

ジョイントフィルタによる
高効率な奥行き推定に関する研究

A Study on Efficient Depth Map Estimation
by Using Joint Filtering

2016 年

松尾 琢也

目次

第 1 章	序論	1
1.1	はじめに	1
1.2	奥行き推定方式	2
1.3	ステレオマッチング	4
1.4	ジョイントフィルタ	15
1.5	本論文の位置づけと構成	17
第 2 章	ジョイントフィルタによるデプスマップの精度改善	19
2.1	まえがき	19
2.2	従来手法	20
2.3	信頼度を用いたジョイントトリラテラルフィルタ	25
2.4	重み付きジョイントバイラテラルフィルタとジョイントニアレストフィ ルタ	36
2.5	重み付きジョイントバイラテラルフィルタによるコストボリュームフィ ルタリング	55
2.6	おわりに	64
第 3 章	ジョイントフィルタを用いたインループフィードバックマッチング	65
3.1	まえがき	65
3.2	提案手法	66
3.3	評価実験	71
3.4	おわりに	73
第 4 章	ジョイントフィルタによるデプスマップ超解像	75
4.1	まえがき	75
4.2	従来手法	76
4.3	提案手法	78
4.4	評価実験	80

4.5	おわりに	83
第 5 章	結論	85
謝辞		87
参考文献		89

第 1 章

序論

1.1 はじめに

近年，映像を扱う様々なアプリケーションは 2 次元から 3 次元に移り変わりつつある．映像が 3 次元になると，これまでの映像情報に加えて奥行き情報が付加される．例えば，奥行き情報を用いるアプリケーションとしては，立体映像が鑑賞できる 3DTV [1]，ユーザが任意に視点を変更できる自由視点映像 [2]，カメラ撮像後に任意にフォーカスを変更できるデジタルリフォーカス [3]，建物など形や色が一定でないものに映像を投影するプロジェクションマッピング [4]，自動車などに搭載され安全運転を支援する障害物検知，コンピュータを直感的に操作するナチュラルユーザーインターフェースなどがある．3DTV では，1 視点の画像と奥行き情報から 2 視点目や多視点の映像を生成し，立体に表示する技術に奥行き情報が活用される．自由視点映像では，映像表示の際に奥行き情報を用いて任意視点の映像が合成され，また，デジタルリフォーカスでは，撮像時の各対象物までの奥行き情報とカメラ固有のぼけ情報から任意焦点画像が生成される．プロジェクションマッピングでは，投影対象物までの距離や形状把握のために奥行き情報が利用される．障害物検知では，対象物の大きさや距離推定のために奥行き情報が活用される．ナチュラルユーザーインターフェースでは，人物検出や動き推定のために奥行き情報が利用される．これら以外にも，様々なアプリケーションで奥行き情報が活用されている．

多くのアプリケーションでは奥行き情報の精度が，アルゴリズム自体の性能に直結するため，奥行き情報を正しく推定することは非常に重要な課題である．例えば，3DTV や自由視点映像といった表示技術に用いる場合では，デプスマップの精度は映像品質に影響を及ぼし，障害物検知やナチュラルユーザーインターフェースなどの認識技術に用いる場合では，認識性能に影響を及ぼす．加えて，多くのアプリケーションではリアルタイム処理を要求しており，奥行き情報取得までの処理時間も重要となる．しかしながら，これまで研究されてきた手法は，高精度な奥行き情報を取得することはできても，取得時間が長いものや，逆に取得は高速に行えるが精度が低い手法が多い．

そこで本論文では，奥行き情報を高速かつ高精度に取得することを目的とした奥行き推定手法について，複数のアプローチから挑み，このトレードオフの問題を解決する．

1.2 奥行き推定方式

奥行き推定の手法としては、推定装置からパターンや光を照射し、パターンを読み取ったり、反射光を計測したりすることで奥行き推定を行うアクティブ方式と、撮影された画像の特徴点を対応付けることで奥行き推定を行うパッシブ方式がある。

アクティブ方式には、コンシューマ向けにも販売されている Microsoft 社の Kinect v1 や, Asus 社の Xtion (Kinect v1 と同等), Intel 社の RealSense などを用いられるパターン照射方式や, Microsoft 社の Kinect v2, MESA 社の SwissRanger, PMD Technologies GmbH 社の CamCube, そのほか測量などで用いられるレーザーレンジファインダなどの Time of Flight 方式 (ToF 方式) がある [5]。図 1.1, 1.2 はパターン照射方式の図示したものである。パターン照射方式では、撮影したいシーンに対して赤外線などで線や円といったパターンを照射する。このとき、図 1.1 のように、均一面にパターンが照射される場合は、パターンは均等大きさ、位置に投影される。一方、図 1.2 のように、前景に物体がある場合では、投影パターンは投影される物体の形状によって変形する。そこで、パターン照射方式では、その変形した歪みを測定することで、奥行きや物体形状を把握する。

次に、ToF 方式では、赤外線を観測対象に照射する。照射された赤外線は観測対象に反射し、その反射波が撮像素子に到達する。このとき、照射された赤外線と、その反射波には位相差が発生する。この位相差から時間差を求め、光の速度を掛け合わせることで、奥行きを計測する (図 1.3)。ここで、ToF 方式はある観測対象の奥行きを一度に取得することができるが、照射する光が干渉して、求める奥行きが安定しない場合がある。そこで、測量などに用いられるレーザーレンジファインダでは、図 1.4 のようにレーザーをミラーで反射させ、そのミラーの角度を調整することで一点毎に奥行きを取得する。こうすることで、周囲の点と干渉が起こらず安定した奥行きの取得が可能となる。一方で、一点毎に奥行きを取得するため、観測対象全体の奥行きを取得するには時間を要する。

また、パッシブ方式には、ステレオ方式 [6]、マルチベースラインステレオ方式 [7]、カメラアレイ [8–12]、レンズアレイによるライトフィールドカメラ [13] などがある。いずれの方式においても原理は同様である。まず、撮影対象を複数視点から撮影する。このとき、得られる複数視点の画像は、それぞれの視点位置に応じて少しずつずれた画像が撮影され、そのずれた画像中で対応点を求める。そして、対応点から三角推量によって奥行きを算出する。それぞれの異なる点としては、ステレオ方式は 2 視点分のカメラを用い、マルチベースラインステレオ方式はさらに多くのカメラを用いる。カメラアレイは、カメラを規則的に配置することで、カメラ配置の規則性を用いた高精度な推定を行う。ライトフィールドカメラは、イメージセンサの直前にレンズアレイを配置することでカメラアレイの機構を 1 つのデバイスで実現する。

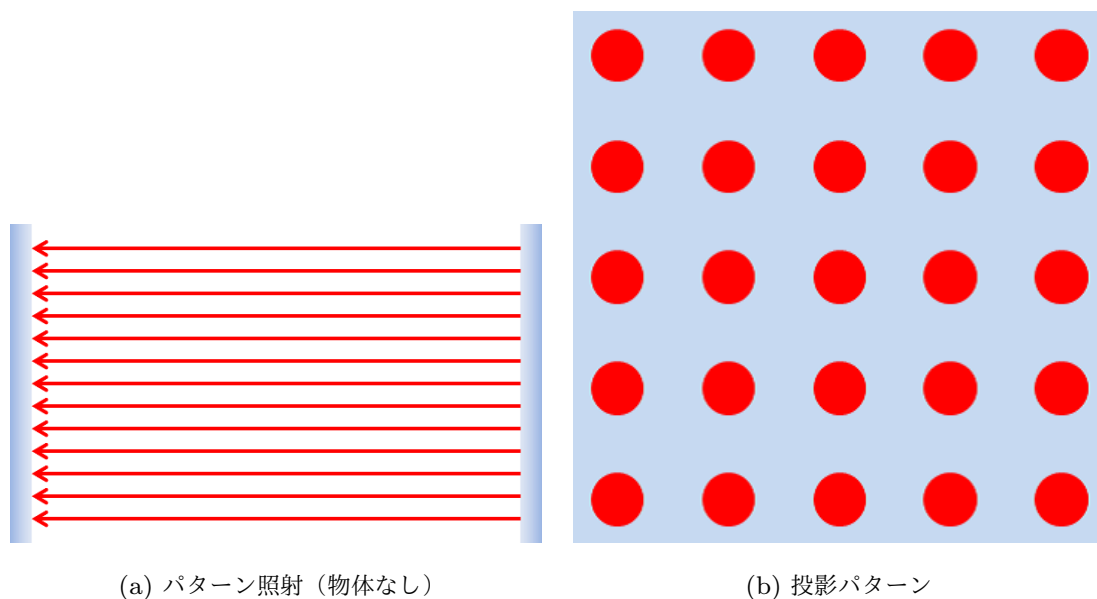


図 1.1 パターン照射方式（前景物体なし）.

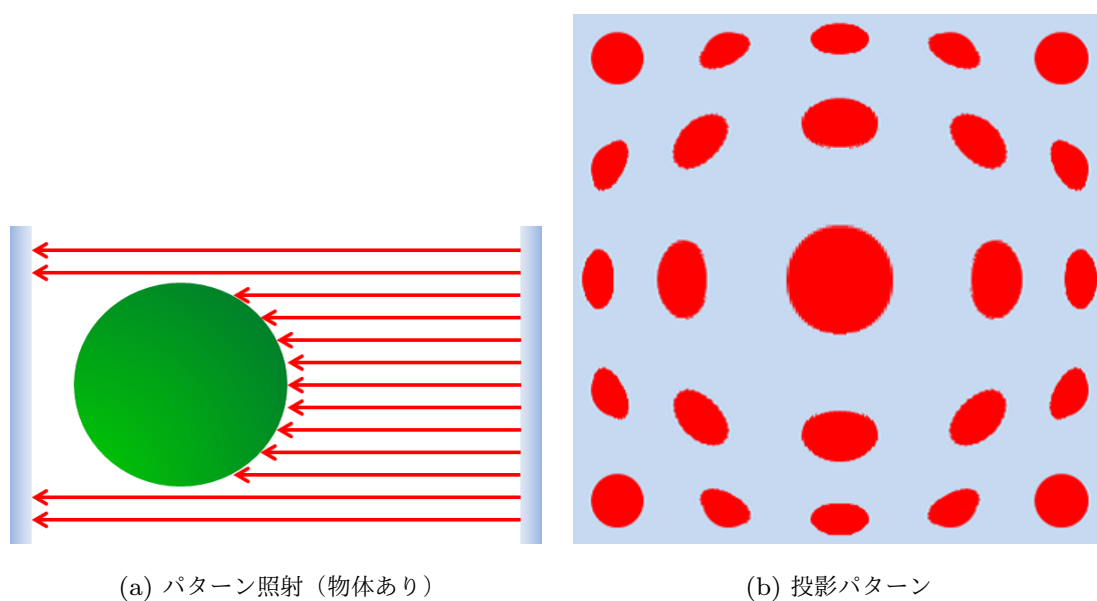


図 1.2 パターン照射方式（前景物体あり）.

アクティブ方式は高精度な奥行き推定が可能な反面，パターンや光を照射するため，カメラに追加してパターン照射用のプロジェクタや赤外線カメラなどが必要になる．また，太陽光の影響により，使用範囲はパターン照射が届く範囲に限定され屋外での使用には向かないことが多い．また，レーザレンジファインダなどでは，屋外での使用も可能であるが，奥行き取得に時間がかかり，取得時間を短縮すると低解像度の奥行きしか推定できな

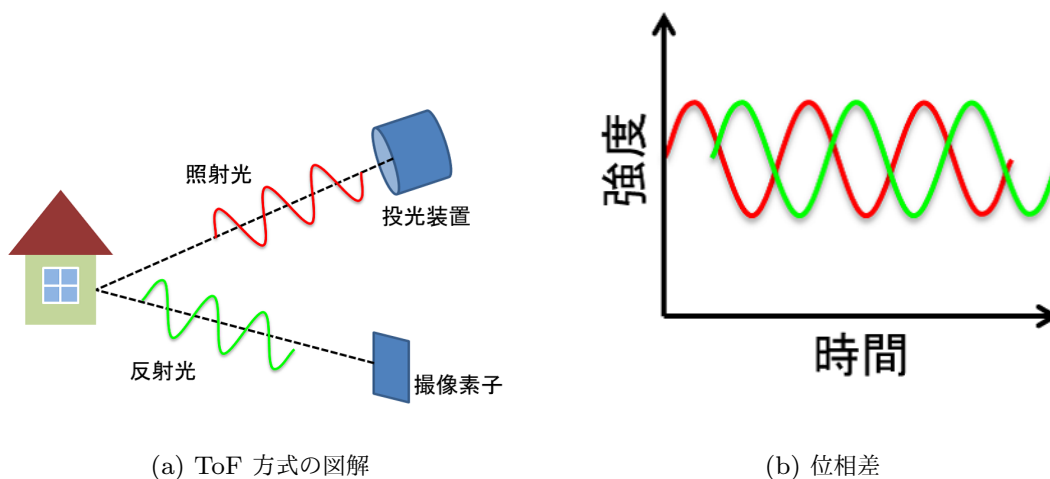


図 1.3 Time of Flight 方式.

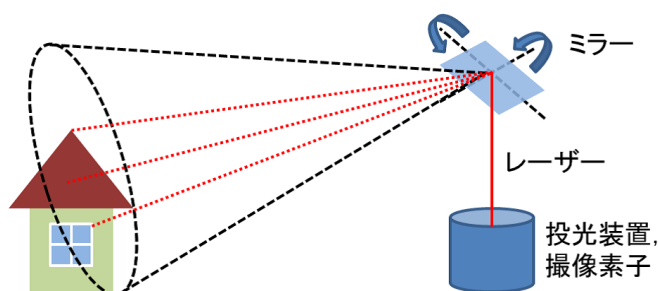


図 1.4 レーザーレンジファインダ方式.

い。そして、装置自体が高額でもある。一方、パッシブ方式では、単純なカメラを用いるため、使用範囲の限定は少ないが、推定精度や推定速度にはまだまだ改善の余地がある。そこで本論文では、特に、使用範囲が柔軟で様々なアプリケーションへの適用が容易な 2 眼ステレオ方式を検討のターゲットとしている。

1.3 ステレオマッチング

1.3.1 2 眼ステレオによる奥行き推定の基本原理

パッシブ方式の奥行き推定で用いられる、2 眼ステレオによる奥行き推定の基本原理について説明する。2 眼ステレオカメラによって、同一シーンを撮影した場合、カメラから近いものは左右の画像で大きく位置が違って写り、一方でカメラから遠いものは位置ずれ

は小さくなる (図 1.5). このずれは視差と呼ばれ, 画像中のすべての画素でその視差を計算する. そして奥行き情報は, 視差とカメラ間の距離, カメラの焦点距離を用いて三角推量によって計算する.

三角推量を行う原理を図 1.6 に示す. ここで, 2 視点間の距離を l , 左右の画像の対応点の視差を d , カメラの焦点距離を f , カメラから対象物体までの奥行きを Z とする. カメラの焦点距離 f から, 対象となる物体までの奥行き Z は, 三角推量を用いて次のようになる.

$$Z = \frac{fl}{d} \quad (1.1)$$

これは, 二つの画像を見比べたとき, カメラから近い位置にある物体の視差は大きく, カメラから遠い位置にある物体の視差は小さくなることを表している. また, 無限遠点にあるものは視差がないため, d の値は次のようになる.

$$d \geq 0 \quad (1.2)$$

視差が求まっていた場合, この処理を全画素で行うことで, 奥行き情報は取得可能である.

次に, 視差を求める基本的な方法を説明する. 理論上, 同一シーンを撮影した場合, 左右の対応する画素は同一画素値となるはずである. しかし, カメラの性能差や照明等の環境光などにより, 完全に一致するとは限らない. そこで, 最も似ている画素を探索し, それらを対応する画素とする. ここで, 2 視点画像を左画像 I_l , 右画像 I_r とし, 左画像 I_l のある画素 $I_l(i, j)$ に対して右画像 I_r の対応する画素を走査する. そして, 画素 $I_l(i, j)$ に最も似ている一点を画像 I_r から探しだし対応する画素とする. ここでは簡単のため, 対応する画素の判断は, 画素値の差の絶対値を評価指標とする. そして, この対応する画素の座標の差 k が視差となる. この画素 $I_l(i, j)$ における視差を $d(i, j)$ とすると, 以下の式で示される.

$$d(i, j) = \arg \min_k |I_l(i, j) - I_r(i - k, j)| \quad (k_{min} \leq k \leq k_{max}) \quad (1.3)$$

$I_l(i, j)$, $I_r(i - k, j)$ は 2 視点の入力画素, k_{min}, k_{max} は最小及び最大視差である. この $d(i, j)$ を, 式 (1.1) を用いて奥行き情報に変換する. これが, 三角推量による奥行き推定の基本となる. このように奥行きを推定するための対応付け手法はステレオマッチングと呼ばれる. なお, 実際に行われる視差計算においては, 早期のステレオマッチングでは,



図 1.5 左右の画像にある視差.

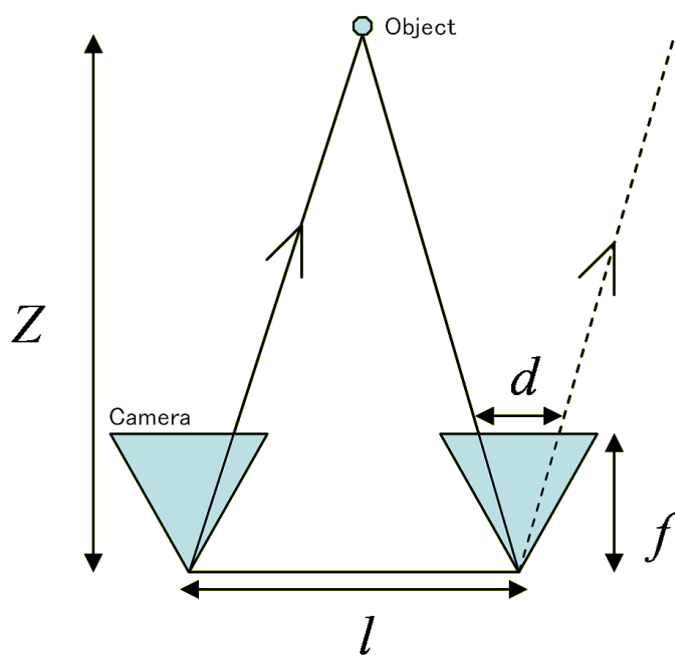


図 1.6 三角推量による奥行き情報の復元.

画素単位ではなく、ブロック単位で一致度を測る Sum of Absolute Difference (SAD) や Sum of Square Difference (SSD), 正規化相互相関が使われていた. 現在でも, この古典的な手法は OpenCV や Matlab で実装されている^{*1}. その他にも公開されているステレオマッチングの実装として, MPEG から公開されているものがある [17, 18].

^{*1} OpenCV [14] と Matlab の実装では, 単純なブロックマッチングだけではなく, 複数のポストフィルタリングが入ったもの [15] とセミグローバルマッチング [16] である.

この式を直感的に実装すると、コスト関数を連続的にメモリ上に配置し、1画素ずつ視差を考える Array of Structure (AoS) の構造となる。しかし、より洗練されたステレオマッチングでは、ある視差に対応するエラー画像を作る Structure of Array (SoA) の構造で実装される。この視差に対応するエラー画像は Disparity Space Image (DSI) と呼ばれ、取り得るすべての視差についての DSI をまとめたものは Disparity Space Volume (DSV) と呼ばれる [19, 20]。近年のステレオマッチングは、この DSI, DSV 上での、より多くの工程を経て出力される。

ステレオマッチングは、コスト計算、コスト集約、最適化と視差計算、精度改善の四つのステップに分解されると言及されている [6]。コスト計算では、各画素単位の対応関係からマッチングコストを計算する。つまり、DSI の構築を行うことに相当する。コスト集約では、各画素のマッチングコストをフィルタ等で集めることで、マッチングコストのノイズを抑制してマッチングを安定化させる。最適化と視差計算では、マッチングコストを最適化し、例えば、周囲と滑らかに連結するような視差を求める。この段階で、奥行き情報（デプスマップ）を取得可能であり、次の精度改善により、一度取得したデプスマップに対して、フィルタリングなどを用いることで、外れ値除去、ノイズ除去等を行い、さらにデプスマップの精度を高める。次節では、各ステップについて詳細に説明する。

1.3.2 コスト計算

コスト計算では、各取り得る視差に対応する一致度（エラー）を何らかの距離尺度等で算出し、それを画素単位でマッチングコストとして計算する。古典的な画素間の一致度を測るコスト関数としては、差の絶対値を用いる Absolute Difference (AD) や差の二乗誤差を用いる Square Difference (SD) がある。これらの関数は、各画素の画素値に対して絶対値差や二乗誤差を計算するが、画素値の差に応じてそのエラー値は単調に増加する。そこで、エラー値が過剰になり後段のコスト集約への影響が大きくなることを防ぐために、打ち切り関数を併用しエラー値をある程度の大きさで打ち切ることが多い。この打ち切り関数を用いることで、ロバストな挙動をさせることができる。

AD や SD は輝度や色差といった画素値を直接用いるため、わずかな輝度変化や色の違いに弱い。屋外で利用することを考慮すると、左右のカメラのカラーバランスが変化することも考えられる。そこで、色変化にロバストな手法が求められる。それを解決する手法としては、正規化相互関数などの相関関数が古くから用いられている。これらは、次に続くコスト集約の処理も含めているが、一方の画像中の処理画素を含む周辺画素集合をテンプレートとして、もう一方の画像の中での相関値を求める。相関値は 0~1 で表され、テンプレートと似ているか否かを判断するため、輝度や色差の変化に強いという特徴がある。

他にも、画像の輝度や色差の変化に強い手法としては、コスト計算の際に画像中の勾配を比較する手法もあり、簡単なものでは Sobel Filter の出力を用いたり、Difference of Gaussian (DoG) を用いたり、エッジ保持平滑化フィルタとの差分（例えば、バイラテラルフィルタ [21] など）を用いたりするものも提案されている [22, 23]。これらの勾配を比較する手法も、AD や SD と同様に打ち切り関数と併用されることが多い。

また、画像変換を用いてコスト計算を行う方法もある。例えば、RANK 変換や CENSUS 変換 [24, 25] などがあり、それらは処理画素周辺（ウインドウ）の画素をそれぞれのルールに従って変換し、変換した値同士を比較することでコスト計算を行う。そのため、輝度や色の変動等がなく、AD や SD と比べてロバストにマッチングが可能である [26]。RANK 変換では、ウインドウ内の画素値の順序関係に変換し、その順序関係を比較する。また、CENSUS 変換では、ウインドウ中心画素とそれ以外の画素値を大小関係にバイナリ変換し、その大小関係を比較する。特に後者は、XOR 命令や popcnt と呼ばれる CPU に備わる Single Instruction Multiple Data (SIMD) 命令を使うことで高速に演算可能なハミング距離を用いて比較することが可能である。

そのほか、自己相互情報量 (Mutual Information: MI) [27] を用いたマッチング手法も色変動にロバストである。MI では、すべてのコスト計算の結果をいったん出し、その後情報量を最大化するように繰り返して計算を行う。この MI は、後に説明する最適化手法を組み合わせることも多い [16, 28]。

これらの輝度や色差の変化にロバストな手法は、特徴的な点以外の対応点を測る能力が弱くなる傾向がある。一方で、輝度や色差を用いる手法では推定する領域による依存性はない。そこで、両者のバランスをとる手法として、AD や SD などの直接値を比較するコスト関数と、色の変化に強いコスト関数をブレンドする方法が提案されており [29]、近年では主流となっている。

また、画素間の対応をとる際の大きな問題として、サブピクセル問題がある。処理画素の色が大きく変化する領域では、カメラレンズの特性や CCD のサンプリングなどの影響でぼけやエイリアシングが発生する。そのような領域では、対応点間のエラーが大きくなり一致度を測る能力が劣化することをサブピクセル問題という。サブピクセル問題を解決するために、文献 [30] では、前後に半画素ずれた対応点もマッチング候補に加えて対応点探索することを提案している。もし、簡単な方法でこれを実装すると対応点探索の計算コストは6倍となる。しかし、この論文では効率的な演算方法も同時に提案しており、わずかな計算コストの増加で抑えられることを示している。また、このコスト関数を用いて、動的計画法による最適化を行うことで、高精度に奥行き推定ができることを示している。

その他、Scale-Invariant Feature Transform (SIFT) [31, 32] などの画像特徴量を画素単位で用いる SIFT Flow [33] 手法など、画素あたりの情報量をより集約することにより、高いマッチング精度を得ることができると報告されている。

1.3.3 コスト集約

コスト集約では、コスト計算で計算した各マッチングコストをスライス毎に集約する。これは、DSI または DSV の平滑化を行うことに相当する。

古典的な方法としては、ブロックマッチングに相当するボックスフィルタで集約することが考えられる。このとき、ボックスフィルタを用いることで、DSI のノイズ除去が行われ、マッチングはノイズに対してロバストになる。このボックスフィルタは、Summed-area-table [34] や積分画像 [35] を用いることで定数時間で実行可能であり、また、高効率な並列化手法 [36] により高速に実行可能である。そのため、ブロックマッチングは、ステレオマッチングをリアルタイム処理をするための代表的な手法となっている。

一方で、ボックスフィルタは物体境界を意識しない単純なフィルタであるため、境界付近の大きなエラーを伝播させることになり、マッチングが不安定になったり、視差が実際の物体よりも大きめに与えられるファッティング効果が発生する。この物体境界のエラーを抑えるために、処理画素の周囲で中心画素の位置をずらしたウィンドウを複数構築し、最もエラーが小さくなるウィンドウをマッチングウィンドウとするシフトブルウィンドウ [37–40] も提案されている。この方法を用いることで背景にまたがるウィンドウを避けることができるため、ファッティング効果も抑制することができる。ここで、直感的な実装では、1つの画素に対して複数のウィンドウを構築するため、ウィンドウの数だけ計算コストが増加する。しかし、ボックスフィルタを用いた後で、シフトしたいウィンドウのサイズに応じた最小値フィルタを DSI に適用することで全く同様の処理が実現可能である。この最小値フィルタは高速に計算可能であり [41, 42]、ブロックマッチングに加えてわずかな計算コスト増で実現可能な処理となる。

また、物体境界をより考慮した手法としては Adaptive Support Weight (ASW) がある [43, 44]。ASW では、入力である左右のステレオ画像を参照してフィルタカーネルを構築するジョイントバイラテラルフィルタ [45, 46] を用いて DSI を平滑化する。このとき、論文 [44] では、より精度を上げるために、左右の両画像をカーネル計算に用いる Dual Bilateral Filter [47] を用いている。この手法は、原画像のはっきりした物体輪郭を直接参照することで、DSI 上の物体境界を良好に維持しながら平滑化することが可能である。ASW は、原画像を参照することでファッティング効果を抑制し、ノイズにもロバストな手法であるため、マッチング精度を大きく向上することが可能である。本手法が登場するまで、後述する最適化手法でなければ高精度なデプスマップの取得はできないと考えられていた。しかしながら、今日では、ASW のようなエッジ保持平滑化フィルタを用いる手法は最適化手法に匹敵する、もしくはそれ以上の推定精度を誇るようになってきている。ただし、ASW の欠点としては、フィルタの計算コストが高いことが挙げられる。

バイラテラルフィルタを代表とするエッジ保持平滑化フィルタの高速化は非常に多くの研究がなされており、エッジ保持平滑化フィルタを DSV に適用するコストボリュームフィルタリング [48, 49] は、高精度な奥行き情報を取得可能である。特に、ASW に用いられているバイラテラルフィルタは多くの高速化手法が研究されており [50–53, 55–64]、これらを用いれば、ASW の高速化が可能である。そこで、バイラテラルフィルタの高速化手法であるバイラテラルグリッド [51] を活用した高速な ASW 手法 [65] では、精度をある程度犠牲にしつつも高速化に成功している。そのほかにも、より効果的なエッジ保持平滑化フィルタが提案されており、例えば、Guided Filter (GF) [66, 67] はボックスフィルタの組み合わせで計算が可能であり、その計算コストは定数時間である $O(1)$ と小さい。GF は、バイラテラルフィルタの定数時間アルゴリズム [57–59] とは異なり、サブサンプリング等による近似計算も無いためエイリアシングの問題もない。GF を DSV に適用するコストボリュームフィルタリング [48, 49] では、高精度なデプスマップを高速に取得することができる。また、ドメイントランスフォームフィルタ [68] も計算オーダーは同じく $O(1)$ と小さく、加えてカーネルを縦横に分離したフィルタなため、非常に高速な計算が可能である。このドメイントランスフォームフィルタを各 DSI に適用することで、高速で高精度なステレオマッチングが可能となる [69]。他にも、再帰型バイラテラルフィルタ [70, 71] は、ドメイントランスフォームに似た定式化をしており、これを用いてコストボリュームフィルタリングも行われている。セパラブルな実装はバイラテラルフィルタ自体もセパラブルで近似すれば高速であり [52, 53]、ハイパースペクトル画像などの高次元に展開することも容易に可能である [54]。また、ステレオ画像に特化したエッジ保持平滑化フィルタとしては、クロススペースのフィルタリングがある [72]。この手法では、各画素が類似すると判定可能な十字線を事前に求め、その範囲だけで横帯のボックスフィルタを行った後に縦のボックスフィルタを行う手法である。この手法は、計算が非常に単純になるため計算コストは大きく削減できる。

これらのほかにも、画素間の距離を測地線距離で計測するフィルタ [73] を用いることで、細かく複雑な構造物へのフィルタリング精度の向上を図ったものや、入力となるステレオ画像の最小全域木 [74] を事前に構築し、その上で距離を計測することでウインドウという概念をなくし、画像全体からコストを集約する方法も提案されている [75, 76]。

これまでのコスト集約は、フロントパラレルモデルと呼ばれる、物体が画像平面に対して正対していることを仮定してコスト集約を行うモデルを採用していた^{*2}。これを、傾いた平面を仮定してカーネル構築を行い、そのカーネル上で集約することで、道路や壁面などのなだらかに変化する平面をより高精度に対応付けができる手法がある [77–81]。素朴なアイデアとしては、あらゆる平面を仮定してマッチングすればよいが、平面のパラ

^{*2} 単純な正方窓を用いれば、フロントパラレルモデルとなる。

メータである位置、回転、傾きは連続量であり、平面の候補は無数にあるため、その計算は収束しない。その問題を解決する古典的な方法としては、所望の平面（例えば地面や道路）にあらかじめ画像を射影してから、フロントパラレルモデルでステレオマッチングする手法がある [77]。この手法は単純だが、仮定できる平面はひとつに限られる。セグメンテーションとそのセグメントのアフィンパラメータ推定を繰り返す手法 [78] は、複数の平面を仮定することもできる。しかし、この推定手法は、非常に大きな領域に分割されるシーンを対象としている。市街地に限定した手法 [79] では、地面や重力方向を消失点から推測し、シーンの平面構造は疎な特徴（点や線）の対応関係から推定することで、壁面等が向くはずの候補の平面を絞ることができる。はじめにフロントパラレルで奥行きを求め、その推定結果から画素毎の傾きを推定する手法も提案されている [80]。しかしながら、初期値がフロントパラレルで仮定されるため、大きく傾いた平面には対応していない。これらの問題を解決した手法として、また現在、最高水準のステレオマッチングを可能とする技術として、パッチマッチ [82] を用いたステレオマッチングが提案されている。これは、あらゆる平面を仮定し、それを乱択アルゴリズム [83] により高速に計算する手法である。2015 年の今でも、Middlebury のステレオベンチマークサイト^{*3}に多くの手法が投稿されており、パッチマッチをベースにした手法が上位のランクに位置している [81, 84–88]。

1.3.4 最適化と視差計算

最適化と視差計算では、DSI や DSV に対して、何らかの拘束条件を満たすように最適化を行い、その結果から視差を計算する。最も簡単な方法としては、一切の拘束条件を持たない Winner-Take-All (WTA) である。WTA では、各画素において DSV 中で単純に最もコストが低い（高い）値を該当する視差とする方法である。古典的なブロックマッチングはこの手法に相当し、すなわち最適化を行わない手法である。従来では、このような手法では精度の高いデプスマップの取得は困難であったが、最近のエッジ保持平滑化フィルタを用いたコスト集約によって、たとえ WTA であっても良好な結果を得ることができる。

動的計画法 (Dynamic Programming: DP) は、左右の画像で各画素は 1 対 1 に対応することと、その対応関係が前後しないことを拘束条件とする。そして、スキャンライン（画素ライン）で拘束条件を満たし、かつ、DSI のコストが最小となる視差の組み合わせを出力する。DP は最適化手法の中でも古くから用いられる手法である [89–91]。DP は、スキャンラインで最適化を行うためストリーキングノイズと呼ばれる線状の引きずりが発生

^{*3} <http://vision.middlebury.edu/stereo/>

する場合や、急峻な物体形状の変化に追従できない場合がある。しかし、前景の背後に隠れて画素の対応がないオクルージョン領域の判定が可能であることから、現在でも用いられることのある方式である。また、DP の計算オーダーは $O(d)$ と高速な手法である。

他にも、マルコフ確率場 (Markov Random Field: MRF) [92, 93] を用いた最適化手法がある。MRF では、次の式で示されるエネルギー $E(\mathbf{D})$ を最小化する解 \mathbf{D} を求める。

$$E(\mathbf{D}) = E_d(\mathbf{D}) + \lambda E_s(\mathbf{D}) \quad (1.4)$$

ここで、 $E_d(\mathbf{D})$ はデータ項、 $E_s(\mathbf{D})$ は平滑化項である。データ項 $E_d(\mathbf{D})$ には、マッチングコストの計算、もしくは、コスト集約で計算された DSI や DSV が入る。そして、平滑化項 $E_s(\mathbf{D})$ は、アルゴリズムが用いる拘束条件を規定し、 \mathbf{D} はデプスマップである。一般に MRF 最適化は NP 困難であり、実用的な時間で解くための近似が必要である。その近似として、スキャンライン最適化やセミグローバルマッチング、マルチパス動的計画法、グラフカット、確率伝播法などが提案されている。

スキャンライン最適化 (Scanline Optimization: SO) [6] は、DP を MRF に対して適用する手法である。前述の DP とは拘束条件の定義が異なり、ある画素とその画素に連結する周囲の画素の値が滑らかになるようにエネルギーを定義し、そのエネルギー関数を最小化する手法である。SO では MRF の確率モデルの連結を 1 次元に近似し、その次元にのみ最適解を導く。SO も DP と同様にストリーキングノイズの発生は避けられず、また、計算オーダーは $O(d^2)$ と DP よりも大きい。一方で SO の拘束条件は、よりステレオマッチングに適したものになっており、DP では画素の対応関係が前後することを許さなかったが、SO では対応関係の反転を許すなどより柔軟なものになっている。

