

時空間画像処理による雨天時画像からの視野妨害ノイズ除去

Removing Adherent Noise from Image Sequences in Rainy Weather by Spatio-Temporal Image Processing

福地 功[†], 正会員 山下 淳^{†,††}, 正会員 金子 透[†], 三浦 憲二郎^{†††}

Isao Fukuchi[†], Atsushi Yamashita^{†,††}, Toru Kaneko[†] and Kenjiro T. Miura^{†††}

Abstract We have developed a method that can remove view-disturbing noise from image sequences by spatio-temporal image processing. In outdoor environments on rainy days, pictures taken by a camera are often degraded because of adherent noise, such as water drops on the surface of the lens-protecting glass of the camera. To solve this problem, our method uses image sequences captured with a pan-tilt camera. The method uses a spatio-temporal image to extract adherent noise regions by examining trajectory differences between adherent noise and other objects in cross-section images. Finally, adherent noise regions are filled in with surrounding information in the cross-section image. Experimental results showed our method is effective.

キーワード：時空間画像処理，ノイズ検出，ノイズ除去，首振りカメラ，画像修復

1. ま え が き

近年，セキュリティニーズの高まりから，屋外監視カメラを用いた侵入者や進入物の自動検出に関する研究・開発が盛んに行われている．その他にも，カメラに代表される視覚センサを搭載した自律移動型ロボットによる周囲環境のセンシングや，交通状況の確認など様々な用途において，屋外でカメラ撮影を行う機会が多い．しかし，屋外環境下で撮影された画像は，時間の経過や晴天・曇天といった照明条件の変化に加え，雨や雪などの悪天候による影響を強く受ける．特に雨天時における撮影では，カメラレンズと被写体との間に介在する保護ガラス面上に水滴や泥など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．

このようなガラス面上に付着した妨害物を取り除く方法

としては，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のは困難なことが多く，さらにワイパーを装着できたとしても，ワイパーの動作時にそれ自体が視野を妨害するという問題もある．

上述の物理的に妨害物を取り除くアプローチに対し，ソフトウェア的に画像処理の技術を用いることで，悪天候時に撮影した画像からノイズを取り除く手法も数多く提案されている．この中には，時間メディアンフィルタを用いて降雪ノイズを除去する手法¹⁾や，雨の物理モデルを用いて降雨ノイズを除去する手法²⁾などもある．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，これらが保護ガラス面上に付着，静止することで発生するノイズを除去することができない．

次に，古い映像フィルム中の引っかき傷により発生したノイズを検出・除去する手法³⁾⁴⁾を，保護ガラス面上に妨害物が付着することにより発生したノイズに適用する場合について考える．引っかき傷により発生したノイズを検出するために，文献³⁾では近接するフレーム間で差分を取る．これにより，フィルム中の引っかき傷といったランダムに出現するノイズを検出することは可能であるが，文献¹⁾²⁾と同様，保護ガラス面上に妨害物が付着，静止することにより，画像上で常に同じ場所に撮影されるノイズを検出，除去することができない．

以上に述べた手法は，主に視野内を移動するノイズ除去に関するものである．これらに対し，保護ガラス面上の静止した視野妨害ノイズを除去する手法も提案されている⁵⁾⁶⁾．ステレオカメラを用いた手法⁵⁾では，取得した2枚の左右

2007年7月26日受付，2007年11月28日再受付，2008年2月12日採録

[†] 静岡大学 工学部 機械工学科

(〒432-8561 静岡県浜松市中区城北 3-5-1, TEL 053-478-1067)

^{††} カリフォルニア工科大学 機械工学科

(〒91125 米国カリフォルニア州パサデナ市)

^{†††} 静岡大学 創造科学技術大学院

(〒432-8561 静岡県浜松市中区城北 3-5-1, TEL 053-478-1074)

[†] Department of Mechanical Engineering, Shizuoka University

(3-5-1 Johoku, Naka-ku, Hamamatsu-shi, Shizuoka 432-8561, Japan)

^{††} Department of Mechanical Engineering, California Institute of Technology

(1200 E. California Blvd. MC104-44, Pasadena, CA 91125, USA)

^{†††} Graduate School of Science and Technology, Shizuoka University

(3-5-1 Johoku, Naka-ku, Hamamatsu-shi, Shizuoka 432-8561, Japan)

