

画像のブロックマッチングにおける相関関数と サブピクセル推定方式の最適化

新 井 元 基[†] 鷲 見 和 彦^{††} 松 山 隆 司^{††}

ステレオ視において精度良く奥行き情報を取得するには、画素の分解能以下の精度で最良の一致を与えるサブピクセル推定が望ましい。これまで多くのサブピクセル推定手法が提案されてきたが、なお実画像において誤差を含むことが問題となっていた。本研究では、実画像から得られたステレオペア画像同士のブロックマッチングにおいて、残差が発生することにより相関関数が理想的な曲線から変形していることを実証し、より現実に応じたモデルとして、二次式と絶対値差との線形結合で相関関数をモデル化する提案を行った。

Optimization of Correlation Function and Sub-Pixel Estimation Method on Block Matching

MOTOKI ARAI,[†] KAZUHIKO SUMI^{††} and TAKASHI MATSUYAMA^{††}

In block matching based stereo, if we need precise depth, sub-pixel estimation of best matching location is required. So far, many researches have been done on this sub-pixel estimation, but nothing is working well for real images. In this paper, I explained that the systematic error in sub-pixel estimation is generated by the mismatch of a correlation function and a fitting function, and I proved that the shape of correlation function is deformed from the ideal theoretical shape because of existing residual error of correlation function which is resulted in stereo calibration and digitization on block matching using real images. I proposed to model real correlation function by a linear combination of quadric and absolute difference. Applying this model, the systematic sub-pixel estimation error is mitigated, and it is effective also to existing the residual error of correlation function. The validity of this model was shown by applying to the simulation including some error and the real images.

1. 序 論

近年、自由な視点の映像生成や3次元映像表示、あるいは3次元の形状を用いた物体認識を目的として、3次元映像を撮影する手法が研究されてきた。

その中でも、カメラを用いて画像から3次元形状を取得する方法として、ステレオ視¹⁾が良く用いられる。このステレオ視において、奥行き情報の精度を十分出すのは大変難しく、これがしばしば問題となる。なぜなら、奥行きは視差に基づいて計算されるが、視差を大きく取るとオクルージョンが発生しやすくなり画像間の対応付けが困難になってしまうため、オクルージョンの発生を抑えるために視差はできるだけ小さく取るのが一般的であるためである。よって、ステレオ画像の奥行き分解能は空間方向の分解能と比べて小さ

いことが多く、これを補うためには、ピクセル単位よりも細かい精度で視差を求めること、つまり画像の対応付けをサブピクセル単位で行うことが必要となる。

ブロックマッチングによって画像間の対応付けを行う際、サブピクセル単位で画像の対応を求めるには、画像を線形補間などによって擬似的に高解像度化し、より細かい単位で対応を求める方法や、ピクセル単位で求めた一致度評価値を関数にフィッティングすることでサブピクセル単位の一致度評価値を内挿し、より細かい単位での対応を推定する方法²⁾³⁾などがある。前者の方法では、精度を上げるためにはより高解像度化した画像を必要とするため、大量のメモリと計算時間を必要とする。一方、後者の方法では、ピクセル単位で求めた一致度評価値によってサブピクセル単位の対応を計算できるため、少ない計算量で推定を行うことができるが、フィッティングする関数に依存する系統的な誤差が発生することが知られている。本論文では、後者の方法について検討する。

サブピクセル推定の際に生じる系統的な誤差については、これまでその性質や発生するメカニズムについ

[†] 京都大学電気電子工学科

School of Electrical and Electronic Engineering, Kyoto University

^{††} 京都大学大学院情報学研究科

Graduate School of Informatics, Kyoto University

てあまり考察されていなかった。

その中で、清水ら²⁾は、一致度評価値を利用したパラボラフィッティングによってサブピクセル推定を行うとピクセルロッキング現象と呼ばれる系統的な推定誤差が発生することを説明し、元画像に対する推定と、0.5画素ずれた画像系列に対する推定の間に逆位相の誤差が発生する性質を使用し、これらの間で誤差をキャンセルする手法を提案しており、高い推定性能を得た。この論文では、系統的な推定誤差は一致度評価関数とフィッティング関数の組み合わせによって発生すると述べられているものの、フィッティング関数を2次関数に限定しており、ピクセル単位で求めた一致度評価値を放物線でフィッティングすることの妥当性については考察されていない。

