



(19) 대한민국특허청(KR)  
 (12) 등록특허공보(B1)

(45) 공고일자 2018년08월03일  
 (11) 등록번호 10-1885356  
 (24) 등록일자 2018년07월30일

- (51) 국제특허분류(Int. Cl.)  
*G01C 15/00* (2006.01) *G01C 19/00* (2006.01)  
*G01D 21/02* (2006.01) *G01S 19/38* (2010.01)  
*G06T 7/70* (2017.01) *H04M 1/725* (2006.01)

- (52) CPC특허분류  
*G01C 15/00* (2013.01)  
*G01C 19/00* (2013.01)
- (21) 출원번호 10-2017-0053865  
 (22) 출원일자 2017년04월26일  
 심사청구일자 2017년04월26일  
 (65) 공개번호 10-2017-0125716  
 (43) 공개일자 2017년11월15일

(30) 우선권주장  
 1020160055515 2016년05월04일 대한민국(KR)

- (56) 선행기술조사문헌

JP2008089314 A\*  
 KR1020150115335 A  
 KR101463906 B1  
 KR101578158 B1

\*는 심사관에 의하여 인용된 문헌

전체 청구항 수 : 총 12 항

심사관 : 홍정훈

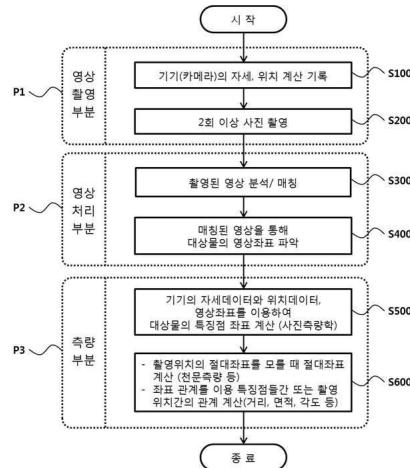
- (54) 발명의 명칭 대상물의 위치 정보 측정 장치 그리고 그 제어 방법

**(57) 요약**

본 발명은 대상물의 위치 정보 측정장치에 관한 것으로서, 위치 정보 측정 장치의 가속도를 감지하는 가속도 센서, 위치 정보 측정 장치의 각속도를 감지하는 각속도 센서, 상기 위치 정보 측정 장치의 자기장의 크기를 감지하는 자기장 센서, 대상물에 대한 복수 번의 촬영 동작에 따라 복수 개의 영상 데이터를 획득하는 촬영부, 그리

(뒷면에 계속)

**대 표 도** - 도3



고 연산부를 포함한다. 연산부는 가속도 센서와 자기장 센서에서 출력되는 가속도 데이터와 자기장 데이터를 이용하여 복수의 촬영 동작이 각각 이루어질 때 위치 정보 측정 장치의 자세 데이터와 위치 데이터를 산출하고, 복수 번의 촬영 동작 시 획득된 복수의 영상 데이터를 매칭하는 매칭 기준을 획득하고, 획득된 복수의 영상 데이터 중 하나인 기준 영상 데이터에서 특징점 가능 영역을 선택하여 특징점 가능 영역에서 특징점을 검출하고, 검출된 특징점을 나머지 영상 데이터에 매칭하기 위한 나머지 영상 데이터에 특징점 기술자를 적용하여 검출된 특징점을 나머지 영상 데이터에서 찾아 검출된 상기 특징점을 나머지 영상 데이터에 매칭시켜, 복수의 영상 데이터에서 특징점의 영상 좌표를 획득하고, 각 촬영 시의 상기 자세 데이터, 위치 데이터, 그리고 상기 영상 좌표를 이용하여 대상물의 특징점 좌표를 계산하며, 각 촬영시 상기 위치 정보 측정 장치의 절대좌표를 파악하고, 각 촬영 시의 상기 위치 정보 측정장치의 절대좌표와 대상물의 특징점 좌표를 이용하여 대상물의 특징점에 대한 위치 정보를 산출하도록 한다.

## (52) CPC특허분류

*G01D 21/02* (2013.01)

*G01S 19/38* (2013.01)

*G06T 7/70* (2017.01)

*H04M 1/725* (2013.01)

*H04M 2250/12* (2013.01)

---

공지예외적용 : 있음

## 명세서

### 청구범위

#### 청구항 1

대상물의 위치 정보 측정 장치에서,

상기 위치 정보 측정 장치의 가속도를 감지하여 해당 상태의 감지 신호를 출력하는 가속도 센서,

상기 위치 정보 측정 장치의 각속도를 감지하여 해당 상태의 감지 신호를 출력하는 각속도 센서,

상기 위치 정보 측정 장치의 자기장의 크기를 감지하여 해당 상태의 감지 신호를 출력하는 자기장 센서,

대상물에 대한 복수 번의 촬영 동작에 따라 복수 개의 영상 데이터를 획득하여 출력하는 촬영부, 그리고

상기 가속도 센서, 상기 각속도 센서 및 상기 촬영부에 연결되어 있는 연산부

를 포함하고,

상기 연산부는,

상기 가속도 센서와 자기장 센서에서 출력되는 가속도 데이터와 자기장 데이터를 이용하여 복수의 촬영 동작이 각각 이루어질 때 상기 위치 정보 측정 장치의 자세 데이터와 위치 데이터를 산출하고,

상기 복수 번의 촬영 동작 시 상기 촬영부로부터 획득된 복수의 영상 데이터를 매칭하는 매칭 기준을 획득하고, 획득된 복수의 영상 데이터 중 하나인 기준 영상 데이터에서 특징점 가능 영역을 선택하여 특징점 가능 영역에서 특징점을 검출하고, 검출된 특징점을 나머지 영상 데이터에 매칭하기 위한 상기 나머지 영상 데이터에 특징점 기술자를 적용하여 검출된 상기 특징점을 상기 나머지 영상 데이터에서 찾아 검출된 상기 특징점을 상기 나머지 영상 데이터에 매칭시켜, 상기 복수의 영상 데이터에서 상기 특징점의 영상 좌표를 획득하고,

상기 각 촬영 시의 상기 자세 데이터, 위치 데이터, 그리고 상기 영상 좌표를 이용하여 상기 대상물의 특징점 좌표를 계산하며, 상기 각 촬영 시의 상기 위치 정보 측정 장치의 절대좌표를 파악하고,

상기 각 촬영 시의 상기 위치 정보 측정 장치의 상기 절대좌표와 상기 대상물의 특징점 좌표를 이용하여 대상물의 특징점에 대한 위치 정보를 산출하도록 하되,

상기 연산부는 상기 가속도 데이터와 상기 자기장 데이터를 이용하여 상기 위치 정보 측정 장치의 가속도 예비 자세 데이터를 산출하고, 상기 각속도 데이터를 이용하여 상기 위치 정보 측정 장치의 자이로 예비 자세 데이터를 산출하며, 상기 촬영부에 의해 복수의 촬영 동작이 이루어질 때 상기 자이로 예비 자세 데이터를 측정 시간으로 적분하여 적분된 자이로 예비 자세 데이터를 산출하고, 상기 가속도 예비 자세 데이터와 적분된 자이로 예비 자세 데이터를 합성하여 상기 위치 정보 측정 장치의 자세 데이터를 산출하고, 상기 자세 데이터와 상기 가속도 데이터를 이용하여 상기 위치 정보 측정 장치의 위치 데이터를 산출하는

대상물의 위치 정보 측정 장치.

#### 청구항 2

삭제

#### 청구항 3

제1항에서,

상기 연산부는 각 영상 데이터를 이용하여 RGB채널별로 적분 영상 데이터로 생성하고, 각 적분 영상 데이터를 정해진 크기를 갖는 일정 영역으로 분할하여 모자이크 영상 데이터를 생성하고, 생성된 모자이크 영상 데이터를 매칭시켜 RGB 유사도 평균이 가장 높은 펙셀 이동점을 영상 데이터의 매칭 기준을 정하여 영상 데이터 매칭 기준을 획득하는 대상물의 위치 정보 측정 장치.

#### 청구항 4

제3항에서,

상기 연산부는,

사용자 입력부를 통해 입력된 데이터를 이용하여 사용자가 선택된 지점의 좌표를 판정하고 판정된 지점의 좌표에 설정된 픽셀 크기를 가감하여 특징점 분석을 위한 특징점 가능 영역을 정하거나,

상기 모자이크 영상 데이터 중 정해진 개수의 픽셀에 대한 픽셀 데이터를 기준으로 하여 RGB 유사도를 계산해 특징점 가능 영역을 판정하여,

상기 특징점 분석을 위한 영역이나 상기 특징점 가능 영역에서 특징점을 검출하는 대상물의 위치 정보 측정 장치.

#### 청구항 5

제1항에서,

상기 연산부는,

상기 각 촬영 시 상기 특징점 좌표는 상기 영상 좌표를 사진좌표로 변경하고 상기 위치 정보 측정장치의 자세 데이터에 포함된 회전각을 이용하여 상기 사진좌표를 회전 변환하고,

상기 회전 변환된 상기 사진좌표와 상기 촬영부의 초점거리를 이용하여 상기 특징점까지의 미소 방위각과 미소 고저각을 계산하고,

상기 미소 방위각, 상기 미소 고저각 및 촬영 동작 시 상기 위치 정보 측정장치의 자세 데이터에 포함된 방위각과 고저각을 이용하여 촬영위치에서부터 상기 특징점까지의 방위각과 고저각을 산출하며,

각 촬영 시 촬영 위치에 대한 위치 데이터와 산출된 상기 특징점까지의 방위각과 고저각을 이용하여 각 촬영 시 상기 특징점 좌표를 구하거나, 상기 각 촬영 시의 위치 데이터, 자세 데이터 및 상기 사진 좌표를 이용하여 공선조건식을 행렬식으로 이용하여 각 촬영 시 상기 특징점 좌표를 구하는

대상물의 위치 정보 측정 장치.

#### 청구항 6

제1항에서,

상기 연산부에 연결되어 있고 상기 위치 정보 측정 장치의 위도와 경도를 출력하는 위치정보 획득부를 더 포함하고

상기 연산부는 상기 위치정보 획득부로부터 인가되는 상기 위도와 경도를 이용하여 상기 위치 정보 측정장치의 절대좌표를 산출하는

대상물의 위치 정보 측정 장치.

#### 청구항 7

제1항에서,

상기 연산부는 상기 대상물의 적어도 하나의 절대좌표를 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하거나 태양, 달, 북극성 또는 별자리의 고저각과 방위각을 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하는 위치 정보 측정 장치.

#### 청구항 8

연산부는 가속도 센서로부터 인가되는 감지 신호를 이용하여 위치 정보 측정 장치의 가속도에 대한 가속도 데이터를 판정하는 단계,

상기 연산부는 각속도 센서로부터 인가되는 감지 신호를 이용하여 상기 위치 정보 측정 장치의 각속도를 판정하

는 단계,

상기 연산부는 자기장 센서로부터 인가되는 감지 신호를 이용하여 상기 위치 정보 측정 장치의 자기장의 크기를 판정하는 단계,

상기 연산부는 촬영부로부터 인가되는 영상 데이터를 이용하여 대상물에 대한 복수 번의 촬영 동작에 따라 복수 개의 영상 데이터를 획득하는 단계,

상기 연산부는 상기 가속도 데이터와 상기 자기장 데이터를 이용하여 복수의 촬영 동작이 각각 이루어질 때 상기 위치 정보 측정 장치의 자세 데이터와 위치 데이터를 산출하는 단계,

상기 복수 번의 촬영 동작 시 상기 촬영부로부터 획득된 복수의 영상 데이터를 매칭하는 매칭 기준을 획득하는 단계,

획득된 복수의 영상 데이터 중 하나인 기준 영상 데이터에서 특징점 가능 영역을 선택하여 특징점 가능 영역에서 특징점을 검출하는 단계,

검출된 특징점을 나머지 영상 데이터에 매칭하기 위한 상기 나머지 영상 데이터에 특징점 기술자를 적용하여 검출된 상기 특징점을 상기 나머지 영상 데이터에서 찾아 검출된 상기 특징점을 상기 나머지 영상 데이터에 매칭 시켜, 상기 복수의 영상 데이터에서 상기 특징점의 영상 좌표를 획득하는 단계, 그리고

상기 각 촬영 시의 상기 자세 데이터, 위치 데이터, 그리고 상기 영상 좌표를 이용하여 상기 대상물의 특징점 좌표를 계산하며 상기 각 촬영시 상기 위치 정보 측정 장치의 절대좌표를 파악하여, 상기 각 촬영 시의 상기 위치 정보 측정장치의 상기 절대좌표와 상기 대상물의 특징점 좌표를 이용하여 대상물의 특징점에 대한 위치 정보를 산출하도록 하는 단계

를 포함하되,

상기 자세 데이터와 위치 데이터의 산출 단계는,

상기 가속도 데이터와 상기 자기장 데이터를 이용하여 상기 위치 정보 측정 장치의 가속도 예비 자세 데이터를 산출하는 단계,

상기 각속도 데이터를 이용하여 상기 위치 정보 측정 장치의 자이로 예비 자세 데이터를 산출하는 단계,

상기 촬영부에 의해 복수의 촬영 동작이 이루어질 때상기 자이로 예비 자세 데이터를 측정 시간으로 적분하여 적분된 자이로 예비 자세 데이터를 산출하고, 상기 가속도 예비 자세 데이터와 적분된 자이로 예비 자세 데이터를 합성하여 상기 위치 정보 측정 장치의 자세 데이터를 산출하는 단계, 그리고

상기 자세 데이터와 상기 가속도 데이터를 이용하여 상기 위치 정보 측정 장치의 위치 데이터를 산출하는 단계

를 포함하는

대상물의 위치 정보 측정 장치의 제어 방법.

## 청구항 9

삭제

## 청구항 10

제8항에서,

상기 매칭 기준의 획득 단계는,

각 영상 데이터를 이용하여 RGB채널별로 적분 영상 데이터로 생성하는 단계,

각 적분 영상 데이터를 정해진 크기를 갖는 일정 영역으로 분할하여 모자이크 영상 데이터를 생성하는 단계, 그리고

생성된 모자이크 영상 데이터를 매칭시켜 RGB 유사도 평균이 가장 높은 팩셀 이동점을 영상 데이터의 매칭 기준을 정하여 영상 데이터 매칭 기준을 획득하는 단계

를 포함하는 대상물의 위치 정보 측정 장치의 제어 방법.

### 청구항 11

제10항에서,

상기 특징점 검출 단계는,

사용자 입력부를 통해 입력된 데이터를 이용하여 사용자가 선택된 지점의 좌표를 판정하고 판정된 지점의 좌표에 설정된 픽셀 크기를 가감하여 특징점 분석을 위한 영역을 특징점 가능 영역을 정하거나, 상기 모자이크 영상 데이터 중 정해진 개수의 픽셀에 대한 픽셀 데이터를 기준으로 하여 RGB 유사도를 계산해 특징점 가능 영역을 판정하는 단계, 그리고

상기 특징점 분석을 위한 영역이나 상기 특징점 가능 영역에서 특징점을 검출하는 단계

를 포함하는 대상물의 위치 정보 측정 장치의 제어 방법.

### 청구항 12

제8항에서,

상기 특징점의 영상 좌표 획득 단계는,

상기 각 촬영 시 상기 특징점 좌표는 상기 영상 좌표를 사진좌표로 변경하고 상기 위치 정보 측정장치의 자세 데이터에 포함된 회전각을 이용하여 상기 사진좌표를 회전 변환하는 단계,

상기 회전 변환된 상기 사진좌표와 상기 촬영부의 초점거리를 이용하여 상기 특징점까지의 미소 방위각과 미소 고저각을 계산하는 단계,

상기 미소 방위각, 상기 미소 고저각 및 촬영 동작 시 상기 위치 정보 측정장치의 자세 데이터에 포함된 방위각과 고저각을 이용하여 촬영위치에서부터 상기 특징점까지의 방위각과 고저각을 산출하는 단계, 그리고

각 촬영 시 촬영 위치에 대한 위치 데이터와 산출된 상기 특징점까지의 방위각과 고저각을 이용하여 각 촬영 시 상기 특징점 좌표를 구하거나, 상기 각 촬영 시의 위치 데이터, 자세 데이터 및 상기 사진 좌표를 이용하여 공선조건식을 행렬식으로 이용하여 각 촬영 시 상기 특징점 좌표를 구하는 단계

를 포함하는 대상물의 위치 정보 측정 장치의 제어 방법.

### 청구항 13

제8항에서,

상기 연산부는 위치정보 획득부로부터 인가되는 상기 위치 정보 측정 장치의 위도와 경도를 이용하여 상기 위치 정보 측정장치의 절대좌표를 산출하는 대상물의 위치 정보 측정 장치의 제어 방법.

### 청구항 14

제8항에서,

상기 연산부는 상기 대상물의 적어도 하나의 절대좌표를 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하거나 태양, 달, 북극성 또는 별자리의 고저각과 방위각을 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하는 위치 정보 측정 장치의 제어 방법.

## 발명의 설명

### 기술 분야

[0001]

본 발명은 대상물의 위치 정보 측정 장치 그리고 그 제어 방법에 관한 것이다.

### 배경 기술

[0002]

일반인이 원하는 대상물 간의 거리를 측량하기 위해서는 측정자의 눈으로 짐작하여 측정하는 목측 방법이나 측정자의 발걸음 수를 계산하여 측정하는 보측 방법을 이용하여 대략적으로 거리를 측정하는 것이 일반적이다.

- [0003] 또한, 스마트폰(smart phone)과 같은 피사체 촬영 기능을 구비한 스마트 기기를 이용하여 거리를 측정하는 경우, 원하는 대상물의 사진 촬영 동작이나 거리 계산 동작 전에 해당 스마트 기기의 촬영 높이를 입력하거나 거리 측정이 기준이 되는 대상물을 촬영해야 한다.
- [0004] 따라서, 기준 대상물의 촬영 후, 스마트 기기를 통해 해당 대상물 사진의 길이를 입력하여, 단순한 촬영 영상의 크기 비교를 통하여 스마트 기기와 원하는 대상물 간의 거리나 크기를 계산하는 방법이 있다.
- [0005] 또한, 스마트 기기의 초점 길이를 이용하여 초점이 맞는 영역에 대한 개략적인 거리 계산을 통하여 원하는 대상물 간의 거리 측정을 하였다.
- [0006] 하지만, 이런 스마트 기기를 이용하여 측거 동작을 실시할 경우, 일반적으로 사용자에게 추가적인 정보를 입력받기 위해 추가 입력 동작이 요구되는 경우, 사용자는 많은 불편함과 부담감을 느끼게 되어 측거 동작을 위한 스마트 기기의 사용을 자제하게 된다.
- [0007] 또한, 사용자에 의해 추가적으로 입력되는 입력값의 정확도가 낮은 경우, 입력값을 이용하여 산출된 거리 측량 값 역시 정확도가 높지 않아 사용자를 만족시키지 못하며, 단순한 사진의 크기 비교나 스마트 기기의 초점거리만으로 거리를 측량하는 방법은 역시 정확도가 높지 않아 산출 결과의 신뢰도가 떨어지는 문제가 있다.
- [0008] 이러한 문제로 인해, 원하는 대상물 간의 정확한 거리나 해당 대상물의 좌표를 측정하기 위해서는 고가의 측량 기기를 구비해야 하므로, 사용자에게 경제적인 부담을 초래하게 된다.

## 발명의 내용

### 해결하려는 과제

- [0009] 따라서 본 발명이 이루고자 하는 기술적 과제는 별도의 측량기기 없이 촬영 기능을 구비한 스마트 기기를 이용하여 간단하고 정확하게 원하는 대상물 간의 거리와 원하는 대상물의 원하는 지점의 좌표 측정이 이루어지도록 하는 것이다.
- [0010] 본 발명이 이루고자 하는 다른 기술적 과제는 별도의 측량기기 없이 원하는 대상물 간의 거리 측정과 대상물의 원하는 지점의 좌표 측정이 이루어지므로, 사용자의 만족도가 향상된다.

### 과제의 해결 수단

- [0011] 본 발명의 한 특징에 따른 대상물의 위치 정보 측정 장치는 상기 위치 정보 측정 장치의 가속도를 감지하여 해당 상태의 감지 신호를 출력하는 가속도 센서, 상기 위치 정보 측정 장치의 각속도를 감지하여 해당 상태의 감지 신호를 출력하는 각속도 센서, 상기 위치 정보 측정 장치의 자기장의 크기를 감지하여 해당 상태의 감지 신호를 출력하는 자기장 센서, 대상물에 대한 복수 번의 촬영 동작에 따라 복수 개의 영상 데이터를 획득하여 출력하는 촬영부, 그리고 상기 가속도 센서, 상기 각속도 센서 및 상기 촬영부에 연결되어 있는 연산부를 포함한다.
- [0012] 상기 연산부는 상기 가속도 센서와 자기장 센서에서 출력되는 가속도 데이터와 자기장 데이터를 이용하여 복수의 촬영 동작이 각각 이루어질 때 상기 위치 정보 측정 장치의 자세 데이터와 위치 데이터를 산출하고, 상기 복수 번의 촬영 동작 시 상기 촬영부로부터 획득된 복수의 영상 데이터를 매칭하는 매칭 기준을 획득하고, 획득된 복수의 영상 데이터 중 하나인 기준 영상 데이터에서 특징점 가능 영역을 선택하여 특징점 가능 영역에서 특징점을 검출하고, 검출된 특징점을 나머지 영상 데이터에 매칭하기 위한 상기 나머지 영상 데이터에 특징점 기술자를 적용하여 검출된 상기 특징점을 상기 나머지 영상 데이터에서 찾아 검출된 상기 특징점을 상기 나머지 영상 데이터에 매칭시켜, 상기 복수의 영상 데이터에서 상기 특징점의 영상 좌표를 획득하고, 상기 각 촬영 시의 상기 자세 데이터, 위치 데이터, 그리고 상기 영상 좌표를 이용하여 상기 대상물의 특징점 좌표를 계산하며, 상기 각 촬영 시 상기 위치 정보 측정 장치의 절대좌표를 파악하고, 상기 각 촬영 시의 상기 위치 정보 측정 장치의 상기 절대좌표와 상기 대상물의 특징점 좌표를 이용하여 대상물의 특징점에 대한 위치 정보를 산출하도록 한다.
- [0013] 상기 연산부는 상기 가속도 데이터와 상기 자기장 데이터를 이용하여 상기 위치 정보 측정 장치의 가속도 예비 자세 데이터를 산출하고, 상기 각속도 데이터를 이용하여 상기 위치 정보 측정 장치의 자이로 예비 자세 데이터를 산출하며, 상기 촬영부에 의해 복수의 촬영 동작이 이루어질 때 상기 자이로 예비 자세 데이터를 측정 시간으로 적분하여 적분된 자이로 예비 자세 데이터를 산출하고, 상기 가속도 예비 자세 데이터와 적분된 자이로 예

비 자세 데이터를 합성하여 상기 위치 정보 측정 장치의 자세 데이터를 산출하고, 상기 최종 자세 데이터와 상기 가속도 데이터를 이용하여 상기 위치 정보 측정 장치의 위치 데이터를 산출하는 것이 좋다.

[0014] 상기 연산부는 각 영상 데이터를 이용하여 RGB채널별로 적분 영상 데이터로 생성하고, 각 적분 영상 데이터를 정해진 크기를 갖는 일정 영역으로 분할하여 모자이크 영상 데이터를 생성하고, 생성된 모자이크 영상 데이터를 매칭시켜 RGB 유사도 평균이 가장 높은 팩셀 이동점을 영상 데이터의 매칭 기준을 정하여 영상 데이터 매칭 기준을 획득할 수 있다.

[0015] 상기 연산부는 사용자 입력부를 통해 입력된 데이터를 이용하여 사용자가 선택된 지점의 좌표를 판정하고 판정된 지점의 좌표에 설정된 팩셀 크기를 가감하여 특징점 분석을 위한 특징점 가능 영역을 정하거나, 상기 모자이크 영상 데이터 중 정해진 개수의 팩셀에 대한 팩셀 데이터를 기준으로 하여 RGB 유사도를 계산해 특징점 가능 영역을 판정하여, 상기 특징점 분석을 위한 영역이나 상기 특징점 가능 영역에서 특징점을 검출하는 것이 바람직하다.

[0016] 상기 연산부는 상기 각 촬영 시 상기 특징점 좌표는 상기 영상 좌표를 사진좌표로 변경하고 상기 위치 측정장치의 자세 데이터에 포함된 회전각을 이용하여 상기 사진좌표를 회전 변환하고, 상기 회전 변환된 상기 사진좌표와 상기 촬영부의 초점거리를 이용하여 상기 특징점까지의 미소 방위각과 미소 고저각을 계산하고, 상기 미소 방위각, 상기 미소 고저각 및 촬영 동작 시 상기 위치 정보 측정장치의 자세 데이터에 포함된 방위각과 고저각을 이용하여 촬영위치에서부터 상기 특징점까지의 방위각과 고저각을 산출하며, 각 촬영 시 촬영 위치에 대한 위치 데이터와 산출된 상기 특징점까지의 방위각과 고저각을 이용하여 각 촬영 시 상기 특징점 좌표를 구하거나, 상기 각 촬영 시의 위치 데이터, 자세 데이터 및 상기 사진 좌표를 이용하여 공선조건식을 행렬식으로 이용하여 각 촬영 시 상기 특징점 좌표를 구할 수 있다.

[0017] 상기 특징에 따른 대상물의 위치 정보 측정 장치는 상기 연산부에 연결되어 있고 상기 위치 정보 측정 장치의 위도와 경도를 출력하는 위치정보 획득부를 더 포함할 수 있고, 상기 연산부는 상기 위치정보 획득부로부터 인가되는 상기 위도와 경도를 이용하여 상기 위치 정보 측정장치의 절대좌표를 산출할 수 있다.

[0018] 상기 연산부는 상기 대상물의 적어도 하나의 절대좌표를 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하거나 태양, 달, 북극성 또는 별자리의 고저각과 방위각을 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출할 수 있다.

[0019] 본 발명의 다른 특징에 따른 대상물의 위치 정보 측정 방법은 연산부는 가속도 센서로부터 인가되는 감지 신호를 이용하여 위치 정보 측정 장치의 가속도에 대한 가속도 데이터를 판정하는 단계, 상기 연산부는 각속도 센서로부터 인가되는 감지 신호를 이용하여 상기 위치 정보 측정 장치의 각속도를 판정하는 단계, 상기 연산부는 자기장 센서로부터 인가되는 감지 신호를 이용하여 상기 위치 정보 측정 장치의 자기장의 크기를 판정하는 단계, 상기 연산부는 촬영부로부터 인가되는 영상 데이터를 이용하여 대상물에 대한 복수 번의 촬영 동작에 따라 복수 개의 영상 데이터를 획득하는 단계, 상기 연산부는 상기 가속도 데이터와 상기 자기장 데이터를 이용하여 복수의 촬영 동작이 각각 이루어질 때 상기 위치 정보 측정 장치의 자세 데이터와 위치 데이터를 산출하는 단계, 상기 복수 번의 촬영 동작 시 상기 촬영부로부터 획득된 복수의 영상 데이터를 매칭하는 매칭 기준을 획득하는 단계, 획득된 복수의 영상 데이터 중 하나인 기준 영상 데이터에서 특징점 가능 영역을 선택하여 특징점 가능 영역에서 특징점을 검출하는 단계, 검출된 특징점을 나머지 영상 데이터에 매칭하기 위한 상기 나머지 영상 데이터에 특징점 기술자를 적용하여 검출된 상기 특징점을 상기 나머지 영상 데이터에서 찾아 검출된 상기 특징점을 상기 나머지 영상 데이터에 매칭시켜, 상기 복수의 영상 데이터에서 상기 특징점의 영상 좌표를 획득하는 단계, 그리고 상기 각 촬영 시의 상기 자세 데이터, 위치 데이터, 그리고 상기 영상 좌표를 이용하여 상기 대상물의 특징점 좌표를 계산하여 상기 각 촬영 시 상기 위치 정보 측정 장치의 절대좌표를 파악하여, 상기 각 촬영 시의 상기 위치 정보 측정장치의 상기 절대좌표와 상기 대상물의 특징점 좌표를 이용하여 대상물의 특징점에 대한 위치 정보를 산출하도록 하는 단계를 포함한다.

[0020] 상기 자세 데이터와 위치 데이터의 산출 단계는 상기 가속도 데이터와 상기 자기장 데이터를 이용하여 상기 위치 정보 측정 장치의 가속도 예비 자세 데이터를 산출하는 단계, 상기 각속도 데이터를 이용하여 상기 위치 정보 측정 장치의 자이로 예비 자세 데이터를 산출하는 단계, 상기 촬영부에 의해 복수의 촬영 동작이 이루어질 때 상기 자이로 예비 자세 데이터를 측정 시간으로 적분하여 적분된 자이로 예비 자세 데이터를 산출하고, 상기 가속도 예비 자세 데이터와 적분된 자이로 예비 자세 데이터를 합성하여 상기 위치 정보 측정 장치의 자세 데이터를 산출하는 단계, 그리고 상기 자세 데이터와 상기 가속도 데이터를 이용하여 상기 위치 정보 측정 장치의 위치 데이터를 산출하는 단계를 포함할 수 있다.

[0021] 상기 매칭 기준의 획득 단계는 각 영상 데이터를 이용하여 RGB채널별로 적분 영상 데이터로 생성하는 단계, 각 적분 영상 데이터를 정해진 크기를 갖는 일정 영역으로 분할하여 모자이크 영상 데이터를 생성하는 단계, 그리고 생성된 모자이크 영상 데이터를 매칭시켜 RGB 유사도 평균이 가장 높은 팩셀 이동점을 영상 데이터의 매칭 기준을 정하여 영상 데이터 매칭 기준을 획득하는 단계를 포함할 수 있다.

[0022] 상기 특징점 검출 단계는 사용자 입력부를 통해 입력된 데이터를 이용하여 사용자가 선택된 지점의 좌표를 판정하고 판정된 지점의 좌표에 설정된 팩셀 크기를 가감하여 특징점 분석을 위한 영역을 특징점 가능 영역을 정하거나, 상기 모자이크 영상 데이터 중 정해진 개수의 팩셀에 대한 팩셀 데이터를 기준으로 하여 RGB 유사도를 계산해 특징점 가능 영역을 판정하는 단계, 그리고 상기 특징점 분석을 위한 영역이나 상기 특징점 가능 영역에서 특징점을 검출하는 단계를 포함할 수 있다.

[0023] 상기 특징점의 영상 좌표 획득 단계는 상기 각 촬영 시 상기 특징점 좌표는 상기 영상 좌표를 사진좌표로 변경하고 상기 위치 정보 측정장치의 자세 데이터에 포함된 회전각을 이용하여 상기 사진좌표를 회전 변환하는 단계, 상기 회전 변환된 상기 사진좌표와 상기 촬영부의 초점거리를 이용하여 상기 특징점까지의 미소 방위각과 미소 고저각을 계산하는 단계, 상기 미소 방위각, 상기 미소 고저각 및 촬영 동작 시 상기 위치 정보 측정장치의 자세 데이터에 포함된 방위각과 고저각을 이용하여 촬영위치에서부터 상기 특징점까지의 방위각과 고저각을 산출하는 단계, 그리고 각 촬영 시 촬영 위치에 대한 위치 데이터와 산출된 상기 특징점까지의 방위각과 고저각을 이용하여 각 촬영 시 상기 특징점 좌표를 구하거나, 상기 각 촬영 시의 위치 데이터, 자세 데이터 및 상기 사진 좌표를 이용하여 공선조건식을 행렬식으로 이용하여 각 촬영 시 상기 특징점 좌표를 구하는 단계를 포함할 수 있다.

[0024] 상기 연산부는 위치정보 획득부로부터 인가되는 상기 위치 정보 측정 장치의 위도와 경도를 이용하여 상기 위치 정보 측정장치의 절대좌표를 산출할 수 있다.

[0025] 상기 연산부는 상기 대상물의 적어도 하나의 절대좌표를 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출하거나 태양, 달, 북극성 또는 별자리의 고저각과 방위각을 이용하여 상기 위치 정보 측정 장치의 절대좌표를 산출할 수 있다.

### 발명의 효과

[0026] 이러한 특징에 따르면, 별도의 측량 기기를 이용하지 않고 휴대용 사용자 단말인 위치 정보 측정 장치를 이용하여 원하는 대상물에 대한 위치 정보를 용이하게 정확하게 측정할 수 있어, 사용자의 편리성이 향상된다.

[0027] 또한, 해당 대상물의 3차원 좌표를 획득할 수 있으므로, 어떠한 건물이나 물체의 컴퓨터 모델링을 위한 자료 획득이 용이하게 이루어진다.

### 도면의 간단한 설명

[0028] 도 1은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 시스템의 블록도이다.

도 2는 본 발명의 한 실시예에 따른 좌표 파악을 위한 개념도이다.

도 3은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치의 개략적인 동작 순서도이다.

도 4a 및 도 4b는 본 발명의 한 실시예에 따른 글로벌좌표계와 기기좌표계의 변환을 나타내는 그림이다.

도 5는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자세와 상대 위치를 계산하는 동작 순서도이다.

도 6은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상촬영부분의 실시간 처리를 위한 동작 순서도이다.

도 7은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상촬영부분의 오차를 보정을 위한 동작의 순서도이다.

도 8a 및 도 8b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치의 영상촬영부분에서 사용자 제어화면이다.

도 9는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치의 영상처리부분에 대한 개략적인 동작 순서

도이다.

도 10a 및 도 10b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 매칭 기준의 개념도이다.

도 11은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 RGB유사도에 대한 개념도이다.

도 12는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 적분영상에 대한 개념도이다.

도 13은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 매칭 기준 계산을 위한 동작 순서도이다.

도 14는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 모자이크 영상의 매칭 알고리즘에 대한 의사코드이다.

도 15는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 처리 과정을 보여주는 예이다.

도 16은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 매칭 이동 기준을 계산하기 위한 개념도이다.

도 17은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 매칭 이동 기준을 계산하기 위해 두 영상의 매칭을 변화시키는 것에 대한 개념도이다.

도 18은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상 데이터 매칭 이동 기준 계산을 하여 최적의 이동기준을 계산한 모습을 보여주는 개념도이다.

도 19a 내지 도 19c는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상처리부분의 사용자 제어 화면을 통해 사용자의 입력을 받는 개념도이다.

도 20은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상처리부분 알고리즘의 의사코드이다.

도 21a 및 도 21b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자동으로 특징점 가능영역을 검출해 내는 개념도이다.

도 22는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자동으로 특징점 가능영역을 검출할 때 4방향 비교를 하여 특징점 가능영역을 제거하는 개념도이다.

도 23은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자동으로 특징점 가능영역을 검출할 때 8방향 비교를 통해 특징점 가능영역을 제거하는 개념도이다.

도 24는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자동으로 특징점 가능영역을 찾아주는 알고리즘의 의사코드이다.

도 25a 및 도 25b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 자동으로 특징점 가능영역을 찾아주는 알고리즘을 적용하여 특징점 분석을 실시한 예이다.

도 26은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 해리스코너 알고리즘을 이용하여 입력 영상에 대하여 특징 가능성 값을 계산하는 의사코드이다.

도 27은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 해리스 코너 알고리즘을 사용해서 특징 점을 검출하는 동작의 순서도이다.

도 28은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 RGB색상 값의 역수를 가중치로 연산한 후 비최대억제를 시행하는 의사코드이다.

도 29는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 특징점 검출 알고리즘을 통한 특징점 검출의 한 예이다.

도 30a 및 도 30b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 특징점 기술자 마스크의 한 예이다.

도 31은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 특징점 매칭을 위한 동작 순서도이다.

도 32는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 특징점 매칭 알고리즘을 통한 특징점 매칭의 한 예를 보여주는 개념도이다.

도 33은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 취득한 자료를 가지고 해당 대상물의 특징점 좌표를 계산하는 동작의 순서도이다.

도 34a 및 도 34b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 카메라의 렌즈에 의해 발생하는 방사왜곡과 접선왜곡의 개념도이다.

도 35a 내지 도 35c는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상좌표와 사진좌표에 대한 개념도이다.

도 36a 내지 도 36c는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 회전각을 이용하여 사진좌표를 회전 변환하는 개념도이다.

도 37은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 카메라의 초점거리와 보정된 사진좌표를 이용하여 촬영위치로부터 대상물까지의 미소방위각과 미소고저각을 구하는 개념도이다.

도 38은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 공간상의 두 점의 위치를 알고 그 두 점에서 바라보는 대상물의 방위각과 고저각을 알고 있을 때 해당 대상물의 특징점 좌표를 취득하기 위한 개념도이다.

도 39a 및 도 39b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 대상물 한 점의 절대좌표를 알 때 촬영위치의 절대좌표를 계산하는 개념도이다.

도 40은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 대상물의 한 점을 알 때 촬영위치의 절대좌표를 계산하는 동작의 순서도이다.

도 41a 및 도 41b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 태양을 촬영하여 태양의 고도각과 방위각을 계산하는 개념도이다.

도 42는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 촬영위치에서 태양의 고저각과 태양의 방위각을 이용하여 촬영위치의 절대좌표를 결정하는 개념도이다.

도 43은 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 태양을 촬영하여 촬영위치의 절대좌표를 계산하는 동작 순서도이다.

도 44a 내지 도 44c는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상처리부분의 사용자 입력화면에서 촬영위치와 대상물간의 위치관계를 표현해주는 한 예와 각 대상물간을 연결해주는 사용자 입력화면에 대한 개념도이다.

도 45a 및 도 45b는 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 장치에서 영상처리부분의 사용자 입력화면에서 사용자가 각 대상을 선택함으로서 촬영위치와 대상물의 위치관계나 대상물들 간의 위치관계를 파악하는 개념도이다.

### 발명을 실시하기 위한 구체적인 내용

[0029]

아래에서는 첨부한 도면을 참고로 하여 본 발명의 실시예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0030]

어떤 구성요소가 다른 구성요소에 "접속되어" 있다거나 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 접속되어 있거나 연결되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 한다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 접속되어" 있다거나 "직접 연결되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0031]

그러면 첨부한 도면을 참고로 하여 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 시스템 및 그 제어 방

법에 대하여 설명한다.

