2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

A Two-Step Approach to See-Through Bad Weather for Surveillance Video Quality Enhancement

Zhen Jia, Hongcheng Wang, Rodrigo Caballero, Ziyou Xiong, Jianwei Zhao and Alan Finn

Abstract— **Adverse weather conditions such as snow, fog or heavy rain greatly reduce the visual quality of outdoor surveillance videos. Video quality enhancement can improve the visual quality of surveillance videos providing clearer images with more details.**

Existing work in this area mainly focuses on quality enhancement for high resolution videos or still images, but few algorithms are developed for enhancing surveillance videos, which normally have low resolution, high noise and compression artifacts. In addition, for snow or rain conditions, the image quality of near-filed view is degraded by the obscuration of apparent snowflakes and raindrops, while the quality of far-field view is degraded by the obscuration of fog-like snowflakes or raindrops. Very few video quality enhancement algorithms have been developed to handle both problems.

In this paper, we propose a novel video quality enhancement algorithm for see-through snow, fog or heavy rain. The proposed algorithm has two major steps: 1. the near-field enhancement algorithm identifies obscuration pixels by snow or rain in the near-field view and removes these pixels as snowflakes or raindrops; different from state-of-the-art methods, the algorithm in this step can detect snowflakes on foreground object and background, and choose different methods to fill in the removed regions. 2. the far-field enhancement algorithm restores the image's contrast information not only to reveal more details in the far-field view but also to enhance the overall image's quality; in this step, the proposed algorithm adaptively enhances the global and local contrast, which is inspired on the human visual system, and accounts for the perceptual sensitivity to noise, compression artifacts, and the texture of image content. From our extensive testing, the proposed approach significantly improves the visual quality of surveillance videos by removing snow/fog/rain effects.

I. INTRODUCTION

Video quality enhancement plays an important role in surveillance video applications. Poor video quality reduces the effectiveness of human operators responsible for monitoring the surveillance video displays, and it reduces the accuracy of video analytic algorithms. Improved perceived visual quality and improved video analytic lead in general to more accurate detection of threats and fewer false alarms.

In outdoor surveillance applications, bad weather conditions, such as snow, fog or heavy rain may hide the details of the scene, and significantly reduce the visibility and degrade contrast information of the video signal. As shown in Fig. 1, a scene with snow is usually composed of snow in close view and distant view. The close view has visually large snowflakes, while the distant view is so foggy that single snowflake is not visible. An algorithm is highly desirable for live viewing of surveillance videos or for robust video analytics, which can remove big snowflakes in close view as well as fog-like snow in distant view.

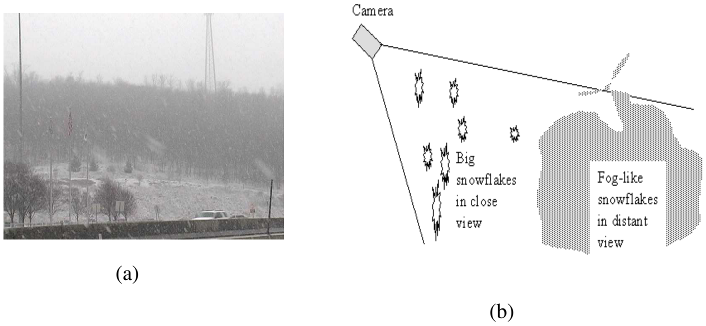

Camera
Big snowflakes in close view
Fog-like snowflakes in distant view
(a)
(b)

Fig. 1. A typical image with heavy snow (Fig. 1(a)) with the illustration example (Fig. 1(b)) for the problems to be solved for see-through snow algorithm development.

This paper proposes a novel algorithm which can improve video visibility during snow, fog or heavy rain weather conditions.

II. LITERATURE REVIEW

A "see-through" algorithm is designed to enhance the perceived visual quality of a video signal when snow, fog or rain is in the field of view. Snow/fog/rain often reduces the image quality such that image details are not visible.

