

全方位カメラと表情認識による画角を考慮した 写真撮影支援システム

Photography support system considering angle of view with omni-directional camera and facial expression recognition

木内 一隆† 今村 真平† 河合 紀彦†
Kazutaka Kiuchi Shimpei Imamura Norihiko Kawai

1. はじめに

日本には日本視覚障害者団体連合[1]が存在しており、これにより視覚障害者同士の交流や結婚をしたという事例がある。加えて、2011年に発表された視覚障害者を対象とした調査[2]では、約71%の人が最近写真を撮影したことがあるという調査結果が報告されており、多くの視覚障害者が晴眼者と同じように重要な出来事をカメラで撮って写真を他人と共有したいという思いがある。一方で、2021年に発表された視覚障害者を対象とした第1回支援技術利用状況調査報告書[3]の調査結果では、約90%の人がiPhoneのVoiceOver[4]を利用していると報告されている。VoiceOverは指が触れているアプリケーションのアイコンやテキストなど、画面内容を音声で利用者にフィードバックするものである。また、iPhoneのカメラ機能で撮影する際には、フレーム内に写る人の数を認識し、音声で教えてくれる機能があり、視覚障害者でも人を対象とした撮影が可能となっている。しかし、この機能はフレーム内に7割程度の顔が写っていれば、人の顔として認識してしまうため、複数人での自撮りをしたい場合、撮影者たちがフレーム内に収まらないことがある。晴眼者同士であれば、撮影者たちの顔がスマートフォンのカメラで撮影された写真のフレーム内に見切れずに収まっているか、表情が笑顔のタイミングで撮影できているか確認ができる。しかし、全盲の視覚障害者同士で撮影する場合は写真の確認が困難であるという問題がある。

本研究では、視覚障害者が複数人で自撮りする場合を想定し、全方位画像を用いた写真撮影支援システムを提案する。具体的には、全方位カメラを用いて動画を撮影し、各フレームの顔検出および表情認識を行い、撮影者たちがフレーム内に収まり、かつ全員の幸せ値が最も高い透視投影画像を取得する手法を提案する。

2. 関連研究

2.1 スマートフォンでの写真撮影支援

スマートフォンでの写真撮影支援システムとして前章で挙げたVoiceOverがあるが、人の顔が7割程度写っていると人の顔として認識してしまうため、フレーム内に収まらないことがある問題がある。この問題に対してVázquezらのPortraitFramer[5]ではカメラの視界に何人の顔が写っているかを音声で知らせ、スマートフォンのタッチパネルを操作して振動やピッチを手掛かりに顔の位置を探り、思い通りの構図を作ることができる。Balataら[6]は中央構図と黄金比構図を用いて利用者に音声でフィードバックし、Kacorriら[7]は物体認識を適用し、WhiteらのEasySnap[8]は音声の指示に従い、構図・ズームレベル・被写体の明るさ

を適切に調整する。しかし、これらの手法は人や物に限らず撮影対象があらかじめスマートフォンのカメラの画角内に含まれていることを前提としているため、あらかじめ撮影対象の位置を把握しておく必要があり、全盲の視覚障害者には困難であるという問題がある。また、視覚障害者がこれらのシステムを扱うにあたり、ある程度の訓練が必要である問題もある。

2.2 全方位カメラでの写真撮影支援

前節ではスマートフォンのカメラを用いた従来手法を述べたが、撮影対象をカメラの画角内に収めるには利用者である視覚障害者の技量に依存している問題がある。これに対して全方位カメラを用いるアプローチがある。全方位カメラとは360°撮影が可能なカメラであり、撮影対象が画角内に収まらない問題が発生しないデバイスである。この全方位カメラを用いた研究として岩村らはVisPhoto[9]を提案している。VisPhotoは全方位画像の撮影と同時に10秒間の音声録音を行い、その録音した撮影時の音声情報と撮影日時に加え、物体検出を用いて検出した物体のリストを手掛かりに、ウェブインタフェース上で全方位画像を選択し、検出された物体の中から任意の撮影対象を選択し、透視投影画像を出力する。この手法は撮影対象を全方位カメラによって画角内に収めることには成功しているが、撮影者の自撮りを目的としておらず、人の表情を考慮する提案手法とは異なる。

