Low latency mitigation of rain induced noise in images

DOI:
10.1049/cp:20081070

Link to publication record in Manchester Research Explorer

Citation for published version (APA):

Published in:
IET Conference Publications/IET Conf Publ

Citing this paper
Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights
Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy
If you believe that this document breaches copyright please refer to the University of Manchester’s Takedown Procedures [http://man.ac.uk/04Y6Bo] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.
LOW LATENCY MITIGATION OF RAIN INDUCED NOISE
IN IMAGES

M F Subhani, Dr. J. P. Oakley

University of Manchester, mfsabhani@ice.org

Keywords: particle mitigation, image enhancement, image restoration, weather degradation.

Abstract

Sometimes, weather conditions, particularly rain and snow, makes it difficult to obtain clear image or videos. Partial occlusion by moving rain drops makes it difficult to process such videos for example in object recognition and feature tracking. The difficulty further increases in presence of camera and objects motion. Motion analysis is typically performed to avoid distortions in camera and object motion. This paper proposes an image processing procedure to carry out rain removal in image sequences without explicit motion analysis. The proposed method analyses temporal changes (with 1 frame latency) and selects rain corrupted image pixels. The selected pixels are further filtered for camera and object motion by using spatio-temporal consistency constraints. The corrupted pixels are recovered by intensity subtraction. A naive selection of algorithms shows comparable results. The proposed scheme has various applications for example outdoor surveillance, feature tracking and object recognition.

1 Introduction

A camera views a scene by the amount of light reflected from the scene. Raindrops appear between scene and camera, therefore change the amount of light by reflecting or refracting it. Images obtained in rain, may have their contents partially obscured by rain streaks. The occlusion makes it difficult, for both human and machines, to recognise objects or features in rainy images [1]. Weather conditions, for example rain and snow, contain moving particles therefore referred as dynamic weather [4](see for example Figure 1).

The particles (rain drops and snow flakes) are usually large enough to be visible to camera. The particulates affect image intensity in a complex way and to some extent partially occlude the necessary image information (see Figure 1). Studies in the field of atmospheric sciences, especially remote sensing, show that the affect of particles in dynamic weather are predictable [2,12,9].

Dynamic weather affects have been removed to some extent by use of temporal filtering [11]. However, this causes blurring especially when camera or objects move. Zhang et al [13] suggest correction for camera motion prior to rain removal. The idea is to extract camera motion, by frame alignment, before processing and once processed for rain removal the extracted motion is injected again in the video. Frame alignment may not be possible for example in presence of object motion. Extraction of motion information in presence of rain is difficult [1] and may produce unreliable results. In more recent research, Garg and Nayar [6] suggest segmentation of individual rain streaks. They assume stationary background within a predefined spatial locality. This allows removal for limited dynamic scenes, however, fails when rain is seen against a moving textured background. Barnum et al [1], overcome this problem by extracting global spatial frequency information of rain to find and remove corrupted image regions. The estimation of rain in frequency domain requires rain streak orientation and size distribution. The method shows promising results; however, it requires rain streaks to be globally consistent, at least in orientation. The scheme also extracts motion information over 30 frames to avoid image distortions.

In this work we use local spatio-temporal signatures of rain drops to detect individual corrupted pixels. The proposed approach requires neither individual position of raindrops in 3D space nor any explicit motion analysis. The process has a latency of only 1 frame which is less than that of existing rain removal methods. This also allows quick adaption with sudden changes, in both scene and rain. A selection of algorithms show results comparable to recently published research [1].

2 Rain detection and removal

The affects of rain are complex to model and depends on several factors including environmental illumination, occluded scene, drop size distribution and velocities. Physical properties of rain have been extensively studied in various
applications [4,7,9,12]. Details about physical properties of rain drops are out of scope of this paper.

A scene without rain forms a clear image with intensity values obtained according to the light received by the camera within the solid angle of a pixel. A pixel gets corrupted when the actual intensity value is changed by raindrop’s reflection or refraction. Photometry of a raindrop [7] is out of scope of this paper. To understand the affect on intensity of a pixel, lets assume a single pixel \( P(x,y) \) where \( (x,y) \) is spatial position on image grid. At time \( t \) a pixel can either be corrupted by rain or clear. For each pixel we create a binary flag \( b \), where

\[
b(x,y,t) = \begin{cases} 
1, & \text{If a pixel is corrupt} \\
0, & \text{otherwise}
\end{cases}
\]  

(1)

This is initially set to 0 for all image pixels.

Consider two images of the same scene obtained in rain at different time \( t \) and \( t-\Delta t \), where \( \Delta t \) is the camera integration (exposure) time. Intensity of a pixel at \((x,y)\) in an image at time \( t \) is \( I(x,y,t) \), and at time \( t-\Delta t \), pixel at same position, has an intensity of \( I(x,y,t-\Delta t) \). Water droplets in rain move at high velocity therefore we can assume that a pixel does not remain corrupted in both images [5]. We are not concerned with the situation when both images have clear pixels. The only possibilities left are either a pixel is corrupted at time \( t \) or \( t-\Delta t \).

