

시변 볼륨 데이터에서의 등위면 콤포넌트 추적 및 시각화

손 봉 수*

Isosurface Component Tracking and Visualization in Time-Varying Volumetric Data

Bong-Soo Sohn *

요 약

본 논문에서는 시간이 변하는 볼륨 데이터로부터 등위면 콤포넌트를 계산하고 그 움직임을 추적하는 새로운 시각화 알고리즘을 기술한다. 등위면 시각화는 볼륨 데이터를 효과적으로 보여주기 위한 가장 흔한 방법 중 하나이다. 그러나 대부분의 기존 등위면 시각화 알고리즘들은 시간이 정지되어 있는 정적 볼륨 데이터에 국한하여 적용되어 왔다. 이미징 기술과 시뮬레이션 기술이 발전함에 따라 대용량의 시변 볼륨 데이터들이 생성되고 있어 이러한 시변 볼륨 데이터의 동적인 특성을 이용할 수 있는 효과적인 시변 등위면 시각화 알고리즘이 필요하게 되었다. 우리는 우선 두 연속되는 시변 볼륨 데이터에서 정의되는 등위면 콤포넌트들간의 시간적 일치성을 정의한다. 이 정의에 기반하여 킨투어 트리를 이용하여 얻어진 등위면 콤포넌트로부터 그것과 일치되는 그 다음 시간 단계에서의 콤포넌트들을 추적하는 알고리즘을 수행한다. 이러한 과정을 매 시간 단계에서 반복하면, 선택된 등위면 콤포넌트의 동적인 변형 모습을 보여주어 효과적인 시각화를 이룰수 있다.

Abstract

This paper describes a new algorithm to compute and track the deformation of an isosurface component defined in a time-varying volumetric data. Isosurface visualization is one of the most common method for effective visualization of volumetric data. However, most isosurface visualization algorithms have been developed for static volumetric data. As imaging and simulation techniques are developed, large time-varying volumetric data are increasingly generated. Hence, development of time-varying isosurface visualization that utilizes dynamic properties of time-varying data becomes necessary. First, we define temporal correspondence between isosurface components of two consecutive timesteps. Based on the definition, we perform an algorithm that

• 제1저자 : 손봉수

• 투고일 : 2009. 09. 18, 심사일 : 2009. 10. 01, 게재확정일 : 2009. 10. 15.

* 중앙대학교 공과대학 컴퓨터공학부 조교수

※ 이 논문은 2009년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(2009-0064604).

tracks the deformation of an isosurface component that can be selected using the Contour Tree. By repeating this process for entire timesteps, we can effectively visualize the time-varying data by displaying the dynamic deformation of the selected isosurface component.

▶ Keyword : 등위면(isosurface), 컴퓨터 그래픽스(computer graphics), 가시화(visualization)

I. 서론

오늘날의 과학적 시뮬레이션과 CT나 MRI와 같은 스캐닝 측정 장치들은 점점 더 대용량의 시간이 변하는 볼륨 이미지 데이터를 생성하고 있다[1]. 등위면 시각화는 볼륨 데이터를 가시화하는 가장 인기 있는 방식 중 하나이다. 등위면은 $I(w) = \{(x, y, z) | F(x, y, z) = w\}$ 로 나타내어지는 등위 집합(level set)으로 정의된다[2]. 여기서 F 는 볼륨이미지 데이터로부터 정의되는 함수이고, w 는 등위값(isovalue)이다. 시변 볼륨 데이터의 시각화를 위하여 등위면 가시화 기법을 매 시간단계의 볼륨 데이터에 적용하여 시각화 할 수 있다. 그러나, 이러한 단순한 적용은 등위면에 여러개의 콤포넌트들이 있을 때, 그 콤포넌트들을 구분할 수 없기 때문에 그 활용 범위가 제한적이며 데이터의 동적인 특성을 나타내는데 한계를 가진다. 이 논문에서는 등위면 콤포넌트를 컨투어(contour)로 줄여 부르기로 한다.