また³⁾では、サブピクセル推定誤差が発生するメカニズムについて述べられており、一致度評価関数とフィッティング関数の組み合わせによるサブピクセル推定誤差を評価し、適した組み合わせが存在することを説明している。しかし、この論文では実画像でブロックマッチングを行う際に起きる現象を全く考慮していないため、実際に実画像でブロックマッチングによってサブピクセル推定を行ってみると、この論文で推定誤差が小さいとされるSADと折れ線近似の組み合わせで推定誤差が大きくなり、逆に推定誤差が大きいとされるSADと放物線近似の組み合わせで推定誤差が小さくなるということがあった。

本研究では、ステレオ視においてより精度よく奥行きを求めるため、サブピクセル推定に特有の系統的な推定誤差の軽減し、かつ実画像においても有効な手法の確立を目指し、以下のようなことを行った。

まず、サブピクセル推定における系統的な推定誤差は、相関関数とフィッティング関数の mismatch によって発生することを説明した。次に、実画像のマッチングの際に生じるマッチング残差が相関関数を変形させ、サブピクセル推定結果に影響を与えていることを説明した。そして、相関関数を1次の対称関数（折れ線）と2次の対称関数（放物線）の和でモデル化することによって、系統的なサブピクセル推定誤差を軽減することができ、かつそれがマッチング残差の発生に対しても有効であることを示した。

2. サブピクセル推定の原理

この章では、ステレオ画像のブロックマッチングにおいて画像間の対応をサブピクセル単位で求める方法について説明する。

2.1 ブロックマッチング

ブロックマッチングでは、画像ブロック間の一致度を輝度差の総和や輝度差の2乗和といった一致度評価関数によって評価し、一致度評価値が最大または最小になる位置を探索することで、画像間の対応をピクセル

単位で求めることができる。

2.1.1 一致度評価関数

ブロック間の一致度を評価する関数として、輝度差の総和 (SAD) や輝度差の2乗和 (SSD) がよく用いられている。

2.1.1.1 SAD

両画像のブロック内の各画素 $f(i, j)$, $g(i, j)$ について差の絶対値をとり、その総和 S を一致度評価値とするものである。SAD においては、一致度評価値が小さいほどブロック間の一致度が高いことを意味する。

$$S_{SAD} = \sum_i \sum_j |f(i, j) - g(i, j)| \quad (1)$$

2.1.1.2 SSD

両画像のブロック内の各画素について差の二乗をとり、その総和を一致度評価値とするものである。SSD においても、SAD と同様に一致度評価値が小さいほどブロック間の一致度が高いことを意味する。

$$S_{SSD} = \sum_i \sum_j \{f(i, j) - g(i, j)\}^2 \quad (2)$$

本研究では扱わないが、その他に相互相関 (CC) や正規化相互相関 (NCC) などの方法も用いられる。

2.2 サブピクセル推定手法

自然画像では通常、濃度値はなめらかに変化する。つまり、ある画素の濃度値はその周りの画素の濃度値とある程度関係があると考えられるので、一方の画像のブロックをテンプレートとして他方の画像を探索する際、ブロックが一致点に近づくにつれて一致度評価値はなんらかの関係を持ってなめらかに変化し、最も一致度が高い箇所が最小 (CC や NCC においては最大) になると考えられる。

そのため、画像間の対応をサブピクセル単位で推定するには、ピクセル単位で求めた一致度評価値からそれらが従う関数、つまり相関関数を推定し、その関数が最小となる点を求めればよい。

本研究では、画像間で垂直方向のサブピクセル変位は無いとして考慮せず、水平方向のサブピクセル単位での変位のみを求める場合を想定する。これは、平行カメラ条件のステレオ視において、左右画像の対応点の垂直方向の変位が無い条件を想定しているためである。

例えば、左画像のある1点の対応を右画像から求めるとき、右画像で同一スキャンライン上を水平方向に順にブロックをずらしながら一致度評価値を計算すると、図1のようになったとする。この時、これらの値が実線のような関数に従うと仮定すると、この関数が最小となる x がサブピクセル単位での一致点であることができる。

