

## ゲイズアウェアネスを考慮した Visual Hullのレンダリング手法

宮田 慎也<sup>†1</sup> 坂本 竜基<sup>†1</sup>

本論文では、Visual Hull を用いた自由視点映像をゲイズアウェアネス等の顔部分の品質保証が要求されるグループウェアに応用する目的で、顔部分のテクスチャに一貫性を持たせる手法を提案する。これは、顔部分のテクスチャが貼られるボクセルに対してテクスチャを取得するカメラを1つに限定することで実現される。このカメラを選択する基準を選択されたカメラの数に依存する方法と仮想視点に依存する方法の2種類挙げた。また、提案手法のレンダリング結果と通常のレンダリング結果をどちらでレンダリングしたほうが顔の表情を読み取りやすいか被験者実験によって評価した結果、提案手法を用いるほうが優れているという結果となった。

### Gaze Awareness Based Rendering for Visual Hull

SHINYA MIYATA<sup>†1</sup> and RYUUKI SAKAMOTO<sup>†1</sup>

This paper describes a rendering method for 3D free viewpoint videos with Visual Hull. The method attempts keeping quality and coherence of textures on human face mapping on its Visual Hull. For keeping coherence, only one camera image is selected as the texture for whole voxels corresponding with facial portion. Two criteria of selection are proposed that are depended on the number of selected cameras as a texture and virtual camera position. As a result of comparison between rendered image with another criterion and the normal rendering method, the proposed method is finer than the normal method.

<sup>†1</sup> 和歌山大学 システム工学部  
Faculty of Systems Engineering, Wakayama University

### 1. はじめに

同期型のグループウェアとして、会議の様子を遠隔地に映像と音声で伝えるテレビ会議システムは古くから研究されている<sup>1)</sup>。また、非同期型のグループウェアとして、会議やミーティングを映像と音声で保存するアプリケーションも研究されている<sup>2)</sup>。どちらの研究においても映像はテキストでは欠落する非言語情報も含むため、会議の内容を伝達するメディアとして有効であるとされている<sup>3),4)</sup>。しかし、このようなアプリケーションの多くはカメラの映像をそのまま利用するか、全方位カメラの映像をトリミングしているだけなので、そのままの映像しか閲覧することはできない。

一方で、複数台のカメラを用いて、閲覧者が実際にはカメラがない位置から見たかのような映像を生成する技術がコンピュータビジョンの分野で提案されており、自由視点映像と呼ばれている。近年では、自由視点映像は応用研究も盛んに行われ、例えばスポーツを自由視点映像化する試みがなされている<sup>5)</sup>。この技術を用いることで、会議やミーティングの様子を自由視点映像化することができれば、例えば、カメラ位置をある会議参加者の視点に合わせ、参加者の目線から会議の様子を閲覧したり、自動的にカメラワーク<sup>6)</sup>をつけてより理解しやすい映像を生成するといったことも可能になるであろう。

しかし、自由視点映像の生成によく用いられる Visual Hull<sup>7)</sup> を用いたアプローチでは、その特性から映像に大きな劣化が起こり、観賞に耐えられない映像が生成される恐れがある。例えば、人間は画像に写ったものの中で顔に注目する傾向があり<sup>8)</sup>、ゲイズアウェアネス<sup>9)</sup>を筆頭に目線は非常に重要な情報であるとされているが、自由視点映像では、カメラの配置や仮想視点の位置によっては顔部分に亀裂が入った映像が生成されることもある。

本研究では、主に会議やミーティングの Visual Hull を生成する際に、ゲイズアウェアネスを補償する目的で顔の部分の品質低下をなるべく抑えるためのレンダリング手法を述べる。Visual Hull の品質向上には様々な手法が提案されているが、どれも比較的重い処理であるうえ、カメラ台数も多いスタジオのような環境での撮影を必要としている。これに対して提案手法は、従来手法に比べて処理時間が短く、かつ顔部分の補償に特化できる点が特徴であり、さらに、少ない台数のカメラで撮影した精度の低い Visual Hull にも対応しているため、将来的にテレビ会議システムへの応用も期待できる。