본 논문이 기여하는 바(main contribution)는 시변 볼륨 데이터에서 정의되는 등위면을 구성하는 콤포넌트들을 구분하여 각 콤포넌트의 변형(deformation)을 추적할 수 있는 새로운 알고리즘을 제시하고, 그것을 시변 등위면의 효과적인 시각화에 적용하는 것이다. 이러한 등위면 추적 알고리즘을 이용하면 시간이 변함에 따라 일어날 수 있는 각 컨투어의 위상 변화를 계산하여 주어진 시변 자료의 동적인 구조 변화를 이해하는데 도움을 준다. 예를 들면 <그림 1>에서 보듯이, 천체 물리학 시뮬레이션을 통하여 우주 공간의 밀도가 시간에 따라 변하는 분포값을 나타내는 시변 볼륨 데이터가 생성된다. 이로부터 우주 갤럭시 클러스터의 생성과정에 대한 정보가 가시화될 수 있다. 이 데이터에서 정의되는 등위면의 각 콤포넌트는 갤럭시 클러스터(cluster)를 나타낸다. 여기에 우리의 추적 알고리즘을 적용하면 등위면 콤포넌트로 표현되는 개개의 갤럭시 클러스터가 언제 생성되고, 다른 클러스터와 합병하고, 소멸되는지 추적해 볼수 있다. 이외에도, 이러한 특징부분 추적(feature tracking) 알고리즘은 다양한 응용 분야를 가지고 있다 [3][4].

제안되는 방법의 입력은 시간이 변하는 볼륨 데이터

$F = \{f^1, f^2, \dots, f^T\}$ 이며 이때 f^t 는 각 시간단계 t 에서의 조각별로 선형인(piecewise linear) 함수이다. 이러한 함수는 정사면체형 그리드에서 정의된다고 가정한다. 다음으로 사용자가 특정 컨투어를 선택하여 추출한다고 가정한다. 이 과정은 컨투어 트리(Contour Tree)를 매 시간단계마다 계산하여 이차원 패널에 그린 뒤 트리의 에지(edge)위 점을 선택함으로써 컨투어를 쉽게 얻을 수 있다. 다음으로 본 논문의 주요 알고리즘인 컨투어 추적(Contour Tracking) 알고리즘을 수행한다. 이 알고리즘은 앞 과정에서 선택된 컨투어로부터 그것과 시간적으로 일치되는 다음 시간단계의 컨투어를 추적하여 계산한다. 위과정을 각 시간 단계에서 반복하여, 선택된 컨투어의 연속적인 변화 과정을 전체 시간에 대하여 추적하여 시각화 한다. 이 과정들을 통해서 각 컨투어들의 위상 변화를 추적할수 있다. 이는 등위면의 동적인 구조 변화를 나타내기 때문에, 시변 데이터에 내재되어 있는 의미있는 구조와 그 변화 패턴을 발견하려는 사용자에게 유용하다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. II장에서 우리는 등위면과 시간변화 데이터의 시각화에 대한 관련 연구를 검토한다. III장은 컨투어의 시간적 일치성을 정의한다. IV장에서 우리는 컨투어 트리를 이용하여 컨투어 추적의 입력이 되는 컨투어를 선택하고 추출하는 과정을 설명한다. V장에서는 선택된 컨투어와 시간적으로 일치되는 컨투어를 선택된 컨투어로부터 추적하는 알고리즘을 설명하고, 이것을 등위면의 위상 변화를 계산하여 시각화에 응용하는 것을 기술한다. VI장은 실험 결과를 보고한다. 마지막으로, VII장에서 우리는 본 논문의 결론을 제시한다.

II. 관련 연구

볼륨 데이터에서 정의되는 등위면을 가시화하기 위해서는, 우선 등위면을 삼각화(triangulation)하여야 한다. 등위면을 삼각화 하는 가장 대표적인 방법은 Lorensen과 Cline이 제안한 Marching Cubes방식이다[5]. 이 방식은 전체 볼륨 이미지를 격자 상에서 존재하는 정육면체형 셀의 집합으로 보아, 각 셀 내의 등위면을 삼각화하면 전체 등위면을 삼각형화할 수 있다는 아이디어를 사용한다. 등위면을 가지고 있는 볼