3. 提案手法

3.1 概要

提案システムの処理手順を図1示す。提案システムでは、撮影条件に沿って全方位カメラで動画を数秒撮影する。次に、全方位動画に写っている人物全員の顔の検出および表情認識を各フレームで行う。顔検出する際に、顔以外の誤検出があるため、全フレームの中から誤検出したフレームを取り除く。次に、全方位動画から幸せの感情が最も高いフレームのみを抽出する。次に、抽出したフレームに写っている顔が複数であるとき、最も左と右の顔を抽出する。その後、それらの人物の顔領域の端の平均位置を中心とし、全員の顔が画角内に入るよう画角の算出を行う。

3.2 撮影条件

本研究では、全方位カメラで自撮りする条件として、以下の2項目の条件を設ける。

1. 腕を伸ばさず、または自撮り棒を用いて、カメラの位置を肩と同じ高さにする。
2. 被撮影者以外の人は全方位カメラの画像に写らない。

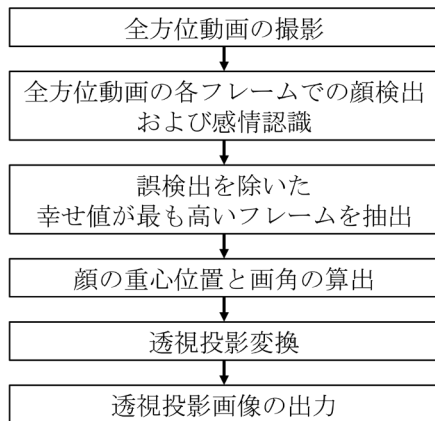


図 1. 提案システムの処理手順

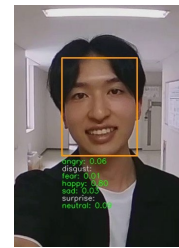
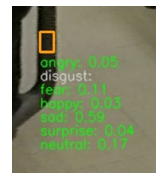


図 2. 顔検出例

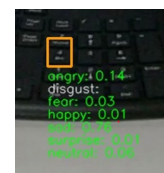
3.3 顔検出と表情認識によるフレームの抽出

3.3.1 顔検出

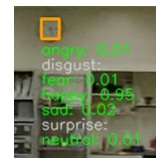
表情認識を行う際には、画像から顔を検出する必要がある。顔の検出には、MTCNN[10] (マルチカスケード畳み込みネットワーク) を用いる。MTCNN は顔を検出するための深層学習モデルであり、3 つの畳み込みネットワークを使用して画像内の顔を検出する。図 2 に顔の検出例を示す。この例では正常に顔検出が行われ、バウンディングボックス取得できている。ただし、全フレームのうちいくつかのフレームで顔以外の誤検出が発生する可能性がある。図 3 は顔検出した際の、椅子・キーボード・時計・モニタといった誤検出の例を示している。



(a)椅子



(b)キーボード



(c)時計



(d)モニタ

図 3. 誤検出例

3.3.2 表情認識

表情認識には、画像内の人物の表情認識を行うことができる FER (Face Emotion Recognizer) [11]を用いる。FER は「恐怖」「無表情」「幸せ」「悲しい」「怒り」「嫌悪」「驚き」の 7 つのカテゴリで感情の認識を行うことが可能である。

本研究の場合は全方位動画から 1 フレームごとに人の顔を検出して感情の認識を行う。図 4 は図 2 の顔検出された際に出力された左下の数値を拡大したものである。その数値は 7 つの感情の数値を全て合わせて 1.00 になるように設定されており、感情の数値が低ければその感情を強く表しておらず、高ければその感情を強く表しているといえる。

3.3.3 誤検出を除いた幸せ値が最も高いフレームの抽出

顔検出と表情認識結果に基づき、全方位動画から幸せの感情が最も高いフレームのみを抽出する。ここではまず、3.3.1 で述べた誤検出を排除するために、全フレームから顔の検出数の最頻値を算出し、フレーム内の顔の検出数が最頻値と一致するフレームのみを抽出対象とする。次に、幸せの感情が最も高いフレームを抽出するため、以下の式(1)により定義する全体幸せ値 H を各フレームで求め、幸せ値が最も高いフレームを抽出する。

