

コンピュータ理工学特別研究報告書

題目 ライトフィールドカメラの研究
-立体画像の生成と評価-

学生証番号 947389
氏名 松本 翔伍

提出日 平成 27 年 1 月 22 日
指導教員 蚊野 浩

京都産業大学
コンピュータ理工学部

要約

本研究では、ライトフィールドカメラを用いて、両眼立体視のための3D画像を生成した。特に、撮影したシーンの立体感を再現する3D画像を生成することを目標とした。

まず、視点位置の異なる2枚の写真画像（左目画像・右目画像）を両眼立体視した時に再生される3D空間の理論式を導いた。この理論式に基づいて3D画像を生成することで、撮影したシーンの立体感を再現することが可能になる。ついで、市販のライトフィールドカメラ Lytro を用いて、3D空間を忠実に再現する3D画像を生成する方式を検討した。Lytro は写真画像とともに、被写体までの奥行き情報を表すデプス画像を出力する。そこで、一枚の写真画像を左目画像とし、この左目画像とデプス画像に3D空間再生のための理論式を適用することで右目画像を生成する方法を考案した。この方法は、結局、左目画像の各画素値を、デプス値と理論式に基づいて x 軸方向に平行移動させることで右目画像を生成するものである。この方法で問題になることは、右目画像において、画素値が決定できない画素が生じることである。これは、隠蔽が生じている箇所で発生する。それらの箇所に対して、適切と思われる画素値を推定することにした。また、デプス画像の精度が不十分であるため、そのままでは、適切な3D画像を生成することができなかった。これを解決するために、手作業によってデプス画像を修正した。

提案手法を、ヘッドマウントディスプレイと Raspberry Pi を用いる3D画像表示システムで観察し主観評価した。また、Lytro アプリケーションが出力する3D画像と比較した。その結果、隠蔽が生じる箇所での不自然さは残るが、元シーンの立体感を再現することに成功した。また、Lytro アプリケーションが出力する3D画像は奥行き感が非常に少ないのに対し、提案手法は十分な奥行き感の再現に成功した。課題として、画素値を持たない画素の補間方法に検討の余地がある。

目次

1章 序論	1
2章 3D 立体映像技術	3
2.1 3D 画像の表示	3
2.1.1 ステレオグラムとステレオ画像	4
2.1.2 立体テレビ	7
2.1.3 ヘッドマウントディスプレイ	8
2.2 3D 画像の入力	9
2.2.1 ステレオカメラ	9
2.2.2 単眼ステレオカメラ	9
2.3 3D 空間の再生理論	10
3章 ライトフィールドカメラを用いた3D 画像の生成	12
3.1 ライトフィールドカメラの原理	12
3.2 ライトフィールドカメラの出力ファイル	13
3.3 3D 画像の生成手法	15
4章 実験と考察	18
4.1 Raspberry Pi を使った立体ディスプレイ	18
4.2 3D 画像の生成	19
4.3 生成画像の評価	23
4.4 Lytro アプリケーションが生成する3D 画像との比較	25
5章 結論	27
参考文献	28
謝辞	28
付録 1 研究で開発したプログラムの説明	28
付録 2 Raspberry Pi とソニーHMZ-T2 で3D 画像を観察する方法	30

1章 序論

人間の視覚は3次元の情景を見ると立体感を感じる。その要因の一つは、被写体までの距離によって、左目に写る像と右目に写る像にずれが生じることである。このずれを両眼視差とよび、視差を有する2枚の画像をステレオ画像とよぶ。左右に並べたステレオ画像を裸眼で観察して立体画像として認識することも可能である。しかし、この方法で立体視を行うには、事前に訓練が必要であり、また、立体視できる範囲が狭いなどの問題がある。通常は、立体ディスプレイを用いてステレオ画像を観察する。

普通の立体ディスプレイは適切な視差がある画像を、ディスプレイ上に重ね合わせた状態で表示する。そして、左目用の画像が左目だけに見え、右目用の画像が右目だけに見えるように制御する。これが両眼視差に基づく立体ディスプレイの原理である。立体ディスプレイには、両眼視差方式以外にもさまざまな方式があるが、実用化されているものは両眼視差に基づくものが多い。

立体ディスプレイに映す映像は、左目画像と右目画像が対になった映像である。これを撮影するカメラは、人間の左右の目に相当する2台のカメラを併置したカメラであり、ステレオカメラとよぶ。ステレオカメラで立体映像を撮影することは、同じ性能を持つ2台のカメラを、適当な間隔を隔てて配置し、2枚の画像を撮影することである。このとき、カメラ間の間隔やカメラの画角などのパラメータを適切に設定する必要がある。

ステレオ画像を両眼立体視により3D画像として観察する場合、再生される3D空間はステレオカメラによる撮影条件、立体ディスプレイの性能、観察者の特性に依存する。その主な要素としてステレオカメラ間の距離である基線長や立体ディスプレイの画面サイズ、ディスプレイを観察する視距離、観察者の瞳孔間隔などがある。再生される3D空間を理論的に評価する手順は以下のようになる。

- ①被写体の3次元座標からステレオカメラの画像座標への変換
- ②ステレオカメラの画像座標から立体ディスプレイ画面上の座標への変換
- ③立体ディスプレイ画面上の座標から認識される3D空間の座標への変換

本研究ではこれらの関係性を明確に示し、撮影した情景を、立体ディスプレイで忠実に再現することを目標とする。

通常のステレオカメラは2台のカメラを一つ筐体にまとめた装置である。一方、見かけ上は1台カメラと同様に単眼のカメラでありながら、ステレオ画像を撮影することができるカメラが存在する。本研究で用いるライトフィールドカメラも、そのようなカメラの一つである。Lytro（図1.1）は、米国のLytro社が開発し

たライトフィールドカメラで、従来のカメラと異なり、撮影時にピントを合わせる必要がないカメラとして知られている。Lytro は、カメラレンズに入射する光を光線に分解して記録し、記録された光線集合に対するコンピュータ処理によって、最終画像を計算するカメラである。この光線集合に対する演算によって、被写体までの奥行きを計算することや、ステレオ画像を計算することが可能になる。

Lytro で撮影した画像を、専用のアプリケーションを用いて書き出すと、奥行き情報を記録したデプス画像や 3D 画像に対応した静止画像(ステレオ画像)などを出力することができる。本研究では、撮影した画像とデプス画像を用いて視差を有するステレオ画像を生成した。そしてそれを、Lytro の専用アプリケーションが生成するステレオ画像と比較し評価した。



図 1.1 ライトフィールドカメラ Lytro (Lytro 社の Web サイトの画像を利用)

本論文は、次のように構成される。2 章で 3D 画像の表示装置と入力装置について述べ、3D 空間の再生理論を説明する。3 章で、ライトフィールドカメラ Lytro の原理を説明し、その出力ファイルを用いてステレオ画像を生成する方法について説明する。4 章で、本研究用に開発した立体ディスプレイを説明した後、具体的なステレオ画像の生成アルゴリズムの説明と、生成されたステレオ画像の評価を行う。5 章で結論を述べる。

なお、本論文では 3D 画像、立体画像、3D 立体映像、ステレオ画像など類似の用語を併用しているが、いずれも、左目画像・右目画像からなる 2 枚の画像で構成されるものを指している。

2 章 3D 立体映像技術

3D 立体画像の表示／入力技術にはさまざまなものが存在する。本章では、両眼立体視に基づく方式について説明する。

2.1 3D 画像の表示

2.1.1 ステレオグラムとステレオ画像

ステレオグラムは、立体感を持つように描かれた一対の図や絵で、眼の焦点を意図的に前後にずらすることで立体視することが可能なものである。この時、両目の網膜に映る像の違いを両眼視差と言う。ステレオグラムは両眼視差を利用して一対の画像を立体として認識させるものである。

裸眼でステレオグラムを観察する方法として、交差法(図 2.1(a))と平行法(図 2.1(b))がある。交差法は左眼で右の画像、右眼で左の画像見る方法で、画像面より近くで視線を交差させる方法である。平行法は右眼で右の画像、左眼で左の画像を見る方法で、画像面より遠くで視線を交差させる方法である。

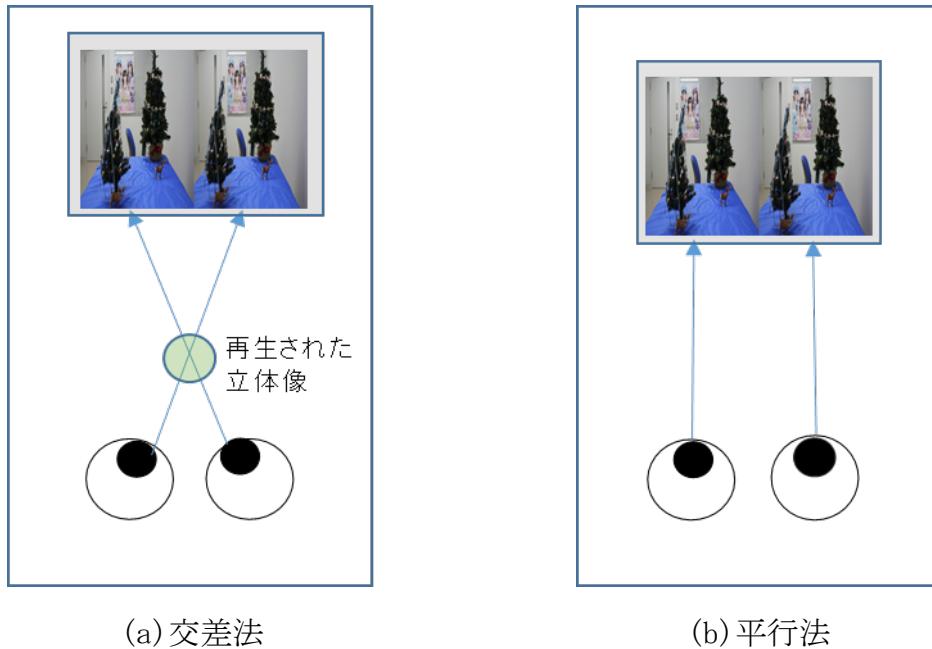


図 2.1 裸眼によるステレオグラムの観察

ステレオグラムは、図 2.1 のように、視差がある 2 枚の画像を左右に併置したもので、裸眼立体視に用いる。ステレオグラムをステレオ画像とよぶこともあるが、ステレオ画像という言葉はお互いに視差を持つ複数の画像を意味する。ステレオグラムのように左右に併置するものもステレオ画像であるが、立体テレビの画面上に重ね合わせて表示される 2 枚の画像もステレオ画像である。簡単なステ

レオ画像の入力方法は、固定された被写体に対して、カメラを平行移動させ左眼画像、右眼画像を撮影する方法である(図 2.2). この 2 画像を左右に併置するとステレオグラムになる.

