

カメラの回転を用いた時空間画像からの水滴除去

Removal of Waterdrops from Spatio-Temporal Images by Using Camera Rotation

福地 功† 山下 淳†‡ 金子 透† 三浦 憲二郎†

Isao Fukuchi†, Atsushi Yamashita†‡, Toru Kaneko†, Kenjiro Miura†

† 静岡大学工学部機械工学科 ‡ カリフォルニア工科大学

†Department of Mechanical Engineering, Shizuoka University

‡Department of Mechanical Engineering, California Institute of Technology

E-mail: f0630062@ipc.shizuoka.ac.jp

Abstract

雨天時にビデオカメラによって風景を撮影する際、レンズ保護ガラス面上に水滴が付着することで視野が妨害されることがある。本論文では、そのような画像の視野を明瞭にするために、時空間画像処理を施すことにより画像から水滴を除去する手法を提案する。画像から水滴を除去するために、本手法ではレンズ保護ガラスと一体となったカメラを回転させ画像を取得する。次に、取得した画像を時空間画像として扱い、時空間断面画像上で水滴と被写体との間における軌跡の違いに着目することで水滴領域を検出する。最後に、検出した水滴領域に対し、画像修復手法を用いることで水滴の除去を行う。実験により、本手法の有効性が確認された。

1 序論

屋外に設置されたカメラは、交通状況の確認、犯罪の予防や捜査といった用途などで広く用いられている。しかし、屋外環境下に設置されたカメラの場合、得られる画像の質は環境条件により大きく左右される。特に雨天時における撮影では、カメラを保護するレンズ保護ガラス面上に水滴が付着することにより視野が妨害され、明瞭な画像を得ることができないという問題が生じる(図1)。

ガラス面上に付着した水滴を除去するためには、一般的にワイパーが用いられる。しかし、ワイパーをカメラに取り付けることは困難なことが多く、またワイパーを装着できたとしても、ワイパーの動作時にワイパー自体が視野を妨害するという問題もある。そこで本手法では、このような物理的なアプローチではなく、ソフト的に画像処理の技術を用いて画像から水滴を除去する。

画像処理により画像からノイズを除去する手法とし



図1 水滴による視野妨害

て、プロの絵画修復技師の技能をまねた画像復元手法 [1] がある。この手法は周囲と相関性の高い部分の再現性は良いが、不規則なテクスチャパターンに対しての再現性が悪いという性質がある。同様の問題は画像のフラクタル性を利用した画像復元手法 [2] にも当てはまり、これらの手法は大きなノイズへの適応性が悪いという特徴もある。また、ノイズの位置を人間が逐次指定する必要があるため、自動作業には適していない。

これとは逆に、人間が逐次指定する必要のない手法としては、降雪ノイズ除去における時間メディアンフィルタを用いる手法 [3] や、雨の物理モデルを作成し用いることによる降雨ノイズ除去手法 [4] が提案されている。しかし、これらの手法は共に視野内を移動する雪や雨を対象としたものであり、これらがレンズ保護ガラス面上に付着することで発生するノイズを除去することができない。

上記の手法に対し、レンズ保護ガラス面上に付着した水滴を検出・除去することが可能なものとしては、2 台のカメラを使用する手法 [5] が提案されている。この手法は 2 台のカメラにより取得した 2 枚の画像に対し、正規化相互相関によるテンプレートマッチングを行うことで水滴を検出する。しかし、2 枚の画像で水滴の見え方に違いが生まれたとき正しく対応点を得ることが難しく、

結果として水滴を検出・除去することができない場合がある．また，1 台のカメラを回転させ取得した 2 枚の画像から水滴を検出・除去する手法 [6] もあるが，同様に 2 枚の画像だけでは同じ場所が水滴で隠れた際，水滴を除去することができない．

この問題を解決するため，本手法では複数の静止画像を時系列順に並べた時空間画像を用いる．同じく時空間画像を用いて画像から遮蔽物を取り除く手法 [7] が提案されているが，カメラを並進させて画像を取得するため，視差の影響を受けにくい遠景画像が対象の場合，レンズ保護ガラス面上の水滴と被写体とを区別することが困難である．

以上の点を踏まえ，本手法ではレンズ保護ガラス面上に水滴が付着した状態でカメラを回転させ，その際に得られた画像に対して時空間画像処理を施し，画像から水滴を除去する．

