

レイヤー構成されたデジタルイラストの立体視画像化

春日秀雄¹⁾ (正会員)

布施信己²⁾

1) 神奈川工科大学情報学部 2) 神奈川工科大学大学院工学研究科

Stereoscopic image conversion for multi-layer digital illustrations

Hideo Kasuga¹⁾(Member) Nobuki Fuse²⁾

1) Faculty of Information Technology, Kanagawa Institute of Technology

2) Graduate School of Engineering, Kanagawa Institute of Technology

kasuga @ ic.kanagawa-it.ac.jp

アブストラクト

本論文では、デジタルイラストがレイヤーごとにオブジェクトを分けて描画されることに着目し、レイヤー構成されたデジタルイラストを立体視画像へ変換する手法を提案する。立体視画像を作成する手法としては、3DCGを用いる方法やDepth Mapを用いる方法があるが、3Dモデルを作成したり距離情報を持った画像の用意をしたりという手間が必要である。本論文では、一般的な2Dのイラストを描く技術だけを使って容易に立体視画像を作れることを目的とする。提案手法では、レイヤーを地面要素、壁要素、物体要素、その他要素という4種類の要素に分類し、それぞれの要素の位置関係に基づく幾何学変換を施して立体視画像を生成する。本手法を実装したアプリケーションを作成し、立体視画像生成実験を行い、デジタルイラストが違和感のない立体視画像に変換されることを確認した。

Abstract

A method for converting digital illustrations to stereoscopic images is proposed. We noticed that digital illustrations are drawn using multiple layers. The popular method to make stereoscopic images is a technique using 3DCG. In this case, creation of a 3D model of a scene is required. Technique using a depth map is another popular method to make stereoscopic images. The goal in this paper is to easily create stereoscopic images using only the techniques of drawing general 2D illustrations. In the proposed method, the layers are classified into four type of elements: "floor", "wall", "object", and "other". Stereoscopic images are created by transforming these elements geometrically.

1. はじめに

本研究は、レイヤー構成されたデジタルイラストを立体視画像へ変換するアプリケーションの開発に関する研究である。

昨今、立体視技術を用いた映画やテレビゲーム等が普及し、立体視技術を用いたデジタルコンテンツが身近なものとなっている。立体視を楽しむテレビやデバイスも一般家庭で容易に手に入れられる環境にある。また、立体視に対応したデジタルサイネージも登場し、街中や建物の中でも立体視画像を見る機会は増えている。例えば、グランフロント大阪に展示されていたNICTのREI[1]は200インチという大画面で裸眼での立体視が可能である。REIはプロジェクタアレイと1つのディスプレイスクリーンから構成されるシステムであるが、多数の3Dディスプレイを並べることで大画面の立体表示を実現する立体ディスプレイとしてカリフォルニア通信情報機構 (Calit2) のディスプレイ[2]や東京農工大の高木の等身大裸眼立体ディスプレイ[3]なども存在する。

立体ディスプレイはエンターテインメント、シミュレーション、医療等で利用されてきたが、今後はデジタルサイネージ他、広い分野での利用が考えられる。そして、立体表示コンテンツを活用した映像ビジネスに求められる要素の中に低コスト化が挙げられる。これにはデバイスの低価格化だけでなく、コンテンツ制作におけるコストの削減も必要である。そこで我々は、安易に立体視画像を作成するために、レイヤー構成されたデジタルイラストを立体視画像に変換するシステムを提案する。本システムが対象とするデジタルイラストは写実的なものに限らない。本システムを用いると、一般的な2Dのイラストを描く技術だけを使って立体視画像を作ることができるため、デザイナーの労力を削減することができる。

2. 関連研究

2Dの画像から立体視画像を作成する処理を2D-3D変換と呼ぶ。その代表的な変換手法のひとつが3DCGを用いる方法である。これは、2D画像内の要素や形状に合わせた3Dモデルを作成し、元となる2D画像を3Dモデルにプロジェクションし、複数視点からレンダリングすることで作成される。3Dモデルを精密に作成することで細かな立体感を表現することも可能となるが、3Dモデルを作成するには高価な3DCGソフトウェアを使いこなす技術が必要となり、コンテンツ制作のコストが比較的高く、制作にかかる労力も多い。