There are in deed very few methods in the literature to deal with see-through snow/fog/rain for visual surveillance. Here we list the major categories of video quality enhancement methods to deal with adverse weather conditions.

- Narasimhan and Nayar's work [9] presents a physics-based model that describes the appearances of scenes in uniformly bad weather conditions. This kind of method requires multiple input images of a scene, which have either different degrees of polarization or different atmospheric conditions. This requirement is the main drawback of this method, since in many situations it is difficult to fulfill. Recently, Tan [14] improves previous physics-based model methods by developing an automated method that only requires a single input image. Most recently, He et al. [6] propose a simple

Zhen Jia and Jianwei Zhao are with United Technologies Research Center (China) Ltd., Shanghai World Financial Center, 100 Century Avenue, Pudong, Shanghai, 200120, P.R. China. {jiaz, zhaoj}@utrc.utc.com

Hongcheng Wang, Rodrigo Caballero, Ziyou Xiong, and Alan Finn are with United Technologies Research Center, 411 Silver Lane, East Hartford, CT 06118, USA. {WangH1,CaballRE,XiongZ,FinnAM}@utrc.utc.com

978-1-61284-385-8/11/$26.00 ©2011 IEEE

but effective image prior - “dark channel prior” to remove haze from a single input image. One limitation of He et al.’s method [6] is that: “when the scene objects are inherently similar to the atmospheric light and no shadow is cast on them, the dark channel prior is invalid”. Overall, this kind of physics-based methods are computationally expensive and not suitable for real-time visual surveillance applications. Another common limitation of most physics-based haze removal methods - the haze imaging model may be invalid for some conditions. More advanced models will be needed to describe complicated weather phenomena, such as the sun’s influence on the sky region, and the blueish hue near the horizon. Also these method are not suitable for generally surveillance videos such as heavy fog or rain conditions, low quality of images with noises and compression artifacts. Under these conditions, it is very hard to get a valid imaging model. Finally this kind of methods are mainly for still images and they do not consider inter-frame information, which will introduce global flickering effects through the video sequence.

- Zhang et al. [17] proposes another kind of method to remove rain effects in the image view. Their method is based on the background and raindrops classification and chromatic property of raindrops to detect and remove raindrops. This kind of methods works mainly for raindrops removal in close view. However the limitation of this kind of methods is that: after raindrops in close view are removed, the fog-like area with distant raindrops or snowflakes is still in the image scene, which still makes image blurry and less clear; their method does not handle the raindrops on the moving objects well, so using their method after raindrop removal holes will be created inside the foreground moving object.
- Histogram processing is another major kind of methods to enhance image contrast [5] for see-through applications. Histogram processing is generally computationally fast and easy to implement, while at the same time it can achieve high performance for surveillance applications. Compared with other histogram based methods, histogram equalization is a commonly used automatic processing method. Histogram equalization (HE) is a technique that aims to maximize the “information efficiency” of the image, in the sense that more frequent pixels should be entitled to a larger intensity range. There are two kinds of histogram equalization methods: global and local. Global histogram equalization method is usually used to compresses the brightness of the pixels to obtain a more uniform exposure characteristics based on statistics of the entire image such as [5] and [16]. This kind of methods does not consider the local image statistics. Local histogram equalization methods like [15] and [2] are proved to provide better performance than global method to reveal more image local details and give stronger image enhancement performance. Some methods like [4][18] just do simple local histogram equalization like mapping the local histograms of different portions of image to the equalized local histograms, while some others do the adaptive local histogram equalization like [15] [2] [13]. One of the most classic adaptive local histogram equalization methods is the Contrast Limited Adaptive Histogram Equalization (CLAHE) method, which was proposed by [19] and widely studied and used, for example in [11] and [8]. The CLAHE method well addresses the limitation of the global histogram equalization method. Therefore we are interested in studying the CLAHE method to make improvements for local contrast enhancement.

In order to better tackle see-through snow/fog/rain problem, we should provide an algorithm to handle both distant and close view snowflakes/raindrops. The algorithm proposed in this paper solves the see-through snow/fog/rain problem that none of the above algorithms alone can do.