- [0032] 먼저, 도면을 참고로 하여 본 발명의 한 실시예에 따른 대상물의 위치 정보 측정 시스템(이하, 대상물의 위치 정보 측정 시스템을 '위치 정보 측정 시스템'이라 함)에 대하여 상세히 설명한다.
- [0033] 도 1에 도시한 것처럼, 위치 정보 측정 시스템은 원하는 대상물의 위치 정보 측정장치(이하, 대상물의 위치 정보 측정장치를 '위치 정보 측정장치'라 함)(1000), 그리고 위치 정보 측정장치(1000)와 통신하는 서버(2000)를 구비한다.
- [0034] 본 예에서, 위치 정보 측정장치(1000)는 촬영 기능을 구비하고 있고 사용자의 휴대가 가능한 휴대 단말기로서, 스마트폰(smart phone)과 같은 스마트 기기일 수 있다.
- [0035] 본 예에서, 대상물의 위치 정보는 대상물의 해당 지점(예, 특징점)에 대한 위치 정보로서, 절대좌표와 상대좌표 중 적어도 하나와 같이 대상물의 해당지점을 명확히 할 수 있는 값뿐만이 아니라, 해당 대상물에서 어느 지점과 측정자 간의 거리, 해당 대상물에서 어느 지점의 좌표(즉 절대좌표나 상대좌표)와 지점간의 거리, 해당 대상물에서 두 지점 간의 수직거리(즉, 높이나 높이차)와 수평거리 및 두 지점간의 경사도 중 적어도 하나를 포함하며, 또한, 위치 정보는 해당 대상물의 어느 부분에 대한 면적과 해당 대상물의 면에 대한 경사방향과 경사각 중 적어도 하나를 포함할 수 있다.
- [0036] 특징점이란 영상이나 사진과 같은 이미지 상에서 다른 곳과 두드러지게 다른 정보를 가진 점으로, 예를 들어, 물체의 모서리와 같이 주변부와 비교하여 볼 때 그 명암이나 색상값들이(예를 들어 RGB값) 급격하게 변하는 점을 뜻한다.
- [0037] 절대 좌표 또는 글로벌 좌표란 예를 들어 경위도좌표계나 평면직교좌표계와 같이 지구 내에서의 절대적인 위치 관계를 파악하기 위한 절대 좌표계 또는 글로벌 좌표계에서, 경위도 좌표계에서의 위도와 경도, 평면직교좌표계에서의 X, Y 및 Z 값과 같이 절대 좌표계나 글로벌 좌표계에서 그 위치를 명확히 할 수 있는 좌표값을 뜻한다.
- [0038] 상대 좌표란 절대 좌표계 내에서 어떤 위치를 절대 좌표 값으로 표현하지 않고, 글로벌 좌표계의 기준 축을 중심으로 어느 한 기준 위치에서 다른 위치까지의 거리 관계나 좌표 관계로 표현한 것으로, 예를 들어 평면직교좌표계에서 어떤 특정 위치 B의 위치가 A의 위치를 기준했을 때 X축으로 a만큼, Y축이 b만큼, Z축이 c만큼 이동한 위치라면 B의 상대좌표는 (a, b, c)가 되고, 기준이었던 A의 상대좌표는 (0, 0, 0)이 된다.
- [0039] 평면직교좌표계란 지구내의 위치관계를 명확히 하기 위한 글로벌 좌표계의 한 종류로서, 타원체상의 위치정보(예를 들어 경도, 위도)는 지도투영을 통하여 평면좌표로 나타낼 수 있다. 지도 투영의 원점을 기준으로 하여, 원점을 지나는 자오선을 X축(N: 북쪽을 +), 동서방향을 Y축(E: 동쪽을 +), 중력면에 수직인 높이 방향을 Z축으로 하는 좌표계를 뜻한다(도 2 참조).
- [0040] 예를 들어, WGS84 경위도 좌표계(EPSG:4326)로 북위 36.306200°, 동경 127.198766° 및 높이 114.733m의 한 점은 대한민국 중부원점을 중심으로 한 평면직교좌표계(Korea2000/Central Belt 2010 EPSG:5186)로 표현하면 (423136.046754, 217829.112793, 114.733)의 3차원 좌표로 표현될 수 있다.
- [0041] 이러한 위치 정보 측정장치(1000)는 감지부(100), 촬영부(110), 위치정보 획득부(120), 사용자 입력부(130), 감지부(100), 촬영부(110), 위치정보 획득부(120), 사용자 입력부(130)에 연결되어 있는 연산 및 제어 유닛(200), 연산 및 제어 유닛(200)에 연결되어 있는 출력부(300), 연산 및 제어 유닛(200)에 연결되어 있는 저장부(400), 그리고 연산 및 제어 유닛(200)에 연결되어 있는 통신부(500)를 구비한다.
- [0042] 감지부(100)는 위치 정보 측정 장치(1000)의 자세 데이터와 위치 데이터를 산출하기 위해 위치 정보 측정 장치(1000)의 물리량 예를 들어 가속도, 각속도 및 자기장의 세기 등을 측정하기 위한 것이다.
- [0043] 위치 데이터는 위치 정보 측정장치(1000)의 글로벌 좌표계 혹은 상대좌표계에서의 측정장치의 절대위치 혹은 상대위치를 나타내기 위한 절대좌표 또는 상대좌표를 뜻한다.
- [0044] 자세 데이터는 위치 정보 측정장치(1000)의 글로벌 좌표계에서 각 축 방향별 회전각을 뜻하며, 방위각, 고저각 및 회전각으로 이루어져 있다.
- [0045] 이러한 감지부(100)는 가속도 센서(101), 각속도 센서인 자이로 센서(102) 및 자기장 센서(103)를 구비한다.
- [0046] 가속도 센서(101)는 위치 정보 측정 장치(1000)의 기기 좌표계에 대한 각 축 방향 별로 가속도(즉, 축방향 가속도)를 측정하여 해당 상태의 가속도 데이터를 연산 및 제어 유닛(200)로 출력한다. 이러한 가속도 센서(101)는

스마트폰에 설치된 센서 중에서 JAVA의 센서 호출명이 'Sensor.TYPE\_ACCELEROMETER'와 'Sensor.TYPE\_LINEAR\_ACCELERATION'인 센서일 수 있다.

[0047] 차이로 센서(102)는 기기 좌표계에 대한 각 축 방향 별로 각속도를 측정하는 센서로서, 측정된 각속도에 해당하는 상태의 각속도 데이터를 연산 및 제어 유닛(200)으로 출력한다. 이러한 차이로 센서(102)는 스마트폰에 설치된 센서 중에서 JAVA센서 호출명이 'Sensor.TYPE\_GYROSCOPE'인 센서일 수 있다.

[0048] 자기장 센서(103)는 기기 좌표계에 대한 각 축 방향 별로 자기장의 세기를 측정하여 해당 크기의 자기장 데이터를 연산 및 제어 유닛(200)으로 출력한다.

[0049] 이러한 자기장 센서(103)는, 예를 들어, 스마트폰에 설치된 센서 중에서 JAVA센서 호출명이 'Sensor.TYPE\_MAGNETIC\_FIELD'인 센서일 수 있다.

[0050] 본 예에서, 가속도 센서(101), 차이로 센서(102) 및 자기장 센서(103)의 동작에 의해 출력되는 가속도 데이터, 각속도 데이터 및 자기장 데이터를 포함하는 데이터를 감지 데이터(DAT1)로 명명하고, 이 감지 데이터(DAT1)는 연산 및 제어 유닛(200)의 연산부(210)로 인가된다. 본 예에서, 감지부(100)의 데이터 취득방법은 일정시간간격으로 센서(101~102)에 수집되는 결과 값을 측정하여 연산 및 제어 유닛(200)으로 출력한다.

[0051] 예를 들어, 안드로이드 스마트폰의 경우 센서에서 읽어오는 시간 간격을 조정할 수 있는 데, 기종마다 차이가 있지만 대략 최소 약 18~20ms[ms = 1/1000s(초)] 간격에서 최대 약 215~230ms간격으로 가속도, 자기장, 각속도 값을 측정하여 기록하고 있다.

[0052] 이때 감지 데이터(DAT1)에 측정시간과 각 센서(101-103)의 측정값이 같이 기록되며 측정시간간격  $\Delta t$ 을 이용하여 일정시간 동안 이동한 거리를 계산하거나, 특정 시간에서의 기기의 자세나 위치를 계산할 수 있게 된다.

[0053] 본 예에서, 기기 좌표계란 감지부(100)의 센서(101-103)를 내장한 기기인 위치 정보 측정장치(1000)(예를들어 스마트폰이나 모바일 디바이스)의 해당 센서(101-103)에서 측정되는 감지값의 기준이 되는 좌표계를 의미한다.

[0054] 촬영부(110)는 사용자가 원하는 대상물의 촬영 동작을 실시하여 촬영된 대상물에 대한 영상 데이터(DAT2)를 생성하여 연산 및 제어 유닛(200)으로 출력한다.

[0055] 이러한 촬영부(110)는 CCD(charge coupled device) 소자 등과 같은 이미지 센서(image sensor)를 이용하여 영상 데이터를 생성하는 디지털 카메라일 수 있다.

[0056] 위치정보 획득부(120)는 GNSS(Global Navigation Satellite System)를 통하여 위치 정보 측정 장치(1000)의 위치정보(예, 위도와 경도)를 획득하여 원하는 대상물의 촬영 위치[예, 촬영부(110)의 렌즈 중심점]에 대한 절대좌표(DAT3)로 이용할 수 있도록 하는 장치이다. GNSS는 인공위성으로부터 수신한 전파신호를 이용하여 장치의 위치를 산출할 수 있는 항법 시스템을 의미하며, 예를 들어 GPS(Global Positioning System)가 대표적인 시스템이다.

[0057] 이러한 위치정보 획득부(120)는 본 발명의 필수 구성요소는 아니므로, 필요할 경우 생략 가능하다.

[0058] 이러한 위치정보 획득부(120)가 존재하지 않거나 위치정보 획득부(120)로부터의 원하는 위치 정보 측정 장치(1000)의 절대좌표(DAT3)위치정보를 획득할 수 없는 경우 또는 적대국의 GPS 방해 전파로 인해 원하는 절대좌표(DAT3)의 파악이 어려운 경우, 절대 좌표(DAT3)를 알고 있는 다른 지점을 촬영하여 위치 정보 측정 장치(1000)의 절대 좌표(DAT3)를 역산하거나 감지부(100)의 감지 데이터(DAT1)를 처리하여 위치 정보 측정 장치(1000)의 상대좌표를 획득할 수 있다.

[0059] 사용자 입력부(130)는 위치 정보 측정 장치(1000)의 동작에 필요한 명령이나 데이터 등을 입력 동작이 이루어져, 해당 입력 동작에 관련된 데이터(DAT4)를 생성해 연산 및 제어 유닛(200)으로 인가된다.

[0060] 이러한 사용자 입력부(130)는 터치 스크린(touch screen)일 수 있고, 경우에 따라 키 패드(key pad), 돔 스위치(dome switch), 터치 패드(touch pad), 조그(jog) 스위치 또는 마우스(mouse) 등일 수 있다.

[0061] 연산 및 제어 유닛(200)은 감지부(100)로부터 인가되는 감지 데이터(DAT1)와 촬영부(110)으로부터 인가되는 영상 데이터(DAT2)를 이용하여 원하는 대상물 간의 거리를 측정하고 또한 해당 대상물의 어느 지점의 좌표, 즉 특징점의 좌표(이하, 특징점의 좌표를 '특징점 좌표'라 함)를 측정한다.

[0062] 이러한 연산 및 제어 제어 유닛(200)은 감지 데이터(DAT1)와 영상데이터(DAT2)를 이용하여 거리와 좌표 계산을 실시하는 연산부(210)와 거리 측정 및 좌표 측정에 필요한 데이터와 계산 중에 생성되는 데이터를 저장하는 저

장부인 메모리(220)를 구비한다.

- [0063] 출력부(300)는 영상 및 제어 유닛(200)의 제어 동작에 의해 인가되는 데이터(DAT5)에 해당하는 영상을 출력하는 영상 출력 유닛이다. 따라서, 사용자는 출력부(300)를 통해 출력되는 영상을 실시간으로 보면서 사용자가 원하는 대상물을 촬영하여 영상 데이터(DAT2)를 취득할 수 있다.
- [0064] 따라서, 출력부(300)는 촬영부(110)의 동작에 의해 촬영된 영상을 출력하고, 원하는 대상물과 사용자[즉, 위치 정보 측정 장치(1000)] 간의 거리와 해당 촬영된 해당 대상물의 특징점 좌표를 측정하기 위해 메뉴 등에 대한 다양한 GUI(graphic user interface) 등을 출력한다.
- [0065] 이러한 출력부(300)는 액정 표시 장치(LCD, liquid crystal display)나 유기 발광 표시 장치(OLED, organic light emitting display) 등으로 이루어질 수 있다.
- [0066] 저장부(400)는 거리 측정 장치(1000)에 의해 산출된 위치 정보, 센서부(100)에 의해 생성된 감지 데이터, 자세 데이터 및 위치 데이터와 같은 저장 필요성이 있는 데이터(DAT6)를 저장할 수 있는 저장 매체이다.
- [0067] 이러한 저장부(400)는 위치 정보 측정장치(1000)의 외부에 설치되어 있는 외부 저장매체일 수 있고, 필요에 따라 생략될 수 있다.
- [0068] 통신부(500)는 서버(2000)와 같은 외부 장치와의 통신을 위한 통신 매체로서, 서버(2000)와 같은 외부 장치와의 통신을 가능하게 하는 하나 이상의 장치를 포함한다.
- [0069] 예를 들어, 통신부(500)는 이동 통신장치, 인터넷 장치 및 근거리 통신장치 중 적어도 하나를 구비할 수 있다.
- [0070] 이러한 통신부(500)를 이용하여 연산 및 제어 유닛(200)의 연산에 필요한 데이터, 예를 들어 시스템에 따른 카메라 내부표정을 위한 데이터, 시스템 업데이팅을 위한 데이터, 그리고 시스템을 사용함에 따라 획득되는 각종 자료, 영상데이터, 좌표데이터 등의 송수신 가능 데이터(DAT7)를 서버(2000)에 전송하거나 서버(2000)로부터 필요한 데이터의 수신을 실시할 수 있다.
- [0071] 서버(2000)는 위치 정보 측정장치(1000)와의 통신을 외부 장치로서 통신을 위한 통신부와 데이터 저장을 위한 저장 장치를 구비하고 있으며, 위치 정보 측정장치(1000)로부터 전송된 데이터, 예를 들어, 해당 대상물에 대한 측정된 거리 정보와 좌표 정보를 저장 장치에 저장할 수 있다.
- [0072] 이러한 서버(2000) 역시 필요할 경우 생략 가능하다.
- [0073] 이러한 구조를 갖는 위치 정보 측정 시스템의 동작을 설명한다.
- [0074] 먼저, 위치 정보 측정장치(1000)의 동작을 설명한다.
- [0075] 위치 정보 측정장치(1000)의 동작에 필요한 전원이 공급되어 위치 정보 측정장치(1000)는 동작 가능한 상태가 된다.
- [0076] 이런 동작 가능한 상태에서 사용자는 위치 정보 측정장치(1000)의 출력부(200)에 설치된 위치 정보 측정장치(1000)의 위치 정보 측정 동작을 위한 애플리케이션(application) 등을 구동하기 구동 아이콘(icon) 등을 동작시키면, 위치 정보 측정장치(1000)의 위치 정보 측정 동작이 시작되며, 연산 및 제어 유닛(200)의 연산부(210)의 동작 역시 시작된다.
- [0077] 먼저, 도 2를 참고로 하여, 본 실시예에 따른 위치 정보 측정 장치(1000)의 동작 개념을 설명한다.
- [0078] 먼저, 북쪽(N)을 X축으로 하는 평면직교좌표계(CS1)안에서 사용자는 두 개의 위치인 제1 및 제2 위치(예, A위치와 B위치)에서 위치 정보 측정을 위한 대상물에 대한 2회의 사진 촬영을 실시한다. 이때, 측정의 정확도를 높이기 위해 추가적인 사진 촬영이 이루어질 수 있다.
- [0079] 해당 대상물에 대한 사진 촬영이 이루어지면, 먼저, 연산부(210)는 A위치에서의 위치 정보 측정장치(1000)의 위치 데이터( $X_A, Y_A, Z_A$ )와 A에서의 위치 정보 측정장치(1000)의 자세 데이터( $\omega_A, \phi_A, \kappa_A$ )를 구하고, 다시 B위치로 위치 정보 측정장치(1000)가 이동하였을 때의 위치 정보 측정장치(1000)의 위치 데이터( $X_B, Y_B, Z_B$ )와 자세 데이터( $\omega_B, \phi_B, \kappa_B$ )를 구한다.
- [0080] A위치와 B위치에서의 각 촬영 동작에 의해 촬영부(110)에서 생성된 영상데이터(DAT2-A, DAT2-B)를 영상처리하여

각 영상 데이터(DAT2-A, DAT2-B)에서의 대상물의 사진 좌표 $[(x_{at}, y_{at})]$  및  $(x_{bt}, y_{bt})$ 를 취득한다.

[0081] 다음, 산출한 위치 정보 측정장치(1000)의 위치 데이터 및 자세 데이터 $[(X_A, Y_A, Z_A), (X_B, Y_B, Z_B), (\omega_A, \phi_A, \kappa_A), (\omega_B, \phi_B, \kappa_B)]$ 와 사진 좌표 $[(x_{at}, y_{at})]$  및  $(x_{bt}, y_{bt})$ ], 촬영부(110)의 촬영 시 카메라의 초점거리( $f$ ) 등의 데이터를 사진측량학을 적용하여 해당 대상물 즉, 목표 대상물(T)의 절대 좌표 $(X_T, Y_T, Z_T)$ 을 구한다.

[0082] 위와 같은 과정을 목표 대상물(T) 상의 다른 점으로 확대하여 반복 계산하여 사진에 촬영된 다양한 대상물이나 특징점의 절대좌표와 상대좌표를 구할 수 있다.

[0083] 절대좌표 혹은 상대좌표가 계산되면, 좌표 관계식을 이용하여 관측자[즉, 위치 정보 측정장치(1000)]에서부터 대상물의 해당 지점까지의 거리와 대상물의 높이, 또는 대상물의 두 지점들간의 거리, 각 지점의 높이, 해당 대상물의 경사(즉 기울기), 그리고 해당 대상물에서 원하는 부분의 면적 및 부피 등을 계산할 수 있다.

[0084] 본 예의 경우, 위치정보 획득부(120)를 이용할 수 없어 촬영 위치의 절대좌표를 취득할 수 없을 때, 좌표관계식이나 천문측량 등의 방법을 통해 촬영위치의 절대좌표를 대안적으로 산출할 수 있다.

[0085] 그러면, 도 3을 참고로 하여, 이러한 개념으로 동작하는 위치 정보 측정장치(1000)의 동작에 대한 개략적인 동작을 살펴본다.

[0086] 위치 정보 측정장치(1000)의 연산부(210)는 감지부(100)로부터 인가되는 감지 데이터(DAT1)를 이용하여 위치 정보 측정장치(1000), 좀더 정확하게는 촬영부(110)의 위치 데이터와 자세 데이터를 계산하는 단계(S100), 원하는 대상물에 대한 2회 이상의 사진 촬영으로 인한 영상 데이터(DAT2) 획득 단계(S200), 해당 대상물에 대한 사진 촬영이 완료된 후, 생성된 두 개 이상의 영상 데이터(DAT2)를 분석 및 매칭하는 단계(S300), 매칭 동작이 완료된 영상 데이터를 통해 대상물의 영상 좌표를 산출하는 단계(S400), 사진측량학을 이용하여 대상물의 각 특징점의 좌표를 계산하는 단계(S500), 그리고 대상물의 촬영위치에 대한 절대좌표를 모를 때 천문측량 등을 이용해 절대좌표를 파악하고, 위치정보 획득부(120)로부터 인가되는 위치정보를 이용하여 대상물의 촬영위치에 대한 절대좌표를 산출하고, 산출된 절대좌표간의 관계식을 이용하여 해당 대상물의 특징점들 사이, 그리고 촬영 위치[즉, 위치 정보 측정장치(1000)]와 대상물의 특징점의 사이 또는 여러 대상물들의 특징점들간의 거리, 높이, 경사 및 면적 등의 관계를 계산하는 단계(S600)를 구비한다.

[0087] 따라서, 이러한 위치 정보 측정장치(100)는 위치 정보 측정장치(1000)의 자세, 위치를 추적하여 기록하고(S100) 2회 이상의 사진촬영으로 인한 영상 데이터를 획득하는(S200) 영상촬영부분(P1), 촬영 동작에 의한 영상 데이터를 이용하여 영상을 분석하고 영상 매칭 동작을 실시하며(S300) 매칭된 영상에서 대상물의 특징점의 영상좌표를 취득하는(S400) 영상처리부분(P2), 그리고 대상물의 특징점의 영상좌표와 위치 정보 측정장치(1000)의 자세 위치 관계를 이용하여 대상물의 특징점의 좌표를 파악하고(S500), 촬영위치의 절대좌표를 모를 때 촬영위치의 절대 좌표를 결정하거나, 좌표 관계식을 이용하여 촬영위치와 특징점간의 위치 관계 그리고 특징점들의 위치 간계를 계산하는 부분(S600)인 측량부분(P3)으로 나눠질 수 있다.

## 1. 영상 촬영 부분(P1)

### 1) 영상 촬영 부분 사용 수식

[0089] 다음, 도 4 및 도 5를 참고로 하여, 감지부(100)로부터 인가되는 감지 데이터(DAT1)를 이용하여 글로벌 좌표계에서의 위치 정보 측정장치(100)의 자세 데이터와 위치 데이터를 산출하는 과정을 설명한다.

[0091] 도 4의 (a)는 기기좌표계와 글로벌좌표계의 변환을 나타내는 그림으로서 굽은 실선으로 표시된 것은 글로벌 좌표계인 평면 직교 좌표계의 X축, Y축 및 Z축 방향으로의 3개 축(GC1, GC2, GC3)을 나타낸다. 이때, X축 방향은 북쪽(N)을 가리키며, Y축 방향은 동쪽(E)을 가리키며, Z축 방향은 지면에 수직인 중력방향을 가리킨다.

[0092] 각 축(GC1, GC2, GC3)에 대한 회전각(GR1, GR2, GR3)은 각각 글로벌 좌표계에서의 위치 정보 측정장치(1000)의 자세를 뜻하는 방위각( $\kappa$ )(GR1), 고저각( $\phi$ )(GR2) 및 회전각( $\omega$ )(GR3)을 각각 나타낸다.

[0093] 고저각은 연직면(중력방향면)에서의 수평선과 시준축이 이루는 각으로서 수평선으로부터 상향으로 측정하여 수평선에 수직으로 하늘이나 천정을 향할 때는 +90도 수평선에 하향으로 측정하여 수평선에 수직으로 바닥이나 땅

을 향할 때는 -90도, 수평선과 시준축이 일치할 때를 0도로 한다. 예를 들어 평면 직교 좌표계에서 Y축 방향 회전각을 뜻한다.

- [0094] 회전각은 기기[예, 위치 정보 측정장치(1000)]가 시준축을 기준으로 시계방향으로 얼마나 회전했는가를 나타내는 각도로서 예를 들어 평면직교 좌표계에서 X축 방향 회전각을 뜻한다.
- [0095] 시준이란 목표를 향해 바라본다는 의미이므로, 시준축은 카메라가 목표를 시준하고 있을 때 카메라의 광축을 의미한다.
- [0096] 방위각은 북쪽을 기준으로 시계방향으로 시준축과 이루는 수평각으로서, 도북이 기준일 경우 도북방위각, 센서 계산을 통한 자북을 기준할 경우 자북방위각, 진북을 기준할 경우 진북방위각이라 한다.
- [0097] 본 명세서에서 일반적인 방위각은 평면직교좌표계를 기준으로 한 도북방위각을 말하며, 평면 직교 좌표계에서 Z 축 방향 회전각을 뜻한다. 부득이 도북방위각 계산이 어려울 경우 자북 방위각을 뜻할 수도 있다.
- [0098] 좌표 중심에 표시된 사각형은 본 시스템이 설치된 기기인 위치 정보 측정장치(1000)를 의미하며, 기기[즉, 위치 정보 측정장치(1000)]에 표시된 기기 좌표계의 3개의 축인 0번축(MC0), 1번축(MC1) 및 2번축(MC2)은 위치 정보 측정장치(1000)의 각 면에 수직으로 구성되어 있다.
- [0099] 안드로이드 스마트폰에서 내장된 센서들의 측정값은 배열(array)형태로 제공되는데 이 측정값의 배열 번호를 각 축의 이름으로 부여함에 따라, 도 4의 (a)에서, 각 기기 좌표축(MC0, MC1, MC2)의 번호는 0, 1 및 2으로 부여되었다.
- [0100] 예를 들어 안드로이드 스마트폰에서 가속도 센서의 측정값은 `event.values[0]`, `event.values[1]` 및 `event.values[2]`와 같이 배열 형태로 제공되는데, 이 때의 0, 1, 2 값이 기기 좌표계의 각 축방향이 된다.
- [0101] 일반적으로, 0번축은 위치 정보 측정장치(1000)의 단축 방향이고, 1번축은 위치 정보 측정장치(1000)의 장축 방향이며, 2번축은 위치 정보 측정장치(1000)의 화면 방향이 된다[도 4의 (b) 참조].
- [0102] 본 예에서, 위치 정보 측정장치(1000)에 설치된 가속도 센서(101)와 자이로 센서(102) 및 자기장 센서(103)는 기기 좌표계의 3개 축[0번축(MC0), 1번축(MC1) 및 2번축(MC2)]에 따라 측정되고 있으며, 그 중심은 도 4의 (b)에 도시한 것처럼, 기기 중심점을 기준으로 측정되고 있다.
- [0103] 각 좌표계의 3개축(MC0, MC1, MC2) 중에서, 0번축(MC0)에서 측정되는 가속도를  $a_0$ , 자기장을  $m_0$  및 각속도를  $G_0$ 이라 하고, 1번축(MC1)에서 측정되는 가속도를  $a_1$ , 자기장을  $m_1$  및 각속도를  $G_1$ 이라 하며, 2번축(MC2)에서 측정되는 가속도를  $a_2$ , 자기장을  $m_2$  및 각속도를  $G_2$ 라고 정의 한다.
- [0104] 이를  $a_0$ ,  $a_1$ ,  $a_2$ ,  $m_0$ ,  $m_1$ ,  $m_2$ ,  $G_0$ ,  $G_1$ ,  $G_2$ 는 이후의 [수식1]~[수식6]까지 사용되는 부호들이다.
- [0105] 아래의 수식들은 본 발명의 알고리즘을 설명하기 위한 일반식으로서 기기에 설치된 센서 측정 방향에 따라 부호가 달라질 수 있으며, 측정 단위를 통일하거나, 2차 계산을 편리하게 하기 위해 각도의 단위를 변경하고, 축 방향을 변경하기 위한 간단한 수학적 처리 등은 생략하였다.
- [0106] 다음, 도 5를 참고로 하여, 감지부(100)인 가속도 센서(101), 자이로 센서(102) 및 자기장 센서(103)에서 각각 출력되는 가속도 데이터, 각속도 데이터 및 자기장 데이터를 이용하여 위치 정보 측정장치(1000)의 자세 데이터를 산출하는 방법과 산출된 자세 데이터와 가속도 데이터를 이용하여 글로벌 좌표계의 각 축방향(GC1, GC2, GC3)에 대한 상대 좌표를 산출하는 방법을 설명한다.
- [0107] 위치 정보 측정장치(1000)의 자세 데이터를 산출하기 위해, 먼저, 연산부(210)는, [수식1]를 이용하여, 가속도 센서(101)와 자기장 센서(103)로부터 인가되는 가속도 데이터와 자기장 데이터를 이용하여 위치 정보 측정장치(1000)의 가속도와 자기장 데이터를 통해 구한 예비 자세 데이터( $\omega_{\text{가속도}}$ ,  $\phi_{\text{가속도}}$ ,  $\kappa_{\text{자기장}}$ )(이하, 이 예비 자세 데이터를 '가속도 예비 자세 데이터'라 함)를 산출한다(S11).
- [0108] [수식1]

[0109]  $\omega_{\text{가속도}} = \sin^{-1} \left[ \frac{-a_1}{g \cos \phi} \right]$  ( $a_1$  : 중력가속도)

[0110]  $\phi_{\text{가속도}} = \sin^{-1} \left[ \frac{a_2}{g} \right]$

[0111]  $\kappa_{\text{자기장}} = \tan^{-1} \left[ \frac{Y_H}{X_H} \right]$

[0112]  $- Y_H = m_2 \cos \phi + m_1 \sin \omega \sin \phi - m_0 \cos \omega \sin \phi$

[0113]  $- X_H = m_1 \cos \omega + m_0 \sin \omega$

[0114] [수식1]에서,  $\omega$ 는 회전각이고,  $\phi$ 는 고저각이며,  $\kappa$ 는 방위각이다.

[0115] 연산부(210)의 동작에 의해, 자이로 센서(102)로부터 인가되는 각속도 데이터를 이용하여 위치 정보 측정장치(100)의 자이로 센서(102)의 데이터를 통해 구한 예비 자세 데이터( $\omega_{\text{자이로}}$ ,  $\phi_{\text{자이로}}$ ,  $\kappa_{\text{자이로}}$ ) (이하, 이 예비 자세 데이터를 '자이로 예비 자세 데이터'라 함)는 다음의 [수식 2]를 측정시간으로 적분되어 적분된 자이로 예비 자세 데이터가 산출된다(S12).

[0116] [수식2]

$$\begin{bmatrix} \dot{\omega}_{\text{자이로}} \\ \dot{\phi}_{\text{자이로}} \\ \dot{\kappa}_{\text{자이로}} \end{bmatrix} = \begin{bmatrix} 1 & \sin \omega \tan \phi & \cos \omega \tan \phi \\ 0 & \cos \omega & -\sin \omega \\ 0 & \sin \omega / \cos \phi & \cos \omega / \cos \phi \end{bmatrix} \begin{bmatrix} G_2 \\ G_1 \\ G_0 \end{bmatrix}$$

[0118] [수식 1]에 의해 산출된 위치 정보 측정장치(1000)의 가속도 예비 자세 데이터는 가속도 변화와 감지값의 무작위 변동에 따른 노이즈에 민감하게 반응하며, [수식 2]에 의해 산출된 자이로 예비 자세 데이터( $\omega_{\text{자이로}}$ ,  $\phi_{\text{자이로}}$ ,  $\kappa_{\text{자이로}}$ )는 시간이 지남에 따라 오차가 누적되어 점점 커지게 된다.

[0119] 따라서, 연산부(210)는 [수식 1]과 [수식 2]를 통해서 산출된 가속도 예비 자세 데이터( $\omega_{\text{가속도}}$ ,  $\phi_{\text{가속도}}$ ,  $\kappa_{\text{자기장}}$ )와 자이로 예비 자세 데이터( $\omega_{\text{자이로}}$ ,  $\phi_{\text{자이로}}$ ,  $\kappa_{\text{자이로}}$ )를 합성하여(S13) 위치 정보 측정장치(1000)의 최종 자세 데이터( $\omega_f$ ,  $\phi_f$ ,  $\kappa_f$ )를 산출한다(S14).

[0120] 본 예에서, 두 예비 자세 데이터[( $\omega_{\text{가속도}}$ ,  $\phi_{\text{가속도}}$ ,  $\kappa_{\text{자기장}}$ ), ( $\omega_{\text{자이로}}$ ,  $\phi_{\text{자이로}}$ ,  $\kappa_{\text{자이로}}$ )]를 합성하기 위해 연산부(210)는 상보필터나 칼만필터를 사용한다.

[0121] 상보필터는 감지 데이터를 합쳐 새로운 결과 값을 얻으려 할 때 합쳐지는 센서의 주파수 특성이 서로 보완적인 특성을 이용하여 오차를 보정해주는 필터로서, 융합되는 센서의 주파수 특성이 서로 보완적인 경우에만 적용이 가능하다.

[0122] 칼만필터는 잡음이 섞여 있는 관측값을 최소제곱법을 통해 분석함으로써 어떤 시스템에서 발생할 수 있는 잡음을 최소화해 계산할 수 있는 수학적 필터이다.

[0123] 상보필터를 사용하는 경우 계산 속도가 빠르며, 칼만필터를 사용할 경우 상보필터 보다 계산 속도는 느리지만, 정확한 결과값을 얻을 수 있다.

[0124] 먼저, 상보필터를 적용할 경우, 연산부(210)는 다음 [수식3]를 이용하여 위치 정보 측정장치(1000)의 최종 자세 데이터( $\omega_f$ ,  $\phi_f$ ,  $\kappa_f$ )를 구한다.

[0125] [수식3]

[0126]  $\omega_f = \omega_{\text{자이로}} \times (1 - k) + \omega_{\text{가속도}} \times k$

[0127]  $\phi_f = \phi_{\text{자이로}} \times (1 - k) + \phi_{\text{가속도}} \times k$

[0128]  $\kappa_f = \kappa_{\text{자이로}} \times (1 - k) + \kappa_{\text{자기장}} \times k [k: \text{상수 } (0 < k < 1)]$

[0129]  $k$ 는 측정 값의 주파수 특성에 따라 조정되는 상수로서, 위치 정보 측정장치(1000)는 자신에게 적합한 상보필터를 구현하여 최종 자세 데이터( $\omega_f, \phi_f, \kappa_f$ )를 구한다(S14).

[0130] 반면, 칼만필터를 이용하여 센서 합성을 실시하는 경우, 자세 데이터는 쿼터니언(quaternion) 함수로 변환된 후 칼만필터 적용되어야 한다. 이를 위한 일반식은 다음 [수식4]와 같다.

[0131] [수식4]

[0132] (1) 칼만필터 일반식

[0133]  $x_{k+1} = A x_k + w_k [x_k: \text{상태변수}(n \times 1 \text{ 열벡터}), A: \text{시스템 모델 행렬}(n \times n\text{행렬}), w_k: \text{시스템 잡음}(n \times 1 \text{ 열벡터})]$

[0134]  $z_{k+1} = H x_k + v_k [z_k: \text{측정값}(m \times 1 \text{ 열벡터}), H: m \times n \text{ 행렬}, v_k: \text{측정잡음}(m \times 1 \text{ 열벡터})]$

[0135] (2) 시스템 모델 행렬

$$A = I + \Delta t \cdot \frac{1}{2} \begin{bmatrix} 0 & -p & -q & r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix}, (p = G_2, q = G_1, r = G_0)$$

[0137] (3) 쿼터니언함수로 변환한 측정값

$$z_k = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \cos \frac{\omega}{2} \cos \frac{\phi}{2} \cos \frac{\kappa}{2} + \sin \frac{\omega}{2} \sin \frac{\phi}{2} \sin \frac{\kappa}{2} \\ \sin \frac{\omega}{2} \cos \frac{\phi}{2} \cos \frac{\kappa}{2} - \cos \frac{\omega}{2} \cos \frac{\phi}{2} \sin \frac{\kappa}{2} \\ \cos \frac{\omega}{2} \sin \frac{\phi}{2} \cos \frac{\kappa}{2} + \sin \frac{\omega}{2} \cos \frac{\phi}{2} \sin \frac{\kappa}{2} \\ \cos \frac{\omega}{2} \cos \frac{\phi}{2} \sin \frac{\kappa}{2} - \sin \frac{\omega}{2} \sin \frac{\phi}{2} \cos \frac{\kappa}{2} \end{bmatrix}$$

[0138] (4) 칼만 필터 계산과정

[0140] ① 초기값 가정

[0141]  $H = I, Q = I \times k_1, R = I \times k_2$

[0142] [I:  $4 \times 4$  단위행렬, Q, R: 잡음의 공분산 행렬, (6)과정을 통해 계산]

$$X = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{상태변수 초기 값})$$

[0144]  $P = I$  (오차공분산 초기 값)

[0145] ②  $X_p = A \times X$

[0146] ③  $P_p = A \times P \times A^T + Q$

[0147] ④  $K = P_p \times H^T \times (H \times P_p \times H^T + R)^{-1}$

[0148] ⑤  $X = X_p + K \times (z_k - H \times X_p)$ ,  $P = P_p - K \times H \times P_p$

[0149] (이후 이 값이 초기 값이 되어 ②과정 반복 계산)

[0150] (5) 쿼터니언함수값을 기기자세값으로 변환

$$X = \begin{bmatrix} q_{1f} \\ q_{2f} \\ q_{3f} \\ q_{4f} \end{bmatrix}$$

[0151] (칼만필터에 의해 최종으로 조정된 쿼터니언)

$$\omega_f = \tan^{-1}\left(\frac{2 \times (q_{3f}q_{4f} + q_{1f}q_{2f})}{1 - 2q_{2f}^2 + q_{3f}^2}\right)$$

$$\phi_f = -\sin^{-1}(2 \times (q_{2f}q_{4f} - q_{1f}q_{3f}))$$

$$\kappa_f = \tan^{-1}\left(\frac{2 \times (q_{2f}q_{3f} + q_{1f}q_{4f})}{1 - 2q_{3f}^2 + q_{4f}^2}\right)$$

[0155] (6) 잡음의 공분산 행렬 계산식

[0156] ①  $Q =$  시스템 잡음  $w_n$  의 공분산 행렬:  $(n \times n)$  대각 행렬

[0157]  $n$  개의 시스템 잡음  $w_1, w_2, \dots, w_n$  이 있고

[0158] 각 잡음의 분산이  $\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2$  이라면,

$$Q = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}$$

[0160] ②  $R =$  측정 잡음  $v_m$  의 공분산 행렬:  $(m \times m)$  대각 행렬

[0161]  $m$  개의 측정 잡음  $v_1, v_2, \dots, v_m$  이 있고

[0162] 각 잡음의 분산이  $\sigma_1^2, \sigma_2^2, \dots, \sigma_m^2$  이라면,

$$R = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_m^2 \end{bmatrix}$$

[0164] 시스템 행렬(A)에 자이로 센서(102)에 의해 측정된 각속도 값을 넣고, 최초 측정값을 0으로 가정하거나, [수식 1]을 통해서 구한 예비 자세 데이터값을 최초 측정값으로 가정한 후 칼만필터를 적용하면 위치 정보 측정장치(1000)의 최종 자세 데이터( $\omega_f, \phi_f, \kappa_f$ )가 산출된다(S14).

[0165] 칼만필터는 비교적 정확한 자세 데이터를 얻을 수 있으나 행렬 연산식에 따라 복잡하고 계산과정이 길어, 실시간 처리를 위해서는 높은 성능의 연산능력이 필요하다.

[0166] 또한, 위치 정보 측정장치(1000)에 설치된 센서 잡음의 공분산 행렬을 반복 계산이나 실험, 센서 제조사에서 제공해주는 잡음 특성을 사용하여 결정해 줘야 한다.

[0167] 이처럼, 상보필터나 칼만필터를 통해서 위치 정보 측정장치(1000)의 최종 자세 데이터( $\omega_f, \phi_f, \kappa_f$ )가

산출된다.

[0168] 이러한 산출된 최종 자세 데이터( $\omega_f, \phi_f, \kappa_f$ )는 글로벌 좌표계에서 각 축방향(GC1-GC3)에 대한 이동거리, 즉 각 축방향(GC1-GC3)에 대한 상대좌표를 구하는 기초 데이터가 된다.

[0169] 즉, 연산부(210)는 산출된 최종 자세 데이터( $\omega_f, \phi_f, \kappa_f$ )와 가속도 센서(101)에 의해 획득된 가속도 데이터( $a_0, a_1, a_2$ )를 판독한 후(S15-1, S15-2), 3차원 회전 변환시켜(S16) 글로벌 좌표계에서의 각 축방향(GC1-GC3)의 가속도( $a_x, a_y, a_z$ )를 구한다.

[0170] 이 때, 각 축 방향(GC1-GC3)에 대한 가속도 즉, 각 축(X, Y, Z) 방향으로의 즉, 축방향 가속도( $a_x, a_y, a_z$ )를 산출하기 위한 일반식은 다음 [수식 5]와 같다.

[0171] [수식5]

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = R_\kappa \times R_\phi \times R_\omega \times \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix}$$

$$R_\omega = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & -\sin\omega \\ 0 & \sin\omega & \cos\omega \end{bmatrix}$$

$$R_\phi = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

$$R_\kappa = \begin{bmatrix} \cos\kappa & -\sin\kappa & 0 \\ \sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

[0176] [수식 5]는 일반식으로서 각 위치 정보 측정장치(1000)에 맞게 부호 및 회전 방향, 회전메트릭스( $R_\omega, R_\phi, R_\kappa$ ) 적용순서를 조정하여야 한다. 예를 들어, 일반적인 안드로이드 스마트폰의 경우  $a_1, a_2$ 는 각각  $-a_1, -a_2$ 로 적용하여야 한다.

