

注視点を考慮した腸管展開画像の適応的表示法

阪井 拓郎, 佐川 立昌, 越後 富夫, 八木 康史
大阪大学産業科学研究所

概要 本論文では、腸管展開画像の適応的表示法を提案する。これまで、内視鏡で撮影された画像列を変形しつつ繋ぎ合わせることによって、パノラマ画像を生成するモザイク手法が提案されている。診断のために医師に画像を提示する際には、入力画像に近いほうがよいが、これまで開始フレームに対して変形を定義し、画像を上書きすることによってパノラマ画像を生成していたため、提示するパノラマ画像は大きく画像処理が施されたものであった。そこで本手法では、注視点を基準としてパノラマ画像を生成し、注視点付近の画像を原画像に近い形で適応的に表示する手法を提案する。本手法は以下の3点からなる。1. 基準フレームに対して変形を定義し、パノラマ画像を生成する。2. 基準フレームを最前面に表示するように画像を整列する。3. 原画像の視野範囲を変更し、異なる視線方向のパノラマ画像を生成する。また、提案手法をグラフィックスハードウェアを用いて実装することにより、インタラクティブな変形、描画を実現した。

Adaptive Displaying Deformable Dissection Image of Intestine by Changing Point of Interest

Takurou Sakai, Ryusuke Sagawa, Tomio Echigo, Yasushi Yagi
Institute of scientific and industrial reserch, Osaka university

Abstract This paper proposes adaptive displaying method for a dissection image of intestine. A method was proposed to get a panoramic image by mosaicing, which deforms and connects a sequence of images. Though an input image without image processing is better for the inspection by a medical doctor, a panoramic image created with the mosaic method includes a lot of image processing, because each of frames is deformed based on the first frame and overwritten sequentially. Therefore, our proposed method creates a panoramic image, which is created for a point of interest, and display a part of the image near by the point without image processing. Our method consists of following three points: 1. Create a panoramic image, which is deformed as to a base frame, 2. Sort images to make the base frame seen on the top of images, 3. Change the scope of original images to create a panoramic iamge of different view direction. We achived interactive deformation and rendering by implementing these methods with a graphics hardware.

1 はじめに

腸管の検査には、しばしば内視鏡が用いられるが、内視鏡は視野が狭く、一度に腸管の一部しか見ることができないという問題がある。腸管の広範囲を一度に見ることができる画像が得られると、医師の検査が容易となり、映像の記録方法としても有用である。内視鏡を用いて、腸管内壁のパノラマ画像を生成する手法として、Pongnumkulらの手法[2]が提案されている。内視鏡を用いて腸管内壁を撮影し、パノラマ画像を生成する際の問題点として、1. カメラ運動、物体形状は未知である、2. 観測対象が近距離から遠距離まで様々な奥行きに分布している、ということが挙げられる。したがって、観測対象を平面に近似するモザイク手法[[1],[3]]は用いることができない。そこで、[2]では多項式関数を用いて画像間の変形を近似し、パノラマ画像を生成する手法を提案した。



図 1: image of endoscope and dissection image

一方、医師が画像を用いて診断を行う際には、入力画像そのものを用いて提示することが望ましい。しかし、[2]の手法では、画像列の最初のフレームを基準として画像の変形を定義し、その後の画像を上書きしていくことによって、パノラマ画像を生成するため、大きく画像処理を施したものを提示することになる。すなわち、開始フレームから離れた画像では、多項式

