

디지털 홀로그래피에서의 수정된 골드스타인 알고리즘을 이용한 위상펼침

윤선규 · 조형준 · 김두철 · 유영훈[†]

제주대학교 물리학과
☎ 690-756 제주시 아라 1동 1번지

김성규

한국과학기술연구원 영상미디어센터
☎ 136-791 서울특별시 성북구 하월곡동 39-1번지

(2007년 3월 12일 받음, 2007년 4월 12일 수정본 받음)

디지털 홀로그래피를 이용하여 얻어진 위상정보는 위상펼침 알고리즘을 이용하여 3차원 상을 구현할 수 있다. 위상펼침 방법에는 다양한 알고리즘이 사용되고 있다. 이중 연산 시 사용하는 메모리용량이 작고 연산시간이 짧은 골드스타인 알고리즘이 많이 사용된다. 골드스타인 알고리즘은 쌍을 이루는 특이점이 아닌 다른 특이점이 가까울 때 잘못된 연산을 수행하여 잘못된 3차원 정보를 준다. 이러한 골드스타인 알고리즘의 오류를 특이점과 경계면 데이터를 이용하여 수정하였다. 수정된 알고리즘을 이용하면 분해 한계 영역의 데이터도 복원될 수 있음을 확인하였다. 수정된 알고리즘의 연산시간은 골드스타인 알고리즘보다 약 10%가 증가되지만 다른 알고리즘에 비하면 연산시간이 짧다.

주제어 : Phase unwrapping, Goldstein algorithm, Modified Goldstein algorithm

I 서 론

물리학 및 정밀기계산업 등 응용과학이 발전됨에 따라 시료의 입체적인 상을 관찰할 수 있는 기술이 필요하게 되었다. 그 중 한 가지가 디지털 홀로그래피를 이용한 현미경이다.

디지털 홀로그래피 (digital holography) 기술은 30여 년 전부터 개발되어온 기술이다.^[1-6] 디지털 홀로그래피를 통해 얻어진 기준파와 물체파의 위상차 (ϕ) 정보의 범위는 $-\pi \leq \phi < \pi$ 이다. 하지만, 실제의 물체 위상은 2π 라디안 범위 이상의 값을 가지고 있다. 위상차 정보를 이용하여 실제 물체 위상을 복원하는 방법이 위상펼침(phase unwrapping)이며, 이에 대한 연구가 많이 진행되어왔다.^[7-15] 이중 하드웨어적인 부담이 적고, 연산시간이 가장 짧은 골드스타인(Goldstein) 알고리즘을 많이 사용되고 있다.^[16] 골드스타인 알고리즘은 변화가 적은 감긴 데이터(wrapped data)에는 적당하다. 하지만 감긴 데이터의 변화량이 급격하게 많은 경우에는 골드스타인 알고리즘으로는 정확한 결과를 얻기 어렵다. 다른 알고리즘을 이용하여 복원할 수는 있으나 골드스타인 알고리즘이 가지고 있는 빠른 연산의 장점을 사용할 수 없게 되어, 대상 물체를 실시간으로 관찰하는데 사용하기 어렵다는 문제가 발생하게 된다.

본 연구에서는 골드스타인 알고리즘이 가지고 있는 문제점을 분석하고 이를 해결하기 위한 방법에 대하여 연구하였다.

II 이론적 배경

1. 위상펼침

최소 표본화 비율 만족하는 감긴 위상과 위상펼침 관계는 다음과 같다.^[17]

$$W\{\phi(n)\} = \psi(n) = \phi(n) - 2\pi k \quad (1)$$

여기서 W 는 감긴 연산자(wrapping operator), $\psi(n)$ 는 감긴 위상, $\phi(n)$ 은 실제 위상, k 는 정수이고, n 은 표본화에 따른 상수이다. 감긴 연산자는 항상 $-\pi \leq \psi(n) < \pi$ 이 되게 한다. Itoh의 계산에 의하면 감긴 위상의 차이와 실제 위상의 차이 관계는 (2)식과 같다.^[16]

$$\Delta\{\phi(n)\} = W\{\Delta\{W\{\phi(n)\}\}\} = W\{\Delta\{\psi(n)\}\} \quad (2)$$

$\Delta\{\phi(n)\}$ 는 인접한 실제 위상 데이터의 미분 값이고, $W\{\Delta\{\psi(n)\}\}$ 은 감긴 데이터의 인접한 점 사이의 미분 값이며, 이들 값은 같다. 그러므로 미분 불가능한 점이 없다면, 복원된 위상은 (3)식과 같다.

$$\phi(m) = \phi(0) + \sum_{n=0}^{m-1} W\{\Delta\{\psi(n)\}\} \quad (3)$$

식 (3)은 실제 위상은 감긴 위상차와 적분을 이용하여 복원될 수 있음을 보여준다. 이를 2차원에도 적용할 수 있다. 하

[†] E-mail: yyhyoung@cheju.ac.kr

지만 불연속점이 있는 경우에는 계산 결과가 적분 경로에 의존하기 때문에 경로에 따라 다른 결과가 나타난다. 불연속점이 있는 경우 경로에 무관한 결과를 얻기 위하여 많은 연구가 이루어져 왔다.^[7-15] 참고로 현재 많이 사용되고 있는 위상필름 알고리즘의 종류와 연산 시간을 표 1에 나타내었다. 표 1과 같이 다양한 알고리즘이 사용되고 있지만 계산에 필요한 메모리 용량의 크기가 상대적으로 작고 연산시간이 가장 짧은 골드스타인 알고리즘이 많이 사용되고 있다.

골드스타인 알고리즘에서 특이점을 유수(residue)라 하며, 유수와 유수를 최단거리로 연결하는 선을 갈래자름(branch-cut)이라 한다. 이 연결선을 제외한 영역에 대해서만 적분을 하게 한다.^[16] 이를 전산시능을 한 결과 그림 1과 같다.