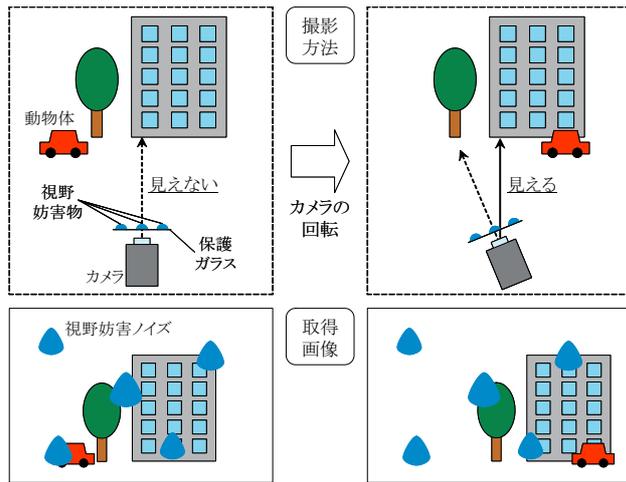


図 1 撮影方法
Image acquisition.

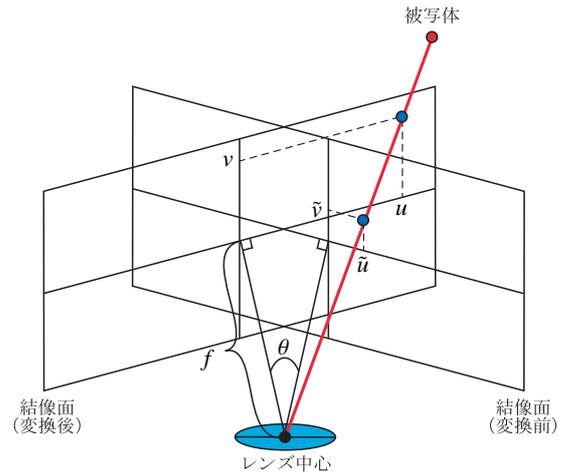


図 2 投影変換
Projective transformation.

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視野妨害ノイズを検出する。しかし、2枚の画像で視野妨害ノイズの見え方に差が生じたときに正しく対応点を得ることが難しく、結果としてノイズを検出、除去することができない場合がある。また、カメラを回転させ取得した2枚の画像から視野妨害ノイズを検出・除去する手法⁶⁾も同様に、2枚の画像だけでは同じ場所がノイズで隠れた際、ノイズを除去することができない。

そこで本研究では、1台のカメラから連続的に取得した複数の静止画像に対して、時空間画像処理を用いることにより視野妨害ノイズの除去を行い、視野の明瞭化を図ることを目的とする。使用するカメラとしては、サーベイランス用途でよく使われる首振りカメラを用いる。同じく時空間画像処理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、保護ガラス面上の妨害物と被写体とを区別することが困難である。その点、本手法では首振りカメラを回転させ画像を取得するため、被写体に遠景対象物が含まれていても問題なく視野妨害ノイズを除去することが可能である。

2. 視野妨害ノイズ除去の原理

2.1 画像取得

保護ガラス面上に視野妨害物が付着したカメラを、レンズ中心を中心として、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、保護ガラス面上に付着した妨害物は画像上で常に同じ場所に撮影されるため、カメラの回転に伴い妨害物で隠れていた被写体が姿を現す(図1)。

2.2 位置合わせ

取得したすべての画像に対して歪み補正を行った後、投影変換を用いてカメラ回転後に得られた画像を変換することで、撮影シーンの位置合わせを行う。

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、次に述べる投影変換による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。歪み補正後の画像中の座標を (\tilde{u}, \tilde{v}) 、補正前の座標を (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。

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

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

次に、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、得られたすべ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。像距離(レンズ中心から結像面までの距離)を f 、1 フレームあたりのカメラの回転角度(反時計回りを正の方向)を $\Delta\theta$ 、フレーム番号を t 、基準画像におけるフレーム番号を t_{base} 、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2)。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。

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

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

なお、投影変換を行うことで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 2.4、2.5 節で述べる処理は行わない。

2.3 時空間断面画像の作成

取得したすべ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3に示すような時空間画像

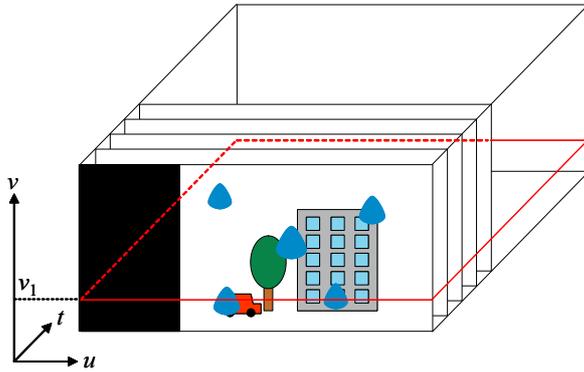


図 3 時空間画像 $I(u, v, t)$
Spatio-temporal image.