III. Two-Step See-through Bad Weather Algorithm

Here we propose one novel algorithm to do two-step processing to enhance the image quality under snow, fog or rain conditions (Fig. 2). We first do snowflakes/raindrops removal and then do local adaptive contrast enhancement. In this way we can first remove the bigger snowflakes/raindrops in close view like removing “noises” and then further enhance the image contrast to increase the image visibility for the fog-like areas with distant smaller snowflakes/raindrops. Here we can also skip the contrast enhancement step if the input video signal already has good contrast information, and the snowflakes or raindrops removal step can be skipped as well if there is only fog effect in the video.

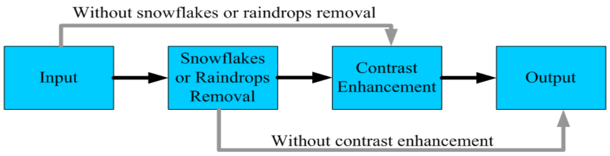

Without snowflakes or raindrops removal
Input
Snowflakes or Raindrops Removal
Contrast Enhancement
Output
Without contrast enhancement

Fig. 2. Flow Chart of the Proposed Algorithm

A. Contributions

In detail, the algorithm proposed in this paper has the following contributions or novelties:

- This method enhances the videos with snow, fog or rain to better visual quality by handling snowflakes or raindrops in both close and distant views.
- This method solves the noises and artifacts problem of over-enhanced snowflakes or raindrops in the image by first removing snowflakes/raindrops and then performing contrast enhancement.
- For the snowflakes/raindrops removal algorithm part in the first step, we have the following contributions or novelties:
    - This method uses the first several frames to train the parameters for close view snowflakes/raindrops

removal. The training can be done off-line, which can save a lot of computational cost.

  - The temporal clustering together with physical property of snowflakes/raindrops is used to generate snow/rain map and clean background map. The snow/rain map and clean background map can be used to accurately classify the current image pixels to find snowflakes.
  - Based on physical dynamic properties of snowflakes/raindrops, we develop a temporal object detection method to distinguish snowflakes/raindrops on background or on a foreground object and then use different methods to fill the removed regions. In this way, the problems of removing foreground object parts and generating "holes" on the foreground object can be avoided.

- For the contribution to the contrast enhancement part, due to the length limitation of this paper, please refer to another paper from us [7] for more technical details.

B. *Snowflakes/raindrops removal algorithm*

For the snowflakes/raindrops removal step, we develop a novel algorithm as shown in Fig. 3. Here we use snowflakes in Fig. 4 as example to demonstrate the snowflakes/raindrops removal algorithm.

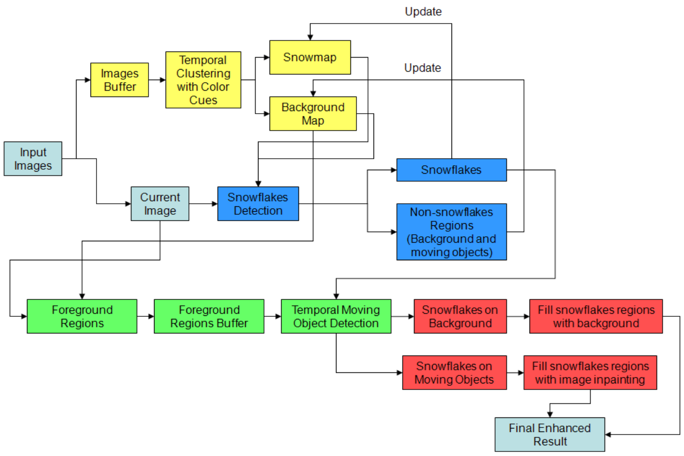

Update
Update
Input Images
Images Buffer
Temporal Clustering with Color Cues
Snowmap
Background Map
Current Image
Snowflakes Detection
Snowflakes
Non-snowflakes Regions (Background and moving objects)
Foreground Regions
Foreground Regions Buffer
Temporal Moving Object Detection
Snowflakes on Background
Fill snowflakes regions with background
Snowflakes on Moving Objects
Fill snowflakes regions with image inpainting
Final Enhanced Result