[0177] 3차원 회전 메트릭스( $R_\omega, R_\phi, R_\kappa$ )의 경우 곱셈 순서와 회전방향의 부호를 조정하면 3차원 회전메트릭스의 역행렬을 구현할 수 있다.

[0178] 따라서, 본 명세서 상에서는 3차원 회전메트릭스( $R_\omega, R_\phi, R_\kappa$ )의 적용순서에 대하여 역행렬로 적용해야 할 경우나, 회전메트릭스 곱셈 순서가 다르게 적용해야 할 때에 대해서도 위 [수식5]와 같은 일반식으로 표현하도록 하겠다.

[0179] 각 시스템 특성 및 상황에 맞는 3차원 회전메트릭스 적용순서 및 회전방향은 실험을 통한 경험식으로 구하거나, 감지부(100)의 각 센서(101~103) 특성 및 모델링 특성을 고려하여 수학적인 계산을 통해 구할 수 있다.

[0180] 본 명세서상에서 3차원 회전변환을 설명하기 위한 한 예로서 타이트 브라이언 각(Tait-Bryan angles) 변환 공식을 사용하였으나, 타이트 브라이언 각 외에도 쿼터니언 함수를 사용한 3차원 회전변환도 가능하고, 공식으로서 적용이 가능하다.

[0181] 쿼터니언 함수를 사용한 3차원 회전변환은 대중에게 공개된 수식이고, 본 발명의 개념 설명은 타이트 브라이언 각 변환 공식으로 충분하기 때문에 쿼터니언 함수를 이용한 자세 변환 공식은 본 명세서 상에서 생략하나, 본 발명에 쿼터니언 함수를 적용하여 자세변환도 가능함을 명시하도록 한다.

[0182] 각 축방향 가속도( $a_x, a_y, a_z$ )를 구하기 위해 위와 같은 3차원 회전변환을 이용하여 바로 적용할 수 있으나, 연산처리능력이 충분하다면 본 회전메트릭스 변환에서도 칼만필터를 적용하여 가속도 데이터와 자세 데이터의

오차를 보정하여 보다 정확한 축방향 가속도( $a_x, a_y, a_z$ )를 얻을 수 있다.

[0183] 다음 [수식6]은 칼만 필터를 적용하여 각 축방향 가속도( $a_x, a_y, a_z$ )를 구하는 방법이다.

[0184] [수식6]

(1) 칼만필터 일반식

[0186]  $x_{k+1} = A x_k + w_k$     [ $x_k$ : 상태변수( $n \times 1$  열벡터),  $A$ : 시스템 모델 행렬( $n \times n$  행렬),  $w_k$ : 시스템 잡음( $n \times 1$  열벡터)]

[0187]  $z_{k+1} = H x_k + v_k$     [ $z_k$ : 측정값 ( $m \times 1$  열벡터),  $H$ :  $m \times n$  행렬,  $v_k$ : 측정 잡음 ( $m \times 1$  열벡터)]

[0188] (2) 시스템 모델 행렬

$$A = \begin{bmatrix} 0 & 0 & 0 & R_{total}(1,1) & R_{total}(1,2) & R_{total}(1,3) \\ 0 & 0 & 0 & R_{total}(2,1) & R_{total}(2,2) & R_{total}(2,3) \\ 0 & 0 & 0 & R_{total}(3,1) & R_{total}(3,2) & R_{total}(3,3) \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

[0189]  $R_{total} = R_\kappa \times R_\phi \times R_\omega$  ([수식5] 참조)

[0191] (회전메트릭스의 적용순서 및 방향은 각 시스템에 맞게 결정)

[0192]  $R_{total}(n,m)$  :  $R_{total}$  행렬의  $n$  행,  $m$  열의 값

[0193] (3) 측정값

$$z_k = \begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix}$$

[0194] (4) 칼만 필터 계산과정

[0195] ① 초기값 가정

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

[0196]  $Q = 6 \times 6$  대각행렬([수식4]의 (6)과정을 통해 계산 가능)

[0197]  $R = 3 \times 3$  대각행렬([수식4]의 (6)과정을 통해 계산 가능)

$$X = \begin{bmatrix} a_x \\ a_y \\ a_z \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

[0200] (상태변수 초기 값, 정지상태 였다면 0)

[0201]  $P = I$  ( $6 \times 6$  단위행렬, 오차공분산 초기 값)

[0202] ②  $X_p = A \times X$

[0203] ③  $P_p = A \times P \times A^T + Q$

[0204] ④  $K = P_p \times H^T \times (H \times P_p \times H^T + R)^{-1}$

[0205] ⑤  $X = X_p + K \times (z_k - H \times X_p), \quad P = P_p - K \times H \times P_p$

[0206] (이) 후 이 값이 초기 값이 되어 ②과정 반복 계산)

$$X = \begin{bmatrix} a_x \\ a_y \\ a_z \\ \vdots \\ a_2 \\ \vdots \\ a_1 \\ \vdots \\ a_0 \end{bmatrix} \quad (\text{칼만필터에 의해 최종으로 조정된 } a_x, a_y, a_z)$$

[0208] 칼만필터를 통해 각 축방향 가속도( $a_x, a_y, a_z$ )를 구할 경우  $6 \times 6$  행렬의 역행렬을 계산하여야 하는 등 그 계산 과정이 대단히 복잡하고, 오래 걸리게 된다.

[0209] 통상적으로는 [수식5]와 같이 회전메트릭스 일반식으로 구하는 것이 쉽고 빠르나, 오차를 줄여 정확한 계산을 위해서는 [수식6]과 같은 칼만필터를 적용한 수식이 보다 정확하다.

[0210] 위와 같은 [수식5] 또는 [수식6]을 통해 구하여진 각 축방향 가속도( $a_x, a_y, a_z$ )에서 Z축 가속도( $a_z$ )에는 중력 가속도가 포함되어 있다.

[0211] 따라서, 정확한 Z축 방향으로의 이동거리를 Z축 방향 이동거리를 계산하기 위해, 연산부(210)는 Z축 가속도( $a_z$ )에서 중력 가속도( $g$ )를 빼준다(S17).

[0212] 중력 가속도( $g$ )는 일반적인 상수로서 약 9.81㎨를 빼서 Z축 가속도( $a_z$ )를 보정한다. 하지만, 대안적인 예로서, 감지부(100)의 각 센서(101-103)의 영점 조정을 위해 위치 정보 측정장치(1000)의 완전 정지 상태에서의 가속도를 측정하여 중력 가속도를 산출해, 산출된 중력 가속도를 이용하여 Z축 가속도( $a_z$ )를 보정할 수 있다. 이 경우, 보정 동작의 정확도를 높일 수 있다.

[0213] 연산부(210)는 중력 가속도( $g$ )를 이용하여 Z축 가속도( $a_z$ )를 보정한 후(S17), 보정된 Z축 가속도( $a_z$ ) 및 산출된 X축 및 Y축 가속도( $a_x, a_y$ )를 시간에 대하여 이중 적분하여 각 해당 축방향(X축 Y축 및 Z축)으로의 이동거리(즉, 축방향 이동거리)를 구하거나, 칼만필터를 적용하여 각 해당 축방향(X축 Y축 및 Z축)으로의 이동거리를 구한다(S18).

[0214] 이중 적분 방식을 이용하여 해당 축방향으로의 이동거리를 구하기 위해, 연산부(210)는 다음 [수식7]에 도시한 적분 수식을 이용하여 X, Y, Z 각 축방향 이동거리를 계산한다.

[0215] [수식7]

$$V_n = V_{n-1} + (a_{n-1} + (a_n - a_{n-1})/2) \times \Delta t \quad [\text{가속도를 적분 속도}(V_n)\text{계산}]$$

$$S_n = S_{n-1} + (V_{n-1} + (V_n - V_{n-1})/2) \times \Delta t \quad [\text{속도를 적분 이동거리}(S_n)\text{계산}]$$

[0218]  $a_n$ : 현재 계산된 가속도,  $a_{n-1}$ : 직전 시간에 계산된 가속도

[0219]  $V_n$ : 현재 속도,  $V_{n-1}$ : 직전 시간에 계산된 속도

[0220]  $S_n$ : 현재 이동거리,  $S_{n-1}$ : 직전 시간에 계산된 이동거리

[0221]  $\Delta t$  : 센서 측정 간격

[0222] 예를 들어, [수식5] 또는 [수식6]을 통해 X축 방향 가속도가  $\Delta t = 200\text{ms} = 0.2\text{sec}$ (초) 간격으로  $a_{x0} = 0 \frac{\text{m}}{\text{s}^2}$ ,  $a_{x1} = 0.1 \frac{\text{m}}{\text{s}^2}$ ,  $a_{x2} = 0.2 \frac{\text{m}}{\text{s}^2}$ 으로 계산되었다면 X축 방향 속도는 다음과 같다.

$$V_{x0} = 0 \frac{\text{m}}{\text{s}}$$

$$V_{x1} = V_{x0} + (a_{x0} + (a_{x1} - a_{x0})/2) \times \Delta t = 0 + (0 + (0.1 - 0)/2) \times 0.2 = 0.01 \frac{\text{m}}{\text{s}}$$

$$V_{x2} = V_{x1} + (a_{x1} + (a_{x2} - a_{x1})/2) \times \Delta t = 0.01 + (0.1 + (0.2 - 0.1)/2) \times 0.2 = 0.04 \frac{\text{m}}{\text{s}}$$

[0226] 속도가 위와 같이 계산되었다면 X축 방향 이동거리는 다음과 같다.

$$S_{x0} = 0 \text{ m}$$

$$S_{x1} = S_{x0} + (V_{x0} + (V_{x1} - V_{x0})/2) \times \Delta t = 0 + (0 + (0.01 - 0)/2) \times 0.2 = 0.001 \text{ m}$$

$$S_{x2} = S_{x1} + (V_{x1} + (V_{x2} - V_{x1})/2) \times \Delta t = 0.001 + (0.01 + (0.04 - 0.01)/2) \times 0.2 = 0.006 \text{ m}$$

[0230] 각 시간에 따른 이동거리는 위와 같고, 최종 X축 방향 이동거리는 0.006m가 된다.

[0231] 위와 같은 방법으로 Y축, Z축 방향 가속도가 측정이 되면 Y축, Z축 이동거리가 계산된다.

[0232] [수식7]에 도시한 적분 방법은 하나의 예로서, 다른 적분 방식이 적용될 수 있음을 당연하다.

[0233] 이중적분을 통한 이동거리 계산은 계산 속도가 빨라 실시간 처리에 적합하다. 대안적인 방법으로, 칼만필터를 적용하여 각 축방향으로의 이동거리를 산출하기 위해서, 연산부(210)는 다음 [수식8]을 이용한다.

[0234] [수식8]은 칼만필터를 적용하여 각축에 대한 속도( $V_n$ )와 이동거리( $S_n$ )를 계산하는 일반식이다.

[0235] [수식8]

[0236] (1) 칼만필터 일반식

[0237]  $x_{k+1} = A x_k + w_k$  [ $x_k$ : 상태변수( $n \times 1$  열벡터),  $A$ : 시스템 모델 행렬( $n \times n$ 행렬),  $w_k$ : 시스템 잡음( $n \times 1$  열벡터)]

[0238]  $z_{k+1} = H x_k + v_k$  [ $z_k$ : 측정값 ( $m \times 1$  열벡터),  $H$ :  $m \times n$  행렬,  $v_k$ : 측정 잡음 ( $m \times 1$ 열벡터)]

[0239] (2) 속도 계산식

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} (\Delta t: \text{센서의 측정 간격})$$

$$z_k = a_n \text{ (가속도 측정값)}$$

[0242] ① 초기값 가정

$$H = [0 \ 1]$$

[0244]  $Q = 2 \times 2$  대각행렬 ([수식4]의 (6)과정을 통해 계산 가능)

[0245]  $R = k (k= \text{상수})$  ([수식4]의 (6)과정을 통해 계산 가능)

$$X = \begin{bmatrix} V_1 \\ a_1 \end{bmatrix} \text{ (상태변수 초기 값, 정지상태 였다면 0)}$$

$$P = I \text{ (2} \times 2 \text{ 단위행렬, 오차공분산 초기 값)}$$

[0248] ②  $X_p = A \times X$

[0249] ③  $P_p = A \times P \times A^T + Q$

[0250] ④  $K = P_p \times H^T \times (H \times P_p \times H^T + R)^{-1}$

[0251] ⑤  $X = X_p + K \times (z_k - H \times X_p), P = P_p - K \times H \times P_p$

[0252] (o) 후 이 값이 초기 값이 되어 ②과정 반복 계산)

[0253]  $X = \begin{bmatrix} V_n \\ \hat{a}_n \end{bmatrix}$  ( $V_n$ : 오차 보정된 속도,  $\hat{a}_n$ : 오차 보정된 가속도)

[0254] (3) 이동거리 계산식

[0255]  $A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$  ( $\Delta t$ : 센서의 측정 간격)

[0256]  $z_k = V_n$  [(2)에서 계산한 오차 보정된 속도]

[0257] ① 초기값 가정

[0258]  $H = [0 \ 1]$

[0259]  $Q = 2 \times 2$  대각행렬 ([수식4]의 (6)과정을 통해 계산 가능)

[0260]  $R = k$  상수 ([수식4]의 (6)과정을 통해 계산 가능)

[0261]  $X = \begin{bmatrix} S_1 \\ V_1 \end{bmatrix}$  (상태변수 초기 값, 정지상태 였다면 0)

[0262]  $P = I$  ( $2 \times 2$  단위행렬, 오차공분산 초기 값)

[0263] ②  $X_p = A \times X$

[0264] ③  $P_p = A \times P \times A^T + Q$

[0265] ④  $K = P_p \times H^T \times (H \times P_p \times H^T + R)^{-1}$

[0266] ⑤  $X = X_p + K \times (z_k - H \times X_p), P = P_p - K \times H \times P_p$

[0267] (o) 후 이 값이 초기 값이 되어 ②과정 반복 계산)

[0268]  $X = \begin{bmatrix} S_n \\ \hat{V}_n \end{bmatrix}$  ( $S_n$ : 오차 보정된 이동거리,  $\hat{V}_n$ : 오차 보정된 속도)

[0269] 이처럼, 칼만필터를 적용하여 각 축방향으로의 이동거리 산출 동작은 각 축에 대한 가속도( $a_x, a_y, a_z$ )에 포함된 오차 성분을 제거하면서 이동 속도와 이동 거리를 계산하므로, 이중 적분 방식을 이용하여 산출된 이동거리 보다 좀더 정확한 이동거리가 산출된다.

[0270] 이와 같이, 글로벌 좌표계에서의 각 축방향 이동거리가 계산되면, 연산부(210)는 각 축(X축 Y축 및 Z축)방향에 대한 위치 정보 측정 장치(1000)의 상대좌표를 결정한다(S19).

[0271] 일반적으로 사진측량을 적용하여 해당 대상물의 특징점 좌표를 구하기 위해서 계산되는 촬영위치는 위치 정보 측정장치(1000)에 설치된 카메라 즉, 촬영부(110)의 렌즈 중심점이다.

- [0272] 예를 들어, 일반적인 안드로이드 기반 스마트폰의 기기 중심점과 렌즈 중심점과의 관계는 도 4의 (b)와 같다.
- [0273] 일반적으로 위치 정보 측정장치(1000)에 설치된 센서들(101-103)과 위치정보 획득부(120)의 측정 중심점은 기기 중심점인 경우가 대부분이다. 예를 들어, 안드로이드 기반의 스마트폰의 경우 센서의 측정 중심점과 GNSS 수신의 기준은 기기 중심점이 된다.
- [0274] 따라서, 기기 중심점을 렌즈 중심점으로 가정하여 계산하여도 약간의 오차가 발생할 뿐 큰 문제는 없지만, 본 예의 경우, 계산의 정확도를 높이기 위하여, 촬영 위치를 기기 중심점에서 렌즈 중심점으로 보정할 수 있다.
- [0275] 기기 중심점을 렌즈 중심점으로 보정하는 방법은 [수식 5]와 유사하다.
- [0276] 즉, 기기 중심점에서 측정되는 가속도(수식 5의  $a_0$ ,  $a_1$ ,  $a_2$ )가 기기 중심점에서부터 렌즈 중심점까지의 거리가 되고, 3차원 회전매트릭스를 적용하면 기기 중심점에서 렌즈 중심점까지의 글로벌 좌표계에서의 상대거리가 계산된다.
- [0277] 이는 다음 [수식9]과 같다.
- [0278] [수식9]
- $$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix} = R_\kappa \times R_\phi \times R_\omega \times \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix}$$
- [0279]
- [0280]  $-d_x, d_y, d_z$  : 기기 중심점에서 렌즈 중심점까지의 거리(도 4의 (b) 참조).
- [0281]  $-X_g, Y_g, Z_g$  : 기기 중심점에서 렌즈 중심점까지의 글로벌 좌표계에서의 상대거리)
- [0282] 또한, [수식6]과 같이 칼만 필터를 적용할 수 있다.
- [0283]  $d_x, d_y, d_z$  와 같은 기기 중심점에서 렌즈 중심점까지의 거리는 고정된 값으로서 저장장치(400)에 저장되어 있거나 서버(2000)를 통해 제공받을 수 있다.
- [0284] 기기 중심점에서 렌즈 중심점까지의 거리( $d_x, d_y, d_z$ )가 위치 정보 측정장치(1000)의 저장 매체 속에 저장되어 있는 경우, 연산부(210)는 해당 저장 매체 속에 저장되어 있는 데이터를 이용하여 별도의 연산 동작없이 원하는 거리( $d_x, d_y, d_z$ )를 획득하게 된다.
- [0285] 또한, 서버(2000)에 해당 데이터( $d_x, d_y, d_z$ )가 저장되어 있는 경우, 연산부(210)는 통신부(500)를 통해 서버(2000)로 해당 데이터( $d_x, d_y, d_z$ )를 요청하여 획득할 수 있다.
- [0286] 상보필터나 칼만필터를 통해서 계산된 위치 정보 측정장치(1000)의 자세 데이터 중에서 방위각( $\kappa$ )의 데이터는 자기장 센서(103)에 의해 측정된 자북 방위각이다.
- [0287] 일반적으로 평면 직교 좌표계의 경우, 도북 방위각(지도상의 북쪽)으로 좌표값을 계산하기 때문에 자기장 센서(103)의 자기장 데이터를 이용하여 산출한 자북 방위각( $\kappa$ )은 도북 방위각으로 변경되어야 평면 직교 좌표계에서의 정확한 좌표값을 계산할 수 있다.
- [0288] 또한, 촬영한 위치를 위도 및 경도 기반으로 결정하기 위해서는 진북을 기준으로 한 진북 방위각을 알아야 할 필요가 있다.
- [0289] 도북 방위각과 자북 방위각의 차이는 도자각이고, 진북 방위각과 자북 방위각의 차이는 자편각이며, 도북 방위각과 진북 방위각의 차이는 도편각이다.
- [0290] 도북은 자북에 도자각의 방향에 따라 더해주거나 빼줘서 구할 수 있다. 예를 들어, 대한민국의 도자각은 약 서편각  $6^\circ \sim 8^\circ$  로 나침반의 자침이 도북을 기준으로 했을 때 서쪽으로 약  $6^\circ \sim 8^\circ$  기울어져 있다는 것을 의미한다.
- [0291] 진북은 북극방향을 뜻하는 것으로 진북을 통과하는 자오선의 북쪽을 말한다.

[0292] 자편각은 진북방향과 자북방향 사이의 차각, 진북을 기준으로 하여 자북까지 측정된 각이다.

[0293] 자편각을 계산하는 수식의 한 예는 [수식10]과 같다.

[0294] [수식10]

[0295] (1) 구면 조화식

$$V = a \sum_{n=1}^{N_{\max}} (a/r)^{n+1} [g_n^m \cos(m\lambda) + h_n^m \sin(m\lambda)] P_n^m(\Psi)$$

[0296]

-  $a$  : 지구중심에서 측정 장소까지의 거리

[0298]

-  $r$  : 지구반지름

[0299]

-  $g_n^m$ ,  $h_n^m$  : 가우스 계수

[0300]

-  $\lambda$  : 측정위치 경도

[0301]

-  $\Psi$  : 측정위치 위도

[0302] -  $P_n^m(\Psi)$  : 위도를 변수로 한 르장다르 다항식(Legendre polynomial)

[0303]

-  $V$  : 자기장의 스칼라 포텐셜

[0304]

- IGRF(International Geomagnetic Reference Field: 국제 표준 지장기장)에서는 5년마다 가우스계수( $g_n^m$ ,  $h_n^m$ )를 결정하여 발표, 2015년의 경우  $N_{\max}=13$ 으로 가우스 계수를 발표 사용 중

[0305]

(2) 자편각( $d$ ) 계산

$$d = 2 \times \tan^{-1} \left( \frac{Y}{\sqrt{X^2 + Y^2 + X}} \right)$$

[0306]

$$(단, X = \frac{1}{r} \frac{\partial V}{\partial \Psi}, Y = \frac{1}{r \sin \Psi} \frac{\partial V}{\partial \lambda}, Z = \frac{\partial V}{\partial r})$$

[0307]

도편각은 TM 투영식(Transverse Mercator projection formula)을 통해 투영 중심점에서부터의 거리 차이를 각도로 변환하여 계산할 수 있는 것으로서, TM 투영식은 공개되어 있다.

[0309]

따라서, 본 예의 시스템은 자편각과 도편각을 계산할 수 있는 알고리즘을 포함하거나, 서버(2000)를 통해 도편각과 자편각을 제공받아, 측정 위치에 따른 도자각과 자편각을 반영하여 자북방위각이나 진북방위각으로 변경하여 계산할 수 있다.

[0310]

또한, 절대좌표를 취득할 수 없어 해당 지역의 도북방위각을 계산할 수 없는 경우, 자북방위각 기반의 상대좌표로 계산하여도, 좌표간의 관계는 충분히 계산할 수 있다.

[0311]

좌표간의 관계를 통해서 구해지는 값들, 예를 들어, 어떤 두 점 간의 높이, 거리, 경사 또는 면적과 같은 값은 자북방위각 기반으로 계산하여도 그 정확도에는 차이가 없다.

[0312]

위와 같이 [수식1] 내지 [수식10]까지의 수식들을 적용하여, 위치 정보 측정장치(1000)의 연산 및 제어 유닛(200)의 연산부(210)에서 영상 촬영부분(도 3의 P1)의 단계(S100)의 일부인 위치 정보 측정장치(1000)의 자세를 결정하는 방법(S14)과 각 축방향 이동거리 및 상대좌표를 결정하는 방법(S19)을 설명하였다.

[0313]

## 2) 영상촬영부분(P1)에 대한 알고리즘의 제1 예-실시간 처리를 위한 알고리즘 예

[0314]

다음, 도 6을 참고로 하여, 촬영부(110)에 의한 촬영 동작과 동시에 위치 정보 측정장치(1000)의 위치 및 자세를 실시간으로 산출하고 촬영 동작에 의한 영상 데이터의 획득 방법에 대하여 설명한다.

[0315]

도 6에 도시한 것처럼, 연산부(210)는 가속도 센서(101), 자이로 센서(102) 및 자기장 센서(103)로부터 인가되

는 신호를 판독하여, 가속도 데이터, 각속도 데이터 및 자기장 데이터로 이루어진 감지 데이터(DAT1)를 획득하고, 촬영부(110)에서 출력되는 신호를 판독하여 촬영부(110)의 동작에 따라 생성된 영상 데이터(DAT2)를 획득한다.

[0316] 또한, 연산부(210)는 위치정보 획득부(120)로부터 인가되는 신호를 판독하여 위치정보 획득부(120)에서 출력되는 절대좌표(DAT3)를 획득한다,

[0317] 따라서, 감지부(100)를 이용하여 감지 데이터(DAT1)가 얻어지면(S101), 이미 설명한 것처럼, 감지 데이터(DAT1)인 가속도 데이터, 각속도 데이터 및 자기장 데이터에는 노이즈 성분을 포함하고 있으므로, 연산부(210)는 노이즈제거필터를 통해 노이즈 성분을 제거한다(S102)

[0318] 본 예에서, 노이즈제거필터는 노이즈를 제거하기 위한 수학적 계산식이나 알고리즘을 의미한다.

[0319] 이때, 연산부(210)는 정지상태에서의 센서값을 측정하여 일정 표준편차 범위내의 값 이하를 무시하는 필터를 적용할 수 있다.

[0320] 예를 들어 2016년 12월 기준 일반적인 스마트폰, 즉 위치 정보 측정장치(1000)에 설치된 가속도 센서(101)의 정지상태의 표준편차는 기종에 따라 약간의 차이가 있지만 약  $\sigma = \pm 0.011\text{m/s}^2$  이다.  $\pm 3\sigma$ 의 범위내 값들, 즉  $-0.033\text{m/s}^2 \sim +0.033\text{m/s}^2$ 을 노이즈로 무시하여 0으로 계산하는 것이 가장 쉽게 구현할 수 있는 노이즈제거필터이다.  $3\sigma$ 의 경우 노이즈 신호가 표준 정규 분포를 이룬다고 가정할 때 발생 가능한 우연오차의 99.73%를 제거 할 수 있다.

[0321] 하지만 일정 영역 내의 값들을 오차로 가정하여 제거할 경우 자칫하면 의미 있는 값들도 제거될 수 있다. 그러나  $s = \frac{1}{2}at^2$  0.033 $\text{m/s}^2$ 의 가속도는  $a$ 의 공식에 의해 1초 동안 해당 가속도로 움직일 경우 약 1.65cm정도 움직이는 가속도로 일반적인 상황에서 충분히 무시 가능한 가속도라 할 수 있다.

[0322] 위와 같은 표준편차를 이용한 노이즈제거필터는 정지상태에서는 효율이 좋을지 모르나 측정값이 지속적으로 변하는 상태에서는 적용하기 어렵다.

[0323] 측정값이 지속적으로 변하는 상태에서 실시간으로 노이즈를 제거하기 위해서는 필터 구성이 복잡하지 않은 지수 가중이동평균필터를 적용하는 것이 용이하다. 지수가중이동평균필터는 다음 수식11로 표현할 수 있다.

[0324] [수식11]

$$\hat{x}_n = k \times \widehat{x_{n-1}} + (1 - k)x_n$$

[0326] ( $\hat{x}_n$ : 최획값,  $\widehat{x_{n-1}}$ : 이전단계의 최획값,  $k$ : 상수( $0 < k < 1$ ),  $x_n$ : 관측값)

[0327] 지수가중이동평균필터는 저주파필터(Low-pass filter) 중 하나로 저주파 신호는 통과시키고 고주파 신호는 걸러내는 필터로서, 최근의 측정값에는 높은 가중치를 두고, 오래된 측정값일수록 가중치를 낮게 주는 채귀형 필터이고, 상수  $k$  값을 적절히 조절하여 본 예의 시스템에 적합한 노이즈 제거가 가능하다.

[0328] 위치 정보 측정장치(1000)의 정지 상황에서는 연산부(210)는 표준편차를 이용한 노이즈 제거 필터를 적용하고 데이터가 급격히 변하는 상황에서는 지수가중이동평균필터 등을 적절히 적용하여 센서 노이즈를 1차로 제거해 줄 수 있다(S102).

[0329] 다음, 연산부(210)는 노이즈가 제거된 가속도 데이터, 각속도 데이터 및 자기장 데이터를 합성하여, 앞서 도 5를 참고로 하여 설명한 방법에 따라 위치 정보 측정장치(1000)의 최종 자세 데이터를 계산되고 이때 최종 자세 데이터는 해당 촬영 동작시 촬영부(110)[즉, 위치 정보 측정 장치(1000)]에 대한 자세 데이터로서 저장된다(S103).

[0330] 이때, 연산부(210)는 자신의 처리능력에 따라 상보필터(수식 3)나 칼만필터(수식 4)를 적용하여 위치 정보 측정 장치(1000)의 최종 자세 데이터(즉, 자세 데이터)를 얻을 수 있다

[0331] 예를 들어, 위치 정보 측정장치(1000)가 안드로이드 스마트폰의 경우는 실시간처리를 하려고 할 때 칼만필터 적용하면 시간지연오차 및 데이터 손실이 나타날 수 있으므로 상보필터를 적용하는 것이 유리하다.

[0332] 또는, 위치 정보 측정장치(1000)에서 자신의 자세 데이터를 직접 제공해주는 센서를 사용할 수 있다.

- [0333] 예를 들어 안드로이드 기반의 스마트폰은 가속도센서, 자이로센서 및 자기장 센서가 합성된 합성센서를 이용하여 간단하게 위치 정보 측정장치(1000)인 스마트폰 기기의 방위각, 고저각 및 회전각을 제공 받을 수 있다.
- [0334] 안드로이드 스마트폰의 합성센서는 JAVA 센서 호출명 Sensor.TYPE\_ORIENTATION과 Sensor.TYPE\_ROTATION\_VECTOR가 있으며, 구글에서는 Sensor.TYPE\_ROTATION\_VECTOR의 사용을 권장하고 있다. 위와 같은 합성센서를 사용하면 배열형태로 방위각, 고저각, 회전각을 얻을 수 있다.
- [0335] 하지만 이런 합성센서를 이용할 경우, 실시간 처리를 위하여 연산부(210)는 성능이 낮은 상보필터를 이용하는 경우가 일반적이고, 각 제조사 별로 합성센서의 출력 데이터의 값에 큰 오차가 발생한다.
- [0336] 하지만, 합성센서를 제공하지 않는 위치 정보 측정장치(1000)의 경우, 최종 자세 데이터를 계산하기 위해서는 도 5를 참고로 하여 설명한 방법을 이용하는 것이 유리하다.
- [0337] 본 예의 경우, 연산부(210)는 위치 정보 측정장치(1000)의 완전정지상태에서의 감지 데이터, 즉 가속도 데이터, 각속도 데이터 및 자기장 데이터의 값들의 평균을 사전에 측정 후 저장하여, 측정값이 0이지만 결과로 얻은 신호 값은 0이 아닌 경우인 제로오프셋의 상태, 즉 영점오차를 보정해준다(S104).
- [0338] 예를 들어, 위치 정보 측정장치(1000)가 완전 정지상태일 때 가속도 센서(101)의 X축, Y축 및 Z축 중 어느 한 축의 평균값이  $0.1\text{m/s}^2$  이었다면, 이후 이 그 축에서 측정된 가속도 센서(101)의 결과 값에  $-0.1\text{m/s}^2$ 를 하여 영점오차를 보정해 줄 수 있다.
- [0339] 하지만, 대안적인 예에서, 이러한 영점오차 보정 동작은 생략될 수 있다.
- [0340] 이와 같이, 위치 정보 측정장치(1000)에 대한 영점오차 보정 동작인 2차 보정 동작이 완료된 후, 촬영부(110)에 의한 1차 사진 촬영 동작이 이루어져 해당 대상물에 대한 영상 데이터(DAT2)가 연산부(210)로 입력되면, 연산부(210)는 입력되는 영상 데이터(DAT1)를 메모리(220) 혹은 저장부(400)에 저장한다(S121).
- [0341] 또한, 연산부(210)는 촬영부(110)에 의한 촬영 동작이 행해질 때, 위치정보 획득부(120)에서 출력되는 절대좌표(DAT3)를 이용하여 1차 사진촬영 시의 위치인 1차 사진촬영위치의 절대좌표를 취득 후 저장한다(S111).
- [0342] 이때, 1차 사진촬영 순간의 위치 정보 측정장치(1000)의 자세 데이터를 메모리(220) 혹은 저장부(400)에 저장하며 또한 초기 상대좌표(즉 초기 위치 데이터)의 값을  $(0,0,0)$ 으로 정하여 메모리(220) 혹은 저장부(400)에 저장한다(S105).
- [0343] 본 예의 위치 정보 측정장치(1000)가 스마트폰과 같은 기기에 설치되어 있는 경우, 1차 사진 촬영과 2차 사진 촬영 사이에 사용자인 활용자의 움직이는 걸음 폭은 보통 성인인 사람이 한 팔로 휘두르는 범위 내이거나 사람이 몇步자국 정도 이동하는 거리인  $1\sim2\text{m}$ 정도일 것이다.
- [0344] 본 예의 경우 이  $1\sim2\text{m}$ 의 움직임의 위치를 정확하게 구하는 것이 본 시스템의 가장 중요한 과정 중에 하나이다.
- [0345] 따라서, 연산부(210)는 대상물에 대한 2차 사진촬영 전까지 이미 설명한 도 5의 이동거리 계산 방식(도 5의 S19 참조)을 이용하여 글로벌 좌표계에서의 각 축(X축, Y축 및 Z축) 방향의 가속도를 계산하고(S106) 실시간 처리에 적합한 이중적분 방법인 [수식7]로 위치 정보 측정장치(1000)의 각 축(X축, Y축 및 Z축) 방향으로의 이동거리를 계산하고 각 축 방향의 이동거리만큼 이동한 위치에 대한 좌표(즉 상대 좌표)를 위치 데이터로서 산출하여 저장한다(S107).
- [0346] 이때, 연산부(210)가 이러한 위치 정보 측정장치(1000)의 실시간 이동거리 계산을 [수식8]과 같은 칼만필터를 적용하여 실행한다면 [수식7] 보다 정확한 결과 값을 얻을 수 있다.
- [0347] 1차 사진촬영 후, 2차 사진촬영이 행해져 해당하는 영상 데이터가 연산부(210)로 입력되면(S122), 연산부(210)는 이미 설명한 것처럼 2차 사진촬영이 행해질 때 위치정보 획득부(120)를 이용하여 2차 사진촬영이 이루어진 위치 정보 측정장치(1000)의 위치 즉, 2차 사진촬영위치에 대한 절대좌표를 취득한 후 저장하고(S112), 2차 사진촬영 순간의 위치 정보 측정장치(1000)의 자세 데이터(즉, 최종 자세 데이터), 이동거리 및 상대좌표인 위치 데이터를 메모리(220) 혹은 저장부(400)에 저장한다(S108).
- [0348] 이때, 단계(S106)에서 산출된 가속도를 이중 적분하여 2차 사진촬영 순간의 위치 정보 측정장치(1000)의 이동거리를 계산하는 방식은 오차가 발생할 수 있으므로, 단계(S112)에서 구한 2차 사진촬영위치의 절대좌표를 취득하여 1차 사진촬영위치의 절대좌표에서 2차 사진촬영위치의 절대좌표를 빼면 글로벌 좌표계에서의 각 축방향별 이동거리를 계산할 수 있다.

- [0349] 다음, 연산부(210)는 1차 사진촬영위치의 절대좌표와 2차 사진촬영위치의 절대좌표를 취득하여 산출한 각 축방향별 이동거리 값과 단계(S106)에서 산출된 가속도를 이중적분하여 산출된 이동거리의 값을 비교하여 오차를 보정한다(S109).
- [0350] 이러한 연산부(210)의 위치정보 획득부(120)를 통해 구한 절대좌표를 이용한 오차 보정 동작은 생략될 수 있다.
- [0351] 일반적으로 GNSS를 통해 획득한 절대좌표는 EPSG:4326 기반의 경위도 좌표계로 결과물이 나타난다. 이런 경우도 좌표계를 평면직각좌표계로 변환하여 사용하는 것이 오차 보정과 계산이 편리해 진다.
- [0352] 예를 들어 본 시스템의 측정위치가 대한민국일 경우, 위도가 북위  $37^{\circ} \sim 32^{\circ}$ 이고, 경도가 동경  $124^{\circ} \sim 126^{\circ}$ 인 경우는 서부원점 기반의 EPSG:5185이고, 동경  $126^{\circ} \sim 128^{\circ}$ 인 경우는 중부원점 기반의 EPSG:5186이며, 동경  $128^{\circ} \sim 130^{\circ}$ 인 경우는 동부원점 기반의 EPSG:5187이고, 동경  $130^{\circ} \sim 132^{\circ}$ 인 경우는 동해원점 기반의 EPSG:5188로 각각 획득된 절대좌표를 변환한 평면직각좌표계 사용이 가능하다.
- [0353] 대한민국 외에 다른 국가에서 본 시스템을 사용할 때에는 획득된 절대좌표는 국제 표준 평면직각좌표계인 UTM좌표계(Universal Transverse Mercator coordinate system)로 변환하여 사용되거나 또는 획득된 절대좌표에 대한 해당 경도와 위도를 분석하여 해당 경위와 위도에 속하는 나라가 사용하는 평면직각좌표계를 EPSG(European Petroleum survey Group)셋에서 찾아 변환하여 사용하면 된다.
- [0354] EPSG셋을 이용한 좌표계 변환을 지원해주는 많은 오픈 소스들이 있고, TM투영식(Transverse Mercator projection formula), TM역투영식(Transverse Mercator reverse projection formula) 역시 대중에게 공개되어 있으므로 생략한다.
- [0355] 여기서, TM투영식은 위도와 경도로 표현되는 구면좌표계를 XY평면직면좌표계상으로 나타내는 방법 중에 하나인 횡축 메카르도(TM) 방법(원통을 적도와 나란하게 옆으로 눕혀 지구의 표면을 투영한 방법)에 의해 위도와 경도로 표현된 구면좌표를 XY 2차원 평면좌표로 표현하는 방법이다.
- [0356] 또한, TM역투영식은 TM투영에 의해 XY평면직교좌표계상의 X와 Y좌표를 위도와 경도로 표현하는 구면좌표계로 변환하는 방법이다.
- [0357] 여기서, EPSG란 European Petroleum Survey Group의 약자로 IOGP(International Association of Oil & Gas Producers)에서 제공하는 글로벌 좌표계의 변환을 위해 Geodetic Parameter Set 데이터 베이스로 예를 들어, EPSG:4326은 구글 어스에서 사용되는 WGS84타원체 기반의 경위도 좌표계를 뜻하고 EPSG:5186은 GRS80기반의 대한민국 중부원점의 평면 직교좌표계를 뜻한다. EPSG 데이터 셋을 이용하면 글로벌 좌표계에서 각 좌표계간의 변환이 용이하며, Java의 경우 EPSG:4326, EPSG:3857, EPSG:900913, KATECH등의 글로벌 좌표계의 사용 및 변환을 기본 지원하고 있으며, 간단한 파라미터의 추가로 EPSG:5185(대한민국 서부원점기준의 평면직교좌표계), EPSG:5186, EPSG:5187(대한민국 동부원점 기준의 평면직교좌표계), EPSG:5188(대한민국 동해원점 기준의 평면직교좌표계)상호간의 변화가 가능하다.
- [0358] 본 예의 경우, 해당 대상물에 대한 거리와 좌표 등을 산출하기 위해서는 2번의 사진 촬영이 이루어지거나, 동작의 정확도를 향상시키기 위하여 해당 대상물에 대한 추가 사진 촬영이 이루어져 3번 이상의 사진 촬영이 이루어질 수 있다.
- [0359] 따라서, 촬영부(110)이 동작으로 인해 추가 사진 촬영 동작이 이루어지면(S130), 연산부(210)는 단계(S131)로 넘어가 추가 촬영된 3번째의 사진 촬영 동작에 따른 3차 영상 데이터, 위치정보 획득부(120)를 이용하여 획득한 3차 사진 촬영위치에 대한 절대좌표, 3차 사진 촬영 시 촬영부(110)의 자세 데이터 및 상대좌표(즉, 위치 데이터)가 산출되어 저장된다.
- [0360] 이와 같이, 3번 이상의 사진 촬영 동작이 이루어져 해당 대상물에 대한 3개 이상의 영상 데이터와 사진 촬영 시 얻어지는 추가적인 데이터, 즉 절대좌표, 자세 데이터 및 상대 좌표(즉, 위치 데이터) 등이 구비한 3개 이상의 영상 데이터군이 얻어지면, 연산부(210)는 3개 이상의 영상 데이터군 중에서 두 개의 영상 데이터군을 선택하게 된다. 이러한 연산부(210)의 동작은 다음에 상세히 설명한다.
- [0361] 하지만, 2번의 사진 촬영 이후, 추가 사진촬영이 설정 시간 내에 이루어지지 않는 경우, 연산부(210)는, 산출된 위치 정보 촬영장치(1000)의 절대 좌표, 위치 데이터( $x, y, z$ ) 및 자세 데이터( $\omega, \phi, \psi$ )를 구비한 2개의 영상 데이터군을 얻게 된다(S150).
- [0362] 즉, 한 개의 영상 데이터군은 1차 사진 촬영 동작에 관련된 영상 데이터군이고 나머지 한 개는 2차 사진 촬영

동작에 관련된 영상 데이터군이다.