これらの 1 次元のスキャンライン最適化問題を多方向に行い、その後最適化結果を結合する方式として、Semi-Global Matching (SGM) [16, 28] や Multi-Pass Dynamic Programing (MPDP) [94, 95] がある。これらの手法は、SO の方向を 4 近傍や 8 近傍の多方向に設定して、その計算過程を総和することで、より周囲の情報を考慮した最適化を行う。しかしながら、計算過程で総和をとるため、最適解の導出とはならず、したがって、Semi-Global (準最適) という名称がつけられている。一方で、準最適な値の導出には、動的計画法の計算過程を保持する必要がない。そのため、最適化を多方向に増やした分の計算コストの増加は避けられないが、効率的に計算すれば $O(d)$ の計算が可能である [95]。また、SGM は計算コストと精度のバランスがよく、様々な実装が提案されている。画像処理を行う公開ライブラリである OpenCV [14] には、SIMD が有効に働くようにコスト関数や DSV が設計されている SGM が実装されており、CPU に最適化されたものを用いることができる。加えて、いくつかのハードウェアに特化した高速な SGM の実装が提案されている (CPU [96], GPU [97], FPGA [98, 99])。さらに、SGM の高精度

化のための提案もなされている [100, 101]. そのほかの DP や SO の拡張手法として、画像を木構造で表現し、その木上で動的計画法を使うことでストリーキングノイズを抑制する全く異なるアプローチ [102] も提案されている。

グラフカット (Graph Cut: GC) は、MRF をグラフ構造で表現して解く手法であり、特に視差が 0 か 1 しかないような 2 値ラベルの場合は、近似なしに最適解を導出することができる [103, 104]. 多値ラベルの場合には、視差を α と β の 2 値に仮定し、その仮定を入れ替えて解く $\alpha - \beta$ スワップと呼ばれる手法や、ある視差とそれ以外の視差という仮定において解く α 拡張と呼ばれる手法で近似して解かれることが多い。GC は、DP や SO のスキャンライン単位の最適化とは異なり、画像全体で視差を最適化可能な手法として広まり [105], 数多くの発展的な手法が提案されている [106–110]. また、GC を高速に実行するために、GPU 実装である CudaCut [111] も提案されている。

確率伝播法 (Belief Propagation: BP) は、確率を周囲に伝播することで MRF を解く手法であり、もしループのない木構造モデルなら近似なしに最適解を導出できる [112, 113]. ステレオマッチングの場合では、ループ構造を仮定した Loopy Belief Propagation (LBP) を用いて MRF を解く場合が多い。LBP については、様々な高速化手法 [114–117] や高精度化手法 [118–120], 手入力の情報をヒントにした高精度化手法 [121] などが提案されている。

1.3.5 精度改善

精度改善では、最適化と視差計算の結果から取得したデプスマップに対して、誤対応除去や除去した欠損値の補間、サブピクセル推定や推定したデプスマップの輪郭補正を行うことで、推定結果のさらに改善する。これらの精度改善手法は、他のステップに比べて高速に動作する処理が多い。

誤対応除去では、対応付けが不安定な場所や、対応結果に信頼が置けない特徴的な模様となる画素を、「対応無し」の欠損値としてラベル付けする処理である。最も簡単な誤対応除去は、メディアンフィルタであり、インパルス的に誤った点の除去と同時に適当な値で除去されたピクセルを代替してくれる。他にも、対応の信頼度 [122, 123] を考え、コスト関数の最小値と 2 番目の最小値の差が小さかったり、曲率が大きかったりした場合に曖昧性が高いとしてその対応を除去する方法がある。この処理を行うと、テクスチャの少ない領域での誤対応を除去することができる。加えて、LR クロスチェック [124] という手法が有効である。ステレオマッチングを行うとき、左右の両画像でデプスマップを推定し、それらの推定結果の一貫性を見ることで誤りを除去することができる。例えば、同じ点を指すはずの左右のデプスマップが大きく異なる値を持っていた場合、その画素を外れ値とする。この手法を用いると、特にオクルージョン部分（片側の画像で観測できて、も

う片方の画像では観測できない部分)の誤対応を検出することが可能である。また、物体は一定以上の大きさを持つと仮定すれば、奥行き値がほぼ同じである小さなブロブ(塊り)やスペックル領域も疑わしい対応であると推定でき、ブロブ検出を行い、該当するものを誤対応とすることで除去可能である [14]。このブロブやスペックルはテクスチャの少ない部分で生じた大きな白色ノイズを中心にして発生しやすい。

誤対応除去で生じた欠損値は、周辺の値から推定する必要がある。高速かつ簡単だが、非常に有効な方法としては、欠損値の画素から左右の画素を探索し、有効な奥行き値の小さい値を採用することで補間する方法がある [30, 125, 126]。

他にも、繰り返しメディアンフィルタ [127] により補間するものや、インペイント [128] を用いて欠損値を埋める方法が提案されている。その他にも時系列の情報を用いて、デプスセンサの不安定な部分のデプスを埋めた後に、オクルージョン部分を有効な画素のメディアンフィルタで埋めるもの [129] やバイラテラルフィルタで埋めるもの [130]、時系列情報で埋めた後で、残りをインペイントした後に画像情報を用いたジョイントマルチラテラルフィルタによる改善手法 [131] も提案されている。また、繰り返しジョイントバイラテラルフィルタで欠損値補間をする方法も提案されている [132]。この手法は、デプスのエッジと画像のエッジの影響範囲をリージョングロウイング [133] により拡張し、共通部分を間違っている場所として欠損値をマスクし、補間を行う。セグメンテーションを使った繰り返しディフュージョンを用いた補間も提案されている [134]。ノンローカルミンとセグメンテーションを使った最適化 [135] や自己回帰モデルによる最適化 [136, 137] は、計算時間がかかるものの、高精度な補間が可能である。これらの欠損値補間は、後に述べるアップサンプリングとほぼ同義である。それは、周期的か非周期的な欠落ピクセルを補間するかの点に違いがあるだけである。

精度改善前のデプスマップは整数値でサンプリングされた奥行き値で取得されることが多い。整数奥行き値しか持たないデプスマップは、滑らかに変化する平坦部分の奥行きが量子化し、千切りにされたような状態で表現される。そこで、奥行き値のサブピクセル補間を行う。サブピクセル補間は古くから行われており、DSV 中のコスト関数をパラボラフィッティングすることで最も小さなコストを与える視差値を小数点精度で求めることができる [37, 122]。しかしながら、このサブピクセル補間の精度については疑問が投げかけられており [138]、非線形な演算を組み合わせる近年のコスト関数では、その精度誤差の影響は大きい [22, 24, 27, 29, 30]。なお、この問題については Szeliski らによって詳細に調査されている [139]。サブピクセル推定結果は、エッジ保持平滑化フィルタによってさらに精度向上させることができる。特に ϵ -フィルタ [140] やその変形である二値レンジフィルタ [142, 143] が効果的である。これらのフィルタは、ガウシアンを重みを持つバイラテラルフィルタ [21] に比べて急峻なエッジに対応しやすく、ぼけの無いデプスマップに適したフィルタである。このエッジ保持平滑化フィルタはノイズ除去の効果も発

揮している。近年、ライトフィールドの登場により低解像度な多視点画像を超解像することが重要になってきており [141]、デプスマップのサブピクセル化は、ライトフィールドの超解像にも重要である。

最後に、輪郭補正のためにフィルタリングが行われる。これまでの精度改善処理の中にも、いくつかのフィルタを使用した⁴が、特に輪郭補正のためのフィルタリングについて簡単に説明する。推定したデプスマップの輪郭情報は、ファッティング効果や輪郭周辺のマッチング精度の曖昧性により、原画像（左右の入力画像）と輪郭の位置にずれが生じている。これを修正するのが、別途参照画像を用いるジョイントフィルタである。例えば、ジョイントバイラテラルフィルタ (Joint Bilateral Filter: JBF) [45] やクロスバイラテラルフィルタ (Cross Bilateral Filter: CBF) [46] は、デプスマップではなく、原画像などの参照画像を用いてカーネル重みを計算しフィルタリングする。原画像の輪郭に沿って重みが定義されるため、輪郭の位置が原画像の位置に再配置されるような補正フィルタリングとなる。このジョイントフィルタを用いれば、デプスマップのアップサンプリングを高精度に行えるため高速化の用途にも使える [144]。このジョイントフィルタに関する詳細な議論は次節で行う。

1.4 ジョイントフィルタ

ジョイントフィルタとは、フィルタ対象以外に別途参照画像を用意し、その参照画像の情報も用いてカーネルの重みを計算するフィルタである。このジョイントフィルタとステレオマッチングの関係は深く、コスト集約や、精度改善中の欠損値補間、輪郭補正で使われる。加えて、デプスマップの高精度なアップサンプリングでも使われる。ここでは、本論文で使われるジョイントフィルタについて概説する。

ジョイントフィルタとは、ある画像をフィルタするとき、別の画像を参照してカーネル計算を行うフィルタである。代表的なジョイントフィルタであるジョイントバイラテラルフィルタ^{*4} は、バイラテラルフィルタを2枚の入力画像から計算する [45]。文献の例では、フラッシュを焚いたフラッシュ画像（ノイズは少ないが、色合いが変化）と焚かなかったノンフラッシュ画像（ノイズが多いが色合い、照明の状態を維持）を入力として、ノンフラッシュ画像の色味を維持しながら、ノイズの少ないフラッシュ画像の情報を用いて高精度にノイズ除去ができることを示している。他にも、赤外線画像や紫外線画像を参照情報に使うなどの応用がある [145]。このように特性の異なる画像を入力することで、ノイズ除去手法 [146–148] よりも高精度なフィルタリングを達成するのがジョイントフィルタである。

^{*4} 同時に発表された文献 [46] ではクロスバイラテラルフィルタと呼ばれる。

このジョイントフィルタは、バイラテラルフィルタだけでなく、その高速化実装やガイデットフィルタやドメイントランスフォームフィルタなどのエッジ保持平滑化フィルタや、高次元フィルタ [60, 61, 149] にも拡張されており、高速かつ高性能なジョイントフィルタが数多く提案されている。このジョイントフィルタは特性に応じて応用例も変わる。例えば、ガイデットフィルタはデプスマップを用いた自由視点画像合成のためのマッティングのためにも用いることができる [150]。

このジョイントフィルタをアップサンプリングに応用することで、1枚超解像 [151] や、エッジの方向を考慮した高度なアップスケーリング [152, 153] を行うよりも容易で高精度な結果を得ることが可能である。この応用では、低解像度のデプスマップと高解像度の参照画像を利用し、デプスマップを高解像度の画像が持つエッジ方法を参照することで、高精度なアップサンプリングを実現する。

代表的なジョイントアップサンプリングであるジョイントバイラテラルアップサンプリング (Joint Bilateral Upsampling: JBU) [144] では、ジョイントバイラテラルフィルタをカーネル計算に用いてこのアップサンプリングを実現している。論文中では、デプスマップのアップサンプリングのほかにも、セグメンテーション結果のアップサンプリングやカラリゼーション結果のアップサンプリングの例が示されている。しかしながら、JBU など汎用的なエッジ保持平滑化フィルタを用いると、物体境界がわずかにぼけたり、参照画像のノイズをデプスマップへ転写するため、アップサンプリングをするためのいくつかの方法が提案されている。

ノイズに応じて JBU とバイラテラルアップサンプリングを切り替える Noise-Aware Filtering for Depth map Upsampling [154] は、ノイズの転写を抑制することができる。ガイデットフィルタ [66] によるアップサンプリング^{*5} は非常に高速にアップサンプリングが可能である。ジョイントバイラテラルフィルタに変わってノンローカルミンフィルタを使ったアップサンプリング [155] や測地線距離を用いたアップサンプリング [156] は、ボケを抑制しつつ高性能に拡大することができる。JBU においてブレンドする代わりに最も一致しているデプス値を使う方法 [157] やヒストグラムベースのアップサンプリング [158] は、ボケを完全に抑制可能である。時間軸を含めた RGBt の高次元ジョイントインターポレーション [159] や、セグメンテーションを用いたアップサンプリング [160, 161] は、不均一にサンプリングされた低解像度のデプスマップを直接高解像度のデプスマップに変換可能である。

また、初期値として何らかの方法でアップサンプリングしたデプスマップに、精度改善のリファインメント処理を行うことで高性能なデプスマップを生成する手法もある。

^{*5} 後に発表された詳細な記述がある文献 [67] では、係数画像 a , b を線形補間することで実現している。

1.5 本論文の位置づけと構成

本論文では、パッシブ方式のステレオマッチングを対象として、高精度なデプスマップを高速に取得する方法を検討する。コスト集約や最適化のステップは多くの研究がされているのに対して、精度改善のステップは比較的研究が少ない。加えて、このステップの計算コストは比較的小さくなりやすい。そこで、まずステレオマッチングの精度改善のステップに対して、ジョイントフィルタを用いた手法で性能向上を試みる。そして、その精度改善のプロセスを生かした新たなステレオマッチングフレームワークを提案する。さらに、高精度なジョイントフィルタによるデプスマップのアップサンプリング手法を提案し、さらなる高速化手法や配信のための符号化応用を示す。

第2章では、高速動作が可能な精度改善に着目し、ジョイントフィルタによる精度改善手法を提案する。これまで提案されてきた多くの手法は、最適化やコスト集約において計算コストをかけることで高精度な奥行き推定を行うものがほとんどであった。そのため、従来の精度改善手法は、外れ値の除去や欠損値の補間など推定誤差を補正する要素が強く、その単体性能は必ずしも高いものではない。また、最適化やコスト集約での計算コストが大きいため、高精度な奥行き推定が行える一方で、高速に動作しない。逆に、最適化やコスト集約に計算コストをかけなければ、高速に動作する一方で、低精度なデプスマップしか取得できず、従来の精度改善手法ではその改善精度は十分でない。そこで、低精度なデプスマップを前提とし、それを高速な精度改善手法によって効率的に改善する。ここでは、低精度なデプスマップに発生するノイズや輪郭周辺のファッティングを考慮して、適応的に重み付けするジョイントフィルタによってフィルタリングを行う。これにより、デプスマップを効率的に精度改善し、また、ステレオマッチング全体の計算コスト低減することで、推定精度と処理速度の両立を図る。

第3章では、ステレオマッチング全体のフレームワークを提案する。ステレオマッチングのコスト集約のステップでエッジ保持平滑化フィルタを用いれば、最適化を行うのと同程度の精度が達成できる。しかし、すべての取り得る奥行き値を示すDSIに対してフィルタリングをする必要がありそのコストは大きい。そこで、計算コストの小さいエッジ保持平滑化フィルタによる輪郭の補正を精度改善ステップで行い、その結果をコスト集約以前にフィードバックし、ボックスフィルタで再集約するという新しいステレオマッチングのフレームワークを構築することで、高精度かつ高速な手法を提案する。

第4章では、ステレオマッチングの後段にアップサンプリングを追加することを提案する。ステレオマッチングにおける計算コストは、画像サイズと取り得る奥行き値に依存する。そのため、画像をサブサンプリングして、画像サイズを小さくすると計算コストを下げることができるが、デプスマップの解像度も低くなる。そこで、デプスマップに特化し

たアップサンプリングを検討することで，ステレオマッチング全体の計算コストを大きく削減しつつ，高精度な奥行き推定を実現する手法を提案する．また，サブサンプリングすることでデプスマップの符号化効率が向上し，サービス等でデプスマップを配信するときに高効率な伝送が可能であることを示す．

最後に，第 5 章で本論文をまとめる．

第 2 章

ジョイントフィルタによるデプスマップの精度改善

2.1 まえがき

デプスマップは 2 眼ステレオカメラを用いたステレオマッチングによって推定することができる。ステレオマッチングは、コスト計算、コスト集約、最適化と視差推定、精度改善の 4 つのステップに分類できると言われており [6]、高精度化に関する様々な研究が行われている。高精度化に対するアプローチとして、古くからは定義した拘束条件を最適化する手法や、近年ではジョイントフィルタによるコスト集約手法が提案されている。これらの手法では、取り得るすべての奥行き値を考慮したり、すべての奥行き値のコストにフィルタリングすることで高精度なデプスマップを取得する。一方で、計算コストが膨大となるため処理時間がかかり、デプスマップの精度と処理時間の両立が難しい。

処理時間を削減する方法として考えられるのは最適化やコスト集約を簡単にすることである。しかしながら、簡便なステレオマッチングでは、マッチングが安定しないためノイズが多く、ファッティングにより奥行き値の画素値境界と原画像の物体輪郭がずれたデプスマップが取得されることが多い。ステレオマッチングの処理ステップには、取得したデプスマップの精度改善を行うステップが存在するが、従来の精度改善手法は外れ値除去や欠損値の補間など限定的な効果しか持たないものが多く、改善精度が十分ではない。したがって、高精度なデプスマップを高速に取得するためには、より効果的な精度改善手法が必要となってくる。そこで本章では、高速動作が可能な精度改善に着目し、ジョイントフィルタを用いたデプスマップのノイズ抑制と輪郭補正を行う。

以下では、まず 2.2 節で、精度改善に用いられる従来手法の説明を行い、従来手法における問題点と提案手法で解決すべき課題を述べる。次に、2.3, 2.4, 2.5 節では、ジョイントフィルタを用いた 3 つの精度改善手法を提案する。

2.2 従来手法

一般に高速に動作するステレオマッチングではノイズが多く，輪郭精度が低いデプスマップが取得される．そのため，フィルタによる精度改善によって境界精度の改善やノイズの低減が行われる．ここでは，従来研究で頻繁に用いられるメディアンフィルタ，ガウシアンフィルタ，バイラテラルフィルタ，ジョイントバイラテラルフィルタについての説明を行う．そして，これらの手法における問題点を明確化し，提案手法で解決すべき課題を述べる．

2.2.1 メディアンフィルタ

メディアンフィルタは平滑化フィルタの中でも原始的なフィルタの一つである．メディアンフィルタは処理画素をフィルタカーネル内の画素値の中央値に置き換える．そのため，特に周囲の奥行き値からインパルス的に誤ったスパイクノイズを除去するのに効果を発揮する．

図 2.1 はメディアンフィルタを図示したものである．図 2.1(a) は入力画像，図 2.1(b) は入力画像にメディアンフィルタをかけたものとする．ここで，各画素の数字は，その画素の画素値とする．ここでは，カーネルサイズが (3×3) のメディアンフィルタをかけるものとして，その周囲の画素も示す．図 2.1 のフィルタ前の画像の処理画素 (186) を中心とした (3×3) のメディアンフィルタをかけるとき，カーネル内に入る画素の輝度値は，(145, 113, 136, 212, 186, 151, 125, 113, 227) となる．そして，このときの中央値は 145 となるため（カーネル内の画素値を昇順ソートし (113, 113, 125, 136, 145, 151, 186, 212, 227)，中央 5 番目画素値），処理画素の画素値を 186 から 145 に置き換える．他の画素でも同様の処理を行った結果，図 2.1(a) となる．なお，ここでは中央の (3×3) の画素にのみ処理を行っている．

メディアンフィルタを用いてデプスマップ改善を行うと，図 2.2 のようになる．ここで，図 2.2(a) はデプスマップの推定に用いる原画像，図 2.2(b) は入力デプスマップ，図 2.2(c) はメディアンフィルタでの改善結果である．図 2.2(b) から，入力のデプスマップには奥行き値が周囲とは大きく異なるスパイクノイズや斑点のようなスペckルノイズが多いことがわかる．また，輪郭周辺は画素値境界の凹凸が多く，さらに，原画像の物体輪郭とも一致していない．対して，メディアンフィルタでデプスマップの改善を行った後では，スパイクノイズが消えていることがわかる．しかし，スペckルノイズは，斑点が大きくまとまったものの，除去されることはない．また，物体輪郭の補正も行われなため，境界精度は低いままである．

233	80	208	207	179
123	145	113	136	159
48	212	186	151	60
121	125	113	227	16
118	23	192	162	34

(a) フィルタ前

233	80	208	207	179
123	145	151	159	159
48	123	145	136	60
121	121	162	151	16
118	23	192	162	34

(b) フィルタ後

図 2.1 メディアンフィルタの図解.

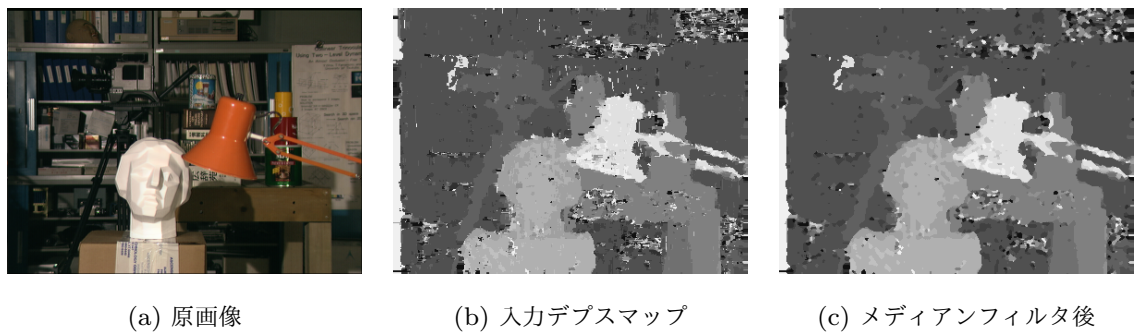


図 2.2 メディアンフィルタの効果.

2.2.2 ガウシアンフィルタ

ガウシアンフィルタはメディアンフィルタと同じく、原始的な平滑化フィルタの一つである。ガウシアンフィルタは、処理画素 \mathbf{p} と周辺画素 \mathbf{s} との距離に応じて画素毎にガウス分布に従う位置重み関数 w で重み付けし、その重み付き平均によってノイズ除去を行う平滑化フィルタである。これは、次の式で示される。

$$D'_{\mathbf{p}} = \frac{\sum_{\mathbf{s} \in N} w(\mathbf{p}, \mathbf{s}) D_{\mathbf{s}}}{\sum_{\mathbf{s} \in N} w(\mathbf{p}, \mathbf{s})} \quad (2.1)$$

$$w(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) \quad (2.2)$$

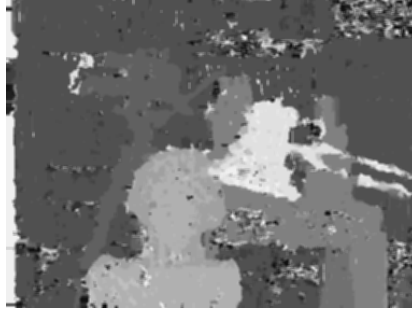


図 2.3 ガウシアンフィルタの効果.

ここで, D, D' は入力及び出力画像, N は周辺画素の集合, σ_s は位置重み関数のガウス変数, $\|\cdot\|_2$ は L2 ノルムである. ガウシアンフィルタでは, ガウス分布に従った重み付けにより, 処理画素との距離が近い周辺画素ほど大きな重みを与えられる. これは, 画像における高い自己相関性を利用し, 位置が近い画素は類似すると仮定するためである.

ガウシアンフィルタを用いてデプスマップの改善を行うと, 図 2.3 のようになる. ここでは, 入力デプスマップとして図 2.2(b) を用いている. 図 2.3 から, デプスマップ全体がぼけていることがわかる. そのため, スパイクノイズ周辺との奥行き値の差が小さくなっているが, ノイズが消えたとは言いがたい. また, 輪郭には凹凸が目立ち, さらにフィルタ前でははっきりしていた画素値境界でぼけが発生している. ガウシアンフィルタでは, 画素間距離に応じた重み付き平均を行うため, 重み付けする画素がノイズか否かは考慮しない. したがって, 画像中の白色ノイズは除去できるが, スパイクノイズやスペckルノイズは除去できず, 周囲に拡散する. さらに, 輪郭は補正できず, ぼけるという問題もある.

2.2.3 バイラテラルフィルタ

バイラテラルフィルタ [21] はガウシアンフィルタの拡張である. バイラテラルフィルタは, 周辺画素との距離に応じて重み付けする位置重み関数に加え, 周辺画素との画素値の差に応じて重み付けする色重み関数 c でガウス分布に従った重み付けする. これは, 次の 2 式で示される.

$$D'_p = \frac{\sum_{s \in N} w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s}) D_s}{\sum_{s \in N} w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s})} \quad (2.3)$$

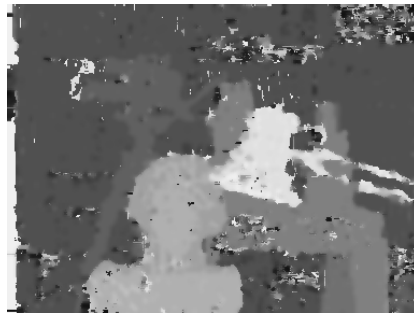


図 2.4 バイラテラルフィルタの効果.

$$w(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) \quad c(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|D_{\mathbf{x}} - D_{\mathbf{y}}\|_2^2}{2\sigma_c^2}\right) \quad (2.4)$$

ここで、 σ_c は色重み関数のガウス変数である．バイラテラルフィルタでは，画素間距離と画素値差のそれぞれに対してガウス分布に従った重み付けを行うことで，処理画素との距離が近く，画素値が類似している周辺画素ほど大きな重みが与えられる．これは，ガウシアンフィルタよりも画像の自己相似性を利用したものである．バイラテラルフィルタでは，画素値を考慮した重み付けによって，類似の画素値に大きな重みを与えるため，画素値境界を維持した平滑化が可能になる．したがって，バイラテラルフィルタはエッジ保持平滑化フィルタに分類される．

バイラテラルフィルタを用いてデプスマップの改善を行うと，図 2.4 のようになる．ここでも，入力デプスマップとして図 2.2(b) を用いている．図 2.4 から，ガウシアンフィルタでのフィルタリングに比べ，奥行き値が大きく異なる輪郭の形を維持したまま，平滑化が行われていることがわかる．これは，色重みによって，奥行き値が大きく異なる部分では重みが小さく与えられ，平滑化に寄与しにくくなるからである．そのため，画素値境界の形を維持し，かつ，ぼけを抑制した平滑化が可能である．しかし，維持する形は入力デプスマップの輪郭であるため，原画像の物体輪郭に対応した画素値境界の補正はできない．また，完全にぼけを防ぐことはできず，画像全体にぼけが見え，ノイズが拡散する．

2.2.4 ジョイントバイラテラルフィルタ

ジョイントバイラテラルフィルタ [45, 46] はバイラテラルフィルタの拡張である．ジョイントバイラテラルフィルタは，バイラテラルフィルタで用いた色重みを，新たに追加する参照画像から計算する．これは，次の式で示される．



図 2.5 ジョイントバイラテラルフィルタの効果.

$$D'_p = \frac{\sum_{s \in N} w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s}) D_s}{\sum_{s \in N} w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s})} \quad (2.5)$$

$$w(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) \quad c(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|I_{\mathbf{x}} - I_{\mathbf{y}}\|_2^2}{2\sigma_c^2}\right) \quad (2.6)$$

ここで、 I は参照画像である．ジョイントバイラテラルフィルタでは，追加する参照画像の画素値に応じて重み付けを行う．したがって，バイラテラルフィルタでは処理画像の輪郭を保持して平滑化したが，ジョイントバイラテラルフィルタでは参照画像の輪郭を保持して平滑化する．

ジョイントバイラテラルフィルタを用いてデプスマップの改善を行うと，図 2.5 のようになる．ここでは，入力デプスマップとして図 2.2(b) を用いている．また，参照画像も図 2.2(a) を用いている．図 2.5 から，バイラテラルフィルタでのフィルタリングに比べ，原画像の物体輪郭に奥行き値の境界が近づいていることがわかる．これは，物体輪郭では参照画像中の処理画素と色が近い周辺画素に対して大きな重みが与えられるため，参照画像の物体輪郭を維持したまま平滑化が行われたからである．しかし，平滑化フィルタであるため，画素値の混合によるぼけが発生する．特に，物体輪郭付近では，奥行き値が大きく異なるため，ぼけが目立ってしまう．また，ガウシアンフィルタやバイラテラルフィルタと同様に画像全体にぼけが見え，ノイズが拡散する．

2.2.5 従来手法の問題点と解決すべき課題

従来手法を確認することで，問題点としては次のことが考えられる．ひとつは，平滑化によってノイズが拡散する問題である．この問題は，ガウシアンフィルタ，バイラテラルフィルタ，ジョイントバイラテラルフィルタにみられた．これらのフィルタは，処理画素

の周辺画素に重みを与えて平滑化する．したがって、ノイズに対して重みを与えられると周囲に拡散し性能低下を及ぼす．その程度はフィルタの設計上、平滑化度合いに由来し、平滑化強度が強いと拡散が広範囲に及び画像全体がぼけたようになる．一方、メディアンフィルタでは重み付き平均は行わず、中央値で画素値を置き換えるためノイズの拡散は発生しにくい．

もうひとつは、輪郭補正ができない問題である．この問題は、メディアンフィルタ、ガウシアンフィルタ、バイラテラルフィルタにみられ、その原因は原画像の情報を一切用いないためである．一方、ジョイントバイラテラルフィルタでは、原画像を参照し原画像の画素値境界を維持した平滑化を行うため輪郭補正が可能である．

そして、最後に画素値境界がぼける問題である．この問題は、ガウシアンフィルタ、ジョイントバイラテラルフィルタにみられた．これも、ひとつ目の問題と同様に周辺画素に重みを与えて平滑化することが原因であり、デプスマップの画素値境界を考慮せずに重みを与えるために発生する．一方、バイラテラルフィルタは、デプスマップの画素値を色重み関数に与えて重み付けするため画素値境界を維持することができる．また、メディアンフィルタは画素値の画素値を置き換えるためぼけが発生しない．

以上から、デプスマップを高精度に精度改善するためには、ノイズの拡散を防ぎつつ効率的にノイズ除去を行い、輪郭を補正し、画素値境界のぼけを抑制する必要がある．次節からは、以上の課題を解決するデプスマップの精度改善手法を三つ提案する．

2.3 信頼度を用いたジョイントトリラテラルフィルタ

2.3.1 提案手法

提案手法では、ノイズ除去や境界精度の向上を行うジョイントバイラテラルフィルタに注目し、これを新たに改良する．なお、提案手法は文献 [162–164] を発展させたものである．

ジョイントバイラテラルフィルタでは、ノイズ除去と輪郭補正を行うことができるが、ノイズの拡散と画素値境界でぼけが発生した．ノイズの拡散は、正解奥行き値とノイズを同じ条件で重み付けすることが原因で発生する．また、画素値境界のぼけは、デプスマップの画素値境界を考慮せずに重み付けすることが原因で発生する．そこで提案手法では、ノイズ拡散の対策として新たに画素毎の信頼度を設定し、それをジョイントバイラテラルフィルタの重みとして加える．信頼度はデプスマップの各画素が誤りか否かを判定し、誤った奥行き値を持つ画素は重みを小さくする．また、画素値境界のぼけ対策としては、デプスマップの奥行き値を考慮した奥行き重みを追加する．

提案手法は信頼度を用いたジョイントトリラテラルフィルタといい、その前段となる奥

行き重み関数 d を追加したジョイントトリラテラルフィルタは次の式で示される。

$$D'_p = \frac{\sum_{s \in N} w(\mathbf{p}, s) c(\mathbf{p}, s) d(\mathbf{p}, s) D_s}{\sum_{s \in N} w(\mathbf{p}, s) c(\mathbf{p}, s) d(\mathbf{p}, s)} \quad (2.7)$$

$$\begin{aligned} w(\mathbf{x}, \mathbf{y}) &= \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) & c(\mathbf{x}, \mathbf{y}) &= \exp\left(-\frac{\|I_{\mathbf{x}} - I_{\mathbf{y}}\|_2^2}{2\sigma_c^2}\right) \\ d(\mathbf{x}, \mathbf{y}) &= \exp\left(-\frac{\|D_{\mathbf{x}} - D_{\mathbf{y}}\|_2^2}{2\sigma_d^2}\right) \end{aligned} \quad (2.8)$$

ここで、 σ_d は奥行き重み関数のガウス変数であり、他のパラメータは式 (2.5)、式 (2.6) に従う。追加した奥行き重み関数は、位置重み関数や色重み関数と同様にガウス分布に従った重み付けを行い、周辺画素の奥行き値が処理画素の奥行き値に近いほど大きな重みが与えられる。これにより、色重みによって原画像の物体輪郭に合わせた輪郭補正が行われ、さらに奥行き重みによって画素値境界のぼけを抑制する。

しかし、重み付き平均による平滑化であるため、誤った周辺画素にもわずかに重みが与えられ、ノイズの拡散が発生する可能性が残る。そこで、処理画素と周辺画素を比較し、ノイズや外れ値となる可能性のある部分には極力重みを与えないように信頼度を設定し、信頼度に従って各重み関数を適応的に制御する。ここでは、信頼度を設定するために次の仮定をおく。まず、参照画像とする原画像で、同一物体を示す処理画素と周辺画素では、画素値は近い値をとると仮定する。また、正しい奥行き値では、同一物体を示す処理画素と周辺画素で、奥行き値は近い値をとると仮定する。そして、ある視点でのデプスマップが与えられたとき、各画素の奥行き値と、他の視点との視点間の距離から、視点間に対応する画素を計算することができる。このとき、奥行き値が正しければ、対応する画素間で一貫性が維持できると仮定する。したがって、左右2視点の原画像と1視点のデプスマップ (ここでは左視点) は与えられているという前提のもと、3つの仮定を次の条件で評価する。なお、前提となる2視点の原画像と1視点のデプスマップや、視点間の距離は一般的なステレオマッチングならクリアできる前提である。

1. 左視点の参照画像である原画像において、処理画素 I_p^l と周辺画素 I_s^l の画素値を比較し、その差が閾値 th_c 以下であること。

$$\|I_p^l - I_s^l\|_1 \leq th_c \quad (2.9)$$

2. 左視点のデプスマップにおいて、処理画素 D_p^l と周辺画素 D_s^l の奥行き値を比較し、その差が閾値 th_d 以下であること。

$$\|D_p^l - D_s^l\|_1 \leq th_d \quad (2.10)$$

3. 左右視点間の距離と、左視点の周辺画素での奥行き値から、右視点における周辺画素の対応画素を一意に定められるため、右視点での原画像の対応画素 $I_{s+D_s^l}^r$ と、左視点での原画像の周辺画素 I_s^l の画素値を比較し、その差が閾値 th_{LR} 以下であること。

$$\|I_s^l - I_{s+D_s^l}^r\|_1 \leq th_{LR} \quad (2.11)$$

ここで、 l, r は左右の視点を示し、 th_d, th_c, th_{LR} はそれぞれ任意の閾値、 $\|\cdot\|_1$ は L1 ノルムである。これらの指標を信頼度として、式 (2.9) によって色重み関数を、式 (2.10) によって奥行き重み関数をそれぞれ適応的に制御する。よって、式 (2.8) の重みの計算式について次式のように変更する。

$$c(\mathbf{x}, \mathbf{y}) = \begin{cases} \exp(-\frac{\|I_{\mathbf{x}} - I_{\mathbf{y}}\|_2^2}{2\sigma_c^2}) & \|I_{\mathbf{x}}^l - I_{\mathbf{y}}^l\|_1 \leq th_c \\ 0 & \text{else} \end{cases} \quad (2.12)$$

$$d(\mathbf{x}, \mathbf{y}) = \begin{cases} \exp(-\frac{\|D_{\mathbf{x}} - D_{\mathbf{y}}\|_2^2}{2\sigma_d^2}) & \|D_{\mathbf{x}}^l - D_{\mathbf{y}}^l\|_1 \leq th_d \\ 0 & \text{else} \end{cases} \quad (2.13)$$

式 (2.12) によって、参照画像での処理画素と周辺画素の輝度値が大きく離れている場合では、わずかな重みも与えられないことになる。また、式 (2.13) によって、入力デプスマップでの処理画素と周辺画素の奥行き値が大きく離れている場合では、同様に重みが与えられないことになる。これにより、誤った奥行き値は処理画素に影響しにくくなる。さらに、式 (2.11) に関しては、この条件を満たす画素のみに重みを与えるように設定する。他の重みでは、処理画素と周辺画素の関係によって重み変動するが、これは、デプスマップと左右の原画像が与えられた際に一意に決定することができるため、あらかじめ算出しマスクとして与える。よって、式 (2.7) の式は次のようになる。

$$D_p' = \frac{\sum_{s \in N} M_s \cdot w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s}) d(\mathbf{p}, \mathbf{s}) D_s}{\sum_{s \in N} M_s \cdot w(\mathbf{p}, \mathbf{s}) c(\mathbf{p}, \mathbf{s}) d(\mathbf{p}, \mathbf{s})} \quad (2.14)$$

表 2.1 画像サイズと奥行き階調数.