Fig. 3. Flow Chart of the Proposed Snowflakes/Raindrops Removal Algorithm. Different colors represent different major components in the algorithm. Grey: algorithm input and output part; Yellow: algorithm training part; Blue: snowflakes/raindrops detection part; Green: temporal foreground object detections part; Red: snowflakes/raindrops removed regions filling part.

First, according to properties that snowflakes are small and falling very fast, most of time they are not at the same places among several consecutive frames. As a result one pixel in the image sometimes may belong to snowflakes and sometimes may not. With this property we can use temporal clustering to get snow map and clean background map from a training image sequence. Here we can use the clustering methods like K-means method [17], probabilistic method or template/model based method [3]. In order to implement temporal clustering, we initially have two clusters: one for clean background map and one for snow map. First few frames (such as first 60 frames) from a video input (such as the training sequence on the left side of Fig. 4) are used to do temporal clustering for each pixel. The up figure in the middle of Fig. 4 shows one example. We plot one pixel's intensity values along the time domain, and the pixel values have large variances because it is sometimes the background pixel while sometimes blocked by snowflakes. Therefore if we cluster this pixel's values into two clusters, the centroids of two clusters will represent the snowflakes and background pixel. Here we need to have the assumption that: there should be no major moving objects in the training frames for temporal clustering, because the moving objects can not be clustered into either snow map cluster or background map cluster and it will affect the accuracy of temporal clustering.

After clustering, the snow map and the clean background map can be classified out for the following frames processing (like the two figures on the right side of Fig. 4). Here in order to distinguish snow map and background map, we use the general color properties of snowflakes [1] (like the bottom figure in the middle of Fig. 4): "Snow is made of ice crystals, and up close the individual crystals look clear, like glass. Incident light is partially reflected by an ice surface. Snowflakes have a lot of partially reflecting surfaces, and then incident light bounces around and eventually scatters back out. Since all colors are scattered roughly equally well, the snow bank appears **white or near white**." Also according to [17], the chromatic property that the background object color changes behind raindrops or snowflakes are with very similar values can be employed here. With the above two properties, we can decide whether this cluster represents snowflakes or background pixel.

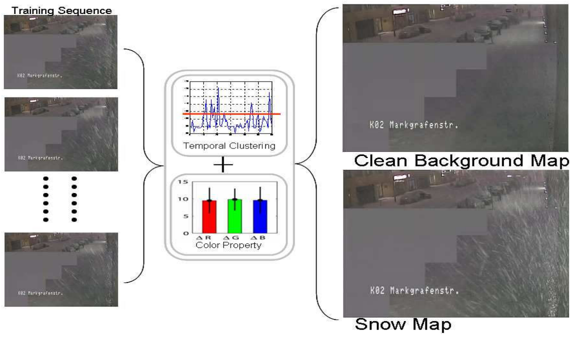


Fig. 4. This figure shows the training example. Here we use a video sequence full of snow. For each image pixel, we do the temporal clustering to generate two clusters: one for snow and one for background. After training process, we use the snowflakes' color property to distinguish the clusters between snow and background.

For the snow map shown in Fig. 4 we can still see some background areas. This is because: for some areas there are always no snowflakes within the training sequence and as a result the two clusters will have very similar values, which both represent the background pixel. For this case, if a foreground pixel does not belong to either cluster, we can also tell that this pixel is a foreground pixel and later if the color of this pixel is close to white, we can label this pixel as one potential snowflake pixel. The training process in this procedure can also be done off-line without causing any computational cost for real-time visual surveillance performance. For the video in Fig. 4, it is the surveillance videos captured from real sites and the surveillance company logos are masked. However, one limitation of this step is that if the background color is similar to white or near white (close to snowflakes' colors), then we will have some mis-classifications. This limitation is reasonable because computer vision algorithms (or even human) can not distinguish the background and foreground if they are very similar.