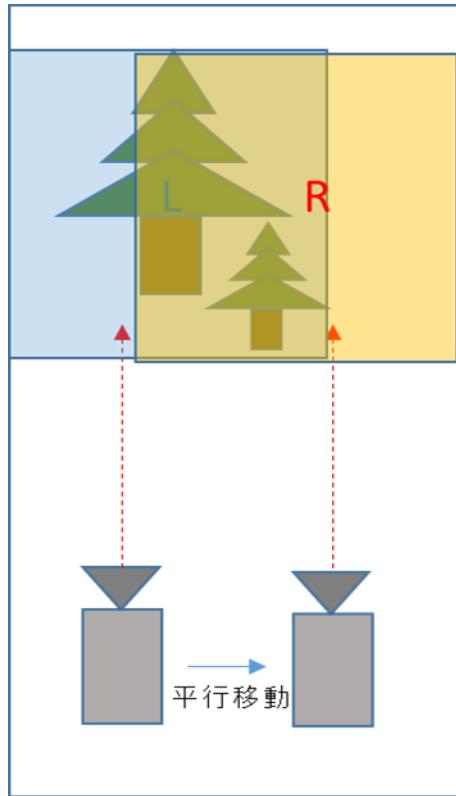


図 2.2 平行法によるステレオ画像の撮影

2.1.2 立体テレビ

立体テレビは、立体映像を表示することができるディスプレイを搭載したテレビである。2005 年頃に始まった 3D 映像ブームによって、立体テレビを用いて、ブルーレイによる 3D 映画の鑑賞や 3D 放送の視聴が可能になっている。ここでは、立体映像をテレビに伝送する放送方式について説明する。

立体映像の放送方式として、サイドバイサイド方式(図 2.3(a))やフレームパッキング 1080i 方式(図 2.3(b))がある。フレームパッキング 1080i 方式は左眼・右眼の両映像を 1920×1080 画素のインターレース映像として送る方式で、MPEG-4 MVC モードで記録した映像を、 1920×1080 画素のフルハイビジョンのまま伝送することが可能である。サイドバイサイド方式は、左眼・右眼の両映像を 960×1080 のインターレース映像として伝送する方式である。画像の横幅を $1/2$ に圧縮し、左右の視点から見た 2 枚の画像を横に並べ、その他は通常の 2D 画像と同様に放送する方式である。実用放送として進められている 3D テレビ放送はサイドバ

イサイド方式が主流である。ブルーレイ 3D にはフレームパッキングが用いられている。

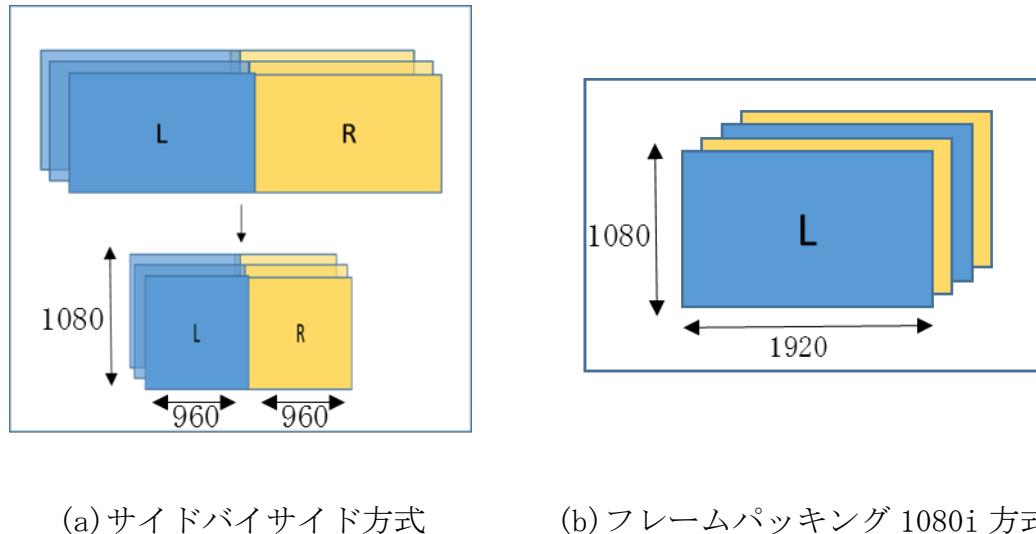


図 2.3 代表的な 3D 伝送方式

図 2.4 は解像度や視点数といった点で、3D 放送の今後の進化の見通しを示したものである。現在はサイドバイサイド方式と MPEG-2 を用いた放送が主流になっているが、今後は一部の衛星放送やネット経由の放送から H.264 を用いた 3D 放送が増える可能性が高いことを示している。

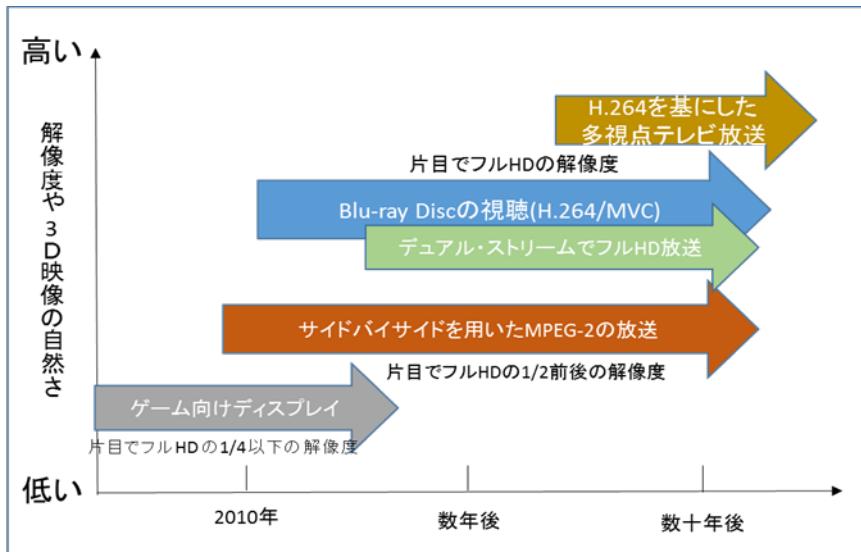


図 2.4 3D 放送の進化
(日系エレクトロニクス 2010 年 4 月 19 日号掲載記事より引用)

2.1.3 ヘッドマウントディスプレイ

ヘッドマウントディスプレイ(HMD)は頭部に装着するディスプレイである。HMDは左右の眼の前方に小型ディスプレイが配置されている。ディスプレイに両眼視差を持つ映像が表示されると、立体感を感じる。図2.5を用いて、装置の構成と再生像の関係を説明する。HMDは眼前にディスプレイが配置されているため、広い視野角を確保することができる。ここで、ディスプレイ面が遠方にあるように見せるため、接眼レンズを用いて、焦点位置を遠方にシフトする。HMDは立体ディスプレイとして用いるだけでなく、頭部の位置と方向を検出するヘッドトラッキングシステムを採用することで、人の動きに応じて映像が変化するインタラクティブなシステムを構築することが可能である。近年では、HMDの小型化や軽量化、高解像度化が進み、消費者向けの製品が低価格で市販されている。

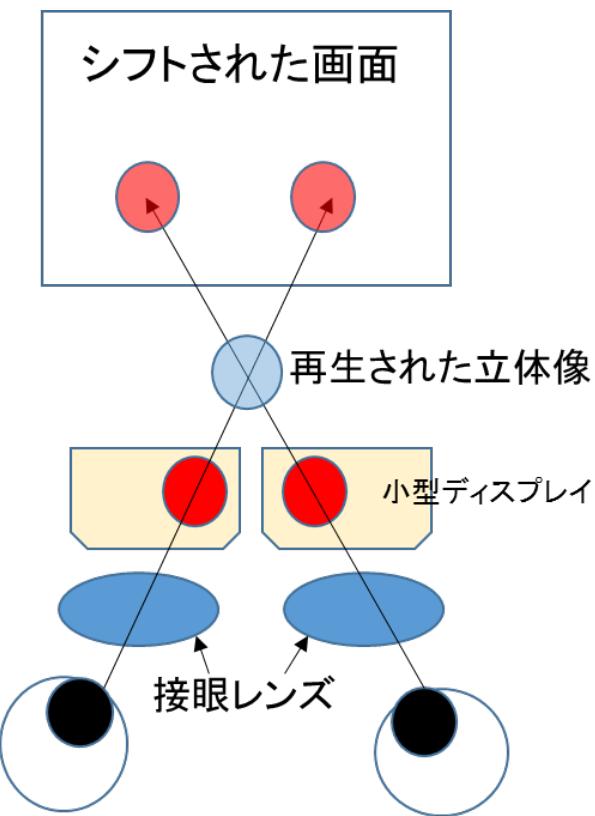


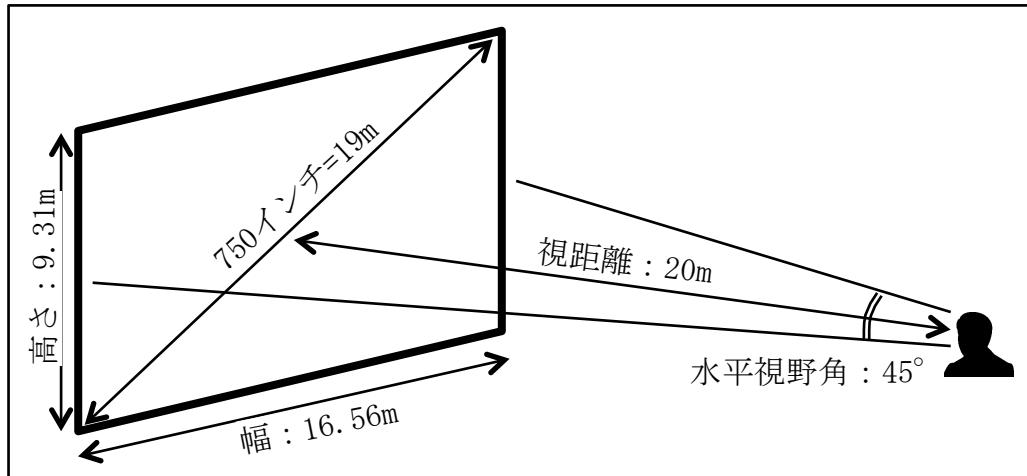
図2.5 ヘッドマウントディスプレイの構成

本研究では、sony 製の HMD である HMZ-T2(図2.6)を使用した。HMZ-T2を装着したときの観察条件は、観察者から 20m の距離に幅 16.56m × 高さ 9.31m の 3D ディスプレイを設置し、水平視野角を 45° にしたものと等価である。HMZ-T2 で立体画像を観察するためには、図2.7 で示すように、左眼画像と右目画像をサイドバイサイドの立体画像に合成し、それを通常の映像信号として HMZ-T2 に送る。



パネル	有機ELパネル×2枚
画素数	1280×720
アスペクト比	16:9
視野角	約45°
仮想画面サイズ	750インチ（仮想視距離約20m）

(a) Sony製HMD HMZ-T2と主な仕様



(b) HMZ-T2による観察条件

図 2.6 HMZ-T2 の外観, 仕様, 観察条件



図 2.7 HMZ-T2 で立体画像を表示するための処理

2.2 3D 画像の入力

2.2.1 ステレオカメラ

立体画像を入力するカメラをステレオカメラとよぶ。立体写真用のステレオカメラは、銀塩写真技術が発明された直後にあたる 1850 年代には考案されている。現在では、デジタル方式のステレオカメラが実用化されており、図 2.8 に一例を示す。また、3D 映画を撮影する時に用いる業務用カメラの例を図 2.9 に示す。これは、2014 年に IMAX 社が発表したデジタル 3D カメラであり、65mm の撮像素子を用いて 4K の立体映像を撮影することができる。



図 2.8 市販されているデジタルステレオカメラの例
(富士フィルム FinePix REAL 3D W3 の Web サイトの画像を利用)



図 2.9 映画撮影用のステレオカメラ
(https://www.imax.com/imax3dcamera/links/IMAX_3D_Digital_Camera.JPG を利用)

2.2.2 単眼ステレオカメラ

通常のステレオカメラは、図 2.8、図 2.9 に示すように、シーンを撮影するためのレンズ・撮像素子を独立に 2 系統持っている。これに対して、見かけ上、1 系統のレンズ・撮像素子でありながら、立体画像の撮影が可能なカメラを単

眼ステレオカメラとよぶ。これが可能であるのは、レンズに入射する光線をレンズ口径の左半分を通過するものと、右半分を通過するものに分離して、記録するからである。図 2.10 を用いて、単眼ステレオカメラの動作原理を説明する。通常のカメラと異なる点は、撮像素子が左側用 PD (Photo Detector) と右側用 PD に分かれていること、撮像素子の直前にマイクロレンズが配置されていることである。右側用 PD が検出する光線は、レンズの右側（図 2.10 では上側）を通過する光線であり、左側 PD が検出する光線は、レンズの左側を通過する光線である。したがって、右側用 PD だけを使って生成される画像は、撮影レンズの右側部分から被写体を観察した像になり、左側用 PD だけを使って生成される画像は、撮影レンズの左側から被写体を観察した像になる。