$$H = Ave - Var \quad (1)$$

ただし、 Ave , Var はそれぞれ表情認識された撮影者たちの幸せの感情の値の平均値, 分散を表す。これにより、全



図 4. 表情認識例

体の幸せ度が高く、かつ撮影対象者の幸せ値が均一であればつきが少ないフレームを抽出する。

3.4 最も左と右の顔の決定

本研究では全方位画像を扱っており、画像の左端と右端がつながっている。このため、全方位画像上での顔の左右の位置関係が必ずしも実際の位置関係と合致しているとは限らない。このため、後述する透視投影画像上で現れる最も左の顔と右の顔を決定する。ここでは、ある顔から画像の x 軸の正方向で最も距離が長い顔までの距離を算出する。これを全ての顔で行い、距離が最も短い顔のペアを左の顔および右の顔として抽出する。例えば、図 5 を用いて説明すると、A, B, C は三人の顔を表している。まず、A が最も左の顔として見た時の、 x の正方向で一番距離の遠い顔を探索して距離を求める。今回の場合は、C である。同様に B, C も求める。次に、それぞれの一番遠い顔の距離の中で最も短い距離を求める。図 5 の場合は B を最も左の顔として見た時に一番距離が遠い A が最も短い距離であると言える。つまり、B が最も左の顔、A が最も右の顔と決定づけられる。

3.5 透視投影画像への変換

3.3 節の手法で得られた幸せ値が最も高い全方位のフレームと 3.4 節の手法で得られた最も左と右の顔の座標を用いて、透視投影画像への変換を行う。

全方位画像から透視投影画像への変換を行うには、全方位画像と透視投影画像の対応関係を示している以下の(1)(2)(3)の3つの対応関係を組み合わせて求める。図6に示すように全方位画像と透視投影画像の座標を対応付け、全方位画像から透視投影画像への変換を行う。

- (1) 全方位画像と球の3次元位置の関係
- (2) 透視投影画像と平面の3次元位置の関係
- (3) 同一光線が通る球と平面の3次元位置の関係

3.5.1 全方位画像と球の3次元位置の関係

全方位画像と球の3次元位置の関係を求める。図7は球の3次元位置を示しており、二つの角 (θ, φ) と半径 r を用いて3次元座標 (x, y, z) を表すと、式(2)のように表現できる。

$$\begin{cases} x = r \sin \theta \cos \varphi \\ y = r \sin \theta \sin \varphi \\ z = r \cos \theta \end{cases} \quad (2)$$

また、光線の色は減衰しないものとする、パノラマ画像の画素は r に依存しないので $r = 1$ とし、式(3)のように表せる。

$$\begin{cases} x = \sin \theta \cos \varphi \\ y = \sin \theta \sin \varphi \\ z = \cos \theta \end{cases} \quad (3)$$

図8は正距円筒図法で展開した全方位画像を示しており、縦の長さを h_p 、横の長さを w_p とすると、それぞれが球の軸周りの $\pi, 2\pi$ に対応する。全方位画像の座標 (u, v) と3次元位置を表す極座標系 (θ, φ) の関係を表すと式(4)、式(5)のようになる。

$$\varphi: u = 2\pi: w_p \quad (4)$$

$$\theta: v = \pi: h_p \quad (5)$$

式(4)、式(5)から3次元位置を表す (θ, φ) は式(6)、式(7)のように表される。

$$\varphi = \frac{2\pi u}{w_p} \quad (6)$$

$$\theta = \frac{\pi v}{h_p} \quad (7)$$

式(3)、(6)、(7)より、全方位画像の座標と球の3次元位置の関係は式(8)によって表すことができる。

$$\begin{cases} x = \sin \frac{\pi v}{h_p} \cos \frac{2\pi u}{w_p} \\ y = \sin \frac{\pi v}{h_p} \sin \frac{2\pi u}{w_p} \\ z = \cos \frac{\pi v}{h_p} \end{cases} \quad (8)$$

3.5.2 透視投影画像と平面の3次元位置の関係

透視投影画像と平面の3次元位置の関係を求める。図9は球と接する平面の3次元位置を表したものであり、 \mathbf{a}, \mathbf{b} は平面を構成する直角な2つの単位ベクトル、 \mathbf{o} は画像の中心となる平面上の1点を表している。図10は透視投影画像と平面の3次元位置の関係を表している。図10の左側の図