### 2. Visual Hull の問題点

自由視点映像の生成には様々なアプローチが存在するが、その中でも Visual Hull と呼ば

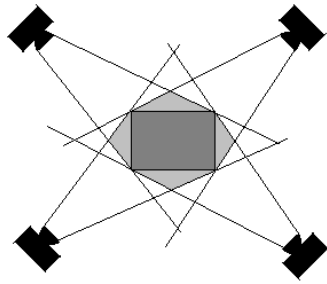


図 1 Visual Hull による形状復元  
Fig. 1 3D model reconstruction with Visual Hull

れる映像から被写体の 3 次元形状モデルを推定するアプローチは、視点位置の自由度が高い映像を頑健に生成できることから多くの研究で用いられている。一度、被写体の形状モデルが推定されれば、OpenGL などの 3 次元コンピュータグラフィックス用ライブラリを用いてそれを 3 次元仮想空間内に復元することができる。これにカメラからの画像を適切にテクスチャマッピングすると、ユーザは自由な位置から被写体を閲覧することができるため、これを映像に対して連続的に処理すると自由視点映像となる。

Visual Hull はカメラ画像における被写体のシルエットから 3 次元空間における被写体の 3 次元形状が削り出されたものであり、原理的に凸包体の推定しかできないが、大量のカメラを用いれば比較的正しい形状を推定することができる。しかし、カメラを大量に配置することが困難な場合は著しくその精度が下がることがある。例えば、ミーティングを撮影するのであればカメラは 4~8 台程度の配置が現実的であろうと考えられるが、図 1 に示す通り、4 台のカメラで直方体を復元する場合、本来の形状からはかけ離れた非常に荒い多角形として近似されてしまう。また、そもそも凸包体ではない部分は正しい形状を復元することができないため、髪と額の間や鼻の淵など人間の顔などの凹部分は最良の状態でも凸包体で近似されてしまう。

Visual Hull の復元は、実装上、撮影空間をボクセル空間として離散化した後におこなわれ、大抵は復元されたボクセルモデルをマーチンキューブ法<sup>10)</sup>などでメッシュ化し、さらにそのメッシュをスムージングするなどの処理を経て最終的にポリゴンモデルとして生成される。しかし、このようなメッシュ化されたモデルは、元々のボクセルモデルの精度が低い場合にはテクスチャに不連続な部分が生じやすい。そこで、ボクセル部分に仮想視点に正対

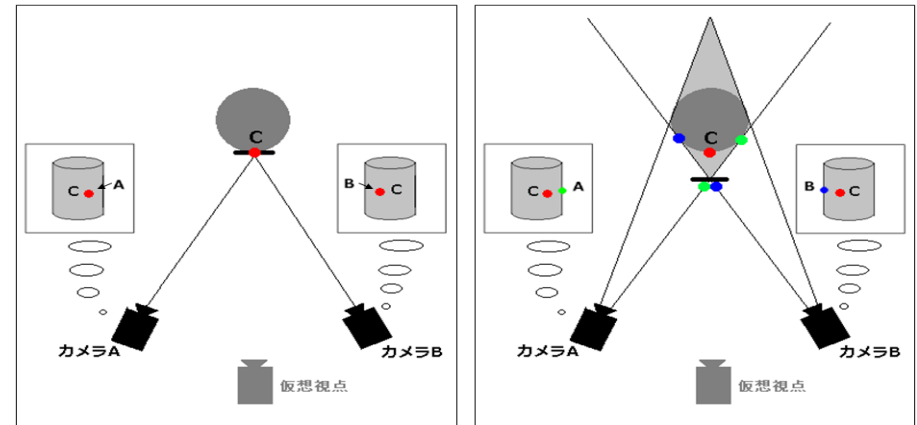


図 2 形状に誤差が引き起こすテクスチャの誤差  
Fig. 2 Texture mapping errors caused by incorrect 3D shape

する微小面 (以後、ファセットと呼ぶ) を立てて、そこにテクスチャマップを施すことで一貫性をもったレンダリングをおこなうマイクロファセット・ビルボーディング法<sup>11)</sup>と呼ばれる手法も提案されている。この手法を用いると、非スタジオ環境下において作成された比較的精度の低いモデルでも、メッシュ化する方法よりも写実的なレンダリング結果を得られるとされている<sup>12)</sup>。

しかし、マイクロファセットビルボーディングを用いても、形状モデルの誤差はテクスチャマッピング時に大きな問題をもたらすことがある。例えば、図 2 のように仮想視点カメラ A とカメラ B の中間部分にあったとしよう。形状が完全に正しい場合には C の部分は、カメラ A を用いてもカメラ B を用いても同じ情報が張り付けられるため、どちらの画像をテクスチャとして用いても問題ない。しかし、形状が凸包体で近似された場合は、C の部分はカメラ A では円柱の右端、カメラ B では円柱の左端になってしまい、どちらの画像を用いても本来は中央部分のテクスチャが貼られるべき部分に、端のテクスチャが貼られてしまう。