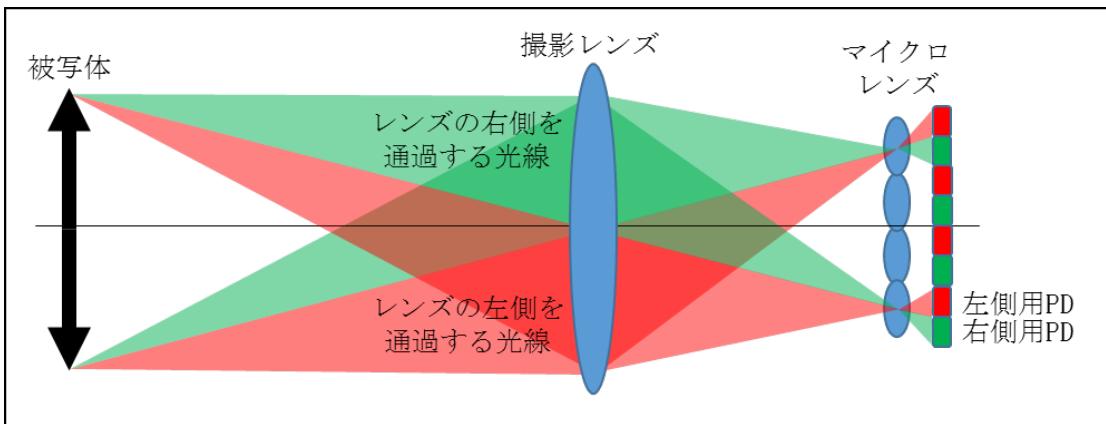


図 2.10 単眼ステレオカメラの動作原理

2.3 3D 空間の再生理論

ステレオ画像を両眼立体視した場合に知覚・再生される 3D 空間の理論的な位置を、撮影・呈示条件から算出することが可能である。計算手順は以下のようになる。

- ①被写体の 3 次元座標をステレオカメラの撮像素子上の座標へ変換する
- ②ステレオカメラの撮像素子上の座標を立体ディスプレイの画面上の座標へ変換する
- ③立体ディスプレイの画面上の座標を観察者の 3D 空間の座標へ変換する

本研究では平行法で観察するため、その場合の一般式を説明する。図 2.11 に平行法でステレオ画像を撮影する様子を示す。2 台のカメラを基線長の間隔で平行に配置する。Z 軸上の「ずれ量 0 とする距離」で示した点は、撮影した 2 枚の画像を立体ディスプレイに表示するときに、画面上でのずれ量を 0 にする点である。この点をカメラで撮影した位置と、光軸が画像面と交わる位置の間

隔を $s (\geq 0)$ とする。撮影対象物の座標を (X_0, Z_0) とし、 X_{cr} を右カメラでの観察位置、 X_{cl} を左カメラでの観察位置とすると式(2-1)の関係がある。

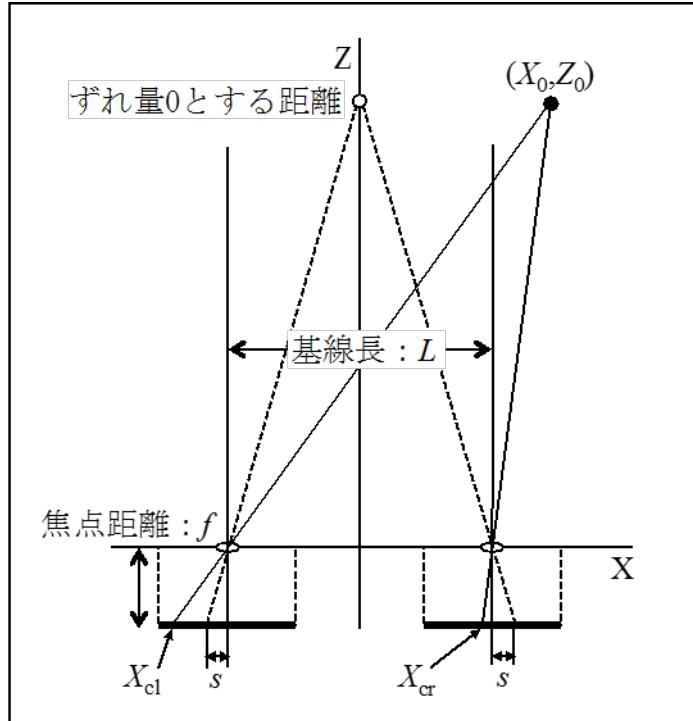


図 2.11 平行法による 3D 撮影

$$X_{cl} = -s + f \tan\left(\arctan\left(\frac{X_0 + L/2}{Z_0}\right)\right) \quad \dots \quad (2-1)$$

$$X_{cr} = s + f \tan\left(\arctan\left(\frac{X_0 - L/2}{Z_0}\right)\right)$$

次にカメラの画像面を拡大してディスプレイ画面に表示する。画面上でのこれらの表示位置を X_{sr} 、 X_{sl} とすると X_{sr} 、 X_{sl} は X_{cr} 、 X_{cl} を画面サイズに合わせて拡大したものである。それが以下の式である。

$$X_{sl} = \text{撮像素子に対する画面サイズの拡大率} \times X_{cl} \quad \dots \quad (2-2)$$

$$X_{sr} = \text{撮像素子に対する画面サイズの拡大率} \times X_{cr}$$

最後に、 X_{sr} 、 X_{sl} を人間が観察したときに再生される 3D 空間の座標を X_i 、 Z_i とすると、これらは式(2-3)の関係がある。

$$X_i = \text{瞳孔間隔} \times (X_{sr} + X_{sl}) / \{2(\text{瞳孔間隔} - \text{画面上のずれ量})\} \quad \dots \quad (2-3)$$

$$Z_i = \text{視距離} \times \text{瞳孔間隔} / (\text{瞳孔間隔} - \text{画面上のずれ量})$$

3章 ライトフィールドカメラを用いた3D画像の生成

3.1 ライトフィールドカメラの原理

ライトフィールドカメラはレンズに入射する光を光線に分解して記録するカメラである。記録された光線集合を計算処理することで、最終的な写真画像を生成する。この計算処理は、光に対するレンズの作用と撮像素子による像の形成をシミュレーションすることに相当する。例えば、レンズと撮像素子の間隔（像面距離）を変更して再計算すると、ピント位置が異なった画像が生成される。

下図はライトフィールドカメラが光線を記録する原理を説明した図である。Aにある被写体から発せられた光線は主レンズ上で屈折しマイクロレンズを通過し、各マイクロレンズがカバーする画素に記録される。これらの画素は光線の明るさを記録するとともに、画素の位置とマイクロレンズの位置を結んだ光線に対応している。一つのマイクロレンズに対応する全ての画素を平均化することでAの被写体に焦点を合わせた粗い写真画像が生成される。

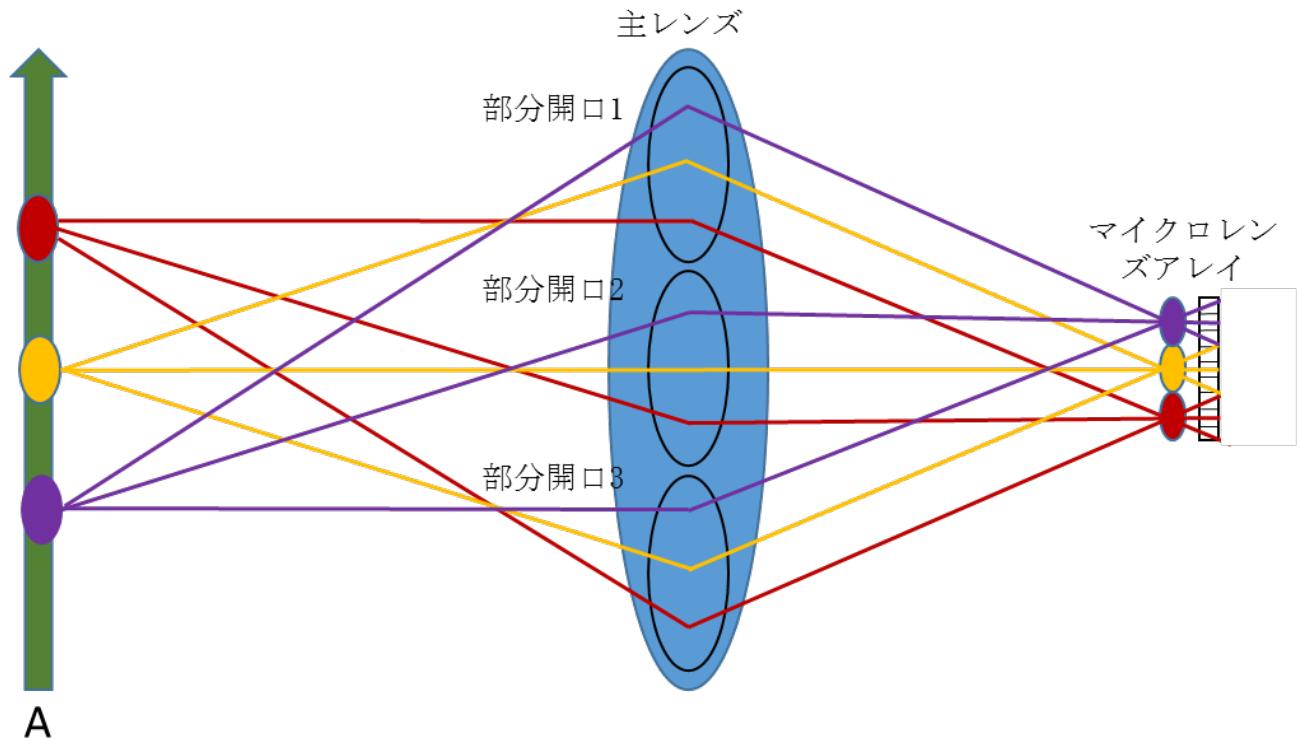


図 3.1 Lytro が光線を取得する原理

3.2 ライトフィールドカメラの出力ファイル

Lytro で記録した光線の集合である生データは, Lytro アプリケーションによって数種類のファイルとして出力することが可能である. 本研究では Lytro Desktop4.0.4 による出力ファイルを利用する. Lytro Desktop が出力するファイルの一覧を表 3.1 に示す. これらの画像の例を図 3.2~図 3.4 に示す.