Second, for the following frames based on each pixel's RGB color differences to the snow map and background map's same location pixels (two figures on the right side of Fig. 4), we can determine whether this pixel belongs to snowflake pixels or not. If this pixel belongs to background (pixel's value closer to background map pixel's value), we just keep it. On the other hand for snowflake pixels (pixel's value closer to the snow map pixel's value), we use the next step to further determine whether this pixel is a snowflake pixel on background or a snowflake on the foreground objects.

Third, we present a novel temporal object detection method to find whether this snowflake pixel is on the foreground object or not.

1) As shown in Fig. 5, we do foreground object detection. The detection is based on background subtraction. The step here is to detect the major moving objects in the image scene. With the foreground objects information, we can next tell whether the detected snowflakes are on the foreground objects or on the background.

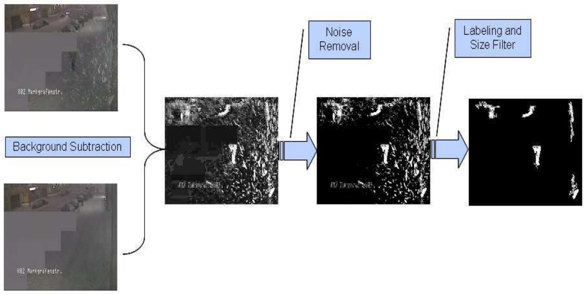


Fig. 5. This figure shows background subtraction process to detect major foreground objects. We have the following steps: the difference between the current image and the clean background map from the first step, noise removal by removing non-significant foreground regions and morphological filtering (erosion and dilation), labeling and then size filtering (small foreground regions are removed).

2) As shown in Fig. 5, the major foreground regions contain both real moving objects and some moving snow regions. In order to differentiate them, we develop the basic rule for this temporal object detection as follows: The basic principle behind this temporal object detection rule is: according to properties that snowflakes are small and falling very fast, most of time they will not be at the same places among several consecutive frames.

$$\begin{aligned}
&if((\mathrm{var}(Min(d(P(x,y),c_i))_{(t-n):t})<thresh)\\
&\&\&(Min(d(P(x,y),c_i))<thresh1))\\
&then\quad P(x,y)\in SnowflakesOnForegroundObject\\
&else\quad P(x,y)\in SnowflakesOnBackground\ \ \textbf{Rule}
\end{aligned}$$

Fig. 6. Temporal Object Detection Rule

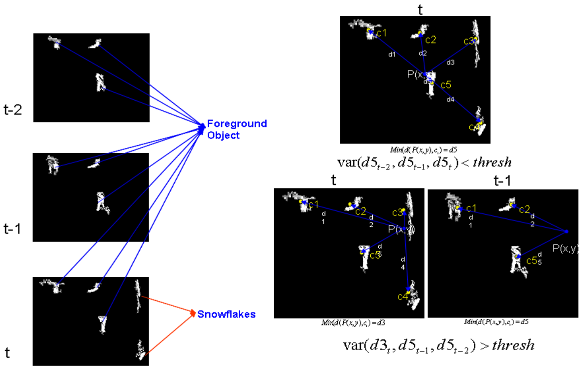


Fig. 7. Fig. 7 illustrates one example by temporal object detector to determine whether an identified snowflake pixel is located over background or foreground objects. On the left part, images captured at three successive points in time (e.g., time $t-2$, $t-1$, and t, respectively) are shown. For each image, foreground objects (including snowflakes/raindrops) are identified ($c1$ $c5$).