를 셀만을 방문하여 삼각화를 수행함으로써 성능을 향상시킬 수도 있다. [6] 시변 볼륨 데이터의 시각화는 일반적으로 매우 큰 저장 용량을 가지며 이를 처리하기 위한 강한 계산력을 요구하므로 도전적인 문제로 인식되어 왔다. 시간 중심의 자료구조가 불필요한 I/O를 최소화하고 외부메모리(out-of-core)를 이용하는 등위면 추출을 수행하기 위하여 사용되었다[7]. 고차원 등위면 계산 방법[3]은 시간 변화 데이터를 사차원 공간에서 생각하여 등위면을 계산한다. 먼저 삼차원 솔리드 메쉬 $\{(x,t)|f(x,t)=w\}$ 가 생성되고 여기에서 시간이 t_0 인 점의 집합을 계산함으로써 시간 t_0 에서의 등위면, $\{x|t(x)=t_0\}$, 을 추출할 수 있다. 시변 볼륨 데이터는 그 용량이 크고 많은 시간단계를 가지고 있는 경우가 많기 때문에, 데이터 내의 흥미로운 패턴을 포함하고 있는 특정 시간 단계와 등위값(isovalue)를 자동으로 찾을 수 있다면 유용할 것이다. 컨투어 스펙트럼[8]은 모든 등위값과 시간단계에 대하여 표현적, 부피, 변화율과 같은 등위면의 기하학적/위상학적 성질들을 계산하여 보여주었다. 비슷한 개념으로 개발된 컨투어 평면(contour plane) [9]은 이차원 평면상에 모든 등위값과 시간단계에 대하여 각 등위면의 컨투어 개수를 표시하여 보여주었다.

Silver et al. [4][10]는 볼륨 데이터에서 미리 정의된 경계값(threshold) 기준을 만족하는 영역을 특징부분(feature)으로 정의하였다. 데이터로부터 특징부분을 추출한 뒤 두 연속적인 시간단계의 특징부분들간에 겹치는 정도를 기준으로 일치성 테스트를 수행하여 일치성을 결정한다. 이로부터 각 특징부분의 모양 변화과정을 추적하였다. 이러한 특징부분의 변화 과정 추적을 통하여 생성, 소멸, 합병, 분리와 같은 각 특징부분의 동적인 변화를 추적할 수 있으며 이를 그래프로 나타내어 효과적인 시각화에 도움을 준다. 고차원 등위면 계산도 특징부분 추적에 사용될 수 있다. 컨투어들간의 시간적인 일치성은 이 컨투어들이 속해 있는 고차원 컨투어가 같은지 여부에 따라 결정된다.

컨투어 트리[11][12][13][14]는 영상처리 및 GIS등의 다양한 분야에서 사용되어 왔으며, 우리의 주요 관심사는 시각화에 사용하는 것이다. 컨투어 트리를 효율적으로 계산하기 위한 다양한 방법이 개발되어 왔다. Carr 와 Snoeyink[11]는 컨투어 트리를 등위면의 위상학적 구조를 보여주고 개별화된 컨투어를 선택하여 추출할 수 있도록 유저 인터페이스를 제공하도록 사용하였다. 이를 위하여 컨투어 트리의 각 에지에 대하여 경로씨앗(path seed)를 계산하여 실행 시 선택된 컨투어가 효율적으로 계산될 수 있도록 하는 씨앗셀(seed cell)을 생성하도록 한다[15]. Sohn 과 Bajaj[16]는 컨투어 트리를 시변 볼륨 데이터의 매 시간단계마다 생성하고 연

속적인 매 두 시간단계의 컨투어 트리간의 시간적 일치성을

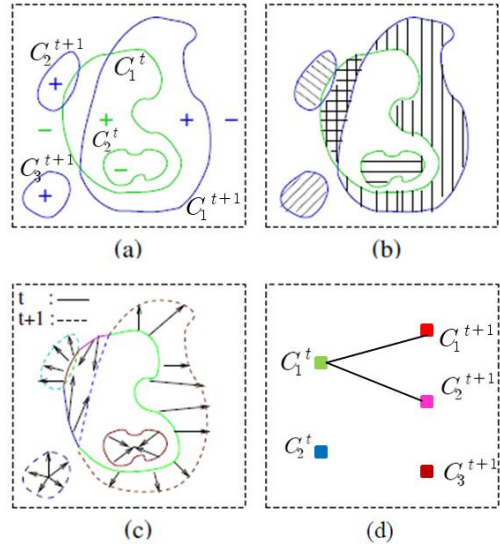


그림 1 두 연속적인 시간단계에서 정의되는 컨투어들간의 일치성 정의의 예.
Fig. 1. Example of Contour Correspondence from Two Consecutive Timesteps

계산하여 시변 등위면의 효율적인 시각화를 제공하였다. [17]은 시간 변화 등위면의 시각화에 대한 최신 연구 결과를 비교 기술하였다.