表 3.1 Lytro Desktop が出力するファイル

2D formats…図 3.2	・リリフォーカス画像(jpg, tif)
3D formats…図 3.3	・フルカラーの jpg 立体画像 ・Red/Cyan3D 画像(jpg)
Lightfield formats	Lytro が捉えた全てのデータを含むライトフィールドデータ(lfr)
Editable Depth Map…図 3.4	編集可能なデプスマップ(png), 画面全体にピントの合った画像(tif), 深度マップのデータ情報を含むテキストファイル(txt)

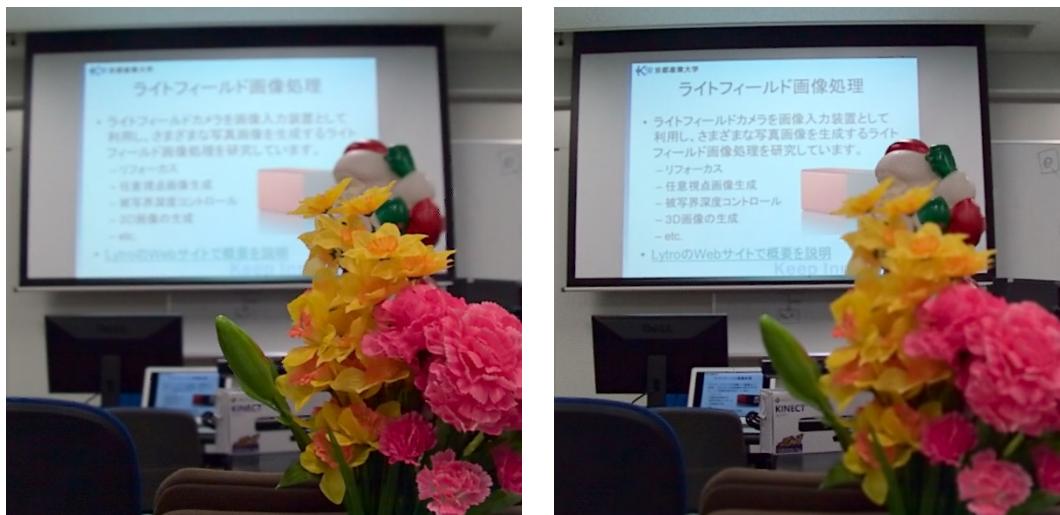


図 3.2 リリフォーカス画像
(左の画像は手前に、右の画像は奥にピントがあっている)

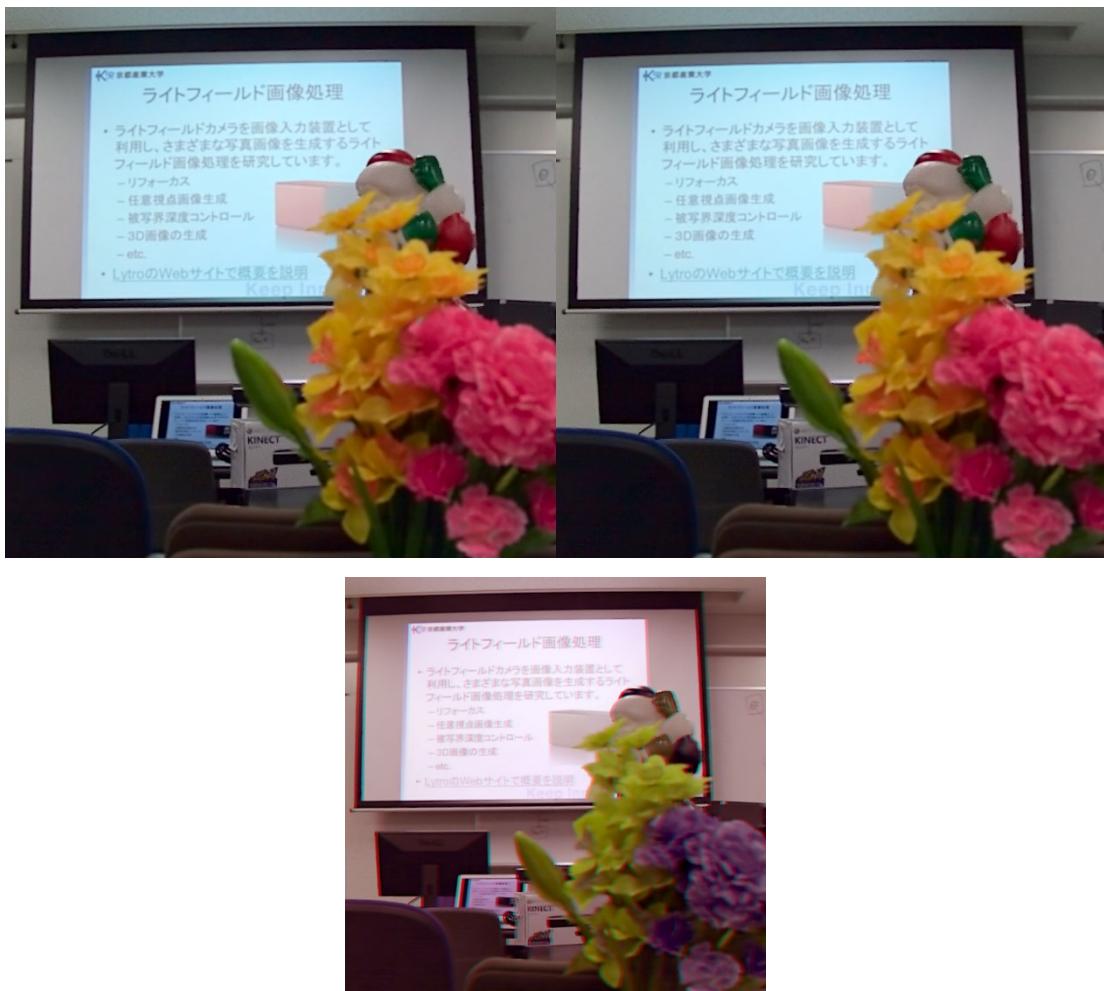


図 3.3 フルカラー立体画像(上)と Red/Cyan3D 画像(下)

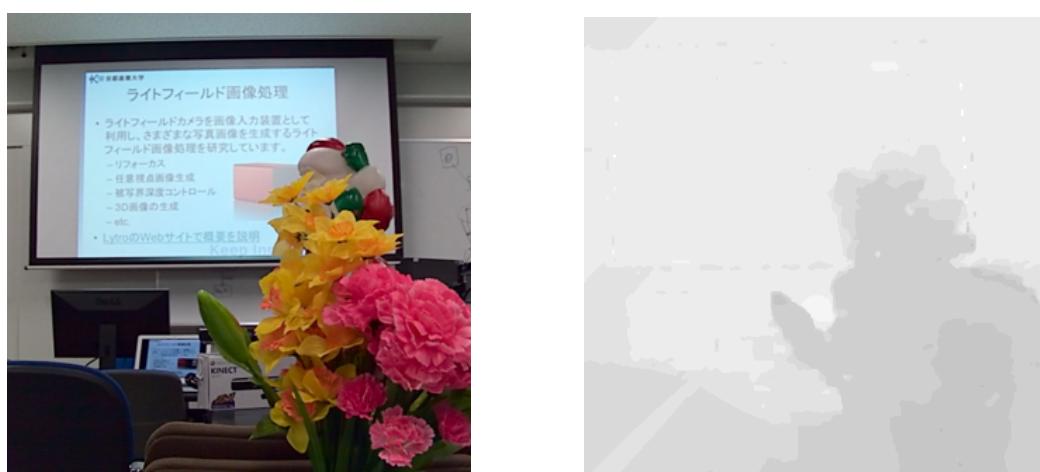


図 3.4 画面全体にピントが合った画像とデプスマップ(330×330)

3.3 3D 画像の生成手法

Lytro の出力ファイルを用いて視差を有するステレオ画像を生成する方法として、大きく分けて 2 つの方法が考えられる。Lytro で記録した生データを処理すると、視点の異なる多数のステレオ画像に分解することができる。これら多数のステレオ画像を元に、左眼画像と右眼画像を生成するのが第一の方法である。第二の方法は、ライトロアプリが出力するデプス画像を利用し、基準となる画像に対して視差を持つ画像を生成する方法である。第一の方法は、ステレオ画像間の視差を制御することが容易ではない。第二の方法は、ステレオ画像間の視差量を容易に制御することができる。このため、今回は第二の方法を採用する。基準となる画像に対して視差を持つ画像を生成する流れを図 3.5 に示す。リフォーカスして出力した画像を左眼画像とし、デプス画像の画素値を基にして基準画像の画素位置をずらすることで、右眼画像を生成する。

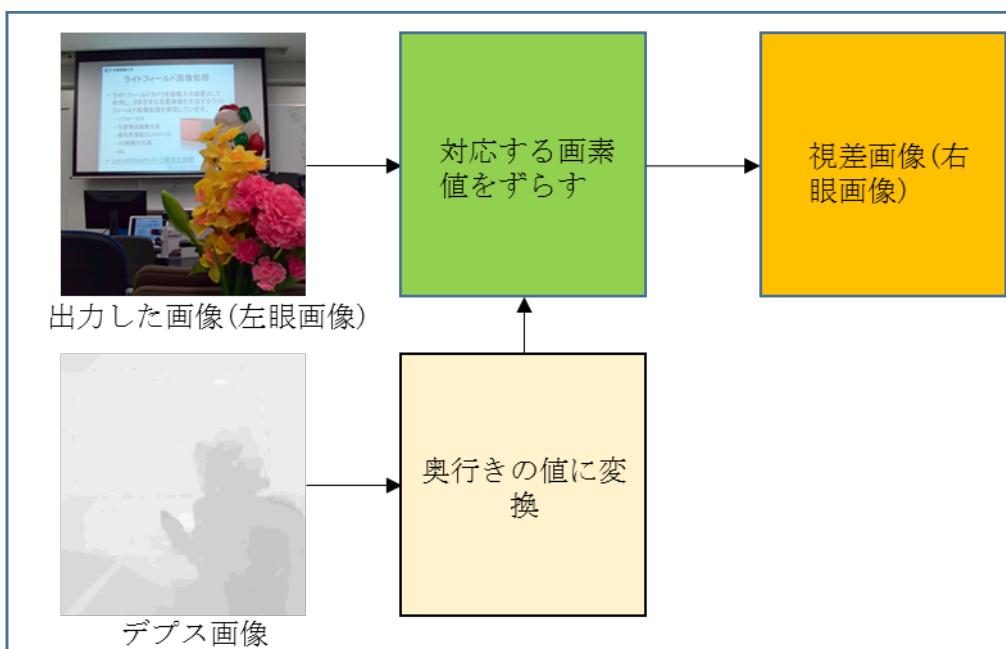


図 3.5 デプス画像を用いた視差画像の生成

ライトロアプリが出力するデプス画像の画素値と奥行き距離の関係を調べた。図 3.6 のように被写体の正面に Lytro を設置し、カメラの先端から被写体までの距離と得られたデプス値の関係を計測した。今回計測した距離は、被写体からレンズまでの距離が 10~100cm の間を 10cm 刻み、と 150cm, 200cm の計 12 回である。得られたデータを図 3.7 に示す。デプス値と奥行き距離の間には反比例の関係があると考えられる。図 3.7 の測定結果を参考にして、 $\text{デプス値} = k_1 - k_2 / \text{奥行き距離}$ となることを想定した。この式の定数 k_1, k_2 を図 3.7 の測定結果から最小二乗法で推定することで、式(3-1)の結果を得た。