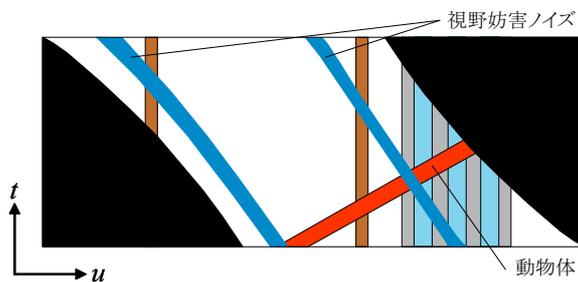


図 4 時空間断面画像 $S(u, t)$
Cross section image.

$I(u, v, t)$ として表すことができる。

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 4 は、図 3 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視野妨害物の軌跡は投影変換により (3), (4) 式で表される曲線となる。また、撮影シーンに動物体が存在する場合、その軌跡は動物体特有のものとなる。

2.4 視野妨害ノイズの検出

(1) 中央値画像の作成

時空間断面画像 $S(u, t)$ 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 5)。視野妨害ノイズの面積は被写体の面積に比べ小さいため、 $M(u, t)$ は視野妨害ノイズが消えた画像となる。しかし、図 4 の場合、動物体の軌跡も視野妨害ノイズと同様に $M(u, t)$ から姿を消す。このため、以降で述べる方法で視野妨害ノイズのみを検出し、検出したノイズ領域に対してのみ除去処理を行う必要がある。

なお、ここで平均値ではなく中央値を用いるのは、(2) 項における差分処理において、被写体領域と視野妨害ノイズ領域との間に明確な違いを生み出すためである。平均値を用いた場合、視野妨害ノイズの画素値もそれに含まれるため、中央値と比べて差分処理結果に違いが生じにくい。

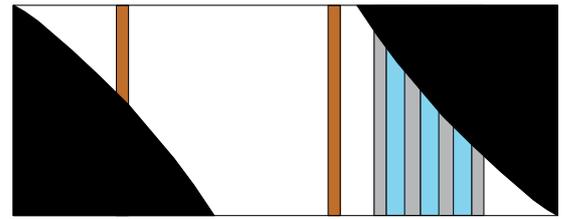


図 5 中央値画像 $M(u, t)$
Median image.

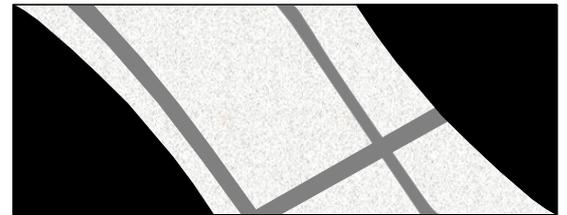


図 6 差分画像 $D(u, t)$
Difference image.

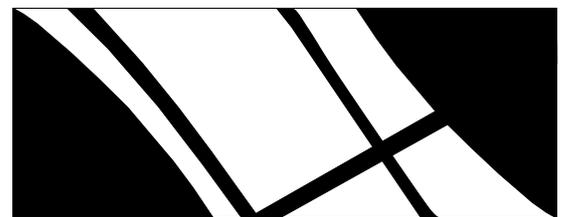


図 7 2 値画像 $B(u, t)$
Binary image.