그림 1(a)은 물체의 위상이고, (b)는 물체의 감긴 위상이다. (c)는 골드스타인 알고리즘을 이용한 위상필름 결과이다. 위상필름 결과에서 꼭짓점 부분이 완벽히 재현되지 않은 것은

표 1. 각 알고리즘에 따라 1024×1024 픽셀 이미지를 계산하기 위한 메모리 상대 용량과 실행시간 비교

Algorithm	Memory Requirement (1024×1024 array)	Executing Time
Goldstein	3	30 sec
Mask Cut	4	4 min
Flynn	5	10 min
PCG	5	4 min
Multi-Grid	7	20 min

골드스타인 알고리즘에서 갈래자름할때 원래의 경로를 정확히 따라가는 것이 아니라, 최단 거리를 선택하여 연결하기 때문이다. 따라서 갈래자름과 꼭짓점까지의 영역에서는 정확한 값을 나타내지는 못한다. 하지만, 전체적으로 볼 때 물체에 대한 형체에는 큰 영향을 받지 않음을 확인할 수 있기 때문에 큰 문제가 되지 않는다. (d)는 갈래자름 결과로 밝은 점과 검은 점은 양·음의 유수이고 회색 선은 이를 연결하는 갈래자름이다. (e)는 (a)와 (c)에서 나타는 데이터 중 점선으로 표시되어진 72번째 행에 대한 값으로 물체의 위상과 위상필름 결과를 나타내고 있다. 이때 절대적인 값은 같지 않지만, 상대적으로 같음을 확인할 수 있다. 72번째 행에 대한 값을 나타낸 것은 이후 제시되는 결과와 비교하기 위함이다.

2. 골드스타인 알고리즘의 문제점

골드스타인 알고리즘은 갈래자름 과정에서 동일한 불연속점이나 경계면에 의해 발생하는 유수를 연결하게 된다. 하지만 동일한 불연속점이나 경계면에 의한 유수가 다른 변화에 의한 유수보다 멀리 위치하고 있을 경우에는 최소거리에 의하여 다른 불연속면에 의한 유수를 연결하게 되는 것이다. 이것이 골드스타인 알고리즘이 가지고 있는 문제점이다. 간단한 예로 1픽셀을 두고 위치한 다른 물체에 대해서 연산을 수행할 경우 원래의 쌍이 아닌 다른 유수와 연결될 수 있다. 다음 그림 2는 이를 확인한 결과이다.

그림 2(a)는 물체의 위상을 나타내고 있다. 연산에 사용된 물체는 그림 1에서 사용한 데이터에서 가운데 사각형 돌출부를 기준으로 각 변마다 1픽셀의 간격을 두고 얇은 사각형

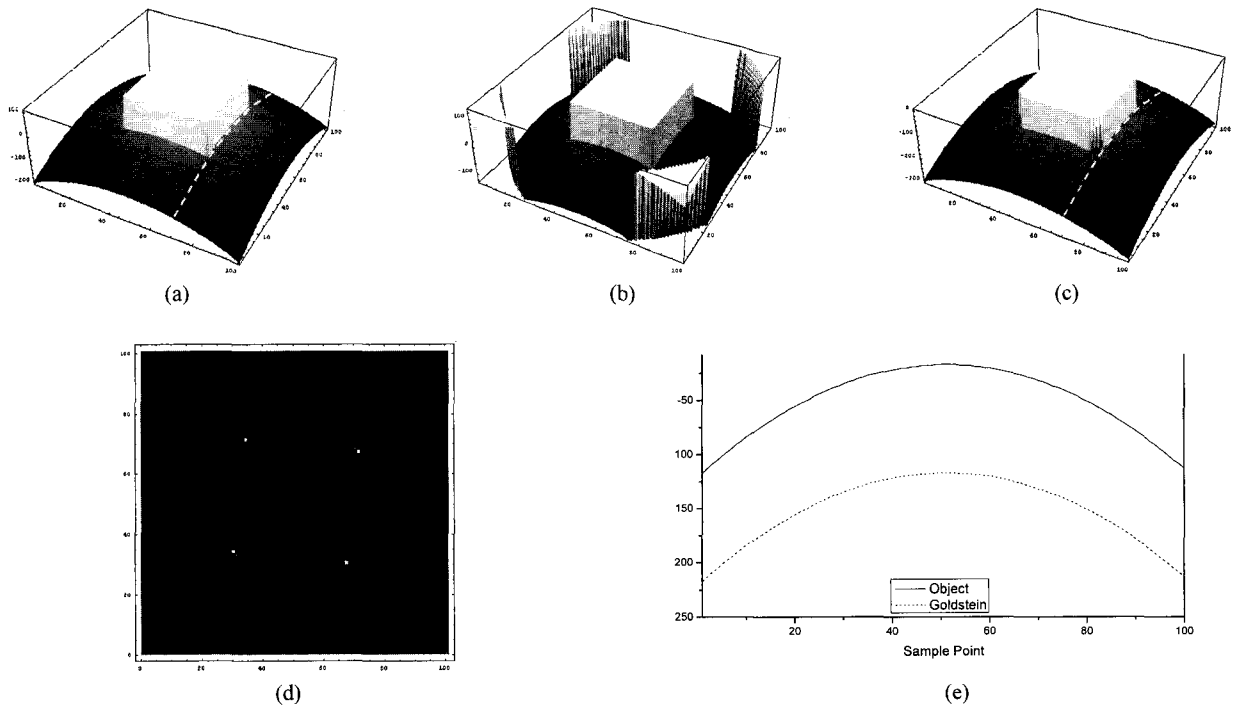


그림 1. 전산시능을 통한 골드스타인 알고리즘 결과: (a) 원본 데이터, (b) 감긴 위상 데이터, (c) 위상필름 데이터, (d) 유수와 갈래자름 데이터, (e) (a)의 점선 부분에서의 물체 데이터와 필름 데이터.