For one pixel belonging to the background snowflakes (like the pixel $P(x,y)$ in the bottom right two figures of Fig. 7), its minimum distance to foreground regions centers will be very different among consecutive frames because some foreground regions move very fast or even the same foreground regions appear in some frames while disappear in other frames. Therefor, for every snowflake pixel, we first estimate its distances to the detected foreground regions' centers (such as $c1$ to $c5$ in Fig. 7). From all these distances we estimate the minimum distance ($d5$ for the upper right figure and $d3$ and $d5$ for the two bottom right figures in Fig. 7). Next, for several consecutive frames, we estimate the variance of these minimum distances. If the variance and minimum distance follow the above rule in Fig. 6, then this snowflake pixel is on the foreground object, otherwise it is a snowflake on the background. The thresholds here are decided by trail and error method.

Fourth, having determined that a snowflake is located over a background portion of the image, the pixels making up the snowflake are filled with background map pixels based on a background filling method. A number of well-known algorithms may be employed to provide background filling [10]. For example, here background filling is based on the following equation 1:

$$P = \alpha * P_b + (1-\alpha) * P_f \tag{1}$$

P_b is the value of the background pixel obtained from the background map, P_f is the value of the foreground object, and α is a weighting factor that determines how much significance to be given to the background pixel versus the foreground pixel. In the case of filling in snowflakes, the weighting factor may be close to one, to weight the significance of the background pixel values more highly.

For snowflakes located over foreground objects, filling in pixels using Eq 1 would create holes in the foreground objects that are undesirable. Therefore, rather than employ background filling, those pixels identified as located over foreground objects are in-painted with pixels associated with the foreground object. A number of methods of image inpainting are well-known in art. For example, in [12] a mask of pixels identified for inpainting (i.e., those snowflake pixels identified over a foreground object) are provided as part of a mask. The masked pixels are then filled based on a form of diffusion in which nearby pixels are used to determine the value of pixels located within the mask.

After the above four steps, we provide the near-field enhancement to enhance the near-field images by removing the near-field view snowflakes (result such as Fig. 8).

C. Content based local adaptive contrast enhancement

In addition to near-field enhancement, local adaptive contrast enhancement algorithm is developed to enhance the far-field fog-like part. The basic idea of this content based local adaptive contrast enhancement is to enhance more in structured (or textured) regions to bring out more image details, but enhance less in homogeneous (or flat) regions to avoid enhancing the noise and the compression artifacts. Due to the length limitation of this paper, in another paper from us [7], we described in details this enhancement algorithms. Please refer to [7] for more technical details.

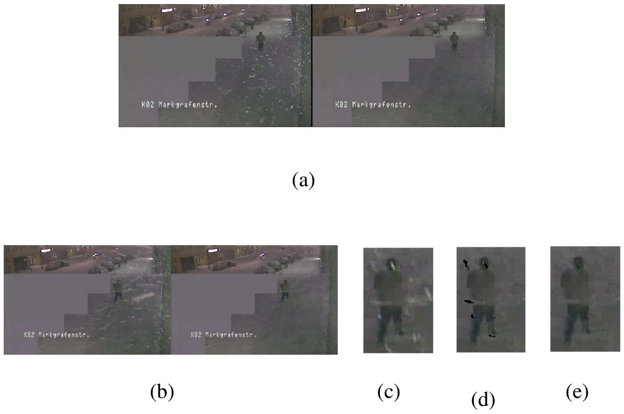
IV. Experimental Results

Extensive experiments were conducted to validate the effectiveness of the proposed algorithm.

Fig. 8 shows the snowflakes removal's final result of a challenging video after all the processing steps in Section III. The snowflakes in the video are large in the close view, and they obscure parts of both the foreground and background objects. From Figs. 8(c) to 8(d), we can see that first the snowflakes around the persons are removed and filled with background regions, while the snowflakes on the person's body regions are detected as black regions. Fig. 8(e) shows the final image inpainting result, which does not cause any "hole" effects on the foreground objects. The snowflakes are clearly removed and removed regions are filled with background information or image inpainting information.

In Fig. 9 we show another result for snowflakes removal. In this example, the video scene does not have large moving foreground objects, but the background trees are swaying. From the result, we can see that the snowflakes can also be accurately removed by our algorithm.