Depth Mapを用いた2D-3D変換も良く使われる。Depth Mapとは画像内の距離情報をグレースケールで表現したものである。画像内の最も遠くに位置するものを黒で描き、最も近いものを白で（もしくはその逆で）描いたものである。Depth Mapを用いた変換としては、3Dテレビ用の立体映像を作るためのDepth image-based rendering技術[4]が挙げられる。文献[5]のような自動的にDepth Mapを作り出す手法は提案されているが、本論文で対象とするようなデジタルイラストの場合、画像自体に距離情報が含まれていないため、Depth Mapを自作する必要がある。

く。複数の位置から撮影した画像があればMulti-View Stereoアルゴリズム[6]によってDepth Mapを作成する事は可能だが、単一の画像からの距離情報を求めることは困難である。

単一の2Dの画像から擬似的に3次元構造を構築する手法としてはHonyらの手法[7]が存在する。この手法では、前景と背景を分離して背景に対して消失点と地面-壁の境界を与えることで、一点透視図法で表すことができる3Dモデルを作成している。ただし、画面内に消失点がある一点透視図法の構図に限られ、消失点や境界を手動で決定する必要がある。

単一の2Dの画像から自動的に3Dモデルを作成する手法としてはHoimらの手法[8]が存在する。全体的にはもっとも美しいモデルが生成されるが、壁の手前にある物体が壁と一体化するといった、局所的には間違ったモデルが生成されるため、立体視画像の作成に利用した場合は違和感のある画像となる場合が多い。また、写真ではなくイラスト（写実的ではないもの）を対象とした場合、意図した構図の3Dモデルとならないことが多いといった問題もある。

単一の画像からステレオ画像を作成するソフトウェアとしては、サンデーフォトスタジオ[9]というソフトウェアが存在する。手動で切り出した領域、または顔検出を使って自動的に切り出した人物領域を、その他の領域より手前に見えるように配置したステレオ画像を作成することができる。得られるステレオ画像は平面の前景と平面の背景という構成となる。例えば、地面の上に立つ人物を前景に選ぶと、地面から人物が浮き上がって見えることになる。奥行きは前景と背景の2段階で、地面や壁のような手前から奥に向かう立体感も表現できないため、立体的な空間を十分に表現できているとは言いにくい。

3. 立体視画像生成アプリケーション

本論文では、単一の画像から立体視画像を作成することを目的とし、実写の画像ではなくデジタルイラストを主な対象としている。我々はデジタルイラストがAdobe Photoshopのようなソフトウェアを使ってレイヤーごとにオブジェクトを分けて描画されることが多い点に着目し、レイヤー構成された画像をレイヤーごとに分類して相互の位置関係から変形や移動といった幾何学変換を施して立体視画像を生成するアルゴリズムを考案し、それを実装したアプリケーションを開発した。本アプリケーションは以下の3つの機能から成り立っている。

- レイヤー別画像生成
- 要素分類
- 画像変換

レイヤー別画像生成は、Adobe Photoshopで作られたPSDファイルから各レイヤーを透過PNG形式の画像で出力する処理である。これはJavaScriptのスクリプトを用いて実装している。要素分類はC#で作成した図1のようなフォームである。このフォームでレイヤーを4種類の要素に分類するが、要素の詳細については4章で述べる。画像変換は、各要素の幾何学変換と合成を行う処理である。これはC++とOpenCVによって作成されている。この変換アルゴリズムについても4章で述べる。

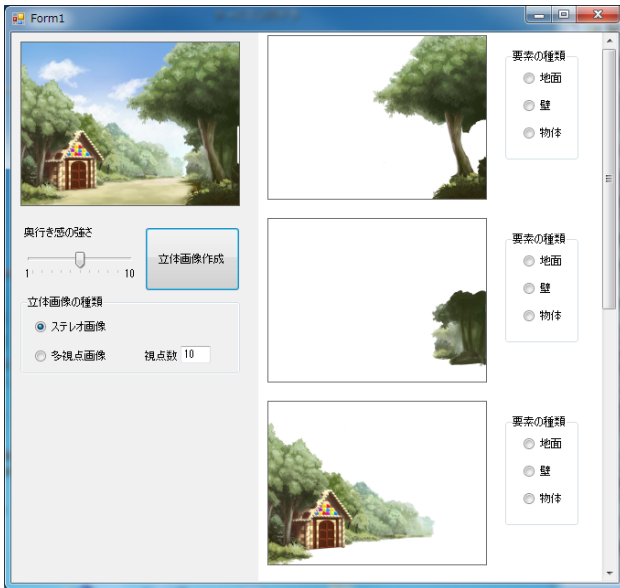


図1. 要素分類フォーム.