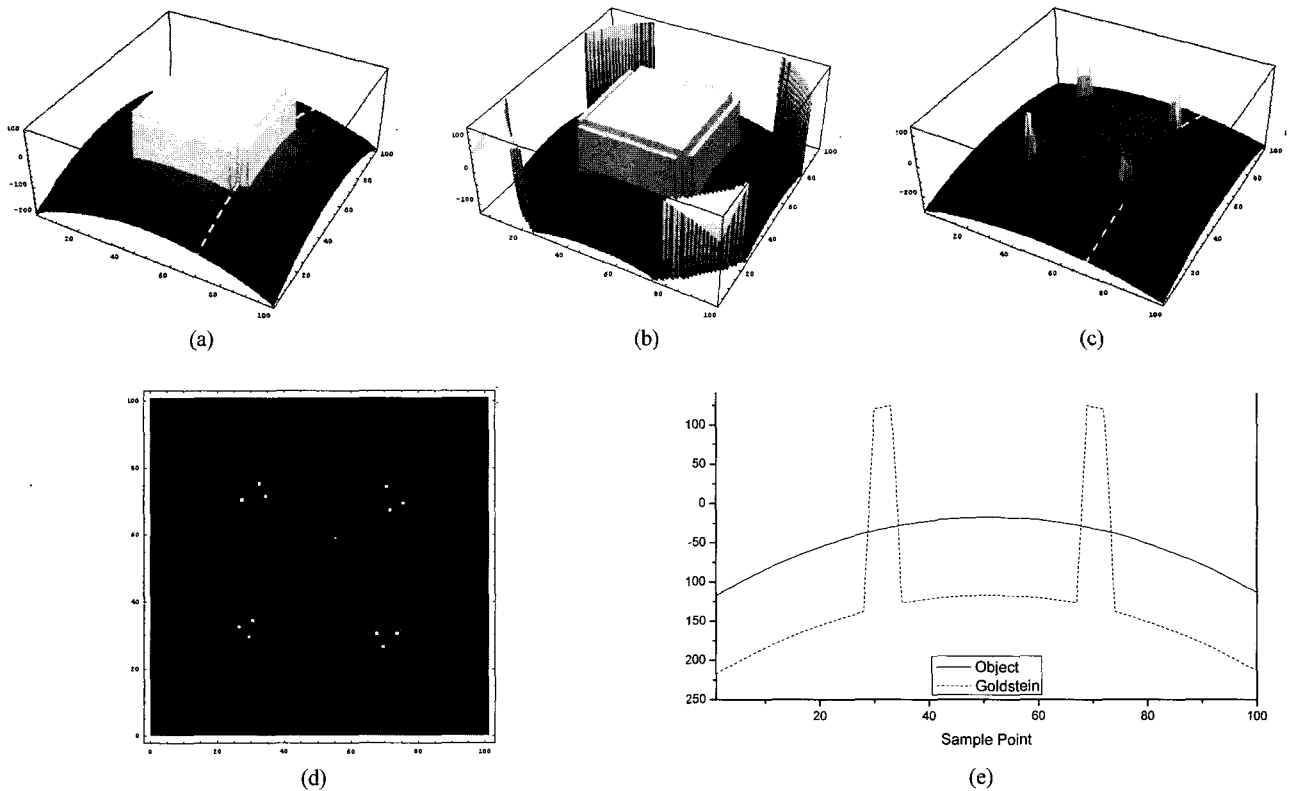


그림 2. 해상도 근처의 구조에서 전산시능을 통한 골드스타인 알고리즘 결과: (a) 원본 데이터, (b) 감긴 위상 데이터, (c) 위상필침 데이터, (d) 유수와 갈래자름 데이터, (e) (a) 점선 부분에서의 물체 데이터와 필침 데이터.

형태의 돌출부를 추가한 것이다. 그림 2(b)는 물체 정보에 대한 감긴 위상 데이터이다. 이를 골드스타인 알고리즘을 사용하여 복원한 것이 그림 2(c)이다. 그림 2(a)와 그림 2(c)를 비교하면 중앙의 사각형 꼭짓점에서 다른 형태의 결과가 나타남을 확인할 수 있다. 이의 원인을 알기위하여 유수점을 비교하였다. 그림 1(d)와 그림 2(d)를 비교하면 공통적으로 나타나는 중앙의 넓은 사각형에 의해 발생되어지는 유수에 대한 갈래자름이 있음을 알 수 있다. 이 사각형의 형태와 위치에는 변화가 없으므로 변해서는 안되는 것이다. 또한 그림 2(e)에서 물체의 정보와 복원된 결과를 비교하기 위해 72번째 행을 비교하였다. 이때 72번째 행에 해당하는 정보는 사각형의 돌출부가 없는 곡면 형태의 정보만이 존재한다. 이 행에 대한 결과를 보면 골드스타인 알고리즘을 적용한 결과 잘못된 연산 결과가 나타남을 확인할 수 있다.

3. 수정된 골드스타인 알고리즘

감긴 연산자 계산에 의해 유수의 값이 “+”이면 양의 유수이고, “-”이면 음의 유수로 나타난다.^[16] 골드스타인 알고리즘에서 잘못된 연산 결과가 나타나는 부분에서 유수의 위치는 반구와 사각형이 만나는 경계이다. 이 중 일부분을 그림 3에 나타내었다.

그림 3(a)의 점선원은 원본 물체의 데이터 영역, 유수 데이터 영역과 경계면 데이터 영역을 비교하기 위한 위치를 나타내고 있다. 그림 3의 (b), (c), (d), (e)에서 숫자는 위상값을

나타내며, 흑색원은 음(-)의 유수, 백색원은 양(+)의 유수이다. 그림 3(b)는 골드스타인 알고리즘에 의한 유수와 갈래자름을 나타내고 있다. 이때 다른 물체에서 발생하는 유수가 연결되고 있음을 확인할 수 있다. 그림 3(c)는 각 픽셀 값에 의한 유수를 나타낸 것으로 양(+)의 유수와 음(-)의 유수가 물체의 경계에 존재함을 확인할 수 있었다. 골드스타인 알고리즘에서는 인접한 유수와 부호가 다르고 두 점 사이의 거리가 1픽셀일 때 두 점을 유수 쌍극자(dipole residue)라 정의하고 이를 유수에서 제외한 후 남은 유수를 이용하여 갈래자름하게 된다. 그러나 그림 (b)에서 중앙에서 보는 바와 같이 다른 물체에 의한 유수가 1픽셀 차이를 두고 존재할 경우 이를 제외시키게 되고, 이로 인하여 잘못된 갈래자름을 하게 된다. 그림 3(d)는 기울기 값을 이용하여 경계면 정보를 함께 나타낸 것이다. 그림에서 보는 바와 같이 유수의 위치와 경계면의 위치가 위상 데이터의 중앙에 위치하기 때문에 인접 픽셀은 대각선 방향으로 계산하였다. 그림 3(e)은 경계면 정보를 이용하여 갈래자름한 결과이다. 이는 무조건 유수 쌍극자를 찾고 나머지 유수에 대하여 갈래자름 하는 것이 아니라, 유수와 경계 정보가 일치할 경우에는 동일한 경계 정보에서 다른 유수를 찾게 하는 것이다. 그리고 경계 정보와 다른 위치에 나타나는 유수는 기존 알고리즘과 동일한 방법으로 갈래자름을 수행한 것이다. 변경되어진 알고리즘의 흐름도는 그림 4와 같다.