For the experimental results of contrast enhancement step, please refer to [7] for more details. Next we will show some final results after the two-step processing.



(a)

(b) (c) (d) (e)

Fig. 8. Snowflakes removal results. In Fig. 8(a) and Fig. 8(b) we compare the input image and the final result image before and after snowflakes removal. Fig. 8(c) shows the foreground region from the input image (the left figure of Fig. 8(b)). The black areas in Fig. 8(d) indicate the detected snowflakes regions on foreground object. Fig. 8(e) shows the final result after image inpainting processing.

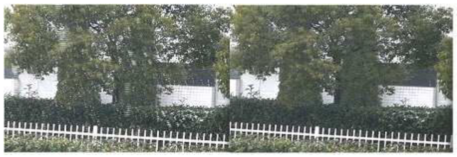
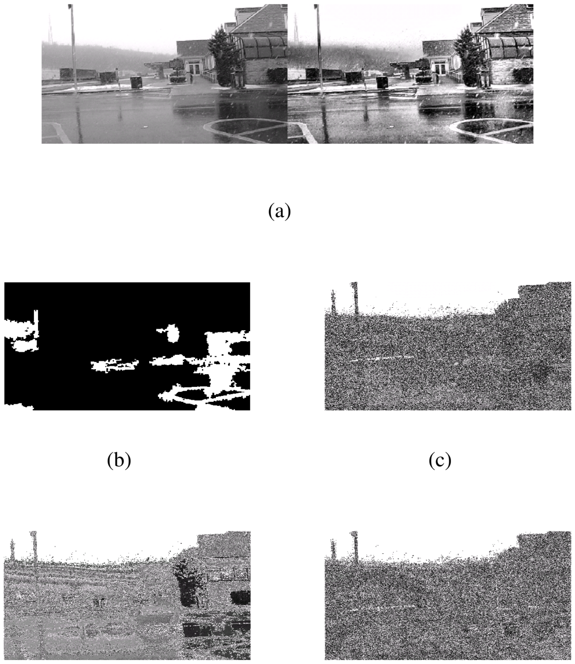


Fig. 9. Snowflakes removal result. Left: an image from the input video with snow; Right: output after snowflakes removal.

In Fig. 10(a), if we first do contrast enhancement, the snowflakes are also enhanced and as a result the snowflakes are more apparent to human, which can be considered as the enhanced noises. Here if we do snowflakes removal after contrast enhancement, it will cause serious errors and artifacts in the result image because the current image scene is changed due to contrast enhancement and the training data is based on normal image scenes, the color differences between the current image and background map and snow map will be too large to correctly detect snowflakes and foreground objects. Fig. 10(c) shows one example, and we can find that there are still some snowflakes in the final result together with noises and artifacts (Fig. 10(c)) and there are large false foreground regions due to contrast enhancement (Fig. 10(b)). On the other hand, in Fig. 10(d) we first do snowflakes removal and then conduct content adaptive contrast enhancement. In the final result image (Fig. 10(e)), there are almost snowflakes and the image's overall quality is greatly enhanced with much less noise and artifacts.

Fig. 11 shows another video quality enhancement result after all the processing steps. The final result image (the right figure) has very few snowflakes and the image is much clearer than the original input image.

The parameters used for our testing are mainly determined based on our extensive experimental test and trial.



(a)

(b)

(c)

(d)

(e)

Fig. 10. Here we show some comparison results between first doing contrast enhancement and first doing snowflakes removal. In Fig. 10(a) the result on the right is by first doing contrast enhancement. Fig. 10(b) shows the false foreground regions if we do contrast enhancement first. In Fig. 10(c) the result shows snowflakes removal after contrast enhancement. Fig. 10(d) shows the results by first doing snowflakes removal. Fig. 10(d) is the final result by first doing snowflakes removal and then performing contrast enhancement.