(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。これにより得られる差分画像 $D(u, t)$ は、(5) 式で表される。 $M(u, t)$ は視野妨害ノイズ・動物体が消えた画像であるため、これらの領域では $S(u, t)$ と $M(u, t)$ と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 6)。

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

(3) 視野妨害ノイズの検出

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 $B(u, t) = 1$ 、小さい画素を $B(u, t) = 0$ として、2 値画像 $B(u, t)$ を作成する (図 7)。

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

次に、作成した $B(u, t)$ を用いて視野妨害ノイズを検出する。視野妨害物の軌跡は (3), (4) 式で表される曲線となるため、この曲線上の $B(u, t) = 1$ となる画素数を加算し、その個数が閾値 T_n より大きい場所を視野妨害ノイズとして検出する。(3), (4) 式の曲線上のみで加算を行うため、図 7 で示される動物体領域中の $B(u, t) = 1$ となる画素による視野妨害ノイズの誤検出を減らすことができる。また、視野妨害ノイズは、撮影シーンの位置合わせのために行った

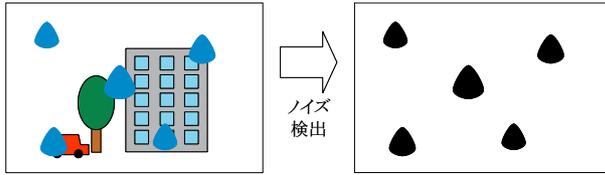


図 8 視野妨害ノイズの検出
Detection of adherent noise.

投影変換により、 v 軸方向に (4) 式で示される変位が発生するため、時間の経過に伴い同一の断面画像上に存在しなくなる。このため、視野妨害ノイズを検出するときには、視野妨害物の軌跡上の画素についてのみ着目する必要があるため、(4) 式による v 軸方向の変位を考慮し、複数の断面画像に渡り $B(u, t) = 1$ となる画素数を加算する。

すべての時空間断面画像 $S(u, t)$ で視野妨害ノイズを検出した後、元の画像の形態に戻したノイズ検出画像を作成する (図 8)。このとき、視野妨害ノイズの誤検出により発生したノイズを除去するために、ノイズ検出画像に対して収縮、膨張処理を順に行う。

2.5 視野妨害ノイズの除去

検出した視野妨害ノイズ領域に対し、画像修復処理を行うことにより、時空間断面画像 $S(u, t)$ から視野妨害ノイズを除去する。

画像修復処理の代表的な手法としては、“Inpainting” アルゴリズムが挙げられる。この手法は、輝度値をノイズ領域の境界から内側へと伝播させることにより、写真に付いた引っかき傷のような細かいノイズに対しては良好な結果を得ることができる。しかし、水滴など比較的大きな視野妨害ノイズの場合、細かいテクスチャパターンを再現することができず、不鮮明な結果になるという問題がある。このため、本手法では、画像修復手法として文献 9) で提案されているものを用いる。この手法は、画像を Structure 画像と Texture 画像に分解し、それぞれの画像に対して “Inpainting” アルゴリズム、“Texture Synthesis” アルゴリズムを適用することにより、画像情報の欠落した領域 (視野妨害ノイズ領域) を補間する。ここで、Structure 画像とは、画像から大まかな構造を抽出したものであり、Texture 画像とは、その他の細かいテクスチャや微小なノイズを含んだものである (図 9)。画像を分解した後、それぞれの画像に対して別々のアルゴリズムを適用することで、“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といった利点を得られる。なお、図 9(c) 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高くしている。

画像分解で使用する手法としては、文献 10) で提案されているものを用いる。Structure 画像を $f(u, t)$ 、Texture 画像を $g(u, t)$ と置くと、 $S(u, t)$ との関係は $S(u, t) = f(u, t) + g(u, t)$ で表される。 $f(u, t)$ は (7) 式に示す繰返し演算を N 回行うことにより求めることができるため、



(a) 原画像 (Barbara) (b) Structure 画像 (c) Texture 画像

図 9 画像分解例
Example of image decomposition.

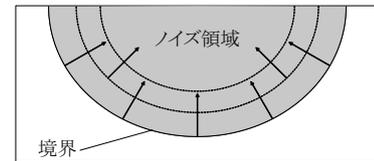


図 10 “Inpainting” アルゴリズム
Inpainting algorithm.

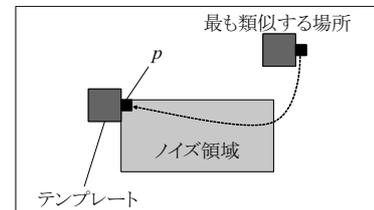


図 11 “Texture Synthesis” アルゴリズム
Texture synthesis algorithm.