本システムでは、まずPSDファイルからレイヤーごとに透過PNG形式の画像が生成される。それが各レイヤーを分類するフォームに渡される。現状では、PSD形式以外のレイヤー構成の画像の場合は、手動で各レイヤーを透過PNGに変換しておく必要がある。ユーザーは、要素分類フォームで各レイヤーの要素の種類を図2のようなラジオボタンで決定する。地面要素、壁要素、物体要素のどれにもチェックが付いていないレイヤーはその他要素として扱われる。



図2. 要素の種類を選択.

奥行き感の強さは図3のようなフォームのスライダーで調節可能である。3Dコンソーシアムの『3DC安全ガイドライン』[10]では立体視画像を快適に視聴できる視差の基準が示されている。それを考慮して、最大の10とした場合は快適視差範囲のほぼ上限の視差となるような立体視画像が生成されるようになっている。

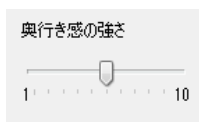


図3. 奥行き感の調節スライダー.

本システムでは、ステレオ画像（2視点の画像）だけでなく多視点画像も作成が可能である（図4）。多視点画像を選択した場合、最も左の視点からの画像と最も右の視点からの画像の間の画像も生成され、合計で指定した枚数の画像が生成される。視点数Nの多視点画像の生成の流れは次のようになる。



図4. 立体画像の種類選択.

- Step1. 通常のスレオ画像の生成プロセスで2枚の画像を生成する
- Step2. 通常のスレオ画像生成の際の最大視差を a 、すでに生成してある画像枚数を n としたとき、 $a*(N-n-1)(N-1)$ を最大視差とした2枚の画像を生成する
- Step3. 上記を視点数（視点数が奇数の場合は視点数-1）の画像が作られるまで繰り返す

要素の種類、奥行き感の強さ、立体画像の種類を決定した後、立体画像作成のボタンを押すと、ステレオ画像の場合は2枚の画像が、多視点画像では視点数の画像が作成される。

4. 変換アルゴリズム

本手法では地面、壁、地面上の物体といった要素ごとに変形や移動を行っている。これは、前景と背景を区別して画像変形を行う Horry らの手法と類似しているが、Horry らの手法では背景をメッシュに分割して変形しているのに対して、本手法ではレイヤーごとに幾何学変換を行っている。各レイヤーには様々な形状や大きさの物体が描かれるため、直線で分割されたメッシュを変形させるような手法と異なり、本手法では描かれた物体の位置関係を求めて変形・移動量を算出する処理が必要になる。なお、本手法では地面や壁が直線で構築された領域とならないため、複雑な形状の境界が自然に表現できるという特長がある。また、一点透視図法で描かれた構図に制限されることもない。本手法は二点透視図法で描かれた構図や消失点の存在しない零点透視図法のイラストにも適用が可能である。なお、本手法では壁や物体は直立しているものとしており、奥や手前に大きく傾いた表現はできない。そのため、三点透視図法で描かれたイラストでは不自然さが残る場合がある。

変換アルゴリズムは、先行研究[11]として発表したアルゴリズムと基本的には同様である。まず、対象となるデジタル画像の各レイヤーは、ユーザーによって表1の4種類の要素に分類される。分類された要素は種類によって異なる幾何学変換が適用される。台形変形処理に関しては4.1で、左右伸縮変形処理に関しては4.2で、水平移動処理に関しては4.3で述べる。また、一点透視図法で描かれたイラストや二点透視図法で描かれたイラストの変形後の消失点の一貫性の保証に関して4.4で述べる。

表1. 要素の種類と適用する幾何学変換.