この場合、そのファセットにカメラ A の画像をテクスチャとして用いるのか、カメラ B の画像をテクスチャとして用いるのかという選択基準は、一般的にファセットから仮想視点からへのベクトルと微小面からカメラへのベクトルが成す角度が最小のカメラを選ぶ方法

が用いられる．よって，カメラの間に仮想視点がある場合は，Cの部分の左側は，Aのカメラの画像が，右側ではカメラBの画像が使われるため，大きな不整合がおこる．実際，図のような極端な例ではなくとも，テクスチャが別のカメラへと切り替わる境界部分では不整合が生じてしまうことが多い．

以後，本章で説明したようなマイクロファセットビルポーディング法に仮想視点に依存したテクスチャ選択を用いたテクスチャマッピングをおこなうレンダリング手法を通常手法と呼ぶ．

### 3. 関連研究

このような Visual Hull における問題点を解決する手法としては，形状モデル自体を正しく矯正するアプローチが考えられる．このアプローチは問題の根本的な解決となりうるため，すでに幾つかの研究がなされている．延原らは，正しい形状モデル下では，どのカメラのテクスチャをマッピングしても結果が同じになるという特性を基に，初期状態の形状モデルを内側に徐々に矯正していく手法を提案している<sup>13)</sup>．また，富山らは，ステレオ視によって各カメラからモデルへの深度マップを作成し，そこから正しいモデルを推定する手法を提案している<sup>14)</sup>．これらのアプローチは，非常に時間がかかる点が問題となるので，モデル自体は矯正せずにテクスチャ座標を適切にずらすことで，あまり時間をかけずにテクスチャ同士の不整合を解消するアプローチも提案されている<sup>15)</sup>．

しかし，これらは非線形最適化問題であるため，実際は元々の形状モデルの精度がある程度高くなければ機能しないと考えられる．つまり，通常の部屋よりも広いスタジオ環境において，多くのカメラを用いて撮影されていることが前提となっている．一方で，本研究が撮影対象とするミーティングは通常の部屋でおこなわれるため，実際は設置可能なカメラ台数には制限があり，精度良く Visual Hull を復元することは元来難しい．

さらに，このような非スタジオ環境では，シルエットを正しくセグメンテーションできるとは限らず，間違ったセグメンテーションが含まれると大幅な誤差がでてしまうという問題がある．特に，シルエットの一部が欠損した場合は，例えば首から上が完全になくなるといった致命的な破綻を引き起こすことになる．そこで，このようなシルエットの精度に不安がある状況下では，ボクセル空間中へ各シルエットへの投影結果を投票し，ある程度の得票があればそのボクセルは存在しているとする実装方法がとられる．これにより，致命的な欠損は防ぐことができるが，本来はシルエットに問題のなかったモデル部分も全体的に太ってしまい，結果としてモデル全体の精度が落ちてしまう．

以上のような複合的な要因により，ミーティングを撮影する環境下においては Visual Hull は正しく復元できないため，既存手法を適用して正しく補正することは難しいと考えられる．また，通常手法に比べて処理時間も非常にかかるため，テレビ会議システムに応用することは勿論，映像をアーカイブするにも非常に時間がかかり，グループウェアに適した方法であるとは言い難い．

### 4. 提案手法

2章で述べたとおり，復元された形状の誤差がテクスチャの不整合を生みだすことが問題の本質であるが，大幅に幾何的な修正は困難である以上，テクスチャの見かけ上の不整合を防ぐ対策を講じる他ない．そこで，ミーティングの映像において特に重要である顔部分の整合性は必ず確保，つまり，顔部分については1つのカメラからの画像のみを使用するようにする．これにより，全体的としては依然として不整合は残るものの，少なくとも顔部分の一貫性は保たれるためゲイズアウェアネスは確保されるはずである．また，最終的なレンダリングはメッシュを用いず，マイクロファセットビルポーディングを用いることによって，この一貫性がより保たれるようにする．