これにより $g(u, t)$ を得る。

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

ここで、

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

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

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

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

であり、 λ 、 h 、 ε はそれぞれスケール係数、ステップ幅、分母が 0 になることを防ぐための微量である。ただし、 $f_{u,t}^0 = S_{u,t}$ である。

次に、得られた $f(u, t)$ 、 $g(u, t)$ に対して、“Inpainting” アルゴリズム、“Texture Synthesis” アルゴリズムを適用し、それぞれの画像で視野妨害ノイズを除去する。 $f(u, t)$



図 12 ノイズ除去例
Example of noise removal.

に対する “Inpainting” アルゴリズムでは、文献 11) で提案されている手法を用いる．具体的な処理としては、偏微分方程式を利用し、視野妨害ノイズ領域の境界からその内側へと輝度値を伝播させることでノイズの除去を行う (図 10)． $g(u, t)$ に対する “Texture Synthesis” アルゴリズムでは、文献 12) で提案されている手法を用いる．具体的な処理としては、視野妨害ノイズ領域に属する画素 p に隣接する矩形領域 (テンプレート) との類似度が最も高い領域を画像から探索し、その領域に隣接する画素に p を置き換えることでノイズの除去を行う (図 11)．

最後に、視野妨害ノイズを除去した $f(u, t)$, $g(u, t)$ を合成することで、ノイズの消えた $S(u, t)$ を得ることができる．なお、元の文献 9) では、ノイズ領域を人間が逐次指定する必要があるが、本手法では、2.4 節で述べた方法で視野妨害ノイズ領域を検出済みであるため、その必要はない．

1 フレーム毎に 1 枚ずつ画像修復を行う場合と比べ、 $S(u, t)$ に対して画像修復を行う場合は、近接するフレームの画像情報を利用できる点で優れている．画像修復手法は一般的に、画像情報の欠落した領域を、その周囲の輝度勾配などの情報を用いて補間していく．よって、あるフレームでは視野妨害物により全く見えていない被写体・動物体を、そのフレームにおける 1 枚の画像のみを用いて補間することは原理的に不可能である．これに対し、 $S(u, t)$ 上において画像修復を行う場合、被写体・動物体の軌跡を用いた補間が可能となり、よりロバストに視野妨害ノイズを除去することができる．図 12 は、図 12(a) に示すノイズ領域に対し、文献 11) の手法を用いて 1 フレームの画像に対して画像修復を行った結果 (図 12(b)) と、 $S(u, t)$ に対して画像修復を行った結果 (図 12(c)) について示した例である．図 12(b) では、1 枚の画像のみから補間を行うため、ノイズで隠れた被写体の部分を再現することができず、不自然な結果となっている．一方、図 12(c) では、 $S(u, t)$ 上において軌跡を用いた補間を行うため、ノイズで隠れた被写体の部分をより正確に再現できている．

すべての $S(u, t)$ から視野妨害ノイズを除去した後、元の画像の形態に戻すことで、視野妨害ノイズのない明瞭な画像を得ることができる．

3. 実験

画像を撮影する前に、格子パターンを用いてカメラの

キャリブレーションを行った．その結果、画像サイズが 360×240 pixels において、歪みパラメータ $\kappa_1 = -2.14 \times 10^{-5}$ 、像距離 $f = 261$ pixel 相当であった．

まず最初に、保護ガラス面上の視野妨害物が水滴の場合についての結果を示す．撮影フレーム数は 100 フレームであり (毎秒 30 フレーム)、1 フレームあたりのカメラの回転角度は $\Delta\theta = 2.79 \times 10^{-2}$ rad である．水滴の検出に必要なパラメータについては $T_b = 50$, $T_n = 10$ と設定した．また、収縮・膨張処理回数は 1、画像分解に必要なパラメータについては $\lambda = 0.01$, $h = 1$, $\varepsilon = 1.0 \times 10^{-6}$, $N = 100$ と設定した．

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 13 に示す．この画像に対し、 $t_{base} = 50$ として投影変換を行った結果を図 14 に示す．例として、図 14 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 15 に示す．水滴検出までの過程として、この走査線上における中央値画像 $M(u, t)$ を図 16、差分画像 $D(u, t)$ を図 17、2 値画像 $B(u, t)$ を図 18、水滴検出画像を図 19 にそれぞれ示す．次に、水滴除去までの過程として、Structure 画像に対して “Inpainting” アルゴリズムを適用した結果を図 20、Texture 画像に対して “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 画像に対しては見やすさのため、輝度値のコントラストを高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．