III. 컨투어의 시간적 일치성 정의

시간단계 t 와 $t+1$ 에서 각각 주어진 두 등위면 $I^t = \{C_1^t, \dots, C_{n_t}^t\}$ 와 $I^{t+1} = \{C_1^{t+1}, \dots, C_{n_{t+1}}^{t+1}\}$ 에 대하여, 일치성(Correspondence) 테스트는 컨투어 $C_k^t \in I^t$ 가 C_k^{t+1} 에 일치하는지 결정한다. 특정한 가정을 세우지 않는 한, 두 이산적인 시간단계 t 와 $t+1$ 사이에 등위면이 어떻게 변하는지 알 수 없기 때문에 컨투어의 시간적인 일치성을 정의하는 데는 여러 가지 방법이 있을 수 있으며, 본 논문은 일치성 테스트를 위한 규칙을 만들어 제공한다.

$X^t(w) = \{x|f^t(x) \geq w\}$ 로 정의되는 영역을 객체 집합이라고 하자. 객체 집합은 객체라고 불리는 콤포넨트들로 구성되어 있다. 이는 $X^t(w) = \{X_1^t, \dots, X_m^t\}$ 로 나타낼 수 있다. 같은 방식으로 Y^t 도 수식으로 나타낼 수 있다. 우리는 X 를 상위객체, Y 를 하위 객체로 명칭한다. 컨투어는 항상 단 하나의 상위 객체와 단 하나의 하위 객체의 경계 위에 있게

된다. 이러한 명칭으로 컨투어의 시간적 일치성은 다음과 같이 정의할 수 있다.

정의1. 두 컨투어 C_k^t 와 C_k^{t+1} 을 생각하자. 만약 C_k^t 가 상위 객체 X^t 의 경계 위에 있고, 하위 객체 Y^t 의 경계위에 있으며, 같은 방식으로 C_k^{t+1} 이 X^{t+1} 과, Y^{t+1} 의 경계에 있다고 가정하자면 다음 명제가 성립한다.

상위객체 X^t 와 X^{t+1} 사이엔 겹침이 있고, 하위객체 Y^t 와 Y^{t+1} 사이엔 겹침이 있다.

$\Leftrightarrow C_k^t$ 는 C_k^{t+1} 과 일치성을 가지고 있다. 이를 $C_k^{t+1} \leftarrow C_k^t$ 로 나타낸다. 이것은 C_k^t 와 C_k^{t+1} 이 다른 시간단계에서의 같은 컨투어임을 의미한다. <정의끝>

<그림 2>는 컨투어 일치성 테스트의 예를 보여준다. (a)에서 볼 수 있듯이, 시간단계 t 의 등위면과 시간단계 $t+1$ 의 등위면이 그려져 있다. 각 컨투어를 경계로 하는 영역을 그 영역내의 벡터들 부호에 따라 "+"와 "-"로 표시되어 있다. (b)에서는 각각의 부호 변화 영역(sign change subregion)이 서로 다른 라인 패턴으로 그려져 있다. (c)에서는 각 컨투어 세그먼트가 다른 색깔로 표시되어 있는데, 검은색 화살표는 각 컨투어 세그먼트의 이동 방향을 나타낸다. 컨투어 C_k^t 위의 각 점은 같은 부호 변화 영역을 공유하는 C_k^{t+1} 의 해당 점으로 이동한다. 화살표가 나타내는 점의 이동을 기준으로 했을 때, $C_1^{t+1} \leftarrow C_1^t$, $C_2^{t+1} \leftarrow C_1^t$, $C_3^{t+1} \leftarrow \emptyset$, $\emptyset \leftarrow C_2^t$ 가 된다. (d)에서 볼 수 있듯이, 이러한 관계는 위상 변화 그래프로 나타내어 사용자에게 유용한 정보로 제공된다.