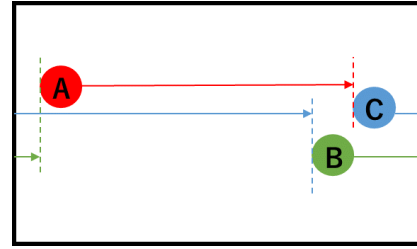


図5. 最も左の顔と右の顔の決定例

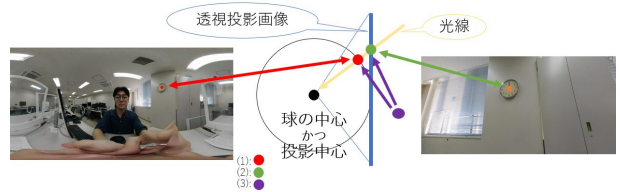


図6. 全方位画像と透視投影画像の対応関係

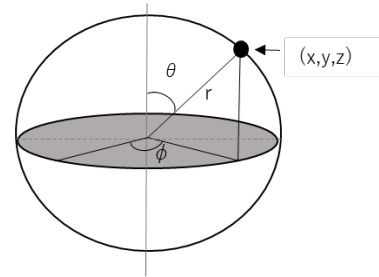


図7. 球の3次元位置

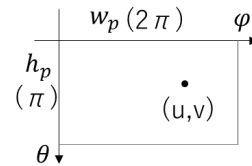


図8. 球の3次元位置の関係を表した全方位画像

は縦の長さ h 、横の長さ w の透視投影画像を表しており、図10の右側の図は縦の長さ m 、横の長さ l の3次元平面を表している。このとき、式(9)は図10の3次元平面上のある点を表す。

$$\mathbf{o} + s\mathbf{a} + t\mathbf{b} \quad (9)$$

図10と式(9)より、 (p, q) と (s, t) の関係は平行移動と拡大縮小の関係であり、透視投影画像と平面上の3次元位置の関係は式(10)によって表すことができる。

$$\begin{bmatrix} s \\ t \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ 1 \end{bmatrix} \quad (10)$$

3.5.3 同一光線が通る球と平面の3次元位置の関係

同一光線が通る球と平面の3次元位置の関係を求める。図11は球の中心に向かって投射される光線が球の表面と3次元平面に交差している関係を表したものである。図11にある点 (x, y, z) は球の3次元座標で式(11)と表すことができ

る。

$$(x, y, z) = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta) \quad (11)$$

このとき、この点と球の中心を通る直線はパラメータ k を用いて、式(12)のように表すことができる。

$$k \begin{bmatrix} \sin \theta \cos \varphi \\ \sin \theta \sin \varphi \\ \cos \theta \end{bmatrix} \quad (12)$$

図 11 より、球の中心を通る直線と平面が交わるので同一光線が通る球と平面の 3 次元位置の関係は式(13)によって表すことができる。

$$k \begin{bmatrix} \sin \theta' \cos \varphi' \\ \sin \theta' \sin \varphi' \\ \cos \theta' \end{bmatrix} = \mathbf{o} + s\mathbf{a} + t\mathbf{b} \quad (13)$$

3.5.4 画像の中心の算出

透視投影画像の中心に対応する 3 次元座標は 3.5.2 節で述べた \mathbf{o} であり、 \mathbf{o} に対応する全方位画像内での座標を求める。ここでは、3.4 節の手法で得られた最も左の顔と右の顔のそれぞれのバウンディングボックスの中心の x 座標の平均、全方位画像内の最も上の顔と下の顔のバウンディングボックスの中心の y 座標の平均を全方位画像内での座標 (u, v) とし、これを式(6)、(7)に代入することで、透視投影画像の中心に対応する 3 次元位置 \mathbf{o} を表す (θ', φ') を求める。