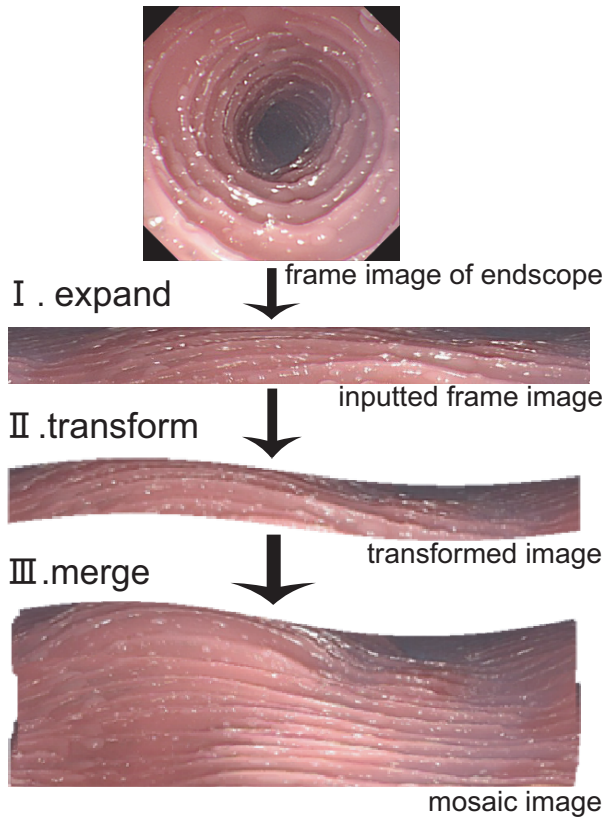


図 2: Generating a panoramic image

変形によって大きく変形し、また、注目すべき点が多数の画像の重なりによって表示される状態となる。

そこで本論文では、ユーザが注視点を指定することによって、以下のような特長を持つパノラマ画像を生成する手法を提案する。

1. 注視点付近を単一の画像で表示する。
2. 注視点付近の画像を基準として変形パラメータを決定する。
3. パノラマ画像生成に用いる視線方向を調節する。

(1)と(2)の特長によって、ユーザが注目する部分を歪みなく提示することができるようになる。また、(3)の特長によって、入力画像の範囲を調節し、視線方向の異なるパノラマ画像の生成を可能にする。以下では、まず2章でパノラマ画像生成法について簡単に説明する。次に、3章において注視点を導入し、パノラマ画像を適応的に表示する手法について説明する。この手法をグラフィックスハードウェアを用いて実装し、高速に表示を行う手法を4章で説明する。5章で腸管模型を用いて撮影した映像に対して提案手法を適用した結果を示す。最後に6章で結論をまとめる。

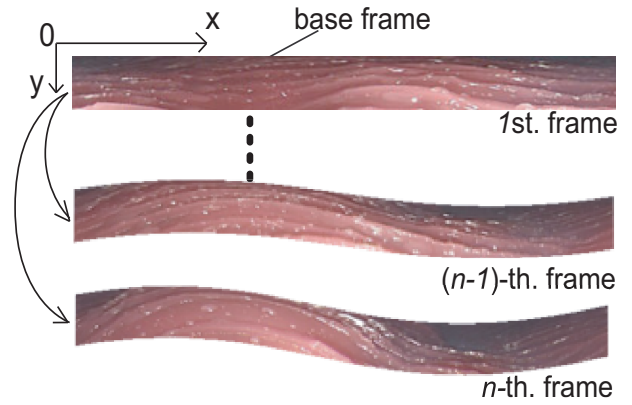


図 3: Transform based on first frame

2 多項式変形によるモザイク画像生成法

本章では[2]において提案されているパノラマ画像生成法について簡単に説明する。パノラマ画像生成の流れは、図2のようになる。図は上からデータの流れを表し、I., II., III. は各データの処理である。まず、図2のI. において入力されたフレーム画像を、腸管を切り開いた状態に展開する。次にII. では、連続する画像間でオプティカルフローを計算し、対応点を求める。その対応を用いて画像を変形する。図3は、この変形の様子を示す。ここでは最初のフレームを基準フレームとし、各フレームは基準フレームにたいして次の式に従って変形される。

$$x_n = f_n^x(x, y) \quad (1)$$

$$y_n = f_n^y(x, y) \quad (2)$$

ここで、 f_n^x, f_n^y は、 n フレーム目の変形関数である。この関数によって n フレーム目の点 (x, y) は、基準フレームの座標系における点 (x_n, y_n) に写像される。 f_n^x, f_n^y は、次式で表される2元多項式関数である。この多項式はフレーム画像間の対応関係から求められる。