IV. 컨투어 선택

우리는 컨투어 표면의 변화를 추적하기 전에 특정 컨투어를 선택하여 추출하여야 한다. 이 과정은 컨투어 트리를 이용하면 쉽게 이를 수 있다. 벡터스 집합 V 와 에지 집합 E 를 가지고 있는 컨투어 트리(Contour Tree)는 볼륨 데이터의 입력 스칼라 함수 f 로부터 정의된다. V 는 컨투어의 위상이 변화되는 지점인 함수의 극점으로 구성된다. 극점을 포함하지 않는 연속적인 컨투어의 최대 집합을 컨투어 클래스라 할 때, 에지 집합 E 는 컨투어 클래스가 생성되고 소멸되는 두 극점을 연결하는 에지들로 구성되어 있다. 컨투어 트리가 위와 같이 정의될 때, 에지위의 한 점은 하나의 컨투어에 해당되며 사용자가 그 점을 마우스로 지정하였을 때, 컨투어 트리로부터 씨앗셀(seed cell)이 찾아지고 그로부터 해당되는 컨투어

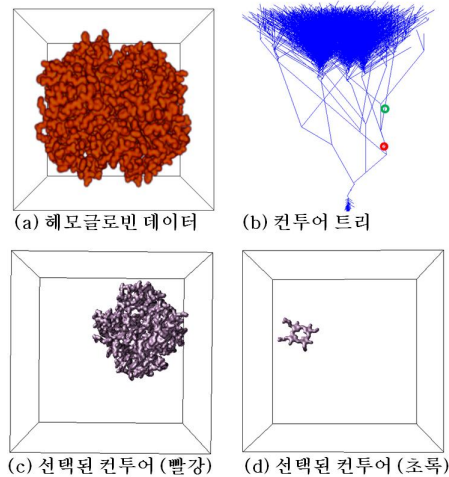


그림 2 헤모글로빈 데이터의 컨투어 트리과 선택된 컨투어 (빨강, 초록)

Fig. 2. Contour Tree and Selected Contours (Red, Green) of Hemoglobin Volume Data

의 표면 메쉬를 계산할수 있다.

우리는 Carr과 Snoeyink의 방법[11]을 이용하여 컨투어 트리를 계산하였다. 이 방식은 우선 병합트리(Join Tree, JT)와 분리트리(Split Tree, ST)를 계산한 후 이를 합하여 컨투어 트리를 생성한다. JT와 ST의 계산 방식은 서로 대칭적이기 때문에 여기서 JT 계산 방식만 대략적으로 소개한다.

우선 볼륨 데이터의 각 그리드 정점에서 함수값들을 크기 순으로 정렬한다. 정렬 후, 최대값으로부터 시작하여 등위값 w 를 점점 감소시키면서 $X(w) = \{x | f(x) > w\} = \{X_1, \dots, X_n\}$ 에 해당하는 영역을 표시한다. w 가 감소함에 따라 상위 객체 집합인 이 영역은 점점 커지게 된다. 이 영역의 연결 컴포넌트들인 X_k 들은 개념적으로 상위 객체에 해당된다. w 가 국소 최대극점(local maximum)을 거치면서 새로운 컴포넌트가 생성되며 그 국소최대극점에 해당되는 벡터는 상위 리프(upper leaf)가 된다. w 가 병합되는 변곡점(saddle point)를 지나치면서 두 개 또는 그 이상 개수의 컴포넌트들이 하나로 합쳐지게 된다. 이 때 새로운 JT 노드가 생성되고 합쳐지기 전의 컴포넌트들이 가지고 있는 가장 최근에 생긴 노드에 연결한다. w 가 최저점(global minimum)에 이르렀을 때, JT 노드가 생성되어 가장 최근의 극점과 연결되어 JT 생성을 위한 계산을 마친다.

실제 구현에서는 도메인 메쉬 각 벡터가 그 벡터에서의 함수값 크기 역순으로 마크된다. 이 과정에서 상위 객체가 생성되기도 하고, 다른 상위 객체와 합쳐지기도 한다. 벡터

스가 마크될때는 그 버텍스가 이웃하는 상위 객체에서 함수값 크기가 가장 작은 버텍스에 연결된다. 이러한 과정의 결과로서 JT가 생성된다.