2 水滴除去の原理

本手法では，図 2 に示す手順で水滴除去を行う．

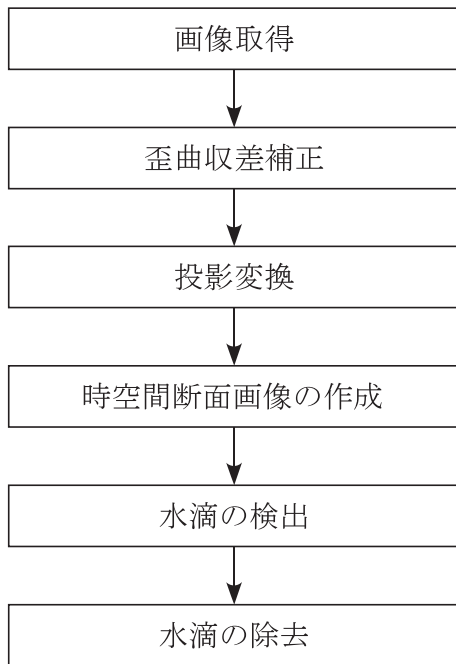


図 2 処理手順

2.1 画像取得

レンズ保護ガラス面上に水滴が付着したカメラを，撮影中等速に回転させ画像を取得する．カメラの回転により，撮影シーンはそれに伴って移動するが，レンズ保護ガラス面上に付着した水滴は画像中で常に同じ位置に撮影され，カメラの回転に伴い水滴で隠れていた被写体が姿を現す（図 3）．

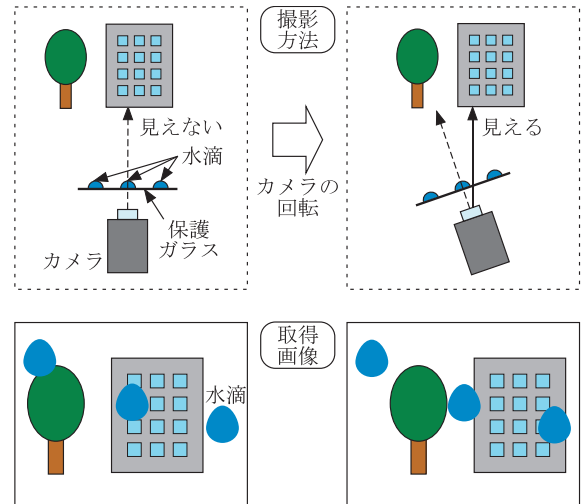


図 3 画像取得

2.2 歪曲収差補正

レンズへの入射角が大きくなると，結像条件が変化することにより結像位置がずれる歪曲収差が生じる．本手法では，2.3 節での位置合わせを正確に行う必要があるため，歪み補正を行う．ここでは，Weng らのレンズの歪みモデル [8] のうち，radial 歪みを考える．歪曲収差補正後の画像面上の座標を $(\tilde{u}, \tilde{v})$ ，補正前の画像面上の座標を (u_0, v_0) ，歪みパラメータを κ_1 とすると，(1)，(2) 式が成り立つ．ただし，(1)，(2) 式で得られる座標は整数値でないため，その座標における画素値を線形補間法により求める．

$$u_0 = \tilde{u} + \kappa_1 \tilde{u} (\tilde{u}^2 + \tilde{v}^2) \quad (1)$$

$$v_0 = \tilde{v} + \kappa_1 \tilde{v} (\tilde{u}^2 + \tilde{v}^2) \quad (2)$$

2.3 投影変換

取得した画像の撮影シーンは，カメラの回転に伴い刻々と変化する．そこで，あるフレームに撮影された画像を基準画像とし，レンズ中心を回転中心とした投影変換を行う．投影変換により，得られた全ての画像を基準画像と同じシーンの画像に位置合わせをすることができる．カメラの回転角を θ ，像距離（レンズ中心から結像面までの距離）を f ，投影変換前の画像中の座標を $(\tilde{u}, \tilde{v})$ とすると，投影変換後の座標 (u, v) は (3)，(4) 式で表される（図 4）．ただし，(1)，(2) 式と同様，(3)，(4) 式で得られる座標は整数値でないため，その座標における画素値を線形補間法により求める．

$$u = f \frac{f \tan \theta + \tilde{u}}{f - \tilde{u} \tan \theta} \quad (3)$$

$$v = f \frac{\sqrt{1 + \tan^2 \theta}}{f - \tilde{u} \tan \theta} \tilde{v} \quad (4)$$