[0363] 본 예의 위치 정보 측정장치(1000)에 위치정보 획득부(120)가 구비되지 않거나 GNSS 신호를 수신할 수 없는 경우, 연산부(210)는 촬영부(110)의 위치 좌표, 즉 위치 정보 측정장치(1000)의 위치 좌표는 상대좌표로 계산되고, 위치정보 획득부(120)를 이용한 위치 오차 보정 동작(S109)은 생략된다.

### 3) 영상촬영부분(P1)에 대한 알고리즘의 제2 예-오차를 최소화 할 수 있는 알고리즘 예

[0365] 다음, 도 7을 참고로 하여, 사용자의 복수 번의 사진 촬영 동작에 따른 복수 개의 영상 데이터군을 획득하는 연산부(210)의 다른 예를 설명한다.

[0366] 도 6을 참고로 하여 설명한 영상 데이터 획득 방법은 사진 촬영과 동시에 위치 정보 측정장치(1000)의 위치와 자세를 실시간으로 산출하여 신속한 결과를 획득할 수 있다.

[0367] 하지만, 연산속도가 상대적으로 느린 위치 정보 측정장치(1000)의 경우 시간 지연 오차나 데이터 손실이 나타날 수 있고, 이런 경우 연산의 정확도가 떨어질 수 있다.

[0368] 예를 들어, 위치 정보 측정장치(1000)가 안드로이드 기반의 스마트폰인 경우, JVM(Java Virtual Machine) 기반으로 동작이 실행되므로 연산 속도가 늦어 오차가 발생할 수 있다.

[0369] 도 7의 방법을 이용하여 영상 데이터군을 획득할 경우, 도 6의 방법에 비해 오차를 좀 더 감소시켜 동작의 정확도를 향상시킬 수 있다.

[0370] 도 7에 도시한 것처럼, 역시 감지부(100), 위치정보 획득부(120) 및 촬영부(110)를 각각 이용하여 위치 정보 측정장치(1000)[즉, 촬영부(110)]의 위치와 자세를 결정할 수 있는 감지 데이터(DAT1), 대상물의 영상 데이터(DAT2) 및 절대좌표(DAT3)를 얻는다.

[0371] 따라서, 연산부(210)는 가속도 센서(101), 자이로 센서(102) 및 자기장 센서(103)로부터 인가되는 신호를 판독하여 감지 데이터(DAT1)를 획득한다(S201).

[0372] 본 예의 경우, 연산부(210)는 획득된 감지 데이터(DAT1)에 대한 별도의 처리 동작을 실시하지 않고 획득된 감지 데이터인 센서 로우 데이터(DAT1)를 일정 시간 간격으로, 예를 들어 위치 정보 측정장치(1000)의 감지기 설정에 따라 20ms(0.02초)에서 200ms(0.2초) 간격으로 로우 데이터(raw data) 형태로 메모리(220) 혹은 저장부(400)에 저장한다(S202).

[0373] 이 시간간격은 가속도 센서(101), 자이로 센서(102), 자기장 센서(103)의 성능에 따라 보다 짧은 시간 간격으로 측정할 수 있다.

[0374] 이때, 연산부(210)는 1차 사진 촬영이 있기 전까지는 약 1~2초 정도 전의 획득된 로우 데이터 형태의 감지 데이터(DAT1)를 메모리(220)에 저장하면서 입력 대기를 기다린다.

[0375] 사용자가 촬영부(110)를 이용하여 해당 대상물에 대한 1차 사진촬영이 이루어지면(S221), 연산부(210)는 1차 사진 촬영 동작으로 생성된 영상 데이터(DAT2)를 메모리(220) 혹은 저장부(400)에 저장하고, 1차 사진촬영을 할 때 위치정보 획득부(120)에서 출력되는 절대 좌표를 이용하여 1차 사진촬영 시의 위치 정보 측정장치(1000)의 절대좌표(DAT3)를 취득 하여 메모리(220) 혹은 저장부(400)에 저장하고(S211), 일정시간 간격으로 저장되고 있는 감지 데이터(DAT1)에 1차 사진 촬영 시의 시간을 표기하여 저장한다(S203).

[0376] 그런 다음, 다시 사용자에 의해 2차 촬영 동작이 행해지면(S222), 연산부(210)는 2차 사진 촬영 동작으로 생성된 영상 데이터(DAT2)를 메모리(220) 혹은 저장부(400)에 저장하며 역시 위치정보 획득부(120)를 이용하여 2차 사진촬영 위치에 대한 절대좌표(DAT3)를 취득 후 저장하고(S212), 역시 일정 시간 간격으로 저장되고 있는 센서 로우 데이터(DAT1)에 2차 촬영 시의 시간을 표기하여 저장한다(S204).

[0377] 이처럼, 해당 대상물에 대한 두 번의 사진 촬영 동작이 이루어진 후, 연산부(210)는 사용자에 의해 추가 사진 촬영 동작이 이루어졌는지 판단하고(S230), 추가 사진 촬영 동작(즉, 3차 사진 촬영 동작)이 이루어지면(S230), 1차 및 2차 사진 촬영 시와 동일하게, 3차 사진 촬영 생성된 영상 데이터(DAT2)를 메모리(220) 혹은 저장부(400)에 저장하고 3차 사진 촬영 시의 사진 촬영 위치에 대한 절대좌표(DAT3)를 획득하여 저장하며, 일정시간 간격으로 저장되고 있는 로우 데이터 형태의 센서 로우 데이터(DAT1)에 3차 촬영 순간을 표기한다.(S231).

[0378] 이와 같이, 3번 이상의 사진 촬영이 이루어지면, 매 사진 촬영이 이루어질 때마다 해당 영상 데이터(DAT2)를 생성해 메모리(220) 혹은 저장부(400)에 저장하고, 사진 촬영시의 사진 촬영에 위치에 대한 절대좌표(DAT3)를 취

득하여 저장하며, 일정 시간 간격으로 저장되고 있는 로우 데이터 형태의 감지 데이터(DAT1)에 매 촬영 시간을 표기한다.

[0379] 2번 사진 촬영 후 추가적인 사진 촬영 동작이 이루어지지 않으면(S230), 연산부(210)는 일정시간 간격으로 기록되고 있던 로우 데이터 형태의 감지 데이터(DAT1)의 기록을 멈추고 기록된 감지데이터(DAT1)의 처리를 실시한다(S240)

[0380] 즉, 연산부(210)는 메모리(220) 혹은 저장부(400)에 저장되어 있는 각 사진 촬영 당시의 감지 데이터(DAT1)의 데이터 값(즉, 센서 로우 데이터 값)들을 노이즈 제거필터를 이용 노이즈를 제거하여 해당 감지 데이터에 대한 1차 오차 보정을 실시한다(S242).

[0381] 노이즈 제거를 위해 사용되는 필터는, 앞서 설명한 [수식11]에 기재된 지수가중이동평균을 적용할 수 있으나, 실시간 처리가 아니기 때문에 효과적인 노이즈 제거필터 중에 하나인 1차원 가우시안 필터를 적용할 수 있다.

[0382] 노이즈 제거를 위한 1차원 가우시안 필터는 다음 수식12과 같다.

[0383] [수식12]

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

[0384]  $\sigma = 1$  일 경우 필터

n-3	n-2	n-1	n	n+1	n+2	n+3
0.004	0.054	0.242	0.399	0.242	0.054	0.004

[0385]  $\sigma = 0.5$  일 경우 필터

n-1	n	n+1
0.108	0.798	0.108

[0386]  $\sigma = 0.5$  일 때 노이즈가 제거된 최적값 계산식

$$\hat{x}_n = 0.108 x_{n-1} + 0.798 x_n + 0.108 x_{n+1} (\hat{x}_n: \text{최적값}, x_n: \text{관측값})$$

[0387] 1차원 가우시안 필터의 경우 현재 센서값의 최적값( $\hat{x}_n$ )을 구하기 위해 과거의 데이터 값( $x_{n-1}$ )과 미래의 데이터 값( $x_{n+1}$ )을 조합하여 구하게 된다. 실시간 처리에서는 미래의 데이터 값을 얻을 수 없기 때문에 실시간 처리에서는 1차원 가우시안 필터를 적용하는 것은 어렵다. 1차원 가우시안 필터의 중요인자  $\sigma$ 의 경우 센서의 오차 범위를 고려하여 반복적인 실험이나 센서 측정간격, 정지상태에서의 센서값의 표준편차 등을 이용하여 결정해 줄 수 있다.

[0388] 노이즈 제거를 위해 연산부(210)는 앞서 도 6의 단계(S102)에서 설명한 표준편차 값을 이용한 필터, 지수가중이동평균필터(수식11) 또는 1차원 가우시안 필터 등을 적절히 사용하여 해당 감지 데이터의 노이즈 값을 제거해 준다(S242).

[0389] 다음, 연산부(210)는 노이즈가 제거되고 매 촬영 시간이 표기된 감지 데이터를 도 5를 참고하여 설명한 것과 같은 방식으로 처리하여 각 사진 촬영 순간의 자세 데이터( $\omega, \phi, \kappa$ )를 계산한다(S243).

[0390] 본 예의 경우, 감지 데이터에 대한 연산이 후처리로 진행되기 때문에 상보필터(수식3)보다는 칼만필터(수식4)를 적용하여 위치 정보 측정장치(1000)의 최종 자세 데이터를 얻는 것이 보다 정확한 결과를 얻을 수 있다.

[0391] 다음, 연산부(210)는 도 6의 경우와 같이 위치 정보 측정장치(1000)의 완전정지 상태에서의 가속도 데이터, 각 속도 데이터 및 자기장 데이터의 값들의 평균을 사전에 저장하여, 즉 영점오차를 보정해 2차 오차 보정을 실시한다(S244). 이러한 영점 오차 보정 동작은 생략 가능하다.

[0392] 다음, 연산부(210)는 각 사진 촬영 사이의 X, Y 및 Z축방향 가속도를 회전변환 일반식인 [수식5]나 칼만필터 적용식인 [수식6]의 방법으로 계산한다(S245). 이때, 보다 정확한 가속도값을 얻기 위해서 칼만필터 적용식인 [수

식6]의 방법을 이용하여 각 축방향 가속도를 계산하는 것이 유리하다.

[0397] 본 예에서 대상물의 거리측량이나 좌표결정의 정확도에 가장 큰 영향을 미치는 부분은 바로 사진 촬영 위치의 정확도이다.

[0398] 대상물 촬영 시 대상물의 상대위치를 정확하게 하기 위해서는 계속해서 변화하는 위치 정보 측정장치(1000)의 자세와 가속도를 적절히 처리하여야 하는데, 실시간 처리 방식에서는 데이터의 전체 양상을 비교 분석하는 것이 어렵다.

[0399] 예를 들어, 자이로 센서(102)나 자기장 센서(103)의 변화는 거의 없고 가속도 센서(101)의 감지값만이 큰 폭으로 변하고 있다면, 이 상황은 위치 정보 측정장치(1000)가 회전하지 않고 이동 중이라는 뜻이다.

[0400] 또한, 가속도 센서(101), 자이로 센서(102) 및 자기장 센서(103)의 감지값이 모두 변하고 있으나 가속도 센서(101)에서 측정되는 가속도 크기의 절대값이 지구 중력가속도와 거의 비슷하다면 위치 정보 측정장치(1000)는 정지상태에서 회전 중이라는 것을 판정할 수 있다.

[0401] 가속도 센서(101)에서 측정되는 가속도의 크기가 지구중력가속도보다 크다면, 측정된 가속도의 크기와 지구중력가속도의 크기의 차이만큼이 위치 정보 측정장치(1000)의 이동에 사용되는 가속도일 것이다.

[0402] 따라서, 연산부(210)는 2차 오차 보정 후, 다음 [수식13]를 이용하여 감지된 가속도 값을 비교하여 이동 상태를 판정하고 이동에 사용되는 각 축 방향의 가속도를 구할 수 있다(S245).

[0403] [수식13]

[0404]  $a_0, a_1, a_2$ : 가속도 센서에서 측정되는 가속도 값

$$a_{all} = \sqrt{a_0^2 + a_1^2 + a_2^2} \approx g \Rightarrow \text{정지상태 } (g: \text{중력가속도})$$

[0406]  $a_{move} = a_{all} - g$  ( $a_{move}$ : 이동에 사용되고 있는 가속도)

[0407] 다음, 연산부(210)는 앞서 [수식6]을 통해 구한 글로벌좌표계의 각 축방향 가속도와 비교 분석하여 매 촬영 시의 각 축방향의 이동거리를 산출한다(S246).

[0408] 이때, 연산부(210)는 가속도값에 오차를 최소화 한 후 가속도를 이중적분하여 이동거리를 구하는 [수식7]이나 가속도를 칼만필터로 적용하여 이동거리를 계산하는 [수식8]를 이용하여 기기의 글로벌 좌표계에서의 각 축방향 이동거리를 계산할 수 있다.

[0409] 오차를 보정하고 정확한 계산을 위해서는 칼만필터를 이용한 [수식8]를 적용하여 위치 정보 측정장치(1000)의 각 축방향 이동거리를 계산하는 것이 유리하다.

[0410] 다음, 연산부(210)는 사진 촬영 시 획득한 절대좌표 데이터와 가속도를 이용하여 산출한 사진 촬영 위치에 대한 위치 오차를 보정한다(S247). 이러한 위치 보정 동작 역시 필요에 따라 생략된다.

[0411] 이러한 방식을 통해 감지 데이터(DAT1)에 대한 처리 동작(S240)이 완료되면, 연산부(210)는 위치 정보 촬영장치(1000)의 위치 데이터( $x, y, z$ )와 자세 데이터( $\omega, \phi, \kappa$ )를 구비한 2개 이상의 영상 데이터군을 획득한다(S250).

#### 4) 영상촬영부분의 사용자 제어 화면

[0413] 도 8은 사용자에 의한 사진 촬영 동작이 이루어질 때의 출력부(300)의 화면으로서, (a)는 1차 사진 촬영 시 출력부(300)에 표시되는 화면이고, (b)는 2차 사진 촬영 시 출력부(300)에 표시되는 화면이다.

[0414] 도 8에 도시된 것처럼, 출력부(300)에 표시되는 사용자 제어 화면은 위치 정보 측정장치(1000)에 설치된 촬영부(110)를 통해 표시되는 영상을 실시간으로 출력해주는 디스플레이 화면(U1-1, U1-2)에는 현재 촬영되는 사진의 번호(U2-1, U2-2), 촬영하고자 하는 대상물을 쉽게 시준할 수 있도록 도와주는 반투명한 십자선(U3), 촬영버튼(U4), 촬영부(110)의 해상도와 노출을 조정하는 동작, 저장부(400)에 저장되어 있는 파일을 불러오는 동작, 감지부(100)의 영점을 조정하기 위해 정지상태의 센서값을 측정하는 작업, 태양, 별, 또는 달을 촬영하여 본 시스템의 절대좌표를 취득할 수 있는 동작 및 기타 설정 등을 조정하는 메뉴를 호출하는 메뉴버튼(U5), 촬영완료 후 영상처리부분을 실행시키는 영상처리버튼(U6-1, U6-2) 등이 표시된다.

- [0415] 사용자는 촬영부(110)가 실시간으로 보여주고 있는 디스플레이 화면(U1-1, U1-2)을 보면서 측정을 원하는 대상물을 시준 후 촬영버튼(U4)을 눌러 사진 촬영을 실시한다.
- [0416] 사용자가 1차 사진 촬영 후 위치 정보 측정장치(1000)를 약 1m 정도 이동시킨 후 2차 사진 촬영을 하면 출력부(300)를 통해 사용자에게 2차 촬영(U2-2)이라는 정보를 제공해주고, 이 경우에는 1차 사진 촬영 때 활성화 되지 않았던 영상처리 버튼(U6-1)이 활성화(U6-2)된다. 사용자가 추가 촬영을 원하면 계속 촬영버튼(U4)을 눌러 촬영이 가능하다.
- [0417] 사용자가 활성화된 영상처리 버튼(U6-2)을 선택하면 촬영을 멈추고 현재까지의 사진 촬영으로 획득된 영상 데이터, 위치 정보 측정장치(1000)의 자세 데이터 및 위치 데이터를 가지고 영상 처리 동작이 이루어질 수 있도록 한다.
- [0418] 도 8에 도시된 사용자 제어화면은 본 발명의 한 예로서 화면 구성이나 버튼 구성, 버튼 모양 등은 다양하게 조합 및 변형될 수 있다.
- [0419] 이러한 사용자 제어 화면을 위한 영상 데이터는 이미 메모리(220) 혹은 저장부(400)에 저장되어 있고, 연산부(210)의 제어 동작에 따라 해당 시기에 해당하는 영상 데이터가 출력부(300)로 출력되어 촬영 동작에 적절한 영상이나 버튼 등이 출력될 수 있도록 한다.
- [0420] 2. 영상처리부분(P2)
- [0421] 1) 본 예에서 촬영되는 영상의 특징
- [0422] 본 예에서, 해당 대상물의 위치 정보를 취득하기 위해서는 일단 위치정보를 취득하고자 하는 대상물에 대한 두 장 이상의 사진 촬영 동작을 실시해야 한다.
- [0423] 이때, 사진 촬영 동작으로 획득된 영상(즉, 사진 영상)들은 사진면적의 약 70%이상 중복되어 촬영된 경우가, 그렇지 않는 경우보다 정확한 측량이 가능하다(도 8의 U1-1, U1-2참조). 즉, 촬영된 사진 영상의 70% 이상이 서로 중첩된 영역인 것이 좋다.
- [0424] 또한, 짧은 시간 동안 2장 이상의 사진 촬영이 이루어지기 때문에 카메라인 촬영부(110)의 노출 정도, 셔터속도 및 ISO감도 등이 거의 동일한 상태에서 촬영이 이루어진다.
- [0425] 따라서, 사진 촬영 동작으로 얻게 되는 대상물의 영상 데이터의 축적의 변화가 거의 없고, 회전변화도 거의 없으며, 색상차도 거의 없고, 촬영부분의 약 70%이상은 동일 부분의 촬영이 대부분일 것이다.
- [0426] 2) 영상처리부분(P2)의 개략적 알고리즘
- [0427] 다음, 도 9를 참고로 하여, 사진 촬영 동작에 의해 획득된 영상 데이터를 처리하는 동작(도 3의 P2 부분)에 대하여 설명한다.
- [0428] 도 9에 도시한 것처럼, 연산부(210)에 의해 행해지는 영상 데이터 처리 동작은 크게 7단계로 나누었다.
- [0429] 즉, 연산부(210)는 영상 처리 동작을 위한 2장 이상의 영상에 대한 영상 데이터(이하, 2장 이상의 영상에 대한 영상 데이터를 '2장 이상의 영상 데이터'라 함)를 입력받는 단계(S301), 영상처리 속도를 높이기 위해 2장 이상의 영상 데이터를 매칭하는 매칭 기준을 계산하는 단계(S310), 2장 이상의 영상 데이터 중 하나(예, 1차 촬영 시 영상 데이터)인 기준 영상 데이터에서 특징점 가능영역을 선택하는 단계(S330), 특징점 가능영역에서 특징점을 검출하는 단계(S350), 검출된 특징점을 2장 이상의 영상 데이터 중 나머지 다른 영상 데이터에 매칭하기 위해 나머지 다른 영상 데이터에 특징점 기술자를 적용하는 단계(S370), 검출된 특징점을 나머지 다른 영상 데이터에서 찾아 검출된 특징점을 나머지 다른 영상 데이터에 매칭하는 단계(S380), 그리고 매칭이 완료되어 2장 이상의 영상 데이터에서 특징점들의 영상좌표를 취득하는 단계(S400)를 구비한다.
- [0430] 영상 좌표란 영상 처리(즉, 영상 데이터 처리)를 용이하게 하기 위해 음(-)의 값을 없앤 좌표체계로서, 일반적으로 행과 열의 행렬 구조로 배열되어 있고 해당하는 픽셀값(즉, 계조값)을 갖는 복수의 픽셀(pixel)로 이루어져 있는 영상 데이터에서의 좌측 최상단 픽셀을 (0,0)으로 하고 우측(즉, 행 방향)으로 이동하는 축을 x축(즉, 행 방향), 아래(즉, 행 방향)로 내려가는 축을 y축(즉, 열 방향)으로 하는 좌표체계에서 취득하는 좌표를 뜻이다(도 35참조). 이러한 영상 좌표는 측량 부분에서 사용하는 사진주점 기준의 사진좌표와는 좌표 중심점이 다르다.
- [0431] 사진좌표는 사진측량학에서 해당 대상물의 특징점 좌표를 파악하기 위한 좌표체계로서 사진의 주점(사진의 중심

점으로 렌즈중심으로부터 사진 면에 내린 수선의 발, 렌즈의 광축과 사진 면과의 교점)을 중심으로 하여, x축은 우측으로 이동하면 +, 좌측으로 이동하면 -, y축은 상단으로 이동하면 +, 하단으로 이동하면- 인 좌표체계에서 취득한 좌표를 뜻한다.

[0432] 사진 주점(principal point)은 사진의 중심점으로 렌즈중심으로부터 사진면에 내린 수선의 발, 렌즈의 광축과 사진면과의 교점이다.

[0433] 영상처리부분의 각 처리단계별 동작은 이후 도면을 통해 설명한다.

[0434] 영상처리부분의 설명을 명확히 하기 위해 2장의 영상을 기준으로 영상 데이터를 처리하는 알고리즘을 설명하고, 2장의 영상 데이터를 이용하여 영상처리 하는 설명이 끝나면, 추가 실시된 사진 촬영으로 인해 생성된 영상들의 영상처리 방법에 대하여 설명하도록 한다.

### 3) 영상데이터 매칭기준(도 9의 S310)

[0436] 도 10은 도 9의 영상 데이터의 매칭 기준 단계(도 9의 S310)에서의 매칭 기준의 개념을 설명하기 위한 개념도이다.

[0437] 본 예에서, 영상 데이터 매칭 기준이란 어느 한 지점의 영상(즉, 기준 영상)에 대한 영상 데이터인 기준 영상 데이터와 매칭될 수 있는 다른 영상(예, 매칭 영상)의 영상 데이터인 매칭 영상 데이터간의 개략적인 좌표 차이를 미리 계산하는 것을 말한다.

[0438] 예를 들어, 도 10의 (a)에 도시한 영상(예, 기준 영상)에서 어느 한 특징점(I1)의 영상 좌표(즉, 행과 열 방향으로의 픽셀 위치)가 (3826, 356)이었다면 도 10의 (b)의 영상(예, 매칭 영상) 전체에서 해당 특징점(I1)과 같은 부분을 찾는 것이 아니라, (b)의 영상(예, 매칭 영상)에서 영상 좌표 ( $3226 \pm 200$ ,  $356 \pm 200$ )에 속하는 영역(I2)에 대해서 특징점 검색을 하여 (a)의 영상(즉, 기준영상)의 특징점(I1)과 매칭되는 부분을 찾는 것이다.

[0439] 즉, 도 10의 (a)에 도시한 기준 영상의 어느 특징점(I1)은 도 10의 (b) 매칭 영상에서 x축 방향으로 -600픽셀 움직이고 y축 방향으로 0픽셀 움직이면 즉, A영상의 영상좌표 (3826,356) 부분을 일치시키기 위해서 B영상의 픽셀에서  $(3826-600)=3226$ 이고  $(356-0)=356$ 하면 거의 일치한다는 것을 사전에 계산하는 것이다.

[0440] 여기서 (-)는 왼쪽 행 방향 또는 위쪽 열 방향으로 이동하여 해당 픽셀의 좌표값이 감소하는 것을 의미하며, (+)는 오른쪽 행 방향 또는 아래쪽 열 방향으로 이동하여 해당 픽셀의 좌표값이 증가하는 것을 의미한다.

[0441] 이런 전처리 동작을 통해서, 각 특징점 별로 영상 전체에 대해서 계산할 필요가 없기 때문에 영상 처리 속도가 빨라지고, 매칭 동작 시 오동작을 최소화 할 수 있다.

[0442] 먼저, 영상 데이터 매칭기준의 알고리즘을 설명하기에 앞서 본 알고리즘에서 핵심적으로 사용되는 'RGB유사도' 개념과 '적분영상'에 대하여 설명한다.

#### (1) 영상 데이터 매칭기준 배경이론

##### ① RGB유사도

[0445] 도 11은 영상 데이터 매칭 기준을 설명하기 전에 알아야 하는 'RGB유사도' 개념을 설명하기 위한 도면이다.

[0446] 'RGB유사도'란 두 색상이 얼마나 비슷하거나 다른지를 나타내는 척도 중에 하나로서 R(적색), G(녹색), B(청색)의 색상 값을 3개의 축을 가진 3차원 공간상의 점으로 가정하여 두 점의 3차원 공간상의 유클리드 거리를 계산하여 두 점이 멀면 두 색상은 다르고, 두 점이 가깝다면 두 색상은 비슷하다고 말 할 수 있는 척도를 뜻한다.

[0447] 예를 들어, 8bit RGB값으로 정의되는 3차원 RGB 공간에서 백색(255,255,255) (C1)과 흑색(0,0,0) (C2)는 그 유클리드 거리가  $\sqrt{(255-0)^2 + (255-0)^2 + (255-0)^2} = 441.6729$ 로 가장 멀다. 이 때의 두 색상의 RGB유사도는 0%라고 가정한다.

[0448] 따라서, A지점과 B지점에 대한 두 색의 RGB유사도는 다음 [수식14]와 같다.

[0449] [수식14]

[0450] A점의 RGB값  $(R_A, G_A, B_A)$

[0451] B점의 RGB값  $(R_B, G_B, B_B)$

$$\text{RGB유사도} (\%) = \left( 1 - \frac{\sqrt{(R_A - R_B)^2 + (G_A - G_B)^2 + (B_A - B_B)^2}}{441.6729} \right) \times 100$$

[0453] 예를 들어, 적색(255,0,0)(C3)과 청색(0,0,255)(C4)의 RGB유사도는 18.35%, 적색(255,0,0)(C3)과 선홍색(255,0,255)(C5)의 경우는 42.27%라고 할 수 있는데, 이는 적색과 청색은 18.35% 비슷한 색상이며 적색과 선홍색은 42.27% 비슷한 색상이라고 할 수 있다.

[0454] 즉, 두 지점의 색상이 비슷할수록 RGB 유사도는 100%에 가깝고, 두 지점의 색상이 다를 수록 RGB 유사도는 0%에 가깝다고 할 수 있다.

[0455] 본 명세서 상에서는 RGB값을 이용해서 두 지점의 색상의 차이를 설명하였으나, 색상 표현 방법에 따라 CMYK, HSV, HSI 또는 YCrCb 등 다양한 방식이 있고, 어떠한 색상의 차이를 설명하는 알고리즘도 상당히 많이 있으나, 본 발명의 개념을 설명하기 위하여 'RGB유사도'를 대표적인 예로서 사용한 것뿐이며, 본 발명에 어떠한 색상의 차이를 나타내는 다른 방식도 적용이 가능함을 명시한다.

## ② 적분영상

[0457] 도 12는 본 발명의 영상처리에 자주 사용되는 '적분영상'을 설명하기 위한 도면이다.

[0458] 적분영상이란 다음 픽셀의 RGB값에 이전 픽셀의 RGB값의 합이 더해진 영상을 뜻하며 적분영상을 이용 시 블록단위의 영상 데이터의 합, 즉 RGB값의 합을 빠른 속도로 계산할 수 있다.

[0459] 예를 들어, 도 12의 좌측 원영상에 대한 원영상 데이터(ii1)에서 음영 처리된 부분(ii3)인 영상좌표 x축 방향(행 방향)으로 0~3이고 y축 방향(열 방향)으로 0~3인 부분(ii3)의 픽셀 값의 합은 우측 적분영상에 대한 적분영상 데이터(ii2)에서의 영상좌표 (3,3)의 값인 341이 된다.

[0460] 이런 방식으로, 적분영상 데이터(ii2)에서 영상좌표(9,9)의 값인 6400은 원영상 데이터(ii1)의 x축 방향으로 0~9이고 y축 방향으로 0~9인 부분[즉, 원영상 데이터(ii1) 전체]의 픽셀 값을 합한 값이다.

[0461] 적분영상 데이터의 일반식은 다음 [수식15]과 같다.

[0462] [수식15]

$$ii(x,y) = \sum_{x' \leq x, y' \leq y} S(x',y')$$

[0464] ( $ii(x,y)$  : 적분영상 데이터의 픽셀값,  $S(x',y')$  : 원영상 데이터의 픽셀값)

[0465] 하지만, [수식15]의 일반식으로 적분영상 데이터를 구축하면 계산시간이 오래 걸리게 된다. 따라서 다음과 같은 변형된 [수식16]을 사용하는 것이 적분영상 데이터 구축에 보다 효율적이다.

[0466] [수식16]

$$ii(x,y) = ii(x-1,y) + ii(x,y-1) - ii(x-1,y-1) + S(x,y)$$

[0468] [ $ii(x,y)$  : 적분영상 데이터의 픽셀값,  $S(x,y)$  : 원영상 데이터의 픽셀값] 예를 들어 적분영상 데이터의 영상 좌표가 (3,3)인 값인 341을 계산하려면, [수식15]의 경우 16회의 덧셈연산을 하여야 하나, [수식16]을 사용하면 ( $ii(2,3)=233+(ii(3,2)=186)-(ii(2,2)=113)+(S(3,3)=35)$ , 즉  $233+186-113+35=341$ 와 같이 단 3회의 가감산만으로 해당 좌표에 대한 적분영상 데이터의 값을 계산할 수 있다.

[0469] 이와 비슷한 원리로 적분영상 데이터가 만들어지면 원영상 데이터의 특정 블록의 픽셀 합계를 블록의 크기에 상관없이 다음 [수식17]과 같이 단 3회의 가감산만으로 계산할 수 있게 된다.

[0470] [수식17]

$$\sum\{S(x_i,y_i) \sim S(x_e,y_e)\} = ii(x_i-1,y_i-1) + ii(x_e,y_e) - ii(x_{i-1},y_e) - ii(x_e,y_{i-1})$$

[0472] ( $S(x_i,y_i)$  블록계산 시작점,  $S(x_e,y_e)$  블록계산 끝점,  $ii(x,y)$  적분영상 데이터의 픽셀값)

- [0473] 예를 들어 원영상 데이터(ii1)에서 깎은 실선 부분(ii4) 픽셀의 합을 계산하고자 한다면 일반적인 경우 16회의 덧셈이 필요하지만 적분영상 데이터를 이용하면  $1098+5194-2325-2929=1038$ 와 같이 3회의 가감산만으로 구할 수 있다.
- [0474] **(2) 영상데이터 매칭기준 알고리즘**
- [0475] 다음 도 13은 앞서 설명한 RGB유사도(도 11 참조)와 적분영상 데이터(도 12 참조)를 사용하여 영상데이터의 매칭기준을 계산하는 연산부(120)의 동작(즉, 도 9의 S310 단계)에 대한 순서도이다.
- [0476] 먼저, 연산부(120)는 매칭을 하려고 하는 2개의 영상데이터(즉, 기준 영상 데이터와 매칭 영상 데이터)를 메모리(220) 혹은 저장부(400)로부터 읽어온다(S301-1, S302-1). 여기서 기준 영상 데이터는 1차 사진 촬영 동작을 통해 획득된 영상 데이터이고, 매칭 영상 데이터는 2차 사진 촬영 동작을 통해 획득된 영상 데이터일 수 있다.
- [0477] 촬영부(110)로 이용되는 디지털 카메라의 현재 해상도는 2016년 12월 기준 현재  $5312 \times 2988$ 인 약 1,600만 화소의 고화소 카메라가 상용화 되어 있으며, 카메라의 해상도는 기술이 발전할수록 더욱 높아질 것이다.
- [0478] 이런 고화질 영상 데이터를 처리하는 데는 매우 큰 메모리(220)의 용량과 연산부(210)의 처리능력이 필요하다.
- [0479] 따라서, 영상 해상도가 연산 및 제어 유닛(200)의 처리능력보다 크다면(S302), 연산부(120)는 적절히 이미지 다운샘플링을 하여(S303) 제1 및 제2 영상 데이터의 용량을 줄이는 비교 처리 과정을 실시한다.
- [0480] 이미지 다운샘플링은 영상의 해상도를 줄이는 연산을 의미하며, 예를 들어, 안드로이드 기반의 스마트폰의 경우, BitmapFactory명령어 셋에서 options.inSampleSize = 4(여기서 4는 1 이상의 정수형 배수인 수이다)로 하여 해상도가  $5,312 \times 2988$ 인 영상 데이터를  $1,328 \times 747$ 의 해상도로 줄이는 연산을 뜻한다.
- [0481] 하지만, 연산부(210)의 성능과 메모리(220)의 용량이 증가한다면, 보다 큰 고해상도의 영상에 대한 영상 데이터를 이미지 다운샘플링 없이 처리할 수 있을 것이다.
- [0482] 연산부(210)의 처리능력에 맞게 이미지 다운샘플링 후에 다운샘플링한 제1 및 제2 영상 데이터를 앞서 설명한 [수식16]의 방법으로 각 RGB채널별 적분 영상 데이터인 제1 및 제2 적분 영상 데이터를 생성한다(S304).
- [0483] 그런 다음, 연산부(210)는 생성된 적분 영상의 제1 및 제2 적분 영상 데이터를 이용하여, 획득된 각 제1 및 제2 적분 영상 데이터를 정해진 크기(예,  $50 \times 50$  또는  $25 \times 25$ )의 영역인 일정 영역으로 분할하여 제1 및 제2 모자이크 영상에 대한 제1 및 제2 모자이크 영상 데이터를 생성한다(S305).
- [0484] 모자이크 영상이란 해당 영상 데이터의 일정 영역안의 RGB값들을 평균값으로 통일 시킨 영상을 뜻한다.
- [0485] 예를 들어, 해상도  $1000 \times 500$ 인 영상을  $50 \times 50$  크기로 모자이크화 했다는 뜻은 가로 50픽셀과 세로 50픽셀의 단위로 영상을 분할하여 분할 된 범위 내의 RGB값들의 평균값들 계산하여 분할된 각 영역들을 계산된 각 RGB평균값들로 채운 영상으로 생성했다는 뜻으로 획득된 모자이크 영상 데이터의 전체 용량은  $20 \times 10$ 크기로 줄어드는 효과와 같고, 효율적인 영상 비교가 가능해 진다.
- [0486] 모자이크 영상을 빠르게 계산하기 위해서 적분영상을 이용하는데, 예를 들어  $50 \times 50$ 크기의 RGB평균값들을 계산할 경우, 적분영상을 사용하지 않을 때는 각 색상별로 2,500회의 덧셈과 1회의 나눗셈이 필요하나, 적분영상을 이용하면 3회의 가감산과 1회의 나눗셈만으로 각 색상별 RGB평균값들을 계산할 수 있다.
- [0487] 모자이크화에 대해서는 다음 도 14를 참고로 하여 다음에 자세히 설명 하겠다.
- [0488] 따라서, 연산부(210)는 이렇게 모자이크화 된 두 장의 제1 및 제2 모자이크 영상 데이터를 매칭하여 최적의 영상 데이터 매칭기준을 찾는다(S306).
- [0489] 최적의 영상 데이터 매칭기준을 찾기 위해 연산부(210)는 두 개의 제1 및 제2 모자이크 영상 데이터 중에서 1차 사진 촬영으로 획득된 1차 영상에 대한 제1 모자이크 영상 데이터를 기준으로 나머지 하나인 제2 모자이크 영상 데이터를 한 픽셀씩 이동시켜 겹쳐지는 부분의 각 픽셀별 'RGB유사도'를 계산 후 평균을 구한다(S307).
- [0490] 이때, 연산부(210)는 제2 모자이크 영상 데이터의 끝점(즉, 마지막 행과 마지막 열에 위치한 픽셀)까지 이동하면 계산을 멈춘다(S308).
- [0491] 연산부(210)는 이와 같이 계산된 RGB유사도 평균이 가장 높은 픽셀 이동점을 영상데이터 매칭의 기준으로 정한다(S309).
- [0492] 영상데이터 매칭기준을 계산할 때 영상 촬영 부분(P1)에서 얻은 위치 정보 측정장치(1000)의 이동 거리와 최종

자세 데이터를 사용하면 매칭기준 계산을 더 단순화 할 수 있다.

[0493] 예를 들어, 위치 정보 측정장치(1000)의 이동 거리에 의해 위치 정보 측정장치(1000)가 A영상 위치에서 B영상 위치까지 왼쪽으로 움직였고, 위치 정보 측정장치(1000)의 자세 데이터는 변화가 없다면, 연산부(210)는 영상데이터 매칭 기준도 왼쪽으로 움직였을 것이라고 가정하여 계산을 수행할 수 있다.

[0494] 또한, 위치 정보 측정장치(1000)의 이동 거리가 왼쪽으로 움직였지만 위치 정보 측정장치(1000)의 자세 데이터가 방향각이 A영상 보다 B영상의 방향각 회전량이 상당히 큰 경우, 연산부(210)는 영상데이터 매칭 기준이 오른쪽으로 움직였을 것이라고 가정하여 계산할 수 있다.

[0495] 이렇게 위치 정보 측정장치(1000)의 자세 데이터와 이동 거리를 이용하여 위치 정보 측정장치(1000)의 자세와 이동 상태를 고려한 후 영상데이터 매칭 기준 계산이 이루어지면 영상데이터 매칭 기준 계산의 정확도가 향상된다.

[0496] 제1 및 제2 모자이크 영상 데이터의 매칭 과정에 대한 도 13의 순서도에 대한 이해를 돋기 위해, 도 14에 모자이크 영상 매칭 단계(도 13의 S306)에 대한 의사코드의 한 예를 첨부하였다.

[0497] 도 14에 도시한 의사코드는 A영상(예, 기준 영상)의 중심점과 B영상(예, 매칭 영상)의 시작점을 맞춰서 비교를 시작하나 이는 모자이크 매칭의 개념을 설명해주기 위한 한 예이며, A영상을 기준으로 B영상을 이동시키는 등의 다양한 알고리즘을 적용할 수 있다.

### (3) 영상데이터 매칭기준 계산의 예

[0499] 도 15는 영상데이터 매칭 기준을 보다 쉽게 설명하기 위해 영상을 예로 든 개념도이다.

[0500] 이해를 돋기 위해 각 영상의 해상도와는 상관없이 크기를 똑같이 하였으나, 영상의 크기는 아래 팔호로 표기한 해상도로 줄어들었다.