データセット	Tsukuba	Venus	Teddy	Cones
画像サイズ	384×288	434×383	450×375	450×375
奥行き階調数	16	32	64	64

$$M_s = \begin{cases} 1 & \|I_s^l - I_{s+D_s^l}^r\|_1 \leq th_{LR} \\ 0 & \text{else} \end{cases} \quad (2.15)$$

ここで、 M は式 (2.11) の条件で作成される重みである．これは、左右の原画像の一致度合いから決定されるため、LR クロスチェックマスクという．これによって、提案手法である信頼度を用いたジョイントトリラテラルフィルタは画素間の位置、参照画像の色、デプスマップの奥行き値を考慮し、かつ、ノイズや外れ値の影響を小さくすることで、ノイズ拡散が発生しにくく、ぼけを抑制しつつ輪郭補正する精度改善フィルタとなる．

2.3.2 評価実験

評価は複数のシーンを撮影した図 2.6 のステレオデータセットを用いて行う．各画像のサイズと奥行き階調数は表 2.1 に示す．これらは、文献 [6] によって提供されており、原画像とともに正解デプスマップも提供されている (図 2.7)．評価は正解デプスマップと取得したデプスマップから計算される次の式に基づく誤り率で行う．

$$(\text{誤り率}) = \frac{|\{(\text{正解デプスマップ}) - (\text{比較デプスマップ}) \geq (\text{閾値})\}|}{(\text{全画素})} \times 100 \quad (2.16)$$

ここでは、閾値は 1 とし、奥行き値が 1 ずれている場合までを正解と見なした．これは、文献 [6] で評価されている基準と同様である．

また、評価するデプスマップの領域を “nonocc”, “all”, “disc” と三つに分けて評価を行う．まず、“nonocc” はオクルージョン部分を除いた領域の評価である．オクルージョンは、ステレオマッチングの際に、2 視点のカメラの一方にしか写らない部分である．この部分では、奥行き値が定まらないため、この領域を除いて評価する．次に、“all” は画像の全領域の評価である．最後に、“disc” は奥行き値が大きく変化する物体境界 (Discontinuity) 周辺のみを評価する．

提案手法の有効性を示すため、従来の精度改善フィルタとの比較を行う．従来手法としては、2.2 節で紹介したメディアンフィルタ (MF)、ガウシアンフィルタ (GF)、バイラテ



図 2.6 使用したステレオデータセット.

ラルフィルタ (BF) [21], ジョイントバイラテラルフィルタ (JBF) [45, 46] を比較手法とする. また, 信頼度や LR クロスチェックマスクを用いない単純なジョイントトリラテラルフィルタ (JTF) [165–167] も比較対象とする. ここで, 提案手法, JBF, JTF の参照画像は, 図 2.6 の画像をそれぞれ用いる.

入力に用いるデプスマップは, 最適化を行わず簡単な局所的手法のブロックマッチング (Block Matching: BM) を用いて取得する. BM では, 2 視点の画像の画像を比較する際に取得したマッチングコストを, ボックスフィルタでコスト集約を行う手法である. BM はボックスフィルタのフィルタカーネルサイズが大きくなるほどカーネル内のノイズ割合が減り, ノイズにロバストなデプスマップを取得できる. しかし, カーネルサイズが大きくなると細かい輪郭が把握できなくなるという問題もある. また, 最適化を行わず, 単純にマッチングコストの最小となる場合を解とするため, 模様のない平面部分などではスパイクノイズやスペckルノイズが発生しやすい. なお, ステレオマッチングで推定できないオクルージョン部分に関しては, 周囲の奥行き値から補間する.

また, 最適化手法との比較を行うため, 最適化手法の中でも比較的高速に動作する Semi-Global Matching (SGM) [16, 28] と Efficient Large-Scale (ELAS) [172] を比較手法とする. SGM はスキャンライン最適化を多方向に行い, 途中の計算過程を保持しない

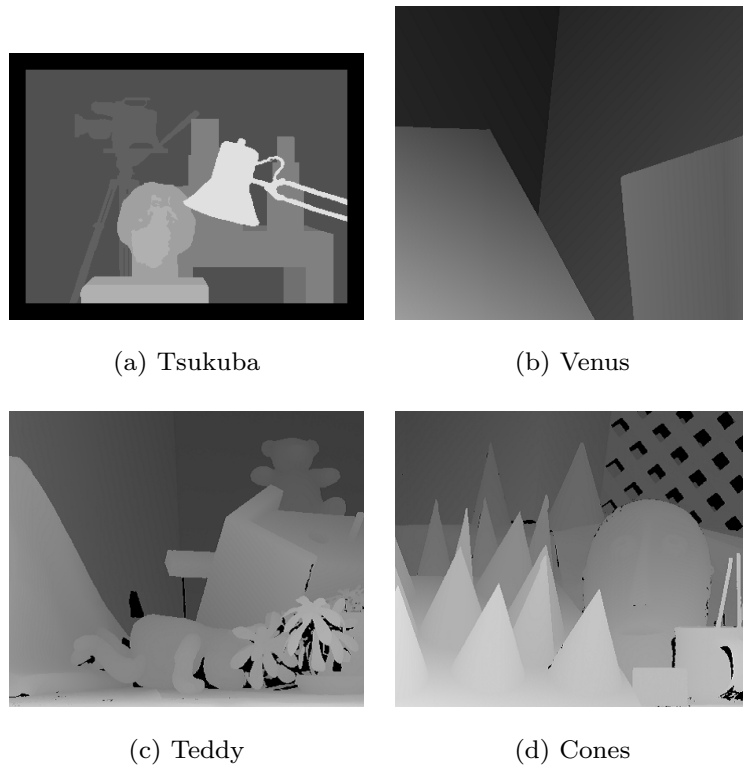


図 2.7 正解デプスマップ.

ことで高速動作を実現する手法であり，ELAS は疎な特徴点をマッチングさせた後に，間を補間する手法である．なお，各手法のパラメータは経験的に決定した．

実験の結果，すべてのデータセットでの誤り率は表 2.2 となった．また，取得したデプスマップは図 2.8 ～ 2.11 となる．まず，表 2.2 の BM と BM with Proposed を比較する．表からわかるように，すべてのデータセットにおいて全領域で改善がみられた．いずれの領域でも誤り率が 1 % 以上減少し，特に奥行き値が大きく変化する輪郭部分である“disc”では大きな改善が確認できる．図を確認しても，ノイズや外れ値が除去されている，または，その領域が小さくなっていることがわかる．加えて，画素値境界の形状を原画像の物体輪郭に近づける輪郭補正の効果も確認できる．しかし，奥行き値が大きく異なる画素値境界ではぼけが発生していることがわかる．これは，奥行き重みや信頼度では除去しきれなかったため，残ってしまったと考えられる．

次に，他の奥行き改善手法と比較する．MF を用いた場合，提案手法と同様にすべてのデータセット，および，その全領域で改善がみられる．また，BF でも，一部誤り率が増加している部分もあるが，概ね改善する傾向にある．しかし，提案手法ほどではなく，提案手法の方が有効であるといえる．図の MF，BF を確認しても，ノイズは除去されているが輪郭は補正されないため，BM で発生した輪郭付近の凹凸が残っている状態である．

一方で、GF, JBF, JTF では、多くのデータセットで悪化傾向にある。GF の劣化具合は他のフィルタに比べても大きい結果となった。また、JBF, JTF は原画像を参照することによる輪郭補正効果があるにも関わらず、“disc” の領域で大きく誤り率が上昇している。それぞれ図を確認すると、GF ではデプスマップ全体がぼけてしまっていることがわかる。また、ノイズ除去の効果も高くはなく、除去というよりも拡散しているため、ノイズの周りの正解領域も悪化させている。JBF と JTF では、平面部分のノイズ除去は行われており、境界の補正も参照画像に合わせて行われているようにみえる。しかし、画素値境界を確認すると、奥行き値が大きく異なる部分で、平滑化による奥行き値の混合によってぼけが発生している。その領域は、奥行き値を重みとして与えられている JTF の方が狭いものの提案手法よりは大きく、奥行き重みを持たない JBF ではさらに広い領域に及んでいる。そのため、“disc” の領域で大きく誤り率が上昇したと考えられる。

そして、提案手法と最適化手法の比較を行う。表より、Teddy を除くすべてのデータセットにおいて、提案手法の方が低い誤り率を示した。図の SGM, ELAS を確認すると、デプスマップ全体を滑らかに接続するように最適化を行う手法であるため、ノイズは BM に比べて少ないことがわかる。しかし、デプスマップ全体を滑らかに接続するため、輪郭の急峻な奥行き変化には対応しづらく、Teddy を除いて提案手法の方が輪郭精度を示す “disc” の値は小さい。Teddy では、画面中央の見たい目は提案手法の方が精度が高く見えるが、画像下部では SGM や ELAS の方が精度が高く見える。これは、元々の奥行き精度が著しく低い BM を提案手法の入力としていることが原因として考えられる。まず、SGM や ELAS が最適化によって、遠くの情報もマッチングに用いることができるのに対して、BM ではブロックサイズ内の情報しかマッチングに用いることができず、そのマッチング精度には大きな差がある。さらに、フィルタもカーネル内の情報しか用いることができないため広い領域の奥行き値を大きく変更することはできない。したがって、このような広く誤った領域では、フィルタを用いたデプスマップ改善では復元できないため、SGM や ELAS を下回る結果となった。

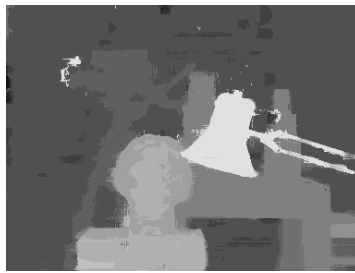
以上から、提案手法は従来の精度改善手法と比較して、“nonocc”, “all”, “disc” のすべての評価領域でより高効率にデプスマップを改善できることを確認した。特に “disc” 領域での改善効果は大きく、奥行き重みや信頼度を用いて輪郭付近でのぼけを抑制することで高効率な改善を可能にした。また、その改善効果は SGM や ELAS といった高速に動作する最適化手法と、一部のデータセットを除き、同等かそれ以上であり、最適化手法の計算コストを考慮すると十分に有効な手法である。一方で、改善効果は精度改善前のデプスマップに依存するため、大きな誤り領域の改善は行えない。加えて、奥行き重みや信頼度によって画素値境界のぼけを抑制しても、完全に取り除くことは難しい。

表 2.2 信頼度を用いたジョイントトリラテラルフィルタと各手法の誤り率の比較.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM	4.60	5.93	17.99	2.08	2.91	16.11	10.63	15.51	26.43	4.97	10.46	14.31
BM with proposed	2.65	3.58	10.73	1.00	1.71	11.92	9.49	14.43	23.26	3.60	9.18	10.58
BM with MF	3.54	4.72	16.96	1.50	2.32	16.12	10.35	15.41	25.96	4.52	9.99	13.16
BM with GF	6.45	8.09	25.05	2.01	3.00	21.88	11.07	16.11	28.58	6.30	11.96	18.47
BM with BF	3.92	5.29	16.73	1.77	2.63	16.90	10.51	15.42	26.21	4.76	10.24	13.91
BM with JBF	4.57	5.82	19.37	2.14	2.98	16.64	11.50	17.17	29.37	6.77	12.37	19.59
BM with JTF	4.25	5.34	18.02	2.73	3.74	26.08	11.45	17.15	29.17	6.54	12.04	18.93
SGM	3.98	5.56	15.47	1.33	2.59	15.23	7.60	14.83	20.90	4.55	11.33	12.81
ELAS	3.99	5.45	18.14	1.84	2.55	20.28	7.99	14.69	22.33	6.85	14.55	17.30



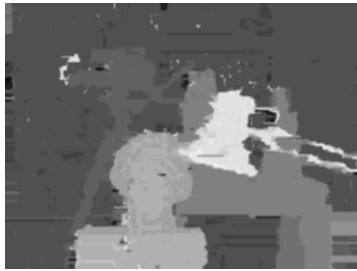
(a) BM



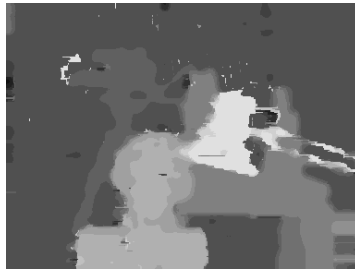
(b) Proposed



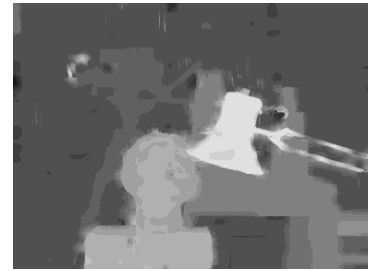
(c) MF



(d) GF



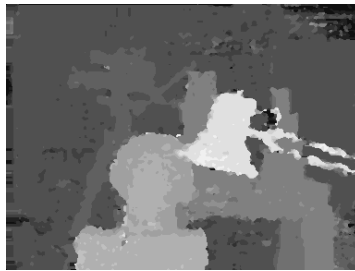
(e) BF



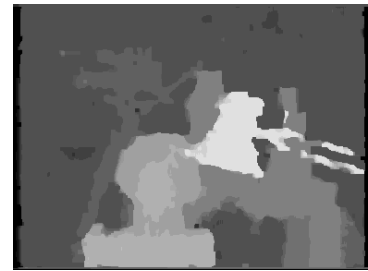
(f) JBF



(g) JTF



(h) SGM



(i) ELAS

図 2.8 信頼度を用いたジョイントトリラテラルフィルタと各手法のデプスマップ : Tsukuba.

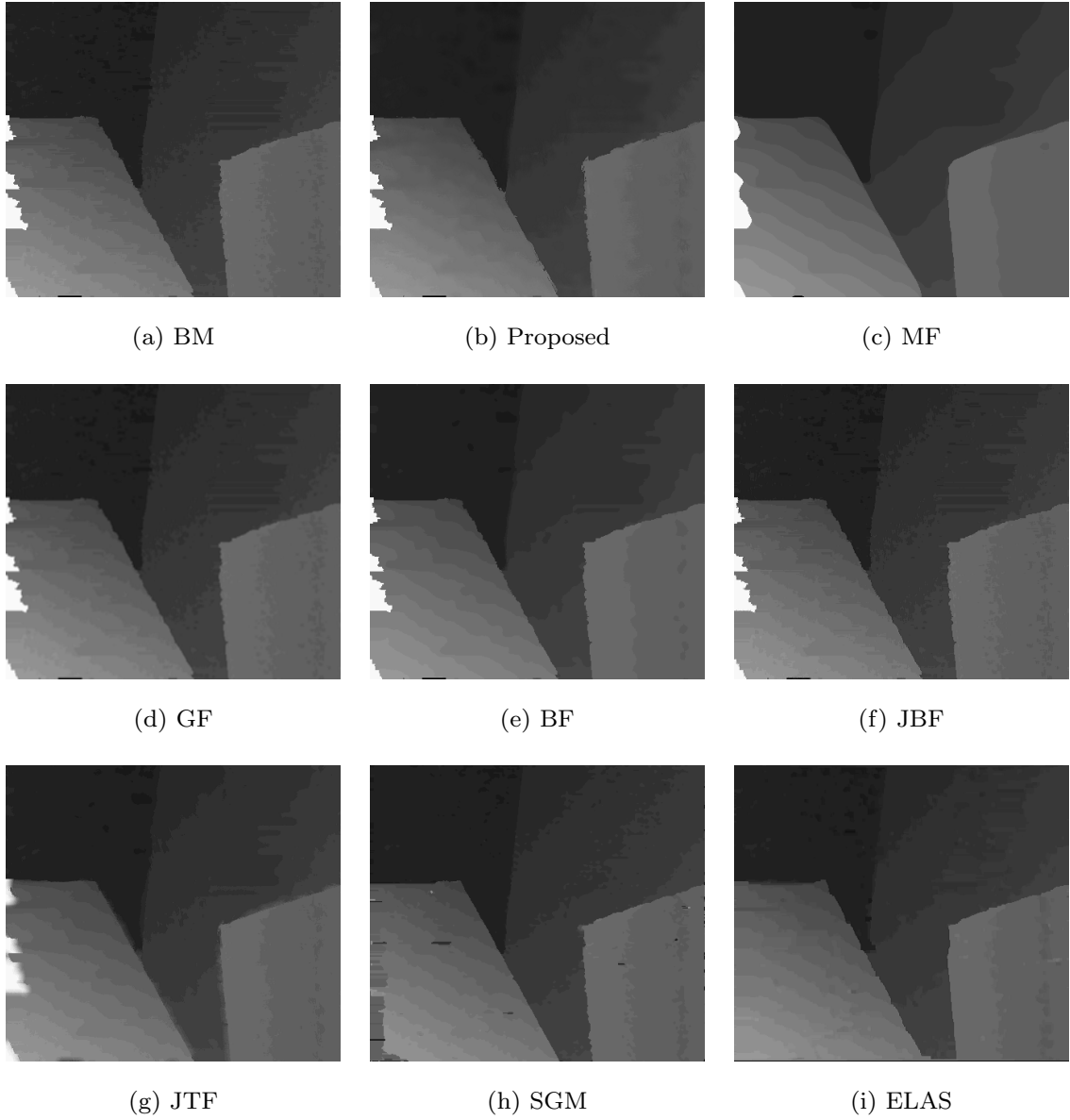


図 2.9 信頼度を用いたジョイントトリラテラルフィルタと各手法のデプスマップ : Venus.

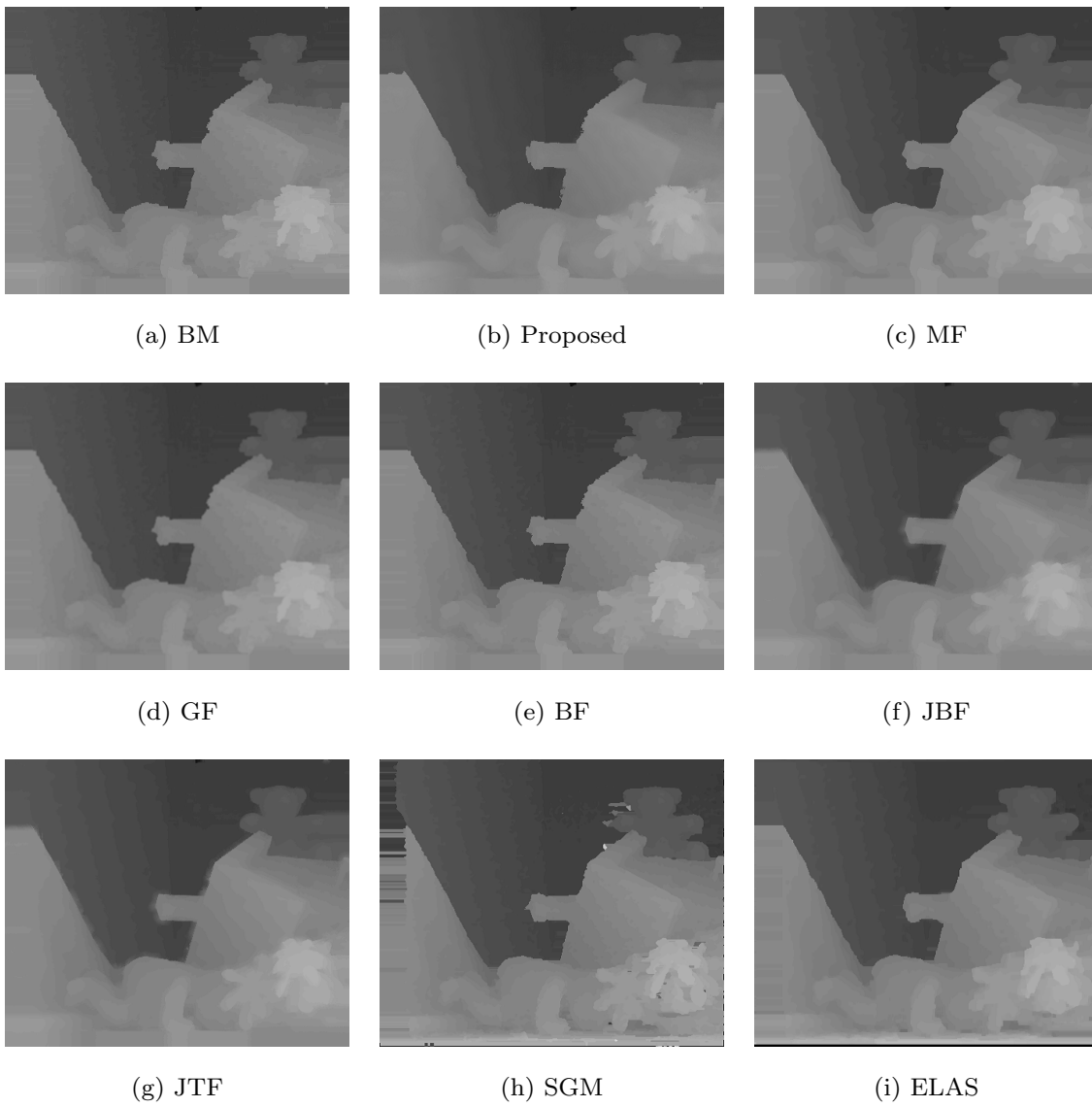


図 2.10 信頼度を用いたジョイントトリラテラルフィルタと各手法のデプスマップ : Teddy.

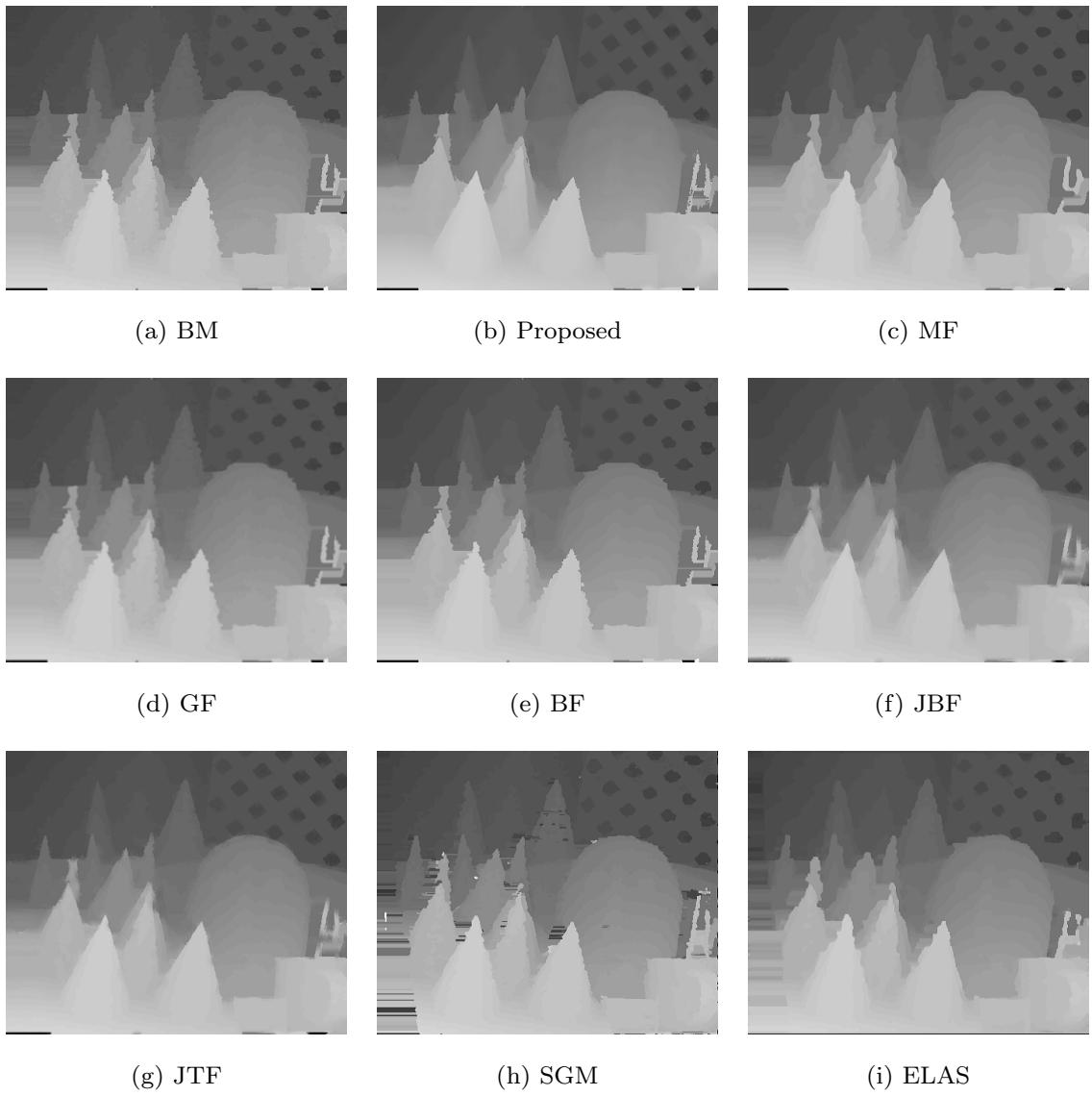


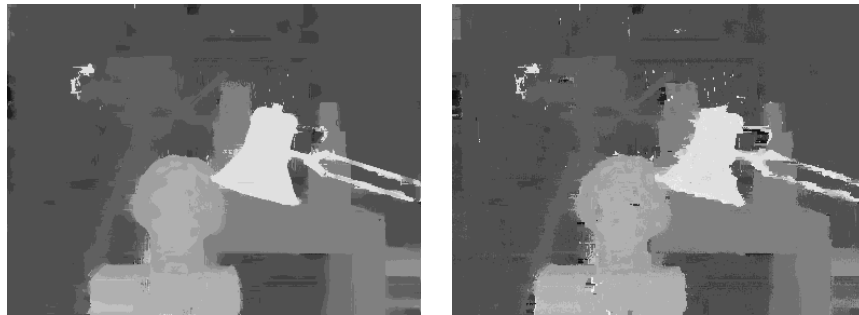
図 2.11 信頼度を用いたジョイントトリラテラルフィルタと各手法のデプスマップ : Cones.

2.4 重み付きジョイントバイラテラルフィルタとジョイントニアレストフィルタ

2.3 節で提案した信頼度を用いたジョイントトリラテラルフィルタを用いることで従来手法よりも高精度にデプスマップを改善することができる。特に、原画像の物体輪郭に対応する部分では、画素値境界が輪郭の形状に補正されるという改善がみられた。しかし、形状の補正はされながらも、同時に画素値境界で奥行き値の混合によるぼけが発生する場合もあった。これは、従来手法であるジョイントバイラテラルフィルタやジョイントトリラテラルフィルタでもみられた問題であり、信頼度を用いたジョイントトリラテラルフィルタでは従来手法よりもぼけが発生する領域は抑制できたものの、完全には抑制できなかった。

そこで、その原因を解析するために、ぼけが発生しにくいようにパラメータ調整を行った。これは、異なる奥行き値の混合を避けるために、奥行き重み関数のガウス変数を小さくすることで調整した。他のパラメータに関しては同様であり、2.3 節の実験で用いたものである。その結果が、図 2.12 である。図から確認できるように、図 2.12(a) の方が図 2.12(b) よりも物体境界がよく維持できている。しかし、図 2.12(a) の方が、図 2.12(b) よりもぼけてしまっている。したがって、2.3 節の信頼度を用いたジョイントトリラテラルフィルタの奥行き重みを用いたぼけ抑制では、ぼけ量と輪郭補正の能力はトレードオフの関係にあるといえる。

そこで、解決すべき課題を、ひとつはノイズ抑制と輪郭補正、もうひとつはぼけ抑制と分け、それぞれに特化した2手法を提案する。ひとつは、信頼度を用いたジョイントトリラテラルフィルタと同じようにノイズ除去と輪郭補正を行い、さらにぼけ量も小さくなることを目指した重み付きジョイントバイラテラルフィルタである。重み付きジョイントバイラテラルフィルタでは、ぼけ量と輪郭補正の効果がトレードオフになる奥行き重みを用いないジョイントバイラテラルフィルタをベースとし、そこにトリラテラル重みで生成される重みマップを追加する。さらに、重みマップの生成では輪郭を識別する閾値と、LR クロスチェックマスクを用いることで、ノイズや輪郭周辺での不要な重み付けを抑制する。もうひとつは、重み付きジョイントバイラテラルフィルタでも発生してしまうわずかなぼけを抑制するジョイントニアレストフィルタである。なお、提案手法は文献 [168–171] を発展させたものである。



(a) 補正効果：大，ぼけ量：大

(b) 補正効果：小，ぼけ量：小

図 2.12 輪郭補正の効果とぼけ量の比較.

2.4.1 提案手法：重み付きジョイントバイラテラルフィルタ

解析の結果を受けて，新たな提案手法である重み付きジョイントバイラテラルフィルタでは，ジョイントバイラテラルフィルタをベースとする．これは，奥行き重みを用いないことで，輪郭補正の効果を高めるためである．ジョイントバイラテラルフィルタでは，色重みの参照画像に原画像を用いることで，デプスマップの輪郭を参照画像の輪郭に補正しつつ，ノイズ抑制が可能である．しかし，参照画像で似た色を示す画素同士では，異なる奥行き値であっても色重みが近い値になるため，異なる物体間で重みの差がつきにくい．したがって，物体輪郭であっても異なる奥行き値が平滑化され，画素値の混色によるぼけが発生する．

理想的には，誤った奥行き値であるノイズを抑制し，物体輪郭上では奥行き値の混色なしに画素値境界の形状を補正したい．そこで，提案手法ではノイズや外れ値とステレオマッチングが誤りやすい輪郭周辺の重みを小さくする重みマップ M をステレオマッチングで取得したデプスマップに基づいて生成する．そして，ジョイントバイラテラルフィルタの重みとして生成した重みマップを追加することで，ノイズ画素の不要な拡散と，物体輪郭での異なる奥行き値の混色を回避する．重み付きジョイントバイラテラルフィルタは次の式で示される．

$$D'_p = \frac{\sum_{s \in N} M_s \cdot w(p, s) c(p, s) D_s}{\sum_{s \in N} M_s \cdot w(p, s) c(p, s)} \quad (2.17)$$

$$w(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) \quad c(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|I_{\mathbf{x}} - I_{\mathbf{y}}\|_2^2}{2\sigma_c^2}\right) \quad (2.18)$$

ここで, D, D' は入力及び出力画像, I は参照画像, \mathbf{p}, \mathbf{s} は処理画素及びの周辺画素の座標, N は周辺画素の集合, w, c は位置重み関数と色重み関数, σ_s, σ_c は位置重み関数及び色重み関数のガウス変数, $\|\cdot\|_2$ は L2 ノルムである.

重み付きジョイントバイラテラルフィルタは, カーネル内の画素の重み付き平均によってノイズ除去を行う. この重みは, 従来のジョイントバイラテラルフィルタと同様に, 処理画素と周辺画素の画素間距離と, 参照画像の画素値差によって決定され, 画素間距離が近く, 参照画像の画素値に近いほど大きな重みが与えられる. さらに, 重みマップ M によって, 位置重みや色重みに関わらず, ノイズや外れ値と輪郭周辺の重みが小さくなるように制御する.

重みマップは, 画素間距離, 周辺画素の画素値差, デプスマップの画素値差に基づくトリラテラル重みを基に生成する. さらに, ステレオマッチングが誤りやすい物体輪郭や奥行き値の画素値境界を識別する閾値と, 取得したデプスマップと左右の原画像からノイズや外れ値を分類する LR クロスチェックマスクを用いる. 閾値処理と LR クロスチェックマスクを合わせて, 閾値付き LR クロスチェックマスクと呼び, 次の仮定をおく. まず, 同一視点の参照画像である原画像とデプスマップにおいて, カーネル内の処理画素と周辺画素の間で物体輪郭をまたがない場合, それぞれの画素は同一物体を指していると考えることができるため, 原画像の画素値とデプスマップの奥行き値はそれぞれ類似すると仮定する. また, ある視点でのデプスマップが与えられたとき, 各画素の奥行き値と, 他の視点との視点間の距離から, 視点間で対応する画素を計算することができる. このとき, 奥行き値が正しければ, 対応する画素間で一貫性が維持できると仮定できる. 以上をまとめると, それぞれ次のように示される.