これらを実現するための処理を以下にステップ毎に示す．

- (1) Visual Hull を推定する
- (2) Visual Hull の各ボクセルにおいてマイクロファセットビルポーディングによるモデリングをおこなう
- (3) 各ファセットに対してテクスチャ画像を取得するカメラを選択する
- (4) 各ファセットに対するテクスチャ座標を計算する
- (5) カメラ毎にテクスチャ座標が顔の領域と交差しているファセットの数をカウントする
- (6) 顔領域とテクスチャ座標が交差しているカメラ群の中で，ある基準で選ばれたカメラ以外が選択されていた顔領域に相当するファセットのテクスチャ座標を，選ばれたカメラへのテクスチャ座標になるように再計算する
- (7) テクスチャ付きのファセットをレンダリングする

このうち，(3)の処理で，各ファセットに貼るべきテクスチャを取得するカメラを選択するが，この選択には，テクスチャの一貫性を確保のために仮想視点に依存したカメラの選択方法を用いる．すなわち，図3のように仮想視点，ファセット，カメラが成す角度  $\theta_i$  が最小となるカメラ  $i$  が各ファセットで選択される．

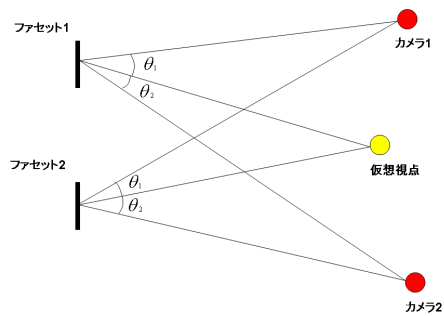


図3 テクスチャ用カメラの選択  
Fig. 3 Camera selections for texture mapping

マイクロファセットビルボーディング法では、ファセットは常に仮想視点に正対するように回転させる。この回転後のファセットの中心から仮想視点を見た場合の右方向のベクトル  $x$  と上方向のベクトル  $y$  は以下のように決まる。

$$x = \frac{w}{2} \mathbf{u} \times \mathbf{z} \quad (1)$$

$$y = \frac{w}{2} \mathbf{z} \times \mathbf{u} \times \mathbf{z} \quad (2)$$

ここで、 $\mathbf{u}$  は仮想視点の上方向の単位ベクトル、 $\mathbf{z}$  はファセットの中心から仮想視点への単位ベクトルであり、 $w$  はファセットの一辺の長さである。この式では、ファセットは仮想視点からの視線方向に依存して仮想視点とファセットとを結ぶ直線を軸とした回転をしてしまうが、ファセットは十分に微小なので見た目の問題にはならない。また、回転角を計算する必要がないので高速に計算できる。

各ファセットの頂点を、ファセットに対して選択されたカメラの射影変換行列によって、カメラ画像に対するテクスチャ座標に変換する。これにより、ファセットの4頂点がカメラ画像に投影され、その4頂点による矩形領域を設定できる。もし、この矩形領域が顔の領域と部分的にでも重複するのであれば、そのカメラに設定されたカウンタをインクリメントする。また、同時にそのファセットに顔領域であることを示すマーキングを付与しておく。この処理をすべてのファセットに対して適用した結果、ある基準に従って選択されたカメラの画像を顔領域に用いる唯一のテクスチャとして設定し、このカメラ以外が選択されていて、

かつマーキングが付与されているファセットのテクスチャと置き換える。この選択をする基準としては、例えば以下のものが考えられる(以後、これらを選択基準 [a],[b] と呼ぶ)。

- [a] カウンタの値が最大のカメラ
- [b] カウンタが1以上のカメラのうち、仮想視点の視線ベクトルとカメラの視線ベクトルの成す角度が最小のカメラ

顔の領域は、顔画像認識等によって各画像に対して顔領域のみを領域抽出した画像を用意してもよいが、例えば肌の色の認識<sup>16)</sup> といった画素毎に独立して判定可能な抽出方法を用いれば、(4)の直後に判定が可能なので高速である。ただし、この場合は、単なる肌の色認識であれば手や腕といった顔以外の肌の領域やノイズも顔領域と認識してしまう問題がある。

## 5. 実 験

### 5.1 GPGPU を用いた実装

提案手法は、ボクセル毎に独立した処理が大半を占めるので並列化計算をおこなうと効率が非常に向上する。そこで、プログラマブルな GPU を利用する並列計算用ライブラリである CUDA を用いて (1)~(4) までの計算を並列的に処理し、その結果を OpenGL でレンダリングした。CUDA では、カメラ毎のカメラ画像、シルエット画像、顔領域画像、射影変換行列を入力として、マイクロファセットの全頂点位置と各ファセットの頂点のテクスチャ座標、カメラ毎のカウンタを出力する。CUDA 内部ではボクセル毎に (1)~(4) の処理を順次におこなうが、このままでは Visual Hull の中身が詰まったモデルになり、(7) のレンダリング時に大変な負荷となる。そこで、中身を剥ぎ抜く処理を (1) と (2) の間でおこなった。このため、全処理を並列におこなうのではなく、(1) と (2) の間で全スレッドに対して同期を取った。