要素の種類	要素の説明	適用する幾何学変換
地面要素	地面や床となる部分を指す要素	台形変形
壁要素	手前から奥へ向かう建物の壁や塀、柵となる部分を指す要素	左右伸縮変形
物体要素	人物等の地面要素の上に乗っている物体を指す要素	水平移動
その他要素	地面要素でも壁要素でもなく、地面の上に乗っていない(浮いている)物体を指す要素	水平移動

先行研究では『地面要素』、『壁要素』、『物体要素』の3種類に分類していたが、それだけでは効果的に表現できないイラストがあることを考慮して、本論文では『その他要素』を含めた4種類に分類している。イラストでは最前面のレイヤーに文字を書くことも多く、それらがその他要素となる。また、最背面に描く空のようなレイヤーもその他要素である。その他要素には物体要素と同じ水平移動処理が適用される。

4.1 台形変形処理

台形変形は、地面要素に施す幾何学変換である。図5のように、左眼用画像では右上部を左方向へ収縮させ、右眼用画像では左上部を右方向へ縮小させるように変形させる。

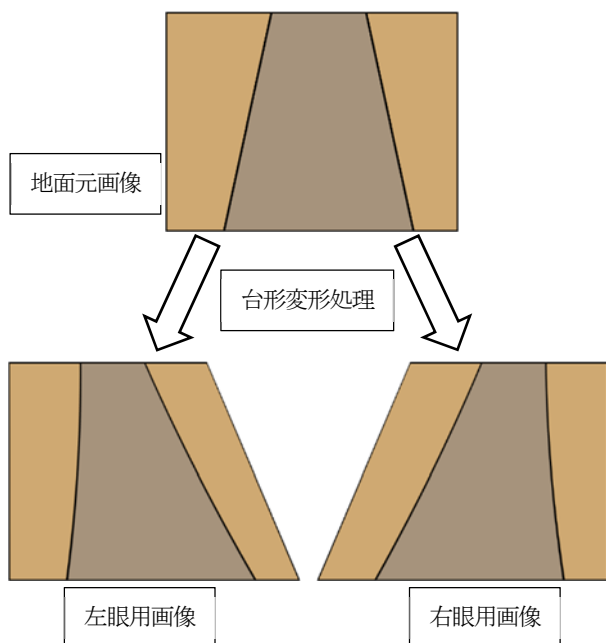


図5. 台形変形処理.

この視差が大きいほど地面要素の上部はスクリーン面より奥(遠く)に知覚される。そのため、地面要素上部の縮小量が大いほど地面の上の壁要素や物体要素の遠近感も強くなり、立

体視画像の奥行き感は強くなる。奥行き感の強さは、3章で説明したようにフォームで調節することができる。このフォームで入力された値から地面要素の変形量は決定される。『3DC安全ガイドライン』では、立体視画像の快適視差範囲は視差1.0度以内としており、標準観視距離での画面幅に対する比は2.9%となる。そのことから、奥行き感の強さが最大の10と設定した場合の変形量(図6に示す画像上部の角部分の移動量)は画像幅の3%としている。最小の1と設定した場合の変形量は画像幅の0.3%で、値が増えるごとに0.3%ずつ増加する。

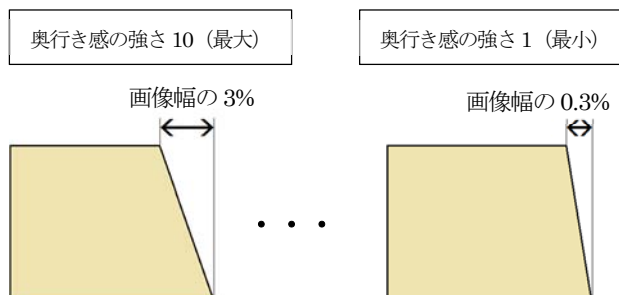


図6. 奥行き感の強さと変形量.

