

疎なカメラ配置を前提としたオブジェクト抽出

Foreground Object Segmentation Method from Sparsely Arranged Multi-view Cameras

三功 浩嗣
Hiroshi Sankoh内藤 整十
Sei Naito

1. まえがき

近年、遠隔拠点間でビデオ会議を行う際に、双方の参加者があたかも同じ空間を共有しているような感覚を味わえるテレプレゼンス技術が注目されている。当該技術により、遠隔拠点間での会議に必要な移動コストを削減できるため、多国籍企業を中心に導入が進んでいる。しかしながら、現状のテレプレゼンスシステム[1]では、会議室の壁紙や机・椅子などの備品も含め、物理的によく似た空間レイアウトを各拠点に構築する必要があるため、導入コストの観点から、普及の障壁となっている。

一方、筆者らは、一般的なオフィスの会議室や、居室空間を対象に、特殊な設備や機器を用いることなく、数台のカメラで撮影される映像情報のみから没入感の高いテレビ会議を実現することを目指している。具体的には、図1に示す通り、各拠点の撮影映像から、会議参加者の人物領域を分離・抽出し、対向拠点の背景領域に当該人物領域を合成、表示するようなシステムを想定している。当該システムを実現するためには、撮影映像から人物領域を正確に抽出する必要がある。しかしながら、グリーンバックのクロマキー環境とは異なる、一般的な撮影環境下で、人物領域を正確に抽出することは技術的に困難であった。

上記の課題に対して、筆者らは、複数カメラを用いる方式を提案し、一定の撮影条件下における有効性を示した[2]。しかしながら、360度全周囲からの密なカメラ配置を前提としているため、オフィス内の会議室や居室空間で想定される疎なカメラ配置に対してそのまま適用することは困難である。また、オフィス内では、蛍光灯による照明変動や、人物とよく似た色をもつ物体が含まれることが多く、抽出精度の向上には方式の洗練化が必須である。

以上を踏まえ、本稿では、オフィス内の会議室や居室等の一般空間において、疎なカメラ配置で取得される映像を対象に、高精度な人物抽出手法を提案する。

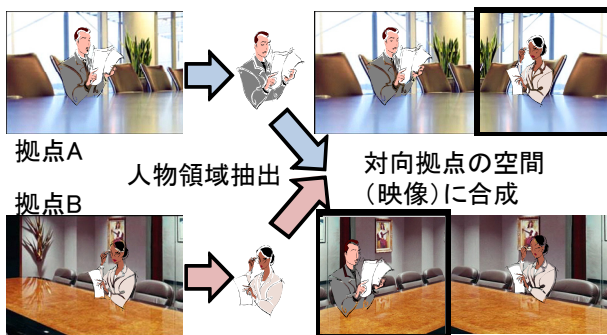


図1. テレプレゼンスのための空間共有イメージ

2. 提案手法

1章で述べた被写体抽出の課題を踏まえ、本稿では、複数カメラの画像情報を統合する手法[2]の方式拡張について提案する。すなわち、各カメラの画素毎に背景か前景かを決定するのではなく、3次元空間における被写体の存在確率、およびカメラ間の色彩類似度に基づき、被写体形状を推定する手法について提案する。提案手法の処理手順は以下の通りである。

0. カメラ毎に固有のボクセル空間を設定・初期化

1. 各カメラの画素毎にオブジェクト存在確率関数を定義
2. カメラ間の色彩類似度関数を定義
3. 各ボクセルのオブジェクト存在確率を算出
4. 各ボクセルのカメラ間での色彩類似度を算出
5. エネルギー関数最適化により前景領域を抽出