1. 左視点の参照画像である原画像において, 処理画素 I_p^l と周辺画素 I_s^l の画素値を比較し, その差が閾値 th_c 以下であること.

$$\|I_p^l - I_s^l\|_1 \leq th_c \quad (2.19)$$

2. 左視点のデプスマップにおいて, 処理画素 D_p^l と周辺画素 D_s^l の奥行き値を比較し, その差が閾値 th_d 以下であること.

$$\|D_p^l - D_s^l\|_1 \leq th_d \quad (2.20)$$

3. 左右視点間の距離と, 左視点の周辺画素での奥行き値から, 右視点における周辺画素の対応画素を一意に定められるため, 右視点での原画像の対応画素 $I_{s+D_s^l}^r$ と, 左

視点での原画像の周辺画素 I_s^l の画素値を比較し、その差が閾値 th_{LR} 以下であること。

$$\|I_s^l - I_{s+D_s^l}^r\|_1 \leq th_{LR} \quad (2.21)$$

ここで、 l, r は左及び右視点、 th_d , th_c , th_{LR} はそれぞれ任意の閾値、 $\|\cdot\|_1$ は L1 ノルムである。そして、トリラテラル重みへの閾値付き LR クロスチェックマスクを用いて生成される重みマップ M は、次の 2 式で示される。

$$M_{s'} = \sum_{s' \in N'} b(\mathbf{p}', \mathbf{s}') w(\mathbf{p}', \mathbf{s}') c(\mathbf{p}', \mathbf{s}') \exp\left(-\frac{\|D_{\mathbf{p}'}^l - D_{\mathbf{s}'}^l\|_2^2}{2\sigma_d^2}\right) \quad (2.22)$$

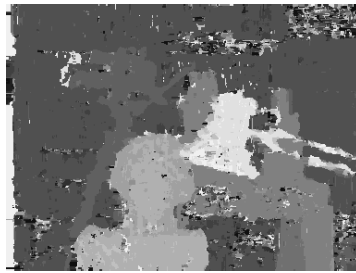
$$b(\mathbf{p}', \mathbf{s}') = \begin{cases} 1 & \text{satisfy conditions} \\ 0 & \text{otherwise} \end{cases} \quad (2.23)$$

ここで、 \mathbf{p}', \mathbf{s}' は処理及び周辺画素の座標、 N' は周辺画素の集合、 σ_d はガウス変数、 b は条件分岐の関数である。なお、式 (2.22) の重みマップを作成する際の、位置重み関数 w と色重み関数 c は重み付きジョイントバイラテラルフィルタと同様に式 (2.18) を用いる。ただし、この際の各重みのガウス変数は個別に設定する。そして、色重みに与えられる参照画像は、デプスマップと同じ視点の原画像である。また、式 (2.23) に与えられる条件は、式 (2.19) ～ (2.21) の閾値付き LR クロスチェックマスクである。

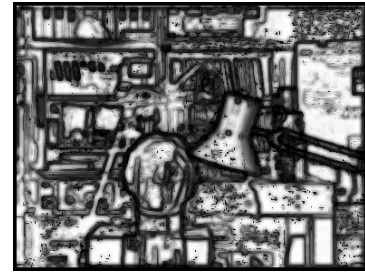
図 2.13 は入力原画像からブロックマッチングで取得したデプスマップに対して生成された重みマップである。重みマップは、色が明るい部分が重みが大きく、色が暗い部分が重みが小さい。図からわかるように、奥行き値の画素値境界で重みが小さくなっていることがわかる。また、ノイズや外れ値に位置する部分の重みも小さくなっている。このような重みマップを用いることで、ノイズやステレオマッチングが不安定な輪郭周辺の影響を抑制することができる。さらに、重みマップにより輪郭周辺の重みを抑制することができるため、ジョイントバイラテラルフィルタの色重みで重みの差がつきにくい原画像の画素値が近い輪郭周辺においても、ぼけの発生を抑制することができる。



(a) 入力原画像 : Tsukuba



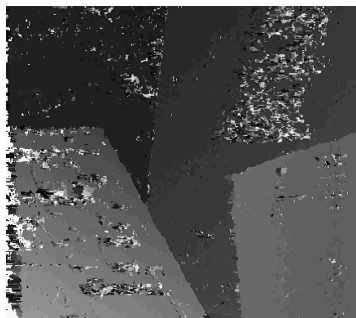
(b) デプスマップ : Tsukuba



(c) 重みマップ : Tsukuba



(d) 入力原画像 : Venus



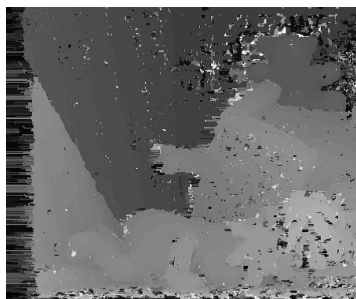
(e) デプスマップ : Venus



(f) 重みマップ : Venus



(g) 入力原画像 : Teddy



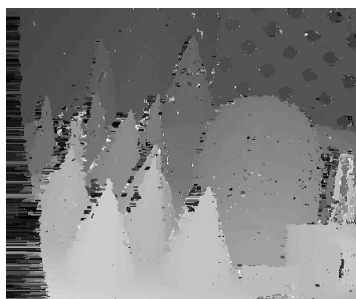
(h) デプスマップ : Teddy



(i) 重みマップ : Teddy



(j) 入力原画像 : Cones



(k) デプスマップ : Cones



(l) 重みマップ : Cones

図 2.13 LR クロスチェックマスクを用いて生成される重みマップ.

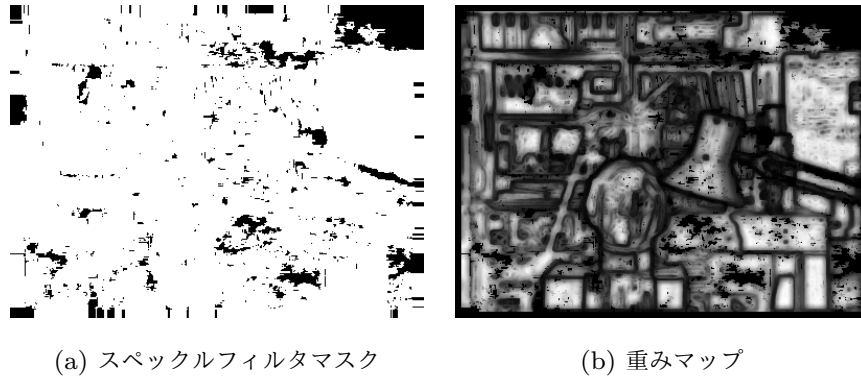


図 2.14 スペックルフィルタマスクを追加して生成される重みマップ。

このほか、式 (2.22) の条件分岐 b に関しては、さまざまな方法で与えることが可能であり、スペックルフィルタを追加で用いる方法も有効である。スペックルフィルタはデプスマップに発生する斑点状のノイズを検出することができる。具体的には、スペックルフィルタは、斑点の最大サイズと、周囲との画素値差閾値をパラメータとして持つ。そして、斑点の最大サイズより小さい画素集合の画素が、その周囲と画素値差閾値以上に離れている場合に斑点状のノイズとみなす。図 2.14 は、スペックルフィルタマスクを追加して生成される重みマップの例である。なお、入力デプスマップは図 2.13 と同様である。スペックルフィルタマスクでは、黒い部分が検出された斑点部分であり、入力デプスマップと確認しても斑点部分を検出できていることがわかる。そして生成された重み画像では、スペックルフィルタによって斑点とされた部分の重みが小さくなっていることがわかる。そのため、スペックルフィルタを用いる場合では、斑点のようなある程度大きな誤りであっても、その誤情報の伝播を防ぐことができる。

2.4.2 提案手法：ジョイントニアレストフィルタ

重み付きジョイントバイラテラルフィルタにより、輪郭部分の精度を改善し平滑化を行う手法を提案した。重み付きジョイントバイラテラルフィルタでは、重みマップを用いることで異なる奥行き値の混合を抑制するが、わずかにぼけが発生する可能性がある。例えば、複数の物体が交差している場所など、複雑で細かく奥行き値が変化する場合では、物体境界の判断が難しく、広範囲にわたり重みマップに与えられる重みが小さくなる。そのような部分では、周囲の異なる奥行き値を示す画素との重みの差が小さくなり、異なる奥行き値の影響を受ける部分も出てくる。さらに、物体がフィルタのカーネルに対して小さい場合、周囲の奥行き値の影響を受ける可能性もある。そこで、わずかに残る輪郭部分のぼけを解消するための処理を施す。

ぼけは重みによって、複数の奥行き値を混合することで発生する。本来、物体境界の奥行き値は前景と背景という二値的な奥行き値で存在するため、混合された中間値はふさわしくない。一方で、フィルタ前の奥行き値は、輪郭の位置は正しくなく、ノイズも含まれるが、混合された中間値は存在せず、正解となる前景、背景の奥行き値を多く含む。そこで、輪郭部分が二値的に表現されるフィルタリング前のデプスマップを用いて、重み付きジョイントバイラテラルフィルタで混合された中間値を前景と背景の奥行き値に振り分ける。その結果、混合部分は存在しなくなることで輪郭部分のぼけが除去され、正しいデプスマップにより近くなる。これをジョイントニアレストフィルタと呼び、次の式で示される。

$$D''_p = \arg \min_{d \in D(W)} |D'_p - D_s| \quad (2.24)$$

ここで、 D'' はジョイントニアレストフィルタの出力画像、 D は精度改善前の入力画像、 D' は重み付きジョイントバイラテラルフィルタの出力画像、 W は周辺画素 s の集合、 $D(W)$ は集合の要素 W が指す奥行き値、 d は取り得る奥行き値である。このフィルタでは、精度改善前の入力画像を参照画像として、重み付きジョイントバイラテラルフィルタで精度改善済みの出力画像と比較する。まず、重み付きジョイントバイラテラルフィルタの出力画像において、ある画素を中心とするカーネルを設定する。このカーネルはステレオマッチングの出力画像でも同位置に設定される。そして、重み付きジョイントバイラテラルフィルタの出力画像のある画素と、ステレオマッチングの出力画像のカーネル内にあるすべての画素を比較し、もっとも奥行き値が似た画素を探し出す。最後に、ある画素は、もともと存在した奥行き値の中で最も近い奥行き値に置き換えられる。つまり、ぼけているが正しい値に近づいたデプスマップのぼけ部分を、ぼけの生じる前の最も近い値に振り分けることでぼけを抑制する。例えば、図 2.15 の場合では、フィルタ後の画素値のうち 1 と 2 はステレオマッチングの出力画像にはない画素値であるため、それぞれ近い値である 0 と 3 に置き換えられる。

以上から、デプスマップを物体輪郭を維持しつつ平滑化を行い、輪郭がぼけた部分に対しては二値的な奥行きに変換する処理を行うことでぼけを低減する。ただし、このジョイントニアレストフィルタにより一部劣化する可能性を持つ場合もある。まず一つ目は、重み付きジョイントバイラテラルフィルタにより生じたぼけと、ノイズの乗った奥行き値の値が偶然にも近かった場合である。二つ目は、隣り合う画素の奥行きがすべて変化するような、非常に急勾配で変化する領域であり、その場合は中間値のほうが望ましい。しかしながら、このような領域は、奥行き推定を行うことも難しいため、正解の奥行き値がほとんど求まらず、そもそも重み付きジョイントバイラテラルフィルタによってデプスマップ

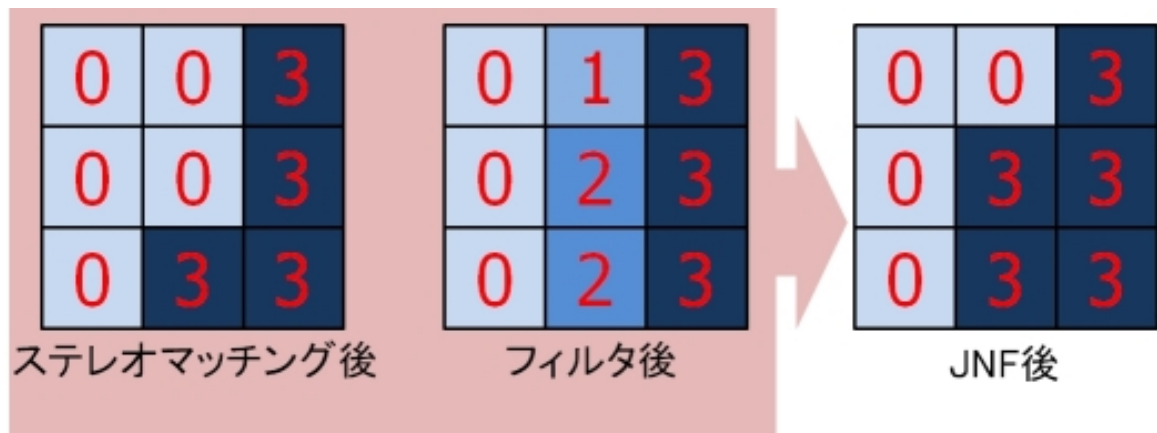


図 2.15 ジョイントニアレストフィルタによる画素値変化.

が改善する可能性は低い．そのため，このジョイントニアレストフィルタにより劣化する領域は非常に少ない．

2.4.3 評価実験

提案手法の有効性を示すための評価実験を行う．2.3 節と同様に，評価は複数のシーンを撮影したステレオデータセットを用いて行う．提案手法の評価では，重み付きジョイントバイラテラルフィルタ (Weighted Joint Bilateral Filter: WJBF) のみを用いた評価と，さらにジョイントニアレストフィルタ (Joint Nearest Filter: JNF) を用いたフィルタセット (Prop.) としての評価を行う．なお，WJBF の重み画像生成方法は，LR クロスチェックマスクを用いた場合と，追加でスペックルフィルタマスクを用いた場合をそれぞれ掲載し，添え字の LR と Spec. はそれぞれ重み画像生成方法を示す．また，精度改善手法として 2.3 節で紹介した従来手法や信頼度を用いたジョイントトリラテラルフィルタ (Reliability based Joint Trilateral Filter: RJTF) とも，一部比較を行う．そして，入力に用いるデプスマップは，ブロックマッチング (BM), Semi-Global Matching (SGM) [16, 28], Efficient LArge-Scale (ELAS) [172] に加えて，動的計画法 (Dynamic Programming: DP) [90], Double Belief Propagation (DBP) [120] の二種を新たに追加して推定する．DP は動的計画法を用いて，画像の横方向の画素を走査線として，各走査線で最適解を解く．また，走査線毎に独立に多項式時間で解くことができるため，SGM や ELAS と同様に高速に動作する最適化手法である．DBP はセグメンテーションと Belief Propagation (BP) による最適化を行う手法である．この手法は，セグメンテーションと BP の各処理を計算するのに多くの時間を要し，実行時間は分単位となる．しかしながら，輪郭の精度も，平面部分の精度も非常に高い手法となる．なお，BM

表 2.3 誤り率：提案手法の効果.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM _h	10.78	12.22	20.03	12.58	13.79	23.06	16.17	24.25	29.89	7.76	17.19	20.01
BM _h with WJBF-LR	6.13	7.33	14.77	10.75	12.07	21.34	16.76	25.10	30.55	6.54	16.44	18.09
BM _h with Prop.-LR	5.43	6.56	13.64	5.83	7.06	13.98	10.77	19.34	22.64	4.09	13.59	11.43
BM _h with WJBF-Spec.	3.39	4.21	13.95	1.69	2.60	13.66	12.18	20.63	24.04	3.94	12.38	11.78
BM _h with Prop.-Spec.	3.25	4.04	13.15	1.36	2.07	9.51	9.20	17.55	20.20	3.03	11.38	9.04
BM _m	7.50	8.75	18.84	5.01	5.96	16.49	12.36	18.10	24.94	5.56	11.54	15.47
BM _m with WJBF-LR	4.38	5.25	14.24	4.07	4.95	11.34	13.89	20.03	25.21	5.99	12.03	16.30
BM _m with Prop.-LR	4.02	4.81	13.08	2.12	2.73	6.39	9.62	15.40	20.34	3.75	9.44	10.75
BM _m with WJBF-Spec.	3.14	3.78	14.59	2.62	3.17	8.52	10.09	15.84	21.84	4.57	10.43	13.67
BM _m with Prop.-Spec.	2.89	3.45	13.29	1.49	1.94	4.22	8.79	14.24	19.22	3.13	8.53	9.21
BM _l	4.60	5.93	17.99	2.08	2.91	16.11	10.63	15.51	26.43	4.97	10.46	14.31
BM _l with WJBF-LR	3.22	4.04	13.10	1.48	2.15	9.94	9.73	14.75	23.58	4.76	10.23	13.85
BM _l with Prop.-LR	2.83	3.59	12.15	1.10	1.60	5.86	9.11	13.90	21.57	3.41	8.60	10.05
BM _l with WJBF-Spec.	2.69	3.20	12.57	1.51	2.14	9.55	9.41	14.41	22.56	4.61	10.08	13.52
BM _l with Prop.-Spec.	2.54	3.25	11.53	1.03	1.50	5.11	8.90	13.71	21.05	3.25	8.43	9.62
SGM	3.98	5.56	15.47	1.33	2.59	15.23	7.60	14.83	20.90	4.55	11.33	12.81
SGM with WJBF-LR	2.94	4.02	12.01	1.11	2.21	9.93	6.98	14.28	18.64	3.78	11.07	11.14
SGM with Prop.-LR	2.73	3.68	10.99	0.68	1.55	6.93	6.18	14.18	16.62	2.62	9.37	7.78
SGM with WJBF-Spec.	2.42	3.09	9.83	0.46	0.97	5.02	6.33	13.41	17.53	3.89	10.22	11.58
SGM with Prop.-Spec.	2.34	2.95	9.42	0.36	0.83	3.73	5.68	12.43	15.42	2.58	8.62	7.68
ELAS	3.99	5.45	18.14	1.84	2.55	20.28	7.99	14.69	22.33	6.85	14.55	17.30
ELAS with WJBF-LR	3.43	4.47	15.43	1.42	1.98	13.01	7.00	13.65	18.59	5.88	13.46	14.17
ELAS with Prop.-LR	3.25	4.21	14.49	1.22	1.57	10.47	6.63	13.05	16.86	4.88	12.38	11.98
ELAS with WJBF-Spec.	2.96	3.68	13.33	0.71	1.18	7.82	6.55	13.23	17.60	5.35	12.80	13.90
ELAS with Prop.-Spec.	2.87	3.54	12.86	0.55	0.81	5.65	6.07	12.48	15.67	4.58	12.04	11.69
DP	4.12	5.04	11.95	10.10	11.03	21.03	14.00	21.58	20.56	10.54	19.10	21.10
DP with WJBF-LR	3.81	4.59	12.04	9.24	10.33	23.41	10.96	19.00	19.77	8.87	17.89	19.39
DP with Prop.-LR	3.37	4.12	11.19	8.40	9.35	19.24	10.32	17.99	17.91	7.53	16.33	16.13
DP with WJBF-Spec.	2.63	3.40	11.12	6.77	7.17	20.65	9.81	17.12	18.44	7.98	15.92	18.49
DP with Prop.-Spec.	2.49	3.22	10.74	6.01	6.36	18.12	9.22	16.53	16.98	7.12	15.64	16.05
DBP	0.88	1.29	4.76	0.13	0.45	1.87	3.53	8.30	9.63	2.90	8.78	7.79
DBP with WJBF-LR	0.86	1.25	4.64	0.15	0.42	1.99	3.60	8.38	9.75	2.96	8.85	8.00
DBP with Prop.-LR	0.85	1.21	4.60	0.12	0.36	1.66	3.72	8.49	9.91	2.89	8.73	7.78
DBP with WJBF-Spec.	0.88	1.31	4.78	0.15	0.39	2.04	3.57	8.37	9.80	2.97	8.74	8.05
DBP with Prop.-Spec.	0.83	1.19	4.78	0.10	0.32	1.44	3.56	8.31	9.69	2.87	8.62	7.74

に関しては、ノイズ量を調整した三種類のデプスマップを用意し、添え字はノイズ量を示し、“h”はノイズ量が多く、“m”はノイズ量の中程度、“l”はノイズ量が少ない状態を示す。さらに、リアルタイムに動くアルゴリズムである、BM、BM とフィルタによる改善、SGM、ELAS と抜粋し、デプスマップの取得時間を計測し、そのリアルタイム性を評価する。

表 2.3 は入力デプスマップに対して提案手法を用いてデプスマップの改善を行った結果である。まず、表より DBP で取得したデプスマップを入力とした場合を除き、すべて

表 2.4 誤り率 : 2.3 節との比較.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM _l	4.60	5.93	17.99	2.08	2.91	16.11	10.63	15.51	26.43	4.97	10.46	14.31
BM _l with WJBF-LR	3.22	4.04	13.10	1.48	2.15	9.94	9.73	14.75	23.58	4.76	10.23	13.85
BM _l with Prop.-LR	2.83	3.59	12.15	1.10	1.60	5.86	9.11	13.90	21.57	3.41	8.60	10.05
BM _l with WJBF-Spec.	2.69	3.20	12.57	1.51	2.14	9.55	9.41	14.41	22.56	4.61	10.08	13.52
BM _l with Prop.-Spec.	2.54	3.25	11.53	1.03	1.50	5.11	8.90	13.71	21.05	3.25	8.43	9.62
BM _l with RJTF	2.65	3.58	10.73	1.00	1.71	11.92	9.49	14.43	23.26	3.60	9.18	10.58

の入力で提案手法が効果を示した．特に，RJTF と同様に “disc” の領域で大きな効果を得ることができている．これは，参照画像の色重みを用いて画素値境界の形状を物体輪郭に近づけ，また，重みマップや JNF によって画素値の混色によるぼけを抑制できたからである．一方，DBP で取得したデプスマップにはほとんど効果がなく，データセット “Teddy” では悪化する結果となった．DBP ではセグメンテーションと高度な最適化を行うため，平面部分と輪郭部分の精度はともに高い．そのため，提案手法の輪郭補正やノイズ除去の効果が表れなかったことが原因の一つと考えられる．したがって，輪郭精度が高いセグメンテーションを用いた手法や，平面の誤りが少ない高度な最適化を行う手法などには，提案手法の適用は適切でないといえる．しかし，SGM や ELAS のように効果が表れている最適化手法もあるため，入力デプスマップの特徴によって適用範囲が限られ，平面部分にノイズの多いデプスマップや境界精度が低いデプスマップに対しては効果があると考えられる．そして，BM のようにノイズ量が多く，境界精度も不十分なデプスマップに対しては大きな効果が期待でき，特に追加でスペックルフィルタマスクを用いる場合には，BM で発生しやすい斑点状のノイズ部分で効果的に重みを抑制できるため，LR クロスチェックマスクのみを用いるよりも高い改善効果が確認できる．これらの手法によって取得したデプスマップは図 2.16 ～ 2.23 となる．なお，提案手法は改善効果の高いスペックルフィルタマスクを追加で用いたものを掲載する．各図の原画像と出力デプスマップを確認しても，デプスマップの画素値境界は原画像の物体輪郭の形状に補正されていることがわかる．また，BM で取得したノイズのデプスマップであっても，ノイズや外れ値を除去できていることが確認でき，数値とともに改善効果が高いことがわかる．そして，数値上効果のなかった DBP では，画像でみても改善を確認することはできない．

また，WJBF と JNF は，それぞれに一定の効果が認められた．ただし，LR クロスチェックマスクのみを用いる場合において，BM でノイズ量を多めに設定した “BMh” や “BMm” を入力とすると一部のデータセットでは WJBF の段階ではわずかな劣化がみられた．これは，ノイズが多い場合ではノイズの領域が広いため，重みマップを作成する際に，ノイズにも重みを与えられている可能性が高い．そのため，この段階ではノイズ除去

表 2.5 実行時間 [ms].

データセット	BM	Proposed	BM+Proposed	SGM	ELAS
Tsukuba	5.7	4.1	9.8	28.8	61.1
Venus	8.4	6.7	15.1	45.9	110.0
Teddy and Cones	10.5	6.8	17.3	71.4	168.2

の機能がうまく働いていない．一方で，スペックルフィルタマスクを追加すると，広い誤りも斑点と認識できたため，WJBF の段階でも劣化はみられなかった．また，ほかに考えられる性能低下の要因として，実験ではフィルタセットで最適となるようにパラメータを設定したことで，WJBF 単体での性能が発揮できなかった可能性もある．したがって，WJBF の効果だけを最大になるように設定すれば，ノイズが多い場合でも一定の効果が得られる可能性もある．

次に，提案手法と RJTF を比較する．2.3 節で入力としたデプスマップは BM で取得した“BMI”と同様であるため，これを比較対象とする．表 2.4 は誤り率の比較である．表より，WJBF と JNF を併用したフィルタセットの方が，多くの場合で RJTF より高精度な改善が行われていることがわかる．しかし，WJBF 単体と RJTF を比較すると，RJTF の方が改善精度が高い場合もある．それぞれの図を確認すると，WJBF の方が RJTF よりも画素値境界でぼけの発生がみられるものの，物体形状はより補正されているようにみえる．これは，WJBF は JNF とともに用いることを前提としており，JNF がぼけを除去できる範囲ではぼけの発生を意識することなく輪郭補正を行うからである．対して，RJTF では，奥行き値を重みとする信頼度を用いてぼけの抑制を行うため，ぼけの発生と輪郭補正の効果はトレードオフにある．そのため，ぼけの発生を抑えると十分な輪郭補正を行うことができない．したがって，ぼけ除去の効果を JNF に任せ，ノイズ除去と輪郭補正を WJBF で最大限に行うことができる 2 手法の併用が実用的であるといえる．

最後に，デプスマップ取得までの実行時間を表 2.5，図 2.24 に示す．ここで，実験環境は Intel Core i7-920 の 2.93GHz である．表 2.5 は，データセットから各手法によりデプスマップを取得するまでに要した計算時間を示しており，BM の結果にデプスマップの改善を行うフィルタを用いる方が，最適化の中でも高速に動作する SGM や奥行きの階調数が多い場合にも高速に動作するとされている ELAS よりも高速である．これは，計算時間のオーダーはフィルタリングである提案手法では $O(H \cdot W)$ であるのに対して，BM，SGM は $O(H \cdot W \cdot D)$ となっている．ここで， H は画像の高さ， W は画像の幅， D は奥行きの階調数である．BM と SGM の視差に応じた計算量の増加を比べると，SGM の増加量は大きい．加えて，この奥行きの階調数は画像サイズに応じて増加していくため，

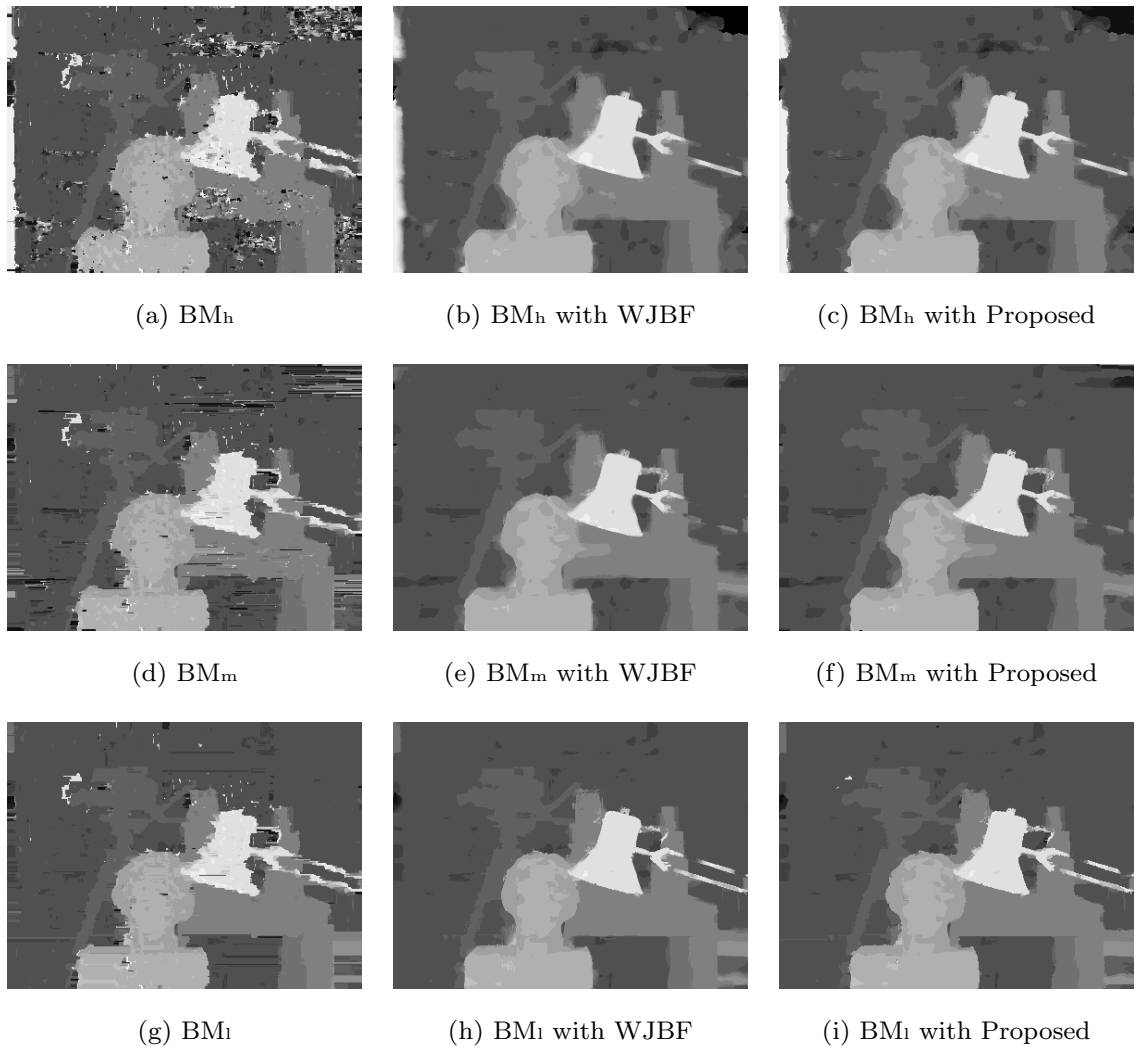
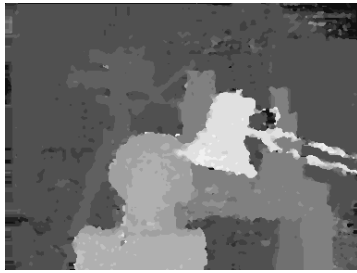


図 2.16 提案フィルタセットによるデプスマップ: Tsukuba (BM 入力).

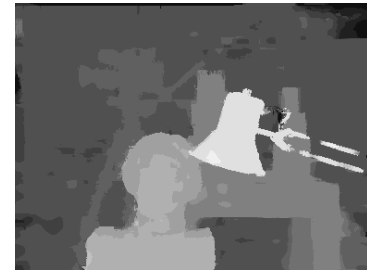
画像サイズが増加するほど，提案手法と最適化手法の実行時間の差は顕著になっていく．また，ELAS は奥行き階調数の影響は比較的少ないが，フィルタリングに比べると重たい処理となっている．図 2.24 は，解像度 128×128 ，奥行きの階調数 16 を基本として解像度，奥行きの階調数を $256 \times 256, 384 \times 384, \dots, 32, 48, \dots$ と n 倍ずつしていったときに，各手法の計算時間をシミュレートした結果を示している．図 2.24(a) は，すべての手法の相対関係を示すために，縦軸のスケールを上限 6000ms で表示している．また，各手法の詳細を示すために，右は上限 400ms で示している．この図から，提案手法のフィルタと他の手法と比べると，解像度や奥行きの階調数が大きくなるほどその差が非常に大きくなることがわかる．また，図 2.24(b) から，解像度や奥行きの階調数が少ない場合には ELAS よりも，SGM のほうが高速であることがわかった．



(a) SGM



(b) SGM with WJBF



(c) SGM with Proposed



(d) ELAS



(e) ELAS with WJBF



(f) ELAS with Proposed



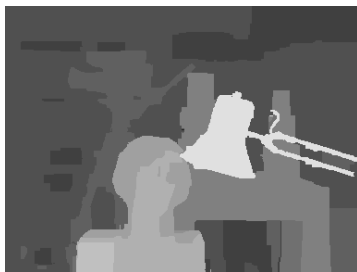
(g) DP



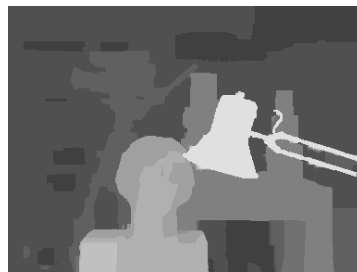
(h) DP with WJBF



(i) DP with Proposed



(j) DBP



(k) DBP with WJBF



(l) DBP with Proposed

図 2.17 提案フィルタセットによるデプスマップ：Tsukuba (その他入力).

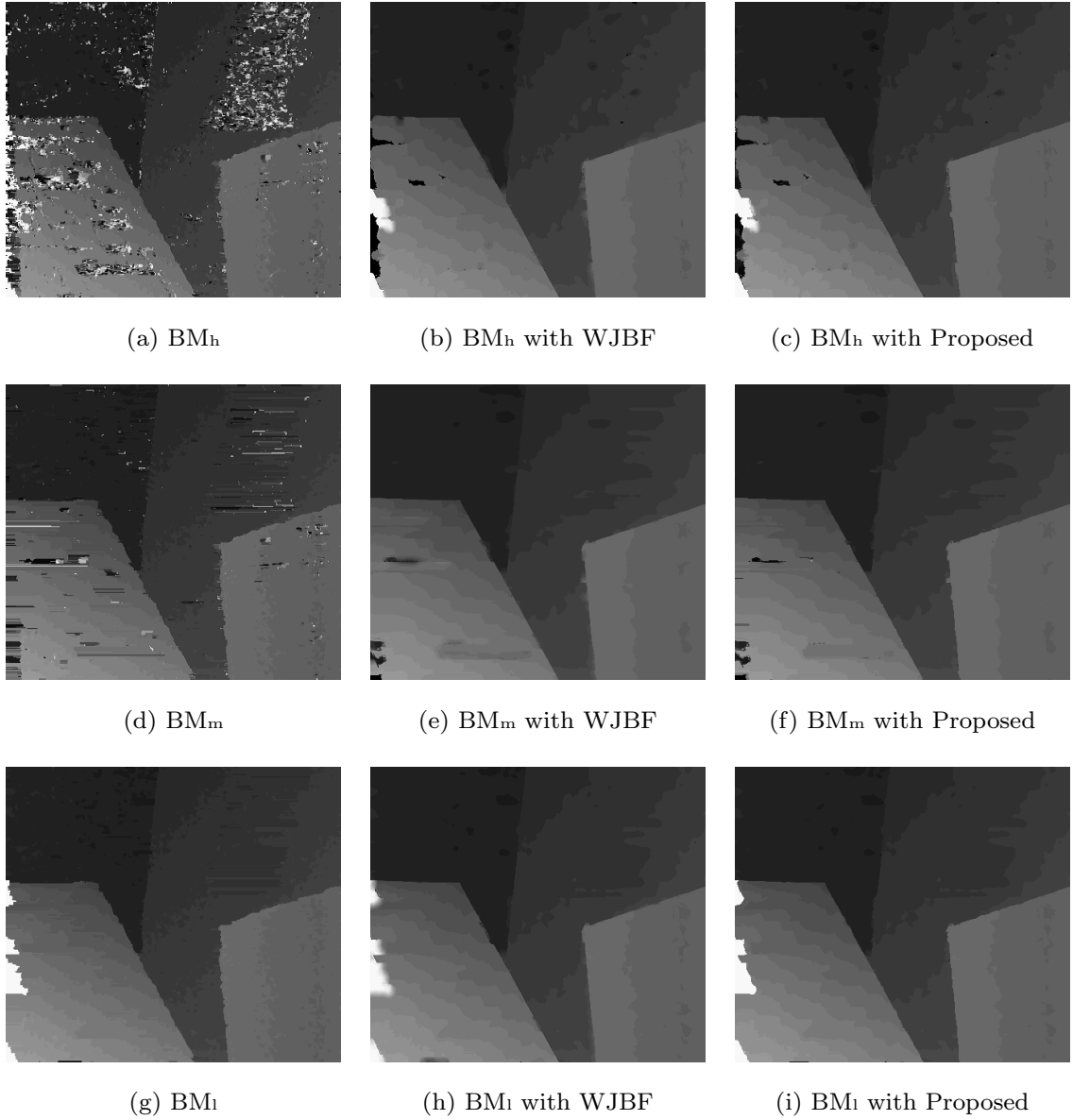
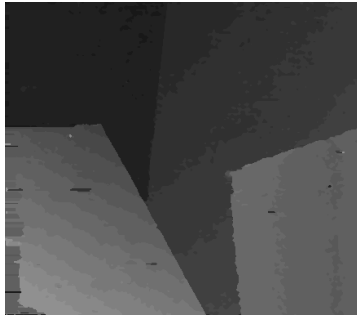
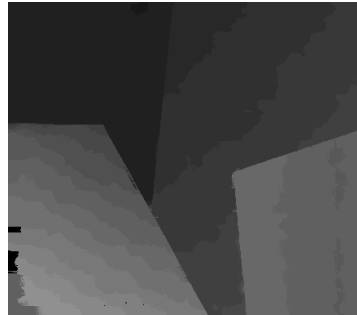


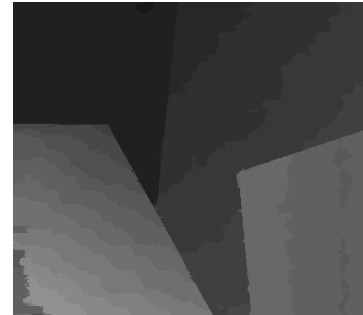
図 2.18 提案フィルタセットによるデプスマップ : Venus (BM 入力).