$$f_n^x(x, y) = \sum_{i=0}^s \sum_{j=0}^t C_{(n)i,j}^x x^i y^j \quad (3)$$

$$f_n^y(x, y) = \sum_{i=0}^s \sum_{j=0}^t C_{(n)i,j}^y x^i y^j \quad (4)$$

ここで、 s, t はそれぞれ、 x 軸方向の次元数、 y 軸方向の次元数である。 $C_{(n)i,j}^x, C_{(n)i,j}^y$ は、 n フレーム目の変形多項式の係数である。

次に図2のIII. において、図4に示すように変形した各フレームを統合しパノラマ画像を生成する。 $n-1$ 番目のフレームまでが統合されているとき、次の n 番目のフレームはII. の過程で変形され、 $n-1$ 番目までが統合された画像に上書きする。この手順を最初のフレームから順番に行うことによって、パノラマ画像を生成する。

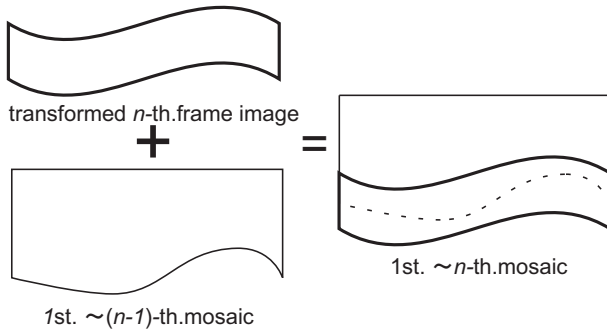


図 4: Merge a new frame

しかし以上で説明した手法では、画像列の最初のフレームを基準として画像の変形を定義するために、最初のフレームから離れた画像は大きく変形し、また計算誤差の蓄積による歪みが発生しているという問題がある。さらに、画像を上書きしていくことによって、パノラマ画像を生成するため、注視点が多数の画像の重なりによって表示されている。

3 注視点を考慮した適応的表示法

本章では、次の3つの手法を説明する。

1. 注視点付近の画像を基準として変形パラメータを決定する手法
2. 注視点付近を単一の画像で表示する手法
3. パノラマ画像生成に用いる視線方向を調節する手法

3.1 基準フレームの変更

まず、 n 番目のフレームを基準フレームに変更することを考える。2章において、基準フレームの点 (x, y) を変形した n フレーム目の点 (x_n, y_n) は、式 (1),(2) で表された。式 (1),(2) における f_n^x, f_n^y の逆関数 f_n^{x-1}, f_n^{y-1} を用いると

$$x = f_n^{x-1}(x_n, y_n) \quad (5)$$

$$y = f_n^{y-1}(x_n, y_n) \quad (6)$$

となる。これにより、1 番目のフレームを基準に変形した n 番目のフレームの点 (x_n, y_n) は、この関数によって、変形していない画像へ変換される。これを $n-1$ 番目のフレームに適用すると

$$x'_{n-1} = f_n^{x-1}(f_{n-1}^x, f_{n-1}^y) \quad (7)$$

$$y'_{n-1} = f_n^{y-1}(f_{n-1}^x, f_{n-1}^y) \quad (8)$$

となり、上式に式 (1),(2) を代入すると

$$x'_{n-1} = f_n^{x-1}(f_{n-1}^x, f_{n-1}^y) \quad (9)$$

$$y'_{n-1} = f_n^{y-1}(f_{n-1}^x, f_{n-1}^y) \quad (10)$$

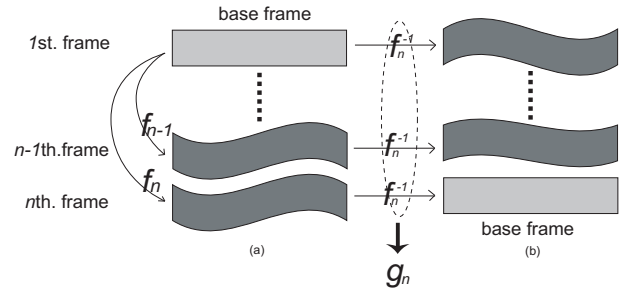


図 5: Change base frame image

(a) 1st frame is base frame. Transform frames with each functions. (b) n -th frame is base frame. Use g_n , instead of f_n^{-1} .