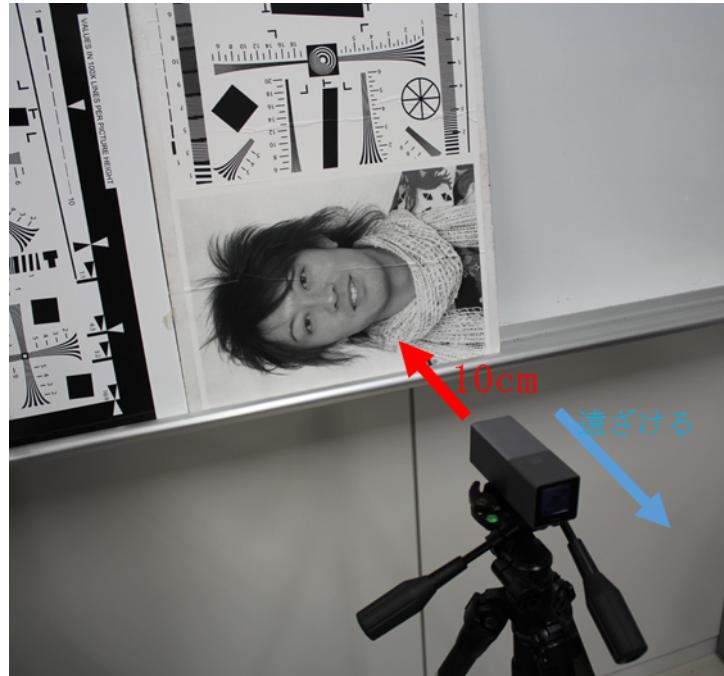
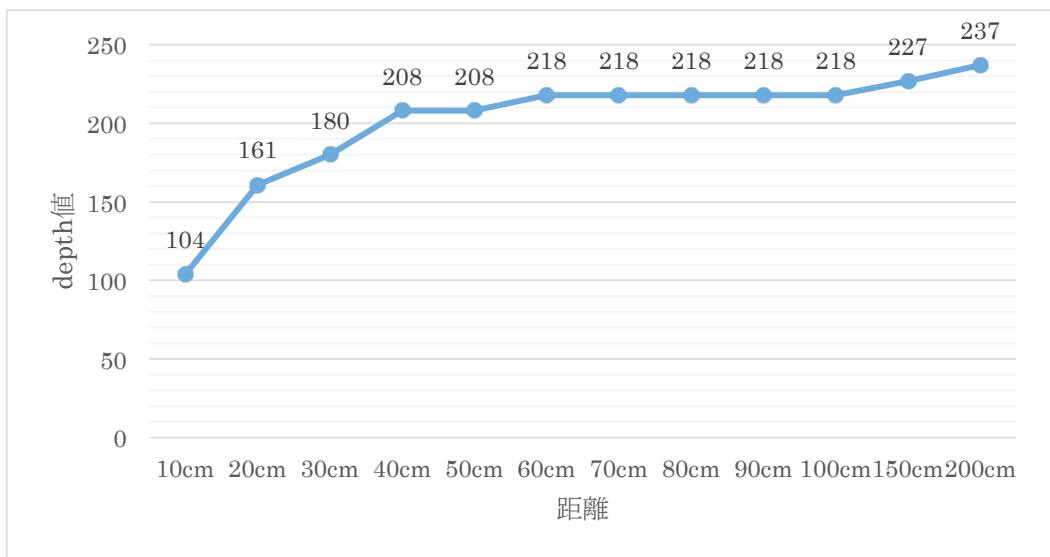


図 3.6 デプス値と奥行き距離の関係を調査する実験の状況



10cm	20cm	30cm	40cm	50cm	60cm	70cm	80cm	90cm	100cm	150cm	200cm
104	161	180	208	208	218	218	218	218	218	227	237

図 3.7 デプス値と奥行き距離の関係

$$\text{デプス値} = 235.70 - 13637.55 / \text{奥行き距離(mm)} \quad \dots \quad (3-1)$$

式(3-1)より

$$\text{奥行き距離(mm)} = 13637.55 / (235.70 - \text{デプス値}) \quad \dots \quad (3-2)$$

式(3-2)よりデプスの値から実距離を求めることが可能である。

次に距離に対応した画素分だけずらす処理を行うため式(2-3)を用いる（式(2.3)をここに再掲する）。

$$X_i = \text{瞳孔間隔} \times (X_{sr} + X_{s1}) / \{2(\text{瞳孔間隔} - \text{画面上のズレ量})\} \quad \dots \quad (2-3)$$

$$Z_i = \text{視距離} \times \text{瞳孔間隔} / (\text{瞳孔間隔} - \text{画面上のズレ量})$$

今回の実験での観察条件は図 2.6(b)で示したように仮想視距離が 20m である。つまり 20m 先のディスプレイ上で映る被写体の視差は 0 である（視差がない）。瞳孔間隔を 6.5cm として式(2-3)下の式に上記の条件をあてはめると

$$20m \text{ 先の画面上のズレ量} = 6.5 - (20000 \text{ (mm)} * 6.5) / \text{奥行き距離} \quad \dots \quad (3-3)$$

Lytro が出力する写真画像のサイズは 1080 × 1080 画素であり、20m 先の仮想ディスプレイ面の高さが 9.31m であることから、ずらす画素を α 画素とすると

$$\alpha = (1080 / 9.31) * 20m \text{ 先のズレ量} \quad \dots \quad (3-4)$$

となる。式(3-2), 式(3-3), 式(3-4)を用いることで、デプス値からズレ量を算出できる。デプス値が 235.70 以上の値を取った場合奥行き距離が負となるため今回は 235.70 以上の値を取った場合、奥行き感は 20m とする（被写体までの最遠距離が 20m）。

4 章 実験と考察

4.1 Raspberry Pi を使った立体ディスプレイ

立体ディスプレイに3D画像を表示する場合、サイドバイサイドのステレオ画像を、HDMI端子を持つパソコンで再生し、HDMI端子を3Dディスプレイに接続する。本研究では3DディスプレイとしてsonyのHMZ-T2を用い、再生するパソコンにRaspberryPi(図4.1)を用いる。

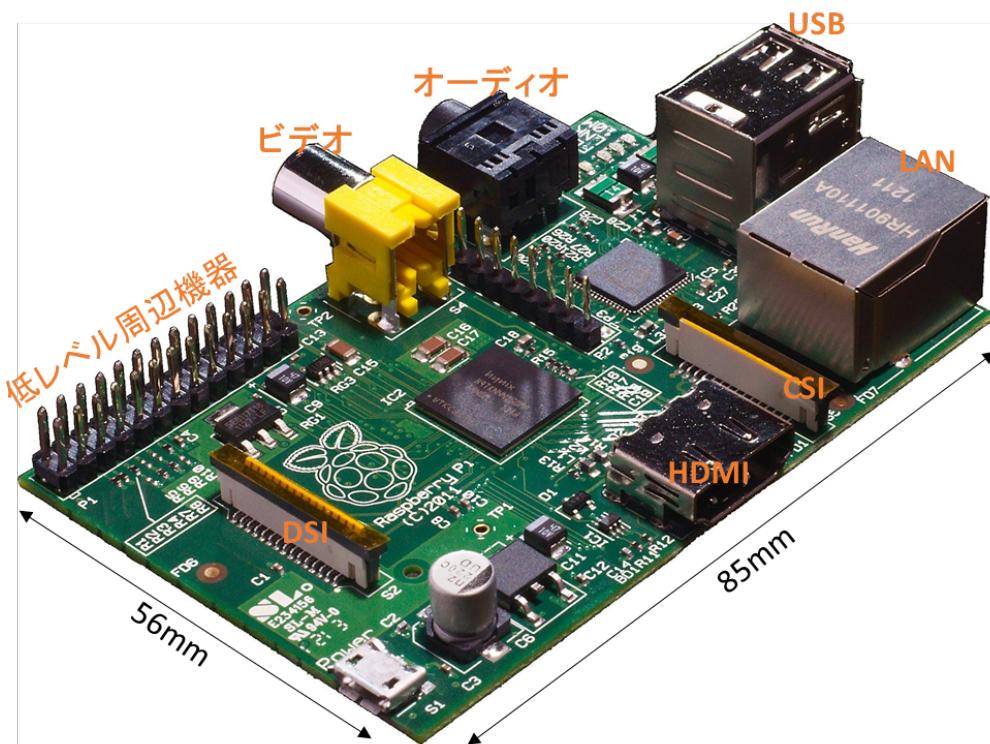


図4.1 Raspberry Pi のインターフェース

Raspberry Piはラズベリーパイ財団が開発した小型コンピュータで子供や学生などにコンピュータ技術を学ばせることを目的として作られたものである。手ごろな価格で販売されており、容易に入手可能である。このRasberry Piを用いることで、簡単に3D画像をHMZ-T2に表示することが可能である。

システム構成を図4.2に示す。Rasberry Piのディスプレイ表示に用いるHDMI信号を分配器で2系統に分け、一方を通常のディスプレイに、他方をHMZ-T2に表示する。HMZ-T2の3D方式として左右分割方式を選択し、2章の図2.7で生成したサイドバイドサイドステレオ画像をVLCメディアプレイヤーで再生することで両眼立体視が可能となる。

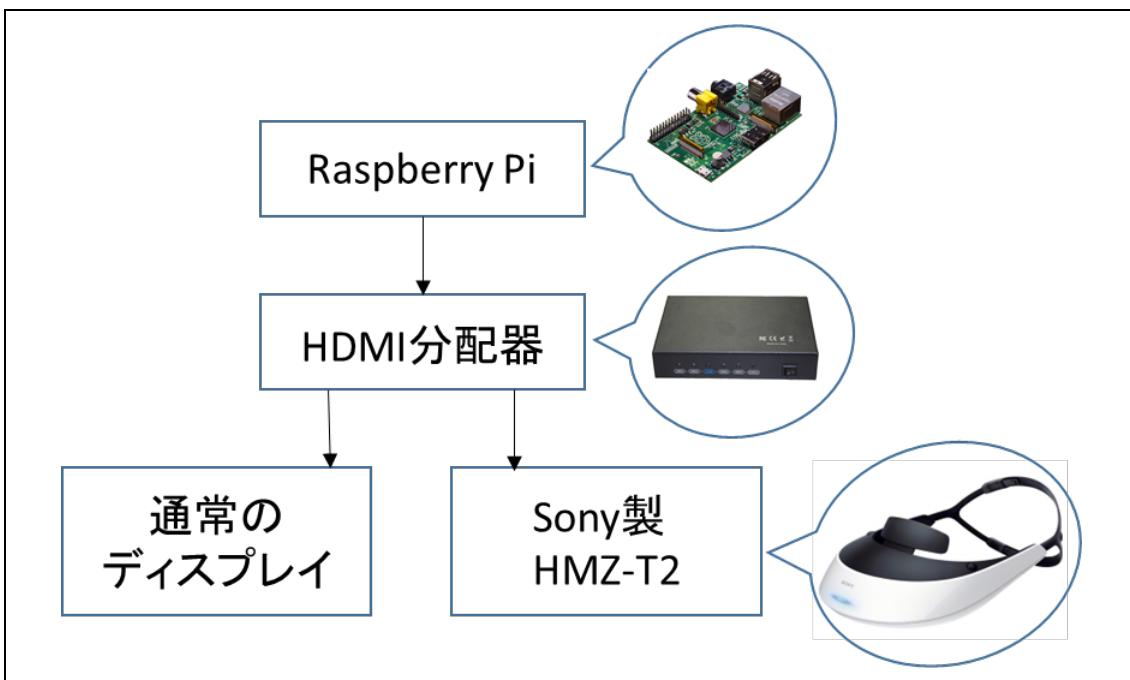


図 4.2 立体ディスプレイのシステム構成

4.2 3D 画像の生成

図 4.3 はライトロアプリが出力する画像とそれに対応するデプス画像である。デプス画像にはノイズが生じている。特に、被写体の境界と特徴が少ない被写体の表面にノイズが多い。このデプス画像を前述した式に適用し、視差画像を生成すると、図 4.4 の画像が得られた。