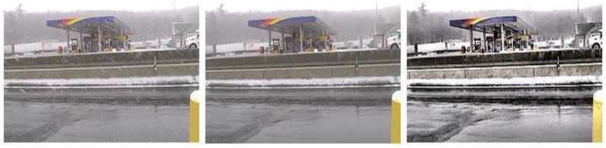


Fig. 11. Final see-through snow result. The left figure is the input figure, the middle figure is the result after snowflakes removal and the right figure is the final output figure.

V. Conclusion and Future Work

This paper proposed a novel video enhancement algorithm that improves the image quality for videos under snow, fog or heavy rain conditions. The system provides near-field enhancement of the video data by detecting the presence of near-field obscuration such as snowflakes/raindrops and determining whether the detected obscuration are located over background pixels or foreground objects. The detected obscuration pixels are filled in depending on whether they are located over the background or foreground to create an near-field enhanced image. The system also provides local/global adaptive contrast enhancement to enhance video in the presence of far-field obscuration by fog-like snowflakes or raindrops in the far-field of view. Our proposed system also solves the noises and artifacts problem of enhanced snowflakes/raindrops in the image by first doing the snowflakes removal. From our extensive testing, this algorithm works well to handle bad weather such as snow, fog or heavy rain conditions, which provides to be a good solution for video quality enhancement. Compared with state-of-the-art approaches, the major novelty of our proposed approach is the fast speed and we can achieve much high efficiency than other methods for real-world surveillance applications.

References

[1] *http*://www.its.caltech.edu/ atomic/snowcrystals/faqs/faqs.htm.

[2] R. Dale-Jones and T. Tjahjadi. A study and modification of the local histogram equalization algorithm. *Pattern Recognition*, 26(9):1373–1381, 2007.

[3] D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2003.

[4] A. Golan and A. Levy. Method of adaptive image contrast enhancement. US Patent 20070031055.

[5] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, 2001.

[6] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. In *Proceedings of 2009 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2009.

[7] Z. Jia, H. Wang, R. Caballero, Z. Xiong, J. Zhao and A. Finn. Real-time content adaptive contrast enhancement for see-through fog and rain. In *Proceedings of 2009 IEEE International Conference on International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1378–1381, March 2010.

[8] H. Malm, M. Oskarsson, E. Warrant, P. Clarberg, J. Hasselgren, and C. Lejdfors. Adaptive enhancement and noise reduction in very low light-level video. In *Proceedings of IEEE 11th International Conference on Computer vision*, pages 1–8, October 2007.

[9] S. Narasimhan and S. Nayar. Contrast Restoration of Weather Degraded Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):713–724, Jun 2003.

[10] M. Piccardi. Background subtraction techniques: a review. In *Proceedings of 2004 IEEE International Conference on Systems, Man and Cybernetics*, pages 3099–3104, October 2004.

[11] A. M. Reza. Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *The Journal of VLSI Signal Processing*, 38(1):35–44, 2004.

[12] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *Proceedings of 2005 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 860–867, June 2005.

[13] R. S. Szeliski. Locally adapted histogram equalization. US Patent 6650774.

[14] R. T. Tan. Visibility in bad weather from a single image. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008)*, June 2008.

[15] Y. Tian, Q. Wan, and F. Wu. Local histogram equalization based on the minimum brightness error. In *Proceedings of The Fourth International Conference on Image and Graphics*, pages 58–61, August 2007.

[16] Q. Wang and R. Ward. Fast image/video contrast enhancement based on weighted thresholded histogram equalization. *IEEE Transactions on Consumer Electronics*, 53(2):757–764, May 2007.

[17] X. Zhang, H. Li, Y. Qi, W. K. Leow, and T. K. Ng. Rain removal in video by combining temporal and chromatic properties. In *Proceedings of 2006 IEEE International Conference on Multimedia and Expo*, pages 461–464, July 2006.

[18] J. Zhao and S. Lei. Methods and systems for automatic digital image enhancement with local adjustment. US Patent 20070092137.

[19] K. Zuiderveld. Contrast limited adaptive histogram equalization. *Academic Press Graphics Gems Series: Graphics gems IV*, pages 474 – 485, 1994.