となる。この様子を図 5 に示す。図 5 の (a) では最初のフレームが基準フレームであり、各フレームが多項式関数 f によって変形されることを示す。図の (b) では、基準フレームが n 番目のフレームとなった場合、(a) の全てのフレームを f_n^{-1} で変形することを示している。これによって、 n 番目のフレームは無変形となり、他のフレームは n 番目のフレームを基準に変形される。

以上で、 f_n^x, f_n^y の逆関数 f_n^{x-1}, f_n^{y-1} によって、基準フレームの変更が実現できるが、一般的に多項式 f_n^x, f_n^y の逆関数を求めることは困難である。そこで、 f_n^x, f_n^y を求めるときの画像の対応関係を、逆に多項式 g_n^x, g_n^y を求める。すなわち、 f は $n-1$ 番目のフレームに対して、 n 番目のフレームの誤差を最小化する多項式関数であるが、 g は n 番目のフレームに対して、 $n-1$ 番目のフレームの誤差を最小化する多項式関数である。この g_n^x, g_n^y を f_n^{x-1}, f_n^{y-1} として用いる。

3.2 フレーム配置順序の整列

本節では、歪みのない基準フレーム全体をみることができるよう、最前面に表示する手法を述べる。図 6 左は、一番下のフレームを基準としてパノラマ画像を生成した場合に、これまでのパノラマ画像生成におけるフレームの配置順序を表している。この場合、歪みのない基準フレームが最も奥にあるため、パノラマ画像上に見える部分が少ない。一方、図 6 右は、中央のフレームを基準フレームとして、基準フレームを最前面に配置したときの状態を示している。配置順序の整列を行うと歪みのない基準フレームが最前面にあり、歪みの小さい画像が優先的にパノラマ画像に表示されると、その付近の単一の画像で表される。

3.3 視線方向、視野角の変更

パノラマ画像を生成するために、内視鏡で撮影された画像を展開しているため、フレーム画像の y 軸方向と視線方向が対応している。図 7 に示す y 軸方向に分

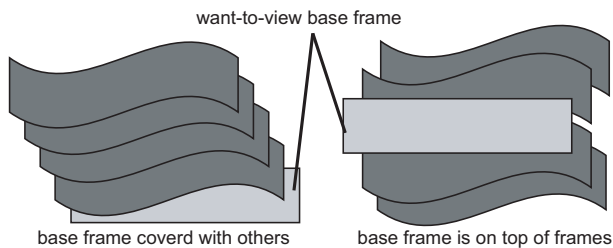


図 6: Arrange frames according to the base frame



図 7: View of part of frame image

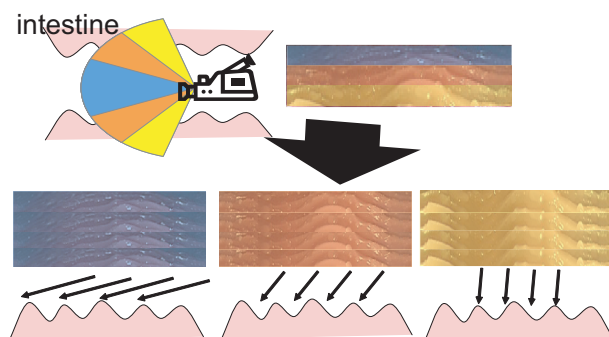


図 8: Eye-direction and part of frame image

けた 2 つの部分で、腸管壁面の見え方が異なっていることがわかる。そこで、 y 軸方向で各フレーム画像の一部を切り取りパノラマ画像を生成することで、視線方向を変更したパノラマ画像を得ることができる。図 8 は内視鏡の視線方向と、その視線方向を切り出した部分画像から、パノラマ画像を生成する様子を示している。図ではフレーム画像を 3 つの方向に対応する部分に分割している。この 3 つの部分画像でモザイクすることによって、3 つの視線方向のパノラマ画像を生成することができる。また図下方では、そのパノラマ画像の腸管に対する視線方向を示している。