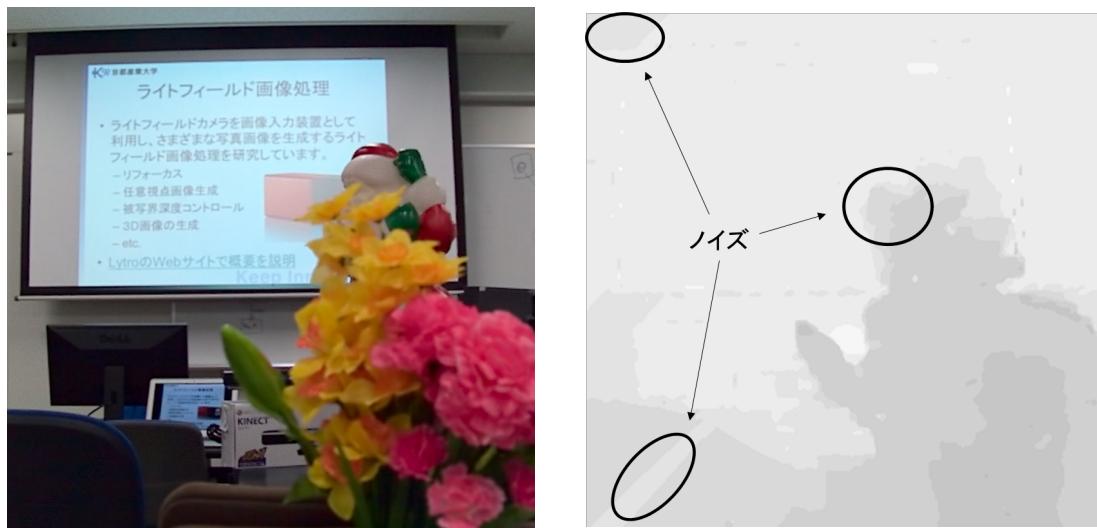


図 4.3 ライトロアプリが出力する画像と対応するデプス画像

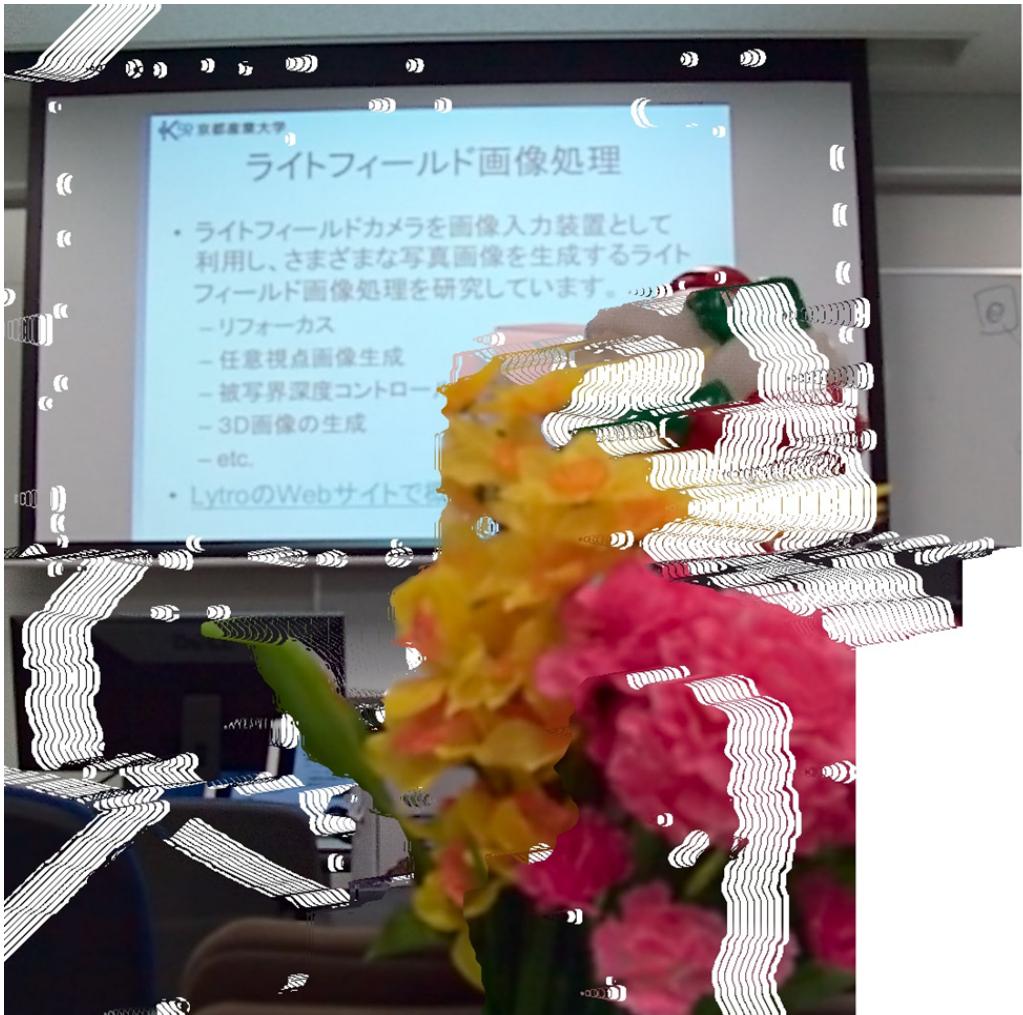


図 4.4 ライトロアプリが出力するデプス画像を用いた視差画像

図 4.4 を見ると、デプス画像上でノイズが発生している部分は正確な距離情報を読みとることができないため、画素値の移動に間違いが発生している。また、白くなっている箇所は、画素値が計算できていない。これは、前景の被写体が後景の被写体を隠蔽している箇所であり、今回の方では、原理的に計算が不可能な箇所である。これらの問題に対応するため、まず、画像編集ソフトウェア (PhotoShop) を用いて被写体の領域ごとにデプス画像を修正した。加工して得られたデプス画像と対応する領域のデプス値を図 4.5 に示す。

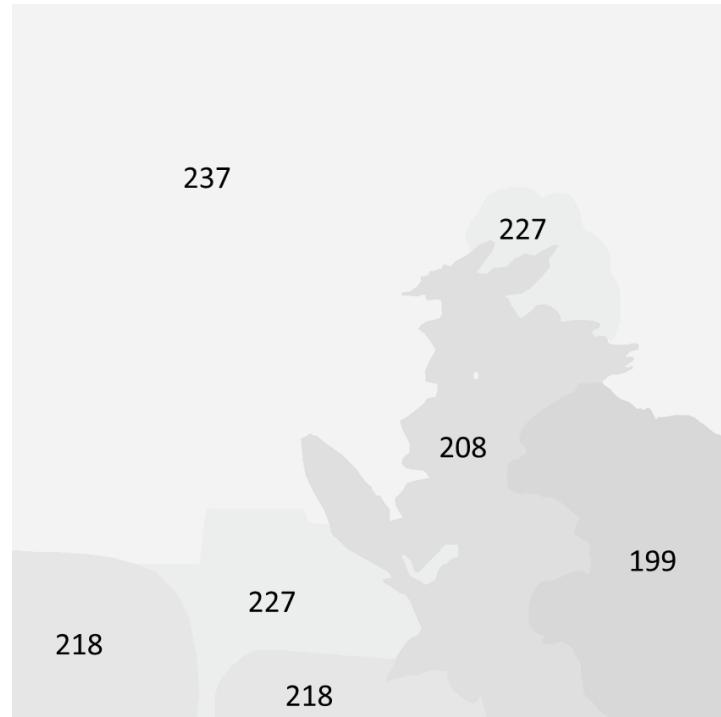


図 4.5 加工したデプス画像

図4.5のデプス画像を用いて視差画像の生成を行うと図4.6の結果が得られた。



図 4.6 加工したデプス画像から生成した視差画像

図 4.6 の白くなっている部分は右眼から見えていない部分である。左眼画像として使用している元画像からはデータを得ることができない。本実験では次の 2 つの方法で画素の補間を試みた。

- ① データのない画素を単色（黒色など）で塗りつぶす。
- ② データのない画素を周りの色情報から補間する。

1 番目の方法はデータのない画素を黒($R, G, B=[0, 0, 0]$)で単純に補間するもので、2 番目の方法はデータのない対象の画素 4 近傍のうち最もデプス値が遠い画素の色で補間する方法である。それぞれの得られた結果を図 4.7, 図 4.8 に示す。