なお，投影変換を行うことにより画像端に画像情報のない領域が生じるが，この領域に対しては 2.5，2.6 節で

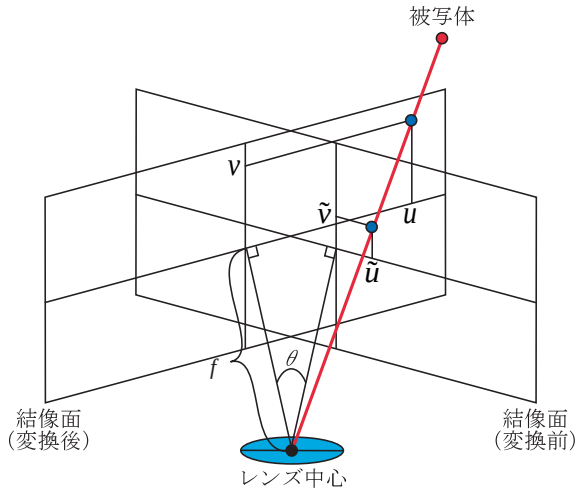


図4 投影変換

述べる処理を行わない。

2.4 時空間断面画像の作成

取得した全ての画像に対して投影変換を行った後，それらを時系列順に並べると，図5に示すような時空間画像 $I(u, v, t)$ として表すことができる．ここで t はフレームである．

時空間画像は，走査線に沿ったある面で断面を切り出すことができる．図6は，図5に示した時空間画像を $v = v_1$ に沿って切り出したときの時空間断面画像である．ここで時空間断面画像を $S(u, t)$ と表すと， $S(u, t) = I(u, v_1, t)$ となる． $S(u, t)$ において，被写体の軌跡は投影変換による位置合わせにより時間軸に対して平行な直線となる．一方，カメラと共に移動する水滴の軌跡は投影変換により(3)，(4)式で表される曲線となる．

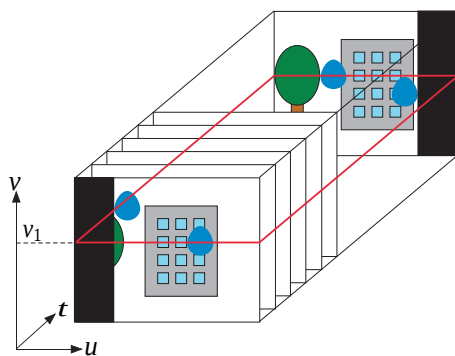


図5 時空間画像 $I(u, v, t)$

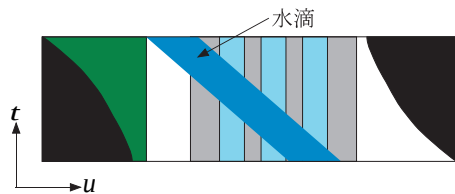


図6 時空間断面画像 $S(u, t)$

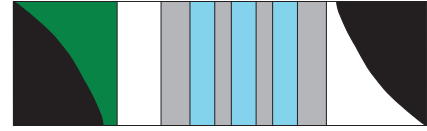


図7 中央値画像 $M(u, t)$

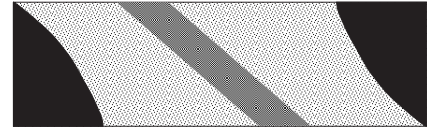


図8 差分画像 $D(u, t)$



図9 2値画像 $H(u, t)$

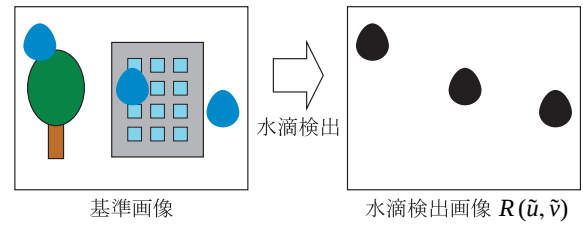


図10 水滴の検出

2.5 水滴の検出

2.5.1 中央値画像の作成

時空間断面画像 $S(u, t)$ 上での被写体の軌跡上の画素値から時間軸方向についての中央値を求める．その後，軌跡上の画素値を求めた中央値に置き換えることで中央値画像 $M(u, t)$ を作成する(図7)．水滴の面積は被写体の面積に比べ小さいため， $M(u, t)$ は水滴が消えた画像となる．