3.5.5 画角の算出

本研究では撮影者たち全員の顔が画角内に見切れずに写されていることが重要であり、画角を適切に算出することが必要である。図 10 の 3 次元平面の図の縦の長さ l 、横の長さ m が画角を決定するため、顔の座標から (l, m) を算出する。まず、3.5.1 節の式(6)、(7)に最も左の顔のバウンディングボックスにおける左端の x 座標、最も上の顔のバウンディングボックスにおける上端の y 座標を (u, v) として代入することによって、透視投影画像の左上端に対応する (θ'', φ'') を求めることができる。3.5.3 節の式(13)のベクトルをその要素で表すと式(14)のように変換できる。

$$\begin{bmatrix} k \sin \theta'' \cos \varphi'' \\ k \sin \theta'' \sin \varphi'' \\ k \cos \theta'' \end{bmatrix} = \begin{bmatrix} o_1 & a_1 & b_1 \\ o_2 & a_2 & b_2 \\ o_3 & a_3 & b_3 \end{bmatrix} \begin{bmatrix} 1 \\ s \\ t \end{bmatrix} \quad (14)$$

ここで、前項で求めた (θ'', φ'') および、ベクトル \mathbf{b} が全方位画像の上方向と同一になるように設定すると、式(14)の右辺は以下のように表すことができる。

$$\begin{bmatrix} \sin \theta' \cos \varphi' & \sin \varphi' & -\cos \varphi' \cos \theta' \\ \sin \theta' \sin \varphi' & -\cos \varphi' & -\sin \varphi' \cos \theta' \\ \cos \theta' & 0 & \sin \theta' \end{bmatrix} \begin{bmatrix} 1 \\ s \\ t \end{bmatrix} \quad (15)$$

式(14)、(15)から、連立方程式を解くことで、 s, t を求めることができる。 (s, t) を式(16)、(17)に代入することによって画角を決める (l, m) を求める。

$$l = 2s + i \quad (16)$$

$$m = 2t + i \quad (17)$$

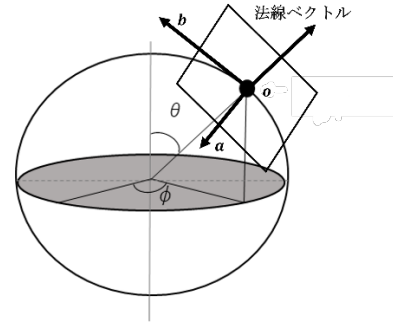


図 9. 球における平面の 3 次元位置

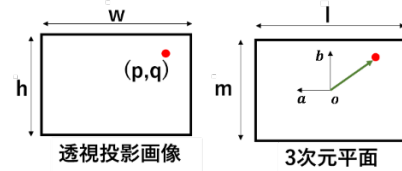


図 10. 透視投影画像と平面の 3 次元位置の関係

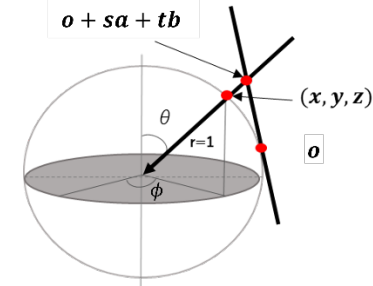


図 11. 同一光線が通る球と平面の 3 次元位置の関係

s, t はそれぞれ左端の顔の x 座標、上端の顔の y 座標から透視投影画像の中心への 3 次元平面上での x 軸方向、 y 軸方向の距離を表しており、 s, t を 2 倍することで全ての顔を画角内に収めることができる。しかし、図 2 の(a)の顔検出例から、髪などがバウンディングボックスの座標に入らないことがあり、顔が少し見切れてしまう可能性がある。そのため、定数 i を加算することで、全員の顔が画角内に入るような (l, m) を算出する。最後に (l, m) を式(18)、(19)に代入し係数 j を乗算することで透視投影画像の横方向の画素数と縦方向の画素数を算出する。

$$w = lj \quad (18)$$

$$h = mj \quad (19)$$

4. 実験と考察

4.1 実験概要

提案手法の有用性を示すために、表 1 に示す 7 種類の全方位動画および条件により透視投影画像の生成を行い、3 種類の検証を行った。実験 1・2 は 誤検出に関する検証をするためのものである。実験 3・4・5 は撮影条件に沿った場合の検証である。実験 6・7 は撮影条件に沿わない場合の検証である。今回の実験では視覚障害者ではなく晴眼者によって実験を行った。検出に関する実験結果を表 2、式(16)、(17)、(18)、(19)を用いた (l, m) と解像度に関する実験結果を表 3 に示す。以下では、3 種類の検証について順に述べる。