[0501] 먼저, 연산부(210)는 최초 2장의 제1 및 제2 영상 데이터를 읽어와 제1 및 제1 영상 데이터에 대한 영상(예, 원영상)(S301-1, S301-2)을 메모리(220)에 저장한다.

[0502] 이때, 입력받은 제1 및 제2 영상 데이터는 해상도가  $5,312 \times 2,988$ 인 약 1,600만 픽셀의 영상 데이터이기 때문에, 연산부(210)는 자신의 영상 처리능력에 따라 적절하게 이미지 다운샘플링을 실시하여  $1,328 \times 747$ 크기의 영상 데이터를 얻고 이에 대응하는 영상(S303-1, S303-2)을 메모리(220) 혹은 저장부(400)에 저장한다.

[0503] 이 후, 연산부(210)는 다운샘플링한 제1 및 제2 영상 데이터(S303-1, S303-2)로 적분영상에 대한 제1 및 제2 적분 영상 데이터를 구축한다.

[0504] 적분영상은 픽셀값의 합을 계산해 놓은 행렬식에 가깝기 때문에 영상으로 표현하기에는 많은 제약이 있어, 본 도면에는 생략하였다.

[0505] 다음, 연산부(210)는 계산된 적분영상의 제1 및 제2 적분영상 데이터를 바탕으로  $50 \times 50$ 크기로 모자이크화한 제1 및 제2 모자이크 영상 데이터를 생성하여 메모리(220) 혹은 저장부(400)에 저장한다.

[0506] 본 예에서는 영상처리 동작의 이해를 돋기 위해  $50 \times 50$ 크기로 모자이크화를 실시하였으나, 이에 한정되지 않고 시스템의 처리 능력과 매칭기준의 정확도를 높이기 위해서 적절한 모자이크화 크기를 결정해 줄 수 있다.

[0507] 예를 들어, 위치 정보 측정장치(1000)가 안드로이드 기반의 스마트폰의 경우 약  $25 \times 25$ 크기로 모자이크화를 실시하여 이미지 매칭하는 것이 보다 효과적이었다.

[0508] 생성된 모자이크 영상에 대한 해상도는  $26 \times 14$ 이며, 이때 영상좌표로 x좌표 1,300~1,328인 부분과 y좌표 700~747인 부분의 영역은 모자이크화 하지 않고 무시한다.

[0509] 도 15에 도시된 각 영상은 이해를 돋기 위해 각 영상의 크기를 통일 시켰는데, 제1 및 제2 영상 데이터에 대한 최초 원영상은 1,600만 픽셀을 가졌지만, 이미지 다운샘플링 동작으로 100만 픽셀의 영상, 마지막 모자이크화 동작으로 364픽셀의 영상으로, 원영상과 비교해 볼 때 영상의 크기가 0.002%로 상당히 줄어들었음을 알 수 있다.

[0510] 생성된 제1 및 제2 모자이크 영상은  $26 \times 14$ 크기를 가지고 영상데이터 매칭기준을 계산한다.

[0511] 도 16은 도 14의 의사코드에 의한 영상데이터 매칭 기준의 첫 단계인 모자이크화된 A영상의 중심점에 모자이크화된 B영상의 시작점을 위치시킨 후 겹쳐지는 부분(S307)의 RGB유사도를 계산하는 과정을 보여준다. (편의상,

도 16에서 S305-1영상을 A영상, S305-2영상을 B영상이라 한다.)

[0512] 이 때의 영상데이터 매칭 x축 기준은 -13이고, 영상데이터 매칭 y축 기준은 -7이 된다.

[0513] 연산부(210)는 모자이크화된 A영상(S305-1)의 중심점에 B영상(S305-2)의 시작점을 일치시킨 후 겹쳐진 부분(S307)의 RGB유사도를 계산한다.

[0514] 영상이 표시되는 부분 아래에 기재된 표(S307-1)에는 A영상(S305-1)에서의 겹쳐진 부분(S307)의 RGB값들이 기재되어 있으며, 표(S307-2)에는 B영상(S305-2)에서의 겹쳐진 부분(S307)의 RGB값이 기재되어 있다.

[0515] 이를 표(S307-1, S307-2)의 각 픽셀별로 RGB유사도를 계산한 것이 표(S307-3)에 도시되며 RGB유사도의 평균값은 76.5%이다.

[0516] 이렇게 최초 매칭된 부분 즉 영상 데이터 매칭 x축 기준이 -13이고 영상데이터 매칭 y축 기준이 -7인 부분(S307)의 RGB유사도 평균은 76.5%가 된다.

[0517] 도 17에 영상데이터 매칭기준을 찾기 위해 매칭기준을 계속 변경해 가면서 RGB유사도를 계산하는 과정을 나타낸다. 도 18에는 총 9단계로 표시됐지만 중간단계가 생략된 것으로, 실제로는 첫 시작인 매칭기준 (-13, -7)부터 마지막 (+13, +7)까지 총 405단계의 비교가 이루어진다.

[0518] 이런 비교 단계는 사용자가 어떠한 알고리즘을 사용했는가에 따라서 다양하게 달라질 수 있다.

[0519] 각 매칭기준별 RGB유사도 평균을 계산하고 이후 가장 높은 RGB유사도를 가질 때가 바로 영상데이터 매칭기준이 된다.

[0520] 도 18에는 영상데이터 매칭기준 계산 결과 RGB유사도가 가장 큰 매칭 부분(S310)을 도시한 도면이다.

[0521] 도 18의 영상데이터 매칭 x축 기준은 -3이고, 영상데이터 매칭 y축 기준은 0이 된다.

[0522] 이해를 돋기 위해, 도 18에 두 번의 사진 촬영에 의해 획득된 두 개의 원영상(S305-1, S305-2)를 매칭기준에 맞게 배치한 도면을 도시한다.

[0523] 도 18에 도시한 것처럼, 아래위에 각각 위치한 원영상(S305-1)과 원영상(S305-2)의 영상위치가 비슷하게 맞는다는 것을 알 수 있다.

[0524] 매칭기준이 x축으로 -3이고, y축으로 0이라는 것을 픽셀 좌표로 환산해 보면  $50 \times 50$  모자이크화 되었기 때문에 다운샘플링 영상좌표로는 x축(즉, 행 방향)으로  $-3 \times 50 = -150$  픽셀, y축(즉 열 방향)으로  $0 \times 50 = 0$  픽셀이 된다. 다운샘플링이 1/4로 되었기 때문에 원영상 좌표로 변환하면 x축으로  $-3 \times 50 \times 4 = -600$  픽셀이고, y축으로  $0 \times 50 \times 4 = 0$  픽셀이 되며, 검색범위는  $\pm 50 \times 4 = \pm 200$  픽셀이 될 것이다.

[0525] 이를 일반식으로 나타내면 다음 [수식18]와 같다.

[0526] [수식18]

[0527] -원영상에서  $1/n$  로 다운샘플링

[0528] - $m \times m$  으로 모자이크화

[0529] -모자이크 영상의 영상데이터 매칭 x축 기준:  $X_m$

[0530] -모자이크 영상의 영상데이터 매칭 y축 기준:  $Y_m$

[0531] -원영상에서의 영상데이터 매칭 x축 기준:  $x_{shift} = X_m \times m \times n$  (pixel)

[0532] -원영상에서의 영상데이터 매칭 y축 기준:  $y_{shift} = Y_m \times m \times n$  (pixel)

[0533] -검색범위:  $S = \pm m \times n$  (pixel)

[0534] 예를 들어, A영상에서 어떤 특징점의 영상좌표가 (3826, 356)이었다면, B영상 전체에서 특징점을 찾는 것이 아니라 B영상에서 (3226, 356)픽셀의  $\pm 200$  픽셀 주변을 검색하면 특징점을 매칭할 수 있다는 뜻이다(도 11 참조).

[0535] 도 15~도 18까지 예를 들어서 영상데이터 매칭기준을 설명하였는데, 이해를 돋기 위해  $50 \times 50$  모자이크를 사용하

였으나,  $25 \times 25$ 크기의 모자이크를 사용하면 연산량은 4배정도로 늘어나나 매칭기준은  $50 \times 50$  모자이크화 일 때 보다 세밀하게 계산할 수 있다.

[0536]  $25 \times 25$ 모자이크로 계산할 경우 모자이크화된 영상 데이터 매칭기준은  $X_m = -7, Y_m = 0$  이었다.

[0537] 이 경우 A영상의 어떤 특징점의 영상좌표가 (3826, 356)이었다면, B영상에서는  $x_{shift} = -700$ 픽셀,  $y_{shift} = 0$ 픽셀 이동한 (3126, 356)픽셀의 ±100픽셀 주변을 검색하면 쉽게 특징점을 매칭할 수 있다.

#### 4) 특징점 가능 영역 선택 (도 9의 S330)

[0539] 다음, 도 9에서 특징점 가능 영역을 선택하는 단계(S330)의 동작에 대하여 설명하다.

##### (1) 영상처리부분(P1)의 사용자 제어 화면을 통한 특징점 가능 영역 선택

[0541] 도 19는 특징점 가능 영역을 선택하기 위한 사용자 제어화면의 예이다.

[0542] 도 19의 (a)는 사용자 제어 부분의 첫 화면으로서, 연산부(210)는 사진 촬영한 첫 영상을 출력부(300)의 영상 출력화면(U21)에 표시한다.

[0543] 도 19의 (a)에 도시한 것처럼, 촬영한 사진 간에 전환을 할 수 있는 슬라이드 형식의 바(U22)가 화면(U21)의 오른쪽 위에 있어, 사용자는 이 슬라이드 바(U22)를 이용하여 촬영된 사진들 간에 이동이 가능하다.

[0544] 예를 들어, 사용자가 세번의 촬영 동작으로 A, B 및 C인 3장의 영상이 차례로 획득한 경우, 사용자가 슬라이드 바(U22)를 옆으로 한 칸 움직이일 때마다 영상 출력 화면(U21)에 현재 표시중인 영상인 A영상이 사라지고 B영상이 출력되며, 옆으로 또 한 칸 움직이면 현재 표시중인 B영상 대신에 C영상이 출력된다.

[0545] 또한, 메뉴버튼(U23)을 이용하여 사용자는 현재의 영상처리를 위해 사진 촬영된 영상, 촬영위치 및 자세 등의 각종 데이터를 저장할 수 있는 파일 저장 기능, 기존에 촬영된 영상을 불러 올 수 있는 파일 불러오기 기능, 기타 각종 설정[자동으로 선택하는 특징점들의 개수, 측정단위, 표시형식(거리, 좌표 등) 등과 같은 기능]을 조절 할 수 있는 기능 등을 선택할 수 있다.

[0546] 클리어(CLEAR)버튼(U24)은 사용자가 선택한 특징점이나 시스템이 자동으로 선택한 특징점들을 삭제하여 초기화 하는 버튼이다.

[0547] 자동(AUTO)버튼(U25)은 시스템에서 자동으로 특징점들을 선택하여 대상물까지의 거리와 좌표를 계산해 주는 버튼이다.

[0548] 도 19의 (b)는 사용자가 특징점을 선택하는 모습으로 사용자가 촬영된 영상을 터치시 터치 위치에 따라 좌측 하단이나 우측 하단에 별도의 화면을 통해 사용자가 선택한 부분의 확대영상(U26)을 출력하여 사용자가 터치한 채로 조금씩 움직여 사용자가 원하는 특징점을 정확히 선택할 수 있도록 도와준다.

[0549] 하지만 아무리 확대 화면(U26)을 제공해 준다 해도 특징점을 픽셀 단위로 정확히 선택하는 것은 상당히 어렵고, 사용자에게 픽셀단위 선택을 요구하는 것은 오히려 큰 불편을 주며, 특징점의 영상좌표의 정확도도 오히려 떨어진다.

[0550] 따라서 사용자가 특징점의 개략적인 위치를 선택했다 가정하고, 사용자가 터치한 지점의 좌표에 정해진 픽셀크기(예 ±50픽셀)만큼 더하고 빼서(예, ±50픽셀) 정해진 영역에 대해 특징점을 분석하고 검출하여, 가장 큰 특징을 가지고 있는 한 점을 특징점으로 선택하는 것이 효율적이다. 이때, 좌표는 행 방향 위치값과 열방향 위치값을 가지므로, 행 방향과 열 방향의 각 위치값에 정해진 픽셀 크기가 가감되어 정해진 영역이 특징점을 분석하기 위한 특징점 가능 영역이 된다.

[0551] 예를 들어, 도 19의 (b)에서 사용자가 선택한 지점의 좌표가 (3820, 340)이었다면 특징점 가능 영역인 (3770,320)~(3850,390) 영역의 영상 데이터를 특징점 분석하여 확대화면(U26)을 통해 보이는 영상의 모서리점인 (3827, 359)의 좌표를 갖는 지점을 특징점으로 도출하여 최적의 특징점을 찾아낼 수 있다.

[0552] 즉, 연산부(210)는 영상 출력화면(U21)에서 사용자가 터치한 영역을 특징점 가능영역으로 선택하고 약  $101 \times 101$  픽셀 크기의 영역에 대해서만 특징점을 분석하여 검출을 하는 것이다.

[0553] 도 19의 (c)는 사용자가 선택한 특징점을 영상처리 하여 A영상과 B영상의 특징점 매칭을 끝내 특징점의 영상좌표를 취득하고, 취득한 특징점의 영상좌표와 위치 정보 측정장치(1000)의 자세 데이터 및 위치 데이터를 이용하

여 측량부분 계산을 끝내 위치정보 측정정치(1000)에서 해당 특징점까지의 거리를 표시해준 화면이다.

[0554] 거리 측량에 대한 동작은 도 33을 참고로 하여 이후에 설명한다.

[0555] 사용자에 의해 클리어 버튼(U24)이 동작되면 영상 출력화면(U21)의 표시 상태는 초기화면인 도 9의 (a)로 돌아가게 된다.

[0556] 도 19에 도시된 사용자 제어화면은 하나의 예로서, 화면 구성, 버튼 구성, 버튼 모양 등은 다양하게 조합 및 변형될 수 있다.

[0557] 도 20에 도 19에 해당하는 영상처리부분의 동작원리를 나타내는 '의사코드'이다.

[0558] 도 20에서, 17라인이 사용자에 의한 터치가 끝났을 때 해당 영역에 대하여 특징점 분석 및 검출을 시작하는 코드이다.

## (2) 특징점 가능영역의 자동선택

[0560] 도 19을 참고로 하여 설명한 것처럼, 출력부(300)의 영상 출력화면(U21)으로 출력되는 영상에서 사용자는 자신이 원하는 점을 터치와 같은 선택 동작을 통해 직접 선택할 수 있지만, 이와 달리, 시스템 상에서 자동으로 특징점 가능영역을 분석할 수 있다.

[0561] 특징점 분석 전에 특징점 가능영역을 사전에 분석하는 이유는 전체 영상에 대하여 특징점 분석을 하기 전에 특징점이 있을 수 있는 영역을 사전에 한정하여 영상처리의 속도를 높이기 위해서이다.

[0562] 물론, 시스템에 설치된 연산 및 제어유닛(210)의 능력이 충분하다면, 본 특징점 가능영역 자동 검출 부분은 생략되고, 영상전체에 대해서 특징점 분석을 하는 것이 더 유리할 것이다.

[0563] 특징점 가능영역의 자동으로 선택하기 위해, 사용자는 도 19의 (a)의 자동버튼(U25)을 누르게 되고, 이러한 자동버튼(US25)이 동작된 상태로 판정되면 연산부(210)는 자동으로 특징점을 검출하는 동작을 실시한다.

[0564] 도 21에 자동으로 특징점 가능영역을 검출하는 동작을 설명하기 위한 개념도를 도시한다.

[0565] 본 예에서, 연산부(210)는 특징점 자동 검출을 위해 영상데이터매칭 기준에 사용되었던 모자이크 영상에 대한 모자이크 영상 데이터를 이용해서 특징점을 자동 검출한다.

[0566] 모자이크 영상 데이터를 사용하기 때문에 본 예의 특징점 자동 검출 방법은 영상의 큰 엣지 즉, 경계가 뚜렷하고 다른 부분과의 RGB유사도 차이가 큰 곳을 빠르게 찾을 수 있게 된다.

[0567] 도 21의 (a)을 참고로 하면, 본 예의 경우, 모자이크 영상에서 정해진 개수(예, 4개)의 픽셀에 대한 데이터인 픽셀 데이터를 기준으로 RGB유사도를 계산하여 다음 3가지 특징 중 한 개라도 있으면 특징점 가능영역(C01, C02, C03)으로 분류한다.

[0568] - 영역(C01): 영역(C01)과 같이 4개의 픽셀 중 3개 픽셀의 RGB유사도가 비슷하지만 1개 픽셀의 RGB유사도가 다른 픽셀에 비교하여 크게 다르면 해당영역에는 특징점이 있는 것으로 판정한다. 단, 예를 들어, NC03과 같이 대각 방향으로 RGB유사도가 비슷한 패턴으로 이루어지면 특징점이 아닌 것으로 하여 이런 영역(NC03)은 특징점 가능 영역에서 배제시킨다.

[0569] - 영역(C02): 영역(C02)과 같이 4개의 픽셀 중 2개 픽셀의 RGB유사도가 비슷하지만 다른 2개 픽셀의 RGB 유사도가 크게 다르면 해당영역엔 특징점이 있는 것으로 판정한다. 단, 예를 들어 NC04와 같이 대각 방향으로 RGB유사도가 비슷한 패턴이 있으면 이런 영역(NC04)은 특징점 가능 영역에서 배제시킨다.

[0570] - 영역(C03): 영역(C03)과 같이 4개 픽셀값의 RGB유사도가 전부 다르면 해당영역엔 특징점이 있는 것으로 판정한다. 단 대각 방향으로 RGB유사도가 비슷한 패턴이 있으면 특징점이 아닌 것으로 판정된다.

[0571] 도 21에서, 특징점 가능영역으로 판정된 영역(C01, C02, C03)의 가운데 붉은 점이 특징점이 있을 수 있는 영역이다.

[0572] 하지만 위 3가지의 특징점 가능영역 판단 기준으로 특징점 가능영역을 선택하는 것 보다는 반대로 특징점이 있을 수 없는 영역, 즉, 특징점 불가능 영역(NC01~NC04)을 파악하여 특징점 가능 영역을 선택할 수 있다.

[0573] 이러한 특징점 불가능 영역을 선별하는 기준은 다음과 같다.

[0574] - 영역(NC01): 영역(NC01)과 같이 4개 픽셀의 RGB유사도가 비슷하면 특징점이 없다.

- [0575] - 영역(NC02-1): 영역(NC02-1)과 같이 가로 방향(행 방향)으로 배열된 2개 픽셀간의 RGB유사도가 비슷하면 특징점이 없다.
- [0576] - 영역(NC02-2): 영역(NC02-2)와 같이 세로 방향(열 방향)으로 배열된 2개 픽셀간의 RGB유사도가 비슷하면 특징점이 없다.
- [0577] - 영역(NC03): 영역(NC03)과 같이 세로 방향으로 1픽셀 그리고 가로 방향으로 1픽셀씩 이동시켜 대각선 방향으로의 이동한 사각형의 RGB유사도가 비슷하면 특징점이 없다.
- [0578] - 영역(NC04): 영역(NC04)과 같이 대각선으로 가로 방향으로 2픽셀 그리고 세로 방향으로 1픽셀씩 또는 가로 방향으로 1픽셀 그리고 세로 방향으로 2픽셀씩 이동시켜 대각선 방향으로 이동한 사각형의 RGB유사도가 비슷하면 특징점이 없다.
- [0579] 이러한 기준에 따르면, 모서리(corner)가 아닌 경계(Edge)는 특징점에서 제외됨을 알 수 있다.
- [0580] 즉, 영역(NC02-1, NC02-2)은 경계가 가로 방향이나 세로 방향으로 있는 영역이고, 영역(NC03과 NC04)은 경계가 대각선 방향으로 있는 영역이다.
- [0581] 이들 영역(NC01, NC02-1, NC02-2)과 같은 특징점이 없는 영역을 찾기 위해서 도 21의 (b)와 같은 행 방향과 열 방향으로 서로 인접하게 위치한 4개 픽셀의 상호간의 RGB유사도[도 21의 (b)에서 RGB 유사도인 RGBd(0)~RGBd(3)]를 비교하여 RGB유사도[RGBd(0)]와 RGB 유사도[RGBd(2)]가 또는 RGB유사도[RGBd(1)]와 RGB유사도[RGBd(3)]의 값이 일정 유사도값(예를 들어 90)을 넘으면 특징점 가능영역에서 제외하면 된다.
- [0582] 또한, 영역(NC03)과 같이 가로 방향으로 1픽셀 그리고 세로 방향으로 1픽셀씩 이동하여 대각선 방향으로 배열되어 있는 해당 픽셀을 비교하는 경우, 도 22에 도시한 것처럼 4가지 방법을 통해 비교하고 각 방법 중 하나라도 RGB 유사도[도 22에서 (0)~(3)]의 값이 모두 일정 유사도값(예를 들어 90)을 넘으면 해당 영역은 특징점 가능영역에서 제외된다.
- [0583] 영역(NC04)과 같이 가로 방향으로 2픽셀 그리고 세로 방향으로 1픽셀씩 또는 가로 방향으로 1픽셀 그리고 세로 방향으로 1픽셀씩 이동하여 대각선 방향으로 배열되어 있는 해당 픽셀을 비교하는 경우는 도 23에 도시한 것처럼 8가지 방법을 통해 비교하고, 각 방법 중 하나라도 RGB유사도[도 23에서 (0)~(3)]의 값이 모두 일정 유사도값(예를 들어 90)을 넘으면 특징점 가능영역에서 제외된다.
- [0584] 도 24에는 이러한 방식을 통해 자동버튼(U25)이 동작될 때 자동으로 특징점 가능영역을 찾아주는 알고리즘의 한 예에 대한 의사코드가 도시되어 있다.
- [0585] 도 24의 의사코드 중에서, 7라인에서부터 15라인까지가 도 21의 (b) 개념을 이용하여 가로 방향이나 세로방향으로 위치하는 경계(수직 경계나 수평 경계)나 경계가 없는 부분을 찾아 특징점 가능 영역에서 제외시키는 코드이고, 이후 16라인에서부터 25라인까지가 각 대각선 방향으로 해당 픽셀 개수만큼 이동시켜 RGB유사도를 비교하는 코드이지만, 도 24에서는 자동으로 특징점 가능 영역을 검색하기 위한 세부적인 코드는 생략하였다.
- [0586] 수직 방향(즉, 열 방향)으로의 경계인 수직 경계 및 수평 방향(즉, 행 방향)으로의 경계인 수평 경계의 분석은 한 픽셀당 4회의 RGB유사도 연산과 4번의 비교만으로 끝나기 때문에  $4 \times 4 = 16$ 회의 연산이 필요하다.
- [0587] 하지만, 대각선 방향으로의 RGB 유사도 비교의 경우, 총 12번씩 4회의 RGB 유사도 분석을 해야 하고, 각 분석 결과를 이용해서, 각 4회의 RGB 유사도 비교 동작을 해야 하기 때문에 한 픽셀당 총  $12 \times 4 \times 4 = 192$  회의 추가 연산이 필요하다.
- [0588] 따라서, 연산부(210)의 성능에 시스템에 따라 대각선 방향으로의 RGB 유사도 비교를 하지 않고 수직 경계와 수평 경계의 분석만을 사용하는 것이 보다 효율적일 수 있다.
- [0589] 도 25는 도 24의 알고리즘을 적용하여 특징점 분석을 실시한 예이며, 도 25에서 하나의 사각형은 하나의 픽셀을 의미하며 '1'로 표시된 부분이 특징점 가능영역이다.
- [0590] 도 25의 (a)는  $50 \times 50$  크기의 모자이크 영상으로 특징점 가능영역 분석을 실시한 결과이다.
- [0591] 영역(C01)과 같이 '1'로 표시된 픽셀을 기준으로 하여 가로 방향과 세로 방향으로 각각 1픽셀씩 추가된 영역이 특징점 가능 영역이며 이 영역에 대하여 특징점 분석을 실시하면 된다.
- [0592] 이러한 특징점 가능 영역 자동 검출 방법을 통해  $50 \times 50$  크기의 모자이크 영상의 총픽셀 수인 364( $=26 \times 14$ )개의

픽셀에서 특징점 가능 영역으로서 85개 픽셀이 선 되었고, 영상 전체의 총 픽셀수(=364)와 비교하여 23.3%(=85/364) 정도의 영상만 영상 처리하면 되기 때문에, 보다 효율적으로 특징점 선택을 위한 영상 처리가 가능한다.

[0593] 만약, 대각선 방향으로의 RGB 유사도 비교를 수행하지 않고 수직 방향과 수평방향으로의 경계 분석을 할 경우 선택되는 특징점 가능영역은 100개픽셀로서, 영상 전체의 총 픽셀수(=364)와 비교하여 27.5%(=100/364)정도의 영상 처리(즉, 픽셀 처리)만 하면 된다.

[0594] 도 25의 (b)의 경우는  $25 \times 25$  크기의 모자이크 영상을 이용하여 특징점 가능영역 분석을 실시한 결과이다.

[0595] 도 25의 (b)에서  $25 \times 25$  크기의 모자이크 영상의 총 픽셀 수는 1456( $=52 \times 28$ )개이며, 특징점 가능영역으로 선택된 픽셀 수는은 231개이다. 따라서, 영상 전체의 총 픽셀수(=1456)와 비교할 때 15.9%( $=231/1456$ )의 픽셀에 대한 영상 처리만 하면 된다.

[0596] 대각선 방향으로의 RGB 유사도 비교를 수행하지 않고, 수직 방향 및 수평방향의 경계분석을 할 경우 선택되는 특징점 가능영역은 308개의 픽셀로서, 영상전체와 비교하여 21.2%( $=308/1456$ )의 영역만 영상 처리 하면 된다.

[0597] 이렇게 특징점 가능영역이 선택 되었으면, 전체 영상 면적 중에서 적은 면적에 해당되는 면적에 대하여 특징점 분석을 실시하여 최종적으로 특징점을 검출한다.

[0598] 물론 시스템의 처리능력이 충분하다면 위에 설명한 특징점 가능영역 자동 검출 부분을 생략하고, 영상 전체에 대해 아래 기술할 특징점 검출 알고리즘을 적용하여, 해당영상에서 특징점을 직접 검출할 수 있다.

#### 5) 특징점 검출 방법(도 9의 S350)

[0600] 이러한 특징점 가능 영역에서 최종적으로 특징점을 검출하는 도 9의 단계(S350)의 동작은 다음과 같다.

[0601] 일방적으로 영상을 분석하여 어느 한 특징점을 검출하는 방법은 이미 많은 오픈소스들과 무료 공개 알고리즘을 통해 공개되어 있다.

[0602] 본 명세서에는 본 발명을 구체적으로 실시하기 위한 한 예로서, 1988년에 발표된 해리스 코너 알고리즘을 기초로 하여 특징점을 검출 방법에 대해서 설명하고자 한다. 해리스 코너 특징점 검출 알고리즘의 수식은 다음 [수식19]과 같다.

[0603] [수식19]

$$A = \begin{bmatrix} G \otimes d_y^2 & G \otimes d_y d_x \\ G \otimes d_y d_x & G \otimes d_x^2 \end{bmatrix} = \begin{bmatrix} p & r \\ r & q \end{bmatrix}$$

[0604] -  $A$  : 자가공관계(auto-correlation) 행렬, 2차 모멘트 행렬

[0605] -  $G$  : 가우시안 필터

[0606] -  $d_y$  : 영상  $y$ 축 도함수,  $d_y$ 를 구하기 위한 필터  $F_{dy} = [-1 \quad 0 \quad 1]^T$

[0607] -  $d_x$  : 영상  $x$ 축 도함수,  $d_x$ 를 구하기 위한 필터  $F_{dx} = [-1 \quad 0 \quad 1]$

[0608] -  $\otimes$  : 컨볼루션 연산

$$C = (pq - r^2) - k(p+q)^2$$

[0609] -  $C$  : 특징 가능성 값

[0610] -  $k$  : 상수 (일반적으로  $k=0.04$  사용)

[0611] 해리스코너 알고리즘의 경우 현재 위치의 영상의 구조를 수학적으로 해석하여 특징가능성 값을 직접적으로 계산 할 수 있다는 특징이 있다.

[0612] 도 26은 해리스코너 핵심 수식을 이용하여 입력 영상에 대하여 특징 가능성값을 계산해내는 의사코드이다.

[0613] 9라인에서부터 12라인은 가우시안 필터를 정의하는 부분이며, 14라인에서부터 21라인까지가 영상도함수( $d_y$ ,  $d_x$ ,

$d_y d_x$ )를 계산하는 부분이다.

- [0616] 25라인에서부터 36라인까지는 앞서 정의한 가우시안 필터와 도함수들 간의 콘볼루션(convolution)을 수행하고, 37라인에서는 특징 가능성 값을 계산한 후 38라인에서 영상배열로 입력해준 후 결과 값을 돌려준다.
- [0617] 콘볼루션은 영상처리의 영역 연산중 하나로 두 신호를 합쳐서 새로운 신호(결과)를 만드는 수학적 방법이다.
- [0618] 최종적으로 각 RGB채널별로 특징가능성 값을 가지는 영상배열이 얻어진다.
- [0619] 도 27에는 도 26에 도시한 것과 같은 해리스 코너 알고리즘을 사용해서 특징점을 검출하는 순서도가 도시되어 있다.
- [0620] 최종적으로 특징점을 검출하는 동작을 위해 연산부(210)는 도 20의 17라인과 같이 사용자의 수동 동작에 따라 선택된 특징점 가능영역에 대한 정보(예, 영역 위치 및 해당 영상 데이터)나, 도 24의 의사코드를 통해 자동으로 판정된 특징점 가능영역에 대한 정보를 메모리(220)로부터 읽어온다(S351)
- [0621] 다음, 연산부(210)는 입력받은 특징점 가능영역에 대한 정보를 이용하여 직접 해리스 코너 특징점 검출을 할 수 있으나, 영상 처리 속도를 높이고 큰 특징점들을 분석 검출하기 위해, 앞선 영상 데이터 매칭기준 단계에서 사용하였던 것과 동일한 방식으로 영상 데이터에 대한 이미지 다운샘플링을 실시하여 이후의 단계에서는 이미지 다운샘플링된 영상 데이터를 사용하도록 한다(S352).
- [0622] 예를 들어 원영상인  $100 \times 100$  크기의 영상을 분석해야 한다면, 이미지 다운샘플링된 영상인  $25 \times 25$  크기의 영상만을 분석하면 되므로, 연산 동작을 실시하는 연산부(210)의 연산량은  $1/4$ 로 줄어들게 된다.
- [0623] 다음, 연산부(210)는, 해리스코너 특징점 검출을 하기 위해, 각 RGB색상별 영상 도함수를 계산한다(S353).
- [0624] 연산부(210)는 각 RGB색상별 계산된 도함수와 가우시안 필터를 콘볼루션한다(S354).
- [0625] 다음, 연산부(210)는 [수식 19]를 이용하여 각 RGB색상별 특징 가능성값을 취득한다(S355).
- [0626] 그런 다음, 연산부(210)는 계산된 각 RGB색상별 특징 가능성 값을 RGB색상별 가중치를 이용하여 특징가능성 값의 최확값을 얻는다(S356)
- [0627] 이때, 사용되는 RGB색상별 가중치는 각 색상별 똑같이 줄 수 있지만, 예를 들어, 앞선 영상데이터매치기준 단계에서 사용하였던 적분영상을 이용하여 분석되는 영상 영역의 RGB 평균값의 역수를 이용하여 RGB 색상별 가중치를 줄 수 있다.
- [0628] 예를 들어, 분석되고 있는  $25 \times 25$  크기의 픽셀 영역의 RGB 평균값이 (95, 117, 143)이었다면, RGB값의 역수인 ( $1/95=0.010526$ ,  $1/117=0.008547$ ,  $1/143=0.006993$ )를 이용해서 다음과 같이 RGB 색상별 가중치( $W_{RED}$ ,  $W_{GREEN}$ ,  $W_{BLUE}$ )를 계산할 수 있다.
- [0629]  $W_{RED} = 0.010526 / (0.010526 + 0.008547 + 0.006993) = 40.4\%$
- [0630]  $W_{GREEN} = 0.008547 / (0.010526 + 0.008547 + 0.006993) = 32.8\%$
- [0631]  $W_{BLUE} = 0.006993 / (0.010526 + 0.008547 + 0.006993) = 26.8\%$
- [0632] RGB값의 역수로 가중치를 계산하는 이유는 해당 영역의 RGB값들 중 어느 값이 높다면 그 해당영역은 높은 값을 갖는 그 색상이 많다고 볼 수 있다.
- [0633] 즉, 예로든 RGB 평균값이 (95, 117, 143)인 영역에서, 적색(R)의 평균값은 95이고, 녹색(G)의 평균값은 117이며, 청색(B)의 평균값은 144이므로, 해당 영역은 청색 즉, 파란색 계통의 색일 것이다.
- [0634] 따라서, 파란색의 보색계열(노란색계열)이 보다 영상을 설명하게 분석할 수 있게 된다. 따라서 해당 색상의 역수를 RGB 색상별 가중치로 계산하면 해당 색상의 보색이 강조되기 때문에 RGB 색상별 가중치 계산을 RGB색상값의 역수로 사용한다.
- [0635] 예를 들어,  $25 \times 25$  크기의 픽셀 영역을 분석하여 어느 한 픽셀 A에서 적색(Red) 색상 특징 가능성값이 32,341, 녹색(Green) 색상 특징가능성값이 64,651, 청색(Blue) 색상 특징 가능성값이 24,541이었고, 해당  $25 \times 25$  크기의 픽셀 영역의 RGB평균값이 앞에서 예를 든 (95, 117, 143)이라면 각 RGB색상별 가중치는  $W_{RED} = 40.4\%$ ,  $W_{GREEN} = 32.8\%$ ,  $W_{BLUE} = 26.8\%$

$= 32.8\%, W_{BLUE} = 26.8\%$  이고, 이 가중치를 이용한 A픽셀의 특징가능성 값의 최확치는  $32,341 \times 40.4\% + 64,651 \times 32.8\% + 24,541 \times 26.8\% = 40,847$ 이 된다.

- [0636] 큰 값의 특징 가능성 값을 갖는 점은 모서리(코너)에 한 점만 있는 것이 아니라 일정한 영역에 퍼져 있기 때문에 해당 영역에서 한 점을 선택하는 방법이 필요하다.
- [0637] 이를 위해 가장 합리적인 방법은 해당 영역 중의 어느 한 점이 모든 이웃보다 큰 특징 가능성 값을 갖고 있다면 그 한 점이 최대 특징점이 되고, 그렇지 않으면(즉, 비최대이면) 특징점의 가능성이 감소한다 즉, 억제된다. 이러한 방법을 비최대억제(Non-maximum suppression)법이라 한다.
- [0638] 따라서, 연산부(210)는 비최대억제법을 시행하여 최대 특징점을 찾아낸다(S357).
- [0639] 특징점 가능영역이 사용자에 의해 수동으로 선택된 경우(S3581), 사용자가 원하는 특징점은 한 개일 것이다.
- [0640] 따라서 연산부(210)는 사용자가 선택한 특징점 가능영역의 중심점을 기준으로 가장 가까운 특징점을 선택한 후, 해당 특징점 분석에 사용된 영상이 원영상에 대해서 4배로 다운샘플링되었다면,  $7 \times 7$  크기의 영역에 대한 해리스 코너 특징점 분석을 실시하여 최종적으로 원영상에서의 특징점을 검출하여 검출된 특징점에 대한 영상좌표를 원영상에서 취득한다(S3591, S3601, S361).
- [0641] 하지만, 특징점 가능영역이 연산부(210)에 의해 자동으로 선택된 경우(S3582), 해당 특징점 가능영역 내에는 여러개의 특징점이 있을 수 있다.
- [0642] 따라서, 이 경우, 연산부(210)는 특징점 분석을 실시하여 특징 가능성 값이 다른 곳 보다 상당히 큰 특징 가능성 값을 가지고 있는 여러 점들을, 즉 일정 크기 이상의 값을 갖는 픽셀을 전부 특징점으로 하여, 원영상 좌표에서 각 특징점을 검출한 후 검출된 각 특징점에 대한 영상좌표를 원영상에서 취득한다(S3592, S3602, S361).
- [0643] 도 28에는 RGB색상값의 역수를 RGB색상별 가중치로 연산한 후 비최대억제법을 시행하는 의사코드가 포함되어 있다. 도 28의 알고리즘의 동작을 위한 입력 자료는 RGB색상별 특징 가능성값을 가지고 있는 배열로서, 도 2628의 의사코드를 통해 생성된 출력물일 수 있다.
- [0644] 도 28에서, 입력값인  $c_v$ (Critical Value)는 일종의 기준으로서  $c_v$ (Critical Value)에 기재된 해당 값 이하의 특징 가능성 값은 노이즈로서 간주하여 무시하며, 적색(Red), 녹색(Green) 및 청색(Blue)값은 영상처리되는 영역의 RGB 색상별 평균값을 뜻한다. 이 RGB평균 색상값은 영상데이터 매칭기준 단계에서 생성한 적분영상(도 13의 S304 참조)을 이용하여 빠르게 구할 수 있다.
- [0645] 도 28의 2라인부터 10라인까지는 변수를 선언하고, RGB각 색상별 가중치를 계산하는 부분이며, 14라인부터 23라인까지는 8방향 비교를 통해 비최대억제법을 시행하는 부분이다.
- [0646] 본 예에서, 8방향 비교를 하나의 예로 들었지만 이는 시스템 상황에 맞게 비교영역을 좁히거나 넓힐 수 있다.
- [0647] 도 29는 위에서 설명한 특징점 검출 알고리즘을 통한 특징점 검출의 한 예를 나타낸다.
- [0648] 도 29의 (a)는 특징점 가능영역으로 사용자의 선택으로 영역(I1)이 선택되었다면, 도 29의 (b)와 같이 특징점 가능영역으로 영상좌표 (3778, 312) ~ (3878, 412)의 범위에 속하는 영역, 즉  $101 \times 101$  크기의 영역이 특징점 가능영역이 되어 특징점 검출을 실시한다.
- [0649] 도 29의 (b)에 도시한  $101 \times 101$  크기의 영역을 도 29의 (c)와 같이  $25 \times 25$  크기의 영상으로 이미지 다운샘플링 한다.
- [0650] 해리스 코너 특징점 검출 알고리즘을 수행하여 도 29의 (d)와 같이 R(적색), G(녹색) 및 B(청색) 각 색상별 특징가능성값을 계산한 후, RGB평균값에 의한 가중치와 비최대억제법을 통해 도 29의 (e)와 같은 특징가능성맵을 얻게 된다.
- [0651] 생성된 특징가능성맵에서 중심점에서 가장 가까운 특징가능성값을 해당영역의특징점으로 결정하고 대한 영상좌표를 얻는다(F1).
- [0652] 도 29의 (e)에서 얻은 영상좌표는 다운샘플링된 영상의 영상좌표이므로 원영상에서의 영상좌표를 얻기 위해 도 29의 (f)와 같이 원영상의 해당 미소 영역에 대하여 특징점 분석을 한 후 가장 큰 특징가능성값을 갖는 점을 최종 특징점의 영상좌표(F2)로 취득한다.
- [0653] 본 예의 경우, 다운샘플링된 영상의 중심점 좌표는 (3828/4, 362/4) 즉, (957, 90)이며, 중심점에서의 특징 가

능성값이 다른 부분에 비해 가장 크므로 ( $957 \times 4$ ,  $90 \times 4$ ) 즉 원영상 좌표로 (3828, 360)를 중심으로  $7 \times 7$  크기 영역에 대한 특징점 분석을 하여 최종 특징점에 대한 영상좌표(3827, 359)를 결정할 수 있다.