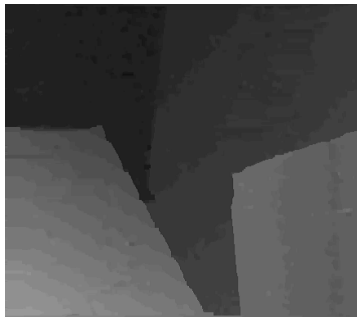
(a) SGM



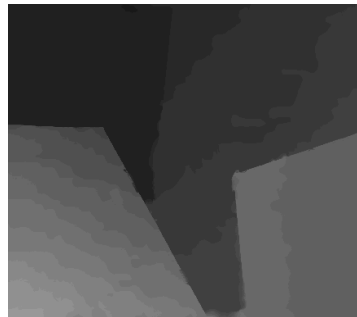
(b) SGM with WJBF



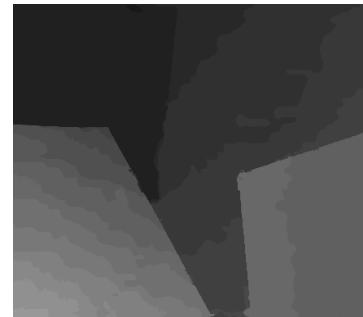
(c) SGM with Proposed



(d) ELAS



(e) ELAS with WJBF



(f) ELAS with Proposed



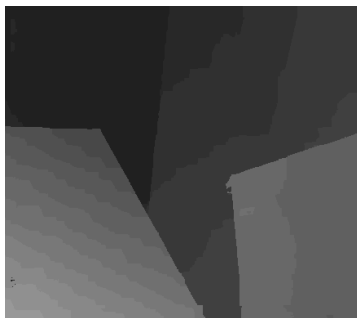
(g) DP



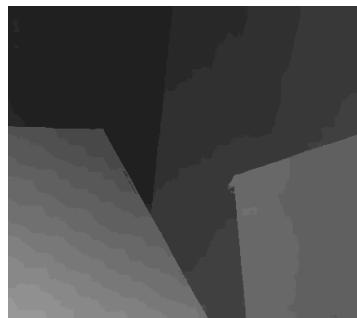
(h) DP with WJBF



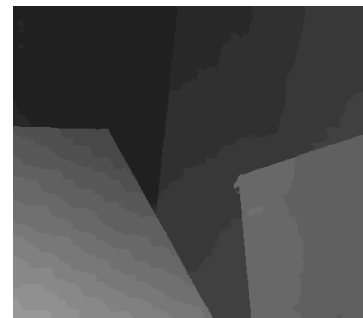
(i) DP with Proposed



(j) DBP



(k) DBP with WJBF



(l) DBP with Proposed

図 2.19 提案フィルタセットによるデプスマップ：Venus (その他入力).

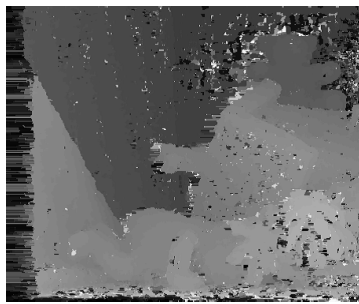
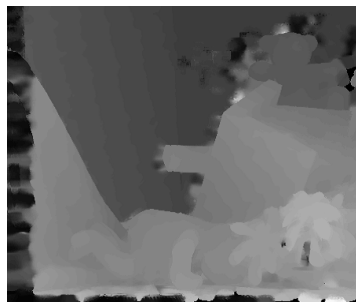
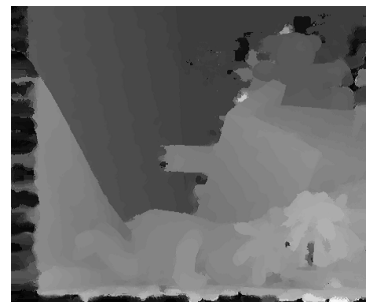
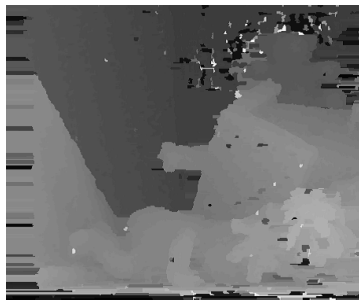
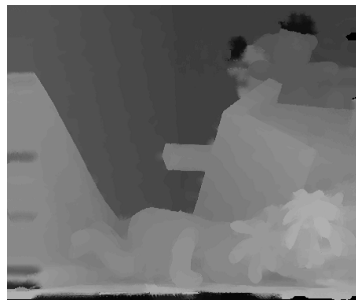
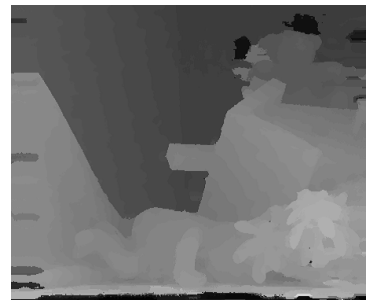
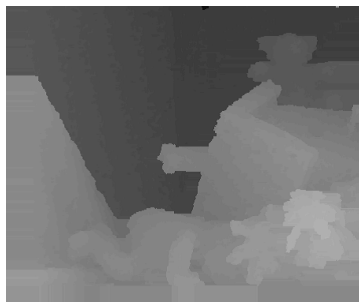
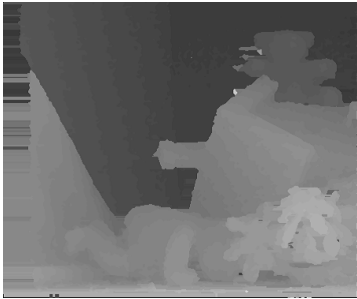
(a) BM_h (b) BM_h with WJBF(c) BM_h with Proposed(d) BM_m (e) BM_m with WJBF(f) BM_m with Proposed(g) BM_l (h) BM_l with WJBF(i) BM_l with Proposed

図 2.20 提案フィルタセットによるデプスマップ : Teddy (BM 入力).



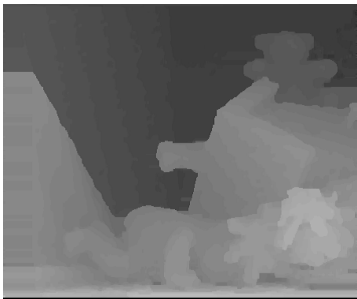
(a) SGM



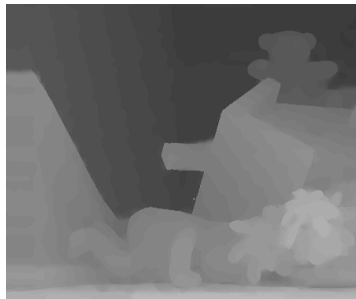
(b) SGM with WJBF



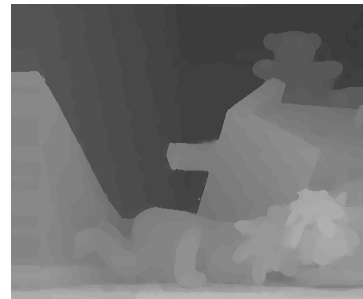
(c) SGM with Proposed



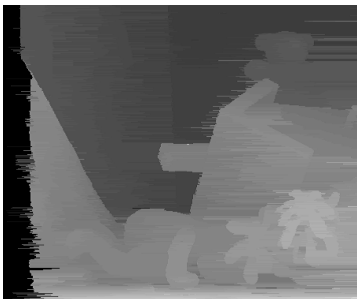
(d) ELAS



(e) ELAS with WJBF



(f) ELAS with Proposed



(g) DP



(h) DP with WJBF



(i) DP with Proposed



(j) DBP



(k) DBP with WJBF



(l) DBP with Proposed

図 2.21 提案フィルタセットによるデプスマップ : Teddy (その他入力).

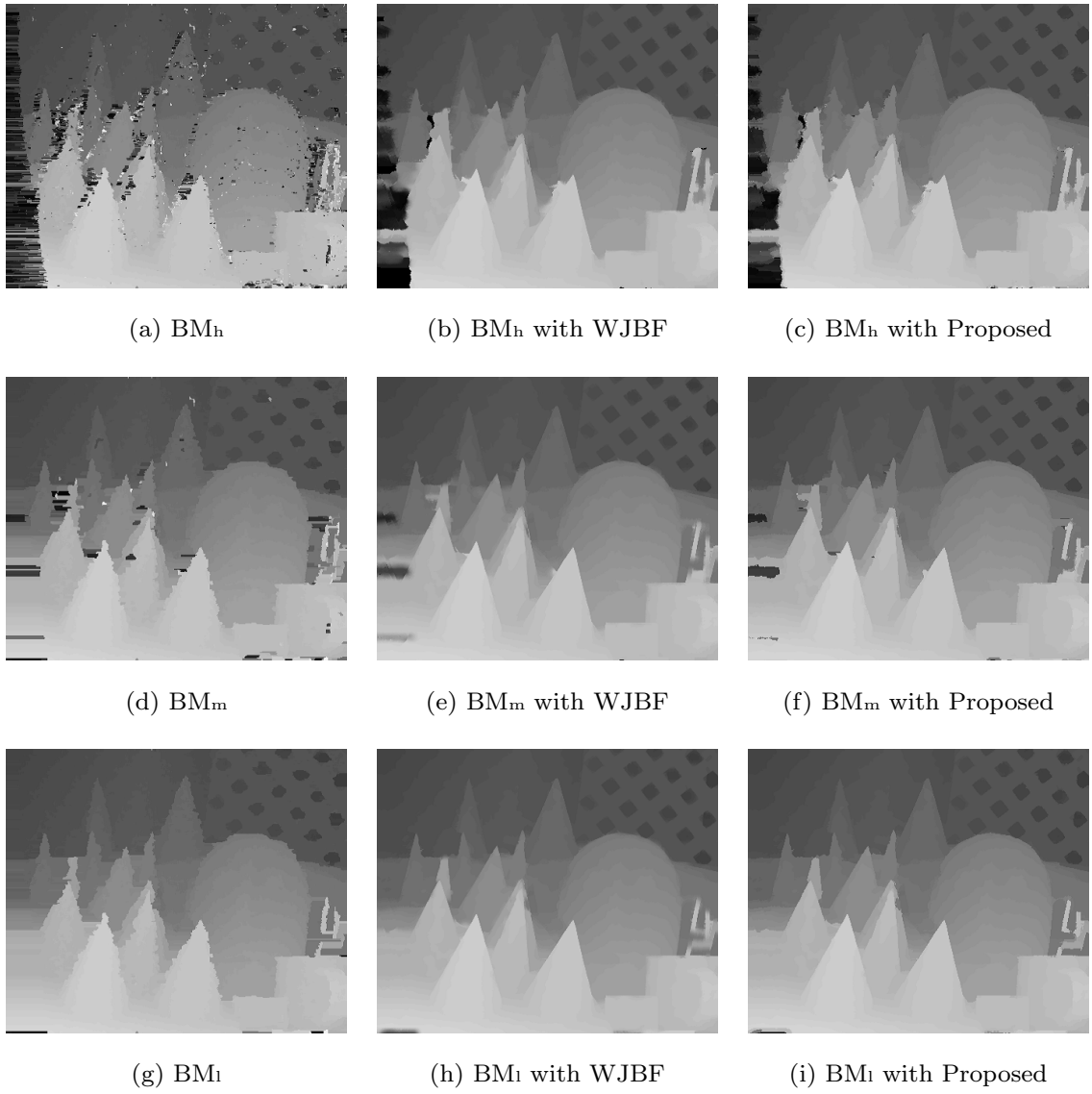
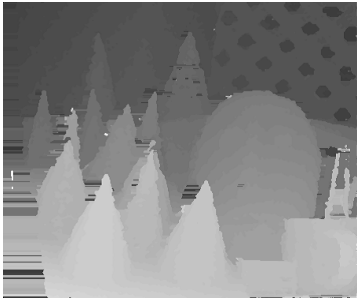
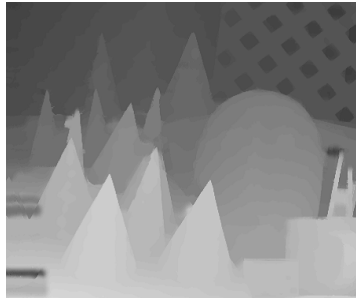


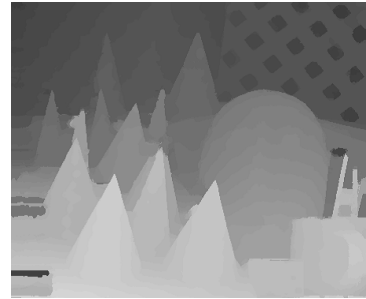
図 2.22 提案フィルタセットによるデプスマップ : Cones (BM 入力).



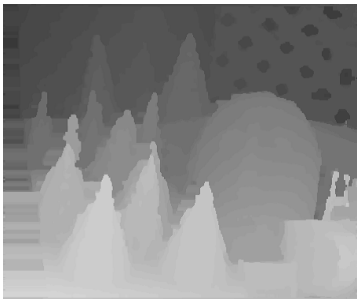
(a) SGM



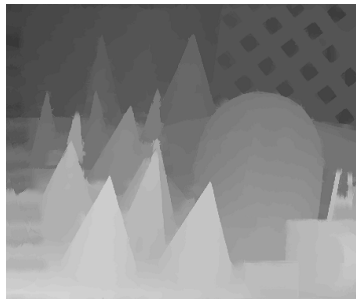
(b) SGM with WJBF



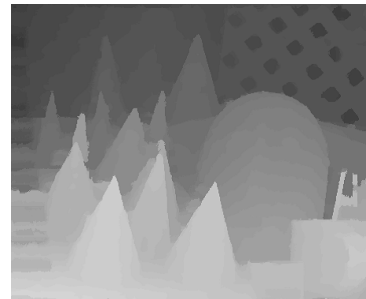
(c) SGM with Proposed



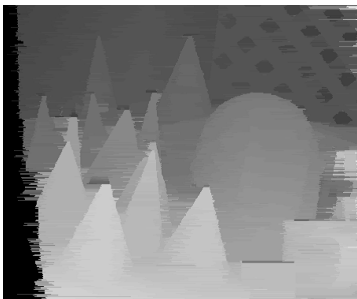
(d) ELAS



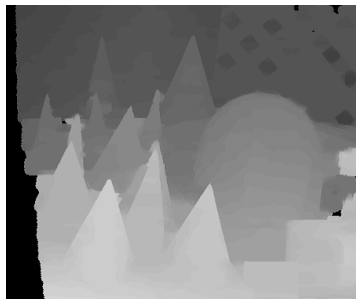
(e) ELAS with WJBF



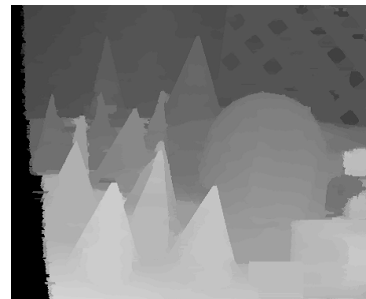
(f) ELAS with Proposed



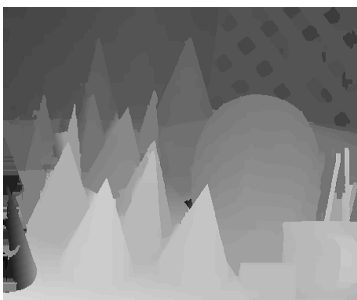
(g) DP



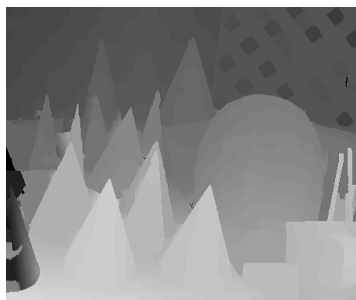
(h) DP with WJBF



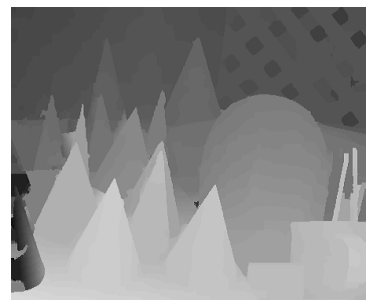
(i) DP with Proposed



(j) DBP

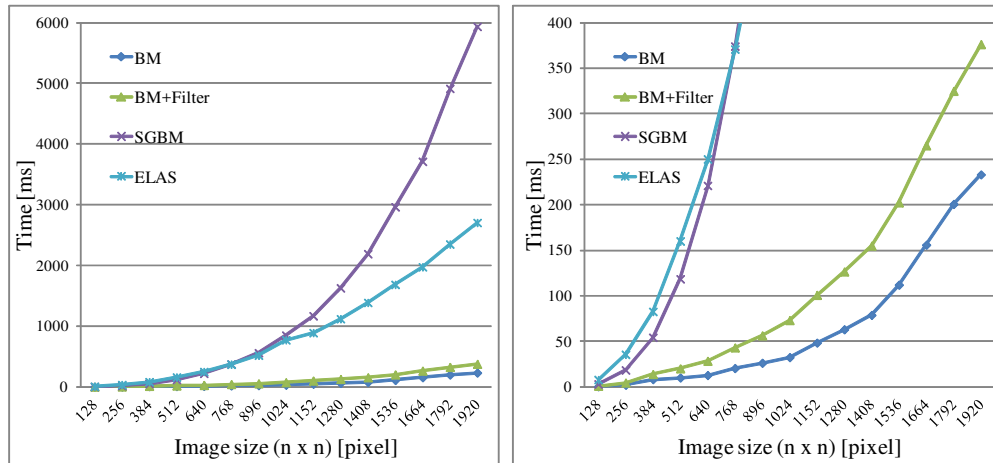


(k) DBP with WJBF



(l) DBP with Proposed

図 2.23 提案フィルタセットによるデプスマップ：Cones (その他入力).



(a) 縦軸 6000ms まで

(b) 縦軸 400ms まで

図 2.24 各手法の実行時間.

2.5 重み付きジョイントバイラテラルフィルタによるコストボリュームフィルタリング

これまでに一度取得したデプスマップに対して、直接フィルタリングを行う精度改善手法を提案した。これらは、いずれも有効な精度改善手法であるが、平滑化処理であるため、原理的に奥行き境界部分でぼけが発生する。また、フィルタリングはフィルタカーネルに閉じた処理であるため、カーネル外の情報を得ることができず、大きなノイズを抑制できない。そこで、本節では取得したデプスマップに直接フィルタリングを行うこれまでの手法と異なり、原理的に奥行き値のぼけが発生せず、大きなノイズに強いという特徴を持つコストボリュームフィルタリングに着目し、重み付きジョイントバイラテラルフィルタの適用を検討する。なお、提案手法は文献 [173–175] を発展させたものである。

2.5.1 提案手法

図 2.25 はコストボリュームフィルタリングの処理フローである。コストボリュームフィルタリングでは、入力のデプスマップを各奥行き値の尤度へと分解しコストボリュームを作成するコストボリューム構築、各ボリュームの尤度を改善するフィルタリング、改善した各ボリュームを再統合しデプスマップを生成するデプスマップ再構成に分かれる [143]。

まず、コストボリューム構築では、取得したデプスマップを参照し、次の式によって奥行き階調 n 毎のコストボリュームスライス V を生成する。

$$V_{\mathbf{p},n} = \min(\|d_n - d_{\mathbf{p}}\|_1, \tau) \quad (n_{min} \leq n \leq n_{max}) \quad (2.25)$$

ここで、 d は奥行き値、 \mathbf{p} は周辺画素位置、 τ は打ち切り変数、 $\|\cdot\|_1$ は L1 ノルム、 n_{min}, n_{max} は取り得る階調の最小と最大である。生成されるコストボリュームスライスは、取り得る各奥行き階調を基準として、各画素の奥行き値が各基準からどれだけ離れた値になるかを示す。したがって、基準から近い奥行き値では尤度が高くなり、基準から離れた奥行き値では尤度は低くなる。

一般に、画像の局所領域においては、同一物体である可能性が高く、同一物体であれば奥行き値も近くなる傾向にある。したがって、ある画素位置で尤度の高い奥行き値は、その周辺画素位置でも尤度が高くなる。しかし、入力 of デプスマップにノイズが乗っている場合では、局所領域においても尤度がばらつく可能性がある。また、境界精度が低いデプスマップでは、必ずしも原画像の物体輪郭に従って画素値境界で尤度の高低が正確に切り替わるとは限らない。そこで、各コストボリュームに対して尤度を改善するフィルタリングを行う。 f をフィルタ、出力されるスライスを V'_n とすると、その過程は次の式になる。なお、すべての周辺画素位置 \mathbf{p} に対して同様の処理を行うため、 V, V' の添え字 \mathbf{p} は縮約する。

$$V'_n = f \circ V_n \quad (n_{min} \leq n \leq n_{max}) \quad (2.26)$$

ここで用いるフィルタは一般的な平滑化フィルタであれば適用可能である。ただし、精度が低いデプスマップが入力されると考えると、ノイズによる局所領域の尤度のばらつきや、原画像の物体輪郭に沿わない尤度の高低差が発生している可能性があるため、原画像の物体輪郭を考慮しつつノイズの抑制を行うフィルタが望ましい。ここでは、2.4 節で提案した重み付きジョイントバイラテラルフィルタを用いる。重み付きジョイントバイラテラルフィルタでは、尤度がばらつく可能性が高いノイズ領域の影響をあまり受けずにコストボリュームスライスの平滑化を行うことができる。また、原画像を参照したフィルタリングを行うため、画素値境界の尤度を原画像の物体輪郭に沿って補正することができる。

フィルタリングによって改善されたコストボリュームスライスは次の式によりデプスマップ再構成が行われる。

$$d'_{\mathbf{p}} = \arg \min_n (V'_{\mathbf{p},n_{min}}, \dots, V'_{\mathbf{p},n_{max}}) \quad (2.27)$$

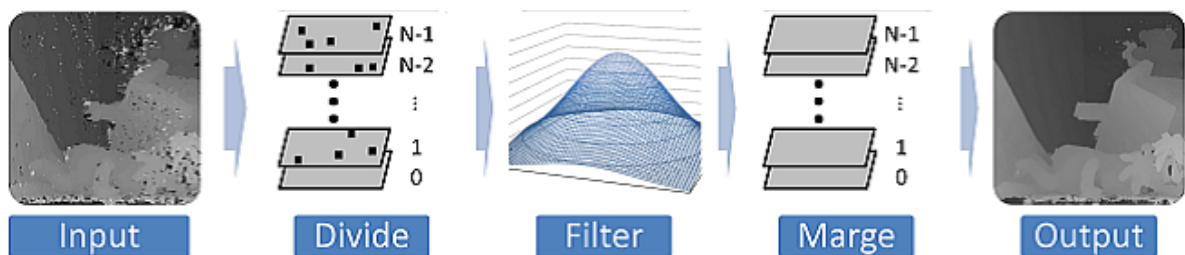


図 2.25 コストボリュームフィルタリングの処理フロー ($N = n_{max} - n_{min} + 1$).

ここで、 d'_p はある画素 p の求める奥行き値である．これによって、画素毎に改善されたコストボリュームを比較し、最小コストを持つ奥行き値によってデプスマップを再構築する．コストボリュームフィルタリングでは平滑化するのは奥行き値の尤度であり、平滑化した尤度から最小コストとなる奥行き値をそれぞれの画素で決定する．そのため、デプスマップの直接フィルタリングでは、特に奥行き値が大きく変化する輪郭部分でぼけが発生したが、コストボリュームフィルタリングでは、輪郭部分のコストの高低ははっきり分離されるため原理的にぼけが発生しない．

2.5.2 評価実験

提案手法の有効性を示すための評価実験を行う．2.3, 2.4 節と同様に、評価は複数のシーンを撮影したステレオデータセットを用いて行う．コストボリュームフィルタリングに用いるフィルタとしては、重み付きジョイントバイラテラルフィルタ (WJBF), ガウシアンフィルタ (GF), ジョイントバイラテラルフィルタ (JBF) をそれぞれ用いた．そして、入力デプスマップは、ブロックマッチング (BM), Semi-Global Matching (SGM) [16, 28], Adaptive Support Weight (ASW) [43, 44] とした．ASW は JBF を用いてコスト集約を行う手法で、ノイズが少なく境界精度も高い局所的な手法である．なお、BM は 2.4 節の最もノイズが多い場合と同様である．

表 2.6 は入力デプスマップに対してコストボリュームフィルタリングを行った結果である．そして、表 2.7 は WJBF を用いたコストボリュームフィルタリングと直接フィルタリングを比較した結果である．また、取得したデプスマップは図 2.26 ～ 2.29 となる．

まず、表 2.6 より、いずれのフィルタをコストボリュームフィルタリングに適用しても、ほとんどの場合でステレオマッチングで取得したデプスマップに比べて精度が向上していることが確認できる．したがって、コストボリュームフィルタはデプスマップの精度改善に有効な手法であることが確認できる．その中でも WJBF を適用する場合では、従来の GF や JBF に比べて改善効果が高い．特に、ノイズが多い BM や SGM を入力デプス

表 2.6 誤り率：コストボリュームフィルタリングの効果.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM	10.78	12.22	20.03	12.58	13.79	23.06	16.17	24.25	29.89	7.76	17.19	20.01
BM with WJBF	2.92	3.21	11.30	0.94	1.61	7.62	6.89	14.35	18.00	2.95	10.67	8.70
BM with GF	6.11	7.13	18.28	4.05	5.27	18.80	9.06	17.15	23.80	3.98	12.52	11.67
BM with JBF	6.17	7.23	14.52	4.88	5.60	16.03	8.48	16.72	21.12	3.45	12.03	9.92
SGM	3.98	5.56	15.47	1.33	2.59	15.23	7.60	14.83	20.90	4.55	11.33	12.81
SGM with WJBF	2.85	3.18	11.05	0.35	0.95	3.98	5.79	12.34	15.85	2.61	8.75	7.83
SGM with GF	3.80	5.03	15.01	1.10	2.17	13.72	6.98	14.51	19.77	3.63	10.02	10.92
SGM with JBF	3.40	4.37	12.18	0.90	1.83	10.51	6.04	13.10	16.73	2.98	9.28	8.85
ASW	1.38	1.85	6.90	0.71	1.19	6.13	7.88	13.30	18.60	3.97	9.79	8.26
ASW with WJBF	1.60	1.91	7.50	0.40	0.63	3.80	7.52	12.53	18.20	3.71	9.37	7.75
ASW with GF	1.38	1.85	6.90	0.58	1.03	6.33	7.55	12.69	18.15	3.71	9.26	7.90
ASW with JBF	1.63	2.04	7.79	0.48	0.87	4.25	7.54	12.65	18.08	3.71	9.23	7.78

表 2.7 誤り率：コストボリュームフィルタリングと直接フィルタリングの比較.

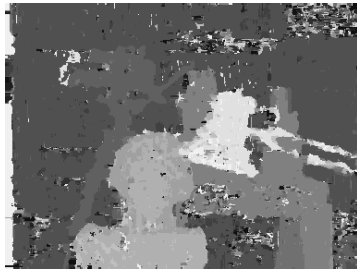
	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM _h	10.78	12.22	20.03	12.58	13.79	23.06	16.17	24.25	29.89	7.76	17.19	20.01
BM _h with WJBF (CVF)	2.92	3.21	11.30	0.94	1.61	7.62	6.89	14.35	18.00	2.95	10.67	8.70
BM _h with WJBF (Direct)	6.13	7.33	14.77	10.75	12.07	21.34	16.76	25.10	30.55	6.54	16.44	18.09
BM _h with WJBF+JNF (Direct)	5.43	6.56	13.64	5.83	7.06	13.98	10.77	19.34	22.64	4.09	13.59	11.43
SGM	3.98	5.56	15.47	1.33	2.59	15.23	7.60	14.83	20.90	4.55	11.33	12.81
SGM with WJBF (CVF)	2.85	3.18	11.05	0.35	0.95	3.98	5.79	12.34	15.85	2.61	8.75	7.83
SGM with WJBF (Direct)	2.94	4.02	12.01	1.11	2.21	9.93	6.98	14.28	18.64	3.78	11.07	11.14
SGM with WJBF+JNF (Direct)	2.73	3.68	10.99	0.68	1.55	6.93	6.18	14.18	16.62	2.62	9.37	7.78

マップとしたときに他手法との差が顕著に表れている。これは、重み付きジョイントバイラテラルフィルタが他の手法に比べてノイズ除去効果や輪郭補正効果が高いためだと考えられる。一方、ASW を入力したときには各フィルタでほとんど性能差がない。これは、ASW はノイズが少なく輪郭も正確であるため生成されるコストボリュームの尤度は各コストスライスで安定しており、元々フィルタによる改善効果が少なかったと考えられる。したがって、どのフィルタを適用しても効果に差が出ない結果となったと考えられる。

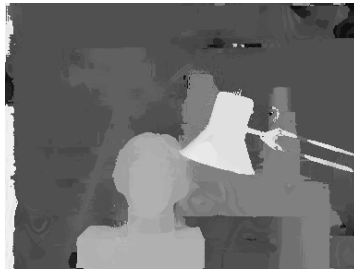
次に、表 2.7 よりコストボリュームフィルタリングと直接フィルタリングの結果を比較する。なお、表ではコストボリュームフィルタリングの結果を CVF と表記し、直接フィルタリングの結果を Direct と表記している。また、直接フィルタリングには、WJBF 単体とジョイントニアレストフィルタ (JNF) を併用した結果を示す。表から、BM を入力とした場合では、コストボリュームフィルタリングの方が直接フィルタリングより、いずれの評価領域でも高精度な改善になっていることが確認できる。これは、BM を入力と

した場合には、大きな誤り領域が多いからだと考えられる。コストボリュームフィルタリングは、直接フィルタリングより大きな誤り領域の精度改善を行う性能が高い。したがって、精度の低い BM を入力とした場合には、コストボリュームフィルタリングの方が改善性能が高い結果となった。一方、SGM を入力とした場合には、コストボリュームフィルタリングと直接フィルタリングの性能差は小さくなり、特に WJBF と JNF を併用したものでは、その差はわずかである。SGM を入力とすると、BM に比べて大きな誤り領域は少なくなるため、コストボリュームフィルタリングの特徴が発揮されにくい。したがって、直接フィルタの性能と同等程度になったと考えられる。

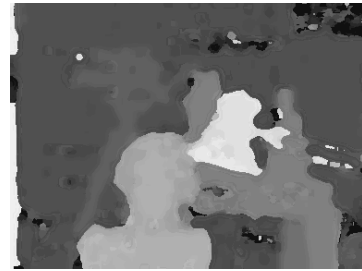
そして、図からコストボリュームフィルタリングの効果を確認する。まず、BM にコストボリュームフィルタリングを適用した各図 (a) ~ (d) について確認すると、いずれのフィルタをコストボリュームフィルタリングに用いてもノイズの除去効果が高く、BM のみの結果よりデプスマップの精度が向上していることが確認できる。しかし、GF を用いた場合には境界精度はあまり向上しないことがわかる。また、JBF も GF に比べれば境界精度が向上しているが十分とはいえない。一方で、WJBF を用いた場合には、他のフィルタと比べて直線の輪郭精度が高く、奥行きが変化する部分のノイズが少ないことが確認できる。これは、WJBF が原画像を参照し、原画像の物体境界を維持したノイズ除去を行う効果である。同様の効果は JBF でも実現できるが、WJBF では重みマップを用いて物体境界やノイズといった奥行き値が不安定な部分の重みを小さくするため、より高精度に輪郭を補正する。また、画素値境界付近のノイズを低減する効果も得ることができる。次に、SGM にコストボリュームフィルタリングを施した各図 (e) ~ (h) について確認すると、ノイズ低減効果は WJBF が他の手法と比べて若干効果が高いが、GF と JBF ではそれほど差はない。これは WJBF は重みマップを用いてノイズを考慮した平滑化を行うことでノイズとなる画素の奥行き値が周囲に伝播しにくい構造になっている一方で、GF や JBF はノイズを考慮しない平滑化手法だからである。さらに、SGM はノイズが発生しにくい最適化手法であるため、GF と JBF のノイズ除去効果にほとんど差が出なかったと考えられる。一方で、物体境界の補正効果は GF, JBF, WJBF の順に高くなる。これは、BM にコストボリュームフィルタリングを行った場合と同様の理由で、JBF や WJBF では原画像を参照画像とし、元の輪郭を維持しながら平滑化を行うためであり、さらに WJBF は重みマップを用いることで奥行き値が不安定な部分の重みを小さくして平滑化できるからである。そして、ASW にコストボリュームフィルタリングを施した各図 (i) ~ (l) について確認すると、いずれのフィルタを用いてもほとんど差がなく、また ASW 単体からの改善効果もほとんどない。これは、ASW ではノイズはほとんど発生せず、物体境界も高精度なコスト集約手法だからである。このように高精度に生成されたデプスマップでは、各領域で尤度は安定しているため、そもそもコストボリュームフィルタリング自体の改善効果が失われる。



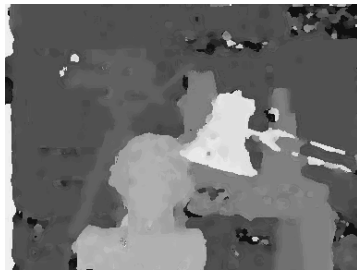
(a) BM



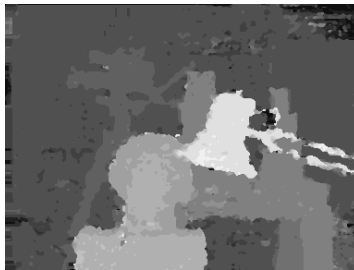
(b) BM with WJBF



(c) BM with GF



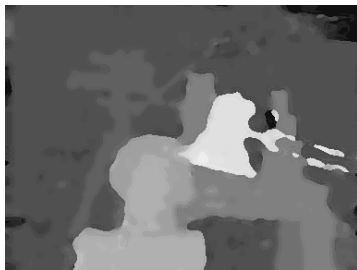
(d) BM with JBF



(e) SGM



(f) SGM with WJBF



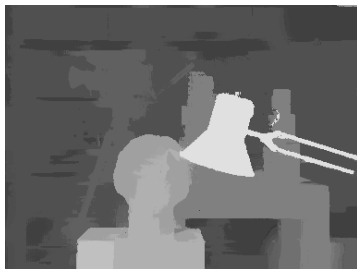
(g) SGM with GF



(h) SGM with JBF



(i) ASW



(j) ASW with WJBF

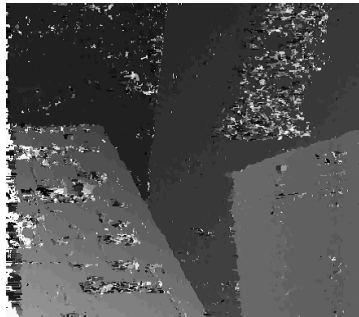


(k) ASW with GF

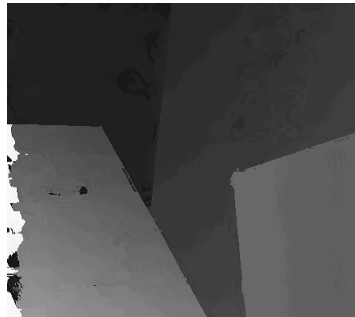


(l) ASW with JBF

図 2.26 コストボリュームフィルタリングによるデプスマップ : Tsukuba.