以下、各処理の詳細について述べる。

初期化処理として、カメラ毎に固有のボクセル空間を設定するものとする。

ステップ1では、各カメラの画素毎にオブジェクト存在確率関数を定式化するために、背景領域のモデル化を行う。すなわち、各カメラ cam において、被写体を含まない背景(空舞台)映像を複数フレーム分用意し、各画素 (x, y) の色成分 c 毎に、輝度値 $I_{cam}^{(c)}(x, y)$ の平均 $u_{cam}^{(c)}(x, y)$ および分散 $\sigma_{cam}^{(c)}(x, y)$ を算出することで、式(1)により存在確率関数を定義する。

$$\rho_{cam}^{(c)}(x, y) = 1 - \exp\left(-\frac{(I_{cam}^{(c)}(x, y) - u_{cam}^{(c)}(x, y))^2}{2(\sigma_{cam}^{(c)}(x, y))^2}\right) \quad (1)$$

ステップ2では、カメラ間の色彩類似度を定義する。カメラ cam の画素 (x, y) 、および当該画素の色彩類似度 $R^N(cam^{(x, y)}, cam^{(x, y)})$ は、各画素を中心とする $N \times N$ のウィンドウを用いて、ゼロ平均正規化相互相関 (ZNCC) に基づき定式化するものとする。

ステップ3では、各カメラの中心射影行列により、各ボクセルの投影画素を特定し、式(1)によりオブジェクト存在確率を算出する。

ステップ4では、各ボクセルの各カメラに対する投影画素を特定し、式(2)、および式(3)に基づきカメラ間の色彩類似度を算出する。注目カメラの隣接カメラにおける色彩類似度、およびオブジェクト存在確率により重み付け和を算出する。

ステップ5では、カメラ毎に設定された各ボクセル空間について、ボクセル間の隣接関係を考慮したエネルギー関数を定義し、最適化の枠組みでオブジェクト3次元形状を推定する。エネルギー関数は、ボクセル単体のオブジェクト存在確率に基づくデータ項と、ボクセル間の色彩類似度の差に基づく平滑化項を用いて式(2)により定式化する。

$$E(\mathbf{a}) = \sum_k U(\alpha_k) + \lambda \sum_{(k, l) \in N_s} V(\alpha_k, \alpha_l) \quad (2)$$

† (株) KDDI 研究所

3. 実験結果

提案手法の有効性を検証する目的で、一般的なオフィスの居室空間を対象に実験を行った。単純な背景差分を適用するだけでは正確な抽出が困難であるようなシーンとして、背景領域に蛍光灯や、照明を反射する窓、被写体テクスチャと似た色を持つ物体等が含まれるような撮影環境を手配した。また、デプスに基づく抽出手法との性能比較を行う目的から、Microsoft Kinect 3台による撮影を行った。RGB映像の解像度は1280x960、デプス映像の解像度は640x480で取得した。実験に用いた各カメラ映像の先頭フレームを図2に示す。

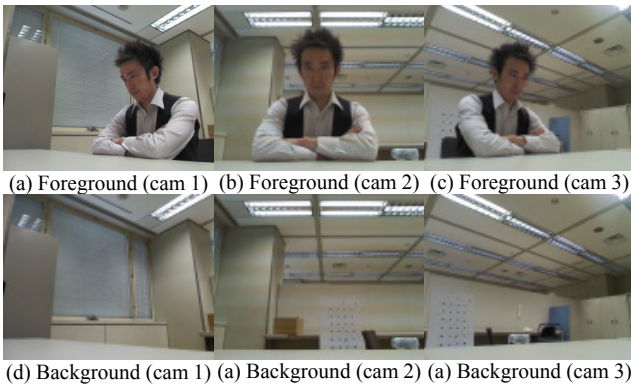


図2. 実験に用いたカメラ映像の例

比較方式として、単一のカメラ映像のみを用いる背景差分方式[3]、複数カメラ映像を用いる方式[2]、デプスに基づく抽出方式[4]を実装し、提案手法と同様に抽出結果を取得した。各手法を適用した結果を、それぞれ図3から図6に示す。各手法のパラメータは予備実験の結果に基づき、最適な値を選択した。

図3の結果より、提案手法では欠損、および不要部の割合を最小限に低減できていることがわかる。図4の結果に注目すると、(a)では頭髪の領域に、(b)では腕の領域にそれぞれ欠損が生じていることがわかる。パラメータチューニングにより、欠損を減らす方向に調整した場合、不要部が生じる現象が確認された。