ここで、サブピクセル推定を精度良く行うためには、相関関数を近似する関数を適切に設定する必要がある。

2.2.1 フィッティング関数

ピクセル単位で求めた一致度評価値から相関関数を

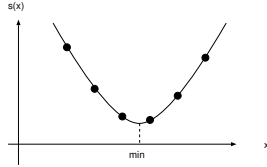


図 1 サブピクセル推定

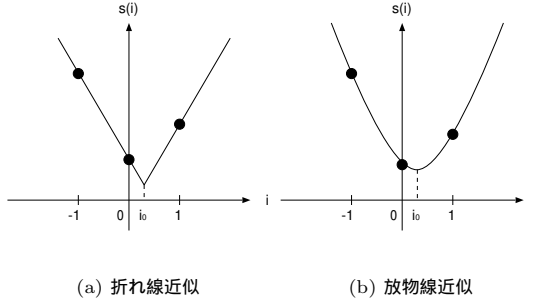


図 2 サブピクセル推定 (対称関数)

推定する関数として、一般的に 1 次の対称関数（折れ線）や 2 次の対称関数（放物線）が良く用いられる。ここでは、相関関数が折れ線や放物線になるとしてピクセル単位で求めた一致度評価値からそれらの関数のパラメータを求め、その関数が最小となる点を求めることでサブピクセル推定を行う手法について説明する。

2.2.1.1 折れ線近似

図 2(a) のように、相関関数が

$$S(i) = a|i - i_0| + S_0 \quad (3)$$

のような折れ線になると仮定し、ピクセル単位で求めた一致度評価値を用いてこの関数のパラメータ a , S_0 を推定することで、相関関数がピークとなる位置 $i = i_0$ 、つまりサブピクセル単位の変位 x_{sub} を推定することができる。

ただし、 $S(i)$ はピクセル単位での対応点からの水平方向のシフト量 i に対する一致度評価値とし、 a , S_0 は点ごとに異なる定数である。

このとき、ピクセル単位で求めた一致度評価値 $S(-1)$, $S(0)$, $S(1)$ を用いて、サブピクセル推定値を以下のように求めることができる。

$$x_{\text{sub}} = \begin{cases} \frac{S(-1) - S(1)}{2S(-1) - 2S(0)} & (S(-1) \geq S(1)) \\ -\frac{S(1) - S(-1)}{2S(1) - 2S(0)} & (S(-1) < S(1)) \end{cases} \quad (4)$$

ここで、 i はピクセル単位での対応点からの水平方向のシフト量であるから、相関関数が折れ線になるという仮定の下では $-0.5 \leq i_0 \leq 0.5$ であり、 x_{sub} は $-0.5 \leq x_{\text{sub}} \leq 0.5$ の範囲に限定される。

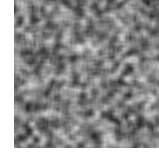


図 3 使用する画像

2.2.1.2 放物線近似

図 2(b) のように、相関関数が

$$S(i) = a(i - i_0)^2 + S_0 \quad (5)$$

のような放物線になると仮定し、ピクセル単位で求めた一致度評価値を用いてこの関数のパラメータを求めることで、サブピクセル単位の変位 x_{sub} を推定することができる。

このとき、ピクセル単位で求めた一致度評価値を用いて、サブピクセル推定値を以下のように求めることができる。

$$x_{\text{sub}} = -\frac{S(1) - S(-1)}{2S(-1) - 4S(0) + 2S(1)} \quad (6)$$

ここで、折れ線近似の場合と同様、 $-0.5 \leq x_{\text{sub}} \leq 0.5$ である。

3. サブピクセル推定誤差の解析

本章では、図 3 の画像を例に用いて、サブピクセル推定における系統的な推定誤差が、相関関数とフィッティング関数のミスマッチによって発生していることを説明する。

図 3 の画像は、数種類の大きさ、色の矩形をランダムに配置したパターンをぼかしたものであり、 1024×768 の領域に 2×2 , 4×4 , 6×6 , 8×8 の 4 種類の大きさ、8 階調の輝度の矩形をランダムに 100000 個配置したパターンに、大きさ 5 のぼかしをかけて得た。どの場所を選んでも他のどの場所ともユニークであるのでブロックマッチングに適しており、かつ輝度の変化が緩やかであるのでサブピクセル推定に適している画像であるといえる。

3.1 相関関数の形状

まず、図 3 の画像について、相関関数の形状を調べる。SAD, SSD それぞれの一致度評価関数を用い、ある 1 点とその点の周りの水平方向の点との一致度評価値をプロットしたところ、それぞれ図 4 のようになった。ただし、この画像の各点について同様のことを行い、どの点においてもほとんど同じ形状が得られることを確認している。また、ブロックサイズは 41×41 とし、以降も断りが無ければ同様とする。