Suppose \( I(x,y,t) \) is a clear pixel and \( I_s \) is the amount of light received by camera at \((x,y)\). Ignoring attenuations we can express intensity of a clear pixel at \((x,y)\) as:

\[
I(x,y,t) = I_o
\]  

(2)

Raindrops are made of clear material and react as lenses. They reflect and refract light from scene towards the camera. If the transmittance of raindrop is \( T \), then the amount of light reaching a corrupted image pixel \( I(x,y,t-\Delta t) \) can be expressed as:

\[
I(x,y,t-\Delta t) = T(I_s + I_o) + R_s
\]  

(3)

Where \( I_s \) is the light incident towards the raindrop excluding \( I_o \) and \( R_s \) is the specular reflection of light from rain drop. However solid angle of the raindrop is larger than that of its occluded background since it is close to camera and this allows more light towards the raindrop. According to Garg and Nayar [5] there are other factors, for example specular and internal reflections from a drop, which also increase this brightness of raindrops [6]. Since raindrop reflects light from a large solid angle its brightness is less dependent on intensity of the scene occluded by raindrop. We obtain temporal difference between two intensities according to equation (4).

\[
D(x,y,t) = I(x,y,t) - I(x,y,t-\Delta t)
\]  

(4)

Pixel intensities that have not changed over time are marked as clear pixels. For each pixel if \( D(x,y,t) \) is zero its binary flag is cleared (set to zero) and all other pixels are selected for processing.

When intensity increases at time \( t \) as compared to \( t-\Delta t \) then \( D(x,y,t) \) becomes greater than zero. All such pixels are potential candidates for rain corrupted pixels in image at time \( t \). All positive values in difference signal are separated and stored as a positive difference \( Dp \) as shown in equation (5). Similarly, negative values belong to pixels affected by rain at time \( t-\Delta t \), and stored as negative difference \( Dn \) as shown in equation (6).

\[
Dp(x,y,t) = \begin{cases} 
D(x,y,t), & D(x,y,t) > 0 \\
0, & \text{otherwise}
\end{cases}
\]  

(5)

\[
Dn(x,y,t) = \begin{cases} 
|D(x,y,t)|, & D(x,y,t) < 0 \\
0, & \text{otherwise}
\end{cases}
\]  

(6)

Raindrops are randomly distributed in spatio-temporal domain. Camera or object motion also cause changes in intensity over time. Thus \( Dp \) and \( Dn \) contains changes due to rain and changes due to scene. Changes due to object or camera motion may be positive or negative, depending on the texture at position of movement. Rain appear in both images thus the affect of rain is present in both \( Dp \) and \( Dn \). Raindrops within a small spatio-temporal window have similar affect on intensity[6]. The signatures of rain in \( Dp \) and \( Dn \) are similar whereas it is rare for scene objects to show this behaviour. Even when a bright object leaves a dark texture, the intensity of \( Dp \) and \( Dn \) will be different unless the object is small and moves on a texture less dark surface. A 1D intensity signature in \( Dp \) and \( Dn \) due to motion of both rain streaks and scene are shown in Figure 2.

![Figure 2: Intensity differences due to Motion of (row 1) Scene Contents (row 2) rain drops.](image)

All Non-zero \( Dp \) and \( Dn \) values are selected for further processing. Due to fast motion and out of focus blur the difference \( Dp \) or \( Dn \) are not sharp and contain blurred artefacts. A blurred Gaussian is generally correct estimate of blurring function for rain streaks [1]. The affect produced is of high intensity at centre due to blurring therefore we search all local maxima in both \( Dp \) and \( Dn \). Two separate lists \( Lp \) and \( Ln \) of local maxima position and intensity are formed from \( Dp \) and \( Dn \) respectively.

For each local maxima in \( Lp \) we search a nearest local maxima in \( Ln \). Euclidean distance between nearest neighbour is measured and stored in the list as well. If nearest neighbour is not within pre-defined radius then it is marked as clear
pixel otherwise its flag is kept unmarked for further processing. At this stage it is possible that a moving object with a local maximum may confuse the algorithm. However it is rare for scene object to have similar signature in \(D_p\) and \(D_n\) with a local maxima within small spatio-temporal window. For generality we leave the problematic scene where scene objects are bright and comparable in size with rain drops.

The intensity of \(L_p\) is subtracted from intensity of nearest valid \(I_n\) and the absolute difference is stored. The local consistency of rain is confirmed by applying pre-selected intensity threshold. All local maxima with intensity difference above threshold are marked as clear pixels and the rest of the pixels are processed further.