2.5.2 差分画像の作成

時空間断面画像 $S(u, t)$ と中央値画像 $M(u, t)$ の差分を取る．これにより得られた差分画像 $D(u, t)$ は(5)式で表される． $M(u, t)$ は水滴が消えた画像であるため，水滴領域では両者の差が大きくなるのに対し，被写体領域では差が小さくなる(図8)．

$$D(u, t) = |S(u, t) - M(u, t)| \quad (5)$$

2.5.3 水滴の検出

差分画像 $D(u, t)$ に対し，(6)式に示すように閾値 T_b より大きい画素を $H(u, t) = 1$ ，小さい画素を $H(u, t) = 0$ として2値画像 $H(u, t)$ を作成する(図9)．この際，水滴は(4)式で示される投影変換により発生する v 軸方向の変位のため，時間の経過により同一の断面画像上に存在しなくなる．このため $H(u, t)$ を作成する際には，水滴が常に同一の断面画像上に存在するよう，(4)式によ

る変位を考慮している．

$$H(u, t) = \begin{cases} 0, & D(u, t) < T_b \\ 1, & D(u, t) \geq T_b \end{cases} \quad (6)$$

次に，作成した $H(u, t)$ を用いて水滴を検出する．水滴の軌跡は (3) 式で示される曲線となるため，この曲線上を追跡し $H(u, t) = 1$ となる画素数を加算し，その個数が閾値 T_n を超える場所を水滴領域として検出する．なお， $H(u, t)$ を作成する際に v 軸方向の変位を考慮しているため，結果としてこの追跡は 3 次元的に行われる．

全ての時空間断面画像 $S(u, t)$ で水滴領域を検出した後，元の画像の形態に戻した水滴検出画像 $R(\tilde{u}, \tilde{v})$ を作成する (図 10)．

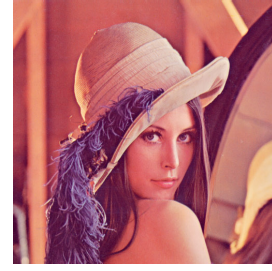
このとき，水滴領域の誤検出により発生したノイズに対し，収縮処理を複数回行うことでこれを除く．また，収縮処理により小さくなった水滴領域を拡げるため膨張処理を複数回行う．

2.6 水滴の除去

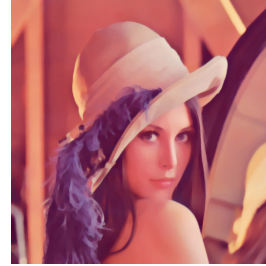
検出した水滴領域に対し，画像修復手法を用いることにより時空間断面画像 $S(u, t)$ から水滴を除去する．本手法で用いる画像修復手法は [9] の文献で提案されているものである．この手法は画像を Structure 画像と Texture 画像に分解し (図 11)，それぞれの画像に対して “Image Inpainting” アルゴリズム，“Texture Synthesis” アルゴリズムを適用して画像を修復するものであり，“Image Inpainting” アルゴリズムが不得意とするテクスチャパターンの再現性が高い．なお，図 11 に示す Texture 画像は，見やすさのため輝度値のコントラストを強くしている．

画像分解は [10] の文献で提案されている手法を用いる． $S(u, t)$ を Structure 画像 $f(u, t)$ と Texture 画像 $g(u, t)$ に分解するには，まず Structure 画像を繰り返し演算により求める． $S(u, t)$ の $n+1$ 回目の繰り返し演算における Structure 画像 $f^{n+1}(u, t)$ は，(7) 式で得られる．

$$f_{u,t}^{n+1} = \left(\frac{1}{1 + \frac{1}{2\lambda h^2} (c_1 + c_2 + c_3 + c_4)} \right) \cdot \left[S_{u,t} + \frac{1}{2\lambda h^2} (c_1 f_{u+1,t}^n + c_2 f_{u-1,t}^n + c_3 f_{u,t+1}^n + c_4 f_{u,t-1}^n) \right] \quad (7)$$