この結果では、SAD では直線的に、SSD では 2 次曲線的に変化しており、一致度評価値はなんらかの関数を持ってなめらかに変化していることが分かる。

3.2 サブピクセル推定誤差

次に、図 3 の画像を水平方向に -0.5 画素から 0.5 画素まで 0.1 画素刻みで変位させた画像をそれぞれ用意し、

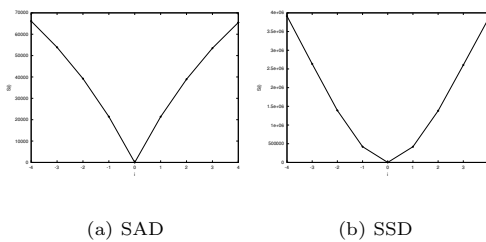


図 4 一致度評価値

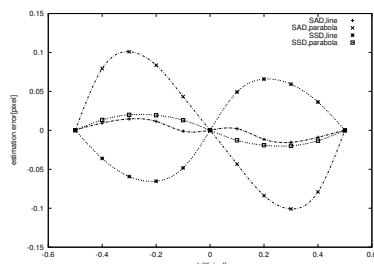


図 5 サブピクセル推定誤差

元の画像と変位した画像とでそれぞれサブピクセル推定を行った。このとき、変位を d ($-0.5 \leq d \leq 0.5$) とすると、平均推定誤差 $x_{\text{sub}} - d$ は図 5 のようになった。

ただし、サブピクセル単位で変位した画像を作成する際、線形補間

$$I(r) = (1-r)I(0) + rI(1) \quad (7)$$

によってサブピクセル単位 r ($0 < r < 1$) の画素値 $I(r)$ を計算した。

この結果から、相関関数を折れ線や放物線で近似すると系統的な誤差が生じていることがわかる。これは、誤差の性質より、実際の相関関数とフィッティング関数とのミスマッチによって生じていると考えられる。

SAD の場合、折れ線近似でも放物線近似でも推定結果が整数値に偏ってしまうのは、相関値が $[-1, 1]$ の区間でほぼ折れ線よりも大きくなっているためであると考えられる。

一方、SSD の場合、折れ線近似すると推定結果が整数値から遠ざかり、放物線近似すると推定結果が整数値に偏ってしまうのは、相関値が $[-1, 1]$ の区間で折れ線よりも小さく、かつ放物線よりも大きくなっているためと考えられる。実際に、3.1 章で求めた相関関数は、SAD、SSD とともにこのような形状になっていることが確認できる。

ここで、推定結果が整数値に偏る現象をピクセルロッキング現象、推定結果が整数値を避ける現象を反ピクセルロッキング現象という。

この結果は、³⁾ で述べられている系統的誤差の性質とよく一致している。³⁾ では、画像に含まれる隣接 3 画素の濃度値が直線的な変化からどの程度はずれてい

るかを表す特徴パラメータを導入し、画像全体でこのパラメータの分布が平均 0 標準偏差 σ のガウス分布に従うとして系統的誤差の性質を説明しており、この結果はこの画像モデルにおける $\sigma = 0.25$ の結果と同じ傾向を示している。また、この画像モデルによると、濃度値がなめらかに変化する自然画像でこのようなサブピクセル推定を行うと、この結果と同様の系統的誤差を生じると言える。

4. マッチング残差が及ぼす影響

実画像ではレンズ特性やカメラ姿勢の違い、外乱などによって、カメラキャリブレーションを行って画像を補正しても画像間で微小のスケール差や回転、量子化誤差、ノイズ、オクルージョンなどが存在し、マッチング位置でも部分的にミスマッチが生じているため、マッチング残差が存在する。また、これまでは垂直方向のサブピクセル変位は無いものとしていたが、実際は同様の理由で垂直方向のサブピクセル変位も存在し、これもマッチング残差の一種であると考えることができる。

そこで、マッチング残差をスケール差、回転、垂直方向の変位でモデル化し、これらがサブピクセル推定に及ぼす影響について考察する。

本章では、図 3 に以下のような処理を行った画像を用意し、元の画像とこれらの画像とのマッチングを行った。

- (1) 点 p を中心に 1% 拡大した画像
 - (2) 点 p を中心に 2 度回転した画像
 - (3) 垂直方向に 0.5 画素ずらした画像
 - (4) 上の (1)~(3) の 3 つの処理を全て行った画像
- ここで、点 p は画像上のある 1 点であるとする。