4 GPU を用いた適応的表示の高速化

提案手法では、ユーザの入力に対してインタラクティブに画像の描画、変形処理を高速に行う必要がある。そのために、画像の変形、描画処理にグラフィックスハードウェア (Graphics Processing Unit; GPU) を用いて、高速に変形、表示を行う手法を実装する。前節で提案した手法を高速化する方法を、以下にあげるように GPU を用いて実装する。

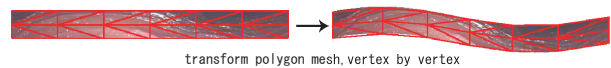


図 9: transform textured polygon mesh

1. メッシュモデルを用いて変形を近似し、パーティックスシェードにおいて変形を実行する。
2. 奥行き情報である z 深度を用いて各フレームの順序を整列する。
3. 視線方向、視野角の変更を実現するため、ピクセルシェードでマスク処理を行い、画像の一部を切り出す。

4.1 メッシュモデルによるフレーム画像の変形

画像の変形のために、ピクセル毎に変形処理を行うことが、計算量の増加の原因となっている。そこで、変形処理を適用する対象を少なくすることによって、計算量の軽減を計り、インタラクティブ性を確保する。具体的には、メッシュモデルを用いて、画像の形状を表し、画像の変形をメッシュモデルの変形によって近似する。図 9 に示すように、画像の形状を表す長方形のメッシュモデルを作成し、各頂点を多項式関数 (f または g) によって変形する。変形を行う画像をメッシュモデルにテクスチャとして貼り付けることによって、画像の変形を実現する。各頂点位置の変更はパーティックスシェードのフラグメントプログラムとして実装する。メッシュモデルの粗さを調節することによって、変形処理対象の数を減らすことができ、計算量を削減可能である。各頂点間のピクセル位置は GPU によって補間されるため、ハードウェアを利用した計算によって高速に実行される。

4.2 z 深度を利用したフレーム順序の整列

3.2 節で、注視点を含むフレームを最前面に表示する方法を述べた。この手法を実現するためには、フレーム順序を整列する必要がある。そこで、各フレームを 3 次元的に描画することによって、前後関係を表現する。フレームの前後関係は、奥行きによって表されるため、フレームの描画順序を考慮する必要がなくなり、フレームの整列処理が簡単化される。

まず、各フレームに対し、 z 深度を用いて奥行き情報を付加する。この奥行き情報を z 深度と呼ぶ。 z 深度は、視点が 3 次元座標の中心にあり、視線が z 軸の正の方向であるときの、 z 軸に対応する。 z 深度が大きいほどオブジェクトは奥にあり、小さいほど手前にある。図 10 に、 z 深度と見え方の関係を示す。図の左は、3 次元座標上で正面からの見え方である。図の右は斜めから見たときの見え方を示している。各フレームに適切な z 深度を設定することによって、フレームの前後関係を表現することができる。