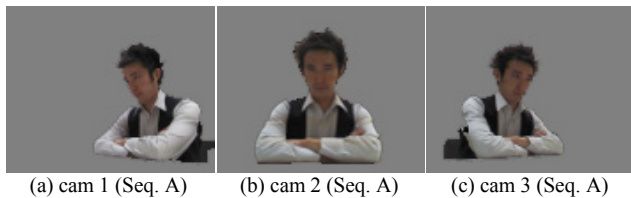


図3. 提案手法を適用した結果

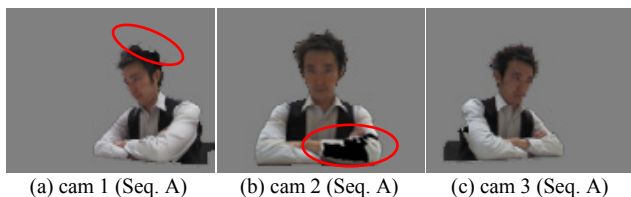


図4. 背景差分方式を適用した結果

一方、図5の結果からは、図4に比べて欠損の割合を低減できているものの、不要部が少なからず生じていることがわかる。これは、複数カメラの活用により、あるカメラでは人物と背景の識別が困難な場合においても、他カメラの情報により識別できていることを示している。しかしながら、当該手法[2]では、密なカメラ配置を前提に、全カメラにおけるオブジェクト存在確率の平均値のみに基づき前景・背景の識別を行うため、カメラ台数が少ない場合には、不要部が誤って抽出される可能性が高いと考えられる。この問題に対して、提案手法では、カメラ間の色彩類似度を取り入れることで、欠損の低減と、不要部の抑制に対して有効に働いていると考えられる。

図6の結果では、エッジ領域を中心にかなり多くの欠損が生じていることがわかる。デプス情報を活用する手法では、ジェスチャー等の大まかな領域を推定するには有効であるが、テクスチャ抽出には精度面で限界があることが確認された。

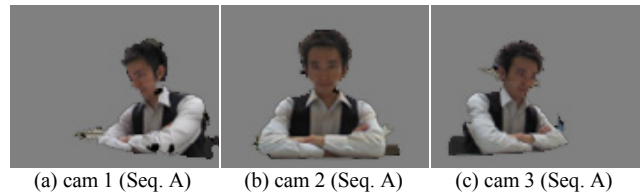


図5. 複数カメラ方式を適用した結果

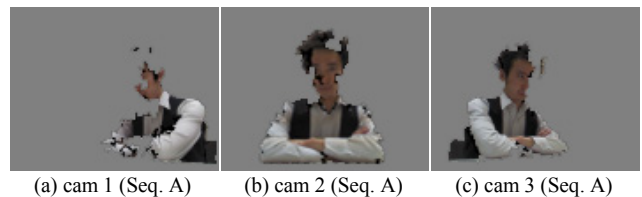


図6. デプスカメラ方式を適用した結果

4. まとめ

本稿では、疎なカメラ配置を前提としたオブジェクト抽出手法について提案した。オブジェクト存在確率とカメラ間の色彩類似度を導入し、3次元空間における隣接関係を考慮したエネルギー関数を導入することで、従来手法に比べて高精度な抽出が可能となることを実験により示した。今後は、GPGPU実装等により処理のリアルタイム化を図る予定である。

参考文献

- [1] T. Szigeti, K. McMenemy and R. Saville, "Cisco TelePresence Fundamentals," Cisco Press, May 2009.
- [2] 三功 浩嗣, 石川 彰夫, 内藤 整, 酒澤 茂之, "確率尤度を用いた3次元形状モデル投影型背景分離法", 映像学誌, Vol.64, No.11, pp.1685-1697(2010)
- [3] S. Morita, K. Yamazawa, and N. Yokoya, "Internet Telepresence by RealTime ViewDependent Image Generation with Omnidirectional Video Camera," In Proc. SPIE Electronic Imaging, Vol.5018, pp.51-60 (2003).
- [4] D. Catuhe, "Programming with the Kinect™ for Windows® Software Development Kit", Microsoft Press, September 2012.