一般的な台形変形では射影変換が用いられるが、射影変換による台形変形では図7上のように画素がY方向にも動いてしまう。そうすると他の要素との上下の位置関係がズレ、生成された立体画像に歪みを感じられるようになる。そのため、本手法では図7下のようにY方向へのズレが生じないように変形を行っている。

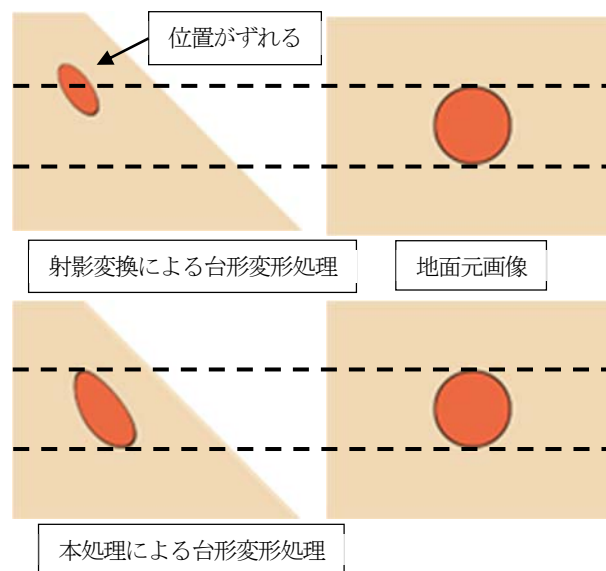


図7. 台形変形処理の比較.

地面要素を台形変形処理で変形させた後に、その地面の上に乗っている他の要素の変形量や移動量が求められるため、台形変形処理は左右伸縮変形処理や水平移動処理に先立って行われる。

4.2 左右伸縮変形処理

左右伸縮変形処理は、壁要素に施す幾何学変換である。変形に先立ち、変形量の算出に用いる基準点の検出が行われる。基準点は2点必要であり、以下の説明では便宜上、画面の下側に位置する点を始点、始点の逆の端にある点を終点と称する。始点は地面要素と接する一番下の点である。左から右に行くにつれて画面奥側に延びてゆく図8のような壁では、終点は地面要素と接する右端下部の点となる。右から左に行くにつれて画面奥側に延びてゆく図9のような壁では、終点は地面要素と接する左端下部の点となる。

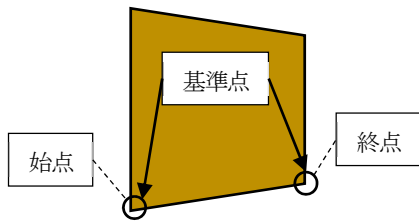


図8. 左から奥に延びる壁要素の基準点.

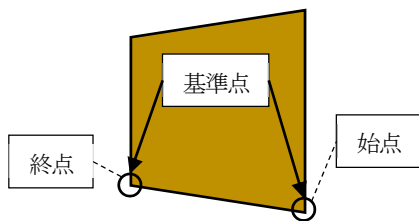


図9. 右から奥に延びる壁要素の基準点.

左右伸縮変形処理の変形量は、地面要素の変形量から求まる。地面要素の変形後の位置に壁要素の基準点も同じように移動するように壁要素は変形される。図10のような壁の場合、左眼用画像では縮小され、右眼用画像では伸張されることになる。図10と向きの異なる壁の場合は逆に、左眼用画像では伸張され、右眼用画像では縮小されることになる。

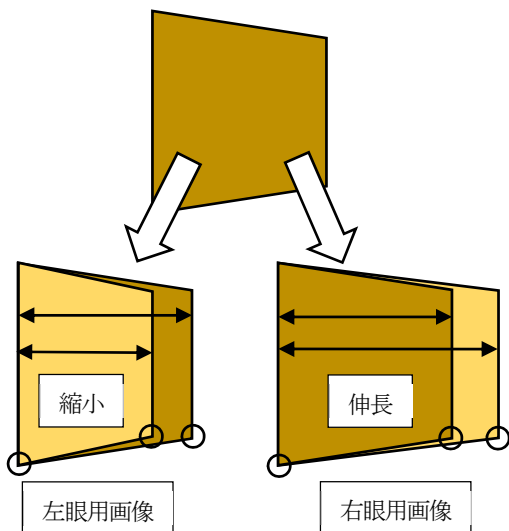


図10. 左右伸縮変形処理.