### 5.2 撮影環境と事前準備

個人用のブースに VGA の解像度をもつ IEEE1394 カラーカメラ (Pointgray 社製 Dragonfly2) を図 4 のように 6 台設置した。これらのカメラは、予めキャリブレーションをして射影変換行列を求めておいた。また、被写体のシルエット画像はレンダリング時にも作成可能であるが、その精度は一定ではない。よって、今回は各被写体に対する条件を一定にする目的で撮影後、事前に全カメラの画像に対して GrabCut<sup>17)</sup> によるセグメンテーションをおこない、比較的精度の高い被写体のシルエット画像を作成しておいた (図 5)。



図 4 撮影環境とカメラ位置  
Fig. 4 Environment and camera position.

### 5.3 結 果

#### 5.3.1 実行時間

5.2 節の環境で 6 人の被写体を 1 人ずつ撮影し、その画像からマルチコアの CPU(Intel 社製 Core i7 930) と CUDA に対応した GPU(GeForce 9600GT) を用いてレンダリングをおこなった。撮影対象空間は縦横奥行き方向にそれぞれ 128 分割したものをボクセルとした。(1) の結果である推定された形状のボクセル数は、どの被写体も約 30 万個であった。表 1 は、それぞれの被写体を 6 か所の仮想視点についてレンダリングした際に各ステップでかかった時間の平均を示している。ただし、各ステップ毎に時間を計測するために無駄な処理が混入したり GPU と CPU 間の転送時間などで余分に所要するので表の合計時間は正確ではなく、実際は全体として表の合計より約 20msec ほど時間がかかる。

この表によると、既存手法でも必ず必要となる (1) の形状推定と既存の研究手法ではメッシュ生成に相当する (2) のモデリング部分が多く時間を占めている。ただし、この時間は GPU の性能向上や最適化により大幅に短縮化する可能性があり、例えば、既存研究では Homography による射影変換と補間によって高速にモデリングできることが報告されている<sup>18)</sup>。実質的には、(5) と (6) が提案手法によって通常手法よりも余分に計算時間がかかる部分であるが、大よそ 20msec 強とあまり計算コストがかかっていないことが判る。

#### 5.3.2 通常手法との比較画像

図 6 と図 7 に、同条件下での通常手法と提案手法によるレンダリング結果を示す。これらの図における左側はレンダリングの結果であり、右側は各ファセットにおいて選択された



図 5 取得画像とシルエット画像  
Fig. 5 Source images and silhouette images.

表 1 各ステップにおける実行時間 (msec)  
Table 1 Runtime by each step (msec)

手順		(1)	(2)	(3)	(4)	(5)	(6)	(7)	合計
選択基準	[a]	43.75	46.55	4.68	12.39	15.89	5.44	7.64	136.39
	[b]						5.64		136.54

カメラを示した図である。右側では、同じカメラが選択されたファセットが同じ色で塗られている。

結果、通常手法によってレンダリングした結果である図 6 では、顔の中央部分にテクスチャの切れ目があり、顔が断裂してしまっている。一方、提案手法の結果である図 7 では、体部分にはテクスチャの切れ目があるものの、顔に関しては同一のカメラからのテクスチャのみで構成されているため、少なくとも顔に関しては違和感が少ない。なお、図 7 は、選択基準 [a] を用いた結果であるが、この条件下では選択基準 [b] を用いても同じレンダリング結果となることを確かめている。

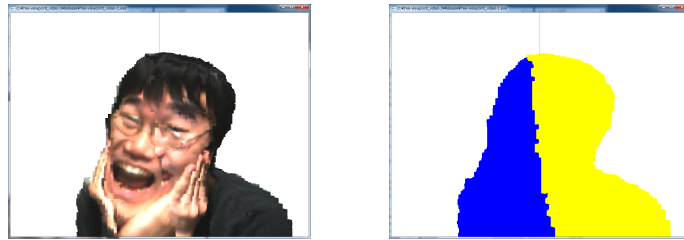


図 6 通常手法によるレンダリング結果  
Fig. 6 Rendered image with normal method.