4.1 相関関数の形状

SAD、SSD のそれぞれの一致度評価関数を用いて、元の画像上のある 1 点 p と (1)~(4) の画像上の点 p に対応する点の周りの水平方向の点との一致度評価値をプロットすると、それぞれ図 6 のようになった。

マッチング位置で一致度評価値が 0 にならない、つまりマッチング残差が確認でき、相関関数のピーク付近が変形してなめらかになっているのがわかる。

また、SAD でも SSD でもマッチング残差によって相関関数に変形しているが、SSD の方が受ける影響が小さいことがわかる。これは、SSD は一致度評価関数が 2 次式であり、一致点からずれるにしたがって一致度評価値が 2 次関数的に変化するため、1 次式の一致度評価関数である SAD に比べてマッチング残差の影響が相対的に小さくなっているためである。

実世界のカメラとコンピュータ内部に設定したカメラモデルとの対応付けを行い、各種のカメラパラメータを決定することである。これによって、実カメラで撮影した画像データをモデル世界で解析することが可能となる。

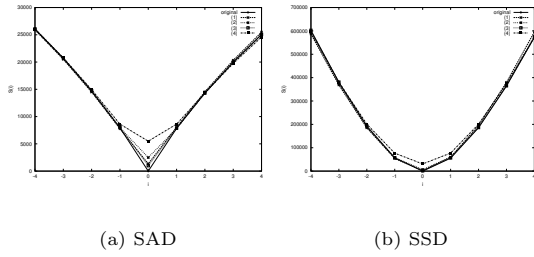


図 6 マッチング残差のある一致度評価値

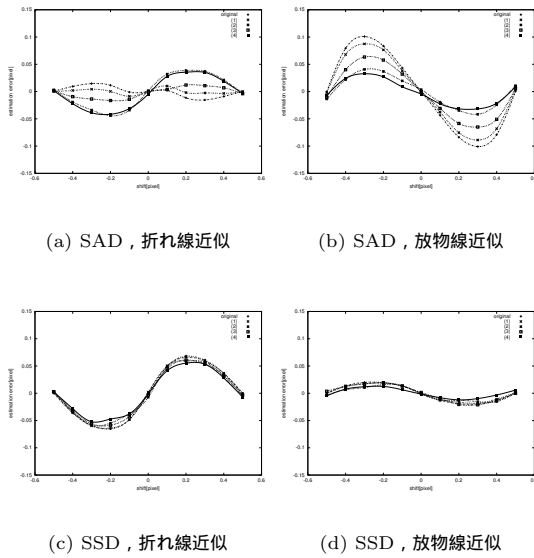


図 7 マッチング残差のあるサブピクセル推定誤差

4.2 サブピクセル推定誤差

元の画像と、(1)～(4)の画像をそれぞれ水平方向に-0.5画素から0.5画素まで0.1画素刻みで変位させた画像とで、点 p でのサブピクセル推定を行った。それぞれの場合について、10000回ずつ点 p を画像中から無作為に選んで推定誤差を計算し、推定誤差の平均をとると、それぞれ図7のような結果が得られた。

この結果からも、マッチング残差によってサブピクセル推定が影響を受けていること、一致度評価関数にSADを用いると大きく影響を受けるが、SSDを用いると受ける影響が小さいことが確認できる。

また、折れ線近似はマッチング残差によって誤差が大きくなる傾向にあり、さらにSADと折れ線近似の組み合わせの場合には誤差の符号が反転することがわかった。一方、放物線近似は誤差はあまり変わらないか、または誤差が小さくなる傾向にあることがわかった。ただし、これよりも大きなマッチング残差が存在する場合、誤差の符号が反転し、誤差が大

きくなる可能性がある。

5. サブピクセル推定方式の最適化

5.1 提案する手法

3.4章の議論から、一致度評価関数にSSDを用いる場合、実画像におけるマッチング残差が存在しても、ピーク付近の相関関数は折れ線よりも小さく、放物線よりも大きくなる、つまり傾きが折れ線と放物線の間になることが予想されるため、相関関数を折れ線と放物線の間の中間の傾きをもつ関数で近似すれば系統的誤差を小さくできると考えられる。

また、一致度評価関数にSADを用いる場合、図7(a)よりマッチング残差の無い状態ではピーク付近の相関関数が折れ線よりも大きくなってしまったため、折れ線と放物線の間の中間の傾きをもつ関数でフィッティングを行っても折れ線で近似する方法とあまり変わらないが、ある程度のマッチング残差があると、ピーク付近の相関関数は折れ線よりも小さく、放物線よりも大きくなるので、相関関数を折れ線と放物線の間の中間の傾きをもつ関数で近似すれば系統的誤差を小さくできる可能性はあると考えられる。