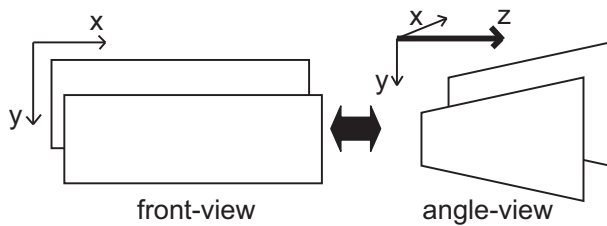


図 10: Sorting by z-depth

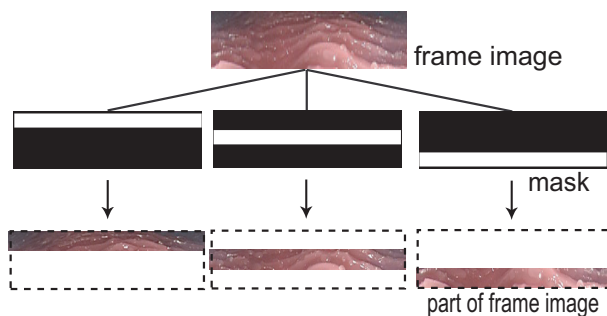


図 11: Mask image for cutting off part of frame image

GPU は出力画像のピクセル毎に z 深度を保持している．そして，新たに描画するフレームの z 深度を，保持している z 深度と比較する．比較の結果 z 深度が小さいとき，対応するピクセルを描画し保持している z 深度を更新する．これより，フレームの描画順序を考慮せずに，フレームの整列を実現することができる．

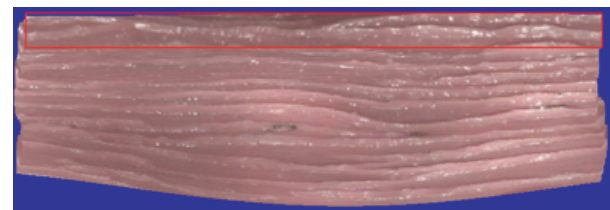
4.3 ピクセルシェードによる部分表示

3.3 節で，視線方向，視野角の変更のために，フレーム画像の一部を切り取る手法を述べた．この手法の実現のために，マスク画像とピクセルシェードによって画像の一部を切り取る方法を述べる．

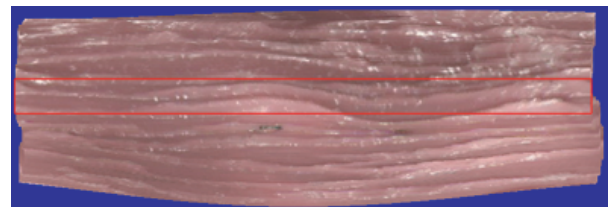
ピクセルシェードはピクセル毎の描画処理を行うが，どのピクセルを描画すべきかを決定するマスク画像をテクスチャとしてピクセルシェードに入力する．マスク画像上で，色値が黒の部分はそのピクセルの色値を透明に設定して出力する．これにより，マスク画像上で白の部分をフレーム画像から切り取ることができる．また，透明部分は描画時に z 深度値が更新されることを防ぐために，透明部分の z 深度を十分大きな値に設定する．図 11 では，フレーム画像からマスク画像を用いて一部分を切り出す様子を示す．図 11 の上段のフレーム画像に対し，中段のマスク画像で切り取ることで，下段のような一部分を切り出した画像を得ることができる．

5 実験

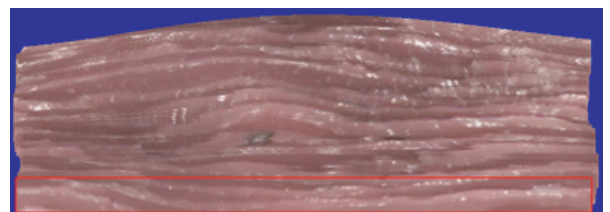
腸管模型を撮影した全 90 フレームの映像に適用し，基準フレームを変更した実験結果を図 12 に示す．図



90th frame is base frame



45th frame is base frame



1st frame is base frame

図 12: Mosaic based on n th frame

Mosaic images from total 90 frames . base frame is, 1st (upper), 45th (middle), 90th (lower) . Enclosed region with red line is undistorted base frame .