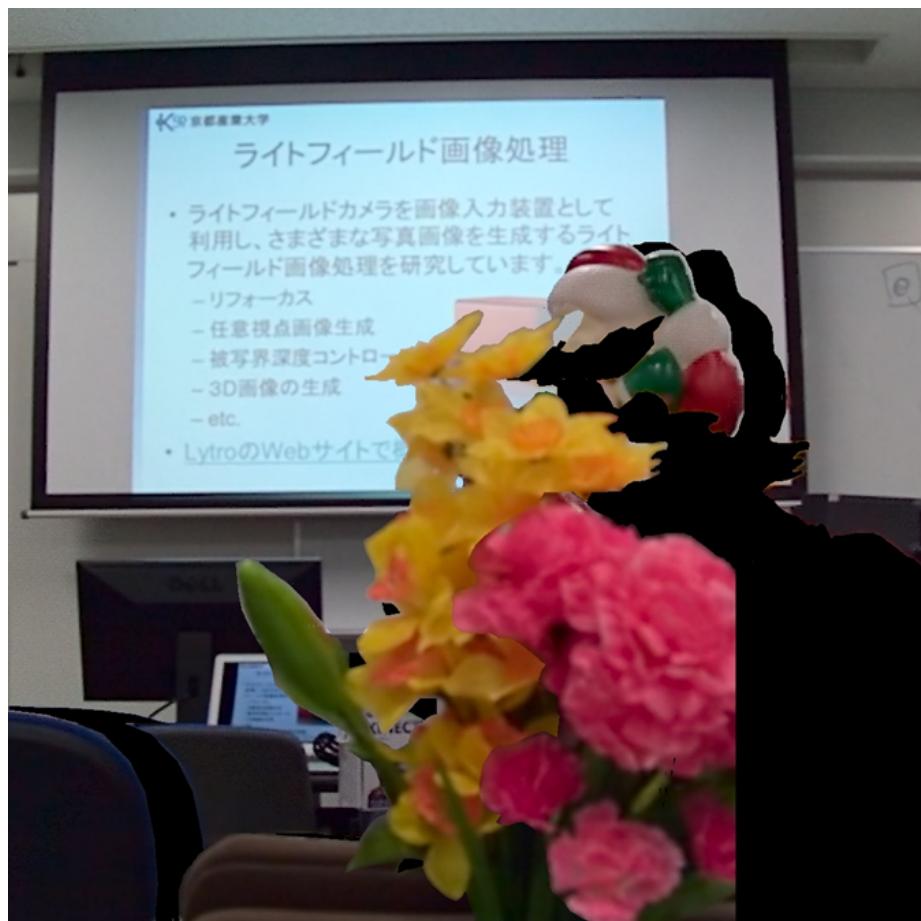


図 4.7 方法 1 で補間した視差画像

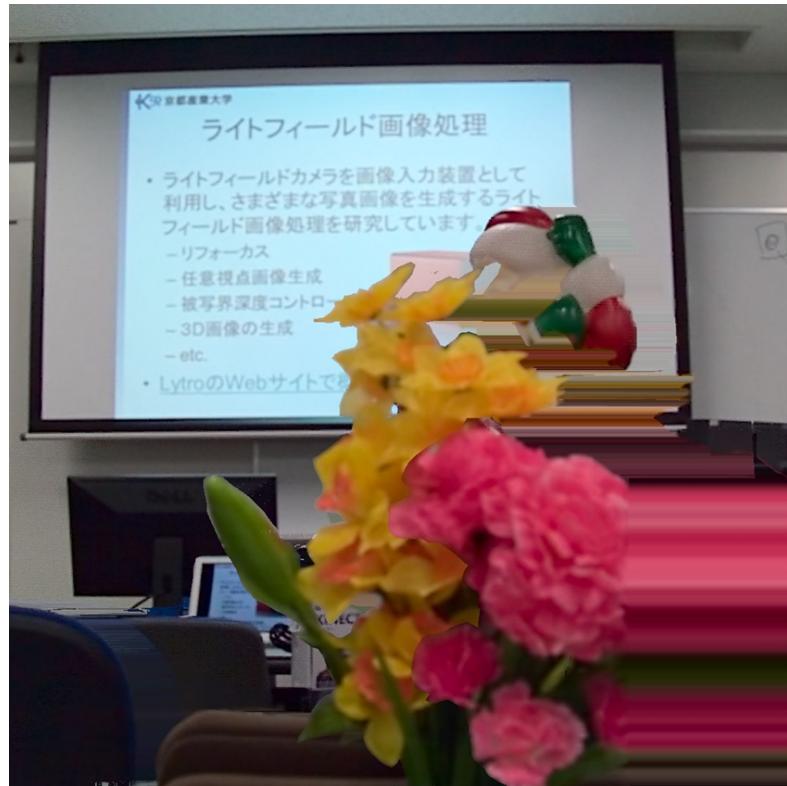


図 4.8 方法 2 で補間した視差画像

4.3 生成画像の評価

元画像を左眼画像、生成した視差画像を右眼画像としたステレオ画像を図 4.9 に示す。



図 4.9 生成したステレオ画像

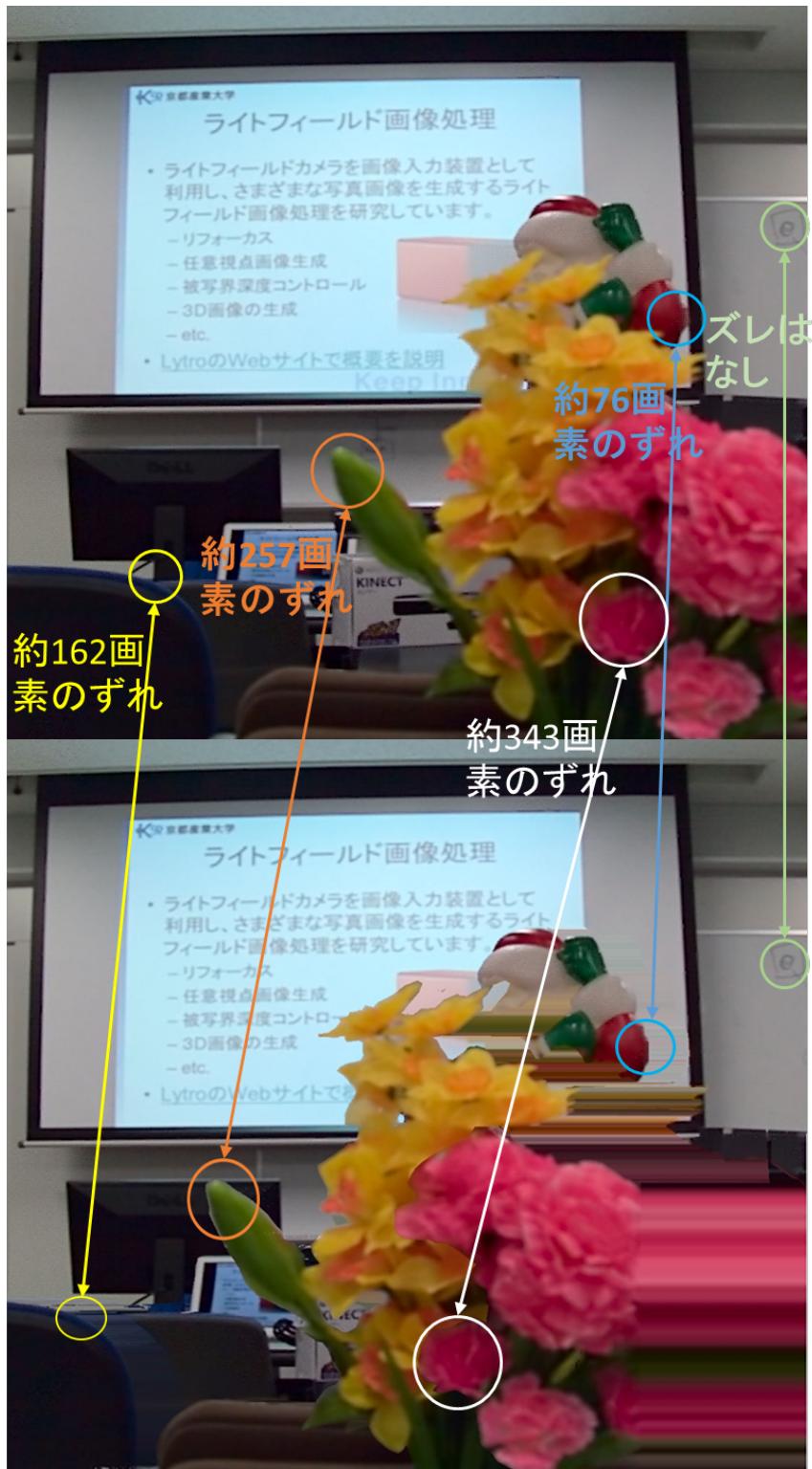


図 4.10 生成したステレオ画像の視差

図 4.10 は被写体ごとの画素のずれを示したものである。2枚の画像を比べると手前のピンク色の花で約 343 画素ずれており、被写体が遠くになるにつれてそれが小さくなっていき、一番奥の壁に対するずれは 0 である。ヘッドマウンドデ

ィスプレイでステレオ画像を観察すると、一番手前のピンクの花が 1m 程度の距離に立体視され、次に黄色の花、椅子、サンタクロース、スクリーンの順に遠近感を感じることができた。一番奥のスクリーンは、ヘッドマウントディスプレイで観察できる仮想画面上に立体画像が再生されているように感じることができた。これらのことから、立体画像の再生される位置は元のシーンと同様であり、3D 空間の再生理論による立体画像の生成ができたといえる。しかしながら、実際に撮影したシーンと比べると、本来、ピンクの花と黄色の花はひとかたまりになっているにも関わらず、ヘッドマウントディスプレイで観察すると、ピンクの花と黄色の花の間には距離感が生じている。また、奥のスクリーンは実際のシーンだと約 5m 先にあるがヘッドマウントディスプレイによる観察では、20m 先にあることになる。これらの問題は、ライトロアプリが出力するデプスマップが精密でないことが原因である。距離と視差の関係を求める実験（図 3.7）によると、Lytro が奥行きを測れる限界は 2~3m である。それ以上の奥行きではデプスの値は変わらない。今回の実験では、立体画像の再生空間がわかり易くなるように、一定の距離以上の被写体は意図的に 20m 先のディスプレイに再生されるよう調節した。また視差画像を生成した際に元画像からは見えていない部分を補間する手法としていくつかの方法を検討したが、いずれもヘッドマウントディスプレイで観察した際に違和感がある。比較的にマシであると感じたのが黒色で画素を塗り潰す方法（図 4.7）である。この場合、黒い部分が影に見えるため、多少の違和感はあるものの、容易に立体視が可能であった。

4.4 Lytro アプリが生成する 3D 画像との比較

Lytro が出力する 3D 画像には Red/Cyan 3D 画像とフルカラー 3D 画像の 2 つがある。Red/Cyan 3D 画像は赤青メガネを用いて立体視する。フルカラー 3D 画像は、水平解像度が 1/2 のサイドバイサイド画像に変換することで、図 4.2 のシステムで立体視することができる。

図 4.11 はライトロアプリにより出力されたフルカラー 3D 画像である。ある特徴点を決め、特徴点に対応する 2 つの画像の座標値を比較したところ、図 4.11 のズレが見られた。この 3D 画像も、提案手法で生成した視差画像同様に、近距離に映る被写体ほど視差のズレが大きく、後方に行くにつれ視差のズレが小さくなっている。ズレ量は一番手前の花で 10 画素、一番奥のスクリーンで 7 画素であり、最近と最遠の被写体でわずか 3 画素のずれしかない。ヘッドマウントディスプレイで観察してもあまり立体感を得ることができなかった。主観的な感想であるが、立体感に関しては、提案手法がはるかに優れている。

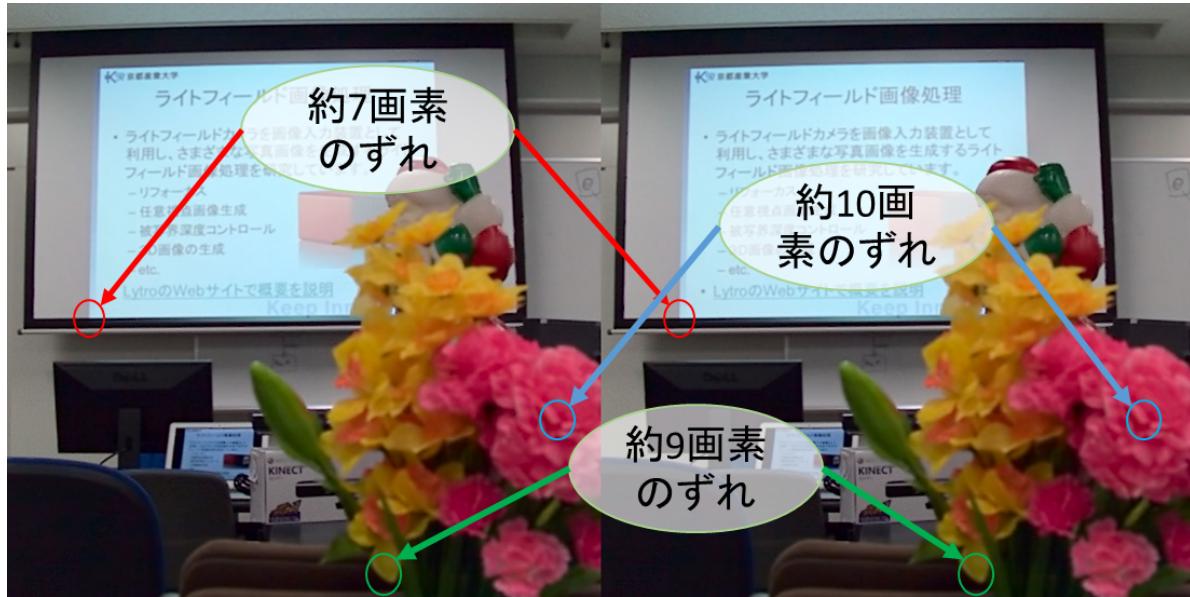


図 4.11 Lytro が生成する視差画像

5 章 結論

ライトロアプリが出力する写真画像とデプス画像を用いて、視差画像の生成を行った。ライトロアプリが生成するデプス画像には多くのノイズが見られたため、ノイズを取り除いたデプス画像を生成する必要があった。生成したデプス画像を使い、デプス値を距離に変換し 3D 空間の理論式を用いることで、被写体までの距離を考慮した視差画像の生成を行った。生成したステレオ画像をヘッドマウントディスプレイで観察したところ、概ね期待通りの立体画像生成を行うことができた。また、ライトロアプリが生成する 3D 画像と比較したところ、ライトロアプリが生成する 3D 画像よりも奥行き感を得ることができた。

しかし、生成された立体画像には現実の撮影シーンとは異なる部分がある。その原因の一つは、Lytro が計測できる奥行き精度が不十分なことである。遠くの被写体を立体画像として生成しようとした場合、一定距離以上の被写体の立体再生空間を細かに制御することは難しい。また今回は観察する条件として仮想視距離が 20m であるヘッドマウントディスプレイを用いたため、立体感は得られたものの現実のシーンのような再現ができなかったことも考えられる。

課題とするべき点としては、現実のシーンにより近づけるために Lytro におけるデプスと距離の関係をもっと細かく算出することで被写体の位置関係を正確にすること、視差画像を生成した際に生じるデータのない(左眼画像からは見えない)部分の最良の補間方法を提案する必要がある。

謝辞

本研究を進めるにあたり、様々な深い御指導を賜りました担当教授の蚊野浩先生に深く感謝申し上げます。

参考文献

- [1] Lytro 社の Web サイト, <https://www.lytro.com>
- [2] LYTRO SUPPORT, <https://support.lytro.com/hc/en-us>
- [3] 河合隆史, 盛川浩志, 太田啓路, 安部信明(2010)「3D 立体映像表現の基礎 – 基本原理から製作技術まで-」, オーム社
- [4] 竹生修己(2012)「現代電子情報通信選書『知識の森』画像入力とカメラ」, オーム社
- [5] 竹村裕夫(2014)「映像情報メディア学会誌」No. 5, pp. 399–404, 映像情報メディア学会