入力画像



Structure 画像



Texture 画像

図 11 画像分解

ここで，

$$c_1 = \frac{1}{\sqrt{\epsilon^2 + \left(\frac{f_{u+1,t}^n - f_{u,t}^n}{h} \right)^2 + \left(\frac{f_{u,t+1}^n - f_{u,t}^n}{2h} \right)^2}},$$

$$c_2 = \frac{1}{\sqrt{\epsilon^2 + \left(\frac{f_{u,t}^n - f_{u-1,t}^n}{h} \right)^2 + \left(\frac{f_{u-1,t+1}^n - f_{u-1,t}^n}{2h} \right)^2}},$$

$$c_3 = \frac{1}{\sqrt{\epsilon^2 + \left(\frac{f_{u+1,t}^n - f_{u,t}^n}{2h} \right)^2 + \left(\frac{f_{u,t+1}^n - f_{u,t}^n}{h} \right)^2}},$$

$$c_4 = \frac{1}{\sqrt{\epsilon^2 + \left(\frac{f_{u+1,t-1}^n - f_{u-1,t-1}^n}{2h} \right)^2 + \left(\frac{f_{u,t}^n - f_{u,t-1}^n}{h} \right)^2}},$$

であり， λ, h, ϵ はそれぞれスケーリング係数，ステップ幅，分母が 0 になることを防ぐための微小量である．ただし， $f_{u,t}^0 = S_{u,t}$ である．

繰り返し演算を N 回行った後，Texture 画像 $g(u, t)$ は (8) 式で得られる．

$$g(u, t) = S(u, t) - f^N(u, t) \quad (8)$$

2.5 節で検出した水滴領域について，得られた $f(u, t)$ と $g(u, t)$ に対し，それぞれ “Image Inpainting” アルゴリズム [1]，“Texture Synthesis” アルゴリズム [11] を適用して水滴を除去する．

全ての $S(u, t)$ から水滴を除去した後，元の画像の形態に戻すことで，水滴が除去された明瞭な画像を得ることができる．

3 実験

雨天時に屋外の風景を撮影した画像を対象に，水滴除去の実験を行った．

撮影フレーム数は 100 フレームであり(毎秒 30 フレーム), 撮影した画像サイズは 360×240 画素である . また , フレーム毎のカメラの回転角は $2.79 \times 10^{-2} \text{rad}$, 像距離は 261 画素相当である . 水滴除去に必要なパラメータについては $T_b = 50$, $T_n = 10$, 収縮・膨張処理回数を 1 , 画像分解に必要なパラメータについては $\lambda = 0.01$, $h = 1$, $\epsilon = 1.0 \times 10^{-6}$, $N = 100$ と設定した . なお , 撮影は市販の首振りカメラ (Sony EVI-D100) を用いて行った .

水滴の付着した原画像 (60 フレーム目) に対して歪曲収差補正を施した画像を図 12 に示す . この画像に対し , レンズ中心を回転中心とした投影変換を行った結果を図 13 に示す . 投影変換の際の基準画像は 50 フレーム目の画像である .

例として , 図 13 の中央付近の走査線 ($v = 150$) での時空間断面画像 $S(u, t)$ を図 14 に示す . 図中に示した領域は動物体である . 水滴検出までの過程として , このラインおける $S(u, t)$ をグレースケール化した画像を図 15 , 中央値画像 $M(u, t)$ を図 16 , 差分画像 $D(u, t)$ を図 17 , 投影変換を考慮した 2 値画像 $H(u, t)$ を図 18 , 水滴検出画像 $R(\tilde{u}, \tilde{v})$ を図 19 にそれぞれ示す . 次に , 水滴除去までの過程として , Structure 画像 $f(u, t)$ に対して “Image Inpainting” アルゴリズムを適用した結果を図 20 , Texture 画像 $g(u, t)$ に対して “Texture Synthesis” アルゴリズムを適用した結果を図 21 , 水滴除去後の時空間断面画像を図 22 にそれぞれ示す . なお , Texture 画像に対しては , 2.6 節と同様の理由により輝度値のコントラストを強くして表示している . 最後に , 本手法を用いて水滴の除去を行った結果を図 23 に示す .