表 1. 実験条件

	撮影対象(人)	被撮影対象(人)	入力解像度(px)	環境	定数i	係数j
実験1	3	0	5760×2880	物が少ない	0.3	848
実験2	2	0	3840×1920	物が多い	0.3	$\frac{1696}{3}$
実験3	3	0	3840×1920	物が少ない	0.3	$\frac{1696}{3}$
実験4	5	0	3840×1920	物が少ない	0.3	$\frac{1696}{3}$
実験5	11	0	3840×1920	物が少ない	0.3	$\frac{1696}{3}$
実験6	6	5	3840×1920	物が少ない	0.3	$\frac{1696}{3}$
実験7	7	4	3840×1920	物が少ない	0.3	$\frac{1696}{3}$

表 2. 検出に関する実験結果

	動画時間(s)	全フレーム	最頻値	全員検出されたフレーム	最頻値フレーム	誤検出総フレーム	検出漏れ総フレーム
実験1	3.93	119	3	102	102	12	5
実験2	4.93	149	2	77	77	72	0
実験3	2.61	79	3	70	70	2	7
実験4	1.39	42	4	3	25	0	39
実験5	2.10	64	9	0	28	0	64
実験6	1.72	52	6	19	19	13	20
実験7	1.17	36	6	10	13	8	18

表 3. (l,m)と解像度の実験結果

	(l,m)	解像度(px)
実験1	(2.073657, 0.927167)	1758×786
実験2	(1.995429, 1.136189)	1128×642
実験3	(2.156079, 1.030228)	1218×582
実験4	(2.408225, 1.053676)	1361×595
実験5	(4.382459, 1.490401)	2477×842
実験6	(4.110976, 1.434210)	2324×810
実験7	(83.400572, 19.488982)	47149×11017

4.2 実験 1・2 : 誤検出と解像度に関する検証

本節では人以外の誤検出と解像度に関する検証を行う。実験 1 では 3.93 秒、全 119 フレームの全方位動画から 102 フレームが 3 人全員検出され、かつ最頻値は 3 であり、人の顔ではない誤検出が合計 12 フレームあった。実験 1 の幸せ値の最も高いフレームを図 12、生成画像を図 13 に示す。実験 2 では 4.93 秒、全 149 フレームの全方位動画から 77 フレームで 2 人全員検出され、かつ最頻値は 2 であり、人の顔ではない誤検出が合計 72 フレーム見られた。実験 2 の幸せ値の最も高いフレームを図 14、生成画像を図 15 に示す。図 13 と図 15 より撮影対象全員が画角内に収まった表情の良い透視投影画像が出力されていることが確認できる。

検出に関して、実験 1 の周囲に物が少ない環境では誤検出数が少ないことが確認でき、実験 2 の周囲に物が多い環境では最頻値フレーム数の 77 に近い合計 72 フレームと誤検出が多いことがわかる。また、誤検出の 72 フレーム中 62 フレームが撮影対象の 2 人より 1 人多い 3 人と検出されている。このことから更に物が多い環境において誤検出が増加した場合、提案手法では、誤検出を含むフレームを抽出してしまう可能性がある。そのため、最頻値ではない誤検出を排除する仕組みの考案が求められる。しかし、実験 1 のように周囲に物が少ない環境において提案手法は有効であると言える。また、実験 1 と 2 では異なる解像度でも画像が生成されていることから、使用する全方位カメラの種類に制限がないと言える。

4.3 実験 3・4・5 : 撮影条件に沿った場合の検証

本節では提案手法の撮影条件に沿った場合の検証を行う。まず、表 1・2 より撮影対象 3 人の実験 3 は全方位動画全 79 フレーム中 70 フレームが最頻値の 3 人と検出されており、これは撮影対象の人数と同一である。また、顔以外の誤検出総フレームが 2、検出漏れ総フレームが 7 と実験 1 と同様に周囲に物が少ない環境であるため、最頻値を用いる提案手法が有効であるといえる。また、図 16 に幸せ値の最も高いフレーム、図 17 にそこから生成した透視投影画像を示す。図 17 より、生成された画像の画角は撮影対象全員が収まっていることが確認でき、画角の決定が適切に行われている。