The Euclidean distance between nearest neighbours for each marked local maxima is used to search corrupted pixels. All non-zero \(D_p\) values with a radius of half Euclidean distance are flagged as corrupted pixels.

The intensity values of corrupted pixels are recovered by subtracting \(D_p\) from Intensity value of flagged pixels.

\[
l_e(x,y,t) = \begin{cases} 
I(x,y,t) - D_p(x,y,t) : b(x,y,t) = 1 \\
I(x,y,t) & \text{otherwise}
\end{cases}
\]  

3 Results

The performance of proposed scheme is compared with existing schemes [1] and [6] using video sequences used in corresponding research. The overall usefulness of image is application dependent therefore a general subjective analysis for enhanced videos is presented.

For all videos, the processing is carried out on 8-bit grayscale images and a threshold of 8 is used. Spatial contents inside radius of 11 pixels are considered local [6] when searching for nearest neighbour. The enhanced video has a lag of 2 frames since two consecutive images are required for processing.

Three clear sequences obtained from [1] are used in conjunction with synthetic rain, obtained according to [4]. The ‘Ground’ sequence is static and all the changes in scene are due to rain. In static scenes it is possible to remove rain without affecting any part of the image since all changed pixels are due to rain. In dynamic scenes the process may cause some distortion. Two other sequences, namely ‘Walkers’ and ‘Trees’ have complex motion of objects. The removal accuracy \(R_a\), is measured by computing the difference between rendered rain component \(T_r\) and the estimated rain component (removed) \(E_r\).

\[
E_r(x,y,t) = I(x,y,t) - I_e(x,y,t)
\]

\[
T_r(x,y,t) = T(x,y,t) - I(x,y,t)
\]

where \(T(x,y,t)\) is the true pixel intensity.

\[
R_a(x,y,t) = \begin{cases} 
T_r(x,y,t) - E_r(x,y,t) & \text{if } E_r(x,y,t) > 0 \\
-E_r(x,y,t) & \text{if } E_r(x,y,t) < 0
\end{cases}
\]

Accuracy is based on two percentage measures of rain removed (Rv) and scene distorted pixels(Sd).

\[
R_v = 100 \times \frac{\sum_{x,y,t} |R_a(x,y,t)|: R_a(x,y,t) > 0}{\sum_{x,y,t} T_r(x,y,t)} \quad (11)
\]

\[
S_d = 100 \times \frac{\sum_{x,y,t} |R_a(x,y,t)|: R_a(x,y,t) < 0}{\sum_{x,y,t} I(x,y,t)} \quad (12)
\]

The evaluation is carried out by separately considering each colour component since evaluating in grayscale does not take account of colour errors. The amount of rain removal and scene corruption by frequency analysis and frame difference analysis is shown in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>Ground</th>
<th>Walkers</th>
<th>Trees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freq/Rv</td>
<td>43%</td>
<td>30%</td>
<td>31%</td>
</tr>
<tr>
<td>FD/Rv</td>
<td>64%</td>
<td>49%</td>
<td>55%</td>
</tr>
<tr>
<td>Freq/Sd</td>
<td>3%</td>
<td>7%</td>
<td>6%</td>
</tr>
<tr>
<td>FD/Sd</td>
<td>0%</td>
<td>2%</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 1: Comparison table for removed rain and scene distortions caused by frequency analysis and frame difference analysis.

Apart from being a low latency process, the proposed scheme is less prone to scene distortions due to its localised matching scheme as opposed to global frequency features. The portion of image distorted by proposed algorithms is less than that distorted by global frequency analysis.

The replacement of corrupted pixel with intensity values from previous image is an obvious cause of distortions (see Figure 3). Notice the removal of bright streak on white stripe (top left) the texture of wall is recovered without any distortions. In a video sequence, since there is a lag of only 1 frame, these distortions are less prominent as compared to previous methods. Figure 4 shows results for images obtained from video sequences with natural rain. Noticeable regions edges of expected moving objects for example people and umbrella.

4 Conclusions

The similar signatures of rain streaks can be separately seen in positive and negative part of frame intensity difference. Generally, the differences due to scene changes, for example motion of objects and camera, does not show such signatures.
The proposed scheme uses the signature of rain to select corrupted pixels in current image. Since intensity difference is subtracted from corrupted pixels, this does not introduce blurring similar to temporal filtering [6]. Frequency analysis[1], temporal and chromatic analysis[13] of rain corrupted pixels require motion analysis to separate camera and object motion however frame difference analysis achieves comparable results without explicit motion analysis. Earlier rain removal methods have a latency of 3 or more frames however this approach only has a maximum of single frame latency. The proposed method is useful in variety of applications including outdoor video retrieval and surveillances. The idea may also be useful for enhancement of medical and underwater imaging.

References