0.9$)
for ($(R; G; B)$ channel separately)
while ($\exists$ pixel unprocessed)
find unprocessed pixel with smallest α $(I(x, y, \cdot))$
find neighbors of (x, y) in $(x + \Delta x, y + \Delta y)$
remove neighbors with intensity difference $> th2$
Do DCT: $\mathcal{I}(x, y, \omega) = I(x, y, t)$
$\mathcal{I}(x, y, \omega_{th} : N) = \frac{1}{1-\alpha(x,y)}\mathcal{I}(x, y, \omega_{th} : N)$
$\mathcal{I}(x, y, 1 : \omega_{th}) = mean(\mathcal{I}(x+\Delta x, y+\Delta y, 1 : \omega_{th}))$
do i-DCT
end
end
end
Repair the remaining areas using Wexler *et al.*'s method.

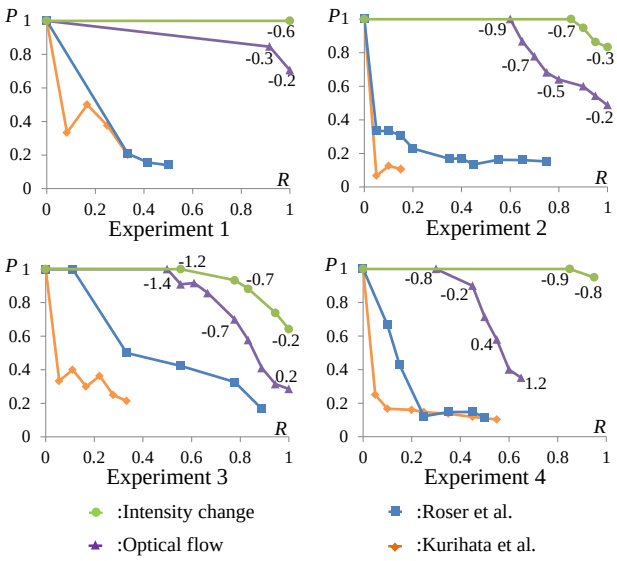


Figure 8. The precision(R)-recall(R) curves of our methods and the two existing methods. The thresholds of our normalized features are labeled.

where $(x + \Delta x, y + \Delta y), \Delta x, \Delta y \leq 1pixel$ are spatial neighborhood of (x, y). When averaging, we exclude neighboring pixels that have intensity difference larger than 40 (in 8-bit RGB value). By combining Eqs. (10) and (11), and performing inverse-DCT, we recover $I_e(x, y, t)$.

Video Completion After restoring the partially occluding raindrop pixels, there are two remaining problems need to be completed: (1) when α is close or equal to 1.0, I_e will be too scarce to be restored, as shown in Eq. (10). We do not restore pixels with $\alpha > 0.9$. (2) When there is glare, the light component from raindrop will be too strong and therefore saturated.

For those areas, we adopt Wexler *et al.*'s [21] space-time video completion method. As discussed in the related work, the method [21] only assumes that missing data reappears elsewhere in the video, which is most likely to be satisfied in outdoor scenes. The overall algorithm of our proposed raindrop removal algorithm is shown in Algorithm 2.

6. Experiments and Applications

We conducted quantitative experiments to measure the accuracy and general applicability of our proposed detection and removal method. Results in video are included in the supplementary material.

Quantitative Tests on Raindrop Detection We created a lab data by dropping water on a transparent panel as the ground truth and taking videos in the real world. We had a few scenarios for the experiments. Experiment 1 included

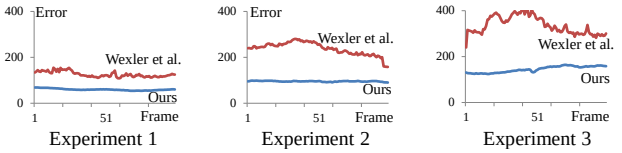


Figure 10. The average $(R; G; B; dx; dy; dt)$ error on recovering 100 continuous frames of the experiments shown in Fig. 11.

the disturbance of the light sources. Experiment 2 emphasized on the varying shape and size of raindrops. Experiment 3 focused on significantly blurred raindrops. Experiment 4 included glare. The input and results are shown in the first four columns in Fig. 9.