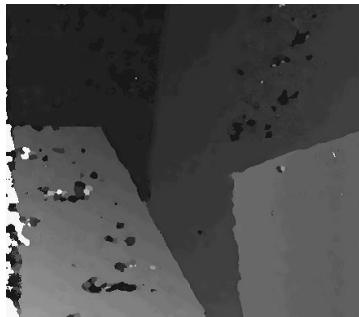
(a) BM



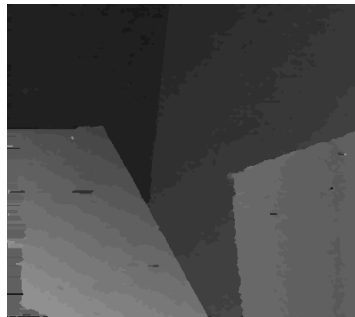
(b) BM with WJBF



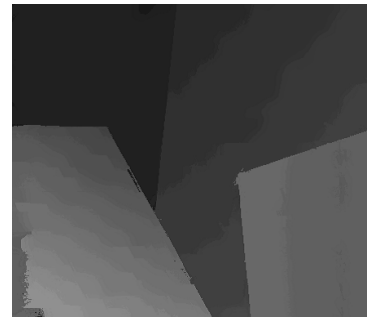
(c) BM with GF



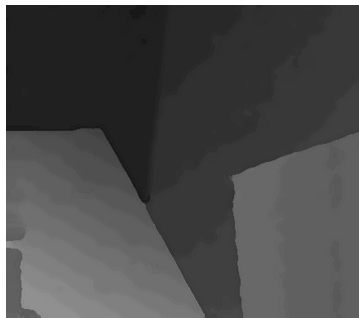
(d) BM with JBF



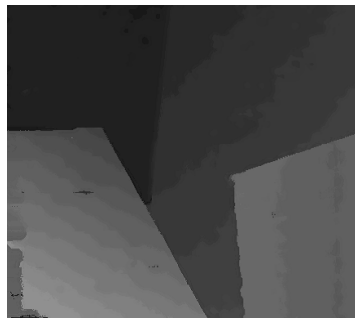
(e) SGM



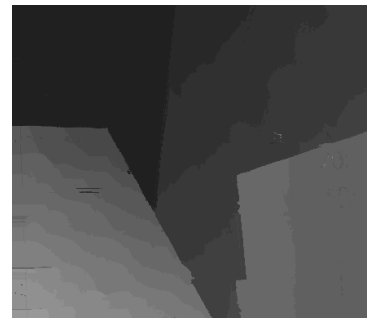
(f) SGM with WJBF



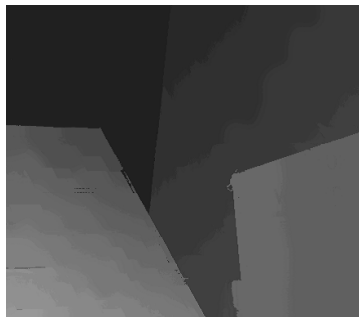
(g) SGM with GF



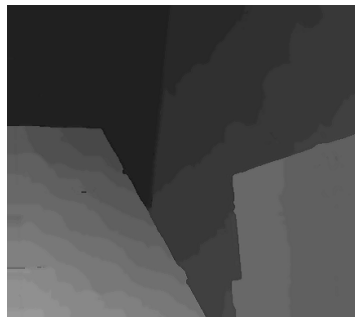
(h) SGM with JBF



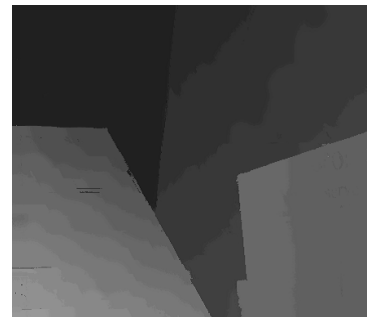
(i) ASW



(j) ASW with WJBF

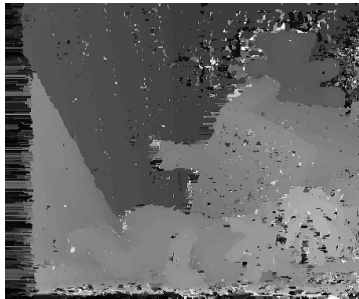


(k) ASW with GF

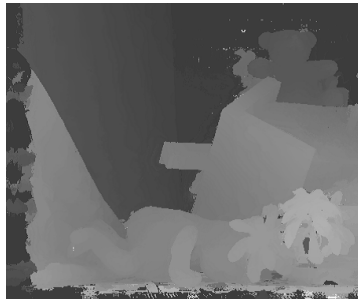


(l) ASW with JBF

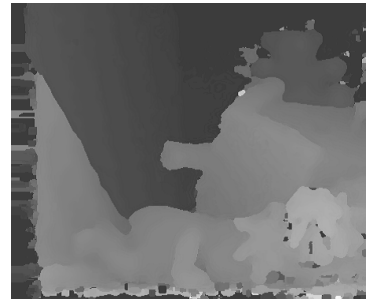
図 2.27 コストボリュームフィルタリングによるデプスマップ : Venus.



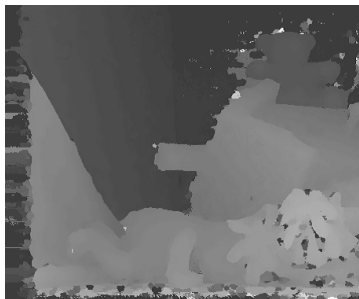
(a) BM



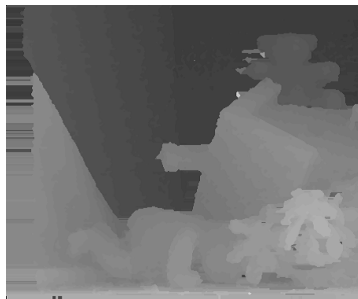
(b) BM with WJBF



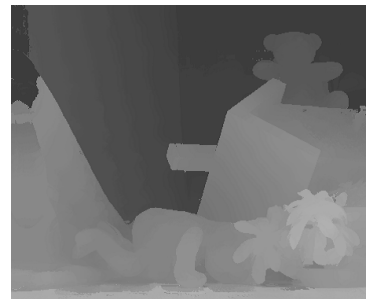
(c) BM with GF



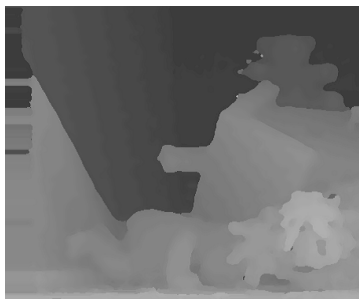
(d) BM with JBF



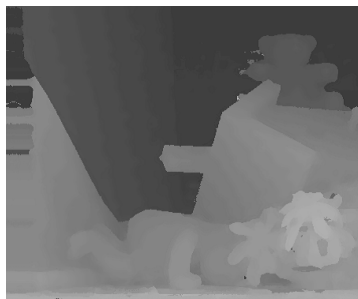
(e) SGM



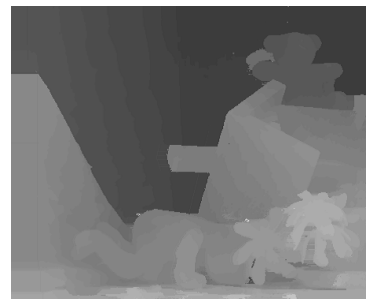
(f) SGM with WJBF



(g) SGM with GF



(h) SGM with JBF



(i) ASW



(j) ASW with WJBF

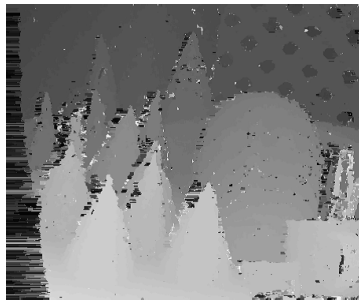


(k) ASW with GF

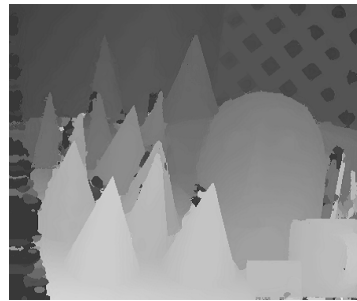


(l) ASW with JBF

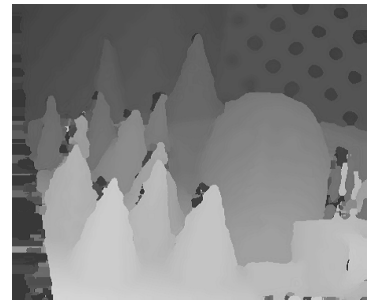
図 2.28 コストボリュームフィルタリングによるデプスマップ : Teddy.



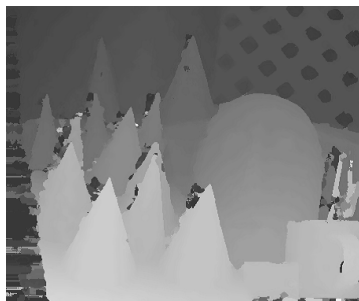
(a) BM



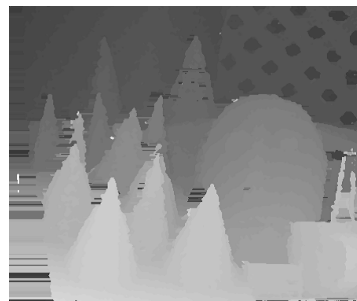
(b) BM with WJBF



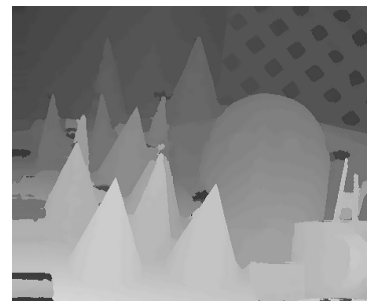
(c) BM with GF



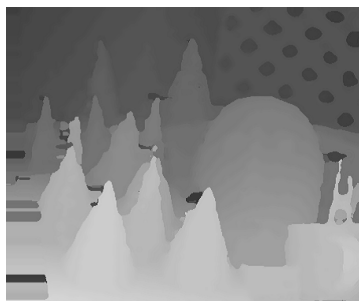
(d) BM with JBF



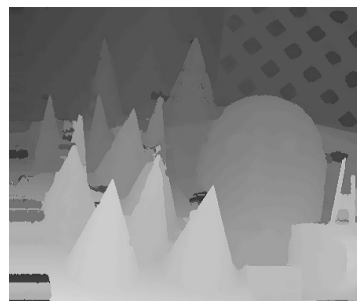
(e) SGM



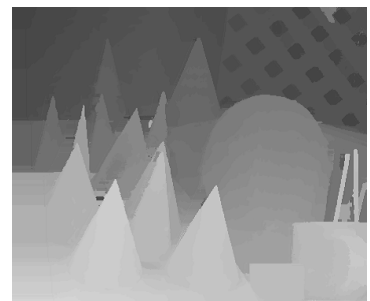
(f) SGM with WJBF



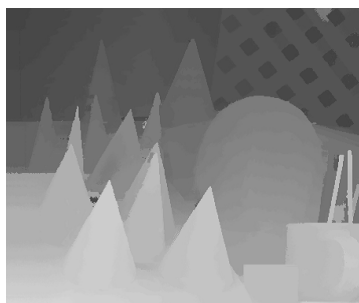
(g) SGM with GF



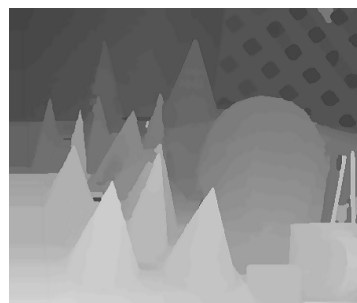
(h) SGM with JBF



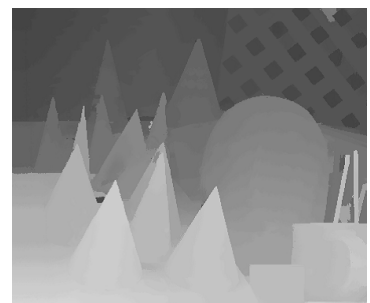
(i) ASW



(j) ASW with WJBF



(k) ASW with GF



(l) ASW with JBF

図 2.29 コストボリュームフィルタリングによるデプスマップ : Cones.

2.6 おわりに

本章では、精度改善のステップにおいて、従来では同時に解決することができなかったノイズ拡散の抑制，輪郭補正，ぼけの抑制の問題を解決する三つのフィルタを提案した．2.3 節では，単体のフィルタとして，画素毎の奥行き値の信頼度と奥行き重みを考慮する信頼度を用いたジョイントトリラテラルフィルタを提案した．2.4 節では，課題を二つに分け，ノイズ抑制と輪郭補正を高精度に行う手法として重み付きジョイントバイラテラルフィルタを，ぼけ抑制を行う手法としてジョイントニアレストフィルタを提案した．2.5 節では，境界付近のぼけが原理的に発生しない枠組みを用いることで，ノイズ抑制と境界精度の改善に重点をおくことができる手法として，重み付きジョイントバイラテラルフィルタによるコストボリュームフィルタリングを提案した．

これらの手法は，ノイズが多く，輪郭精度が低いデプスマップを入力画像したとしても，従来手法よりも高精度なデプスマップの改善が可能である．また，一部データセットを除き，比較的高速に動作する最適化手法である Semi-Global Matching や高精度な特徴点ベースの手法である Efficient Large-Scale よりも高精度なデプスマップが，それらよりもさらに高速に取得可能である．一方で，本章で提案した精度改善手法は局所的な手法であるため，大きな誤り領域を補正しきることはできず，より精度の高いデプスマップを取得するためにはある程度入力デプスマップの推定精度が必要なことや，極めて高精度に取得される最適化手法には効果がほとんどないことが確認された．

また，三つのフィルタの使い分けとしては，デプスマップへの直接フィルタである信頼度を用いたジョイントバイラテラルフィルタと，重み付きジョイントバイラテラルフィルタとジョイントニアレストフィルタのフィルタセットは，ともに高速な手法であるが，フィルタセットでは重みマップの生成とジョイントニアレストフィルタの分だけ処理に時間がかかる．したがって，より高速に精度改善を行いたい場合は信頼度を用いたジョイントバイラテラルフィルタの利用が適しており，より高精度な精度改善を行いたい場合はフィルタセット利用が適している．そして，重み付きジョイントバイラテラルフィルタによるコストボリュームフィルタリングは，デプスマップへの直接フィルタよりは処理時間がかかるが，大きな誤り領域の除去を行うことができる．したがって，直接フィルタでは改善しきれないデプスマップを精度改善のステップで高精度化する際に有効である．

今後の課題としては，入力原画像や改善対象となるデプスマップと各手法のパラメータの関係性を調査し，より有効なパラメータの設定方法や自動設定方法の研究などが考えられる．

第 3 章

ジョイントフィルタを用いたイン ループフィードバックマッチング

3.1 まえがき

高精度なデプスマップを取得する方法としては、古くから提案されている最適化手法が有効である。最適化手法は設定した拘束条件に従う最適解を導出するため、非常に高精度なデプスマップが取得できる。また、高速化の検討もなされており、Semi-Global Matching (SGM) [16, 28] は、途中の計算過程でマッチングコストの総和をとっていくことで、準最適解しか導出できないものの、他の最適化手法よりも高速に動作する。また、近年ではジョイントフィルタを用いたコスト集約手法が登場し、最適化を行わなくても高精度なデプスマップが取得可能になっている。Adaptive Support Weight (ASW) [43, 44] では、ジョイントバイラテラルフィルタ (JBF) [45, 46] を用いて、原画像の物体輪郭を考慮したコスト集約を行うことで高精度なデプスマップを取得することができる。そして、ジョイントフィルタを用いた精度改善手法によっても、高精度なデプスマップの改善が可能である。第 2 章の信頼度を用いたジョイントトリラテラルフィルタや重み付きジョイントバイラテラルフィルタは、デプスマップの誤った奥行き値や輪郭周辺での不確かさを考慮してジョイントフィルタを適用するため、ノイズの拡散やぼけを抑制したフィルタリングが可能である。また、デプスマップへの直接フィルタとして用いれば、取り得るすべての奥行き値を考慮せず、フィルタカーネル内に閉じた計算を行うため非常に高速である。

しかしながら、これらの手法にもそれぞれ問題点があり、最適化手法では高速化の検討がなされているものの、コスト集約手法や精度改善手法に比べて相対的に計算コストが大きい。また、コスト集約手法も、取り得るすべての奥行き値の DSI に対してフィルタリングを行うため計算コストが大きい。そして、精度改善手法では、改善後の精度は入力されるデプスマップの精度に依存するため、大きな誤り領域が存在するデプスマップは直接フィルタリングによっては改善しきれない。さらに、入力されたデプスマップをコスト分解し、各コストスライスに対してフィルタリングするコストボリュームフィルタであれ

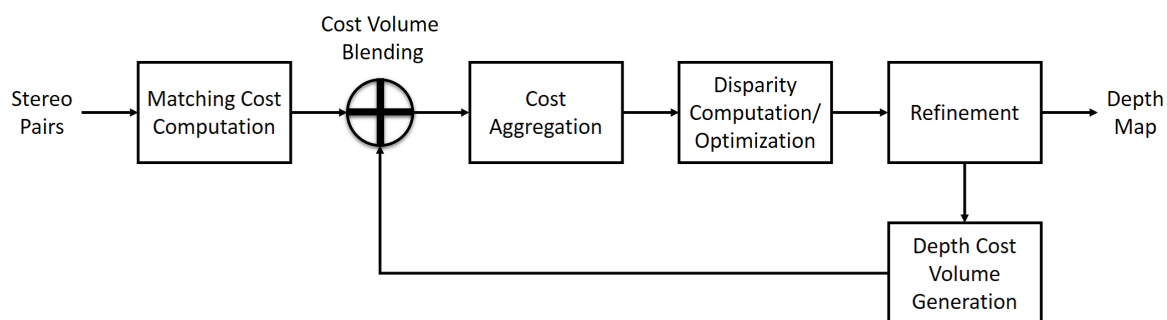


図 3.1 インループフィードバックマッチングの処理フロー。

ば、大きな誤り領域も改善するが、今度はコスト集約手法と同様に計算コストが大きくなる。

そこで本章では、ジョイントフィルタを用いた精度改善による高速で高精度なデプスマップ取得手法をさらに発展させ、精度改善の効果をより活用できる新しいステレオマッチングフレームワークを提案する。インループフィードバックマッチング (In-loop Feedback Matching: IFM) と名付けた提案手法は、一度精度改善を行ったデプスマップをコスト成分に分解し、コスト集約のステップにフィードバックする。そして、マッチングコストとフィードバックされたコスト成分から再度デプスマップを生成し、再び精度改善を行う処理を繰り返す。IFM ではジョイントフィルタによって精度改善したデプスマップをコスト分解して、コスト集約にフィードバックするため、コスト集約にジョイントフィルタを用いた場合と同様の効果が期待でき、デプスマップの輪郭精度を向上させつつ大きな誤りを抑制できる。また、フィードバック 1 回あたりのフィルタリングは、視差計算したデプスマップに対する直接フィルタリングの 1 度で済むため、コスト集約でフィルタ処理するよりも計算コストを小さく抑えることが期待できる。これにより、コスト集約でジョイントフィルタを用いなくても高精度なデプスマップが取得できる。なお、提案手法は文献 [176, 177] を発展させたものである。

3.2 提案手法

図 3.1 は、インループフィードバックマッチング (IFM) の処理フローである。IFM では、これまでのステレオマッチングの処理フローである、マッチングコスト計算、コスト集約、最適化と視差計算、精度改善 [6] に加えて、二つのステップを追加している。それは、コストボリューム構築ステップ (図中, Depth Cost Volume Generation) と、コストボリュームブレンドステップ (図中, Cost Volume Blending) である。IFM はステレオペアを入力し、マッチングコスト計算、コスト集約、最適化と視差計算、精度改善まで

の処理は従来のステレオマッチングと変わらない。従来のステレオマッチングと異なるのは、従来のステレオマッチングの各ステップを処理した後からになる。IFM では、精度改善によって出力されたデプスマップからコストボリュームを生成する。そして、生成したコストボリュームとマッチングコストをブレンドし、ブレンドされたマッチングコストを再度コスト集約以降のステップで再計算する。コストボリューム構築ステップではフィルタなどによって精度が改善されたデプスマップからコストボリュームを生成し、コストボリュームブレンドステップでマッチングコストとのブレンドを行うため、精度改善の結果をコスト集約以降のステップにフィードバックすることができる。このフィードバック機構を繰り返し行うことで高精度なデプスマップを取得する。以降では、各ステップについて詳しく説明する。

マッチングコスト計算，コスト集約と視差計算

ここでは、従来のステレオマッチングと同様の方式をとるマッチングコスト計算，コスト集約，視差計算についてそれぞれ説明する。

マッチングコスト計算では、ステレオペアで画素単位のマッチングコストを計算する。そこでは、絶対値差 (Absolute Difference: AD) や二乗誤差 (Square Difference: SD) が用いられる。これらの AD や SD は、エッジ検出結果や、RANK 変換，CENSUS 変換 [24, 25] の結果とともに用いられることもある。提案手法では、原画像の AD コストとエッジ画像の AD コストとのブレンドを用いている。エッジ画像は X 方向のソーベルフィルタによって計算する。したがって、画素位置 \mathbf{p} の取り得る奥行き値 d に対するマッチングコスト $C(\mathbf{p}, d)$ は次の式によって計算される。

$$C(\mathbf{p}, d) = AD(\mathbf{p}, d) + \alpha \cdot XSobel(\mathbf{p}, d) \quad (3.1)$$

ここで、 $AD(\mathbf{p}, d)$ は入力画像の AD コスト， $XSobel(\mathbf{p}, d)$ はエッジ画像の AD コスト， α はブレンド比で、マッチングコストはすべての画素位置における取り得るすべての奥行き値に対して計算される。

コスト集約ステップでは、計算されたマッチングコストの集約を行う。集約方法は従来のステレオマッチングで既にいくつかの方法が知られており、提案手法ではボックスフィルタを用いてコスト集約を行う。ボックスフィルタを用いたコスト集約は次の式によって計算され、式 3.1 で計算されたマッチングコスト C から集約されたマッチングコスト C_{agr} を得る。

$$C_{agr}(\mathbf{p}, d) = \sum_{\mathbf{q} \in N} C(\mathbf{q}, d) \quad (3.2)$$

ここで、 \mathbf{p}, \mathbf{q} は処理画素及び周辺画素の位置、 N は周辺画素の集合である．従来のステレオマッチングのうち局所的手法に分類されるものは、コスト集約でエッジ保持平滑化フィルタを用いることで、物体輪郭を維持してデプスマップの高精度化を行うことが多い．しかし、ボックスフィルタよりも計算コストが高くなるため、提案手法ではボックスフィルタを用いている．

視差計算では、Winner-Take-All (WTA) を用いてデプスマップを取得する．なお、同時に行われる最適化について、提案手法ではリアルタイム性を考慮して行わない．視差計算は次の式で行われ、各画素に対する奥行き値 d を得る．

$$d_{\mathbf{p}} = \arg \min_d C_{agr}(\mathbf{p}, d) \quad (3.3)$$

この方法は非常にシンプルで高速に動作するが、計算されるデプスマップの精度は高くない．しかしながら、繰り返し処理を行う提案手法のフレームワークでは高速な処理が求められるためこれを用いる．

精度改善

ここでは、取得したデプスマップの精度改善を行う．提案手法の精度改善としては、奥行き値の正確さを判断し有効領域と無効領域に分離する方法と、エッジ保持平滑化フィルタをそれぞれ用いている．

奥行き値の正確さを判断し有効領域と無効領域に分離する方法としては、LR クロスチェックとユニークネスフィルタ [6]、スペックル判定を用いている．LR クロスチェックでは、ステレオペアのデプスマップを用いて、ステレオペアの対応する奥行き値が、定められた閾値以内かどうかで、奥行き値が正しいか否かの判定を行う．もし、定められた閾値以内でなければ、不正確な奥行き値として無効領域にする．なお、通常、LR クロスチェックではステレオペアのデプスマップが必要となるが、提案手法では片方のデプスマップだけで処理する近似版 [14] を用いている．

ユニークネスフィルタでは、コスト集約の結果、発生する曖昧な奥行き値部分を除去する．ここで、曖昧な奥行き値部分とは奥行き値が一定に定まりにくい部分を指し、そのような部分では、マッチングが不安定になる．そこで、ユニークネスフィルタは、マッチングコストの最小コストと二番目に小さいコストを比較する．もし、その違いが閾値以下の場合、曖昧な画素として無効領域に設定する．

スペckル判定では小さな斑点領域を除去する．一般に，原画像にテクスチャのない領域では，どの画素と対応を取ったとしてもマッチングコストが一定，または，近い値になりやすい．そのため，マッチングコストが一意の奥行き値に収束しにくく，誤ったマッチング結果が斑点として出現する傾向がある．斑点になったデプスマップは，本来の正しい姿とかけ離れるため，そのような斑点領域は無効領域とする．

以上の方法によって，無効領域となった部分は，周囲の有効領域から奥行き値を補間することで，正しい奥行き値だけを残す．補間処理では，無効領域から左右に有効領域を探索し，より小さな奥行き値をもつ有効領域の画素値で置換する．

エッジ保持平滑化フィルタとしては，2.4 節で提案した重み付きジョイントバイラテラルフィルタとジョイントニアレストフィルタを用いている．重み付きジョイントバイラテラルフィルタは，画素値境界を原画像に物体輪郭に近づけるように補正を行いつつ，ノイズ抑制を行い，ジョイントニアレストフィルタは重み付きジョイントバイラテラルフィルタによって発生した画素値境界のぼけを抑制する．

コストボリューム構築

ここでは，IFM のフィードバック機構で重要となるコストボリューム構築について説明する．まず，精度改善したデプスマップの情報をマッチングコスト計算にフィードバックするために，精度改善されたデプスマップからコストボリュームを生成する．ここで生成されるコストボリューム V は，コストスライス $V_n (n \in 0, \dots, N-1)$ からなる．なお， N は取り得る奥行き値の階調数である．奥行き値の階調 n のコストスライス V_n における，ある座標 \mathbf{p} のコストは次の式で示されるコスト関数 L_x で生成される．

$$V_{\mathbf{p},n} = L_x(n, l_{\mathbf{p}}, \tau) \quad (0 \leq n \leq N-1) \quad (x \in L1, L2, \exp) \quad (3.4)$$

$$\begin{cases} L_{L1} &:= \frac{1}{\tau} \min(\|n - l_{\mathbf{p}}\|_1, \tau) \\ L_{L2} &:= \frac{1}{\tau^2} \min(\|n - l_{\mathbf{p}}\|_2^2, \tau^2) \\ L_{exp} &:= 1 - \exp(-\frac{\|n - l_{\mathbf{p}}\|_2^2}{2\tau^2}) \end{cases} \quad (3.5)$$

ここで， $\|\cdot\|_1, \|\cdot\|_2, \exp$ はそれぞれ L1 ノルム，L2 ノルム，指数関数， $l_{\mathbf{p}}$ は精度改善された画素 \mathbf{p} の奥行き値， τ は打ち切り閾値である．ここで計算されるコストは，取り得る奥行き値の階調 n と，実際のデプスマップの奥行き値 $l_{\mathbf{p}}$ の差に従って，単調に増加するのが適切であるため，L1 ノルムや L2 ノルム，指数関数を用いる．なお，予備実験において，L2 ノルム関数でコストボリュームを生成するのが最も性能がよかったため，提案手法では L2 ノルム関数を用いている．

コストボリュームブレンド

ここでは、ステレオマッチングの最初のステップで計算された初期のマッチングコストと、構築されたコストボリュームを合成するコストボリュームブレンドについて説明する。このステップでは、次の式に従って、初期のマッチングコストのコストスライス C と、構築したコストボリュームのコストスライス V をブレンド比 α に従ってアルファブレンドする。

$$C'_{\mathbf{p},n} = \alpha \cdot C_{\mathbf{p},n} + (1 - \alpha) \cdot V_{\mathbf{p},n} \quad (0 \leq n \leq N - 1) \quad (3.6)$$

ここで、 C' はブレンド後のマッチングコスト、 \mathbf{p} は画素位置、 n は奥行き値である。初期のマッチングコストは、輪郭やオクルージョン領域などのマッチング難易度が高い領域では、コストが一意的奥行き値に収束せず曖昧になることが多い。一方で、精度改善されたデプスマップから生成されたコストボリュームは、ノイズが除去され、輪郭精度も向上しているため、初期のマッチングコストに比べると、一意的奥行き値に収束しやすくなっている。そこで、提案手法のフィードバック機構によって輪郭やオクルージョン領域の精度が改善されたデプスマップから生成されたコストを、曖昧な初期のマッチングコストにフィードバックすることでマッチングの精度を高めていく。なお、提案手法では、ブレンド比は 1 : 1 としている。

繰り返し処理とポスト処理

IFM では、これまでに説明した各ステップをパラメータを変化させながら繰り返し処理する。例えば、コスト集約ステップでは、ボックスフィルタのカーネルサイズを前のループより小さくなるように処理する。こうすることで、ループの前半では画素の情報をより遠くまで伝播させ、ループの後半では局所領域でコストを安定化させることができる。

また、繰り返し処理を抜ける際の精度改善では、前述した処理に加えてポスト処理を行う。ここでは、サブピクセル補間とバイナリレンジフィルタ処理を行っている [143]。まず、サブピクセル補間に関しては次の式で示される。

$$d'_{\mathbf{p}} = d_{\mathbf{p}} - \frac{C(\mathbf{p}, d_+) - C(\mathbf{p}, d_-)}{2(C(\mathbf{p}, d_+) + C(\mathbf{p}, d_-) - 2C(\mathbf{p}, d))} \quad (3.7)$$

ここで、 \mathbf{p} は処理画素の位置、 d' はサブピクセル補間された奥行き値、 d は入力奥行き値、 d_+, d_- は入力の奥行き値からそれぞれ ± 1 した奥行き値、 C は式 (3.1) のマッチングコスト関数である。バイナリレンジフィルタに関しては、次の式で示される。

$$D_p = \frac{\sum_{q \in N} g(p, q) D_q}{\sum_{q \in N} g(p, q)} \quad (3.8)$$

$$g(p, q) = \begin{cases} 1 & |D_p - D_q| \leq 1 \\ 0 & \text{else} \end{cases} \quad (3.9)$$

これらのポスト処理は，デプスマップのサブピクセル精度を向上させるために行う．通常，ステレオマッチングの計算コストは，処理する奥行き値の階調数に応じて線形に増加する．したがって，ステレオマッチングでサブピクセルまで推定しようとする，特に提案手法のようなループ機構を持つ手法では処理量が大きく増大する．そこで，提案手法ではデプスマップを出力する直前の精度改善で，一度だけサブピクセルの補間処理を行うことで処理コストを低減している．

3.3 評価実験

ここでは，インループフィードバックマッチング (IFM) の有効性を検証するために，デプスマップの推定精度を評価する．評価は，第 2 章と同様に複数のシーンを撮影したステレオデータセットを用いて行う [6]．また，誤り率もこの評価方法も第 2 章と同様である．

また，IFM のパラメータに関して，コスト集約の集約半径は初期値を 5 として，フィードバックを行う毎に小さくしていく．具体的には，フィードバック回数を重ねる毎に，5, 3, 1, 1, 1 とした．また，精度改善に用いる重み付きジョイントバイラテラルフィルタ (WJBF) とジョイントニアレストフィルタ (JNF) のフィルタカーネルサイズはそれぞれ 7×7 , 5×5 とした．ほかのフィルタパラメータについては，各画像について実験的に最適解となる値を導出して適用した．

表 3.1 はフィードバック回数と奥行き推定精度の推移である．なお，比較手法として Semi-Global Matching (SGM) を掲載する．また，Before refinement は精度改善を行わない場合の推定結果であり，After refinement は精度改善を 1 回のみ行ってフィードバックしない場合の結果である．表から，IFM による奥行き推定では，精度改善を行わないものと比べて，データセットによって効果に差はあるが 1.8～2.4 % の性能向上が確認できる．また，データセット毎に最大効果を示す最適なフィードバック回数が異なることが確認できる．さらに，すべてのデータセットにおいて最適なフィードバック回数でフィードバックを行った IFM は SGM より誤り率が小さくなっている．

また，図 3.2 に各手法で取得した “Teddy” のデプスマップを示す．図 3.2(a) は精度改善前のデプスマップであり，すなわちブロックマッチングでの推定結果である．精度改善を行わない場合では，画素値境界が原画像の物体輪郭と一致していないため境界精度は低

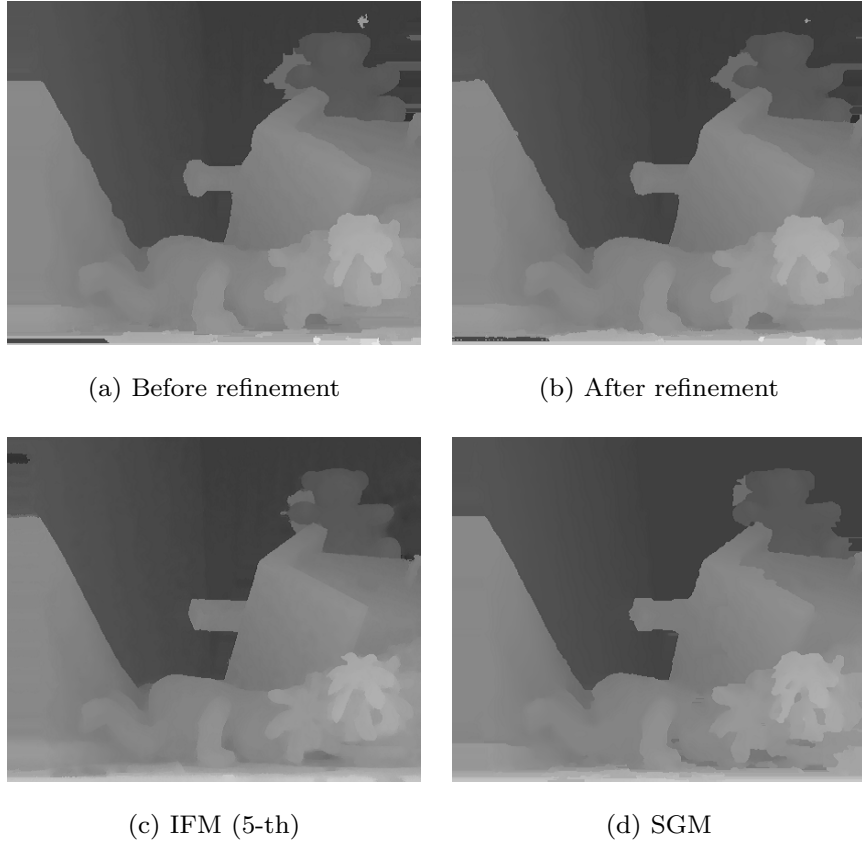


図 3.2 各手法のデプスマップ：Teddy.