4.3 水平移動処理

水平移動処理は、地面要素と壁要素以外に施す幾何学変換である。物体要素では地面要素の変形量に応じて移動量が決まる。その他要素ではレイヤーの前後関係から移動量が決定し、地面要素との位置関係は考慮されない。物体要素では壁要素と同じく、変形に先立ち移動量の算出に用いる基準点の検出が行われる。水平移動処理では基準点は1つでよく、地面要素と接する一番下の点が基準点となる。

物体要素の水平移動処理の移動量も、地面要素の変形量から求まる。地面要素の変形後の位置に物体要素の基準点が移動するように水平移動処理は行われる。図11のように物体要素は、左眼用画像では左に、右眼用画像では右に移動される。

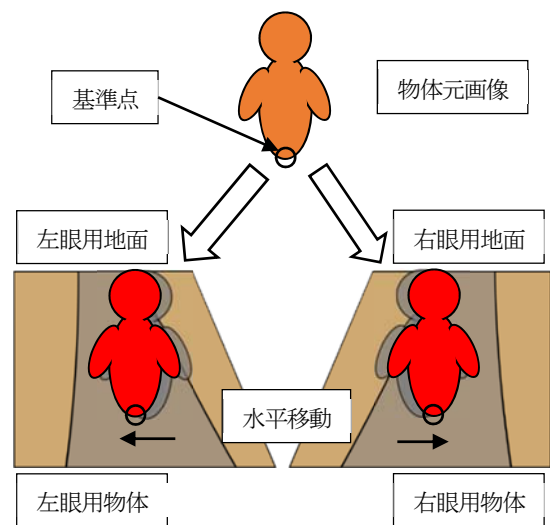


図11. 水平移動処理.

その他要素は、他のすべての要素の幾何学変換が終わってから処理される。その他要素の水平移動処理の移動量の計算は、壁要素や物体要素のように地面要素の変形量を元に計算せず、他のレイヤーとの前後関係から決定される。最前面レイヤーの場合は、スクリーン面の手前に飛び出すように設定される。このとき、その他要素は左眼用画像で右に、右眼用画像で左に移動される。移動量は、強すぎる奥行き感を与えないように、地面要素の変形量の半分となる。最背面レイヤーは、地面要素の奥行きよりさらに奥に配置するため、地面要素の変形量の1.1倍の移動量となる。それ以外のその他要素は、前後のレイヤーの物体要素の中間の奥行きに配置する。移動量は、前後の物体要素の移動量の平均となる。

4.4 消失点の保証

要素ごとに前述の幾何学変換を施すが、本手法では一点透視図法で描かれたイラストや二点透視図法で描かれたイラストの消失点は変形後も一点、または、二点に収束する。本手法では地面要素は図7下のように変形されるため、地面要素上に描かれた任意の直線は歪んで曲線になることなく、変形後も直線が維持される。そして壁要素は、基準点が地面要素の変形後の同じ位置に移動するように変形される。そのため、すべての壁要素

素の基準点の始点-終点を結ぶ直線は、変形後も同じ点に収束する。

5. 立体視画像生成実験

開発したアプリケーションを使って Adobe Photoshop で作成したデジタルイラストを立体視画像に変換し、違和感のない画像が生成されることを確認する実験を行った。表示装置には NVIDIA 3D Vision システムを使ったアクティブシャッター方式のフル HD 解像度の 3D ディスプレイとグラスレス 3D ディスプレイを搭載した東芝 dynabook (Qosmio T851/D8DR) の 2 種類を用いた。被験者は立体視画像を認識できる能力と十分な視力を持った 20 代の大学生 7 人である。被験者は事前に市販の 3D 映画を実験環境で視聴して立体感を認識する能力に問題がないことを確認した。実験として、被験者には本システムの仕組み等一切説明せずに画像を見せ、立体感の有無、違和感・不自然さの有無、その他コメントを回答してもらった。なお、2 種類の表示装置を用意して実験を行ったが、すべての被験者の回答で表示装置の違いによる立体感の知覚の違いはなかった。