そこで、一致度評価関数にSADやSSDを用いる場合、次のような折れ線と放物線の和をとった関数で相関関数を近似する手法を提案する。

$$S(i) = a(i - i_0)^2 + b|i - i_0| + S_0 \quad (8)$$

ただし、 $S(i)$ はピクセル単位での対応点からのシフト量 i に対する一致度評価値であり、 a, b, S_0 は点ごとに異なる定数である。

このとき、ピクセル単位で求めた一致度評価値を用いて、サブピクセル推定値を以下のように求めることができる。

$$x_{\text{sub}} = \begin{cases} \frac{S(-1) - S(1)}{S(-1) - S(0) - S(1) + S(2)} & (S(-1) \geq S(1)) \\ -\frac{S(1) - S(-1)}{S(-2) - S(-1) - S(0) + S(1)} & (S(-1) < S(1)) \end{cases} \quad (9)$$

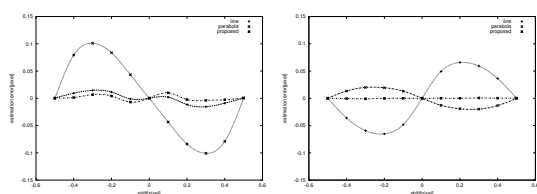
なお、折れ線近似、放物線近似では3点の一致度評価値を用いてサブピクセル推定値を計算していたが、このモデルでは4点の一致度評価値を用いて計算する。

5.2 提案手法のシミュレーション

図3の画像と、この画像を水平方向に-0.5画素から0.5画素まで0.1画素刻みで変位させた画像について、それぞれ提案する関数を用いたサブピクセル推定を行った。

平均推定誤差は図8のようになり、SADの場合は折れ線近似とほとんど同じ結果であるが、SSDの場合は折れ線近似や放物線近似のような系統的誤差がほとんどみられない。

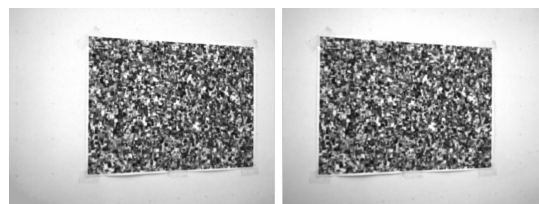
さらに、3.2章と同様の方法で元の画像と部分的ミスマッチのある画像(1)～(4)を水平方向に-0.5画素から0.5画素まで0.1画素刻みで変位させた画像とで、



(a) SAD

(b) SSD

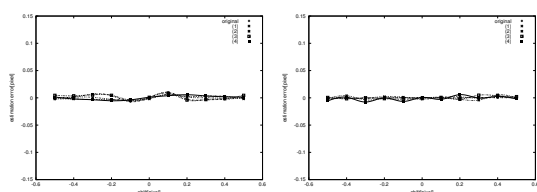
図 8 提案手法のサブピクセル推定誤差（従来手法との比較）



(a) 左画像

(b) 右画像

図 10 実験に使用したステレオ画像



(a) SAD

(b) SSD

図 9 提案手法のサブピクセル推定誤差（マッチング残差の影響）

それぞれ提案する関数を用いて点 p でのサブピクセル推定を行ったところ、平均推定誤差は図 9 のようになった。

この結果から、SSD を使った場合はマッチング残差の大きさに関係なく誤差が小さく、SAD を使った場合はマッチング残差によっては誤差が小さくなることを確認した。

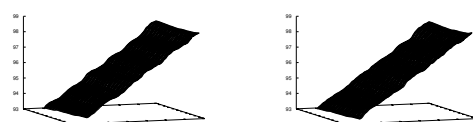
6. 実 験

平行カメラの条件で撮影した 2 枚の画像（図 10）を使用してステレオ視を行った。撮影対象は、カメラ光軸に対して直交する平面を、カメラ座標の y 軸を中心に回転したようなものであるが、厳密にそうになっているわけではない。領域ベースのマッチングを行うために、平面にはテクスチャとして図 3 のようなパターンを張り付けてある。画像サイズは $640 \times 640[\text{pixel}]$ 、ブロックサイズは $41 \times 41[\text{pixel}]$ である。このステレオ画像の一部（ $120 \times 50 = 6000$ 点）について、視差を計算した。