[0654] 위와 같은 방법으로 자동 버튼(U25)을 이용하여 특징점 가능영역이 자동으로 선택되었을 경우도 특징점 가능영역에서 특징점이 큰 몇 개의 영역을 같은 개념으로 계산하면 특징점의 영상좌표를 결정할 수 있다.

[0655] 만약, 본 예의 시스템에 대한 처리능력이 충분하다면, 특징점 가능영역뿐만 아니라 영상 전체에 대해서 특징점 분석 동작을 실시하여, 영상전체에 대한 특징점을 검출할 수 있을 것이다.

#### 6) 특징점 기술 (도 9의 S370)

[0657] 도 9에 도시한 특징점 기술 단계(S370)에 대한 연산부(210)의 동작은 다음과 같다.

[0658] 앞서 설명한 특징점 검출 단계(S350)에서 검출한 특징점은 해당 픽셀의 해리스 코너 자가공관계 행렬, RGB값, 영상좌표 정도의 제한된 정보만을 가지고 있다. 하지만, 이를 정보만으로 다른 영상에서 같은 특징점을 찾는 것은 상당히 어렵다.

[0659] 특징점 기술(describe)이란, 특징점 주변 또는 내부를 분석하여 해당 특징점에 대한 추가적인 정보, 예를 들어, 주변 일정 영역 부분의 RGB값이나 영상 히스토그램 등을 추출하여, 추출된 추가적인 정보를 바탕으로 해당 특징점의 특징을 확정하는 특징 정보를 생성하고, 이 특징 정보를 이용해서 다른 영상에서 같은 특징점을 찾을 수 있게 해주는 것을 뜻한다.

[0660] 이렇게 추출된 정보 또는 특징 정보를 추출해 주는 알고리즘을 기술자(descriptor)라 한다.

[0661] 영상매칭에 사용되는 기술자(descriptor)는 이미 널리 알려진 알고리즘이 개발되어 있으며, 공개된 오픈소스들도 상당히 많이 있다.

[0662] 본 예에서 사용하는 특징점 기술자는 특징점 주변의 RGB픽셀값을 이용하여 영상 정보를 추출하는 방법을 예로 들어 설명하고자 한다.

[0663] 도 30은 본 예에서 사용하는 특징점 기술자 마스크의 한 예이다.

[0664] 도 30의 (a)와 같이, ( $i, i$ )의 영상 좌표를 갖는 특징점인 특징점 픽셀(P)을 중심으로 반지름의 크기가 설정 픽셀수(예, 5픽셀), 즉 특징점 픽셀(P)을 중심으로 하여 설정 픽셀수(제1 설정 픽셀수)만큼 행 방향과 열 방향으로 각각 이격되어 있는 픽셀들을 기준으로 하여 형성된 고리 형태의 마스크인 원형 마스크(예, 제1 원형 마스크)와 반지름의 크기가 설정 픽셀수(예, 3픽셀) 즉 특징점 픽셀(P)을 중심으로 하여 설정 픽셀수(제2 설정 픽셀수)만큼 행 방향과 열 방향으로 각각 이격되어 있는 픽셀들을 기준으로 하여 형성된 고리 형태의 마스크인 원형 마스크(예, 제2 원형 마스크)를 결정한다.

[0665] 다음, 제1 원형 마스크를 형성하는 픽셀들(총 28개의 픽셀)에 대하여 기준 픽셀(0번)을 중심으로 하여 시계 방향으로 순차적으로 1씩 증가하게 번호를 부여한다. 따라서, 제1 원형 마스크의 총 28개의 픽셀들은 0번부터 27번의 번호가 부여된다.

[0666] 이와 유사하게, 제2 원형 마스크를 형성하는 픽셀들(총 16개의 픽셀)에 대하여도 기준 픽셀(0번)을 중심으로 하여 시계 방향으로 순차적으로 1씩 증가하게 번호를 부여하여, 제2 원형 마스크를 이루는 픽셀에게는 0번부터 15번까지의 번호가 부여된다.

[0667] 이때, 제1 및 제2 원형 마스크에서 기준 픽셀(0번)은 특징점 픽셀(P)과 동일한 열에 위치하는 픽셀로서, 기준 픽셀(P)보다 상부쪽의 행에 위치한 픽셀이다.

[0668] 다음, 제1 및 제2 원형 마스크를 이루는 총 44개( $=28+16$ )와 특징점 픽셀(P)을 더한 총 45개의 픽셀에 대한 RGB값들을 이용하여 특징점들을 매칭할 수 있다.

[0669] 도 30의 (b)는 특징점 기술자를 도 29의 (c)영상에 적용한 결과이다.

[0670] 본 예의 특징점 기술자의 주요 특징은 다음과 같다.

[0671] - 제1 및 제2 원형 마스크가 원형의 형상을 갖고 있기 때문에 회전된 영상에 대하여서도 기술자를 회전시킴으로서 특징점의 매칭이 가능하다.

[0672] - 서로 다른 크기의 반지름을 각각 가진 제1 및 제2 원형 마스크를 사용하여 제1 및 제2 원형마스크의 RGB변화

율을 계산하여 영상의 축척변화에 대하여서도 적용이 가능하다.

[0673] - 특징점 기술자에 3차원 아핀변환(Affine Transform)을 적용하여 회전 및 축적이 동시에 변화하는 영상에서도 적용이 가능하다.

[0674] 아핀변환은 n차원 공간의 1차식으로 나타내어지는 점대응  $(x, y) \rightarrow (x', y')$  을 말하며, 일반적으로 병진(translation), 축척(scaling), 회전(rotation), 충밀리기(shearing) 등과 같은 변환을 말한다.

[0675] 본 예의 사진 촬영동작으로 얻어지는 영상의 특징은 앞서 설명한 것과 같이 RGB색상 변화와 축척 및 회전 변화가 거의 없고, 이들 변화가 발생하더라도 사진 촬영 시 위치 정보 측정장치(1000)의 자세와 위치를 추적하여 해당 자세 데이터와 위치 데이터를 기록한다. 따라서, 영상이 촬영되는 순간의 위치 정보 측정장치(1000)의 자세와 위치관계를 특징점 기술자에 3차원 아핀변환을 적용해서 영상에 정확한 매칭이 가능하게 할 수 있다.

[0676] 특징점 기술자의 3차원 아핀 변환은 이미 공개된 공식이고, 3차원 아핀 변환까지 하지 않고, 사진 촬영 동작으로 얻게 되는 영상에 대해서는 특징점 기술자의 회전만으로도 충분히 매칭이 가능하기 때문에 본 예에서의 3차원 아핀변환의 구체적인 적용수식은 생략하도록 하겠다.

[0677] 영상 처리 및 매칭을 빠르게 하기 위해, 특징점 기술을 이용해 원영상에서의 특징 정보와 다운샘플링된 영상에서의 특징 정보인 두개의 특징 정보를 획득하고, 최초 다운샘플링된 영상에서의 특징 정보를 이용하여 빠르게 특징점을 찾고, 찾은 특징점을 이용해 원영상의 좌표를 변환한 후 원영상에서의 특징 정보를 이용하여 미세조정을 할 수 있다.

#### 7) 특징점 매칭 (도 9의 S380)

[0679] 다음, 연산부(210)에서 행해지는 도 9의 특징점 매칭 동작에 대한 설명은 다음과 같다.

[0680] 도 31은 이러한 특징점 매칭 동작을 위한 순서도이다.

[0681] 연산부(210)는 앞서 기술한 특징점 검출 동작과 특징점 기술 동작을 통해 획득되어 메모리(220)에 저장되어 있는 어떤 한 영상(예, 제1 영상)(A영상)에 대한 영상 데이터(제1 영상 데이터)에서 특징점의 영상좌표와 특징점의 특징 정보(즉, 원영상에서의 특징 정보와 다운샘플링된 영상에서의 특징 정보인 두 개의 특징 정보)를 취득한다(S381).

[0682] 다음, 연산부(210)는 획득된 A영상의 특징점에 대한 영상좌표와 앞서 설명한 영상데이터 매칭기준을 통해 나머지 하나의 영상(예, 제2 영상)(B영상)에 대한 영상 데이터(제2 영상 데이터)에서의 특징점 가능영역을 선택한다(S382).

[0683] 영상처리를 빠르게 하기 위해, 연산부(210)는 B영상의 다운샘플링영상을 이용해서 선택 영역 부분에 이미 설명한 해리스코너 알고리즘을 적용해 특징점 분석을 실시하여 특징점을 검출한다(S383).

[0684] 그런 다음, 연산부(210)는 검출된 특징점에 특징점 기술자를 적용해서 B영상의 특징 정보를 취득한다(S384).

[0685] 연산부(210)는 단계(S384)에서 검출된 B영상의 특징 정보와 A영상에서 얻은 특징 정보를 비교하여 특징 정보가 일치하면 매칭된 특징점으로 판정하고, 두 특징 정보가 일치하지 않으면 특징점을 다시 찾는다(S385).

[0686] 본 예의 경우, 특징점 기술자가 제1 및 제2 원형 마스크인 2개의 원형 마스크를 이용한 RGB유사도를 사용하는 매칭이기 때문에, 앞서 설명한 특징점 기술자는 총45개 픽셀의 RGB유사도를 서로 비교하여 그 평균이 일정 유사도(예를 들어 95%) 이상이면 매칭된 특징점이라고 할 수 있다.

[0687] 이 때, B영상에서 얻은 특징점 기술자를 1픽셀씩 회전 시켜봄으로써 가장 높은 RGB유사도를 가진 회전각을 찾을 수 있고, 그 회전각을 유지한 채 특징점의 위치를 1픽셀씩 이동시킴으로서 가장 유사도가 높은 매칭된 특징점을 얻을 수 있다.

[0688] 여기서, 특징점 기술자를 1픽셀 회전시킨다 함은, 예를 들어 제 1원형 마스크의 경우 0번부터 27번까지 픽셀을 매칭할 때(도 30 참조) A영상에서 얻은 특징점 기술자 n번 픽셀과 B영상에서 얻은 특징점 기술자 n+1번 픽셀과의 RGB유사도를 계산하는 것을 뜻한다. 즉 A영상 특징점 기술자 0번 픽셀과, B영상의 특징점 기술자 1번 픽셀의 RGB유사도를 계산하고, A영상 특징점 기술자 1번픽셀과 B영상 특징점 기술자 2번 픽셀의 RGB유사도를 계산하는 식으로 연속적인 비교를 하여 최종적으로 A영상 특징점 기술자 27번 픽셀과 B영상 특징점 기술자 0번 픽셀의 RGB유사도를 계산하는 것을 뜻한다.

- [0689] 특징점 기술자를 회전시켜도 일정 값 이상의 RGB유사도를 얻을 수 없다면, 연산부(210)는 그 특징점은 다른 영상 데이터와 매칭된 정확한 특징점이 아니기 때문에 다시 특징점 검출을 하여 검출된 다른 특징점을 기준으로 매칭작업을 다시 시작한다(S385).
- [0690] 최종적으로 A영상과 B영상에서 특정 정보가 서로 일치되는 특징점을 찾았다면, 연산부(210)는 다운샘플링된 영상에서의 매칭된 영상 좌표이기 때문에 원영상에서의 특징점 검출을 하여 원영상에서 특징점을 검출하고, 검출된 원영상의 특징점에 다시 특징점 기술자를 적용하여, A영상의 원영상에서 얻은 특정 정보와 비교하여 B영상에서 매칭되는 영상좌표를 픽셀단위로 미세조정 한다.
- [0691] 도 32는 특징점 매칭 알고리즘을 통한 특징점 매칭의 한 예를 보여주는 개념도이다.
- [0692] 도 32의 (a)는 특징점 검출을 통해 원영상에서 얻은 (3827, 359)의 영상좌표를 갖는 특징점(즉, 특징점 픽셀)에 특징점 기술자를 적용한 모습이고, 도 32의 (b)는 다운샘플링한 영상에 특징점 기술자를 적용한 모습이다.
- [0693] 도 32의 (c)는 A영상의 특징점 픽셀의 영상 좌표가 (3827, 359)일 때 앞서 계산한  $25 \times 25$  크기의 모자이크 영상으로 영상데이터 매칭기준을 계산한 결과가  $x_{shift} = -700$  픽셀,  $y_{shift} = 0$  픽셀이므로 B영상에서 매칭되는 특징점이 있을 영역(I2)은 (3127, 359)의 특징점 픽셀의 영상좌표를 중심으로  $\pm 100$  픽셀이 적용된 범위의 영역이다.
- [0694] 도 32의 (d)는 특징점이 있는 영역(I2)을 확대한 영상이고 도 32의 (e)는 특징점이 있는 영역(I2)을 다운샘플링한 영상에서 특징점 분석을 하여 특징점을 찾은 모습이다.
- [0695] 특징점 분석 결과, 해당 미소 영상인 도 32의 (e)에서 해리스코너 특징 가능성값이 (33, 37)의 영상좌표를 갖는 점(M01)이 1,612,814로 가장 높았다. 해당 지점(M01)에 대하여 도 32의 (b)의 특징점 기술자를 적용하여 비교한 결과 45개 픽셀의 RGB유사도 평균값이 81.72%가 나왔으며, 특징 기술자를 회전하여 비교 하여도 90%가 넘지 않았다. RGB 유사도 평균이 90%이상이 아니므로 해당 특징점은 매칭점이 아니라고 판단된다.
- [0696] 해리스 코너 특징 가능성값이 723,621로 두 번째로 높은 영상좌표 (32, 23)의 점(M02)에 대하여, 도 32의 (b)의 특징점 기술자를 적용하여 비교한 결과 RGB평균값은 94.15%가 나왔다. 산출된 RGB평균값이 90%이상이므로 이 점(M02)을 매칭특징점이라 가정하고, 특징점 기술자를 회전 비교하여 최대값이 94.15%로 나왔다.
- [0697] 이 후, 한 픽셀씩 이동하여 특징점 RGB유사도를 비교 분석해본 결과 영상좌표가 (33, 24)인 점(즉, 픽셀)일 때가 최대 RGB유사도가 95.97%로 상당히 일치하였다. 따라서, 영상좌표(33, 24)인 점을 매칭 특징점으로 결정한다.
- [0698] 미소영역의 다운샘플링된 영상의 좌표들 중 중심인 (25, 25)의 영상좌표는 원영상 좌표로 (3124, 356)이고 미소영역 영상좌표(33, 24)는 원영상 좌표로 (3156, 352)가 된다.
- [0699] 해당 원영상 좌표를 중심으로  $7 \times 7$  크기의 영역에 대한 특징점 분석을 하고 도 32의 (a)의 원영상에서의 특징점 기술자를 적용 및 비교하여 최종적으로 B영상에서의 매칭 기준점인 영상좌표(3155, 352)가 결정된다.
- [0700] 이와 같은 방법으로 원영상인 A영상에서의 특징점과 특징점 정보(예, 원영상과 비교되는 비교영상인 B영상의 매칭 기준점의 영상좌표)를 획득하고, 추가로 행해진 사진 촬영 동작에 의해 얻어지는 C영상, D영상 또는 E영상이 존재한다면, 같은 방법을 통해, A-C, A-D 또는 A-E와 같이 A영상과 쌍으로 비교하여 각 영상(C영상, D영상, E영상)에 대하여 매칭되는 특징점의 영상좌표를 최종적으로 취득할 수 있다.
- [0701] 3. 측량 부분(P3)
- [0702] 다음, 도 3의 측량부분(P3)에 대하여 설명한다.
- [0703] 1-1) 대상물의 특징점 좌표 파악 (사진 측량부분)
- [0704] 앞서 영상촬영부분(P1)에서 촬영당시의 위치 정보 측정장치(1000)에 대한 자세 데이터와 위치 데이터, 그리고 영상처리부분(P2)에서 획득한 2장 이상의 영상데이터에서 취득한 측정 대상물의 매칭된 특징점의 영상좌표를 활용하여 사진측량학을 적용해 해당 대상물의 특징점 좌표를 계산할 수 있다.
- [0705] 계산과정의 설명을 명확히 하기 위해 2장의 영상을 기준으로 대상물의 특징점 좌표를 취득하는 알고리즘을 설명하고, 2장의 영상에서 대상물의 특징점 좌표를 취득하는 법에 대해 설명이 끝나면, 추가되는 사진을 활용해서 정확도를 높이는 방법에 대해 설명하고자 한다.
- [0706] 도 33은 본 예에서 대상물의 특징점 좌표를 계산하는 연산부(210)의 동작 순서도이다.

- [0707] 각 단계별 세부설명은 해당 단계의 도면 설명에 첨부하도록 한다.
- [0708] 연산부(210)는 해당 대상물에 대하여 제1 영상(예, 1차 사진 촬영에 의해 획득한 영상)에서의 대상물에 대한 영상 데이터(예, 제1 영상 데이터)와 제2 영상(예, 2차 사진 촬영에 의해 획득한 영상)에서의 대상물에 대한 영상 데이터(예, 제2 영상 데이터)를 메모리(220) 혹은 저장부(400)에서 읽어온다(S401-1, S401-2).
- [0709] 대상물의 특징점 좌표를 얻기 위한 데이터로는 제1 및 제2 영상 데이터 이외에 2장의 영상데이터에 대한 촬영 당시 해당 기기[즉, 위치 정보 측정장치(1000)]의 자세 데이터( $\omega, \phi, \kappa$ )와 위치 데이터(X, Y, Z) 및 좌표를 얻고자 하는 대상물의 특징점에 대한 영상좌표가 필요하다.
- [0710] 이로 인해, 연산부(210)는 도 3의 단계(P2)에서 획득한 각 촬영 동작 시의 영상 좌표에서 촬영부(110)의 내부표정(interior orientation) 과정을 통해 렌즈 왜곡이 제거된 보정된 영상좌표를 얻는다(S402-1, S402-2)(도 34 참조).
- [0711] 그런 다음, 연산부(210)는 보정하여 얻은 영상좌표를 근접사진측량학을 적용하기 위해 사진좌표로 변경한다(S403-1, S403-2)(도 35참조).
- [0712] 본 예에서, 촬영부(110)의 내부표정 과정은 정확도를 향상시켜주기 위한 것이므로, 경우에 따라 내부 표정 과정은 생략될 수 있다.
- [0713] 다음, 연산부(210)는 각 사진 촬영 시의 제1 및 제2 영상에 대해 위치 정보 측정장치(1000)의 자세 데이터에 포함된 회전각( $\omega$ )을 이용하여 사진좌표를 회전 변환한다(S404-1, S404-2)(도 36 참조).
- [0714] 그런 다음, 연산부(210)는 회전 변환된 제1 및 제2 영상의 사진좌표와 촬영부(110)의 초점거리를 이용하여 대상물의 특징점까지의 영상 상에서의 미소 방위각과 미소 고저각을 계산한다(S405-1, S405-2)(도 37 참조).
- [0715] 연산부(210)는 계산된 미소 방위각과 미소 고저각 그리고, 촬영 당시의 기기에 대한 자세 데이터 중 방위각( $\kappa$ )과 고저각( $\phi$ )을 이용하여, 각 촬영위치에서부터 대상물의 특징점까지의 방위각과 고저각을 계산한다(S406-1, S406-2)(도 38참조).
- [0716] 다음, 연산부(210)는 제1 및 제2 영상을 획득한 촬영위치의 좌표(즉, 위치 데이터)와 대상물의 특징점까지의 방위각과 고저각을 이용하여 대상물의 특징점 좌표를 구한다(S407)(도 38참조).
- [0717] 이들 각 단계에 대한 세부설명은 다음과 같다.
- [0718] (1) 카메라 내부표정
- [0719] 촬영부(210)인 카메라의 내부표정 즉, 카메라 내부표정에 대하여 도 34를 참고로 하여 설명한다.
- [0720] 도 34는 촬영부(110)의 렌즈에 의해 발생하는 방사왜곡(radial distortion)[도 34의 (a)]과 접선왜곡(tangential distortion)[도 34의 (b)]의 개념도이다.
- [0721] 방사왜곡은 카메라에 설치된 볼록렌즈의 굴절률에 의해 발생되는 렌즈왜곡 중 하나로 영상의 왜곡정도가 중심에서의 거리에 의해 결정되는 왜곡이고, 접선왜곡은 카메라의 제조 과정에서 카메라의 렌즈와 이미지 센서의 수평이 맞지 않거나, 또는 렌즈 자체의 중심이 맞지 않아서 발생하는 렌즈왜곡 중하나로, 타원형 형태로 발생하는 왜곡이다.
- [0722] 카메라 내부표정이란 촬영당시의 광속의 기하 상태를 재현하는 작업으로 기준점 위치, 렌즈의 왜곡, 촬영부(110)의 초점거리와 영상의 주점을 결정하여 부가적으로 영상의 오차를 보정하여 영상좌표의 정확도를 향상시키는 것을 말한다.
- [0723] 카메라 내부표정의 주요 요소는 방사 왜곡을 보정해 주기 위해 결정되는 일련의 상수인 방사왜곡계수, 렌즈의 접선 왜곡을 보정해 주기 위해 결정되는 일련의 상수인 접선왜곡계수, 사진의 주점 등이 있으며, 위치 정보 측정장치(1000)에 사용되는 카메라 제조사마다 결정되는 특성으로서, 체스보드 촬영을 통해 사전에 보정계수들이 계산한 후 계산된 보정 계수들을 시스템의 메모리(220) 등에 저장한다. 따라서, 연산부(210)는 시스템에 사용되는 카메라(110)의 종류에 따라 각 카메라(110)에 맞는 보정 계수들을 필요에 따라 읽어와 사용할 수 있다.
- [0724] 예를 들어, 위치 정보 측정장치(1000)가 안드로이드 기반 스마트폰의 경우, 각 스마트폰 모델별 카메라 내부표정요소를 데이터베이스화 하여 스마트폰에 저장하여, 본 예의 구성이 설치된 스마트폰 모델에 따른 내부표정 계

수를 적용하여 계산한다.

[0725] 발매되지 않은 스마트폰의 내부표정요소들은 서버(2000)를 통한 스마트폰 업데이트나 내부표정 요소만을 서버에서 전송받아 저장되므로 필요시 사용되도록 한다.

[0726] 카메라 내부표정의 과정은 계산의 정확도를 크게 향상시켜줄 수 있는 과정이 아니기 때문에, 카메라 내부표정 요소를 알 수 없는 상황인 경우는 내부표정계수를 이용한 동작은 생략될 수 있다.

[0727] 카메라 내부표정 요소인 사진의 주점, 접선왜곡계수 및 방사왜곡계수를 산출하는 수식은 이미 공개되어 있으므로, 계산할 수 있는 비상용 프로그램이나, 오픈 소스들이 많기 때문에 내부표정 요소를 구하는 수식은 본 명세서에서 생략한다.

[0728] 카메라 내부표정 요소를 이용하여 사진좌표를 보정하는 공식은 다음 [수식 20]과 같다.

[수식20]

$$\begin{bmatrix} x_{n\_u} \\ y_{n\_u} \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & skew \cdot f_x & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x_{p\_u} \\ y_{p\_u} \\ 1 \end{bmatrix} \quad ①$$

$$\begin{bmatrix} x_{n\_d} \\ y_{n\_d} \end{bmatrix} = (1 + k_1 r_u^2 + k_2 r_u^4 + k_3 r_u^6) \begin{bmatrix} x_{n\_u} \\ y_{n\_u} \end{bmatrix} + \begin{bmatrix} 2p_1 x_{n\_u} y_{n\_u} + p_2 (r_u^2 + 2x_{n\_u}^2) \\ p_1 (r_u^2 + 2y_{n\_u}^2) + 2p_2 x_{n\_u} y_{n\_u} \end{bmatrix} \quad ②$$

[0732] (단,  $r_u^2 = x_{n\_u}^2 + y_{n\_u}^2$ )

$$\begin{bmatrix} x_{p\_d} \\ y_{p\_d} \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & skew \cdot f_x & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{n\_d} \\ y_{n\_d} \\ 1 \end{bmatrix} \quad ③$$

[0734] - 카메라 특성으로 구해지는 계수[즉, 위치 정보 측정장치(1000)에 포함되거나 서버(2000)를 통해 제공 되는 계수]는 다음과 같다.

[0735]  $f_x, f_y$ : x축, y축 초점거리

[0736]  $skew$ : 비대칭계수 (현재의 카메라는 0으로 계산하여도 됨)

[0737]  $c_x, c_y$ : 주점좌표

[0738]  $k_1, k_2, k_3$ : 접선왜곡계수

[0739]  $p_1, p_2$ : 방사왜곡계수

[0740] - 사진좌표의 의미

[0741]  $x_{p\_d}, y_{p\_d}$ : 왜곡된 영상좌표 (영상 처리부분에서 얻게 되는 영상좌표)

[0742]  $x_{p\_u}, y_{p\_u}$ : 왜곡되지 않은 영상좌표 (보정해서 구해야 할 영상좌표)

[0743]  $x_{n\_d}, y_{n\_d}$ : 왜곡된 정규이미지좌표 (정규이미지 좌표계는 카메라 내부표정 과정상 계산의 편의를 위해 도입한 좌표계로서, 카메라의 내부표정의 영향을 제거한 이미지 좌표계로, 좌표계의 단위가 없으며, 카메라 초점 거리가 1인 가상의 이미지 평면을 정의하는 좌표계를 말한다. 원래의 이미지 평면을 평행이동 시켜서 카메라 초점과의 거리가 1인 지점으로 옮겨놓은 이미지 평면이다. 따라서, 정규이미지 좌표는 정규이미지 좌표계의 좌표이다.)

[0744]  $x_{n\_u}, y_{n\_u}$ : 왜곡되지 않은 정규이미지좌표

[0745] 첨자의 의미

[0746]  $p$ : projection 투영된 좌표(영상좌표),  $n$ : normalized 정규화된 좌표

[0747]  $u$ : undistorted 왜곡되지 않은 좌표,  $d$ : distorted 왜곡된 좌표

[0748] 본 예에서는 카메라 내부표정을 통해 왜곡된 영상좌표로 왜곡되지 않은 영상좌표를 구하기 위해서는 좌표관계식을 이용한 시행착오법을 사용한다.

[0749] 시행착오법은 수치계산법의 일종으로 어느 변수  $x$ 에 대하여 관계식  $f(x)=0$ 이 있어 이 관계를 만족하는  $x$ 를 구하고자 하나, 해석적으로 풀 수 없을 때,  $x$ 에 여러 값을 대입하여  $f(x)$ 값을 계산하고 그 값이 0이 될 때까지 계산을 반복하는 수치계산법이다.

[0750] 구하고자 하는 왜곡되지 않은 영상좌표( $x_{p\_u}, y_{p\_u}$ )를 구하기 위해, 일단 영상 처리부분(P2)에서 구한 왜곡된 영상좌표( $x_{p\_d}, y_{p\_d}$ )를 왜곡되지 않은 영상좌표( $x_{p\_u}, y_{p\_u}$ )으로 가정하고 [수식20]의 ①식, ②식 및 ③식을 통해 계산하여 새롭게 계산된 왜곡된 영상좌표( $x'_{p\_d}, y'_{p\_d}$ )와 영상처리부분(P1)에서 구했던 왜곡된 영상좌표( $x_{p\_d}, y_{p\_d}$ )의 좌표 차이를 반영하여 새롭게 왜곡되지 않은 영상좌표( $x_{p\_u}, y_{p\_u}$ )를 가정하고 다시 [수식20]의 ①식, ②식 및 ③식을 통해 계산한다.

[0751] 이런 반복계산을 통해 새롭게 계산된 왜곡된 영상좌표( $x'_{p\_d}, y'_{p\_d}$ )와 이미 알고 있는 왜곡된 영상좌표( $x_{p\_d}, y_{p\_d}$ ) 차이가 일정 허용오차 범위 안에 존재하면 계산을 멈추고, 최종으로 왜곡되지 않은 영상 좌표으로 가정된 왜곡되지 않은 영상 좌표( $x_{p\_u}, y_{p\_u}$ )를 최종 보정된 영상좌표로 정한다.

[0752] 예를 들어, 영상처리부분(P2)을 통해 취득한 영상좌표가 (3826, 356)이었다면, 이 좌표만으로 바로 보정된 영상 좌표를 구하는 것은 어렵다.

[0753] 따라서, 먼저, (3826, 356)의 영상좌표를 왜곡이 안된 영상좌표( $x_{p\_u} = 3826, y_{p\_u} = 356$ )라 가정하고 [수식20]의 ①식, ②식 및 ③식을 통해 계산된 왜곡된 영상좌표가 (3836, 366)이었다면, 왜곡된 영상좌표(3836, 366)와 왜곡이 안된 영상좌표(3826, 356)간의 차이는 (+10, +10)이다.

[0754] 따라서, 이 차이(+10, +10)를 왜곡이 안된 영상좌표(3826, 356)에 반영하여 왜곡이 안된 영상좌표( $x_{p\_u}, y_{p\_u}$ )를 다시 (3816, 346)로 가정한 후 위와 같은 계산을 반복한다.

[0755] 다시 계산한 왜곡된 영상좌표가 (3825, 354)이었다면 이미 알고 있는 왜곡된 영상좌표 (3826, 356)와의 차이가 (-1, -2)이므로, 현재 왜곡이 안된 영상좌표인 (3816, 346)에 이 차이를 반영하여 새로운 왜곡이 안된 영상좌표( $x_{p\_u}, y_{p\_u}$ )를 (3817, 348)로 가정한 후 다시 계산한다.

[0756] 다시 계산한 왜곡된 영상좌표가 (3826.3, 356.4) 이었다면 이미 알고 있는 왜곡된 영상좌표(3826, 356)와의 차이가 (+0.3, +0.4)이고, 이 차이가 허용오차 범위(예를 들어, 1픽셀) 이내라면 최종으로 가정했던 (3817, 348)이 왜곡이 보정된 영상좌표가 된다.

## [0757] (2) 영상좌표의 사진좌표 변환

[0758] 이미 기술한 것처럼, 일반적인 영상의 특징점 검출, 특징점 기술, 특징점 매칭 같은 경우 계산을 용이하게 하기 위해 좌표값에 음의 값(-)이 나오지 않도록 영상의 좌측 최상단을 (0,0)으로 하는 좌표체계를 사용한다.

[0759] 하지만, 근접사진측량학에서는 대상물의 특징점 좌표를 구하기 위해 사진의 주점에서부터의 거리관계가 중요하기 때문에 사진(즉 사진 촬영동작에 획득된 영상)의 주점을 중심으로 한 좌표체계를 사용하여야 한다.

[0760] 본 명세서에서는 영상의 좌측 최상단을 (0, 0)으로 하는 좌표 체계와 사진의 주점을 중심으로 하는 좌표체계를 구분하기 위해, 영상의 특징점 검출, 특징점 기술, 특징점 매칭에 사용되는 좌표는 영상좌표라고 하고, 근접사진측량학에 적용하는 좌표는 사진좌표라 정의한다.

[0761] 도 35는 영상좌표와 사진좌표에 대한 개념도이다.

- [0762] 도 35의 (a)는 영상좌표로 되어 있는 영상(I402-1)을 사진좌표(I402-2)로 변환한 개념도이다.
- [0763] 도 35의 (a)는 좌측상단의 중심인 좌표체계를 사진 주점을 중심으로 하는 좌표체계로 변환하는 것이다.
- [0764] 도 35의 (b)는 영상좌표의 개념으로서 해상도의 크기가  $5313 \times 2989$ 인 영상을 영상좌표로 표현한 그림이다. 좌측 상단을 0으로 한 좌표체계이며, 정수로 표현되는 각 픽셀들의 좌표[예, (0, 0)]는 사각형 안에 표현하였다.
- [0765] 일반적으로 영상좌표는 픽셀 단위 정수로 표현되는 좌표이나, 실수 좌표로 표현 할 경우 픽셀의 좌측 상단점 기준이 된다.
- [0766] 예를 들어 (0,0)픽셀 중심(IS1)의 실수좌표는 (0.5,0.5)가 되고, (5312,2988) 픽셀 중심(IS2)의 실수좌표는 (5312.5,2988.5)가 된다.
- [0767] 도 35의 (c)는 사진좌표의 개념도로서, 예를 들어 사진주점이 영상좌표로 (2656.5, 1494.5)일 때의 사진좌표 체계이다.
- [0768] 도 35의 (c)에서 사각형 안에 표현된 숫자들은 각 픽셀의 중심점의 좌표이다.
- [0769] 예를 들어 도 35의 (b)에서 영상좌표가 (0.5,0.5)인 점(IS1)은, 사진좌표로는 (-2656, 1494)의 좌표를 갖고 있는 점(PS1)에 해당하고, 영상좌표가 (5312.5,2988.5)인 점(IS2)는 사진좌표(PS2)는 (2656, -1494)의 좌표를 갖는 점(PS2)이 된다.
- [0770] 사진주점은 카메라 내부표정요소 중 하나로 앞서 설명한 카메라 내부표정 과정에서 얻어지는 계수로, 위치 정보 측정장치(1000)에 데이터베이스 형태로 저장되어 있거나 서버(2000)로부터 다운받아 사용될 수 있다. 하지만, 사진주점의 계산이 불가능하거나 제공되지 않는 경우, 영상의 중심점을 사진 주점으로 사용할 수 있다. 다음 [수식21]은 사진 주점을 이용하여 영상좌표를 사진좌표로 변환하는 공식이다.
- [0771] [수식21]
- [0772]  $x_p = x_i - c_x$
- [0773]  $y_p = (y_i - c_y) \times -1$
- [0774]  $(c_x, c_y)$ : 사진주점의 영상좌표,  $(x_i, y_i)$ : 영상좌표,  $(x_p, y_p)$ : 사진좌표
- [0775] (대안적인 예에서  $c_x =$ 영상의 폭/2,  $c_y =$ 영상의 높이/2 일 수 있음)
- [0776] (3) 위치 정보 측정장치(1000)의 회전각을 이용한 사진좌표의 회전변환
- [0777] 도 36은 위치 정보 측정장치(1000)의 회전각을 이용하여 사진좌표를 회전 변환하는 개념도이다.
- [0778] 사진좌표의 회전 변환 동작은 도 36의 (a)와 같이 회전되어 촬영된 영상을 도 36의 (b)와 같이 지면과 완전히 수평되게 사진주점을 중심으로 회전 변환시키는 것이라 할 수 있다.
- [0779] 이러한 회전변환 동작을 수행하여 대상물을 지면에 수행하게 위치시켜야 수평상태에서의 대상물의 고저각과 방위각을 계산하여 대상물의 특징점 좌표를 구할 수 있다.
- [0780] 본 예의 위치 정보 측정장치(1000)는 일반적으로 사용자가 위치 정보 측정장치(1000)을 손으로 들고 사진 촬영을 실시하는 경우가 대부분이기 때문에 위치 정보 측정장치(1000)의 완전한 수평이나 수직을 유지하면서 촬영하는 것은 실질적으로 불가능하다.
- [0781] 하지만 본 예의 영상촬영부분(P1)에서 위치 정보 측정장치(1000) 회전각을 측정하므로, 이때 측정된 회전각을 이용하여 사진좌표를 회전변환시켜 수평상태의 사진좌표 취득이 가능하다.
- [0782] 다음 [수식22]은 수평상태에서의 사진좌표를 취득하는 수식이다.
- [0783] [수식22]
- [0784]  $(x_t, y_t)$ : 회전변환 전의 대상물 t의 사진좌표
- [0785]  $(x'_t, y'_t)$ : 회전변환 후의 대상물 t의 사진좌표

[0786]  $\omega$ : 촬영당시의 기기의 회전각

$$\begin{bmatrix} x'_t \\ y'_t \end{bmatrix} = \begin{bmatrix} \cos\omega & -\sin\omega \\ \sin\omega & \cos\omega \end{bmatrix} \begin{bmatrix} x_t \\ y_t \end{bmatrix}$$

[0788] (4) 보정된 사진좌표를 이용하여 촬영위치부터 대상물까지의 미소 방위각과 미소 고저각의 계산

[0789] 도 37은 촬영부(110)의 초점거리와 보정된 사진좌표를 이용하여 촬영위치로부터 대상물까지의 미소 방위각( $\Delta\kappa$ ) 및 미소 고저각( $\Delta\phi$ )을 구하는 개념도이다.

[0790] 도 37에서, 미소 방위각( $\Delta\kappa_{at}, \Delta\kappa_{bt}$ )은 사진좌표를 통해 각 영상의 사진주점( $O_A, O_B$ )로부터 대상물(T)까지의 방위각 변화량을 뜻한다.

[0791] 또한, 미소 고저각( $\Delta\phi_{at}, \Delta\phi_{bt}$ )은 사진좌표를 통해 각 영상의 사진주점( $O_A, O_B$ )로부터 대상물(T)까지의 고저각 변화량을 뜻한다.

[0792] 미소 방위각과 미소 고저각은 다음 [수식23]으로 구할 수 있다.

[0793] [수식23]

[0794] - A영상에서의 미소 방위각 및 미소 고저각

$$\Delta\kappa_{at} = \tan^{-1} \frac{x_{at}}{f_x}$$

[0795] ( $x_{at}$ : 대상물(T)의 A영상에 대한 사진좌표 중 x좌표,  $f_x$ : x축 초점거리)

$$\Delta\phi_{at} = \tan^{-1} \frac{y_{at}}{f_y}$$

[0796] ( $y_{at}$ : 대상물(T)의 A영상에 대한 사진좌표 중 y좌표),  $f_y$ : y축 초점거리)

[0797] - B영상에서의 미소 방위각 및 미소 고저각

$$\Delta\kappa_{bt} = \tan^{-1} \frac{x_{bt}}{f_x}$$

[0798] ( $x_{bt}$ : 대상물(T)의 B영상에 대한 사진좌표 중 x좌표,  $f_x$ : x축 초점거리)

$$\Delta\phi_{bt} = \tan^{-1} \frac{y_{bt}}{f_y}$$

[0799] ( $y_{bt}$ : 대상물(T)의 B영상에 대한 사진좌표 중 y좌표),  $f_y$ : y축 초점거리)

[0800] (5) 촬영위치에서부터 대상물까지의 방위각과 고저각 계산

[0801] 촬영위치 A에서의 방위각( $\kappa_A$ )과 고저각( $\phi_A$ )은 도 37에서 촬영위치 A의 시준축( $\overline{AO_A}$ )이 평면직교좌표계에서 이루는 각도이고, 촬영위치 B에서의 방위각( $\kappa_B$ )과 고저각( $\phi_B$ )는 도 37에서 촬영위치 B의 시준축( $\overline{BO_B}$ )이 평면직교좌표계에서 이루는 각도이다.

[0802] 따라서 우리가 알고자 하는 촬영위치 A에서 대상물(T)까지의 방위각( $\kappa_{AT}$ )과 고저각( $\phi_{AT}$ ) 그리고 촬영위치 B에서 대상물(T)까지의 방위각( $\kappa_{BT}$ )과 고저각( $\phi_{BT}$ )은 다음 [수식24]로 구할 수 있다.