그림 4는 기존의 골드스타인 알고리즘과 수정된 골드스타

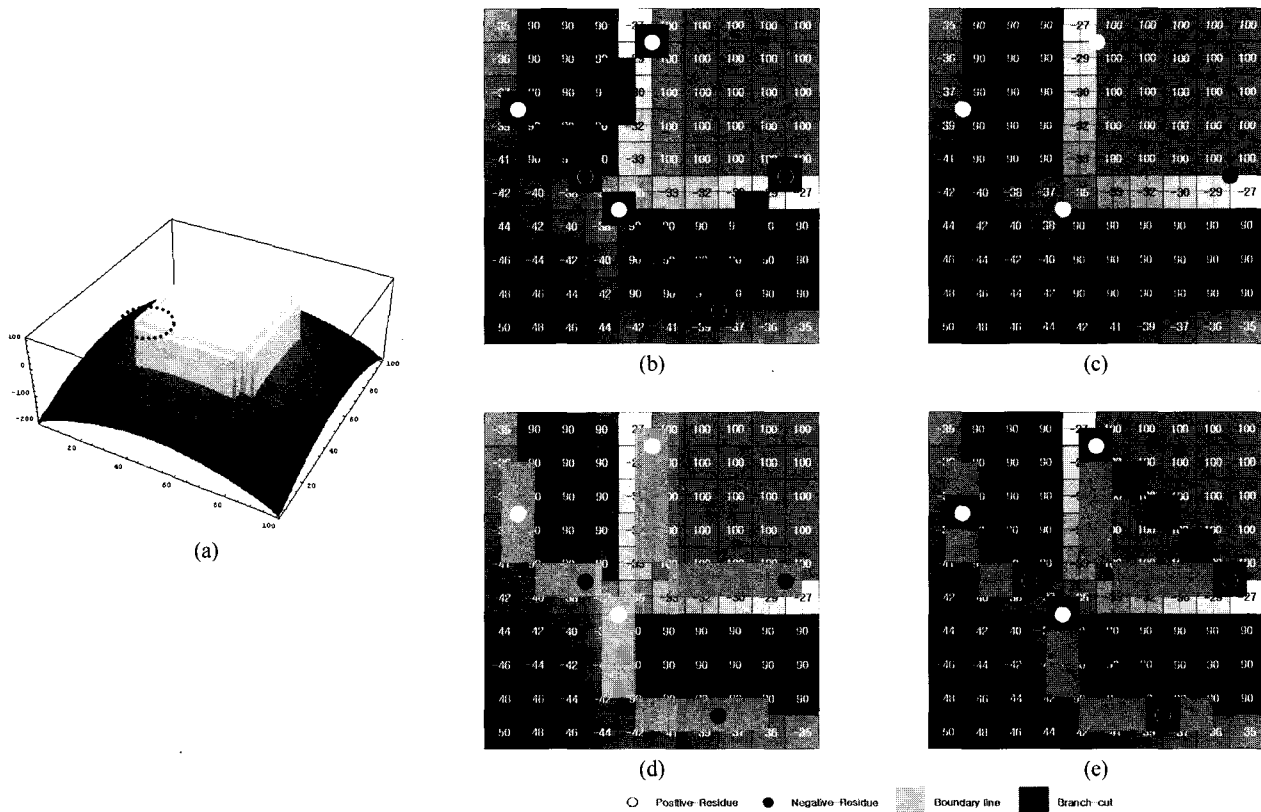


그림 3. 감긴 데이터 영역, 우수 데이터의 영역과 경계면 데이터 영역: (a) 원본 물체에서 데이터 추출 위치, (b) 골드스타인 알고리즘에서의 위상유수와 갈래자름 데이터, (c) 위상 데이터와 우수데이터, (d) 위상, 우수 데이터와 기울기 데이터, (e) 위상, 우수, 기울기와 갈래 자름 데이터.