図 12 取得画像

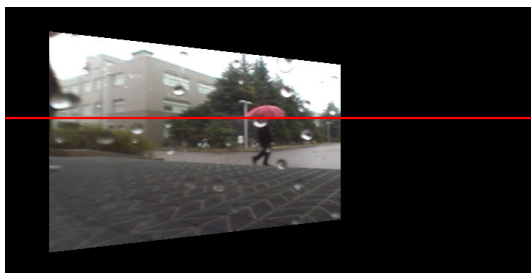


図 13 取得画像 (投影変換後)

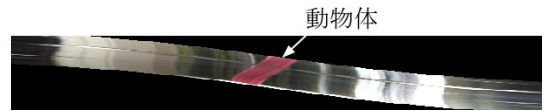


図 14 時空間断面画像 $S(u, t)$

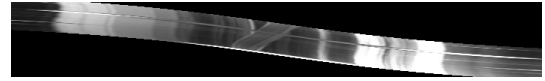


図 15 時空間断面画像 (グレースケール)



図 16 中央値画像 $M(u, t)$

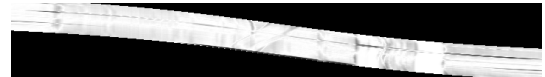


図 17 差分画像 $D(u, t)$



図 18 投影変換を考慮した 2 値画像 $H(u, t)$

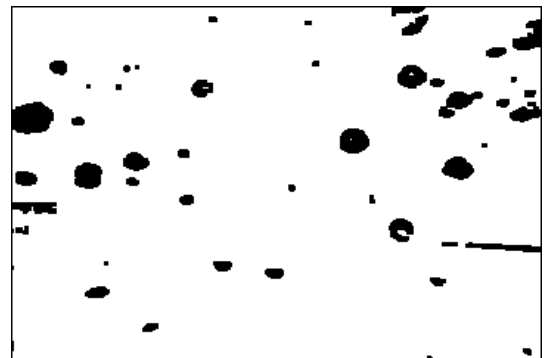


図 19 水滴検出画像 $R(\tilde{u}, \tilde{v})$

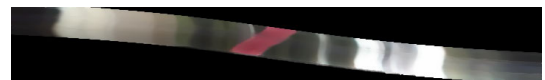


図 20 Structure 画像

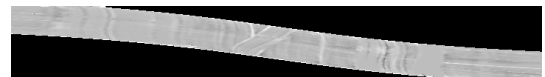


図 21 Texture 画像

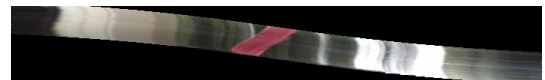


図 22 水滴除去後の時空間断面画像



図 23 除去画像

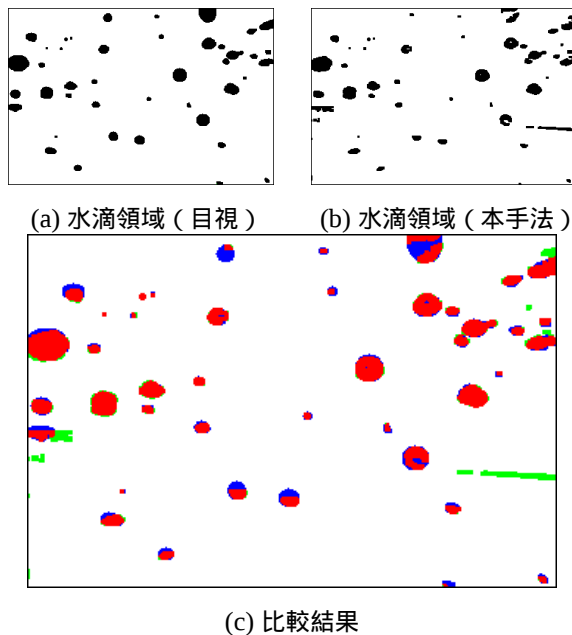


図 24 水滴検出精度

表 1 水滴検出精度

検出できた水滴の割合（％）	79.4
検出できなかった水滴の割合（％）	20.6
誤検出した水滴（％）	0.8

今回の実験では、一部の雨が画像中に写り込んだが、水滴の検出・除去に与える影響は少なく、良好な結果を得ることができた。

次に、本手法を用いた水滴の検出精度について、目視により判定した水滴領域との比較を行うことで検証した結果を図 24、表 1 に示す。図 24(c) において、赤で示した領域は検出できた領域、青で示した領域は検出できなかった領域、緑で示した領域は誤検出した領域を表す。また、表 1 において、検出できた水滴の割合とは、目視により判定した水滴面積を 100% としたとき、本手法を用いて検出することができた水滴面積の割合である。同様に、検出できなかった水滴の割合とは、目視により判定した水滴面積を 100% としたとき、本手法を用いて検出することができなかった水滴面積の割合である。誤検出した水滴の割合とは、画像サイズを 100% としたとき、目視により判定した水滴領域以外を水滴として検出した面積の割合である。