本手法を用いた水滴の検出精度について調べるため、目視により判定した水滴領域と図 19 との比較を行った．その結果、水滴全体の約 80% を検出できていることが確認できた．それ以外の検出できなかった領域は、主に水滴の中心付近における領域である．この領域に関しては、水滴をレンズとみなした場合の光学中心の背後にある被写体が透けて写ること、差分処理において被写体と水滴との間に大きな差が生じず検出が困難となっている．しかし、この領域は本来の背景が写っているため、水滴として除去する必要がなく、結果として明瞭な画像を得ることができている．誤検出 (水滴以外を水滴として検出) に関しては、画像の下半分の領域に若干だが確認された．これはカメラとの距離が近い地面が多く写ることにより、レンズ中心と回転中心のズレによって引き起こる投影変換の誤差が影響したと考えられる．しかし、誤検出に関しても画像全体の約 0.8% と少なく、良好な結果を得ることができている．

今回の実験では、撮影シーンに動物体が存在する画像から視野妨害ノイズの除去を行った (図 23)．動物体が視野妨害物により隠れた場合、2.5 節で述べた通り、そのフレームにおける 1 枚の画像のみでは、動物体の正確な情報を知る事はできない．しかし、時空間断面画像上で画像修復を行うことにより、視野妨害ノイズで隠れていた動物体をより正確に再現することが可能となっている．ただし、画像



図 13 投影変換前の画像
Image before the projective transformation.

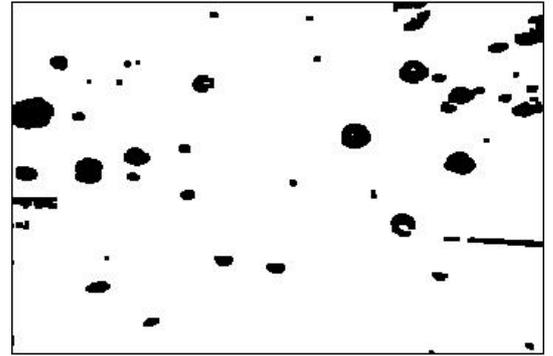


図 19 水滴検出画像
Region of waterdrop.



図 14 投影変換後の画像
Image after the projective transformation.



図 20 水滴除去後の Structure 画像
Structure image after removing waterdrop.



図 21 水滴除去後の Texture 画像
Texture image after removing waterdrop.



図 15 時空間断面画像 $S(u, t)$
Cross section image.



図 22 水滴除去後の時空間断面画像
Cross section image after removing waterdrop.



図 16 中央値画像 $M(u, t)$
Median image.



図 17 差分画像 $D(u, t)$
Difference image.



図 18 2 値画像 $B(u, t)$
Binary image.



(a) 取得画像 (b) 除去画像

図 23 ノイズ除去結果 (水滴)
Results of waterdrop removal.

上で動物体が視野妨害ノイズと同様の動きをした場合、動物体を視野妨害ノイズとして検出、除去してしまう問題が発生すると考えられる。

なお、本手法は多数の時空間断面画像に対して画像修復処理を行うため、実時間処理は難しい。そのため、本手法の処理対象としては、屋外監視カメラなどで撮影、保管された画像データを考えている。

最後に、他の条件下で撮影した画像に対して、本手法を用いて視野妨害ノイズの除去を行った結果を図 24、図 25

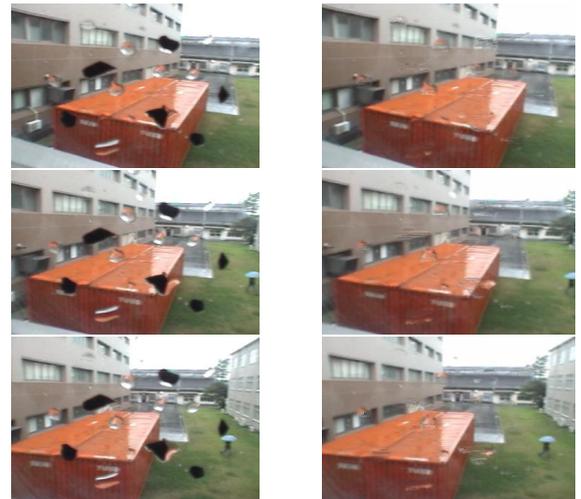
に示す。視野妨害物が泥の場合、水滴のようなレンズ効果は起こり得ないので、より高い精度でノイズ領域を検出することが可能となっている。



(a) 取得画像

(b) 除去画像

図 24 ノイズ除去結果 (泥)
Results of mud blob removal.