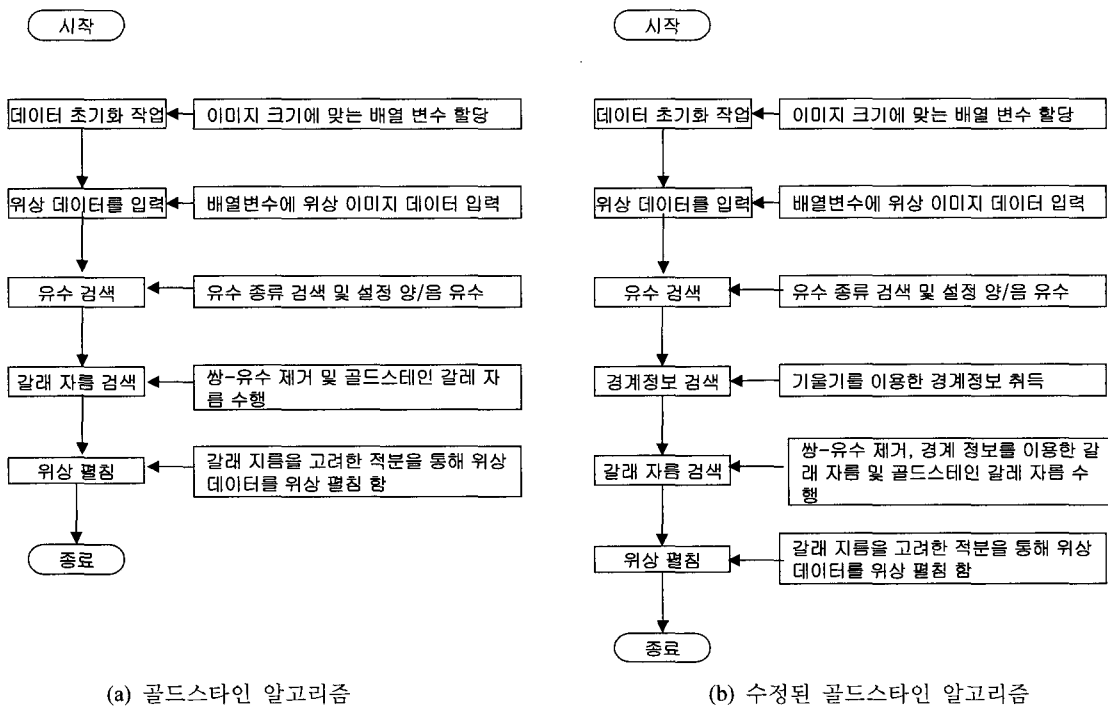


그림 4. 골드스타인 알고리즘과 수정된 골드스타인 알고리즘 흐름도 비교: (a) 골드스타인 알고리즘 흐름도, (b) 수정된 골드스타인 알고리즘 흐름도.

인 알고리즘 흐름도 비교이다. 그림 4(a)는 기존의 골드스타인 알고리즘의 흐름도이고 (b)는 수정된 골드스타인 알고리즘의 흐름도이다. 수정된 알고리즘은 기존의 알고리즘에서 유수 검색을 통해 얻어진 유수정보만을 가지고 갈래자름을 수행하지만, 수정된 알고리즘에서는 유수 검색 후 기울기를 이용한 경계 정보를 검색하고, 이를 이용하여 갈래자름 검색을 한다. 수정된 알고리즘에서 갈래자름은 경계 정보와 유수 정보를 이용하여 유수의 위치가 경계 정보와 동일한 위치에 있을 경우 경계 정보를 이용하여 갈래자름 하며, 그렇지 않을 경우 유수 쌍극자 제거 및 골드스타인 갈래자름을 하는 것이다. 이외의 과정은 기존의 골드스타인 알고리즘과 같다.

그림 5는 수정된 알고리즘의 결과와 기존의 골드스타인 알고리즘 결과를 비교한 것이다. 그림 5(a)와 (b)는 기존의 알고리즘 결과이고, (c)와 (d)는 수정된 알고리즘 결과이다. 그림 5(a)와 (c)를 비교하면 갈래자름의 형태가 변화되었음을 확인할 수 있다. 이는 중앙에 위치한 사각 돌출부에 의한 유수와 주변의 각각의 사각 돌출부에 대한 유수가 구별되어 갈래자름된 결과이다. 위상펼침 결과인 (b)와 (d)를 비교하면, 중앙 사각 돌출부의 꼭지에서 나타나는 잘못된 연산이 수정되었음을 확인할 수 있다. 위상펼침 결과 중 72번째 행에 대한 결과를 비교한 것이 그림 5(e)이다. 골드스타인 알고리즘

에서 29~34, 68~73 픽셀 위치에서 나타나는 오류가 수정된 알고리즘에서는 나타나지 않음을 확인할 수 있고, 물체 데이터의 위상과 비교하여도 차이가 없음을 확인할 수 있다. 즉, 경계면 정보를 이용하면 골드스타인 알고리즘에서 발생 하였던 문제점을 해결할 수 있다.

물체가 2, 3픽셀 간격을 두고 위치한 경우 기존의 골드스타인 알고리즘과 수정된 알고리즘을 이용한 위상펼침 결과는 그림 6과 같다.

그림 6은 골드스타인 알고리즘과 수정된 알고리즘을 비교한 것으로 물체를 2픽셀과 3픽셀 간격을 두고 위치하도록 한 물체에 대한 결과이다. 골드스타인 알고리즘을 이용한 경우에 간격이 1픽셀인 경우에 비하여 잘못된 연산 결과를 나타내는 영역이 줄어들었지만, 여전히 문제가 되는 부분이 존재함을 확인할 수 있다. 하지만 수정된 알고리즘에서는 문제되는 부분이 수정되었음을 확인할 수 있다.

III. 전산시능 결과와 고찰

실제 물체를 디지털 홀로그램을 이용하여 실험하기 전에 그림 7과 같이 CGH(Computer Generated Hologram)로 만들어진 홀로그램을 이용하여 복잡한 형태의 물체에 대한 전산