検出できなかった 20% は、主に水滴の中心付近における領域であり、この領域に関しては、水滴のレンズ効果により背後にある被写体の色が透けて写ることで、差分画像において被写体領域と水滴領域の間に大きな差が生まれず検出が困難となっている。また、誤検出に関しては、主に画像の下半分に多く見られ、カメラと距離が近い地面が多く写ることで、レンズ中心と回転中心のズレ



取得画像

除去画像

図 25 水滴除去結果

により引き起こる投影変換の誤差が大きく影響したと考えられる。しかし、実際には水滴の中心付近における領域は目視ではほとんど水滴と認識することができない程度であり、誤検出に関しても画像全体の 0.8% 程度と少なく、良好な結果を得ることができている。さらに、撮影シーンに動物体が存在する条件下でも、時空間断面画像上で画像修復手法を用いることでその軌跡を復元し、図 25 に示す明瞭な画像を得ることができた。

4 結論

雨天時に撮影したカメラ映像に対して時空間画像処理を用いることにより、レンズ保護ガラス面上で静止した水滴を画像から除去する手法について検討した。具体的には、レンズ保護ガラスと一体となって回転するカメラで撮影し得られた時空間断面画像上で、水滴と被写体との間における軌跡の違いに着目することにより水滴を検出し、その領域に対して画像修復手法を用いることで画像からの水滴除去を行った。

今後の課題としては、撮影中にレンズ保護ガラス面上の水滴が動く際の対応や画像修復手法の改良などが挙げられる。

謝辞

本研究の一部は、文部科学省大都市大震災軽減化特別プロジェクトの補助を受けた。

参考文献

- [1] M.Bertalmio, G.Sapiro, V.Caselles and C.Ballester: "Image Inpainting", Proceedings of SIG-GRAPH2000, pp.417-424, 2000.
- [2] 木幡恭久, 剣持雪子, 小谷一孔: "画像のフラクタル性を活用した局所的な画像推定法による画像復元手法", 電子情報通信学会技術報告 (PRMU2002-126), Vol.102, No.471, pp.37-42, 2002.
- [3] 三宅一永, 米田政明, 長谷博行, 酒井充, 丸山博: "時間メディアンフィルタによる降雪ノイズ除去", 画像電子学会誌, Vol30, No.3, pp.251-259, 2001.
- [4] K.Garg and S.K.Nayar: "Detection and Removal of Rain from Videos", Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.1, pp.528-535, 2004.
- [5] Y.Tanaka, A.Yamashita, T.Kaneko and K.T.Miura: "Removal of Adherent Waterdrops from Images Acquired with a Stereo Camera System", IEICE Transactions on Information and Systems, Vol.89-D, No.7, pp.2021-2027, 2006.
- [6] A.Yamashita, T.Harada, T.Kaneko and K.T.Miura: "Virtual Wiper -Removal of Adherent Noises from Images of Dynamic Scenes by Using a Pan-Tilt Camera-", Advanced Robotics, Vol.19, No.3, pp.295-310, 2005.
- [7] 羽下哲司, 鷲見和彦, 橋本学, 関明伸, 黒田伸一: "時空間画像処理による注目距離対象の強調に基づいた遠隔監視システム", 電子情報通信学会技術報告 (PRMU97-126), pp.41-46, 1997.
- [8] J.Weng, P.Cohen and M.Herniou: "Camera Calibration with Distortion Models and Accuracy Evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.14, No.10, pp.965-980, 1992.
- [9] M.Bertalmio, L.Vesa, G.Sapiro and S.Osher: "Simultaneous Structure and Texture Image Inpainting", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.2, pp.707-712, 2003.
- [10] L.I.Rudin, S.Osher and E.Fatemi, "Nonlinear total variation based noise removal algorithms", Physica D, Vol.60, pp.259-268, 1992.
- [11] A.A.Efros and T.K.Leung: "Texture Synthesis by Non-parametric Sampling", IEEE International Conference on Computer Vision, Vol.2, pp.1033-1038, 1999.