We compared our method with Roser *et al.*'s [14] and Kurihata *et al.*'s [11] method. Yamashita *et al.*'s [23, 22] methods require stereo cameras or a pan-tile camera and were, thus, not included in the comparison. The results are shown in the last two columns of Fig. 9.

We used the precision-recall curve to quantitatively analyze the performances. Precision is defined as the number of the correct detection divided by the number of all the detection. Recall is defined as the correct detection divided by the number of the detectable raindrops. The results for each experiment are shown in Fig. 8. According to the results, both of our proposed method outperformed the existing methods. The detection using the intensity change performed best. Unlike the existing methods that only detect the center and size of raindrops, our proposed method can detect raindrops with a large variety of shapes. Our method also achieved high robustness in detecting highly blurred and glared raindrops.

Raindrop Detection Fig. 9 shows the results of our detection method in the following 3 situations: (1) A daily use hand held camera, as in experiments 1-4. (2) A vehicle-mounted camera, which is widely used for navigation and data collection. (3) A surveillance camera which was stuck into one place. Our method outperformed the existing methods in the all three situations.

Quantitative Tests on Raindrop Removal As illustrated in the first two columns of Fig. 11, the synthesized raindrops were generated on a video, and became the input. Our method was compared with the method proposed by Wexler *et al.* [21]. The other video repairing methods were not included because their constraints cannot be satisfied in outdoor conditions. In [14], there is insufficient description for the removal algorithm and thus it was not compared here. The results are shown in the last four columns of Fig. 10.

As shown in Fig. 10, for the quantitative evaluation, we ran each of them on 100 continuous frames and calculated the average error per pixel for each frame. The same as Wexler *et al.* [21], the error was calculated on both the 8 bit