次に撮影対象 5 人の実験 4 では 42 フレーム中 25 フレームで最頻値である 4 人が検出され、その中の幸せ値が最も高いフレームを図 18 に示す。図に示すように 1 人が未検出であるが、5 人全員検出されている図 19 に示すようなフレームは 3 つだけであり、最頻値を用いる提案手法ではこのようなフレームを抽出することはできない問題がある。ただし、提案手法により生成された図 20 に示す結果画像では、バウンディングボックスが左端・右端・上端・下端になる人物が検出されていることから結果的には、撮影対象全員が画角内に収まる透視投影画像が生成できた。

撮影対象 11 人の実験 5 では 64 フレーム中 28 フレームで最頻値となる 9 人が検出され、その中の最も幸せ度が高いフレームでは右側の 2 人が未検出になっていることが図 21 からわかる。未検出の 2 人は全く検出されていない訳ではなく、図 22 に示すようにフレームによっては検出されている。このことから、最頻値の 9 人は同一の 9 人を示すものではなく、検出人物が入れ替わっていることが推察される。図 23 に示す生成画像では右端の人物が検出されていないため、撮影対象全員が画角内に収まった透視投影画像の生成には失敗していることが確認できる。また、表 2 に示すように全員検出されているフレームは 0 であり、すべてのフレームに検出漏れが含まれていることがわかる。

実験 3 から未検出や誤検出が少ない場合において提案手法が有効に動作することが確認できた。しかし、実験 4 と 5 より、髪の毛や帽子の影響によって顔検出の制度が低下して、最頻値が撮影対象の人数と異なった場合には、検出された人物の位置によっては全員が画角内に収まる場合もある一方、人物が見切れる場合も生じた。このことから一度検出した人物の座標を記録し、未検出が発生した場合にその座標を用いるなどの改善が課題として挙げられる。

4.4 実験 6・7 : 撮影条件に沿わない場合の検証

本節では提案手法の撮影条件である、撮影者以外の人は全方位カメラの画像に写らないという条件に沿わない場合の検証を行う。図 24 に示す実験 6 の幸せ値が最も高いフレームでは撮影対象の 6 人全員の顔が検出されており、被撮影対象の 5 人の顔は横や後ろを向いていることや遠い位置にいたことから検出されておらず図 25 に示す結果が得られた。しかし、図 26 に示す実験 7 では左端の非撮影対象者の顔が検出されているため、図 27 に示す結果が得られた。また、表 3 より解像度が 47149×11017 であり、x 軸方向に離れた人物がいる場合に透視投影画像の画角が大きくなりすぎる結果、違和感のある画像が生成されている。



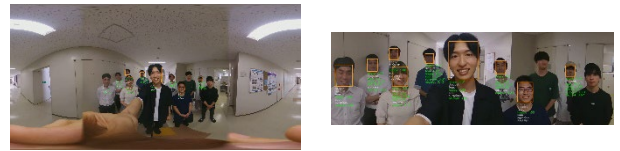
(a)全方位画像 (b)(a)の拡大画像
図 12. 実験 1 の幸せ値の最も高いフレーム



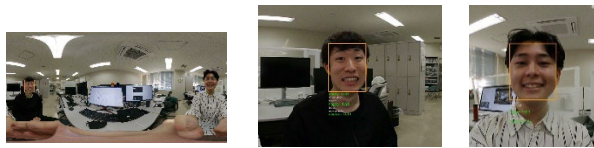
図 20. 実験 4 の生成画像



図 13. 実験 1 の生成画像



(a)全方位画像 (b)(a)の拡大画像
図 21. 実験 5 の幸せ値の最も高いフレーム



(a)全方位画像 (b) (a)の拡大画像 1 (c) (a)の拡大画像 2
図 14. 実験 2 の幸せ値の最も高いフレーム



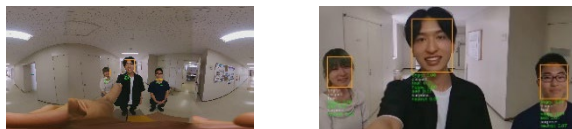
(a)全方位画像 (b)(a)の拡大画像
図 22. 実験 5 の未検出の 2 人が検出されたフレーム