図 12 は 6 つのレイヤーから構成された画像である。サイズは 1024×746 ピクセルである。手前のレイヤーから、物体要素 (右手前の木)、物体要素 (右奥の木)、壁要素 (左の林と家)、地面要素 (道)、その他要素 (空)、その他要素 (空の色調整用の単色レイヤー) となっている。変換結果は図 13 となる。視差の分かりやすい場所を拡大した画像も合わせて示している。奥行き感の強さは 10 段階中の 5 と設定した。地面要素、壁要素は手前から奥へ延びるように知覚される構図となっている。実験の結果、すべての被験者から立体感が感じられるという回答が得られ、違和感や不自然さがあるという回答もなかった。そのことから、地面の上に乗っている壁要素や物体要素の位置は適切に計算されており、違和感のない立体画像になっていることが確認された。ただし、違和感というほどではないが左の林と家の細かな立体感が分かりにくいというコメントがあった。本手法では原理上、壁要素と地面要素が平面で表現されるので、壁や地面の凹凸までは再現できない。これは、Hoiem らの手法のような画像を領域分割し 3D モデルに当てはめる手法でも同様に発生する問題である。ただし、他の被験者では不自然さはないという回答であったため、陰影の書き込まれたイラストでは陰影の効果によって立体感を感じるため違和感が生じることは少ないと考えられる。

図 14 は 7 つのレイヤーから構成された画像である。本画像は大画面での表示を想定して比較的高い解像度で作成しており、サイズは 2745×2231 ピクセルとなっている。手前のレイヤーから物体要素 (左の石垣)、壁要素 (左奥の林)、物体要素 (街灯)、壁要素 (右の林)、地面要素 (道)、その他要素 (空)、その他要素 (空の色調整用の単色レイヤー) となっている。変換結果は図 15 となっている。奥行き感の強さは最大の 10 と設定した。こちらも地面要素、壁要素が手前から奥に延びるような構図である。実験の結果、すべての被験者から立体感が感じら

れるという回答が得られ、違和感や不自然さがあるという回答もなかった。こちらの画像でも、地面の上に乗る要素の位置が適切に計算されており、違和感のない立体画像になっていることが確認された。

図 16 は特に立体感 (空間構造) を意識して描いていない写実的ではないイラストの例である。レイヤーは 6 つで壁要素はない。サイズは 900×600 ピクセルである。最前面のレイヤーがその他要素となるテキストとなっている。続いて物体要素 3 つ、地面要素、その他要素となっている。奥行き感の強さは 5 と設定している。変換結果は図 17 となる。実験の結果、すべての被験者から立体感が感じられるという回答が得られ、違和感や不自然さがあるという回答もなかった。写実的な画像ではないが、最前面の文字の飛び出しの立体視効果と、猿、羊、馬が順番に奥に向かって並ぶ立体感が感じられることが確認できた。また、シンプルなデザインで奥行き感を強く感じる地面要素ではなかったが、地面上の物体要素の配置から違和感が生じないことも確認できた。



オリジナル画像



図12. 実験画像1.



オリジナル画像



図14. 実験画像2.



図13. 変換結果1 (左: 左眼用画像, 右: 右眼用画像) .

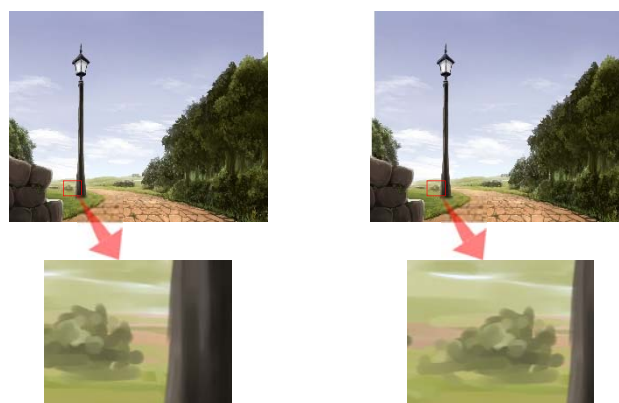


図15. 変換結果2 (左: 左眼用画像, 右: 右眼用画像) .



オリジナル画像

各レイヤー	要素の種類
	その他要素
	物体要素
	物体要素
	物体要素
	地面要素
	その他要素

図16. 実験画像3.



図17. 変換結果3 (左: 左眼用画像, 右: 右眼用画像) .