[0803] [수식24]

[0804] - 촬영위치A에서 대상물T까지의 방위각( $\kappa_{AT}$ )과 고저각( $\phi_{AT}$ )

[0809]  $\kappa_{AT} = \kappa_A + \Delta\kappa_{at}$

[0810] ( $\kappa_A$ : 촬영위치A의 방위각,  $\Delta\kappa_{at}$ : A영상에서의 미소 방위각)

[0811]  $\phi_{AT} = \phi_A + \Delta\phi_{at}$

[0812] ( $\phi_A$ : 촬영위치A의 고저각,  $\Delta\phi_{at}$ : A영상에서의 미소 고저각)

[0813] - 촬영위치B에서 대상물T까지의 방위각( $\kappa_{BT}$ )과 고저각( $\phi_{BT}$ )

[0814]  $\kappa_{BT} = \kappa_B + \Delta\kappa_{bt}$

[0815] ( $\kappa_B$ : 촬영위치B의 방위각,  $\Delta\kappa_{bt}$ : B영상에서의 미소 방위각)

[0816]  $\phi_{BT} = \phi_B + \Delta\phi_{bt}$

[0817] ( $\phi_B$ : 촬영위치B의 고저각,  $\Delta\phi_{bt}$ : B영상에서의 미소 고저각)

#### [0818] (6) 대상물의 특징점 좌표취득

[0819] 도 38은 공간상의 두 점(예, A, B)의 위치를 알고 그 두 점에서 바라보는 대상물(예, T)의 방위각과 고저각을 알고 있을 때 대상물의 특징점 좌표를 취득하기 위한 개념도이다.

[0820] 공간상의 두 점의 위치, 즉 촬영위치 A의 좌표( $X_A, Y_A, Z_A$ )와 B의 좌표( $X_B, Y_B, Z_B$ )를 알고, 앞선 [수식 24]에서 구한 촬영위치A에서 대상물(T)까지의 방위각( $\kappa_{AT}$ ) 및 고저각( $\phi_{AT}$ ), 그리고 촬영위치 B에서 대상물(T)까지의 방위각( $\kappa_{BT}$ ) 및 고저각( $\phi_{BT}$ )을 이용하여 대상물(T)의 좌표를 구하기 위해, Z값이 0인 XY평면상으로 투영한 후 방위각  $\kappa_{AT}$  와  $\kappa_{BT}$ 를 이용해서 대상물(T)의 X좌표  $X_T$  와 Y좌표  $Y_T$ 를 구한다.

[0821]  $X_T$  와  $Y_T$  를 구한 후 고저각  $\phi_{AT}$  와  $\phi_{BT}$  를 이용해서 최종으로 대상물의 Z좌표인  $Z_T$ 를 구한다.

[0822] 다음 [수식25]은 두 점(A, B)의 위치를 알고 두 점(A, B)에서 바라보는 방위각과 고저각을 알고 있을 때 대상물(T)의 좌표( $X_T, Y_T, Z_T$ )를 구하는 일반식이다.

[0823] [수식25]

$$X_T = \frac{\tan(90^\circ - \kappa_{AT})X_A - \tan(90^\circ - \kappa_{BT})X_B - (Y_A - Y_B)}{\tan(90^\circ - \kappa_{AT}) - \tan(90^\circ - \kappa_{BT})}$$

[0824]  $Y_T = \tan(90^\circ - \kappa_{AT})X_T - \tan(90^\circ - \kappa_{AT})X_A + Y_A$

[0825]  $Z_T = \frac{Z_{TA} + Z_{TB}}{2}$

[0826]  $Z_{TA} = \overline{A'T'} \tan \phi_{AT} + Z_A \quad (\text{단}, \quad \overline{A'T'} = \sqrt{(X_A - X_T)^2 + (Y_A - Y_T)^2})$

[0827]  $Z_{TB} = \overline{B'T'} \tan \phi_{BT} + Z_B \quad (\text{단}, \quad \overline{B'T'} = \sqrt{(X_B - X_T)^2 + (Y_B - Y_T)^2})$

[0828] 1-2) 공선조건 행렬식을 이용해서 대상물(T)의 좌표( $X_T, Y_T, Z_T$ )를 구하는 방법

[0829] 1-1)를 참고로 하여 설명한 지금까지의 대상물(T)의 좌표를 구한 방법은 사진측량학의 공선조건을 기하학적인 방법으로 풀어서 계산한 방법이다.

[0830] 일반적인 사진측량학에서는 행렬식을 이용해서 카메라 즉, 촬영부(110)의 외부표정 요소, 즉 카메라의 위치 데이터와 자세 데이터를 알고 두 영상에서 동일대상물에 대한 사진좌표를 알 때, 공선조건식을 행렬식으로 치환하여 대상물의 특징점 좌표를 계산할 수 있다.

[0832] 다음 [수식26]은 공선조건식을 행렬식으로 풀어서 대상물(T)의 좌표를 계산하는 일반식이다.

[0833] [수식26]

[0834] (알고 있는 상수)

[0835] - 촬영위치 A의 좌표 ( $X_A, Y_A, Z_A$ )와 자세 데이터( $\omega_A, \phi_A, \kappa_A$ )

[0836] - 촬영위치 B의 좌표 ( $X_B, Y_B, Z_B$ )와 자세 데이터( $\omega_B, \phi_B, \kappa_B$ )

[0837] - 대상물(T)의 A영상에서의 사진좌표 ( $x_{at}, y_{at}$ )

[0838] - 대상물(T)의 B영상에서의 사진좌표 ( $x_{bt}, y_{bt}$ )

[0839] - 카메라의 초점거리:  $f$

[0840] (계산과정)

[0841] (1) 대상물T의 초기 값 설정 ( $X_T, Y_T, Z_T$ ): 기하학적인 방법을 이용해서 구한 값을 사용하거나 아래 수식으로 가정한다.

$$X_T = \frac{(X_A + X_B)}{2}$$

$$Y_T = \frac{(Y_A + Y_B)}{2}$$

$$Z_T = \frac{(Z_A + Z_B)}{2}$$

[0845] (2) 디자인 매트릭스  $B$ , 오차 매트릭스  $C$ 의 구성은 다음과 같다.

$$B = \begin{bmatrix} b_{11}^A & b_{12}^A & b_{13}^A \\ b_{21}^A & b_{22}^A & b_{23}^A \\ b_{11}^B & b_{12}^B & b_{13}^B \\ b_{21}^B & b_{22}^B & b_{23}^B \end{bmatrix}, \quad C = \begin{bmatrix} x_{at} + f \cdot r_a/q_a \\ y_{at} + f \cdot s_a/q_a \\ x_{bt} + f \cdot r_b/q_b \\ y_{bt} + f \cdot s_b/q_b \end{bmatrix}$$

$$R_{\omega A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega_A & \sin\omega_A \\ 0 & -\sin\omega_A & \cos\omega_A \end{bmatrix}, \quad R_{\phi A} = \begin{bmatrix} \cos\phi_A & 0 & \sin\phi_A \\ 0 & 1 & 0 \\ -\sin\phi_A & 0 & \cos\phi_A \end{bmatrix}, \quad R_{\kappa A} = \begin{bmatrix} \cos\kappa_A & \sin\kappa_A & 0 \\ -\sin\kappa_A & \cos\kappa_A & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_{\omega B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega_B & \sin\omega_B \\ 0 & -\sin\omega_B & \cos\omega_B \end{bmatrix}, \quad R_{\phi B} = \begin{bmatrix} \cos\phi_B & 0 & \sin\phi_B \\ 0 & 1 & 0 \\ -\sin\phi_B & 0 & \cos\phi_B \end{bmatrix}, \quad R_{\kappa B} = \begin{bmatrix} \cos\kappa_B & \sin\kappa_B & 0 \\ -\sin\kappa_B & \cos\kappa_B & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_{all\_A} = R_{\kappa A} \times R_{\phi A} \times R_{\omega A}, \quad R_{all\_B} = R_{\kappa B} \times R_{\phi B} \times R_{\omega B}$$

$$dXYZ_A = \begin{bmatrix} X_T - X_A \\ Y_T - Y_A \\ Z_T - Z_A \end{bmatrix}, \quad dXYZ_B = \begin{bmatrix} X_T - X_B \\ Y_T - Y_B \\ Z_T - Z_B \end{bmatrix}$$

$$\begin{bmatrix} r_a \\ s_a \\ q_a \end{bmatrix} = dXYZ_A \times R_{all\_A}, \quad \begin{bmatrix} r_b \\ s_b \\ q_b \end{bmatrix} = dXYZ_B \times R_{all\_B}$$

$$b_{11}^A = f/q_a^2 \times (r_a \times R_{all\_A}(3,1) - q_a \times R_{all\_A}(1,1))$$

[0853]  $b_{12}^A = f/q_a^2 \times (r_a \times R_{all\_A}(3,2) - q_a \times R_{all\_A}(1,2))$

[0854]  $b_{13}^A = f/q_a^2 \times (r_a \times R_{all\_A}(3,3) - q_a \times R_{all\_A}(1,3))$

[0855]  $b_{21}^A = f/q_a^2 \times (s_a \times R_{all\_A}(3,1) - q_a \times R_{all\_A}(2,1))$

[0856]  $b_{22}^A = f/q_a^2 \times (s_a \times R_{all\_A}(3,2) - q_a \times R_{all\_A}(2,2))$

[0857]  $b_{23}^A = f/q_a^2 \times (s_a \times R_{all\_A}(3,3) - q_a \times R_{all\_A}(2,3))$

[0858] (단,  $R_{all\_A}(x, y)$ :  $R_{all\_A}$  행렬의 x행 y열의 값)

[0859]  $b_{11}^B = f/q_b^2 \times (r_b \times R_{all\_B}(3,1) - q_b \times R_{all\_B}(1,1))$

[0860]  $b_{12}^B = f/q_b^2 \times (r_b \times R_{all\_B}(3,2) - q_b \times R_{all\_B}(1,2))$

[0861]  $b_{13}^B = f/q_b^2 \times (r_b \times R_{all\_B}(3,3) - q_b \times R_{all\_B}(1,3))$

[0862]  $b_{21}^B = f/q_b^2 \times (s_b \times R_{all\_B}(3,1) - q_b \times R_{all\_B}(2,1))$

[0863]  $b_{22}^B = f/q_b^2 \times (s_b \times R_{all\_B}(3,2) - q_b \times R_{all\_B}(2,2))$

[0864]  $b_{23}^B = f/q_b^2 \times (s_b \times R_{all\_B}(3,3) - q_b \times R_{all\_B}(2,3))$

[0865] (단,  $R_{all\_B}(x, y)$ :  $R_{all\_B}$  행렬의 x행 y열의 값)

[0866] (3) 보정량 행렬 계산은 다음과 같다.

[0867]  $\Delta = (B^T \times B)^{-1} \times B^T \times C$

[0868] (4) 이때,  $|\Delta| > small$  이면 ( $small$ : 정확도에 기준이 되는 값( $10^{-3}$ )의 경우 mm정확도) 아래의 수학식을 이용하여 다시 계산한 후 (2)과정부터 다시 반복한다.

[0869]  $X_T = X_T + \Delta(1,1)$  ( $\Delta(1,1)$ :  $\Delta$  행렬 1행 1열의 값)

[0870]  $Y_T = Y_T + \Delta(2,1)$  ( $\Delta(2,1)$ :  $\Delta$  행렬 2행 1열의 값)

[0871]  $Z_T = Z_T + \Delta(3,1)$  ( $\Delta(3,1)$ :  $\Delta$  행렬 3행 1열의 값)

[0872] (5)  $|\Delta| < small$  이면 동작은 종료된 후, 마지막으로 계산된  $(X_T, Y_T, Z_T)$ 가 대상물의 특징점 좌표가 된다.

[0873] 공선조건의 기하조건으로 계산하는 [수식23]~[수식25]식을 사용하여 대상물의 특징점 좌표를 취득할 경우 대상물이 한 점일 때는 빠르게 계산할 수 있으나, 여러 점을 계산할 경우 여러 점에 대해서 똑같은 계산을 반복 수행 하여야 한다.

[0874] [수식26]의 경우 수식 계산이 복잡해 보이나 복수의 여러 특징점에 대한 좌표를 동시에 계산할 경우 행렬식을 이용하기 때문에 단 한 번의 계산으로 여러 점을 동시에 계산할 수 있다.

[0875] 하지만, 각 풀이 방식에 따른 정확도는 기하조건을 이용하는 것과 행렬식을 이용하는 것의 차이는 크지 않다.

[0876] [수식26]과 같은 공선조건 행렬식 외에 공면조건 행렬식을 이용해서 구하는 방법도 있으나, 공면조건 행렬식도 대중에게 공개된 수식이기 때문에 본 명세서 상에서 자세한 수식은 생략하도록 한다.

[0877] 이외에도 대상물의 특징점 좌표를 구하는 다양한 방법이 공개되어 있다.

[0878]

[0879] 1-3) 3번 이상의 촬영 동작이 이루어질 때 대상물의 특징점 좌표를 구하는 방법

- [0880] 본 시스템의 개념을 설명하기 위해 이제까지 2장의 영상을 이용해서 대상물의 특징점 좌표를 취득하는 방법에 대하여 설명하였다.
- [0881] 하지만 본 시스템의 영상촬영부분에서는 2번의 사진촬영뿐만이 아니라 정확도를 향상시키기 위해 동일 대상물에 대하여 3장 이상의 추가인 사진촬영을 통한 영상을 취득할 수 있다.
- [0882] 만약  $n$ (여기서  $n$ 은 3 이상)장의 사진(즉, 영상)을 취득하였다면,  $n$ 장의 사진 중 2장의 사진을 뽑아 대상물의 특징점 좌표를 구하는 반복계산을 할 수 있다.
- [0883] 즉,  $n$ 장의 사진이 촬영되었다면 동일 대상물에 대하여  $n \times (n-1)/2$  개의 좌표가 계산된다. 예를 들어 3장의 사진이 촬영되면 동일 대상물에 대해  $3 \times (3-1)/2 = 3$ 개의 좌표가 계산되고, 4장의 사진이 촬영되면 동일 대상물에 대해  $4 \times (4-1)/2 = 6$ 개의 좌표가 계산된다.
- [0884] 이 값을 단순한 산술평균으로 계산할 수 있지만, 본 시스템에서 정확도에 가장 큰 영향을 미치는 것은 두 촬영 위치의 거리 또는 측정시간이라 할 수 있다.
- [0885] 이동거리를 계산할 때 시간에 대한 함수로 이중 적분되기 때문에 측정시간이 길어질수록 오차가 커지고, 촬영위치 거리가 멀어질수록 오차가 커질 것이다.
- [0886] 따라서 경중률을 거리와 측정시간의 역수로 하여 거리가 멀고, 측정시간이 길수록 경중률을 낮게 잡고, 거리가 가깝고, 측정시간이 짧을수록 경중률을 높게 잡아 계산하는 것이 보다 정확한 계산이 될 것이다.
- [0887] 다음 [수식27]는 거리와 시간의 경중률에 따라서 가중평균을 계산하는 수식이다.
- [0888] [수식27]
- $$\hat{T} = \frac{\sum_{k=1}^n T_k \times W_k}{\sum_{k=1}^n W_k} \quad (\hat{T} : \text{최적값}, \quad T_k : \text{측정되거나 계산된 값}, \quad W_k : \text{경중률})$$
- [0889]
- $$W_k = \frac{1}{S_k \times t_k} \quad (S_k : \text{두 촬영위치의 거리}, \quad t_k : \text{이동시간})$$
- [0890]
- [0891] 예를 들어, 좌표를 구하고자 하는 목표 즉 대상물(T)가 A, B 및 C라는 3장의 사진으로 순서대로 촬영되었다면, A와B, B와C, A와C와 같이 이렇게 2장씩 짹으로 선택되어 계산되는 좌표가 다음과 같이 3개가 있다고 가정한다.
- [0892] 첫 번째로, A와B 영상을 이용해 구한 대상물 T의 좌표  $T_{AB}$  (1, 2, 3), A촬영위치와 B촬영위치의 거리  $S_{AB}=1m$ , A에서 B로 이동한 시간  $t_{AB}=1\text{초}$ 인 경우.
- [0893] 두 번째로, B와C 영상을 이용해 구한 대상물 T의 좌표  $T_{BC}$  (1.3, 2.3, 3.3), B촬영위치와 C촬영위치의 거리  $S_{BC}=0.7m$ , B에서 C로 이동한 시간  $t_{BC}=1\text{초}$ 인 경우.
- [0894] 그리고 세 번째로, A와 C영상을 이용해 구한 대상물 T의 좌표  $T_{AC}$  (1.5, 2.5, 3.5), 촬영위치A와 촬영위치 C의 거리  $S_{AC}=1.5m$ , A에서 C로 이동한 시간  $t_{AC}=t_{AB}+t_{BC}=2\text{초}$ 인 경우.
- [0895] 위와 같을 때 보정되는 좌표(X, Y, Z)는 다음과 같다.
- [0896]  $X = (1/(1 \times 1) + 1.3/(0.7 \times 1) + 1.5/(1.5 \times 2)) / (1/(1 \times 1) + 1/(0.7 \times 1) + 1/(1.5 \times 2)) \approx 1.22$
- [0897]  $Y = (2/(1 \times 1) + 2.3/(0.7 \times 1) + 2.5/(1.5 \times 2)) / (1/(1 \times 1) + 1/(0.7 \times 1) + 1/(1.5 \times 2)) \approx 2.22$
- [0898]  $Z = (3/(1 \times 1) + 3.3/(0.7 \times 1) + 3.5/(1.5 \times 2)) / (1/(1 \times 1) + 1/(0.7 \times 1) + 1/(1.5 \times 2)) \approx 3.22$
- [0899] 따라서, 최종 보정된 대상물(T)의 좌표는 (1.2, 2.2, 3.2)가 된다.

- [0900] 이와 같이, 촬영된 사진이 많을수록 가중평균하여 최적값을 찾을 수 있다.
- [0901] 가중평균을 위한 경중률은, 본 예의 경우, 측정시간과 두 촬영영상위치의 거리로 하였으나, 시스템 특성에 따라 다르게 될 수 있다.
- [0902] 또는, 최소제곱법을 이용하여 동시에 여러 장의 사진에서 좌표 관계식을 통해 최적값을 구하는 광속조정법(Bundle Adjustment)을 사용할 수 있다. 광속조정법은 본 시스템에 적용할 수 있는 계산방법 중의 한 예로서 광속조정법에 대한 일반식은 대중에게 공개된 공식 중에 하나이므로, 본 명세서상에서 광속조정법은 생략하도록 하겠다.
- [0903] 2) 촬영위치의 절대좌표 파악
- [0904] GNSS, 즉 위성을 이용한 위치정보 획득부(120)인 위치 결정 시스템은 상업적뿐만 아니라 군사적으로 굉장히 중요하게 사용되고 있다.
- [0905] 하지만, 위성신호를 수신할 수 없는 실내나 위성신호의 난반사가 심한 숲, 도심지, 그리고, 적대국의 위성신호 교란 시에는 위성을 이용한 위치결정시스템을 사용할 수 없게 된다.
- [0906] 본 시스템은, GNSS를 이용한 위치결정이 불가능한 상황에서 촬영위치의 절대좌표를 파악하여, 사용자에게 자신의 현재 위치를 결정해 줄 수 있다.
- [0907] (1) 대상물 한 점의 절대좌표를 알 때 촬영위치(즉, 위치 정보 측정 장치)의 절대좌표 계산(좌표관계식 이용)
- [0908] 도 39는 대상물 한 점의 절대좌표를 알 때 촬영위치의 절대좌표를 계산하는 개념도이다.
- [0909] 도 39의 (a)와 같이 촬영위치의 절대좌표를 모를 때 본 시스템은 자북방위각을 기준으로 촬영위치 및 대상물의 특징점 좌표는 상대좌표로 계산된다.
- [0910] 도 39의 (b)는 이해를 돋기 위해 상대좌표로 계산된 공간상의 3점을 2차원 XY평면상으로 투영한 모습이다.
- [0911] 도 40은 대상물의 한 점을 알 때 촬영위치의 절대좌표를 계산하는 연산부(210)의 동작 순서도이다.
- [0912] 평면직교좌표계로 표현되는 절대좌표는 일반적으로 도북방위각을 기준으로 표현된다.
- [0913] 따라서, 연산부(210)는 자북방위각으로 계산된 대상물의 상대좌표를 도북방위각으로 회전 변환하여야 한다.
- [0914] 이를 위해, 연산부(210)는 현재 알고 있는 대상물의 절대좌표를 이용하여 도자각(자북과 도북의 차이)을 계산한다(S601).
- [0915] 절대좌표를 이용한 도자각은 대중에게 공개된 자료로서 앞선 [수식10]의 자편각 공식과 TM투영식을 이용하여 시스템 상에서 직접 구할 수 있으며, 서버(2000) 등을 통해 외부로부터 제공받을 수 있다.
- [0916] 다음, 연산부(210)는 도자각을 이용하여 자북방위각 기반의 좌표계를 도북방위각 기반의 좌표계로 회전변환하여 준다[S602, 도 39(b)참조].
- [0917] 그런 다음, 연산부(210)는 좌표축의 회전변환 후 촬영위치를 (0,0,0)으로 가정하여 구한 대상물의 상대좌표와 절대좌표의 차이( $\Delta X$ ,  $\Delta Y$ ,  $\Delta Z$ )만큼 촬영위치의 상대좌표(X, Y, Z)를 평행 이동하여 촬영 위치의 절대좌표(X', Y', Z')를 구한다(S603, S604).
- [0918] 위 과정을 수식으로 나타내면 다음 [수식28]과 같다.
- [0919] [수식28]
- $$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} \cos\xi & -\sin\xi & 0 \\ \sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix}$$
- [0920]
- [0921] ( $X, Y, Z$ : 촬영위치의 상대좌표,  $\xi$ : 도자각,  $\Delta X, \Delta Y, \Delta Z$ : 평행이동량 즉, 대상물의 상대좌표와 절대좌표의 차이,  $X', Y', Z'$ : 촬영위치의 절대좌표)
- [0922] 이와 같이, 촬영위치의 절대좌표를 이용하여 촬영위치의 도자각을 구한 후, 연산부(210)는 현재 도자각, 예를 들어, 최초 계산에 사용된 가정된 도자각과 산출된 도자각을 비교하여 이를 두 도자각의 차가 오차 허용범위보다 크다면(S606), 촬영위치 도자각을 현재 도자각으로 다시 가정한 후(S607), 단계(S602)로 넘어가 현재 도자각

을 이용하여 좌표계를 회전한다.

[0923] 하지만 일반적으로 도자각의 변화량은 본 시스템이 주로 측정하게 될 1,000m이내 에서는 그 차이가 거의 없으므로, 일반적인 경우 촬영위치 도자각과 대상을 위치의 도자각은 차이가 없을 것이다.

[0924] 따라서, 비교된 두 도자각의 차가 오차 허용범위보다 크지 않아 도자각보정이 필요 없는 경우(S606), 단계(60 4)에서 취득한 절대좌표를 최종적으로 촬영위치의 절대좌표로 정한다(S608).

[0925] 예를 들어, 촬영위치의 절대좌표를 몰라 자북방위각 기반으로 계산된 촬영위치 A의 상대좌표가 (0,0,0)이고 B의 상대좌표가 (1,1,0)이며 대상물(T)의 상대좌표가 (25,1,50)이었다면, 이때 대상물(T)의 절대좌표가 (318482.630, 231145.260, 96.78)이었고 해당위치의 도자각이 8°라면, 도자각을 회전변환하여 변환된 상대좌표 A`, B`, T`는 다음과 같다.

[0926]  $A' = (0, 0, 0)$ ,  $B' = (0.8511, 1, 1294, 0)$ ,  $T' = (24.6175, 4.4696, 50)$ 이 된다.

[0927] T의 절대좌표가 (318482.630, 231145.260, 96.78)이므로 좌표변화량  $\Delta X = +318458.013$ ,  $\Delta Y = +231140.790$ ,  $\Delta Z = +46.78$ 이고, 촬영위치 A, B의 절대좌표( $A_{\text{절대}}$ ,  $B_{\text{절대}}$ )는 다음과 같다.

$$A_{\text{절대}} = (0+318458.013, 0+231140.790, 0+46.78) = (318458.013, 231140.790, 46.78)$$

$$B_{\text{절대}} = (0.851+318458.013, 1.129+231140.790, 0+46.78) = (318458.864, 231141.919, 46.78)$$

## (2) 대상물의 두 점 이상의 절대좌표를 알 때 촬영위치의 절대좌표 계산(좌표관계식 이용)

[0931] 본 시스템으로 촬영된 영상에서 2개 이상의 대상물의 절대좌표를 알면, 도편각을 고려하지 않고, 각 점의 위치 관계를 이용하여 촬영위치의 절대좌표를 계산할 수 있다.

[0932] 다음 [수식29]는 절대좌표를 계산하는 수식이다. [수식28]은 [수식29]와 같이 변형될 수 있다.

[0933] [수식29]

-  $X, Y, Z$ : 촬영위치의 상대좌표,  $\xi$ : 도자각,

-  $\Delta X, \Delta Y, \Delta Z$ : 평행이동량,  $X', Y', Z'$ : 촬영위치의 절대좌표,

[0936] -  $s$  : 축적계수)

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = s \times \begin{bmatrix} \cos \xi & -\sin \xi & 0 \\ \sin \xi & \cos \xi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} a & -b & 0 \\ b & a & 0 \\ 0 & 0 & s \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} c \\ d \\ e \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} aX - bY + 0Z + c \\ aY + bX + 0Z + d \\ 0X + 0Y + sZ + e \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} X & -Y & 0 & 1 & 0 & 0 \\ X & Y & 0 & 0 & 1 & 0 \\ 0 & 0 & Z & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ s \\ c \\ d \\ e \end{bmatrix}$$

[0940]

[0941] 2점 이상을 알면 다음과 같은 최소제곱법을 통해  $a, b, s, c, d, e$  값을 구할 수 있다.

$$\begin{bmatrix} X_1' \\ Y_1' \\ Z_1' \\ \vdots \\ X_n' \\ Y_n' \\ Z_n' \end{bmatrix} = \begin{bmatrix} X_1 - Y_1 & 0 & 1 & 0 & 0 \\ X_1 & Y_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & Z_1 & 0 & 0 & 1 \\ X_2 - Y_2 & 0 & 1 & 0 & 0 \\ X_2 & Y_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & Z_2 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n - Y_n & 0 & 1 & 0 & 0 \\ X_n & Y_n & 0 & 0 & 1 & 0 \\ 0 & 0 & Z_n & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ s \\ c \\ d \\ e \end{bmatrix}$$

[0942]

$$B = AT$$

$$T = (A^T A)^{-1} A^T B = \begin{bmatrix} a \\ b \\ s \\ c \\ d \\ e \end{bmatrix}$$

[0944]

[0945]  $a, b, s, c, d, e$  값이 정해지면 좌표관계식이 완성되어 촬영위치의 상대좌표를 넣으면 촬영위치의 절대좌표가 계산된다.

[0946]

위와 같은 [수식29]를 통해 연산부(210)는 촬영위치의 절대좌표를 구할 수 있으나, 이는 대상물의 절대좌표를 한 점 이상 알아야 구할 수 있는 방법이다. 보통의 경우 사용자가 어떤 대상물의 절대좌표를 알 수 있는 경우는 거의 없을 것이다.

[0947]

따라서, 특정 랜드마크의 절대좌표를 서버(2000) 상에서 보관하고 있다가, 사용자의 요청 동작 시 사용자가 촬영한 영상을 분석하여 촬영된 영상에 관련된 랜드마크, 예를 들어 촬영된 영상 주변에 위치하는 유명 건물의 절대좌표를 사용자 즉, 해당 위치 정보 측정장치(1000)에게 전송하여, 연산부(210)가 사용자의 촬영위치에 대한 절대좌표를 구할 수 있도록 한다.

[0948]

예를 들어, 서울의 경우 남산타워의 첨탑 꼭대기, 63빌딩 모서리 등과 같이 미리 정해진 건물의 특정 지점의 좌표를 서버(2000)에 랜드마크의 절대좌표로서 보관하고 있다가, 사용자가 남산타워나 63빌딩을 찍었다면, 형상분석이나 사용자의 선택을 통해 서버(2000) 상에 보관되어 있는 남산타워의 좌표를 사용자가 소유하고 있는 위치 정보 측정장치(예, 스마트폰)으로 전송하여 랜드마크의 절대 좌표를 이용하여 해당 촬영위치의 절대좌표를 구할 수 있게 된다.

[0949]

### (3) 영상촬영부분의 촬영위치 결정시스템을 이용하여 촬영위치의 절대좌표를 파악하는 방법

[0950]

본 시스템의 영상촬영부분(P1)에서 시스템에 설치된 가속도센서(101), 자이로센서(102), 자기장 센서(103)와 같은 감지부(100)의 감지값들을 처리하여 촬영위치를 계산할 수 있었다.

[0951]

이를 이용하여 본 시스템의 감지부(100)를 통한 위치 정보 측정 동작을 계속 활성화 시키면 공간상에서 사용자가 얼마나 움직였는지를 실시간으로 파악이 가능하다.

[0952]

이런 실시간 파악을 위해서는 도 6의 알고리즘 이용해서 공간상에서의 이동거리를 실시간으로 파악할 수 있다.

[0953]

예를 들어, GNSS와 같은 위치정보 획득부(120)를 사용할 수 없는 실내 측위 같은 경우, 절대좌표를 실외에서 얻어온 후에 위치 정보 측정장치(1000)를 사용자가 가지고 움직임으로서 사용자의 위치가 추적되고 기록되어, 위치 정보 측정장치(1000)의 실시간 절대좌표를 얻을 수 있게 된다.

[0954]

이를 이용하면, 넓은 지하 시설물, 예를 들어 서울 코엑스 지하 같은 곳에 사용자는 위치 정보 측정장치(1000)의 위치 즉, 자신의 위치를 계속 확인하면서 이동이 가능하다.

[0955]

감지부(100)에 의해 누적되는 위치 오차는 곳곳에 위치오차 보정을 위한 시설물(예, 랜드마크)를 촬영하여 위치

보정해 주거나, 위치 정보 측정장치(1000)가 스마트폰의 경우 가까운 통신 기지국의 신호를 받아 사용자가 별다른 조정이나 추가 입력 없이 위치 오차를 위치 정보 측정장치(1000) 상에서 자동으로 보정해서 사용자에게 자신의 절대위치를 제공해 줄 수 있다.

[0956] (4) 천문측량을 이용하여 촬영위치의 절대좌표를 취득하는 방법(태양, 달, 북극성, 별자리 등의 고도각과 방위각을 이용)

[0957] 이미 설명한 것처럼, 어떤 대상물의 절대좌표를 알거나 서버(2000)에 촬영된 영상 속 랜드마크의 절대좌표가 있다면, 촬영위치의 절대좌표 파악이 가능하다.

[0958] 그러나, 사용자 주변에 서버(2000)에 저장되어 있는 랜드마크가 없고, 사용자가 어떤 대상물의 특징점 좌표를 파악할 수 없는 경우, 예를 들어, 육지가 보이지 않는 바다 위에 있는 경우, 본 시스템은 태양, 달, 북극성 또는 별자리 등의 고저각과 방위각을 측정하여 사용자의 위치를 결정해 줄 수 있다.

[0959] 천문측량을 이용하여 촬영위치의 절대좌표 취득방법은 다양하나 본 명세서 상에서는 태양을 촬영하여 절대위치를 취득하는 법을 예로 들어 설명하도록 하겠다.

[0960] 본 위치 정보 측정장치(1000)로 영상을 촬영하는 경우 특정대상물의 방위각과 고저각을 계산할 수 있다. 이를 이용하여, 태양을 촬영하면 태양의 고저각과 방위각을 얻을 수 있어 촬영위치의 절대좌표를 구할 수 있게 된다.

[0961] 도 41은 태양을 촬영하여 태양의 고도각과 방위각을 계산하는 개념도이다.

[0962] 도 41의 (a)는 본 위치 정보 측정장치(1000)를 사용하여 태양을 촬영한 예시이다. 위치 정보 측정장치(1000)에 장착된 촬영부(110)인 디지털 카메라를 활용하여 태양을 촬영할 때, 태양의 높은 광량 때문에 이미지 센서인 CCD나 CMOS가 손상될 수 있다.

[0963] 따라서, 하드웨어적인 흑색필터의 사용이나, 소프트웨어적인 처리(셔터속도를 빠르게 하거나, 이미지 센서의 감도를 줄이는 방법 등)를 통해 이미지센서의 손상을 예방하여야 한다.

[0964] 도 41의 (b)는 촬영된 영상에서 태양의 사진좌표를 취득하여 태양의 방위각과 고도각을 구하는 개념도로서, 다음 [수식 30]를 통해 촬영된 영상과 위치 정보 측정장치(1000)의 자세 데이터를 이용해 촬영위치에서의 태양의 방위각과 고도각을 계산할 수 있다.

[0965] [수식30]

$$\Delta\kappa_s = \tan^{-1} \frac{x_s}{f_x}$$

[0966] ( $x_s$ : 태양의 중심점에 대한 사진좌표 중 x좌표,  $f_x$ : x축 초점거리)

$$\Delta\phi_s = \tan^{-1} \frac{y_s}{f_y}$$

[0967] ( $y_s$ : 태양의 중심점에 대한 사진 좌표 중 y좌표,  $f_y$ : y축 초점거리)

$$\kappa_{AS} = \kappa_A + \Delta\kappa_s$$

[0968] ( $\kappa_{AS}$ : 촬영위치 A에서의 태양의 방위각,  $\kappa_A$ : 촬영위치A의 방위각)

$$\phi_{AS} = \phi_A + \Delta\phi_s$$

[0969] ( $\phi_{AS}$ : 촬영위치 A에서의 태양의 고저각,  $\phi_A$ : 촬영위치A의 고저각)

[0970] 도 42는 촬영위치 A에서 태양의 고저각( $\phi_{AS}$ )과 태양의 방위각( $\kappa_{AS}$ )을 이용하여 태양의 고도방정식을 사용하여 위치를 결정하는 개념도이다.

[0971] 태양의 고도와 방위각을 알 때 촬영위치의 위도와 경도를 구하는 공식은 다음 [수식 31]과 같다.

[0972] [수식31]

- [0977] (1) 태양의 고도와 방위각 관계식

$$\sin\phi_{AS} = \sin\Psi_A \sin\delta + \cos\Psi_A \cos\delta \cos h \quad \text{---(1)}$$

[0978]  $\sin\kappa_{AS} = \frac{\cos\delta \sin h}{\cos\phi_{AS}} \quad \text{-----(2)}$

[0979]  $-\phi_{AS}$ : 태양의 고도각

[0980]  $-\kappa_{AS}$ : 태양의 방위각

[0981]  $-\Psi_A$ : 관측지점 A의 위도

[0982]  $-\delta$ : 태양의 적위(declination)

[0983]  $-h$ : 태양의 시간각

[0984] (2) 태양의 적위, 적경(right ascension), 시간각 계산식

[0985] - 수정율리우스적일

$$MJD = \lfloor 365.25 \times \text{year} \rfloor + \lfloor \text{year}/400 \rfloor - \lfloor \text{year}/100 \rfloor + \lfloor 30.59(\text{month}-2) \rfloor + \text{day} - 678912$$

[0986] year: 년, month: 월(단1월과 2월은 전년의 13월, 14월로 계산), day: 일,  $\lfloor \quad \rfloor$ : 가우스 정수(해당수보다 작은 정수 중 가장 큰 정수)

[0987] 예를 들어 2016년 1월 1일의 경우 year=2015, month=13, day=1

$$\begin{aligned} MJD &= \lfloor 365.25 \times 2015 \rfloor + \lfloor 2015/400 \rfloor - \lfloor 2015/100 \rfloor \\ &\quad + \lfloor 30.59(13-2) \rfloor + 1 - 678912 \\ &= 735978 + 5 - 20 + 336 + 1 - 678912 \\ &= 57388 \end{aligned}$$

[0988] - 윤리우스적일:  $JD = MJD + 2400000.5$

[0989] -  $D = JD - 2451545.0$  ( $D$ : 2000년 1월 1일 이후 윤리우스적일)

[0990] - 태양의 평균 변동  $g(^{\circ}) = 357.529 + 0.98560028 D$

[0991] - 태양의 평균 횡경  $q(^{\circ}) = 280.459 + 0.98564736 D$

[0992] - 태양의 곁보기 횡경  $L(^{\circ}) = q + 1.915 \sin g + 0.020 \sin 2g$

[0993] - 황도 경사각  $e(^{\circ}) = 23.439 - 0.00000036 D$

[0994] - 태양의 적위  $\delta(^{\circ}) = \sin^{-1}(\sin e \sin L)$

[0995] (단,  $\ell$ : 황도경사각,  $L$ : 태양의 곁보기 횡경)

[0996] - 태양의 적경  $\alpha(^{\circ}) = \tan^{-1}\left(\frac{\cos e \sin L}{\cos L}\right)$

[0997] - 그리니치 평균 항성시 GMST(Greenwich mean sidereal time)

[0998]  $GMST(\text{시간}) = 6^h.6070238 + 0^h.0657098245 \times d + 1^h.00273791 \times t$

[1003] (2016년 기준 한국천문연구원 제공 수식 d: 2016년 일수, t: 세계시)

$$\text{GMST} (\text{ }^\circ) = 100.46061837 + 36000.770053608 \times T + 0.000387933 \times T^2 - T^3 / 38710000 + (15 \times 1.00273790935) \times t$$

[1004] (단,  $T = (JD - 2451545.0) / 36525$ , t: 세계시)

[1005] - 태양의 시간각:  $h = \text{GMST} + \lambda_A - \alpha$

[1006] (GMST: 그리니치평균황성시,  $\lambda_A$ : 관측지점A의 경도,  $\alpha$ : 태양의 적경)

[1007] (3) 태양의 고도와 방위각을 이용하여 촬영위치의 위도와 경도 결정

[1008] - (1)의 ②식을 변형하여 경도( $\lambda_A$ ) 계산

$$\lambda_A = -\sin^{-1}\left(\frac{\sin \kappa_{AS} \times \cos \phi_{AS}}{\cos \delta}\right) - \text{GMST} + \alpha$$

[1009] (∵ 시간각 방향이 반대방향이므로)

[1010] - (1)의 ①식을 변형하여 위도( $\Psi_A$ ) 계산

$$a \sin \Psi_A + b \cos \Psi_A = c \quad (\text{단, } a = \sin \delta, b = \cos \delta \cos h, c = \sin \phi_{AS})$$

$$\sqrt{a^2 + b^2} \sin(\Psi_A + \theta) = c$$

$$\theta = \cos^{-1}\left(\frac{a}{\sqrt{a^2 + b^2}}\right) = \sin^{-1}\left(\frac{b}{\sqrt{a^2 + b^2}}\right)$$

(단,

$$\Psi_A = \sin^{-1}\left(\frac{c}{\sqrt{a^2 + b^2}}\right) - \theta$$

[1011] 예를 들어 2016년 12월 1일 대한민국 표준시 11시 30분 30초에 태양의 방위각과 고도를 본 위치 정보 측정장치 (1000)로 측정하였더니 태양의 진북 기준 방위각은  $\kappa_{AS} = 166^\circ 42' 55''$ , 태양의 고도각은  $\phi_{AS} = 30^\circ 38' 53''$  이었다.