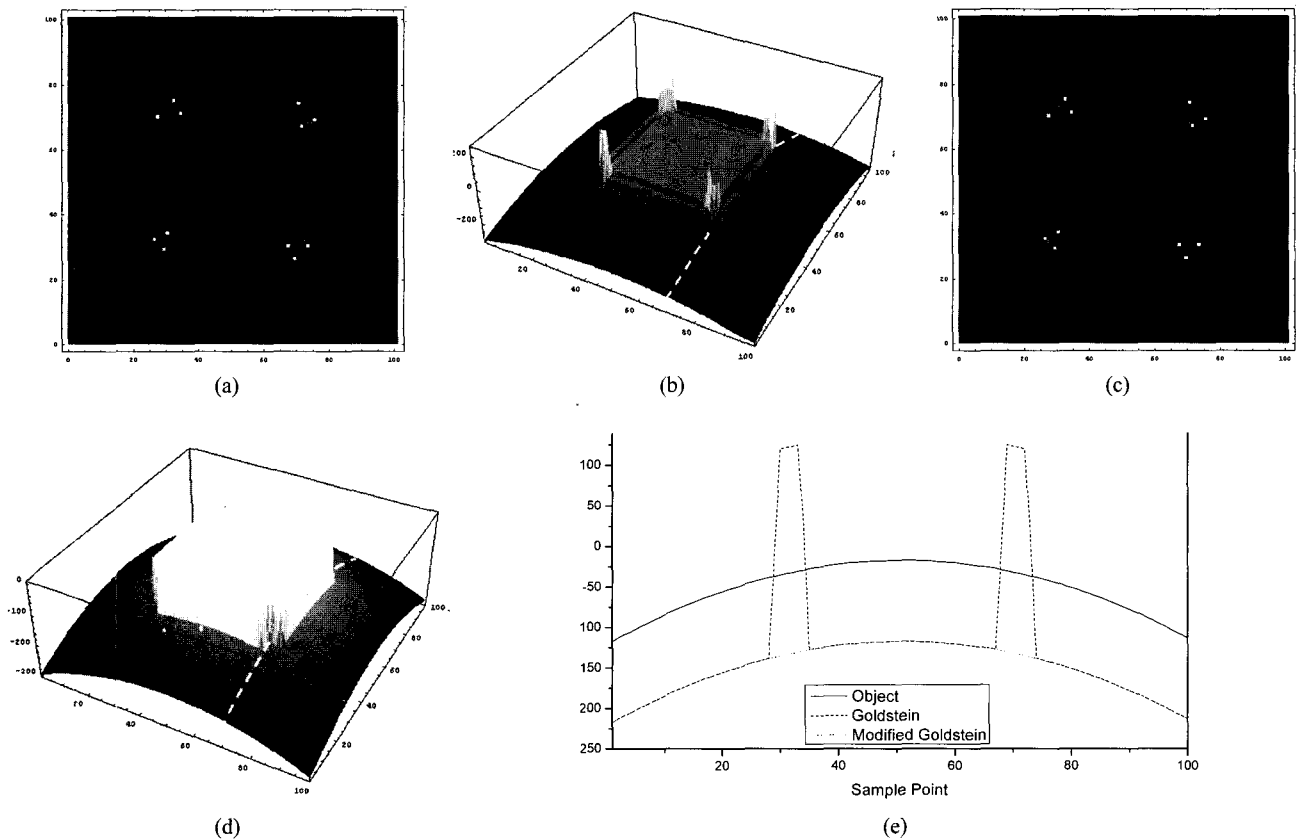


그림 5. 수정된 알고리즘의 전산시능 결과: (a) 골드스타인 알고리즘의 유수와 갈래자름, (b) 골드스타인 알고리즘의 위상펼침 데이터, (c) 수정된 골드스타인 알고리즘의 유수와 갈래자름, (d) 수정된 골드스타인 알고리즘의 위상펼침, (e) 한 단면에서의 물체 데이터와 펼침 데이터.

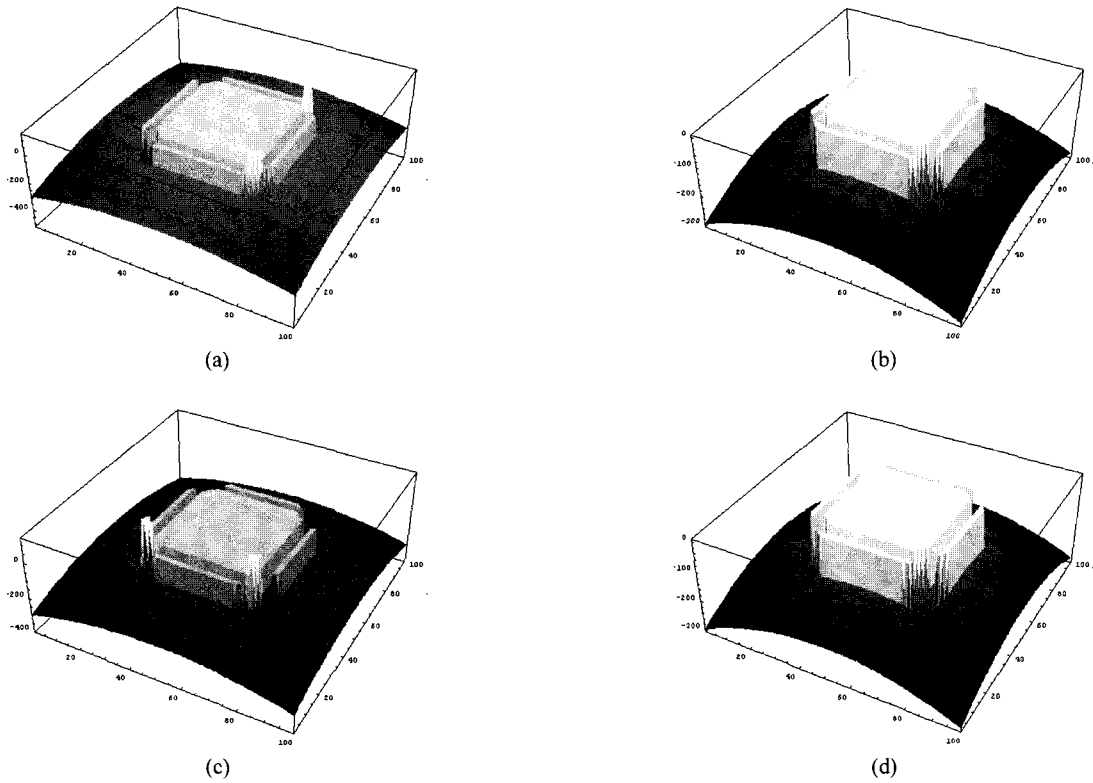


그림 6. 수정된 알고리즘 비교: (a) 2픽셀 떨어진 물체에 대한 골드스타인 알고리즘, (b) 2픽셀 떨어진 물체에 대한 수정된 알고리즘, (c) 3픽셀 떨어진 물체에 대한 골드스타인 알고리즘, (d) 3픽셀 떨어진 물체에 대한 수정된 알고리즘.

기 시늉을 하였다.

그림 7의 데이터를 골드스타인 알고리즘을 이용하여 위상펼침한 결과 점선원 부분을 제외하고는 복원이 잘 되었다. 그림 7의 점선원 안의 데이터를 골드스타인 알고리즘과 수정된 골드스타인 알고리즘을 수행하여 결과를 비교하였다.