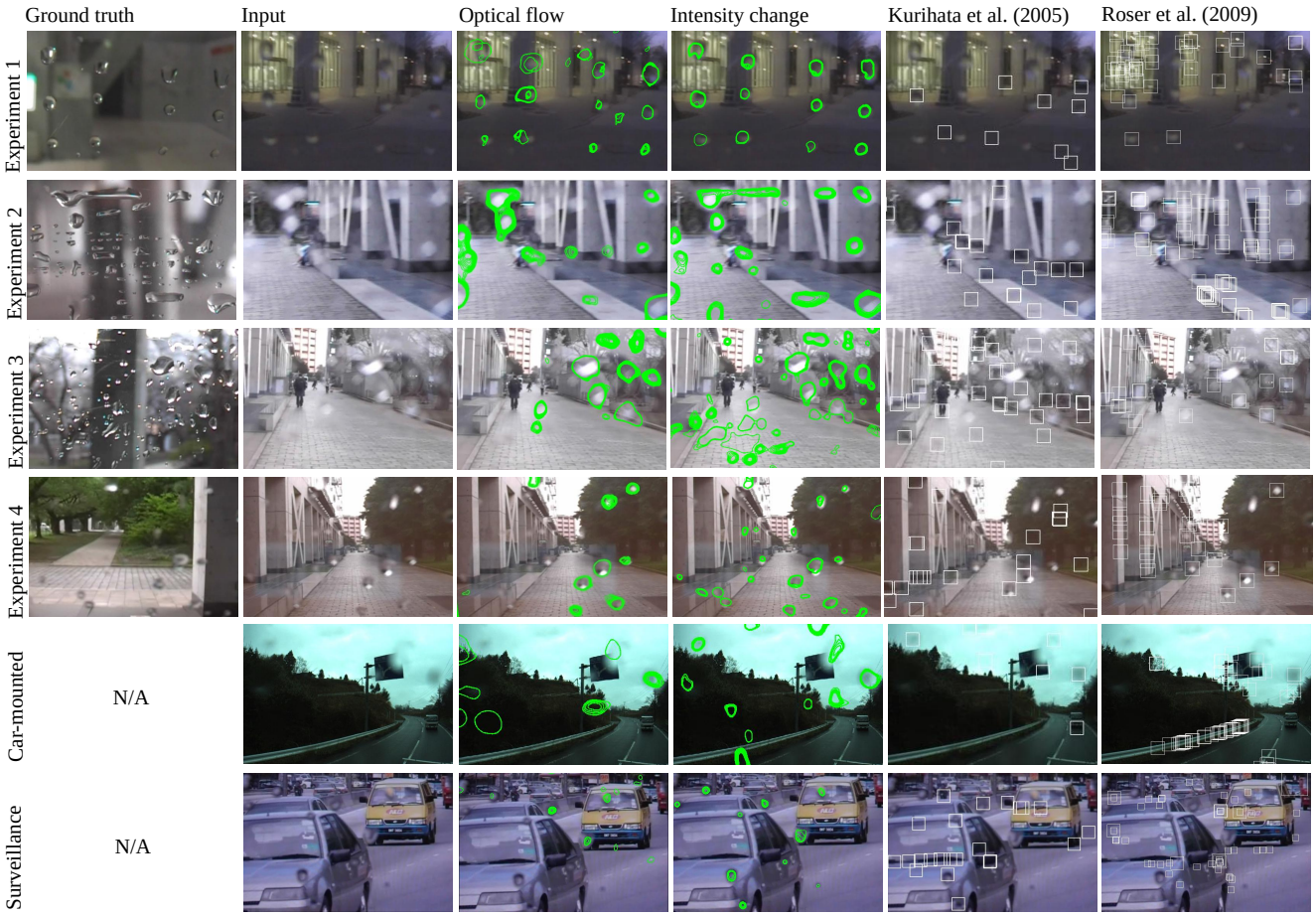


Figure 9. The detection experiment using our methods and the existing methods.

$(R; G; B)$ value and spatial-temporal gradients $(dx; dy; dt)$. The proposed method benefits from the restoration in all the 3 situation. Using the same computer, our method needed 5 seconds per frame to remove raindrops, and Wexler *et al.*'s needed 2 minutes.

Raindrop Removal We show a few results of removing raindrops in videos taken by a handle held camera and a vehicle-mounted camera, as shown in the first and second row of Fig. 12 we can see the significant improvement. To demonstrate the performance of our raindrop removal method, the manually labeled raindrops were also included.

Overall Evaluation The overall automatic raindrop detection and removal results in videos taken by a hand held camera and a car mounted camera are shown in the third row of Fig. 12, where we can see the significant visibility improvement.

7. Conclusion

We have introduced a novel method to detect and remove adherent raindrops in video. The key idea of detecting raindrops is based on our theoretical findings that the motion of raindrop pixels is slower than that of non-raindrop pixels, and the temporal change of intensity of raindrop pixels is smaller than that of non-raindrop pixels. The key idea on raindrop removal is to solve the blending function with the clues from detection and intensity change in a few consecutive frames, as well as to employ a video completion technique only for those that cannot be restored. To our knowledge, our automatic raindrop detection and removal method is novel and can benefit many applications that suffer from adherent raindrops. One limitation of our work is that it does not work with highly dynamic raindrops.

Acknowledgment

This research is granted by the Japan Society for the Promotion of Science (JSPS) through the "Funding Program for Next Generation World-Leading Researchers (NEXT Program)," initiated by the Council for Science and Technology Policy (CSTP).

References

[1] P. Barnum, S. Narasimhan, and T. Kanade. Analysis of rain and snow in frequency space. *IJCV*, 86(2-3):256–274, 2010. 2

Figure 11. The raindrop removal results using our methods and the method of Wexler *et al*. [21].