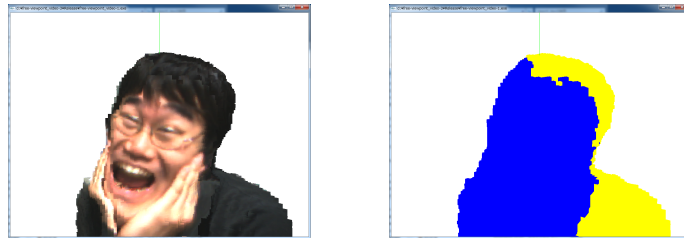


図 7 提案手法によるレンダリング結果  
Fig. 7 Rendered image with proposed method



図 8 通常手法と選択基準 [a][b]  
Fig. 8 Comparison among normal method, selection criteria [a] and [b].

### 5.3.3 カメラの選択基準に対する考察

ここでは、4章で述べた顔部分に対する統一化されたテクスチャマッピングのために唯一のカメラを選択する基準 [a][b] について、図 8 に示すレンダリング結果とともにその特性を検証する。図 8 は、左が通常手法、中央が選択基準 [a]、右が選択基準 [b] を用いたレンダリング結果を示している。

まず、基準 [a] は、本来貼られるべきと主張するファセットが最大になる、換言すればマイノリティなファセットをマジョリティが選択しているカメラで塗りつぶす方針であり、(6)で再計算する座標が少ないため多少高速である。しかし、視線方向とファセットの分布する位置によっては、マジョリティなファセットでカバーしきれない部分まで無理にテクスチャマッピングをすることになり、不自然な結果となることがある。これは、本来の各ファセットのテクスチャを選択する基準は、仮想視点とファセットとカメラが成す角であり、必ずしも仮想視点に対して近いカメラが選択されるわけではないという事実に基づいている。極端な例では、仮想視点とあるカメラが非常に近い位置にいても、顔の位置が視線方向から遠くにあれば、隣のカメラのテクスチャが選択されることもありうる。図 8 は、このような状態を示しており、通常手法では顔の途中で、左側のカメラのテクスチャと右側のそれに分断されている。これを選択基準 [a] ではマジョリティである左側のカメラですべて置き換えてしまっているが、実際は仮想視点は左側のカメラの近くに位置しているため、特に右側のカメラが選択されていたファセット部分で不自然な結果となっている。

これに対して選択基準 [b] は、仮想視点とカメラの関係性のみ依存しているため、このようなことは起こらない。図 8 においても、右側のカメラですべて置き換えられており、選択基準 [a] に比べて自然である。しかし、カウンタの値を比較に利用していないため、非常に少数のマイノリティのカメラによって他のすべてのファセットが塗り替えられる可能性があり、そのような場合は、選択基準 [a] よりも無理なテクスチャマッピングとなることがある。また、マイノリティによって塗り替えられる場合は (6) の再計算に多少の時間がかかる。表 1 では、平均なので殆ど差が現れていないが、条件によっては 3msec ほどの差があることがあった。もしこの時間を短縮したいのであれば、カウンタの値がある一定数以上の場合にのみ選択基準 [b] を用いて、それ以下では選択基準 [a] を用いる方法も考えられる。

## 6. 評価実験

### 6.1 実験概要

提案手法を用いてレンダリングした結果を評価する被験者実験をおこなった。被験者は 21~22 歳のコンピュータサイエンスを専攻している大学生である。画面のサイズが 11 インチのノート PC に異なる手法によってレンダリングされた同一条件下 (同一の被写体かつ同一の仮想視点) での画像を左右に並べて被験者に閲覧させ、左右どちらのほうが表情が判別しやすいか回答してもらった。1 回の評価実験で計 20 組の画像を次々に閲覧してもらったが、画像間で、ある手法が右側や左側に偏らないよう、ランダムに位置の入れ替えをしてカ



図 9 実験 A に用いた画像の一部  
Fig. 9 Images used for experiment A.