(a) 取得画像

(b) 除去画像

図 25 ノイズ除去結果 (水滴 + 泥)
Results of waterdrop and mud blob removal.

4. む す び

本論文では、カメラを回転させて取得した画像に対して時空間画像処理を行うことにより、画像中から視野を妨害するノイズを除去する手法について提案した。具体的には、保護ガラスと一体となって回転する首振りカメラを用いて撮影を行い、得られた時空間断面画像上での軌跡の違いから視野妨害ノイズを検出し、その領域に対して画像修復手法を利用した。

今後の課題としては、各種パラメータの自動設定や、撮影中に保護ガラス面上の視野妨害物が動く際の対応が挙げられる。また、現段階ではそれぞれ独立した1枚の時空間断面画像のみを用いて画像修復を行っているが、上下に隣接する断面画像の情報をここに加えることで、より良い結果を得ることができると考えられる。

なお本研究の一部は、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。

(文 献)

- 1) 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”, 画電学誌, **30**, 3, pp.251-259 (2001)
- 2) K. Garg and S. K. Nayar: “Detection and Removal of Rain from Videos”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, **1**, pp.528-535 (2004)
- 3) A. C. Kokaram, R. D. Morris, W. J. Fitzgerald and P. J. W. Rayner: “Detection of Missing Data in Image Sequences”, IEEE Transactions on Image Processing, **4**, 11, pp.1496-1508 (1995)
- 4) A. C. Kokaram, R. D. Morris, W. J. Fitzgerald and P. J. W. Rayner: “Interpolation of Missing Data in Image Sequences”, IEEE Transactions on Image Processing, **4**, 11, pp.1509-1519 (1995)
- 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, **89-D**, 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, **19**, 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”, 信学技報

(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, **14**, 10, pp.965-980 (1992)
- 9) M. Bertalmio, L. Vesa, G. Sapiro and S. Osher: “Simultaneous Structure and Texture Image Inpainting”, IEEE Transactions on Image Processing, **12**, 8, pp.882-889 (2003)
- 10) L. I. Rudin, S. Osher and E. Fatemi, “Nonlinear Total Variation Based Noise Removal Algorithms”, Physica D, **60**, pp.259-268 (1992)
- 11) M. Bertalmio, G. Sapiro, V. Caselles and C. Ballester: “Image Inpainting”, Computer Graphics (SIGGRAPH 2000), pp.417-424 (2000)
- 12) A. A. Efros and T. K. Leung: “Texture Synthesis by Non-parametric Sampling”, IEEE International Conference on Computer Vision, **2**, pp.1033-1038 (1999)



ふくち いさお
福地 功 2008年, 静岡大学大学院工学研究科機械工学専攻修士課程修了。同年, プラザー工業株式会社入社。在学中は, 視野遮蔽画像の明瞭化に関する研究に従事。



やました あつし
山下 淳 2001年, 東京大学大学院工学系研究科精密機械工学専攻博士課程修了。同年, 静岡大学工学部機械工学科助手。2006年, カリフォルニア工科大学機械工学科客員研究員(併任)。2007年, 静岡大学工学部機械工学科助教。2008年, 静岡大学工学部機械工学科准教授。画像処理, コンピュータビジョン, ロボットの知能化に関する研究に従事。博士(工学)。正会員。



かねこ とおる
金子 透 1974年, 東京大学大学院工学系研究科物理工学専攻修士課程修了。同年, 日本電信電話公社(現 NTT)入社。1997年, 静岡大学工学部教授。画像処理, ロボットビジョンの研究に従事。工学博士。正会員。



みうらけんじろう
三浦憲二郎 1984年, 東京大学大学院工学系研究科精密機械工学専攻修士課程修了。同年, キヤノン(株)入社。1991年, コーネル大学機械工学科博士課程修了。1993年, 会津大学コンピュータ理工学部コンピュータソフトウェア学科助教授。1997年, 静岡大学工学部助教授。2004年, 同教授。2006年, 同大学創造科学技術大学院教授。形状処理工学, CAD/CAM, 物理シミュレーションの研究に従事。Ph.D.