図 11 は、それぞれの一致度評価値とフィッティング関数の組み合わせで推定したサブピクセル単位の視差をプロットしたものである。

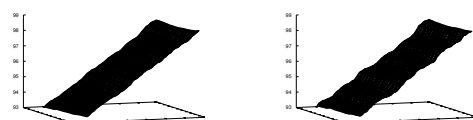
さらに、系統的な推定誤差の存在を示すため、推定視差の大きさをヒストグラムで表したものを図 12 に示す。

ここで、撮影対象が平面であり、かつ視差の変化が



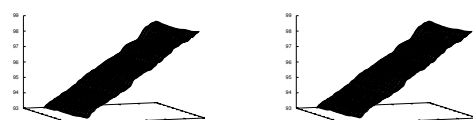
(a) SAD, 折れ線近似

(b) SAD, 放物線近似



(c) SAD, 提案手法

(d) SSD, 折れ線近似



(e) SSD, 放物線近似

(f) SSD, 提案手法

図 11 推定視差

ほぼ x 方向のみに依存していることを考慮すると、もし系統的な推定誤差がなければ、このヒストグラムは（左右の立ち上がり部分を除いて）一樣になると考えられる。よって、これらのヒストグラムを分散によって評価することを考えると、分散が小さいほど系統的な推定誤差が小さいと言える。ヒストグラムの左右の立ち上がり部分を除くため、 $94 \leq \text{disparity} \leq 98$ の領域に限定して分散を取ると、表 1 を得た。

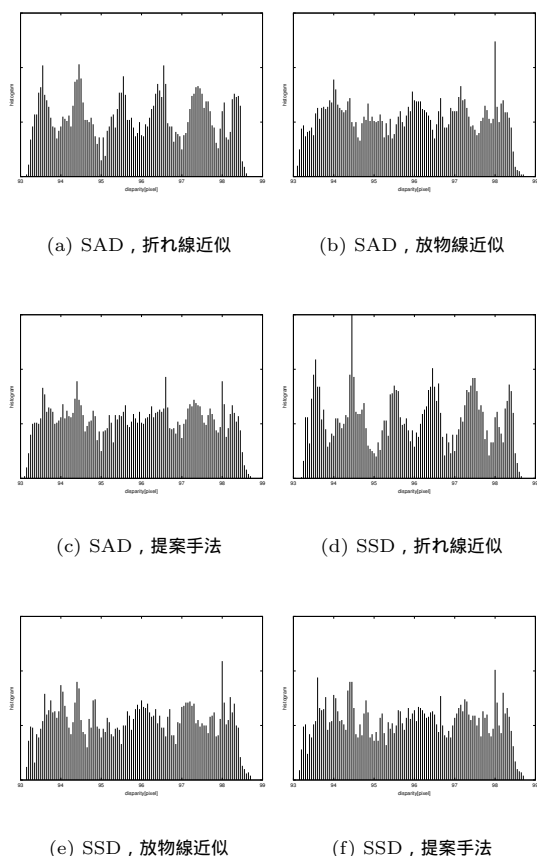


図 12 サブピクセル推定値のヒストグラム

表 1 ヒストグラムの統計量

手法	平均	分散	最小値	最大値
SAD, 折れ線近似	56.01	368.06	15	103
SAD, 放物線近似	56.68	192.05	33	124
SAD, 提案手法	56.32	132.32	25	93
SSD, 折れ線近似	56.06	568.38	20	161
SSD, 放物線近似	56.69	194.64	30	109
SSD, 提案手法	56.59	174.67	32	101

以上の結果から、それぞれの一致度評価関数とフィッティング関数との組み合わせで次のようなことを確認することができた。

- SAD, 折れ線近似
視差マップでは明らかな推定誤差が、ヒストグラム上では反ピクセルロッキング現象が顕著に現れていることが確認できる。3.2 章の結果では、この組み合わせは SSD と放物線近似の組み合わせよりも良い結果が得られていたが、4.2 章の結果のうちマッチング残差が大きい時と同様に反ピクセルロッキング型の誤差になった。
- SAD, 放物線近似

視差マップでは SAD と折れ線近似の組み合わせに比べると推定誤差は小さいが、ヒストグラム上ではピクセルロッキング現象が確認できる。3.2 章の結果では SSD と折れ線近似の組み合わせの次に大きな推定誤差が存在していたが、4.2 章におけるマッチング残差が大きいときの結果と同様、マッチング残差によって推定誤差が小さくなったと考えることができる。