ウンターバランスをとった。

### 6.2 実験 A:カメラの選択基準の比較

まず、カメラの選択基準 [a][b] 間で比較をおこなった結果を示す。被験者は 5 人であり、1 人につき 20 組の画像を閲覧してもらったので、計 100 個のデータを収集できた。その結果、49 組のデータにおいて選択基準 [a] が選ばれ、51 組のデータにおいて選択基準 [b] が選ばれた。 $\chi^2$  検定をおこなった結果、有意差は認められなかった。本実験は、図 9 に示すように図上段のように選択基準 A が有利であると考えられるケースと図下段のように選択基準 B が有利であると推測できるケースの両方が閲覧対象として含まれていた。これらの画像を出力する際に選択した仮想視点の位置は、選択基準 [a] と選択基準 [b] でレンダリング結果に差異があり、かつミーティングを撮影するカメラワークとして不自然ではない場所を選んだのであるが、全体的にはどちらのほうが優れているということは言えないようである。さらに、殆どの仮想視点において選択基準 [a] と選択基準 [b] の結果は同じになり、異なるのはカメラ間に仮想視点がある狭い区間だけであることも考慮すると、実際にアプリケーションに応用する際は、どちらの選択基準でも差はあまりないと結論付けられるであろう。

### 6.3 実験 B:通常手法との比較

次に、提案手法と通常手法間での比較をおこなった結果を示す。ここでは提案手法として、

上記実験で僅かながらも低い評価となった選択基準 [a] を用いた。被験者は 14 人であり、1 人につき 20 組の画像を閲覧してもらったことから、280 個のデータが得られた。図 10 は実験に用いた画像の一部であり、左側が通常手法、右側が提案手法である。図の最上段は比較的明確に提案手法の効果が見れるが、中段の画像は提案手法のほうがやや不自然な結果となっている。下段は、通常手法において顔の中央部分にテクスチャの切れ目があるが、目立たない。一方、提案手法でも比較的 unnatural マッピングになっていないため、違いが判りにくい結果となっている。このように、明らかに有利な条件だけでなく、様々な条件の画像も用いて実験をおこなった。

その結果、提案手法による結果画像が 201 組のデータにおいて選択された。これは全体の 71% にあたり、 $\chi^2$  検定によって有意水準 1% で有意差が認められた。この結果は、提案手法によるレンダリングは表情の判別やゲイズアウェアネスが必要なテレビ会議システムやミーティングのアーカイブシステム等のアプリケーションにおいて自由視点映像を用いる場合に有効であることを示唆している。

## 7. おわりに

本論文では、Visual Hull を用いた自由視点映像をゲイズアウェアネス等の顔部分の品質保証が要求されるグループウェアに応用する目的で、顔部分のテクスチャに一貫性を持たせるレンダリング手法を提案した。これは、顔部分のテクスチャが貼られるファセットにおいて、テクスチャを取得するカメラを限定することで実現され、この限定するカメラの選択基準を 2 つ挙げた。この 2 つの基準のうち、どちらが顔の表情を読み取りやすいか被験者実験によって評価したところ、どちらかに優劣はつけられないことが。また、提案手法によるレンダリング結果と通常手法によるレンダリング結果を比較した結果、提案手法が優れているという結果となった。

提案手法は、品質の低いモデルに対してあまり計算コストをかけずにグループウェアにおいて重要視される顔部分のテクスチャの一貫性を、単一のカメラのテクスチャを用いるという戦略で補償した。しかし、被験者実験において少数ながら通常手法のほうが優れているとされた画像では、2 つのカメラのテクスチャが顔に含まれているにも拘らず、さほどの不整合はおこっておらず、むしろ単一のカメラを無理にマッピングするよりも自然な結果となっていた。そこで、整合性を安価に計測できれば、無理に提案手法で補正する必要がなくなり、レンダリング結果がより安定すると考えられるが、これは今後の課題とする。

謝辞 本研究は、科研費 (20700122) の助成を受けたものである。

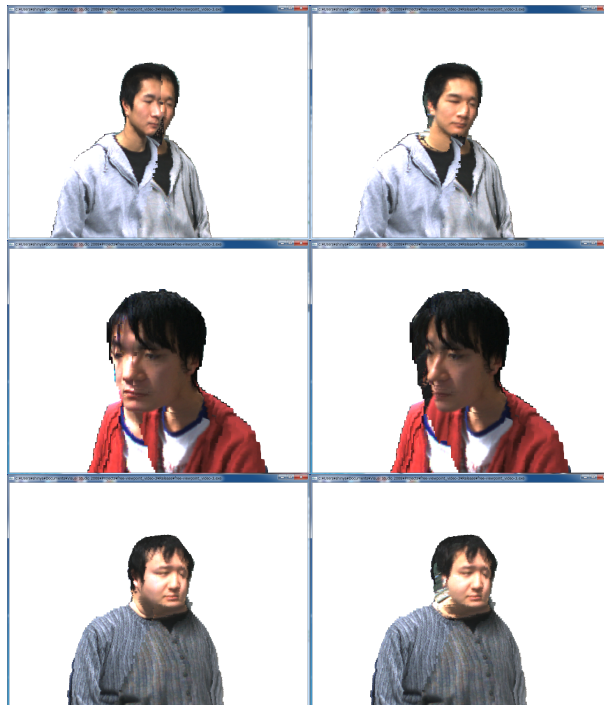


図 10 実験 B に用いた画像の一部  
Fig. 10 images used for experiment B.