그림 8은 골드스타인 알고리즘을 이용하여 위상펼침 한 경우와 수정된 알고리즘을 이용하여 위상펼침 한 결과이다. 그림 8(b)의 결과를 보면 위상변화가 급격하게 일어나는 부분에서 잘못된 결과가 나타나고 있음을 확인하였다. 즉 그림 8(b) 점선원 부분과 같이 잘못된 갈래자름에 의하여 원본 데이터와는 다르게 나타난다. (c)에서는 잘못된 결과가 많이 줄어들었음을 확인할 수 있다. 문제가 되는 1~80번째 데이터에 대한 1차원 그래프를 이용하여 확인하면 그림 9와 같다.

그림 9(a)는 잘못 연산된 범위의 데이터를 나타낸 것이다. 이 결과를 보면 골드스타인 알고리즘에서 잘못된 결과를 나타낸 부분이 수정된 알고리즘에 의해 물체의 정보와 함께 수정되었음을 확인할 수 있다. 그리고 (b)는 골드스타인 알고리즘에서 잘못된 결과가 나타나는 부분에 대한 그래프이다. 골드스타인 알고리즘의 잘못된 결과가 물체의 위상보다 255 또는 -255만큼 차이를 확인할 수 있다. 이는 갈래자름이 잘못되어 적분과정에서 ± 255 이상 차이가 있는 위상 정보영역을 통과하므로 적분과정에서 ± 255 의 차이가 나타날 수 있기 때문이다. 하지만 수정된 알고리즘을 통하여 갈래자름에서 발생하는 문제를 해결하여 올바른 연산을 할 수 있게 되었다.

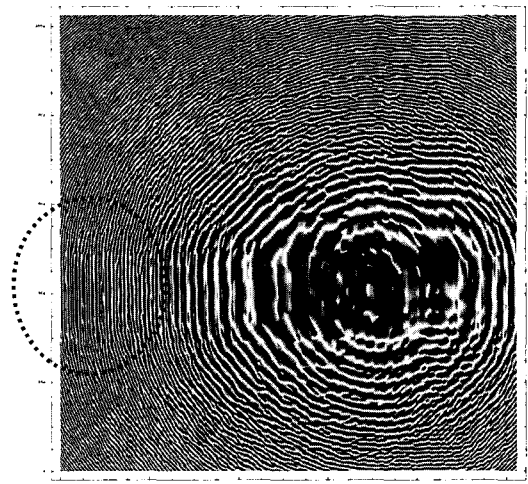


그림 7. 위상정보.

연산을 위하여 사용한 컴퓨터의 CPU는 Intel T2500이거 운영체제는 Windows XP이다. 프로그램은 Mathematica 프로그램을 사용하였다. 이때 기존의 골드스타인 알고리즘의 연산시간은 2144초이고 수정된 알고리즘을 적용하여 계산된 시간은 2387초로 약 10%가 증가하였다. 이는 골드스타인 알고리즘을 사용하는 이유 중 하나인 짧은 연산시간에 큰 영향을 주지 않는다.

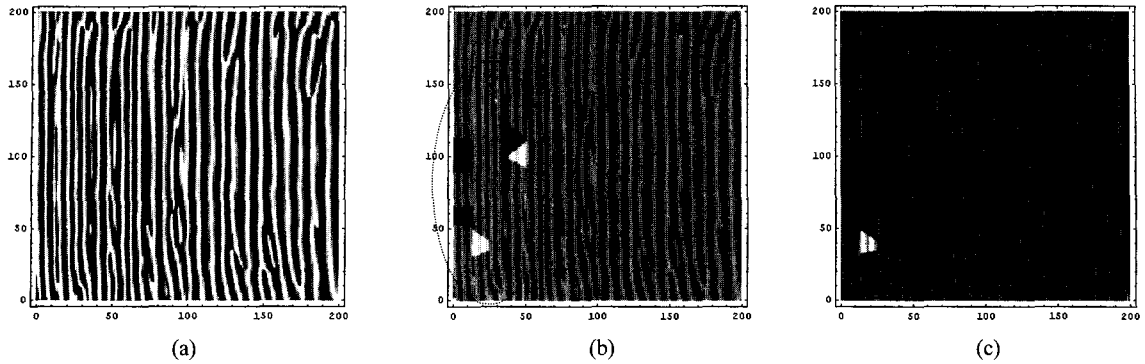


그림 8. 위상펼침 결과: (a) 원본 데이터, (b) 골드스타인 알고리즘, (c) 수정된 골드스타인 알고리즘.

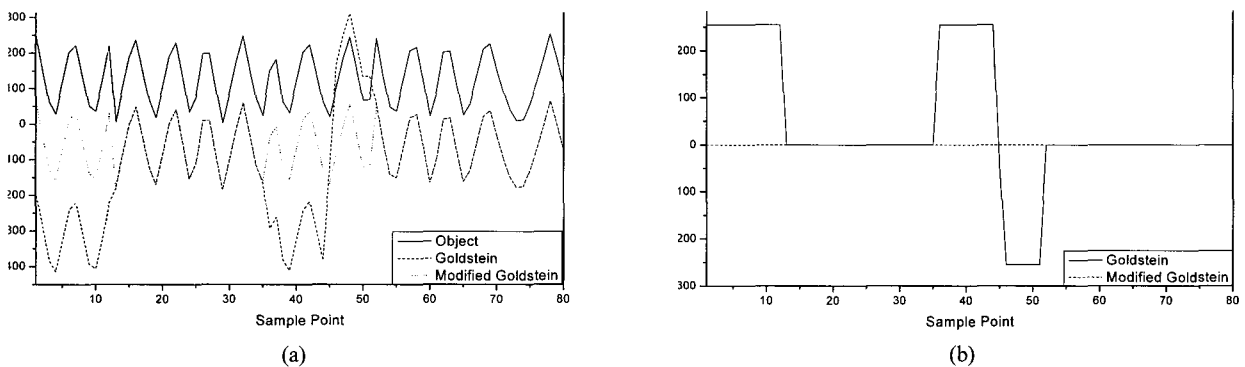


그림 9. 그림 7의 407번째 행의 위상펼침 결과: (a) 물체데이터와 각 알고리즘비교, (b) 수정된 부분에 대한 비교.