JT의 상위 리프와 ST의 하위 리프들이 반복적으로 삭제되고 그 리프에 인접한 에지들이 삽입되어 증가된 컨투어 트리(Augmented Contour Tree)가 생성된다. 컨투어 트리는 ACT에서 정규 버텍스들을 삭제함에 의해서 최종적으로 계산된다.

V. 컨투어 추적 알고리즘

우리는 이 장에서 전체 과정의 핵심 부분인 등위면 콤포넌트, 즉 컨투어의 추적 알고리즘을 기술한다. 이 알고리즘은 주어진 입력 함수값들이 시간에 따라 변하면서 이루어지는 컨투어 표면의 변형을 추적한다.

주어진 스칼라 함수는 삼각화 메쉬(simplicial mesh)에서 정의되며 w 를 등위값으로 가정할 때, 메쉬의 노드에서의 부호는 그 노드에서의 함수값이 등위값보다 클 때 "+"로 하고, 그렇지 않을 때 "-"로 한다. 추적 알고리즘은 3장에서 소개된 컨투어의 시간적 일치성 정의에 기반하여 있는데, 알고리즘의 개략적인 아이디어는 다음과 같다.

두 연속적인 시간 단계 t 와 $t+1$ 에서 정의되는 컨투어 C_k^t 와 C_k^{t+1} 는 시간적인 일치성을 가지고 있다고 가정하자. 그러면 우리의 추적 알고리즘은 C_k^t 로부터 추적하여 C_k^{t+1} 을 계산하는 것을 목표로 한다.

〈그림 2〉(c)에서 보듯, 두 C_k^t 와 C_k^{t+1} 의 내부와 외부를 부호인 "+"와 "-"로 표시하면, 컨투어 일치성 정의에 따라, C_k^t 의 경계면은 부호가 바뀌는 부분 즉 "+"에서 "-"로 혹은 "-"에서 "+"로 변하는 부분 쪽으로 이동하게 된다. 이러한 이동을 반복하면 C_k^{t+1} 의 경계를 찾게 된다. 자세한 추적 알고리즘은 다음과 같이 기술된다.

input : C^t , Contour at time t
 output : C^{t+1} deformed from C^t

ContourTracking(C^t)

NewTSet ← ∅ ;

TSet ← get a set of cells that contain C^t ;

Enqueue(each cell in TSet) ;

Mark(each cell in TSet) ;

```

while (queue is not empty) {
    T ← Dequeue() ;
    if (T contains isosurface at timestep t+1)
        NewTSet ← NewTSet ∪ T ;
    CV ← get a set of vertices in T ;
    for each v ∈ CV {
        if v changes sign over time {
            NVSet ← get neighboring cells of v ;
            for each c ∈ NVSet {
                if (isMarked(c)){
                    Mark(c);
                    Enqueue(c);
                }
            }
        }
    }
}
Ct+1 ← triangulate each cell in NewTSet;
    
```

컨투어 추적 알고리즘을 이용하여 우리는 시간 변화 등위면의 위상학적 사건을 분류하고 그 사건들을 나타내는 그래프로 나타낼 수 있다. 등위값 w 가 선택되어 졌을 때, 시간단계 t 에서의 등위면은 $I^t = \{C_1^t, C_2^t, \dots, C_n^t\}$ 으로 나타내어진다. 여기서 C_k^t 는 개별 컨투어를 나타낸다. 다섯가지 위상학적 사건들이 컨투어 일치성에 따라 정의될 수 있다.