### 参 考 文 献

- 1) 井上智雄, 岡田謙一, 松下温: 空間設計による対面会議と遠隔会議の融合: テレビ会議システム HERMES, 電子情報通信学会論文誌 D, Vol. J80-D2, pp. 2482-2492 (1997).
- 2) Stiefelwagen, R., Chen, X. and Yang, J.: Capturing Interactions in Meetings with Omnidirectional Cameras, *International Journal of Distance Education Technologies*, Vol.3(3), pp.32-45 (2005).
- 3) 坂本竜基, 金韓成, 伊藤禎宣, 鳥山朋二, 北原格, 小暮潔: 全方位カメラによる会議撮影システムが意思決定の非同期的伝達に及ぼす影響の評価, 情報処理学会論文誌, Vol.50, No.1, pp.289-301 (2009).
- 4) Takao, S. and Innami, I.: The effects of the two modes of video-conferencing on

the quality of group decisions, *SIGCPR '98: Proceedings of the 1998 ACM SIGCPR conference on Computer personnel research*, New York, NY, USA, ACM Press, pp. 156-158 (1998).

- 5) 古山孝好, 向川康博, 亀田能成, 大田友一: サッカーの自由視点映像のネットワークを用いたライブ配信, 第3回情報科学技術フォーラム (FTS2004) (2004).
- 6) 井上智雄, 岡田謙一, 松下温: テレビ番組のカメラワークの知識に基づいたTV会議システム, 情報処理学会論文誌, Vol.37, pp.2095-2104 (1996).
- 7) Magnor, M., Pollefeys, M., Cheung, G., Matusik, W. and Theobalt, C.: *Video-based rendering*, AK Peters (2005).
- 8) Cerf, M., Harel, J., Huth, A., Einhäuser, W. and Koch, C.: Decoding What People See from Where They Look: Predicting Visual Stimuli from Scanpaths, pp.15-26 (2009).
- 9) ISHII, H. and Kobayashi, M.: Clearboard: A Seamless Medium for Shared Drawing and Conversation with Eye Contact, *Proc. of CHI'92*, pp.525-532 (1992).
- 10) Cline, W. E. L. H.E.: Marching Cubes a High Resolution 3D Surface Construction Algorithm, *Computer Graphics*, Vol.21, No.4, pp.163-169 (1987).
- 11) 山崎俊太郎, 佐川立昌, 川崎洋, 池内克史, 坂内正夫: 微小面ビルボードイングを用いた複雑なシーンの表示手法, 画像の認識・理解シンポジウム (MIRU2002) 予稿集, Vol.1, pp.127-132 (2002).
- 12) Kim, H., Kitahara, I., Sakamoto, R. and Kogure, K.: An immersive free-viewpoint video system using multiple outer/inner cameras, *Proc. 3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pp.782-789 (2006).
- 13) 延原章平, 和田俊和, 松山隆司: 弾性メッシュモデルを用いた多視点画像からの高精度3次元形状復元, 情報処理学会論文誌. コンピュータビジョンとイメージメディア, Vol.43, No.11, pp.53-63 (2002).
- 14) 富山仁博, 片山美和, 折原豊, 岩館祐一: 局所的形状特徴に拘束された3次元形状復元手法とそのリアルタイム動画表示, 映像情報メディア学会誌: 映像情報メディア, Vol.61, No.4, pp.471-481 (2007).
- 15) 高井勇志, 松山隆司: Harmonized texture mapping, 映像情報メディア学会誌, Vol.63, No.4, pp.488-499 (2009).
- 16) Sherrah, J. and Gong, S.: Skin Colour Analysis, *University of Edinburgh* (2001).
- 17) ROTHER, C.: GrabCut: interactive foreground extraction using iterated graph cuts, *ACM Transactions on Graphics*, Vol.23, No.3, pp.309-314 (2004).
- 18) Wada, T., Wu, X., Tokai, S. and Matsuyama, T.: Homography Based Parallel Volume Intersection: Toward Real-Time Volume Reconstruction using Active Cameras, *Computer Architectures for Machine Perception, International Workshop on*, Vol.0, p.331 (2000).