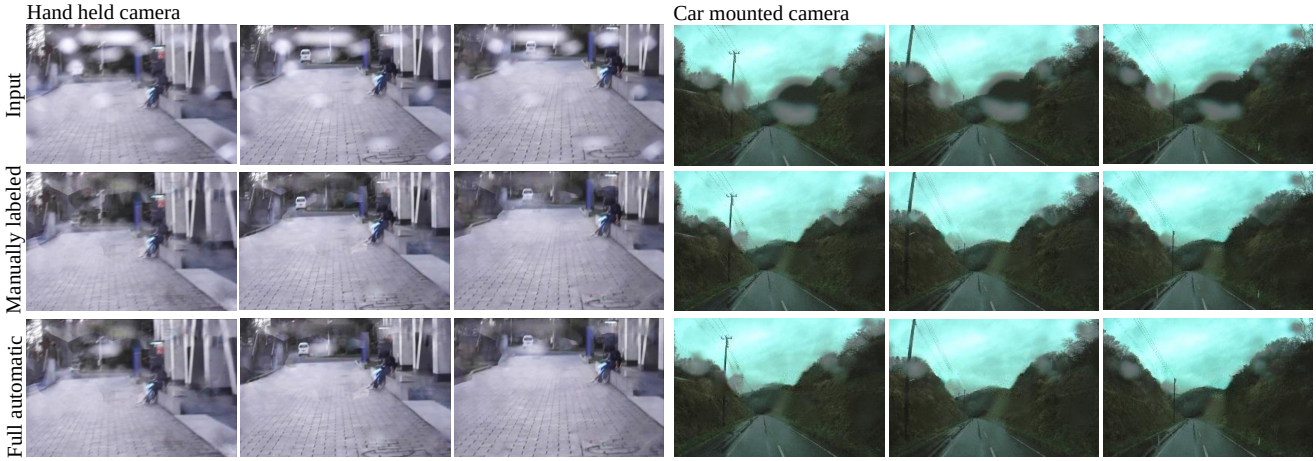


Figure 12. The raindrop removal using the our method. First row: the input sequence. Second row: the removal result with the raindrops manually labeled. Third row: the removal result with the raindrops automatically detected.

[2] R. Fattal. Single image dehazing. *SIGGRAPH*, 2008. 2

[3] K. Garg and S. Nayar. Photometric model of a raindrop. *CMU Technical Report*, 2003. 2

[4] K. Garg and S. Nayar. Detection and removal of rain from video. *CVPR*, 2004. 2

[5] K. Garg and S. Nayar. Vision and rain. *IJCV*, 75(1):3–27, 2007. 2

[6] T. Hara, H. Saito, and T. Kanade. Removal of glare caused by water droplets. *Conference for Visual Media Production*, 2009. 2

[7] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. *CVPR*, 2009. 2

[8] J. Jia, Y. Tai, T. Wu, and C. Tang. Video repairing under variable illumination using cyclic motions. *TPAMI*, 28(5):832–839, 2006. 2

[9] J. Jia, T. Wu, Y. Tai, and C. Tang. Video repairing: Inference of foreground and background under severe occlusion. *CVPR*, 2004. 2

[10] L. Kang, C. Lin, and Y. Fu. Automatic single-image-based rain streaks removal via image decomposition. *TIP*, 21(4):1742–1755, 2012. 2

[11] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara. Rainy weather recognition from in-vehicle camera images for driver assistance. *IEEE Intelligent Vehicles Symposium*, 2005. 2, 6

[12] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *TPAMI*, 33(5):978–994, 2006. 4

[13] M. Liu, S. Chen, J. Liu, and X. Tang. Video completion via motion guided spatial-temporal global optimization. *ACMM*, 2009. 2

[14] M. Roser and A. Geiger. Video-based raindrop detection for improved image registration. *ICCV Workshops*, 2009. 2, 5, 6

[15] M. Roser, J. Kurz, and A. Geiger. Realistic modeling of water droplets for monocular adherent raindrop recognition using bezier curves. *ACCV*, 2010. 2

[16] G. Sapiro and M. Bertalmio. Video inpainting under constrained camera motion. *TIP*, 16(2):545–553, 2007. 2

[17] T. Shiratori, Y. Matsushita, S. B. Kang, and X. Tang. Video completion by motion field transfer. *CVPR*, 2006. 2

[18] O. Stanley and F. Ronald. Level set methods and dynamic implicit surfaces. *Springer-Verlag*, ISBN 0-387-95482-1, 2002. 4

[19] M. Subbarao. Depth recovery from blurred edges. *CVPR*, 1988. 5

[20] R. Tan. Visibility in bad weather from a single image. *CVPR*, 2008. 2

[21] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. *CVPR*, 2004. 2, 6, 8

[22] A. Yamashita, I. Fukuchi, and T. Kaneko. Noises removal from image sequences acquired with moving camera by estimating camera motion from spatio-temporal information. *IROS*, 2009. 2, 5, 6

[23] A. Yamashita, Y. Tanaka, and T. Kaneko. Removal of adherent waterdrops from images acquired with stereo camera. *IROS*, 2005. 2, 6