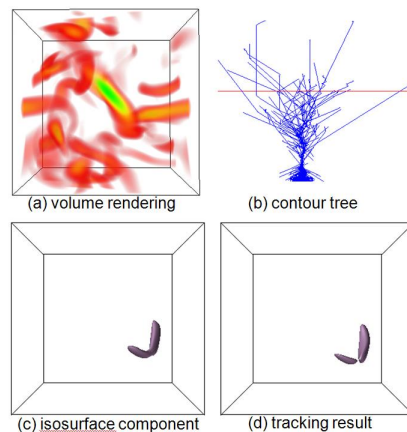


그림 3 회오리(vortex) 데이터 컨투어 추적 예
 Fig. 3. Example of Vortex Data Contour Tracking

- 생성 : C_k^t 로 변형되는 시간단계 $t-1$ 에서의 컨투어가 존재하지 않는다.
- 소멸 : C_k^t 로부터 시간단계 $t+1$ 에 있는 어떠한 컨투어에도 변형되지 않는다.
- 합병 : 시간단계 $t-1$ 에 있는 두 개 이상의 컨투어들이 C_k^t 로 변형된다.
- 분리 : C_k^t 로부터 시간단계 $t+1$ 에 있는 두 개 이상의 컨투어들로 변형된다.
- 유지 : C_k^t 로부터 시간단계 $t+1$ 에 있는 단 하나의 컨투어로 변형된다.

VI. 실험 결과

우리는 제안된 새로운 알고리즘의 성능을 측정하기 위하여 두 시변 볼륨 데이터를 적용하였다. 컨투어 트리와 경로사이의 계산을 선행처리로서 각 시간단계마다 계산하였다. 실험은 2기가바이트 메인메모리를 가진 인텔 펜티엄 듀얼코어 (2.4GHz) 프로세서에서 수행되었다.

〈표 1〉은 각 데이터별로 알고리즘의 수행 속도 및 결과로 얻어지는 컨투어들의 메쉬 크기에 관한 실험 결과를 보여준다. 실험 결과에서 보듯이 컨투어 추적 알고리즘의 수행 속도는 대체로 컨투어 표면의 메쉬 크기에 비례함을 알 수 있으며, 실험한 데이터에 대해서는 한번의 추적이 1초 이내로 수행되기 때문에 상호반응형(interactive) 환경을 제공하고 있다. 본 컨투어 추적 알고리즘은 볼륨 데이터 전체 영역이 아니라 컨투어 영역 근처의 국소적인(local) 영역만을 대상으로 계산이 이루어지므로 빠른 수행 속도를 이룰수 있다.

〈그림 4〉는 회오리(vortex)데이터에서 컨투어 추적의 예를 보여준다. 볼륨 데이터는 여러개의 볼티시티(vorticity) 등위면 콤포넌트를 가지고 있으며, 각 콤포넌트는 데이터의 중요한 특징 부분이 된다. 또한 이 특징 부분의 시간적인 추적을 통해서 동적인 구조와 성질들을 분석할 수 있다.

표 1. 실험 결과. 각 단계에서 추적된 컨투어의 삼각형 개수와 계산에 걸리는 시간을 나타냄.

Table 1. Experimental Results. The number of contour triangles as a result of tracking and computation time.

영상자료	해상도	총시간	$t=1$	$t=2$	$t=3$	$t=4$
회오리	128^3	5	7090 (102ms)	8521 (250ms)	9913 (273ms)	10661 (281ms)
헤모글로빈	128^3	5	292310 (687ms)	291966 (718ms)	291994 (688ms)	291946 (703ms)

기존의 등위면 시각화는 등위면을 구성하는 여러개의 콤포넌트들간에 구분을 못함으로써 한계가 있었다. 이를 보완할수 있는 개별 콤포넌트를 추출하는 것과 시간에 따른 그 변형을 추적하는 것은 시각화의 정도를 향상시킨다 (〈그림 4〉 참조). 더욱이, 개별 콤포넌트를 구분한다면 이것의 표면 면적이나 부피 등을 계산할수 있기 때문에 개개의 등위면 콤포넌트의 변형 과정에 대한 정량적인 정보를 제공할 수 있다. 이는 데이터를 그려보지 않고도 사용자에게 데이터에 대한 이해를 높일 수 있는 정량적인 시각화 방법이 된다.

VII. 결론

본 논문에서는 시간이 변하는 볼륨 이미징 데이터로부터 정의되는 등위면의 콤포넌트를 주어진 시간에 선택하고, 선택된 등위면 콤포넌트의 시간적인 변형을 추적하는 알고리즘을 제시하였다. 기존의 시각화 알고리즘은 대부분 시간이 정지된 볼륨 데이터의 시각화에 초점을 맞춘 반면, 본 알고리즘은 시간이 변하는 데이터에서 기존의 등위면 시각화 알고리즘이 가지는 한계를 넘어서 각 등위면 콤포넌트 단위의 상호 반응적(interactive) 시각화를 가능하게 하여 시각화의 효과를 증대하였다.