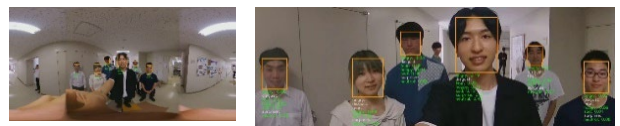
図 15. 実験 2 の生成画像



図 23. 実験 5 の生成画像



(a)全方位画像 (b)(a)の拡大画像
図 16. 実験 3 の幸せ値の最も高いフレーム



(a)全方位画像 (b)拡大画像
図 24. 実験 6 の幸せ値の最も高いフレーム



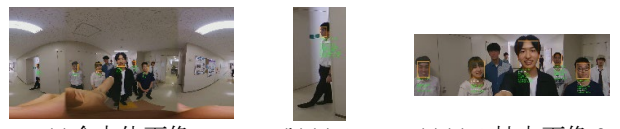
図 17. 実験 3 の生成画像



図 25. 実験 6 の生成画像



(a)全方位画像 (b)(a)の拡大画像
図 18. 実験 4 の幸せ値の最も高いフレーム



(a)全方位画像 (b)(a)の拡大画像 1 (c)(a)の拡大画像 2
図 26. 実験 7 の幸せ値の最も高いフレーム



(a)全方位画像 (b)(a)の拡大画像
図 19. 実験 4 の全員検出されたフレーム



図 27. 実験 7 の生成画像

5. おわりに

本研究では、視覚障害者が複数人で自撮りする場合を想定し、全方位画像と表情認識を用いた写真撮影支援システムを提案した。実験では全方位動画から表情認識を行い、幸せ値が最も高い全方位画像から検出された顔を含める透視投影画像の生成ができることを確認できた。しかし、顔の検出漏れにより全員が画角に収まらない画像が生成されたことも確認できた。今後の課題として、最頻値ではなく、全員が検出され、かつ誤認識したフレームも含めた幸せ値が最も高いフレームの抽出が挙げられる。

<参考文献>

- [1] 社会福祉法人, 日本視覚障害者団体: “加盟団体一覧”, <http://nichimou.org/introduction/member/> (閲覧日 2022 年 10 月 2 日)
- [2] C. Jayant, H. Ji, S. White, and J. Bigham. “Supporting blind photography” Proc. 13th international ACM SIGACCESS Conf. Computers and accessibility, pp.203–210, 2011.
- [3] JBICT.net, 日本視覚障害者 ICT ネットワーク: “第 1 回支援技術利用状況調査報告書”, <https://jbict.net/survey/at-survey-01>, (閲覧日 2022 年 10 月 4 日)
- [4] iPhone: “Supporting VoiceOver in your app”, https://developer.apple.com/documentation/accessibility/supporting_voiceover_in_your_app, (閲覧日 2022 年 10 月 15 日)
- [5] M. Vázquez and A. Steinfeld, “Helping visually impaired users properly aim a camera,” Proc. 14th international ACM SIGACCESS Conf. Computers and accessibility , pp.95-102, 2012.
- [6] J. Balata, Z. Mikovec, and L. Neoproud, “BlindCamera: Central and golden-ratio composition for blind photographers,” Proc. Multimedia, Interaction, Design and Innovation on ZZZ, No.8, pp.1-8, 2015.
- [7] H. Kacorri, K.M. Kitani, J.P. Bigham, and C. Asakawa, “People with visual impairment training personal object recognizers: Feasibility and challenges,” Proc. CHI Conf. Human Factors in Computing Systems, pp.5839-5849, 2017.
- [8] S. White, H. Ji, and J.P. Bigham, “EasySnap: real-time audio feedback for blind photography,” Adjunct proceedings of the 23rd annualACMSymp. User interface software and technology – UIST '10, pp.409-410, 2010.
- [9] 岩村雅一, 平林直樹, 程征, 南谷和範, 黄瀬浩一, “全方位カメラでの撮影とその後処理による視覚障害者の写真撮影支援”, 電子情報通信学会論文誌 D, Vol.J104-D, No.8, pp.663-667, 2021.
- [10] ipacz. (2016). ipacz/mtcnn (GitHub). Zenodo.<https://github.com/ipacz/mtcnn> ,(閲覧日 2023 年 7 月 18 日)
- [11] J. Shenk, A. CG, O. Arriaga, Owlwasrowk, “fer: Zenodo”, <https://doi.org/10.5281/zenodo.5362356>, (閲覧日 2023 年 7 月 18 日)