付録 1 研究で開発したプログラムの説明

【プログラム名】

depth-sisa.cpp

【内容】

このプログラムは元画像とデプス画像を入力し、視差画像を生成する。まず、デプス画像のデプス値を距離に変換することで被写体までの距離を求める、求めた距離に応じた画素単位のズレ量を元画像に適用することで視差画像を生成する。生成した際に画素の値が 0 になる画素にはその対象画素の 4 近傍を探索し一番デプス値の遠い色情報で補間する。

付録 2 Raspberry Pi とソニーHMZ-T2 で 3D 画像を観察する方法

1. 概要

左目画像・右目画像からなる 3D 画像を、Raspberry Pi とソニー製ヘッドマウントディスプレイ HMZ-T2 で観察する方法を説明する。

2. はじめに

左目画像・右目画像のペアからなる 3D 画像を両目に提示すると、2枚の画像が脳内で融合され、立体物として認識される。これを行う最も簡単な方法は、2枚の写真をディスプレイかハードコピー上に横並びに置き、平行法あるいは交差法で観察することである。左目が左目画像だけを、右目が右目画像だけを見ることが理想であるが、この方法では両方の画像が両目に入ってしまうため、立体視することが容易でない。左目の視線を左目画像の方向に、右目の視線を右目画像の方向に向ける必要があり、それができるようになるにはトレーニングが必要である。このように平行法・交差法による立体視は容易でなく、また、2枚の画像を併置するため、視野の広い画像を提示することが難しい。

通常の 3D テレビは、上記の問題の多くを解決しており、広視野な 3D 映像を容易に鑑賞することが可能である。また、3次元空間への没入感をさらに増す装置として、ヘッドマウントディスプレイを用いるものもある。

3D テレビは、HDMI インタフェースを介して立体映像信号を入力する。HDMI で 3D 映像を伝送する方式として、フレームパッキング、サイドバイサイド、トップアンドボトムがある。フレームパッキングはブルーレイ 3D で用いられている方式であり、他の方式に比べて高画質である。サイドバイサイドとトップアンドボトムは、通常の映像信号に左目画像と右目画像を重畳させる方法である。サイドバイサイドでは、画面の左半分に左目画像を、右半分に右目画像を表示する。従って、一枚あたりの水平解像度は半分になる。トップアンドボトムでは、上半分と下半分に分割する。この資料では、サイドバイサイドによる 3D 画像として 3D ディスプレイに表示する方法を説明する。

3. 左目画像と右目画像をサイドバイサイドの 3D 画像に変換する方法

左目画像・右目画像は異なる視点から撮影した2枚の写真である。図 1 上にその例を示す。この2枚の画像から、図 1 下に示すように、2枚の画像を横に並べた 1920×1080 画素の1枚の画像を作る。この時、左目画像・右目画像の解像度を半分にする。また、画素数を 1920×1080 に変換する場合に、水平解像度を半分にした画像の縦横比が変化しないように注意する。

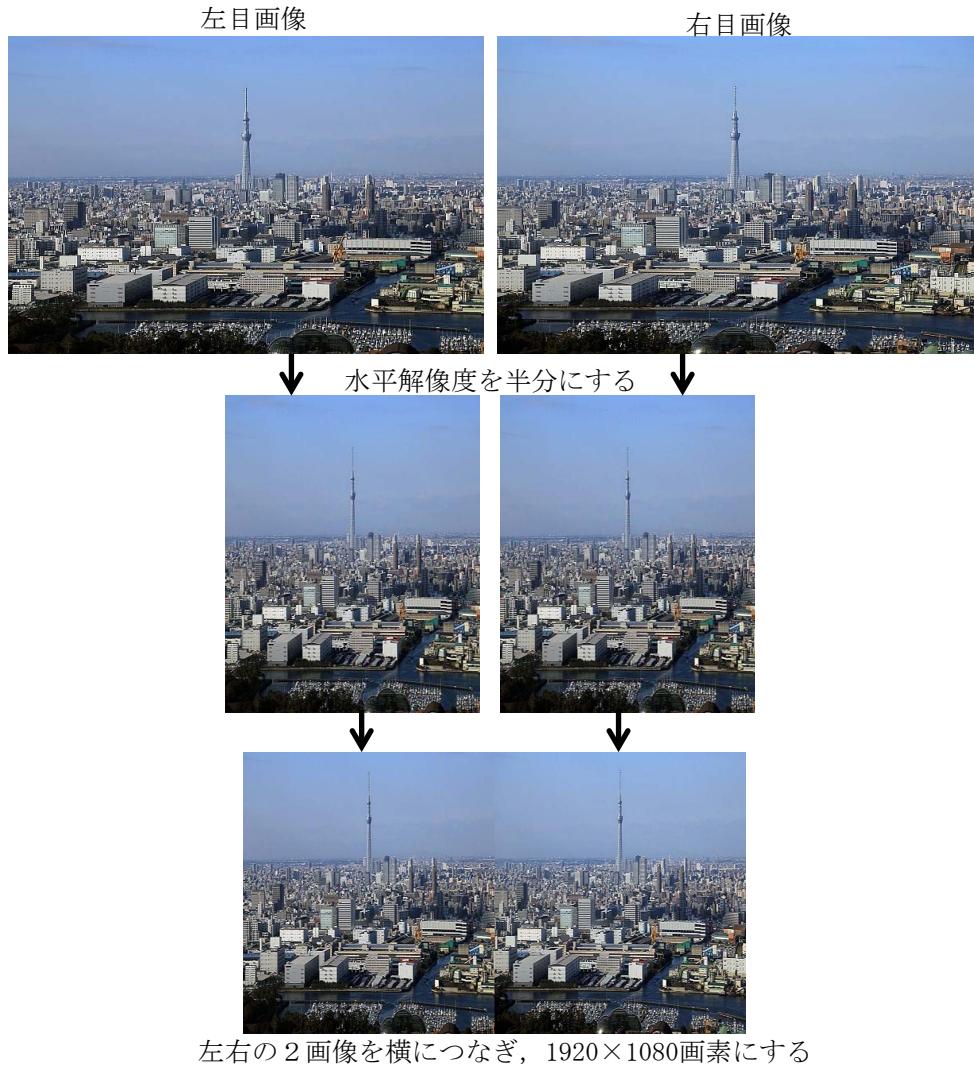


図 1 立体写真 (http://tokyo-sky-tree.seesaa.net/にある写真を利用)

1920×1080 画素のサイドバイサイド画像から H.264 でエンコードされた mp4 ファイルを作成する。Premiere Pro を使うと、次のように作成することができる。

- ① Premiere Pro を起動し、デフォルト設定で新規プロジェクトを作る。
 - ② 新規シーケンス（シーケンス 01）を、使用可能なプリセット Digital SLR->1080p->DSLR 1080p30@29.97 として OK する。
 - ③ Premiere の操作画面で、「ファイル->読み込み」によって、1920×1080 画素のサイドバイサイド画像を読み込む。読み込んだ画像を、図 2 のようにシーケンス 01 のトラック Video 1 のドラッグする。一枚の静止画像を映像中で再生するデフォルトのフレーム数を、「環境設定->一般」で設定できる。
- 今回の場合、1 フレームの再生でもよい。

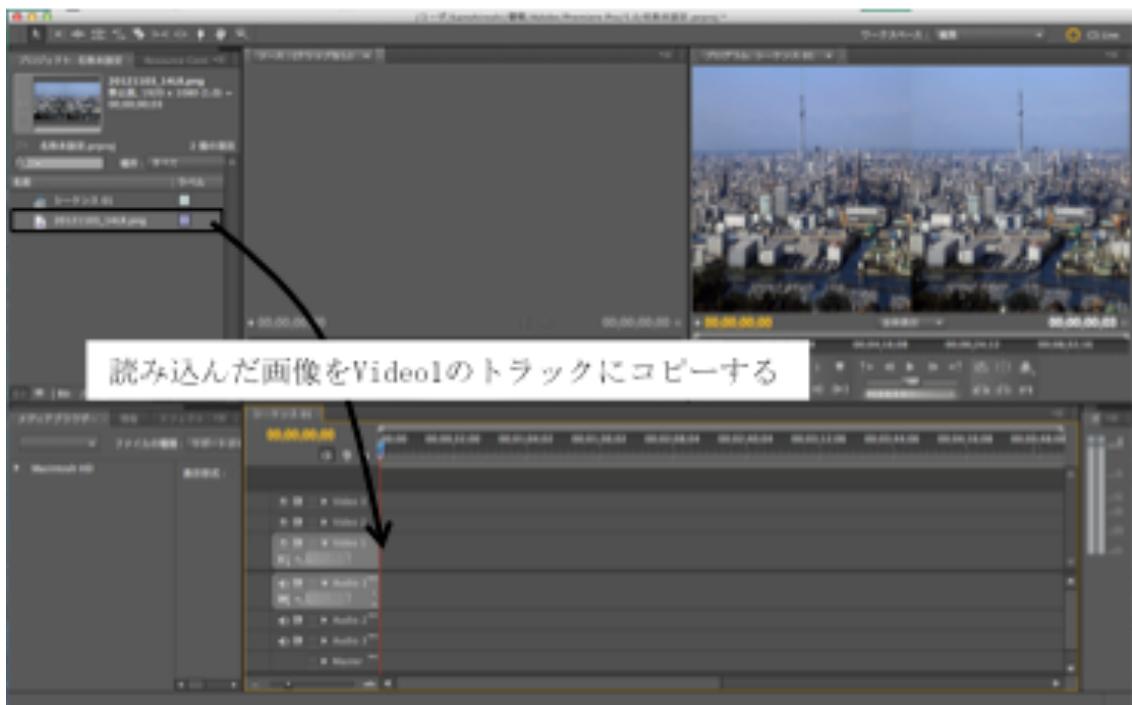


図 2 読み込んだ画像を映像のトラックにドラッグする

- ④ 一枚の画像だけの映像をファイルに出力ため, シーケンス 01 を選択した状態で, 「ファイル->書き出し->メディア」を実行する。書き出し設定のウィンドウが表示される。形式を H.264, プリセットを HDTV 1080p 29.97 高品質に設定し, 書き出しを行う。mp4 ファイルが生成されるので, その内容を確認する。

4. サイドバイサイド 3D 映像を Raspberry Pi で HMZ-T2 に表示する方法

サイドバイサイドの 3D 映像を, HDMI 端子を持つパソコンで再生し, HDMI 端子を 3D ディスプレイに接続することで, 3D 表示することが可能になる。この資料では, Raspberry Pi とソニーのヘッドマウントディスプレイ HMZ-T2 を用いる場合について説明する。

図 3 に表示システムの構成を示す。Raspberry Pi のディスプレイ表示に用いる HDMI 信号を分配機で 2 系統に分け, 一方を通常のディスプレイに, 他方を HMZ-T2 に表示する。なお, Rasoberry Pi の HDMI 出力は, デフォルトでは 1280×720 に設定されている。これを 1920×1080 にするため, /boot/config.txt の設定を次のようにする。

```
framebuffer_width=1920
framebuffer_height=1080
hdmi_force_hotplug=1
hdmi_mode=5
```

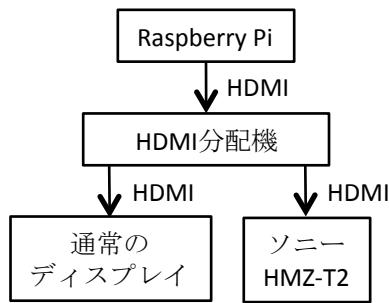


図 3 表示システム

作成した mp4 ファイルを VLC メディアプレイヤーで再生する。ディスプレイ画面にサイドバイサイドの画像が表示される。表示を全画面、最小化インターフェースに設定する。HMZ-T2 の 3D 方式として左右分割方式を選択する。以上で、HMZ-T2 で両眼立体視が可能になる。