- SAD, 提案したモデル
視差マップでは最も推定誤差が小さく、ヒストグラム上でもピクセルロッキング現象、反ピクセルロッキング現象はほとんど確認できない。ヒストグラムの分散は全ての組み合わせの中で最も小さく、SAD においては提案モデルを用いることで従来手法である折れ線近似、放物線近似よりも系統的な推定誤差が軽減されている。
- SSD, 折れ線近似
視差マップでは明らかな推定誤差が確認でき、ヒストグラム上でも反ピクセルロッキング現象が最も顕著に現れていることが確認できる。4.2 章の結果では、マッチング残差の大きさにほとんど関係なく推定誤差が大きい、その通りの結果となった。
- SSD, 放物線近似
視差マップでは一部に推定誤差が見られるものの、ヒストグラム上ではピクセルロッキング現象、反ピクセルロッキング現象は確認できず、分散からも系統的な推定誤差は少ないことが確認できる。4.2 章の結果においてマッチング誤差の大きな場合の結果によく一致し、マッチング残差によって推定誤差が小さくなったと考えることができる。
- SSD, 提案したモデル
視差マップでは最も推定誤差が小さいものの 1 つであり、ヒストグラム上でもピクセルロッキング現象、反ピクセルロッキング現象はほとんど確認できない。ヒストグラムの特徴は SSD と放物線近似の組み合わせの場合とあまり変わらないように見えるが、分散を比較するとこの組み合わせのほうが小さいので、SAD においても提案モデルを用いることで従来手法である折れ線近似、放物線近似よりも系統的な推定誤差が軽減されていると言える。

この結果から、実画像を用いたサブピクセル推定において一致度評価関数に SAD や SSD を用いる場合、提案するフィッティング関数を用いてサブピクセル推定を行えば系統的な推定誤差を軽減できることが確認できた。さらに、この実験では SSD よりも計算速度の速い SAD を用いた方が若干良い結果が得られていることが特徴的である。

また、実画像でのサブピクセル推定における系統的な推定誤差の特徴が、4 章で述べた内容と良く一致し

ていることが確認できた。

(2002).

しかし、系統的な推定誤差以外にも誤差が存在しており、視差が完全に平面にはならなかった。

7. 結 論

本論文では、ブロックマッチングにおけるサブピクセル推定誤差の性質を解析し、さらに実画像のマッチングの際に起きるマッチング残差がサブピクセル推定に与える影響を解析することにより、実画像においても有効な手法の確立を目指した。

一致度評価関数に SAD や SSD を用いた場合、相関関数を折れ線や放物線で近似すると、近似関数のミスマッチにより系統的な推定誤差が発生することを説明した。

実画像のブロックマッチングにおいては、さまざまな要因でマッチング残差が生じ、それがサブピクセル推定に影響を与えているため、マッチング残差が無い理想的な状態とは異なる結果になることが分かった。

提案したモデルでフィッティングを行えば、SAD や SSD を用いた場合、従来の手法である折れ線近似や放物線近似を適用した場合よりも系統的な推定誤差を低減でき、また計算量は従来の手法とほとんど変わらない。さらに、実験では、SSD よりも計算は高速だが精度は劣ると考えられてきた SAD において、SSD よりも良い結果が得られた。

しかし、実画像における実験で得られた結果では、本研究で考察対象としたピクセルロッキング現象や反ピクセルロッキング現象といった系統的な誤差以外にも誤差が存在しており、これについては今後の検討課題とする。

また、本研究では画像間で垂直方向のサブピクセル変位は無いとして考慮せず、水平方向のサブピクセル単位での変位のみを求める場合を想定したが、ブロックマッチングを利用する他の分野に応用する場合、2次元に拡張して水平方向と垂直方向のサブピクセル推定を行う手法が必要である。これについても、今後の検討課題としたい。

参 考 文 献

- 1) Brown, M. Z., Burschka, D. and Hager, G. D.: Advances in Computational Stereo, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 08, pp. 993–1008 (2003).
- 2) 清水雅夫, 奥富正敏: 画像のマッチングにおける高精度なサブピクセル推定手法, 電子情報通信学会論文誌, Vol. J84-D-II, No. 7, pp. 1409–1418 (2001).
- 3) 清水雅夫, 奥富正敏: 画像のマッチングにおけるサブピクセル推定の意味と性質, 電子情報通信学会論文誌, Vol. J85-D-II, No. 12, pp. 1791–1800