Acknowledgement

Prof. Bajaj 와 Prof. Shamir는 본 논문의 기본적인 연구 아이디어를 제공하였음.

참고문헌

- [1] 손봉수, "정육면체형 셀의 분해를 이용한 삼중선형 등위면의 계산," 한국컴퓨터정보학회 논문지, 제 12권 제 2호, 83-91쪽, 2007년. 7월.
- [2] 손봉수, "시변 볼륨 데이터의 압축과 가시화 기법," 한국컴퓨터정보학회 논문지 제 12권, 제 3호, 85-93쪽, 2007년. 9월.
- [3] G. Ji, H.-W. Shen, R. Wenger, "Volume tracking using higher dimensional isocontouring," In Proc. of IEEE Visualization Conference, pages 209-216, 2003.
- [4] R. Samataney, D. Silver, N. Zabusky, J. Cao, "Visualizing features and tracking their evolution," In proc. of IEEE Visualization, pages 159-166, 1998.
- [5] W. E. Lorensen, H. E. Cline, "Marching Cubes: A high resolution 3D surface construction algorithm," ACM SIGGRAPH, pages 163-169, 1987.
- [6] C.L. Bajaj, V. Pascucci, D.R. Schikore, "Fast isocontouring for improved interactivity," Proc. of Symposium on Volume Visualization, pp. 39-46, 1996.
- [7] C.L. Bajaj, V. Pascucci, D.R. Schikore, "The contour spectrum," In IEEE Visualization Conference, pp. 167-173, 1997.
- [8] P.M. Sutton, C. D. Hansen, "Isosurface extraction in time-varying fields using a temporal branch-on-need tree (T-BON)," In Proc. of IEEE Visualization Conference, pages 147-154, 1999.
- [9] L. Kettner, J. Rossignac, J. Snoeyink, "The Safari interface for visualizing time-dependent volume data using iso-surfaces and contour spectra," Computational Geometry : Theory and Applications, 25(1-2) : 97-116, 2003.
- [10] D. Silver, X. Wang, "Tracking and visualization of turbulent 3d features," IEEE Transactions on Visualization and Computer Graphics, 3(2) : 129-141, 1997.
- [11] H. Carr, J. Snoeyink, U. Axen, "Computing contour trees in all dimensions," Computational Geometry: Theory and Applications, 24(2). pp. 75-94, 2003.
- [12] V. Pascucci, K. Cole-McLaughlin, "Efficient computation of the topology of level sets," In proc. of IEEE Visualization, pages 187-194, 2002.
- [13] S. P. Tarasov, M. N. Vyalvi, "Construction of contour trees in 3d in $O(n \log n)$ steps," In proc. of ACM Symposium on Computational Geometry, pages 212-220, 1997.
- [14] M. J. van Kreveld, R. van Oostrum, C. L. Bajaj, V. Pascucci, and D. Schikore, "Contour trees and small seed sets for isosurface traversal," In proc. of ACM Symposium on Computational Geometry, pages 212-220, 1997.
- [15] H. Carr, J. Snoeyink, "Path seeds and flexible isosurfaces using topology for exploratory visualization," In Proc. of VisSym, pages 49-58, 2003.
- [16] B.-S. Sohn, C. L. Bajaj, "Time-Varying Contour Topology," IEEE Transactions on Visualization and Computer Graphics, pages 14-25, 2006.
- [17] A. Mascarenhas, J. Snoeyink, "Isocontour based Visualization of Time-Varying Scalar Fields," Mathematics and Visualization, Mathematical Foundations of Scientific Visualization, Computer Graphics and Massive Data Exploration. pages 1612-3786, 2009.

저 자 소개



손 봉 수

1999년 2월 : 학사(전산과학) 서울대학교

2005년 8월 : 박사(전산과학) 텍사스오스틴대학교

2008~ 현재 : 중앙대학교 컴퓨터공학부 조교수