表 3.1 誤り率：フィードバック回数と推定精度の推移.

Number of feedback	SGM	Before refinement	After refinement	1	2	3	4	5
Tsukuba	3.594	5.272	3.143	2.724	2.748	2.835	2.856	2.841
Venus	0.187	1.849	0.544	0.080	0.118	0.117	0.121	0.123
Teddy	7.390	9.556	7.098	6.287	4.810	4.151	3.807	3.557
Cones	3.630	4.392	3.201	2.852	2.748	2.678	2.644	2.686

く，また，奥行き値が誤った領域が多く存在する．図 3.2(b) は精度改善後のデプスマップであり，WJBF によって輪郭補正やノイズ抑制が行われるため，精度改善前のデプスマップに比べて境界精度は向上し，奥行き値が誤った領域も小さくなっていることがわかる．そして，図 3.2(c) は IFM のフィードバックを 5 回行った場合のデプスマップである．精度改善後のデプスマップと比べて，境界精度はさらに向上し，奥行き値が誤った領域もさらに小さくなっていることがわかる．これは，IFM のフィードバック機構によって，誤った奥行き値を持つ領域のマッチングコストが，正しい奥行き値のマッチングコストに収束していったためだと考えられる．図 3.2(d) は SGM で推定したデプスマップで

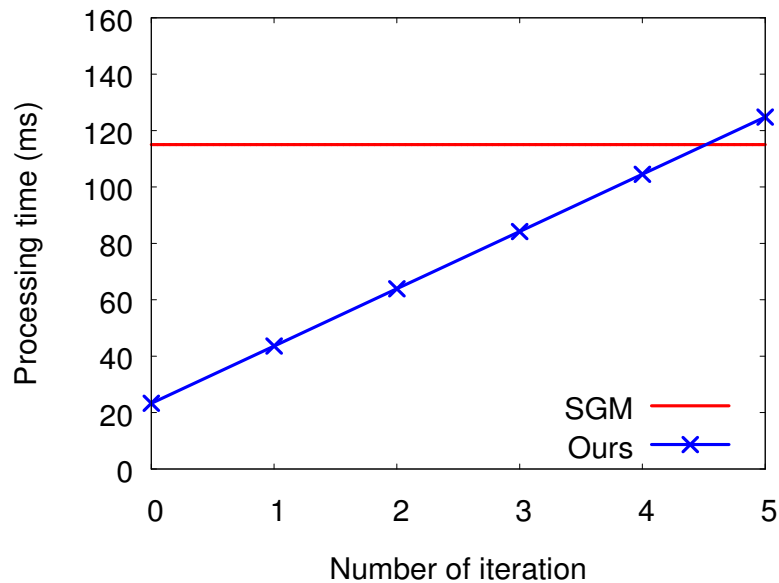


図 3.3 フィードバック回数と処理時間の推移 : Teddy.

ある．IFM の結果と比較すると，IFM の方が境界精度が高く，奥行き値が誤った領域も小さくなっている．従って，IFM がもつフィードバック機構は従来の精度改善や最適化に比べて，より効果的に奥行き推定を実行できると言える．

図 3.3 は IFM のフィードバック回数と処理時間の推移を示した図である．各手法は Visual Studio 2010 Windows 7 64bit を用いて C++ で実装した．実装は Intel Threading Building Block と SSE を用いて並列化を行っている．また，CPU のスペックは 3.50GHz Intel Core i7-3770K である．図 3.3 を確認すると，IFM の処理時間はフィードバック回数に従って単調に増加し，フィードバックを 4 回行うまでは SGM よりも高速に，5 回で SGM と同等の処理時間であることが確認できる．さらに表 3.1 で，フィードバック回数 4 回の際の IFM の誤り率と SGM の誤り率を比較すると，いずれのデータセットにおいても IFM の誤り率が SGM の誤り率を下回ることが確認できる．従って，計算コストと推定精度の関係から，処理コストが小さい処理を繰り返す IFM の方が高速な最適化を行う SGM よりも効率的な方法であるといえる．

3.4 おわりに

本章では，従来のステレオマッチングアルゴリズムを発展させ，新しいステレオマッチングのフレームワークであるインループフィードバックマッチングを提案した．インループフィードバックマッチングはエッジ保持平滑化フィルタで精度改善した結果をコ

スト集約以前にフィードバックし，これを繰り返すことでデプスマップの推定精度を高めていく．その結果，奥行き値の画素値境界は原画像の物体輪郭に補正され，また，誤った奥行き値領域が小さい高精度なデプスマップを取得できる．このインループフィードバックマッチングは，局所的手法でありながら，最適化手法のひとつである Semi-Global Matching よりも高精度にデプスマップを取得可能であり，さらに計算時間も短いため，高効率な手法であると言える．

今後の課題としては，最適なフィードバック回数の導出方法やフィードバックに適した精度改善手法の研究などが考えられる．

第 4 章

ジョイントフィルタによるデプスマップ超解像

4.1 まえがき

ステレオマッチングの計算コストは、画像の幅と高さ、そして奥行き階調数に依存する。高精度なデプスマップを推定するためには、最適化手法を用いることが有効である。しかしながら、最適化手法では、取り得るすべての奥行き値を考慮した処理を行うことが多いため、奥行き階調数が計算コストに大きく影響する。一方、第 2 章で提案したジョイントフィルタを用いた高効率な精度改善手法は、計算コストが奥行き階調数に依存しないフィルタリングを 1 度だけデプスマップに適用する手法であり、フィルタリングは低コストな処理となっているため、計算コストをかけずに高精度なデプスマップが取得可能である。また、第 3 章で提案した精度改善の結果をコスト集約以前にフィードバックするインループフィードバックマッチングは、繰り返し処理を行うものの、各ステップの処理コストは低く抑えられており、最適化手法の中でも高速な Semi-Global Matching と比較しても、高速に高精度なデプスマップが取得可能である。

しかしながら、近年の画像を用いたさまざまなアプリケーションやシステム、サービスは高解像度化が進んでおり、画像処理の計算コストは増加する流れにある。ステレオマッチングも例外ではなく、計算コストには画像の幅と高さが影響するため、高解像度な画像ほど計算コストは増加する。さらに、奥行き階調数は、画像解像度が高くなるほど多くなるため、画像解像度だけでなく奥行き階調数の増加分もステレオマッチングの計算コストに影響を与える。したがって、リアルタイムに動作し、かつ、高解像度な画像を用いるアプリケーションでは、ステレオマッチングへの速度要求は厳しいものになってくる。

また、デプスマップを用いたアプリケーションは、ネットワークを通じてサービス展開されることが想定される。通常、画像、映像は符号化処理によって元のデータ量を削減して伝送される。つまり、通常の画像と同様にデプスマップのデータ量削減も考慮する必要がある。自由視点映像通信 [183] などのアプリケーションでは、効率的なデプス伝送は重

要な課題である。デプスマップを伝送する手段として、デプスマップを伝送元で生成して伝送する方法と、伝送先でデプスマップを生成する方法が考えられる。例えば、伝送元でデプスマップを生成する場合では、元の画像とデプスマップのそれぞれ1枚を伝送する必要がある。また、伝送先でデプスマップを生成する場合では、同一シーンで視点の異なる2枚の画像を伝送する必要がある、伝送しなければならない画像の枚数は変わらない。しかしながら、ノイズがあるような品質の低い画像を用いてステレオマッチングを行うと、ノイズの影響を受けるため、奥行き推定精度が低下する。画像を高効率に符号化する方法としては H.264/AVC や H.265/HEVC が存在するが、これらは非可逆符号化であるため、符号化ノイズの発生によって画像品質は低下する。したがって、伝送元で劣化のない画像からデプスマップを推定した方が、推定直後のデプスマップの品質は高くなる。ところが、前述したように近年のアプリケーションなどは高解像度化しているため、推定したデプスマップのデータ量も無視はできない。そのため、デプスマップのデータ量を、デプスマップの品質を低下させずに削減する必要がある。

厳しくなるステレオマッチングの速度要求やデプスマップのデータ量の削減要求に対応する方法として、データのサブサンプリングが考えられる。これは、サブサンプリングされた低解像度の原画像からステレオマッチングすることで計算コストを下げる方法である。原画像が低解像度であれば、画像の幅と高さが小さくなるのに加えて奥行き値の階調数も少なくなるため高速なステレオマッチングが可能になる。加えて、デプスマップの画像解像度も小さくなるためデータ量も削減される。しかしながら、最終的にデプスマップを利用するためには、デプスマップの解像度をもとに戻す超解像手法が必要となる。

そこで、本章では、デプスマップの超解像をジョイントフィルタを用いて実現する。なお、提案手法は文献 [184, 185] を発展させたものである。

4.2 従来手法

ステレオマッチングで取得されるデプスマップは、推定元の原画像の解像度と同様の解像度で推定される。低解像度のデプスマップを高解像度化する方法としては、画像サイズを大きくするだけなら線形拡大などの拡大処理、より高度な方法を用いるなら従来の画像処理で用いられる超解像処理などが考えられる。しかしながら、低解像度のデプスマップを高解像度化すると、ステレオマッチングの推定誤差に加えて、高解像度化する際の誤りが発生する可能性がある。

また、推定精度や画像符号化によって劣化したデプスマップをアプリケーションで用いる場合では、アプリケーションの性能や品質が低下することが想定される。例えば、デプスマップを用いるアプリケーションとして自由視点画像合成 [178] があり、H.264/AVC などの非可逆符号化によって、デプスマップを低ビットレートで符号化すると、自由視点

画像の品質が大幅に劣化することが分かっている．文献 [179] では，H.264/AVC で符号化した画像とデプスマップを用いて，自由視点画像合成を行っており，デプスマップが低ビットレートで符号化されている場合は，たとえ画像に多くのビットを割り当てたとしても画像品質が向上しないことが示されている．また，文献 [180] では，H.264/AVC の多視点映像用の拡張である Multiview Video Coding (MVC) [181] を用いて複数のデプスマップを符号化した場合，低ビットレート時には自由視点映像の品質は大きく劣化し，逆に一定以上のビットをデプスマップに割り当てたとしても符号化利得がほとんど無くなることが示されている．一方で，デプスマップに発生する符号化劣化がポストフィルタにより解決することは，JPEG 符号化を例にして示されている [182]．文献 [182] では，デプスマップの JPEG 符号化によって発生するノイズや歪みが自由視点画像合成の際に，合成画像の品質を著しく低下させていることを指摘している．そして，それらはポストフィルタリングによって除去することができ，品質の低下を抑制できるとしている．流れとしては，デプスマップの圧縮，伝送，伸張，ポストフィルタリング，合成といった流れである．そして，ポストフィルタリングとして，Multi-Lateral Filter (MLF) [165] を用いている．

ここで，フィルタにより情報が復元できるということは，サブサンプリングした情報を可逆圧縮して送信し，復号側で情報を復元することも可能であることを示している．この復元アルゴリズムとして，デプスマップ超解像手法 [143, 144, 154, 157] が提案されている．

文献 [144] で提案されているジョイントバイラテラルアップサンプリング (JBU) では，サブサンプリングされた処理画素 p' と周辺画素 q' までの距離と，原画像での処理画素 p と周辺画像 q の画素値 I の類似度を重みとした線形和で画素を補間する．この JBU は次の式で示される．

$$D'_p = \frac{\sum_{q' \in N} w(p', q') c(p, q) D_{q'}}{\sum_{q \in N} w(p', q') c(p, q)} \quad (4.1)$$

$$w(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma_s^2}\right) \quad c(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|I_{\mathbf{x}} - I_{\mathbf{y}}\|_2^2}{2\sigma_c^2}\right) \quad (4.2)$$

ここで， D, D' はサブサンプリングされたデプスマップとアップサンプリングするデプスマップ， N は周辺画素集合， w, c は位置重み関数と色重み関数， σ_w, σ_c は各関数におけるガウス変数， $\|\cdot\|_2$ は L2 ノルムである．

また，文献 [157] では，線形和ではなく，最も重みが大きくなる画素位置 \mathbf{v} の値を採用することで画素値の混合を防ぎ，JBU の輪郭がぼける問題を解決している．ここでは，処理画素 p の求める奥行き値 D' は次の式で示される．

$$D'_p = D_v$$

$$v = \arg \min_{q' \in N} w(p', q') c(p, q) \quad (4.3)$$

ここで、補間に用いる周辺画素 q' の集合 N は、 q' の最近傍 4 画素である。

以上のようなデプスマップのアップサンプリング手法を用いれば、デプスマップの推定時には、低解像度の原画像を用いることで、高速なステレオマッチングが可能となるし、伝送時には低解像度のデプスマップを伝送することでデプスマップのデータ量を削減できる。そして、伝送先で非可逆符号化された原画像を復号し、低解像度のデプスマップをアップサンプリングすれば良い。

しかしながら、このようなデプスマップのアップサンプリング手法は精度は高いが計算時間がかかる [143]、輪郭がぼける [144, 154]、平面部分の滑らかな補間が不可能 [157] など、各手法には欠点がある。

4.3 提案手法

ここでは、輪郭がぼけず、それ以外の部分は滑らかな補間が可能となり、かつ、計算コストの低いデプスマップの超解像手法を提案する。また、提案手法はサブサンプリングしたデプスマップを可逆圧縮し、受信側で超解像することで、デプスマップを JPEG 符号化するよりも伝送量が削減できる手法である。

提案手法では、線形和で画素を混合することで、平面部分で滑らかな補間できる。ぼけを防ぐため、輪郭部分では大きな差がある奥行き値同士を混色しないように、重み w を用いてステレオマッチングで求めたデプスマップ D の画素に対して重み付き線形和を行う。求めるデプスマップ D'' は、次の式で計算する。

$$D''_p = \frac{1}{\sum_{q \in N} w(p, q)} \sum_{q \in N} w(p, q) D_{q'} \quad (4.4)$$

ここで、 p, q は処理画素と周辺画素位置、 q' はサブサンプリングされた周辺画素、 N は周辺画素 q の集合である。重み w は、文献 [157] の手法である式 (4.3) でアップサンプリングしたデプスマップの画素 p における奥行き値 D'_p と、サブサンプリングされたデプスマップにおいて画素 p の周辺に位置する画素 q' の奥行き値 $D_{q'}$ との差が閾値より小さければ、一般的な双線形補間の距離に応じた重み $li(p', q')$ と色重みの積を重みとして与え、それ以外は 0 としてその奥行き値を使わないように補間する。

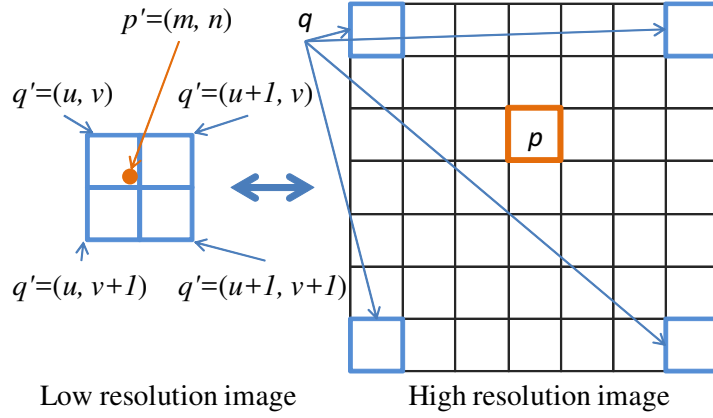


図 4.1 高解像度画像座標と低解像度画像座標.

具体的には，まず処理画素のサブサンプリングされた小数点座標位置を $\mathbf{p}' = (m, n)$ とした時，その最近傍 4 画素の内の左上の画素は床関数 $\lfloor \cdot \rfloor$ を用いて $\mathbf{q}' = (\lfloor m \rfloor, \lfloor n \rfloor) = (u, v)$ と表される．このとき，近傍画素の集合は $N \in (m, n), (m+1, n), (m, n+1), (m+1, n+1)$ となる．この座標の様子を図 4.1 に示す．この近傍画素から双線形補間する重み $li(\mathbf{p}', \mathbf{q}')$ は次の式で示される．

$$li(\mathbf{p}', \mathbf{q}') = \begin{cases} (u+1-m)(v+1-n) & \mathbf{q}' = (u, v) \\ (m-u)(v+1-n) & \mathbf{q}' = (u+1, v) \\ (u+1-m)(n-v) & \mathbf{q}' = (u, v+1) \\ (m-u)(n-v) & \mathbf{q}' = (u+1, v+1) \end{cases} \quad (4.5)$$

ここで， u は X 座標の整数位置， v は Y 座標の整数位置である．前記の式に示されるように，重み $li(\mathbf{p}', \mathbf{q}')$ は選択される近傍画素に応じてそれぞれ決定される．そして，前述したように，求めた奥行き値と近傍の奥行き値との差が大きければ，重み w は線形和の重み $w(\mathbf{p}, \mathbf{q})$ で与えられ，差が大きければ 0 とする．この処理は画素毎の閾値 $th_d(\mathbf{q}')$ に応じて選択され，次の式で示される．

$$w(\mathbf{p}, \mathbf{q}) = \begin{cases} li(\mathbf{p}', \mathbf{q}')c(\mathbf{p}, \mathbf{q}) & |D'_p - D_{q'}| < th_d(\mathbf{q}') \\ 0 & \text{else} \end{cases} \quad (4.6)$$

ここで， D' は文献 [157] の手法で求めるデプスマップ， th_d は閾値である．閾値 th_d は，

奥行き値の画素値境界では小さく、それ以外では大きく設定することで、より輪郭のぼけを抑制しつつ平面部分の滑らかな補間が可能になる。そのため、閾値 th_d は奥行き値の画素値境界とそれ以外の領域で適切に設定されることが望ましい。そこで、画素値境界はサブサンプリングされたデプスマップでラプラシアンフィルタを用いたエッジ検出によって検出する。ラプラシアンフィルタカーネルは、ラプラシアンフィルタ関数 *Laplacian* を用いて次の式で示される。

$$Laplacian = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (4.7)$$

求める閾値 th_d は、ラプラシアンフィルタの出力結果によって切り替えを行う。それは、サブサンプリングされたデプスマップ D からラプラシアンフィルタによってエッジ検出を行い、その際の応答値の絶対値を閾値処理することで切り替える。そのとき、エッジだと判定された領域には低い閾値 th_l 、それ以外の平坦な領域だと判定された領域には、高い閾値 th_h を設定する。したがって、閾値 th_d はラプラシアンフィルタ関数を用いて次の式で求められる。

$$th_d(\mathbf{q}') = \begin{cases} th_l & |Laplacian(D_{\mathbf{q}'})| > th_L \\ th_h & \text{else} \end{cases} \quad (4.8)$$

ここで、 th_l, th_h はそれぞれ低い閾値 th_d と高い閾値 th_d 、 th_L はエッジ検出のための閾値であり、これらは経験的に $th_l = 3, th_h = 9, th_L = 6$ と設定した。このように、デプスマップの画素値境界か否かによって適応的に重みを変化させる。

この方法では、エッジではないと判定された領域の閾値が低く、その領域で外れ値が存在した場合に、望まない奥行き値との線形和になる恐れがある。しかしながら、そのような外れ値を持つ領域は、色情報が異なっている可能性が高いため、そのような場合には、式 (4.6) の色重み $c(\mathbf{p}, \mathbf{q})$ が有効に働く。なお、この色重みは式 (4.3) を用いて奥行き値 d'_p を求める過程で算出する色重みを再利用することで、余分な計算なしに与えることができる。

4.4 評価実験

提案方式の有効性を検証するために、デプスマップの復元精度を評価する。また、デプスマップのデータ量削減の評価として、復元したデプスマップを文献 [178] の Depth

表 4.1 各手法のデプスマップの復元効果 (リサイズ 8 倍).

Method	[143]	[144]	[157]	Proposed	[182]
Th = 0.5 [%]	6.86	7.56	7.26	6.23	34.57
Th = 1.0 [%]	4.56	4.99	5.27	3.98	5.50

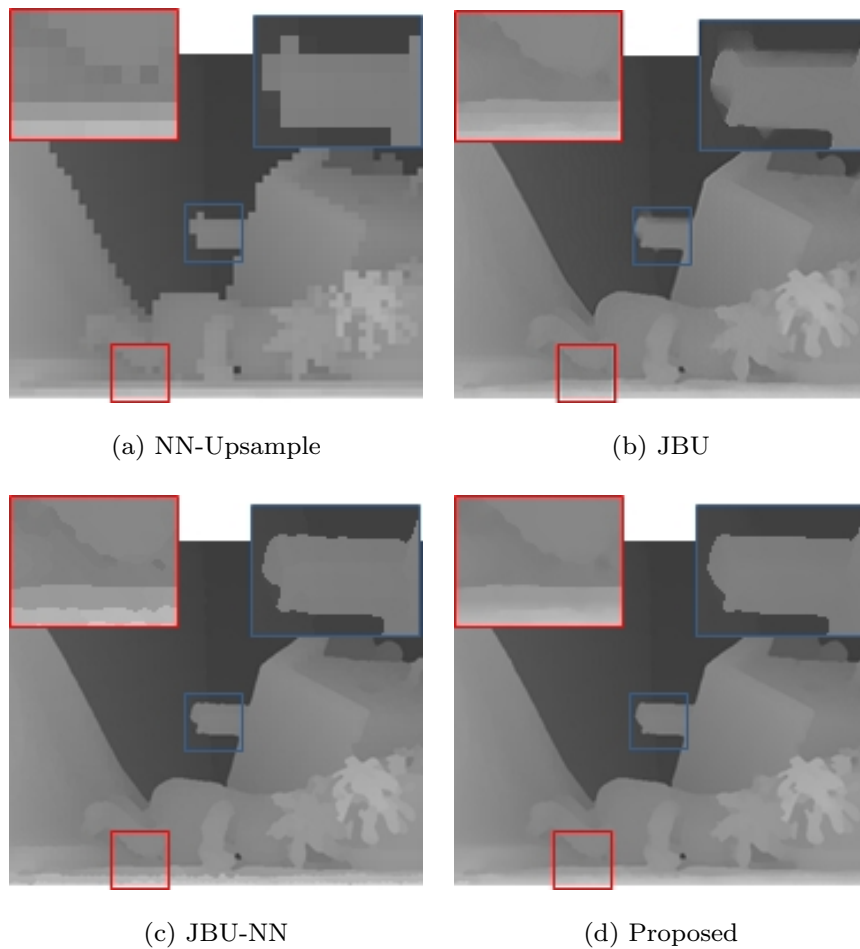


図 4.2 各手法の超解像結果.

Image Based Rendering (DIBR) に適用し, 自由視点画像合成による Peak Signal to Noise Ratio (PSNR) による評価も行った.

評価には, 多視点画像データセットの “Teddy” を用いる [6]. この評価では, デプスマップの復元精度を評価するため, 入力デプスマップには正解デプスマップを用いて, 復元手法による劣化具合を評価する. なお, 正解デプスマップには欠損部分がある. これはステレオマッチングによって奥行きが求まらない部分に対応するが, 欠損部分が超解像

処理によって伝播することを防ぐため、欠損部分については周囲の最小奥行き値でパディングした．評価では，文献 [6] のマスク “all” を用い，真値との差が閾値以上のピクセルが何%あるかで比較した．

また，データ量削減の評価では，復元したデプスマップを用いて，多視点データセットの中心に位置する画像を自由視点画像合成によって生成し，原画像と生成画像を比較した PSNR で評価した．このとき，提案手法では，伝送元でサブサンプリングした原画像から低解像度のデプスマップを取得することを想定し，取得したデプスマップは PNG による可逆圧縮を行う．そして，伝送先では低解像度のデプスマップを提案手法によって超解像し，自由視点画像合成を行う．一方，比較手法には，JPEG とポストフィルタを用いる文献 [182] の方法を用いた．比較手法では，伝送元で取得した高解像度のデプスマップを JPEG により非可逆圧縮することを想定し，伝送先ではポストフィルタによって符号化歪みを除去する．ここで，JPEG 圧縮の場合の品質は総当たり実験の結果，最も効率が良くなるように設定している．また，原画像の圧縮は提案手法，比較手法ともに JPEG で圧縮している．

表 4.1 に，各手法でのデプスマップの復元結果を示す．ここでは，高解像度の原画像を用いて， $1/8$ 倍の解像度のデプスマップを原画像と同じ解像度まで超解像している．なお，表中の文献 [182] の結果に関しては，提案手法と同サイズまで小さく圧縮できなかったため，データサイズが 3 倍程度でありクオリティ 15 での評価結果である．評価の結果，誤差が 1 画素以内のときも，サブピクセル精度である 0.5 画素以内のときも提案手法の復元精度が最も高いことが確認できる．文献 [143] の手法は精度が高いとされているが今回のようなノイズが少ない場合には効果が小さく，また，デプスマップを超解像するのに 1 枚あたり約 3 分かかった．また，文献 [182] の手法では，サブピクセル精度で評価すると，他と比べて性能差が大きくなることが確認された．さらに，文献 [182] の手法では，カーネルサイズ 7 の際にポストフィルタに 44ms，符号化に 3ms の計 47ms を要した．提案手法は，約 1.5ms で計算可能であり，計算時間と復元精度の観点から，提案手法の方が効率的な手法であるといえる．なお，使用した計算機は Intel Core i7 2.93GHz である．

図 4.2 は，実際にデプスマップの超解像結果である．まず，図 4.2(a) は，最近傍法 (NN-Upsample) を用いたものであり，単純な画像拡大では，図中青枠に示すように輪郭に凹凸が発生し，図中赤枠に示すように滑らかな奥行き値の変化が表現できない．図 4.2(b) は，文献 [144] の JBU を用いたものであり，JBU は原画像を参照した処理を行うため，図中青枠に示すように物体輪郭がもとの形状に近くなるように補正されている．ただし，線形補間を行っているため，ぼけが発生していることも確認できる．一方で，図中赤枠のように奥行き値が滑らかに変化する部分では，線形補間によって滑らかな変化が再現できている．図 4.2(c) は，文献 [157] の JBU を利用した最近傍法 (JBU-NN) であり，JBU と同様に原画像を参照した処理を行うため，図中青枠に示すように輪郭補正ができている

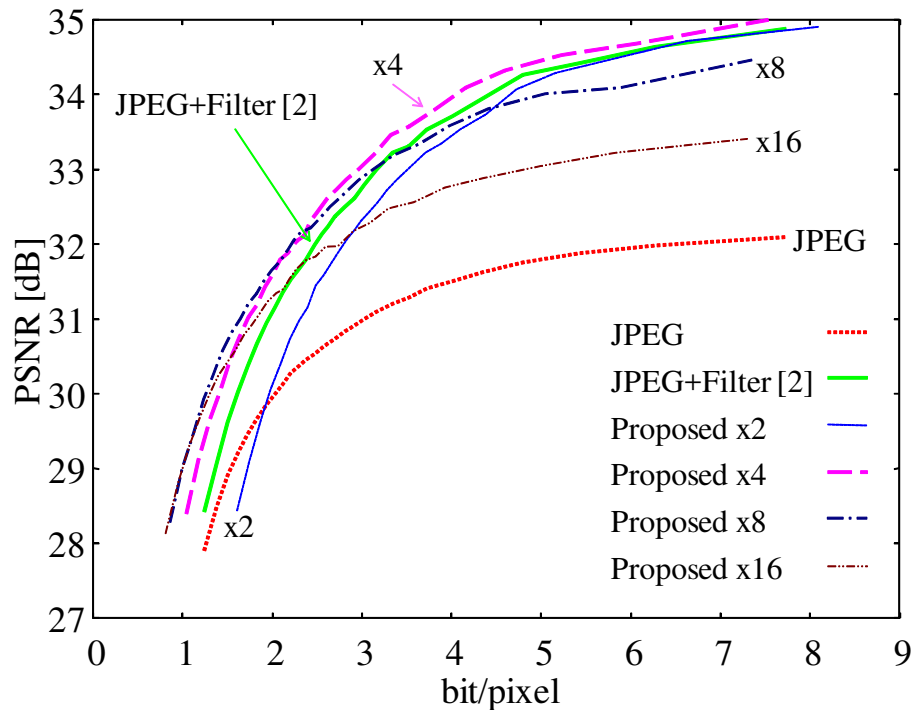


図 4.3 JPEG と超解像を基にした圧縮手法の符号化結果.