IV. 결 론

참고문헌

골드스타인 알고리즘은 유수 쌍극자 보다 다른 인접한 유수가 존재할 때 잘못된 결과를 나타내게 된다. 이는 측정된 위상정보가 CCD의 해상도인 1픽셀을 두고 급격히 변화되는 물체가 존재할 때 나타나는 것이다. 이를 해결하기 위하여 경계면과 유수 사이의 연관성을 확인하였다. 그 결과 경계면에 유수가 위치할 경우 쌍을 이루는 유수 또한 경계면에 위치함을 확인할 수 있었다. 이는 경계면에 존재하는 유수는 골드스타인 알고리즘으로 원래의 쌍을 찾을 수 없음을 의미한다. 이를 이용하여 경계면에 있는 유수는 경계면에서 그 외의 유수는 골드스타인 알고리즘을 이용함으로써 더욱 정확한 갈래자름을 할 수 있다. 따라서 1픽셀을 두고 급격하게 변화하는 물체에서 얻어지는 위상 데이터에 대해서도 경계면을 이용하여 갈래자름을 하면, 잘못된 연산을 수행하는 문제를 해결할 수 있다. 경계면을 사용함으로써 정확한 갈래자름이 가능하게 되어 급격히 변화하는 부분이 많은 물체에 대한 구별이 가능하게 되었다.

감사의 글

이 논문은 산업자원부 차세대기술개발사업 지원으로 수행되었으며, 연구에 참여한 연구자의 일부는 '2단계 BK21 사업'의 지원비를 받았음.

- [1] J. W. Goodman and R. W. Lawrence, "Digital image formation from electronically detected holograms," *Appl. Phys. Lett.*, vol. 11, pp. 77-79, 1967.
- [2] M. A. Kronrod, N. S. Merzlyakov, and L. P. Yaroslavskii, "Reconstruction of a hologram with computer," *Sov. Phys. Tech. Phys.*, vol. 17, pp. 434-444, 1972.
- [3] Haddad W, Cullen D, Solem J, Longworth J, McPherson A, Boyer K, and Rhodes K, "Fourier-transform holographic microscope," *Appl Opt.*, vol. 31, pp. 4973-4978, 1992.
- [4] H. J. Kreuzer and R. A. Pawlitzek, "Numerical Reconstruction for in-line Holography in Reflection and under glancing Incidence," *Proc. 3rd International Workshop on Automatic Processing of Fringe Patterns*. Akademie, Berlin, pp. 364-367, 1997.
- [5] V. Kebbel, H. J. Hartmann, and W. Juptner, "Application of digital holographic microscopy for inspection of micro-optical components," *SPIE*, vol. 4398. pp. 189-198, 2001.
- [6] Coppola G, De Nicola S, Ferraro P, Finizio A, Grilli S, Iodice M, Magro C, and Pierattini G, "Evaluation of residual stress in MEMS structures by means of digital holography," *SPIE*, vol. 4933, pp. 226-31, 2003.
- [7] J. M. Huntley, "Noise-immune phase unwrapping algorithm," *Appl. Opt.*, vol. 28, pp. 3268-3270, 1989.

- [8] T. J. Flynn, "Two-dimensional phase unwrapping with minimum weighted discontinuity," *J. Opt. Soc. Am. A*, vol. 14, No. 10, pp. 2692-2701, 1997.
- [9] G. Fornaro, G. Franceschetti, R. Lanari, E. Sansosti, and M. Tesaro, "Global and local phase unwrapping technique: A comparison," *J. Opt. Soc. Am. A*, vol. 12, pp. 2355-2366, 1997.
- [10] H. O. Saldner and J. M. Huntley, "Profilometry using temporal phase unwrapping and spatial light modulator-based fringe projector," *Opt. Eng.*, vol. 36, pp. 610-615, 1997.
- [11] M. D. Pritt, "Phase unwrapping by means of multigrid techniques for interferometric SAR," *IEEE Trans. Geoscience and Remote Sensing*, vol. 43, pp. 728-738, 1996.
- [12] Howard A. Zebker and Yanping Lu, "Phase unwrapping algorithms for radar interferometry: residue-cut, least-squares, and synthesis algorithms," *J. Opt. Soc. Am. A*, vol. 15, pp. 586-597, 1998.
- [13] Vyacheslav V. Volkov and Yimei Zhu, "Deterministic phase unwrapping in the presence of noise," *Opt. Lett.*, vol. 28, pp. 2156-2158, 2003.
- [14] Wei Xu, "A Region-Growing Algorithm for InSAR Phase Unwrapping," *IEEE Trans. Geoscience and Remote Sensing*, vol. 37, pp. 124-134, 1999.
- [15] Yasuhiro Takaki and Hitoshi Ohzu, "Fast numerical reconstruction technique for high-resolution hybrid holographic microscopy," *Appl. Opt.*, vol. 38, pp. 2204-2211, 1999.
- [16] D. C. Ghiglia and M. D. Pritt, "Two-Dimensional Phase Unwrapping," John Wiley & Sons, Ch3-Ch5, 1998.
- [17] K. Itoh, "Analysis of the phase unwrapping problem," *Appl. Opt.*, vol. 21, pp. 2470-2474, 1982.

Phase Unwrapping using Modified Goldstein Algorithm in Digital Holography

Seon Kyu Yoon, Hyungjun Cho, Doocheol Kim, and Younghun Yu[†]

Dept. of Physics, Cheju National University, Jeju 690-756, Korea

[†]*E-mail: yyhyoung@cheju.ac.kr*

Sung Kyu Kim

Imaging Media Research Center, KIST, Seoul 136-791, Korea

(Received March 12, 2007, Revised manuscript April 12, 2007)

Generally, many kinds of phase unwrapping algorithm are used to obtain three-dimensional features in digital holography. The Goldstein algorithm is a representative method, which requires small memory capacity and short execution time for an unwrapping process. However, the Goldstein algorithm has some problems when the dipole residue is located at the boundary. When the opposite residues are located at the boundary and the distance between the opposite residues is longer than the boundary, an incorrect branch cut occurs and results in incorrect calculation. We have modified the Goldstein algorithm to solve the incorrect calculation problem using boundary information. We found that the modified Goldstein algorithm could resolve the Goldstein algorithm's problem.

OCIS code : 120.2880, 100.6890, 120.5050, 170.6900.