6. まとめと今後の課題

本論文では、デジタルイラストがレイヤーごとにオブジェクトを分けて描画されることに着目し、レイヤー構成されたデジタルイラストを立体視画像へ変換するシステムを提案している。本システムを使って立体視画像を作成する場合、必要となるのは2Dイラストを作成する技術のみである。従来の立体視画像の作成方法である3DCGやDepth Mapを用いる方法と比べて容易に立体視画像を生成することができる。Horryらの手法やHoiemらの手法のような比較的簡単な操作で、または、自動

で3次元情報を構築する手法も存在するが、構築される3Dモデルは直線的な領域で構成されたモデルとなり、例えば草木のような複雑な境界を違和感なく再現することは難しい。それらと比較すると、本手法はすべてのオブジェクトをレイヤーごとに描く必要はあるものの、境界が曲線となるオブジェクトやより複雑な境界を持つ草木のようなオブジェクトで自然な立体画像を生成することができると考えられる。また、本手法の変換自体はシンプルな幾何学変換なので、複雑な計算が必要な2D-3D変換と比べてメモリ不足などの問題を起こしにくく、高解像度の画像に対応しやすいという特長もある。32ビットOSでアプリケーションのメモリが2GBまでしか確保できない環境でも4500×3000ピクセル6レイヤーの画像の処理が可能で、メモリを8GB搭載した64ビットOS環境では9000×6000ピクセル6レイヤーの画像の処理が可能であった。

本手法では描画されたオブジェクトを移動・変形して視差を作り出すため、もともとは前面の物体に遮蔽されて見えない背面のオブジェクトまである程度描き込んでいないと、その場所が空白になってしまう場合がある。Adobe Photoshopを使っている場合は修正ブラシツールを使って欠けている部分の補完が容易にできるが、画像補完のアルゴリズムをシステムに組み込むことで、物体背面への書き込みが不十分なイラストでも画像の欠けがなくなり、より利便性は上がると考えられる。

本論文が対象としているのはレイヤー構成されたデジタルイラストであるが、レイヤー構成されていない画像を自動的にレイヤー分けして変換ができると適用範囲が広がるため、画像を領域分割して自動的にレイヤーに分ける機能を追加することも今後の展開としては有効と考えられる。

参考文献

- [1] S. Iwasawa, M. Kawakita, and N. Inoue, REI: an automultiscopic projection display, Proc. 3D Systems and Applications 2013, Selected Paper Session 1, 2013.
- [2] R. Kooima, A. Prudhomme, J. Schulze, T. DeFanti, A Multi-viewer Tiled Autostereoscopic Virtual Reality Display, Proc. ACM Virtual Reality Software and Technology 171-174, 2010.
- [3] 等身大裸眼立体ディスプレイ, <http://shingi.jst.go.jp/abst/p/14/1408/tuat06.pdf>
- [4] C. Fehn, Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV, Proc. SPIE 5291, Stereoscopic Displays Virtual Reality Systems XI, pg. 93. 2004.
- [5] Wa James Tam, L. Zhang, 3D-TV Content Generation: 2D-to-3D Conversion, 2006 IEEE International Conference on Multimedia and Expo, pg.1869, 2006.
- [6] S. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski, A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms, CVPR 2006, vol. 1, pp. 519-526, 2006.
- [7] Y. Horry, K. Anjyo, K. Arai, Tour Into the Picture: Using a Spidery Mesh Interface to Make Animation from a Single Image, SIGGRAPH '97, pp. 225-232, 1997.

[8] D. Hoiem, A.A. Efros, M. Hebert, "Automatic photo pop-up", SIGGRAPH2005, pp. 577-584, 2005.

[9] <http://www.kumagaya.or.jp/~mcc/spstudio/>

[10] 3DC 安全ガイドライン ,
http://www.3dc.gr.jp/scmt_wg_rep/3dc_guideJ_20111031.pdf

[11] 布施, 春日, デジタルイラストの立体視画像化変換アルゴリズム, NICOGRAPH 2015, 2015.

春日 秀雄



1995 年信州大学工学部情報工学科卒業。1997 年信州大学大学院工学系研究科情報工学専攻博士前期課程修了。2000 年信州大学大学院工学系研究科システム開発工学専攻博士後期課程修了 博士(工学)。2000 年信州大学工学部情報工学科助手。2000 年神奈川工科大学助手, 2007 年同大学准教授, 現在に至る。画像処理, メディア理解, 立体表示に関する研究に従事。電子情報通信学会, 情報処理学会, 芸術科学会, 他会員。

布施 信己



2015 年神奈川工科大学情報学部情報メディア学科卒業。現在神奈川工科大学大学院工学研究科情報工学専攻博士前期課程在学中。