は上から，1 番目，45 番目，90 番目のフレームを基準フレームとしたときのモザイク画像である．図の線で囲まれた基準フレームが変更され，パノラマ画像全体が基準フレームに合わせて変形されていることがわかる．

次に，黒い印を付けた腸管模型を撮影した映像に対して適用し，視線方向を変更した結果を図 13 に示す．図は上から，カメラ側方，カメラ斜め後ろ，カメラ後方，の視線方向の部分画像を用いてモザイクした結果である．図の円で囲まれた部分に注目すると，上段のモザイク画像では見ることができない黒い印が，中段のモザイク画像では僅かに見ることができ，下段のモザイク画像でははっきりと確認できる．

次に，処理時間を示す．表 1 は，処理に GPU を使用したときと，ソフトウェアエミュレーションにより全ての処理を CPU で行ったときの結果である．全 90 フレームの画像を使用し，10 回の変形，描画処理の平均をとった．実験を行った計算機の構成は，CPU は Xeon2.8GHz $\times$ 2，メモリが 2Gbyte，そして GPU として Geforce 6800Ultra である．この結果から，インタラクティブ性を得るために十分な高速化が実現されていることがわかる．

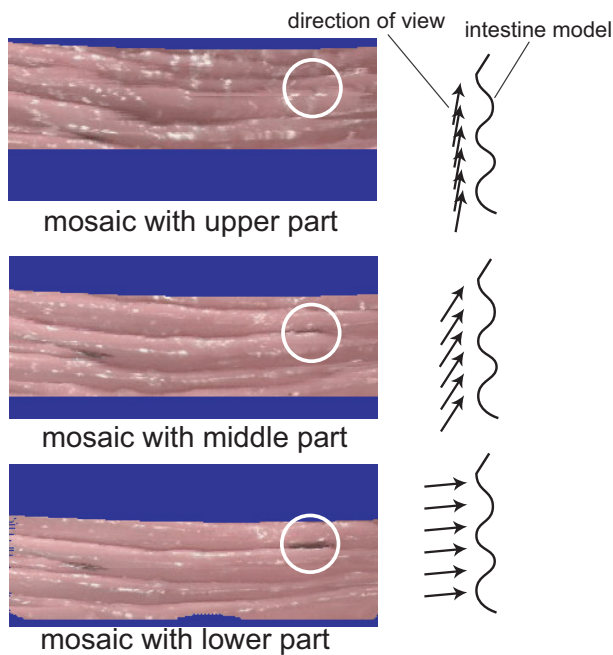


図 13: Change the scope of original images

Mosaic images of different view direction . Figures on right side indicates eye-direction to intestine model.

表 1: Time to draw

	Time(msec)
GPU + CPU	1.4
Only CPU	24438.7

6 おわりに

本論文では、基準となるフレームを選択的に変更し、腸管展開画像の適応的表示法を提案した。提案手法は以下の3点から構成される：1．基準フレームに対して変形を定義し、パノラマ画像を生成する。2．基準フレームを最前面に表示するように画像を整列する。3．原画像の視野範囲を変更し、異なる視線方向のパノラマ画像を生成する。これにより、ユーザに対して、注視点付近の画像を入力画像に近い形で提示するパノラマ画像生成できるようになった。また、提案手法をグラフィックスハードウェアを用いて実装することにより、インタラクティブな変形、描画を実現した。しかし、より長時間の映像に対してモザイクを行う場合、その変形量が発散することが考えられる。よって、そのような問題を解決する表示方法が今の課題である。また提案手法では、変形量と3次元形状との整合性がとられていない。よって、3次元形状を求める手法と組み合わせ、整合性をもったモザイク画像を生成することも今後の課題である。

参考文献

- [1] S. Peleg and J. Herman. Panoramic mosaics by manifold projection, 1997.
- [2] S. Pongnumkul, R. Sagawa, T. Echigo, and Y. Yagi. Deformable registration for generating dissection image of an intestine from annular image sequence. In *Computer Vision for Biomedical Image Applications*, pages 271–280, 2005.
- [3] H. Shum and R. Szeliski. Panoramic image mosaics. Technical Report MSR-TR-97-23, Microsoft Research, 1997.