ことが確認できる．さらに，重みを考慮した最近傍補間であるため，輪郭補正しつつぼけも発生しない．一方で，最近傍補間であるため，図中赤枠のように滑らかな変化は再現できない．そして，図 4.2(d) は，提案手法を用いた結果である．提案手法は，奥行き値境界を考慮して，フィルタ効果を適応的に切り替えるため，画素値境界ではぼけのない輪郭補正が可能であり，滑らかに奥行き値が変化する部分でも奥行き値が分離されることなく滑らかに再現できていることが確認できる．

そして，図 4.3 に JPEG, 文献 [182], 提案手法の 2, 4, 8, 16 倍で超解像した符号化結果を示す．このグラフの X 軸は自由視点画像合成に用いた入力画像と入力デプスマップの和である総 bit 量である．文献 [182] の手法に対して，提案手法で 4 倍に拡大した場合はすべてのビットレートで高効率であり，また，低ビットレートでは 8 倍，16 倍といった倍率でも有効であった．

4.5 おわりに

本章では，ジョイントフィルタを用いた高精度かつ高速なデプスマップの超解像手法を提案した．提案手法を用いれば，低解像度の原画像を用いたステレオマッチングにより取

得した低解像度のデプスマップを高精度に超解像できるため，解像度を落としたステレオマッチングが可能となり，ステレオマッチングの高速化が実現できる．さらに，提案手法の復元精度は従来手法よりも高精度なものである．加えて，解像度の不足するデプスセンサ等のデプスマップの解像度を RGB 画像と同じものに高解像度化するといった利用も可能である．また，超解像をサブサンプリングと組み合わせて用いることでデプスマップの高効率な圧縮も可能であり，JPEG 等のリアルタイム性の高いエンコーダに比べて同程度の計算速度に押さえながら，高効率な符号化を実現できる．

今後の課題として，より高解像度なデータへの適用やノイズの影響などの調査，高度なポストフィルタ [186, 187] や画像合成方法 [188] の適用，他の符号化形式との比較があげられる．

第 5 章

結論

本論文では、高精度なデプスマップを高速に取得することを目的とした奥行き推定手法について、様々なアプローチで検討した。

第 2 章では、ステレオマッチングの精度改善に着目し、2.3 節では、奥行き値による重みと各画素の信頼度を考慮することで、単体フィルタでデプスマップのノイズ抑制、境界精度の改善、境界付近のぼけ抑制が可能な信頼度を用いたジョイントトリラテラルフィルタを提案し、従来の精度改善手法よりも、高精度にデプスマップの精度改善が可能となった。2.4 節では、各画素の信頼度を考慮することで、ノイズ抑制と境界精度の改善を行う重み付きジョイントバイラテラルフィルタと、境界付近のぼけを抑制するジョイントニアレストフィルタをそれぞれ提案した。それぞれの効果に特化したフィルタを用意することで、2.3 節の提案手法よりさらに効果的な精度改善が可能となった。2.5 節では、取り得る奥行き値の尤度に対してフィルタ処理するコストボリュームフィルタリングに重み付きジョイントバイラテラルフィルタを適用することで、直接フィルタでは改善が難しい大きな誤り領域の改善が可能となった。以上の手法によって、計算コストの小さい精度改善において、デプスマップの大幅な改善と推定コスト低減の両立が可能となった。

第 3 章では、ステレオマッチング全体のフレームワーク、特にコスト集約に着目し、計算コストの小さい処理を精度改善で行い、その結果をコスト集約にフィードバックするという新しいステレオマッチングのフレームワークであるインループフィードバックマッチングを提案した。提案手法は、計算コストの小さな処理を繰り返すことで、従来の最適化手法よりもコストを抑えつつ高精度なデプスマップを取得可能となった。

第 4 章では、ステレオマッチングの後段に処理を追加することに着目し、デプスマップに特化したアップサンプリング手法を提案した。提案手法は、高精度なデプスマップの超解像が可能であるため、ステレオマッチングを行う画像解像度を小さくすることで、ステレオマッチング全体の計算コストを下げる事が可能となった。また、デプスマップの圧縮手法としても有効に利用できる。

本論文のいずれの提案手法でも、ジョイントフィルタを用いており、ジョイントフィルタがステレオマッチングの各ステップで有効に活用できることが明らかになった。デプス

マップは一般にノイズが多く、画素値境界が不正確なものであるが、ジョイントフィルタを精度改善に用いることで、ノイズ除去と境界精度の改善を高速に行うことができる。また、その結果をコスト集約以前にフィードバックすることで、最適化手法と同等精度のデプスマップを、より低コストに取得可能となる。そして、ジョイントフィルタを用いたデプスマップの超解像手法によって、全体の計算コストを削減することが可能となった。

これらの手法の活用例として、例えば、提案の精度改善手法は、ステレオマッチングの最終ステップの処理であるため、容易に既存のステレオマッチングに導入でき、少ない処理量の増加で高精度なデプスマップの取得が可能となる。また、提案手法では左右のステレオカメラで取得したデプスマップを精度改善するように信頼度や重みを設定するが、これを単眼用に改良すれば、単眼のデプスセンサにも適用可能である。ほかにも、インループフィードバックマッチングは、計算コストの小さい処理の繰り返しであるため、従来の最適化手法やコスト集約手法より低コストに実現可能である。そして、これらの精度改善手法やインループフィードバックマッチングは、計算コストの小さい局所的な手法であるため、ステレオマッチングを行う SoC や HW のワークメモリやランダムロジックを少なく抑えることができ、導入コストを低減することができる。さらに、デプスマップの超解像手法は、ステレオマッチング全体の計算コストを削減するだけでなく、デプスマップの容量を削減することもでき、デプスマップを通信するアプリケーションやサービスに運用することはもちろん、デプスマップを扱う種々のデバイス間のバンド幅を抑制するなどの利用も考えられる。以上のように、本論文で検討した手法は、デプスマップを用いる様々なデバイス、アプリケーション、サービスの発展に貢献できる。

今後、この技術をより効率的に活用していくためには、さらなる推定精度の向上や処理速度の高速化などの検討はもちろんのこと、利便性の追求や適用範囲の拡大も検討していかなければならない。例えば、各提案手法には種々のパラメータが存在し、実験的に最適値を導出することで最大性能を出している。しかしながら、実利用を想定すると、変化する撮影対象や撮影環境に合わせて随時パラメータを探索し直すのは煩雑な作業であり、自動的に適したパラメータが設定されることが望ましい。また、本論文ではデプスマップの推定対象は静止画像であったが、適用範囲を拡大し動画対応も検討する必要がある。ここでは、フレーム毎に変化する最適なパラメータは自動で導出されるか、そもそもパラメータにロバストな手法であることが望ましい。さらに、フレーム間でデプスマップの出力結果がばらつくと、フリッカーのような現象が発生する可能性があるため、フレーム間で変動の少ないデプスマップであることが望ましい。

したがって、今後の課題としては、ひとつは入力原画像やデプスマップとフィルタパラメータの依存性を調査し、その依存性解析手法とパラメータの自動設定アルゴリズムの検討が考えられる。また、時間情報を用いてフレーム間の変動を抑えるなど、動画に適したステレオマッチングアルゴリズムの検討が考えられる。

謝辞

本博士論文は，筆者が名古屋工業大学大学院工学研究科創成シミュレーション工学専攻に在学中に行った研究をまとめたものである．本研究において，ご指導頂いた石橋豊教授に感謝致します．そして，本研究をまとめるにあたり貴重な御意見を頂きました本学の松尾啓志教授，布目敏郎准教授，名古屋大学の藤井俊彰教授に深く感謝致します．また本研究は，福嶋慶繁准教授に終始直接ご指導いただきました．心より感謝申し上げます．親身になってお世話になりました本研究室の先輩，後輩の皆様，特に，小寺直紀氏，井上知彦氏，藤田秀氏には日頃から熱心な討議をしてくださいましたことに深く感謝致します．加えて，現所属の株式会社東芝研究開発センターの上長，同僚の皆様には，博士課程に進むにあたり多大なご理解とご協力をいただきましたことに深く感謝致します．最後に，日頃お世話になりました諸氏，その他関係者各位に感謝致します．

参考文献

- [1] B. Javidi and F. Okano, “Three-dimensional television, video, and display technologies,” Springer Science & Business Media, Springer, 2002.
- [2] M. Tanimoto, “Overview of free viewpoint television,” *Signal Processing: Image Communication*, vol. 21, no. 6, pp. 454–461, 2006.
- [3] S. Bae and F. Durand, “Defocus magnification,” *Computer Graphics Forum*, vol. 26, no. 3, pp. 571–579, 2007.
- [4] M. Naimark, “Spatial correspondence in motion picture display,” *Los Angeles Technical Symposium, International Society for Optics and Photonics*, pp. 78–81, 1984.
- [5] D. Piatti and F. Rinaudo, “SR-4000 and CamCube3.0 Time of Flight (ToF) cameras: Tests and comparison,” *Remote Sensing*, vol. 4, no. 4, pp. 1069–1089, 2012.
- [6] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,” *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, 2002.
- [7] M. Okutomi and T. Kanade, “A multiple-baseline stereo,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 1993.
- [8] B. Wilburn, N. Joshi, V. Vaish, E. V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, “High performance imaging using large camera arrays,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 765–776, 2005.
- [9] T. Fujii, K. Mori, K. Mase, M. Tanimoto, and Y. Suenaga, “Multipoint measuring system for video and sound – 100-camera and microphone system,” *IEEE International Conference on Multimedia and Expo*, pp. 437–440, 2006.
- [10] 藤井俊彰, 森健策, 竹田一哉, 間瀬健二, 谷本正幸, 末永康仁, “大規模実世界データベース構築のための多元多点計測装置の開発,” *3次元コンファレンス*, 3-1, 2006.
- [11] 福嶋慶繁, 丹羽健太, 圓道知博, 藤井俊彰, 谷本正幸, 西野隆典, 武田一哉, “多視点・多聴点データ取得システムを用いた自由視聴点映像生成,” *電子情報通信学会論文誌*, vol. 91-D, no. 8, pp. 2039–2041, 2008.
- [12] M. Tanimoto, T. Fujii, and N. Fukushima, “1D parallel test sequences for

- MPEG-FTV,” ISO/IEC JTC1/SC29/WG11 MPEG 2008 M15378, 2008.
- [13] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” Stanford Computer Science Technical Report, vol. 2, no. 11, 2005.
 - [14] G. Bradski and A. Kaehler, “Learning OpenCV: Computer vision with the OpenCV library,” O’Reilly Media, Inc., 2008.
 - [15] K. Konolige, “Small vision systems: Hardware and implementation,” International Symposium in Robotic Research, pp. 203–212, 1997.
 - [16] H. Hirschmüller, “Accurate and efficient stereo processing by semi-global matching and mutual information,” IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 807–814, 2005.
 - [17] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, and Y. Mori, “Reference softwares for depth estimation and view synthesis,” “ISO/IEC JTC1/SC29/WG11 MPEG 2008 M15377, 2008.
 - [18] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, and Y. Mori, “Depth estimation reference software (DERS) 5.0,” “ISO/IEC JTC1/SC29/WG11 MPEG 2009 M16923, 2009.
 - [19] R. Szeliski, “Computer vision: Algorithms and applications,” Springer Science & Business Media, Springer, 2010.
 - [20] R. Szeliski 著, 玉木徹, 福嶋慶繁, 飯山将晃, 鳥居秋彦, 栗田多喜夫, 波部斉, 林昌希, 野田雅文 訳, “コンピュータビジョン—アルゴリズムと応用—,” 共立出版, 2013.
 - [21] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color image,” IEEE International Conference on Computer Vision, pp. 839–846, 1998.
 - [22] A. Ansar, A. Castano, and L. Matthies, “Enhanced real-time stereo using bilateral filtering,” IEEE International Symposium on 3D Data Processing, Visualization, and Transmission, pp. 455–462, 2004.
 - [23] H. Hirschmüller and D. Scharstein, “Evaluation of cost functions for stereo matching,” IEEE Conference on Computer Vision and Pattern Recognition, 2007.
 - [24] R. Zabih and J. Woodfill, “Non-parametric local transforms for computing visual correspondence,” European Conference on Computer Vision, pp. 151–158, 1994.
 - [25] D. N. Bhat and S. K. Nayar, “Ordinal measures for image correspondence,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 4, pp. 415–423, 1998.

- [26] J. Banks and M. Bennamoun, "Reliability analysis of the rank transform for stereo matching," *IEEE Transactions on Systems, Man, and Cybernetics PartB: Cybernetics*, vol. 31, no. 6, pp. 870–880, 2001.
- [27] G. Egnal, "Mutual information as a stereo correspondence measure," *Technical Reports (CIS)*, Paper 113, 2000.
- [28] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 1, pp. 328–341, 2008.
- [29] M. Xing, S. Xun, Z. Mingcai, J. Shaohui, W. Haitao, and Z. Xiaopeng, "On building an accurate stereo matching system on graphics hardware," *IEEE International Conference on Computer Vision Workshops*, pp. 467–474, 2011.
- [30] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," *International Journal of Computer Vision*, vol. 35, no. 3, pp. 269–293, 1999.
- [31] D. G. Lowe, "Object recognition from local scale-invariant features," *IEEE International Conference on Computer Vision*, pp. 1150–1157, 1999.
- [32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [33] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 978–994, 2011.
- [34] F. Crow, "Summed-area tables for texture mapping," *ACM SIGGRAPH computer graphics*, vol. 18, no. 3, pp. 207–212, 1984.
- [35] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 511–518, 2001.
- [36] B. Bilgic, B. K. P. Horn, and I. Masaki, "Efficient integral image computation on the GPU," *IEEE Intelligent Vehicles Symposium*, pp. 528–533, 2010.
- [37] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920–932, 1994.
- [38] A. F. Bobick and S. S. Intille, "Large occlusion stereo," *International Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, 1999.
- [39] H. Tao, H. S. Sawhney, and R. Kumar, "A global matching framework for stereo computation," *IEEE International Conference on Computer Vision*, vol. 1, pp. 532–539, 2001.

- [40] S. B. Kang, R. Szeliski, and J. Chai, "Handling occlusions in dense multi-view stereo," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 103–110, 2001.
- [41] I. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 6, pp. 795–804, 1989.
- [42] J. Gil and M. Werman, "Computing 2-D min, median, and max filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 5, pp. 504–507, 1993.
- [43] K. J. Yoon and I. S. Kweon, "Locally adaptive support-weight approach for visual correspondence search," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 924–931, 2005.
- [44] K. J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006.
- [45] G. Pestschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 664–672, 2004.
- [46] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 673–678, 2004.
- [47] S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, "Bilateral filtering: Theory and applications," Now Publishers Inc., 2009.
- [48] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3017–3024, 2011.
- [49] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 504–511, 2013.
- [50] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 257–266, 2002.
- [51] C. Jiawen, S. Paris, and F. Durand, "Real-time edge-aware image processing with the bilateral grid," *ACM Transactions on Graphics*, vol. 26, no. 3, 2007.
- [52] T. Q. Pham and L. J. V Vliet, "Separable bilateral filtering for fast video preprocessing," *IEEE International Conference in Multimedia and Expo*, 2005.
- [53] N. Fukushima, S. Fujita, and Y. Ishibashi, "Switching dual kernels for separable

- edge-preserving filtering,” IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1588–1592, 2015.
- [54] S. Fujita and N. Fukushima, “Hyperspectral gaussian filtering: edge-preserving smoothing for hyperspectral image and its separable acceleration,” International Conference on Advanced Mechatronics, 2015.
 - [55] S. Paris and F. Durand, “A fast approximation of the bilateral filter using a signal processing approach,” European Conference on Computer Vision, pp. 568–580, 2006.
 - [56] S. Paris and F. Durand, “A fast approximation of the bilateral filter using a signal processing approach,” International Journal of Computer Vision, vol. 81, no. 1, pp. 24–52, 2009.
 - [57] F. Porikli, “Constant time $O(1)$ bilateral filtering,” IEEE Conference on Computer Vision and Pattern Recognition, 2008.
 - [58] Q. Yang, K. H. Tan, and N. Ahuja, “Real-time $O(1)$ bilateral filtering,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 557–564, 2009.
 - [59] Q. Yang, N. Ahuja, and K. H. Tan, “Constant time median and bilateral filtering,” International Journal of Computer Vision, vol. 112, no. 3, pp. 307–318, 2014.
 - [60] A. Adams, N. Gelfand, J. Dolson, and M. Levoy, “Gaussian kd-trees for fast high-dimensional filtering,” ACM Transactions on Graphics, vol. 28, no. 3, 2009.
 - [61] A. Adams, J. Baek, and M. A. Davis, “Fast high-dimensional filtering using the permutohedral lattice,” Computer Graphics Forum, vol. 29, no. 2, pp. 753–762, 2010.
 - [62] S. Yoshizawa, A. Belyaev, and H. Yokota, “Fast gauss bilateral filtering,” Computer Graphics Forum, vol. 29, no. 1, pp. 60–74, 2010.
 - [63] K. N. Chaudhury, D. Sage, and M. Unser, “Fast $O(1)$ bilateral filtering using trigonometric range kernels,” IEEE Transactions on Image Processing, vol. 20, no. 12, pp. 3376–3382, 2011.
 - [64] K. Sugimoto and S. Kamata, “Compressive bilateral filtering,” IEEE Transactions on Image Processing, vol. 24, no. 11, pp. 3357–3369, 2015.
 - [65] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson, “Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid,” European Conference on Computer Vision, pp. 510–523, 2010.
 - [66] K. He, J. Shun, and X. Tang, “Guided image filtering,” European Conference on Computer Vision, pp. 1–14, 2010.

- [67] K. He, J. Shun, and X. Tang, “Guided image filtering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.
- [68] E. S. L. Gastal and M. M. Oliveira, “Domain transform for edge-aware image and video processing,” *ACM Transactions on Graphics*, vol. 30, no. 4, 2011.
- [69] C. C. Pham and J. W. Jeon, “Domain transformation-based efficient cost aggregation for local stereo matching,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 7, pp. 1119–1130, 2013.
- [70] Q. Yang, “Recursive approximation of the bilateral filter,” *European Conference on Computer Vision*, pp. 399–413, 2012.
- [71] Q. Yang, “Recursive approximation of the bilateral filter,” *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1919–1927, 2015.
- [72] K. Zhang, J. Lu, and G. Lafruit, “Cross-based local stereo matching using orthogonal integral images,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073–1079, 2009.
- [73] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann, “Local stereo matching using geodesic support weight,” *IEEE International Conference on Image Processing*, pp. 2093–2096, 2009.
- [74] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, “Introduction to algorithms,” MIT press, 2009.
- [75] Q. Yang, “A non-local cost aggregation method for stereo matching,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1402–1409, 2012.
- [76] L. Bao, Y. Song, Q. Yang, H. Yuan, and G. Wang, “Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree,” *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 555–569, 2014.
- [77] P. Burt, L. Wixson, and G. Salgian, “Electronically directed “focal” stereo,” *IEEE International Conference on Computer Vision*, pp. 94–101, 1995.
- [78] S. Birchfield and C. Tomasi, “Multiway cut for stereo and motion with slanted surfaces,” *IEEE International Conference on Computer Vision*, vol. 1, pp. 489–495, 1999.
- [79] D. Gallup, J. M. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys, “Real-time plane-sweeping stereo with multiple sweeping directions,” *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [80] Y. Zhang, M. Gong, and Y. H. Yang, “Local stereo matching with 3D adaptive cost aggregation for slanted surface modeling and sub-pixel accuracy,” *IEEE International Conference on Pattern Recognition*, 2008.

- [81] M. Bleyer, C. Rhemann, and C. Rother, “Patchmatch stereo - stereo matching with slanted support windows,” British Machine Vision Conference, 2011.
- [82] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, “Patchmatch: A randomized correspondence algorithm for structural image editing,” *ACM Transactions on Graphics*, vol. 28, no. 3, 2009.
- [83] M. Mitzenmacher, E. Upfal 著, 小柴健史, 河内亮周 訳, “確率と計算 —乱択アルゴリズムと確率的解析—,” 共立出版, 2009.
- [84] F. Besse, C. Rother, A. Fitzgibbon, and J. Kautz, “PMBP: Patchmatch belief propagation for correspondence field estimation,” British Machine Vision Conference, pp. 132.1–132.11 2012.
- [85] P. Heise, S. Klose, B. Jensen, and A. Knoll, “Pm-huber: Patchmatch with huber regularization for stereo matching,” *IEEE International Conference on Computer Vision*, pp. 2360–2367, 2013.
- [86] J. Lu, H. Yang, D. Min, and M. Do, “Patch match filter: Efficient edge-aware filtering meets randomized search for fast correspondence field estimation,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1853–1841, 2013.
- [87] S. Xu, F. Zhang, X. He, X. Shen, and X. Zhang, “Pm-pm: Patchmatch with potts model for object segmentation and stereo matching,” *IEEE Transactions on Image Processing*, vol. 24, no. 7, pp. 2182–2196, 2015.
- [88] X. Huang, C. Yuan, and J. Zhang, “Graph cuts stereo matching based on patch-match and ground control points constraint,” *Advances in Multimedia Information Processing –PCM 2015*, pp. 14–23, 2015.
- [89] H. Baker and T. Binford, “Depth from edge and intensity based stereo,” *International Joint Conference on Artificial Intelligence-Volume 2*, pp. 631–636, 1981.
- [90] Y. Ohta and T. Kanade, “Stereo by intra- and inter-scanline search using dynamic programming,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, no. 2, pp. 449–470, 1985.
- [91] I. Cox, S. Hingorani, S. Rao, and B. Maggs, “A maximum likelihood stereo algorithm,” *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 542–567, 1996.
- [92] S. Z. Li, “Markov random field modeling in image analysis,” Springer Science & Business Media, Springer, 2009.
- [93] S. Z. Li, “Markov random field modeling in computer vision,” Springer Science & Business Media, Springer, 2012.

- [94] N. Fukushima, Y. Yendo, T. Fujii, and M. Tanimoto, "Free viewpoint image generation using multi-pass dynamic programming," *IS&T/SPIE Electronic Imaging, Stereoscopic Displays and Virtual Reality Systems XIV*, 64901F, 2007.
- [95] N. Fukushima, T. Fujii, Y. Ishibashi, Y. Yendo, and M. Tanimoto, "Real-time free viewpoint image rendering by using multi-pass dynamic programming," *IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, 2010.
- [96] S. K. Gehrig and C. Rabem, "Real-time semi-global matching on the CPU," *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 85–92, 2010.
- [97] I. Ernst and H. Hirschmüller, "Mutual information based semi-global stereo matching on the GPU," *Springer International Symposium on Visual Computing*, pp. 228–239, 2008.
- [98] S. K. Gehrig, F. Eberli, and T. Meyer, "A real-time low-power stereo vision engine using semi-global matching," *Springer International Conference on Computer Vision Systems*, pp. 134–143, 2009.
- [99] C. Banz, S. Hesselbarth, H. Flatt, H. Blume, and P. Pirsch, "Real-time stereo vision system using semi-global matching disparity estimation: Architecture and FPGA-implementation," *IEEE International Conference on Embedded Computer Systems*, pp. 93–101, 2010.
- [100] M. Humenberger, T. Engelke, and W. Kubinger, "A census-based stereo vision algorithm using modified semi-global matching and plane fitting to improve matching quality," *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 77–84, 2010.
- [101] S. Hermann and R. Klette, "Iterative semi-global matching for robust driver assistance systems," *Springer Asian Conference on Computer Vision*, pp. 465–478, 2013.
- [102] O. Veksler, "Stereo correspondence by dynamic programming on a tree," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 384–390, 2005.
- [103] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.
- [104] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?," *IEEE Transactions on Pattern Analysis and Machine Intelligence*,

- vol. 26, no. 2, pp. 147–159, 2004.
- [105] V. Kolmogorov and R. Zabih, “Computing visual correspondence with occlusions using graph cuts,” *IEEE International Conference on Computer Vision*, vol. 2, pp. 508–515, 2001.
 - [106] S. B. Kang, R. Szeliski, and J. Chai, “Handling occlusions in dense multi-view stereo,” *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 103–110, 2001.
 - [107] L. Hong and G. Chen, “Segment-based stereo matching using graph cuts,” *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 74–81, 2004.
 - [108] J. Sun, Y. Li, S. Bing, and K. H. Y. Shum, “Symmetric stereo matching for occlusion handling,” *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 399–406, 2005.
 - [109] V. Kolmogorov and R. Zabih, “Graph cut algorithms for binocular stereo with occlusions,” *Handbook of Mathematical Models in Computer Vision*, Springer US, pp. 423–437, 2006.
 - [110] Y. Boykov and O. Veksler, “Graph cuts in vision and graphics: Theories and applications,” *Handbook of mathematical models in computer vision*, Springer US, pp. 79–96, 2006.
 - [111] V. Vineet and P. J. Narayanan, “CUDA cuts: Fast graph cuts on the GPU,” *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2008.
 - [112] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, “Learning low-level vision,” *International Journal of Computer Vision*, vol. 40, no. 1, pp. 25–47, 2000.
 - [113] J. Sun, N. Zheng, and H. Y. Shum, “Stereo matching using belief propagation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 787–800, 2003.
 - [114] P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient belief propagation for early vision,” *International journal of computer vision*, vol. 70, no. 1, pp. 41–54, 2006.
 - [115] A. Brunton, C. Shu, and G. Roth, “Belief propagation on the GPU for stereo vision,” *IEEE Canadian Conference on Computer and Robot Vision*, pp. 76, 2006.
 - [116] Q. Yang, L. Wang, and N. Ahuja, “A constant-space belief propagation algo-

- rithm for stereo matching,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 1458–1465, 2010.
- [117] C. K. Liang, C. C. Cheng, Y. C. Lai, L. G. Chen, and H. H. Chen, “Hardware-efficient belief propagation,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 21 no. 5, pp. 525–537, 2011.
 - [118] A. Klaus, M. Sormann, and K. Karner, “Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure,” IEEE International Conference on Pattern Recognition, pp. 15–18, 2006.
 - [119] Q. Yang, L. Wang, R. Yang, S. Wang, M. Liao, and D. Nistér, “Real-time global stereo matching using hierarchical belief propagation,” British Machine Vision Conference, vol. 6, pp. 989–998, 2006.
 - [120] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nistér, “Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 3, pp. 492–504, 2008.
 - [121] M. O. Wildeboer, N. Fukushima, T. Yendo, M. P. Tehrani, T. Fujii, and M. Tanimoto, “A semi-automatic multi-view depth estimation method,” IS&T/SPIE Visual Communications and Image Processing, pp. 77442B–77442B, 2010.
 - [122] L. Matthies, T. Kanade, and R. Szeliski, “Kalman filter-based algorithms for estimating depth from image sequences,” International Journal of Computer Vision, vol. 3, no. 3, pp. 209–236, 1989.
 - [123] R. Szeliski, “Shape from rotation,” IEEE Conference on Computer Vision and Pattern Recognition, pp. 625–630, 1991.
 - [124] P. Fua, “A parallel stereo algorithm that produces dense depth maps and preserves image features,” Machine Vision and Applications, vol. 6, no. 1, pp. 35–49, 1993.
 - [125] D. Scharstein, “View synthesis using stereo vision,” Springer-Verlag, 1999.
 - [126] H. Hirschmüller and D. Scharstein, “Evaluation of stereo matching costs on images with radiometric differences,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 9, pp. 1582–1599, 2009.
 - [127] K. Lai, L. Bo, X. Ren, and D. Fox, “A large-scale hierarchical multiview RGB-D object dataset,” IEEE International Conference on Robotics and Automation, pp. 1817–1824, 2011.
 - [128] A. Telea, “An image inpainting technique based on the fast marching method,” Journal of Graphics Tools, vol. 9, no. 1, pp. 23–34, 2004.

- [129] S. Matyunin, D. Vatolin, Y. Berdnikov, and M. Smirnov, "Temporal filtering for depth maps generated by Kinect depth camera," *IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, 2011.
- [130] M. Camplani and L. Salgado, "Efficient spatio-temporal hole filling strategy for kinect depth maps," *IS&T/SPIE Electronic Imaging, Three-Dimensional Image Processing and Applications II*, pp. 82900E, 2012.
- [131] Y. S. Ho and S. B. Lee, "Joint multilateral filtering for stereo image generation using depth camera," *The Era of Interactive Media*, Springer New York, pp. 373–383, 2013.
- [132] L. Chen, H. Lin, and S. Li, "Depth image enhancement for Kinect using region growing and bilateral filter," *IEEE International Conference on Pattern Recognition*, pp. 3070–3073, 2012.
- [133] S. A. Hojjatoleslami and J. Kittler, "Region growing: A new approach," *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 1079–1084, 1998.
- [134] A. Dakkak and A. Husain, "Recovering missing depth information from Microsoft's Kinect," *Proc. Embedded Vis. Alliance*, 2012.
- [135] J. Park, H. Kim, Y. W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," *IEEE International Conference on Computer Vision*, pp. 1623–1630, 2011.
- [136] J. Yang, X. Ye, K. Li, and C. Hou, "Depth recovery using an adaptive color-guided auto-regressive model," *European Conference on Computer Vision*, pp. 158–171, 1994.
- [137] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, "Color-guided depth recovery from RGB-D data using an adaptive autoregressive model," *IEEE Transactions on Image Processing* vol. 23 no. 8, pp. 3443–3458, 2014.
- [138] M. Shimizu and M. Okutomi, "Precise sub-pixel estimation on area-based matching," *IEEE International Conference on Computer Vision*, pp. 90–97, 2001.
- [139] R. Szeliski and D. Scharstein, "Sampling the disparity space image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 419–425, 2004.
- [140] 原島博, 小田島薫, 鹿喰善明, 宮川洋, " ϵ -分離型非線形デジタルフィルタとその応用," *電子情報通信学会論文誌*, vol. J65-A, no. 4, pp. 297–304, 1982.
- [141] N. Fukushima and Y. Ishibashi, "Free viewpoint image generation with super resolution," *Picture Coding Symposium*, 2010.

- [142] N. Fukushima, T. Inoue, and Y. Ishibashi, "Removing depth map coding distortion by using post filter set," IEEE International Conference on Multimedia and Expo, 2013.
- [143] Q. Yang, R. Yang, J. Davis, and D. Nistér, "Spatial-depth super resolution for range images," IEEE Computer Vision and Pattern Recognition, 2007.
- [144] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," ACM Transactions on Graphics, vol. 26, no. 3, pp. 96, 2007.
- [145] D. Krishnan and R. Fergus, "Dark flash photography," ACM Transactions on Graphics, vol. 28, no. 3, 2009.
- [146] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," IEEE Transactions on Image Processing, vol. 16, no. 8, pp. 2080–2095, 2007.
- [147] A. Buades, B. Coll, and J.M. Morel, "A non-local algorithm for image denoising," IEEE Computer Vision and Pattern Recognition, 2005.
- [148] S. Fujita, N. Fukushima, M. Kimura, and Y. Ishibashi, "Randomized redundant DCT: efficient denoising by using random subsampling of DCT patches," SIGGRAPH Asia Technical Briefs, 7, 2015.
- [149] S. Fujita and N. Fukushima, "High-dimensional guided image filtering," International Conference on Computer Vision Theory and Applications, 2016.
- [150] N. Kodera, N. Fukushima, and Y. Ishibashi, "Filter based alpha matting for depth image based rendering," IEEE Visual Communications and Image Processing, 2013.
- [151] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," IEEE International Conference on Computer Vision, pp. 349–356, 2009.
- [152] X. Li and M. T. Orchard, "New edge-directed interpolation," IEEE Transactions on Image Processing, vol. 10, no. 10, pp. 1521–1527, 2001.
- [153] Z. Dengwen, X. Shen, and W. Dong, "Image zooming using directional cubic convolution interpolation," IET Image Processing, vol. 6, no. 6, pp. 627–634, 2012.
- [154] D. Chan, H. Buisman, C. Theobalt, and S. Thrun, "A noise-aware filter for real-time depth upsampling," Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications, 2008.
- [155] B. Huhle, T. Schairer, P. Jenke, and W. Straier, "Fusion of range and color images for denoising and resolution enhancement with a non-local filter," Computer Vision and Image Understanding, vol. 114, no. 12, pp. 1336–1345, 2010.

- [156] M. Y. Liu, O. Tuzel, and Y. Taguchi, “Joint geodesic upsampling of depth images,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 169–176, 2013.
- [157] M. O. Wildeboer, T. Yendo, M. P. Tehrani, T. Fujii, and M. Tanimoto, “Depth up-sampling for depth coding using view information,” *IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, 2011.
- [158] D. Min, J. Lu, and M. N. Do, “Depth video enhancement based on weighted mode filtering,” *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1176–1190, 2012.
- [159] J. Dolson, J. Baek, C. Plagemann, and S. Thrun, “Upsampling range data in dynamic environments,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1141–1148, 2010.
- [160] V. Garro, P. Zanuttigh, and G. Cortelazzo, “A new super resolution technique for range data,” *National Telecommunications and Information Theory Group*, 2009.
- [161] V. Garro, C. D. Mutto, P. Zanuttigh, and G. Cortelazzo, “Edge-preserving interpolation of depth data exploiting color information,” *Annals of Telecommunications*, vol. 68, no. 11, pp. 597–613, 2013.
- [162] 松尾琢也, 福嶋慶繁, 石橋豊, “自由視点画像合成のためのクロストリラテラルフィルタによる奥行き推定精度の向上,” *3次元画像コンファレンス*, 2011.
- [163] T. Matsuo, N. Fukushima, and Y. Ishibashi, “Reliability based cross trilateral filtering for depth map refinement,” *International Workshop on Advanced Image Technology*, 2012.
- [164] T. Matsuo, N. Kodera, N. Fukushima, and Y. Ishibashi, “Depth map refinement using reliability based joint trilateral filter,” *ECTI Transactions on CIT*, vol. 7, no. 2, pp. 107–116, 2013.
- [165] P. L. Lai, D. Tian, and P. Lopez, “Depth map processing with iterative joint multilateral filtering,” *IEEE Picture Coding Symposium*, pp. 9–12, 2010.
- [166] M. Mueller, F. Zilly, and P. Kauff, “Adaptive cross trilateral depth map filtering,” *IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, 2010.
- [167] J. Jachalsky, M. Schlosser, and D. Gandolph, “Confidence evaluation for robust, fast-converging disparity map refinement,” *IEEE International Conference on Multimedia and Expo*, pp. 1399–1404, 2010.
- [168] 松尾琢也, 福嶋慶繁, 石橋豊, “奥行き推定精度向上のための重み付きジョイントバ

- イラテラルフィルタ,” 情報処理学会全国大会, 2R-2, 2012.
- [169] 松尾琢也, 福嶋慶繁, 石橋豊, “重み付きクロスバイラテラルフィルタによる奥行き推定精度の向上,” 映像情報メディア学会論文誌, vol. 66, no. 11, pp. J434–J443, 2012.
 - [170] T. Matsuo, N. Fukushima, and Y. Ishibashi, “Depth map refinement with joint bilateral filter using LR-consist weight map,” International Workshop on Advanced Image Technology, pp. 739–744, 2013.
 - [171] T. Matsuo, N. Fukushima, and Y. Ishibashi, “Weighted joint bilateral filter with slope depth compensation filter for depth map refinement,” International Conference on Computer Vision Theory and Applications, pp. 300–309, 2013.
 - [172] A. Geiger, M. Roser, and R. Urtasun, “Efficient large-scale stereo matching,” Springer Asian Conference on Computer Vision, pp. 25–38, 2010.
 - [173] 松尾琢也, 福嶋慶繁, 石橋豊, “重み画像を用いた奥行き画像の適応的バイラテラルフィルタリング,” 画像符号化シンポジウム/映像メディア処理シンポジウム, I-1-04, 2012.
 - [174] 藤田秀, 松尾琢也, 福嶋慶繁, 石橋豊, “オプティカルフロー推定高精度化のためのコストボリュームフィルタ,” 画像符号化シンポジウム/映像メディア処理シンポジウム, I-4-05, 2013.
 - [175] S. Fujita, T. Matsuo, N. Fukushima, and Y. Ishibashi, “Cost volume refinement filter for post filtering of visual corresponding,” IS&T/SPIE Electronic Imaging, Image Processing: Algorithms and System XIII, pp. 93990Q, 2015.
 - [176] T. Matsuo, S. Fujita, N. Fukushima, and Y. Ishibashi, “Efficient edge-awareness propagation via single-map filtering for edge-preserving stereo matching,” IS&T/SPIE Electronic Imaging, Three-Dimensional Image Processing, Measurement, and Applications, pp. 93930S, 2015.
 - [177] 中村将大, 藤田秀, 松尾琢也, 福嶋慶繁, “リファインメント画像のフィードバックによる高効率なステレオマッチング,” 画像符号化シンポジウム/映像メディア処理シンポジウム, 2015.
 - [178] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, “View generation with 3D warping using depth information for FTV,” Signal Processing: Image Communication, vol. 24, no. 1-2, pp. 65–72, 2009.
 - [179] S. Zinger, L. Do, and P. H. N. de With, “Free viewpoint depth image based rendering,” Journal of Visual Communication and Image Representation, vol. 21, vol. 5-6, pp. 533–541, 2010.
 - [180] K. Klimaszewski, K. Wegner, and M. Domanski, “Distortions of synthe-

- sized views caused by compression of views and depth maps,” IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2009.
- [181] P. Pandit, Y. Chen, and S. Yea, “Text of ISO/IEC 144965:2001/PDAM 15 reference software for multiview video coding,” ISO/IEC JTC1/SC29/WG11 MPEG 2008 W9974, 2008.
 - [182] 福嶋慶繁, 石橋豊, “Depth image based rendering のための奥行き画像へのポストフィルタリング,” 電子情報通信学会論文誌, vol. J94-D, no. 12, pp. 1912–1915, 2011.
 - [183] N. Fukushima and Y. Ishibashi, “Client driven system of depth image based rendering,” ECTI Transactions on CIT vol. 5, no. 2, pp. 15–23, 2011.
 - [184] 福嶋慶繁, 松尾琢也, 石橋豊, “符号化のための重み付きジョイントバイラテラルフィルタを用いた奥行き画像超解像,” 画像符号化シンポジウム/映像メディア処理シンポジウム, P-1-04, 2011.
 - [185] 松尾琢也, 福嶋慶繁, 石橋豊, “奥行き画像符号化のための重み付きジョイントバイラテラルアップサンプリングによる超解像,” 電子情報通信学会論文誌, vol. J95-D, no. 9, pp. 1694–1697, 2012.
 - [186] N. Fukushima, T. Inoue, and Y. Ishibashi, “Removing depth map coding distortion by using post filter set,” IEEE International Conference on Multimedia and Expo, 2013.
 - [187] T. Inoue, N. Fukushima, and Y. Ishibashi, “Non-essentiality of correlation between image and depth map in free viewpoint image coding: Accurate depth map case,” IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2014.
 - [188] N. Fukushima, N. Kodera, Y. Ishibashi, and M. Tanimoto, “Comparison between blur transfer and blur re-generation in depth image based rendering,” IEEE 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2014.