[1012] 태양의 적경과 적위를 구하기 위해 2016년 12월 1일 대한민국 표준시 11시 30분 30초의 윤리우스 적일을 계산한다.

$$\begin{aligned} JD &= \lfloor 365.25 \times 2016 \rfloor + \lfloor 2016/400 \rfloor - \lfloor 2016/100 \rfloor \\ &\quad + \lfloor 30.59(12-2) \rfloor + 1 - 678912 + 2400000.5 \\ &= 2457723.5 \end{aligned}$$

[1013] 11시 30분 30초를 세계표준시로 바꾸면 대한민국표준시는 세계표준시에서 9시간 빠르므로 2시 30분 39초가 되고 이를 윤리우스적일에 반영한다.

$$- 2/24 + 30/1440 + 30/864000 = 0.10451389$$

$$- D = 2457723.5 + 0.10451389 - 2451545.0 = 6178.6045139$$

[1014] 이를 이용해 태양의 평균변동, 평균황경, 태양의 겉보기 황경, 황도 경사각을 다음과 같이 계산한다.

$$- g(\text{ }^\circ) = 357.529 + 0.98560028 D = 6447.163389^\circ = 327.163389^\circ$$

$$- q(\text{ }^\circ) = 280.459 + 0.98564736 D = 6370.3842276^\circ = 250.3842276^\circ$$

$$- L(\text{ }^\circ) = q + 1.915 \sin g + 0.020 \sin 2g = 6369.3276024^\circ = 249.3276^\circ$$

[1027]  $- e(^{\circ}) = 23.439 - 0.00000036 D = 23.4367757^{\circ}$

[1028] 태양의 적경( $\alpha$ )과 적위( $\delta$ )를 계산한다.

[1029]  $- \alpha(^{\circ}) = \tan^{-1}\left(\frac{\cos e \sin L}{\cos L}\right) = 247.6452774^{\circ} \quad (\because \delta < 0)$

[1030]  $- \delta(^{\circ}) = \sin^{-1}(\sin e \sin L) = -21.8469334^{\circ}$

[1031] 시간각을 계산하기 위해 다음과 같이 그리니치 표준 항성시(GMST)를 계산한다.

[1032]  $\text{GMST} (^{\circ}) = 100.46061837 + 36000.770053608 \times T + 0.000387933 \times T^2$   
 $- \quad - T^3 / 38710000 + (15 \times 1.00273790935) \times t$

[1033]  $(T = (JD - 2451545.0) / 36525 = 0.1691581, t = 2.5083333)$

[1034]  $- \text{GMST} (^{\circ}) = 6228.0108959^{\circ} = 108.0108959^{\circ}$

[1035] 다음, 다음과 같이 경도( $\lambda_A$ )를 계산한다.

[1036]  $- \lambda_A = -\sin^{-1}\left(\frac{\sin \kappa_{AS} \times \cos \phi_{AS}}{\cos \delta}\right) - \text{GMST} + \alpha$

[1037]  $- \lambda_A = -12.2975430 - 108.0108959 + 247.6452774 = 127.3368385^{\circ}$

[1038]  $- \lambda_A = (\text{E})127^{\circ}20'13''$

[1039] 다음의 식을 이용하여 위도( $\Psi_A$ )를 계산한다.

[1040]  $- \Psi_A = \sin^{-1}\left(\frac{c}{\sqrt{a^2 + b^2}}\right) - \theta$

[1041]  $(a = -0.372128273, b = 0.906883791, c = 0.509762317, \theta = 112.3101707^{\circ})$

[1042]  $- \Psi_A = 36.3558132^{\circ} = (\text{N})36^{\circ}21'21''$

[1043] 위와 같이 태양의 고도각과 방위각을 측정하여 촬영위치의 위도와 경도를 결정할 수 있다.

[1044] 태양의 고도각과 방위각이 초단위로 정밀하게 측정되었다면, 본 수식에 의한 위도와 경도의 오차는 분 단위 정도이다.

[1045] 이는 현재 스마트폰과 같은 모바일 기기에 설치되어 있는 GNSS의 평균 오차인 약 3~30m보다는 크다. 하지만, GNSS에 의한 위치결정이 불가능할 때, 단지 태양을 촬영하는 것만으로 위치를 결정해 줄 수 있는 것은 상당히 편리할 것이다.

[1046] 또한, 태양의 위치관계, 지구 형상, 지구 공전 관계 등을 보다 정밀하게 반영한 수식을 사용하면 오차를 줄여줄 수 있다.

[1047] [수식31]에서 계산되는 태양의 방위각은 진복을 기준으로 하는 방위각이다.

[1048] 일반적으로 촬영위치의 절대좌표를 알 경우 자편각을 이용하여 진복방위각을 쉽게 구할 수 있으나, 촬영위치의 절대좌표를 모를 때는 자편각을 알 수 없어 자복을 기준으로 태양의 방위각을 계산하여야 한다.

[1049] 자복을 기준으로 한 태양의 자복방위각과 고도각을 이용하여 위도와 경도를 구한 뒤 해당 위치의 자편각을 보정하여 태양의 방위각을 조정한 후 다시 위도와 경도를 계산한다.

[1050] 조정된 위도와 경도에서의 자편각을 새로 구해 태양의 방위각을 다시 조정하고 위도와 경도를 계산한다.

[1051] 이와 같은 과정을 계속 반복하여 보정된 위도와 경도의 차이가 오차범위 이내라면 자편각 보정을 멈추고, 마지막

막으로 조정된 위도와 경도가 최종 촬영위치의 위도와 경도가 된다.

- [1052] 예를 들어, 2016년12월1일 11시30분30초에 측정된 자북을 기준으로 한 태양의 자북 방위각이  $\kappa_{AS} = 158.69228^\circ$ , 태양의 고도각이  $\phi_{AS} = 30.64806^\circ$  일 때 [수식 31]의 과정을 통해 구한 위도와 경도가 각각  $\Psi_A = 34^\circ 28' 45''$ ,  $\lambda_A = 119^\circ 57' 7''$ 라면 해당 위치에서의 자편각은  $6^\circ 23.94' = 6.399^\circ$  이다.

- [1053] 이 자편각으로 방위각을 보정하면  $\kappa_{AS} = 158.69228^\circ + 6.399^\circ = 165.09128^\circ$  가 되고 보정된 방위각을 이용하여 위도와 경도를 구하면 각각  $\Psi_A = 36^\circ 3' 8''$ ,  $\lambda_A = 125^\circ 50' 18''$  가 된다. 이 위도와 경도의 위치에서의 자편각은  $7^\circ 44.22' = 7.737^\circ$  이다.

- [1054] 다시, 이 자편각으로 방위각을 보정하면  $\kappa_{AS} = 158.69228^\circ + 7.737^\circ = 166.42928^\circ$  이고, 다시 보정된 방위각을 이용해 구한 위도와 경도는  $\Psi_A = 36^\circ 18' 18''$ ,  $\lambda_A = 127^\circ 4' 22''$  가 되며, 이 위도와 경도의 위치에서의 자편각은  $7^\circ 58.80' = 7.980^\circ$  이 된다.

- [1055] 이 값을 이용해 보정한 방위각은  $\kappa_{AS} = 158.69228^\circ + 7.980^\circ = 166.67228^\circ$  이고, 이를 이용해 구한 위도와 경도는  $\Psi_A = 36^\circ 20' 54''$ ,  $\lambda_A = 127^\circ 17' 50''$  가 된다.

- [1056] 이 위치에서의 자편각은  $8^\circ 0.78' = 8.013^\circ$  이고, 이 값을 이용해 보정한 방위각은  $\kappa_{AS} = 158.69228^\circ + 8.013^\circ = 166.70528^\circ$  이 되며, 이를 이용해 구한 위도와 경도는  $\Psi_A = 36^\circ 21' 15''$ ,  $\lambda_A = 127^\circ 19' 39''$  가 된다.

- [1057] 이 때의 자편각은  $8^\circ 1.08' = 8.018^\circ$  이고, 이 자편각을 이용해 보정한 방위각은  $\kappa_{AS} = 158.69228^\circ + 8.018^\circ = 166.71028^\circ$  이고, 이를 이용해 구한 위도와 경도는  $\Psi_A = 36^\circ 21' 18''$ ,  $\lambda_A = 127^\circ 19' 56''$  가 된다.

- [1058] 마지막 위도와 경도 보정량이  $1'$  이내 이므로, 자편각 보정을 멈추고  $\Psi_A = 36^\circ 21' 18''$ ,  $\lambda_A = 127^\circ 19' 56''$  에서의 자편각  $8^\circ 1.14' = 8.019^\circ$  를 이용해 마지막으로 방위각을  $\kappa_{AS} = 158.69228^\circ + 8.019^\circ = 166.71128^\circ$  로 보정한 후 구한 최종 위도와 경도  $\Psi_A = 36^\circ 21' 18''$ ,  $\lambda_A = 127^\circ 19' 58''$  를 본 위치 정보 측정장치(1000)의 최종 절대좌표로 결정한다.

- [1059] 도 43은 태양을 촬영하여 촬영위치의 절대좌표를 계산하는 연산부(210)의 동작 순서도이다.

- [1060] 본 위치 정보 측정장치(1000)의 촬영부(110)를 이용하여 태양의 촬영이 이루어지면, 연산부(210)는 촬영할 때의 위치정보 획득부(120)를 이용하여 위치 정보 측정장치(1000)의 자세 데이터를 측정하여 메모리(220) 혹은 저장부(400)에 기록한다(S701).

- [1061] 다음, 연산부(210)는 촬영된 영상에서 태양의 중심점 영상좌표를 취득한다(S702).

- [1062] 태양의 중심점 영상좌표 취득 알고리즘은 이전까지의 영상처리 알고리즘에 비교하여 상당히 간단하다. 즉, 촬영된 영상에서 가장 크고, 가장 밝은 지점의 가운데 부분을 판정하여 태양의 중심점으로 선택하면 된다(도 41의 (a) 참조)

- [1063] 다음, 연산부(210)는 태양의 중심점 영상좌표를 카메라 내부표정 과정, 즉 [수식20]를 통해 렌즈왜곡에 따른 영상좌표를 보정하고, [수식21]을 사용하여 태양의 중심점에 대한 사진좌표를 취득한다(S703).

- [1064] 그런 다음, 연산부(210)는 취득한 영상좌표를 [수식22]를 사용하여 기기회전각에 따른 영상좌표보정을 한다(S704).

- [1065] 취득한 영상좌표와 기기의 자세 데이터를 이용하여, 연산부(210)는 다음 [수식 30]을 통해 태양의 고저각과 방위각을 구한다(S705).

- [1066] 이때, 촬영위치의 절대좌표를 모르기 때문에, 연산부(210)는 자북방위각 기준으로 방위각을 취득하게 된다.
- [1067] 다음, 연산부(210)는 [수식31]를 사용하여 태양의 고도와 방위각 촬영일시와 시간을 이용해서 촬영위치의 위도와 경도를 계산한다(S706).
- [1068] 이때, 위도와 경도는 자북방위각으로 계산하였기 때문에, 연산부(210)는 계산된 위도와 경도에서의 자편각양만큼 보정해준 후 위도와 경도를 다시 계산하여, 위도와 경도의 보정량이 허용오차 이내라면 자편각 보정을 면친다(S707).
- [1069] 자편각 보정이 종료되면, 연산부(210)는 최종으로 결정된 위도와 경도를 촬영위치 A에서의 절대좌표으로 정한다(S708).
- [1070] 이처럼, 태양을 이용하여 촬영위치의 절대좌표를 구하는 방법은 태양을 관측할 수 없는 날씨하거나 밤에는 사용이 불가능 하다.
- [1071] 촬영 시점이 밤인 경우 북극성의 고도를 측정하여 촬영장소의 위도를 결정해 줄 수 있거나, 별자리 혹은 달의 고도와 방위각을 이용하여 위치를 결정해 줄 수 있다. 북극성의 고도, 별자리 또는 달을 이용한 위치결정 방법은 이미 설명한 태양을 이용해 위치를 결정해 주는 방법과 개념적으로 비슷하므로, 본 명세서 상에서 세부 계산 방법은 생략하도록 하겠다.
- 3) 좌표관계식을 이용하여 좌표들 간의 관계를 구하는 방법**
- [1073] 대상물의 어느 두 점간의 좌표를 알 때 그 두 점간의 위치 관계는 다음 [수식32]와 같다.
- [1074] [수식32]
- [1075] 대상물의 A점에 대한 좌표인 A좌표가  $(X_A, Y_A, Z_A)$ 이고, 대상물의 B점에 대한 좌표인 B좌표가  $(X_B, Y_B, Z_B)$ 일 때,
- [1076] (1) A점과 B점 사이의 거리( $L$ )는
- [1077] 
$$L = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2 + (Z_A - Z_B)^2}$$
 이고,
- [1078] (2) A점과 B점 사이의 수직거리(높이 차)( $H$ )는
- [1079] 
$$H = Z_A - Z_B$$
 이며,
- [1080] (3) A점과 B점 사이의 수평거리( $D$ )는
- [1081] 
$$D = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2}$$
 이며,
- [1082] (4) A점과 B점의 기울기( $\theta$ )는
- [1083] 
$$\theta = \tan^{-1} \left( \frac{H}{D} \right)$$
 이다.
- [1084] 또한, 대상물의 세 점 이상의 좌표를 알면 그 세 점이 있는 공간상의 평면방정식을 세워 줄 수 있으며, 이를 이용해 사면의 주향(즉, 경사방향), 경사각 및 면적 등 적어도 하나를 계산할 수 있다.
- [1085] 따라서, 어떠한 사면의 경사는 사면의 안정도를 파악할 수 있어, 산사태나 눈사태 등의 자연재해 예방 및 감독이 가능하다.
- [1086] 다음 [수식33]는 3점 이상의 좌표를 알 때 해당 사면이 이루는 경사각을 계산하는 수식이다.
- [1087] [수식33]
- [1088] (1) 평면 방정식이  $\alpha X + \beta Y + \gamma Z = 1$  일때, 벡터  $\vec{v} = (\alpha, \beta, \gamma)$

$$\begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n \end{bmatrix} \times \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

[1089] [1090] (2) 평면방정식을 구하기 위해 다음과 같은 3점 이상의 좌표  $(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), \dots, (X_n, Y_n, Z_n)$ 를 알면 다음과 같은 최소제곱법을 이용하여 벡터  $\vec{v} = (\alpha, \beta, \gamma)$ 를 구할 수 있음.

$$\begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n \end{bmatrix}, \quad V = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

[1091] [1092]  $-AV = B$  (행렬연산)

$$\begin{bmatrix} V \\ - \end{bmatrix} = (A^T A)^{-1} A^T B = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

[1093] [1094] (3) 벡터  $\vec{v} = (\alpha, \beta, \gamma)$ 가 결정되면, 지면과 수직인  $\vec{p} = (0, 0, 1)$ 과 이루는 벡터 각이 사면을 이루는 경사각, 사면각( $\theta$ )이 된다.

$$\theta = \cos^{-1}\left(\frac{|\gamma|}{\sqrt{\alpha^2 + \beta^2 + \gamma^2}}\right)$$

[1095] [1096] 다음 [수식34]은 공간상의 3점의 길이를 알 때 그 평면이 이루는 면적을 계산하는 방법이다. 면적을 계산하는 방법은 다양하게 있으나 본 명세서상에서는 변의 길이를 이용해서 면적을 계산하는 헤론의 공식을 한 예로 설명하겠다. 공간상의 좌표가 결정되면 [수식32]의 (1)를 이용하여 공간상의 길이가 계산되기 때문에 공간상의 면적을 계산할 수 있다.

[1097] [수식34]

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

$$[1098] [1099] (단, 삼각형의 세변의 길이가  $a, b, c$  일 때,  $s = \frac{1}{2}(a+b+c)$ )$$

[1100] 다음 도 44와 도 45는 영상처리부분의 사용자 입력화면에서 사용자가 각 대상물을 선택함으로서 촬영위치와 대상물(즉, 해당 대상물의 특징점)의 위치관계나 특징점들 간의 위치관계를 파악하는 개념도이다.

[1101] 도 44와 도 45의 경우, 결정된 특징점 즉 해당 대상물의 특징점이 5개가 존재하지만, 이 특징점의 개수는 사용자의 선택이나 영상의 종류에 따라 변경된다.

[1102] 도 44의 (a)는 해당 대상물의 각 특징점과 촬영위치에서부터의 거리(L)를 표시해준 화면이다.

[1103] 도 44의 (b)는 사용자가 설정을 조정하여 해당 대상물의 각 특징점과 촬영위치와의 각종 위치관계인 거리(L), 수직거리(H), 수평거리(D) 및 경사각( $\theta$ )를 표시해준 화면이다.

[1104] 사용자는 본 위치 정보 측정장치(1000)의 설정을 조정하여 다양한 결과를 확인할 수 있으며, 설정의 조정에 따라 특징점의 절대좌표를 평면직교 좌표계나 경위도 좌표계로도 표시해 줄 수 있다.

[1105] 또한, 도 44의 (c)와 같이 사용자 입력부(130)를 이용하여 해당대상물의 특징점들 간을 연결 동작이 이루어져, 연결 동작에 대한 입력이 연산부(210)로 입력될 수 있다.

[1106] 이런 경우, 연결 동작이 이루어진 두 특징점 사이의 거리에 대한 계산이 이루어져, 도 45의 (a)와 같이 두 특징점이 선으로 연결되고 두 특징점 간의 거리이 화면에 표시된다.

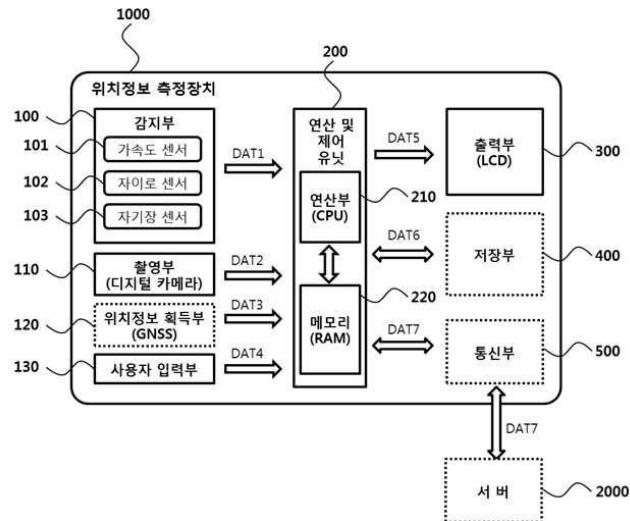
- [1107] 도 45의 (b)는 사용자가 설정을 조정하여 특징점을 간의 각종 위치관계를 표시해준 화면이고, 특징점의 개수를 3점을 선택했을 경우는 3점이 이루는 평면의 면적과 그 평면이 이루는 경사각등이 표시될 수 있다.
- [1108] 이와 같이 본 시스템을 활용하면, 단지 2장의 사진촬영만으로 해당 특징점들의 위치관계와 면적, 경사각등을 확인할 수 있다.
- [1109] 또한, 해당 위치 정보 측정장치(1000)의 절대위치를 모를 때에도 원하는 지점들 간의 좌표는 상대좌표로 계산되어 각종 위치관계는 그 정확도의 차이 없이 구할 수 있다.
- [1110] 본 발명을 이용하면 사용자는 자신이 어떤 부분이나 어떤 지점의 면적을 알고 싶을 때 단지 2장의 사진 촬영만으로 해당 부분의 면적이나 경사도 및 위치관계를 파악할 수 있다.
- [1111] 예를 들어, 이사를 가기 위해 새집의 방의 크기 등을 쟁 필요 없이 단지 2장의 사진촬영만으로 각 방의 크기와 해당 방향으로의 길이, 각종 콘센트나 스위치의 높이 등이 한 번에 측정되고, 사용자는 본 자료를 저장하여, 언제든 다시 불러와서 각종 위치관계들을 다시 확인할 수 있다.
- [1112] 본 시스템은 일상생활의 작은 편의를 제공해주는 것 외에도, 각종 검측이나, 재해재난의 사전 파악, 군사 목표 파악 등에 적용될 수 있다.
- [1113] 예를 들어, 대상물의 3차원 좌표를 획득함으로서 어떠한 건물이나 물체의 컴퓨터 모델링을 위한 자료로서 획득이 가능하고 영상데이터도 같이 얻음으로서 텍스쳐 매핑 자료로도 활용이 가능하다.
- [1114] 이러한 본 발명으로 인해 용이하고 정확하게 3차원 좌표와 영상자료를 얻게 된다.
- [1115] 이상에서 본 발명의 바람직한 실시예에 대하여 상세하게 설명하였지만 본 발명의 권리범위는 이에 한정되는 것은 아니고 다음의 청구범위에서 정의하고 있는 본 발명의 기본 개념을 이용한 당업자의 여러 변형 및 개량 형태 또한 본 발명의 권리범위에 속하는 것이다.

### **부호의 설명**

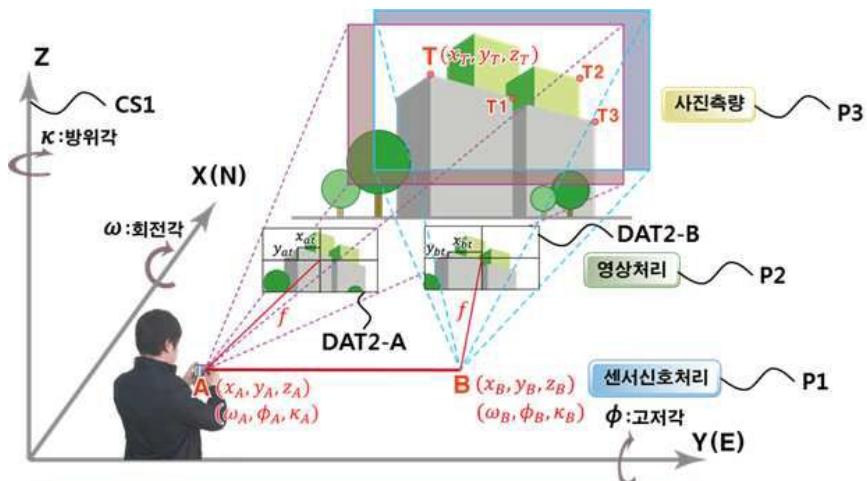
- |                  |                 |
|------------------|-----------------|
| 1000: 위치 정보 측정장치 | 100: 감지부        |
| 101: 가속도 센서      | 102: 자이로 센서     |
| 103: 자기장 센서      | 110: 촬영부        |
| 120: 위치정보 획득부    | 200: 연산 및 제어 유닛 |
| 210: 연산부         | 220: 메모리        |
| 300: 출력부         | 400: 저장부        |
| 500: 통신부         | 2000: 서버        |

## 도면

## 도면1



## 도면2

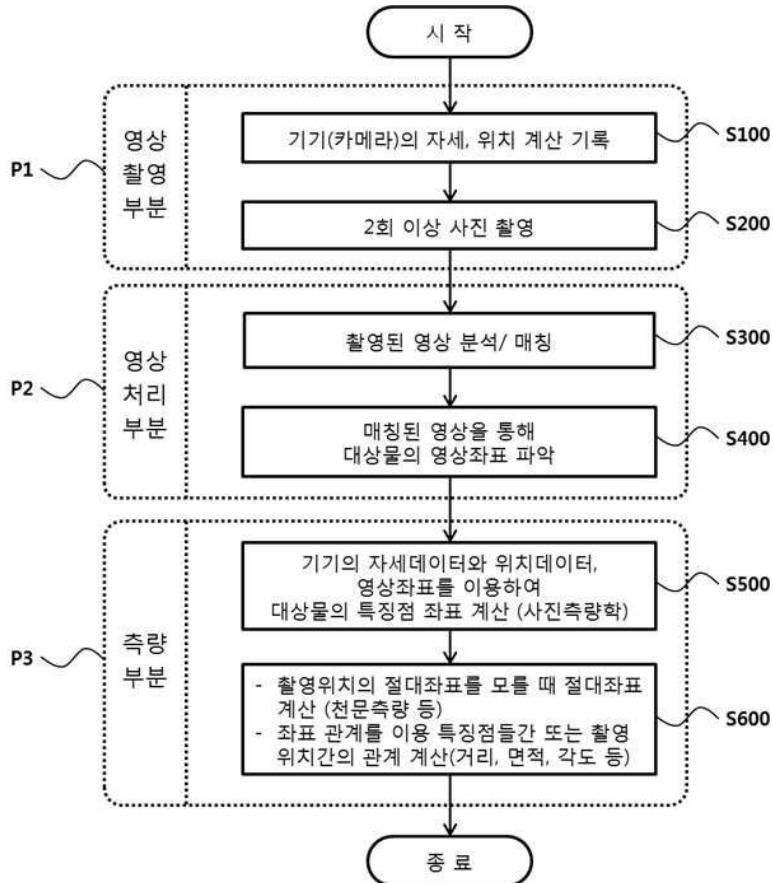


센서신호처리 : 내장플러센서값을 처리하여 A, B위치의 상대좌표 (x,y,z)와 자세( $\omega, \phi, \kappa$ )를 구함

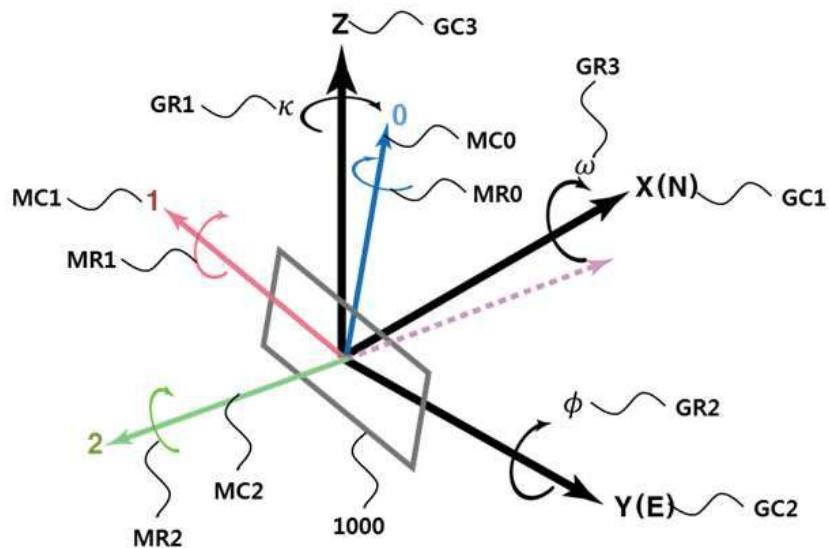
영상 처리 : 촬영된 영상을 이미지 처리기법으로 영상좌표( $x_t, y_t$ )를 취득

사진 축량 : 카메라의 상대좌표와 자세, 영상좌표의 처리를 통해 대상물의 좌표( $x_T, y_T, z_T$ ) 계산

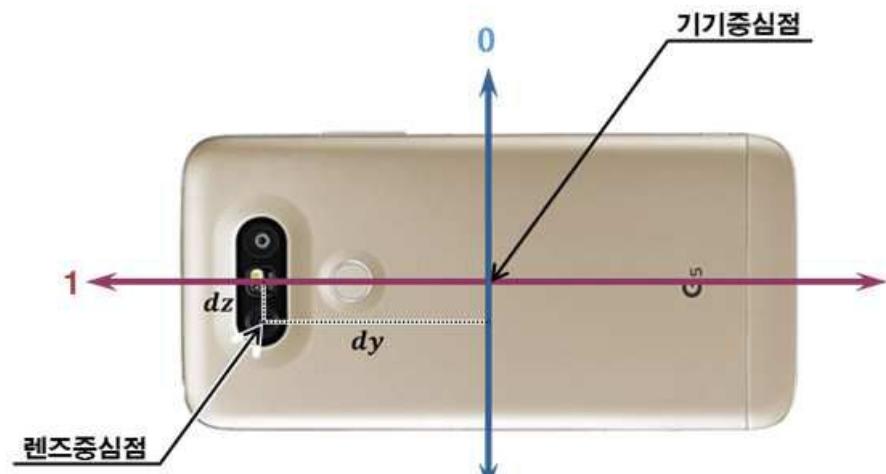
## 도면3



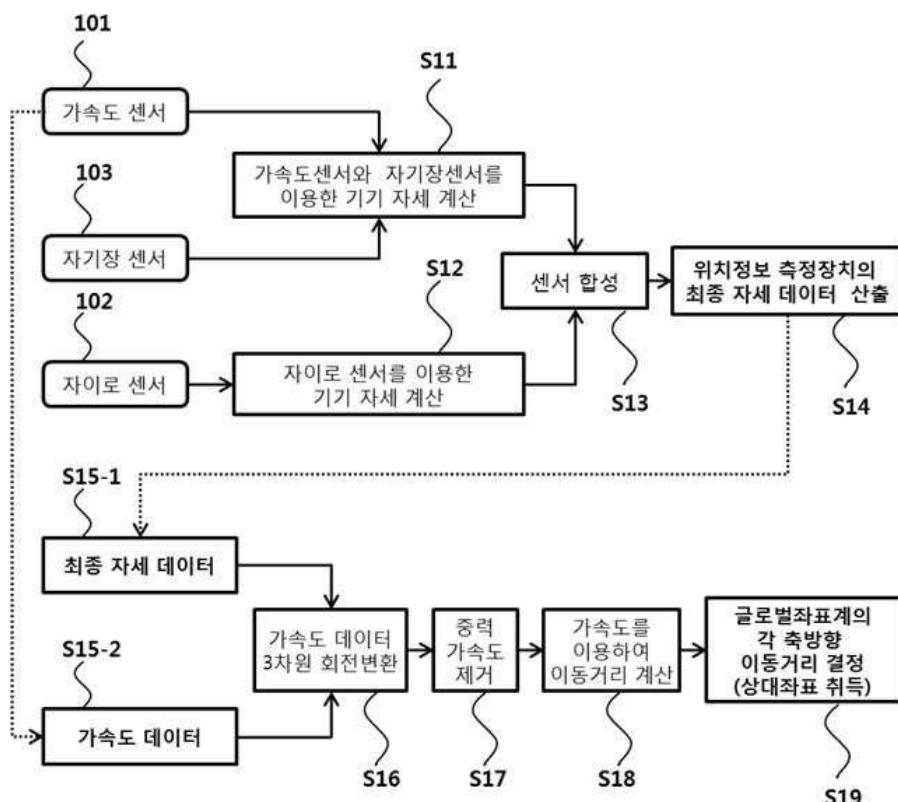
## 도면4a



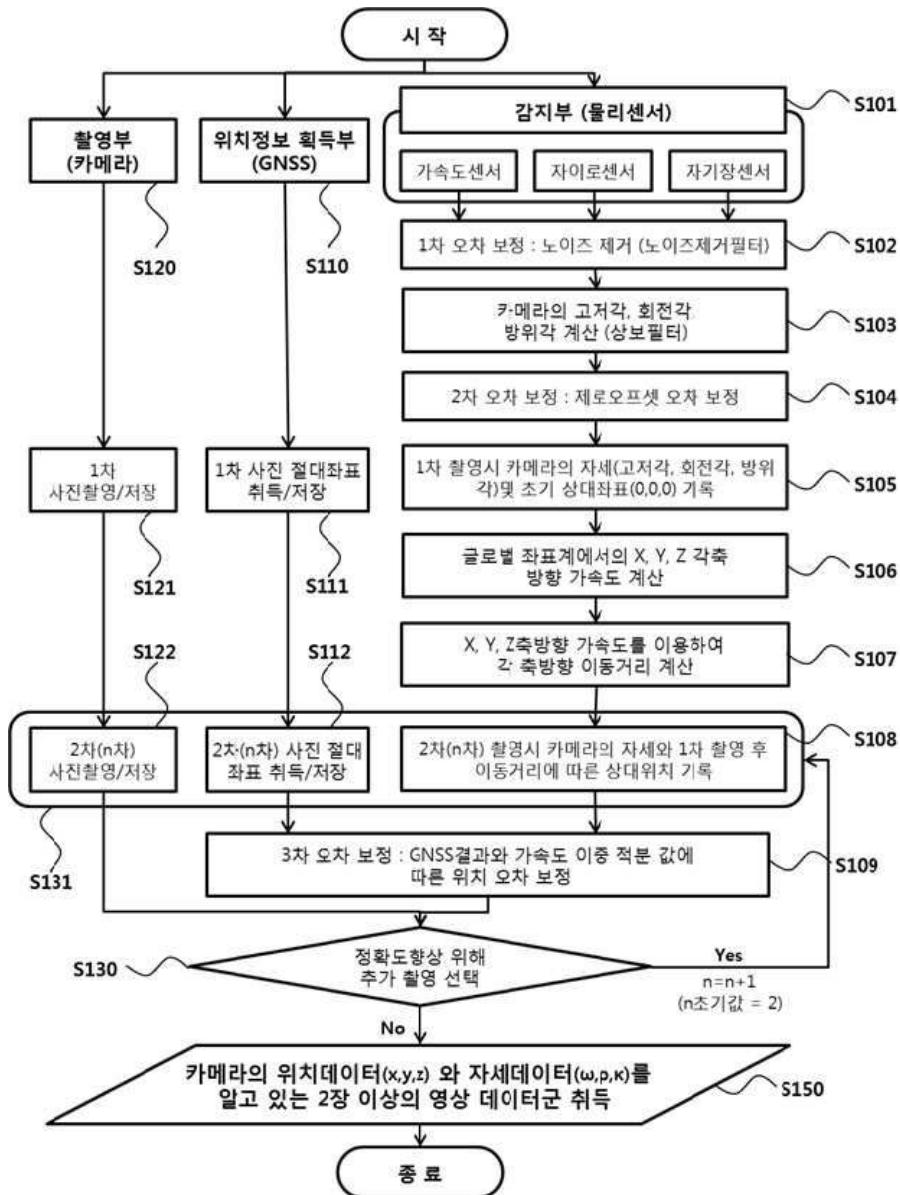
도면4b



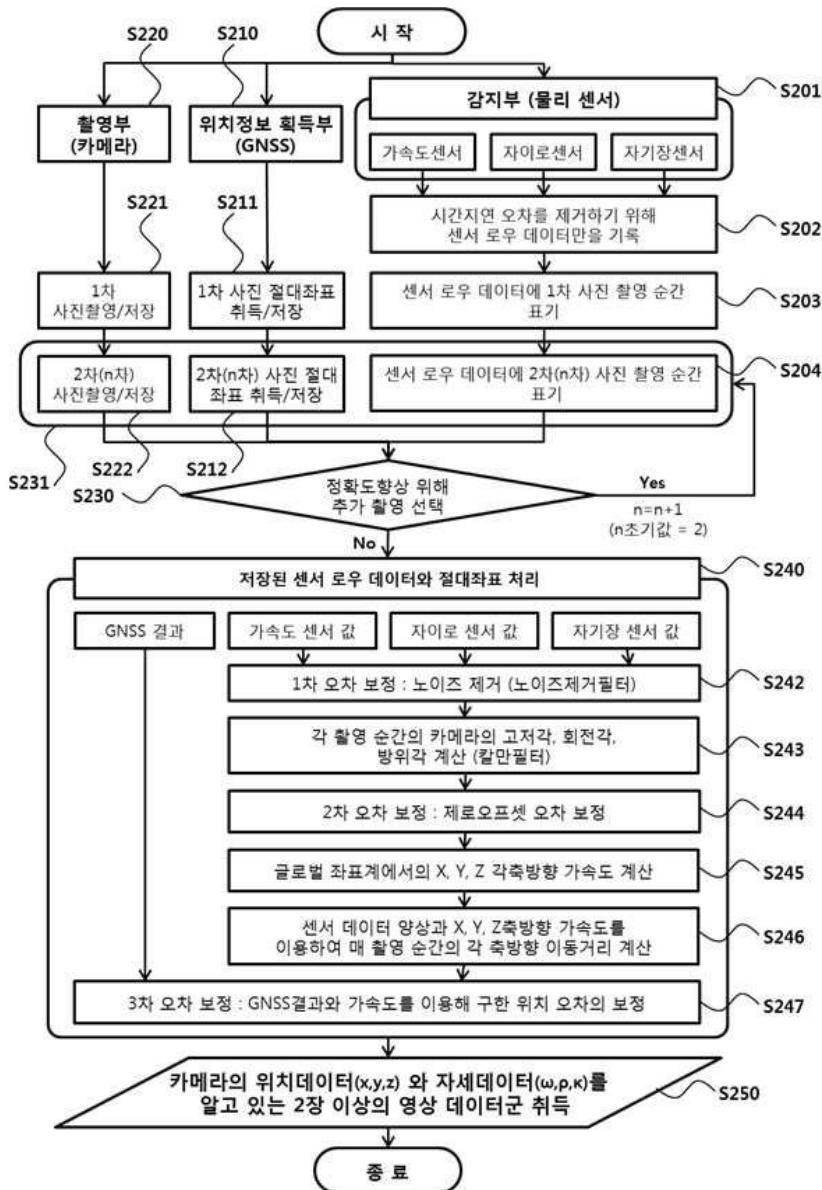
도면5



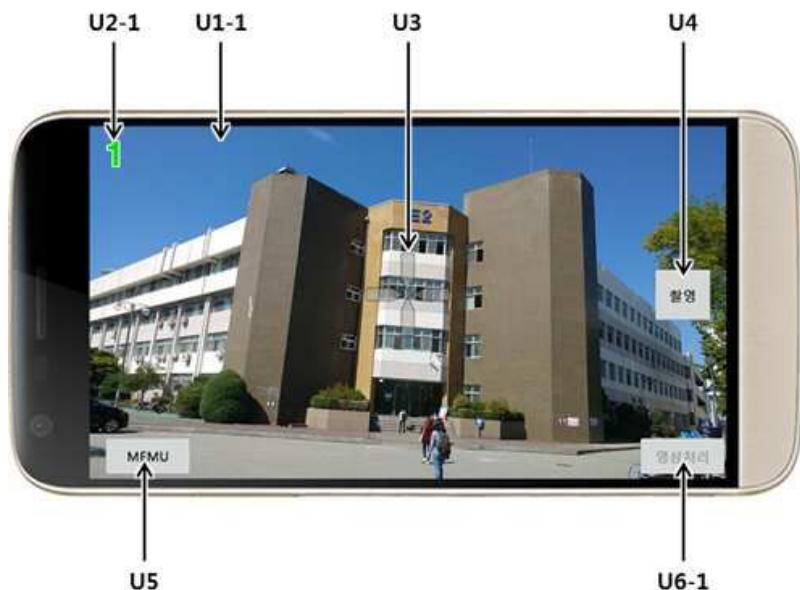
## 도면6



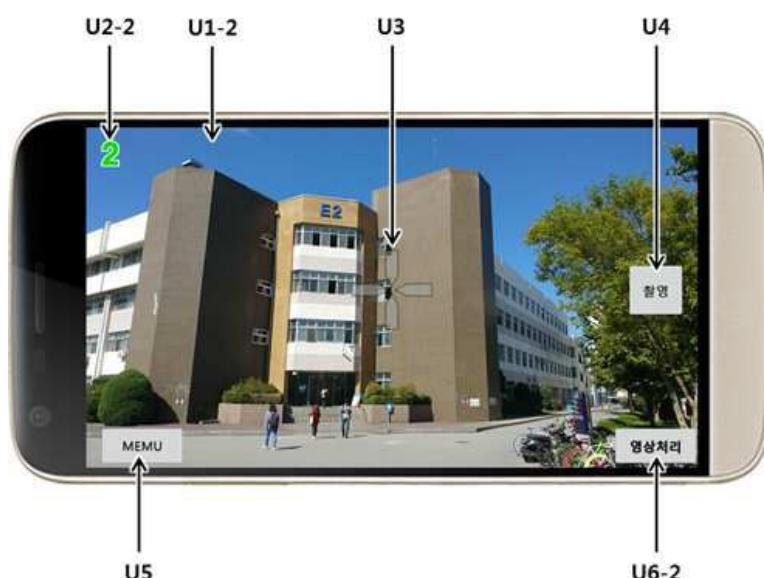
## 도면7



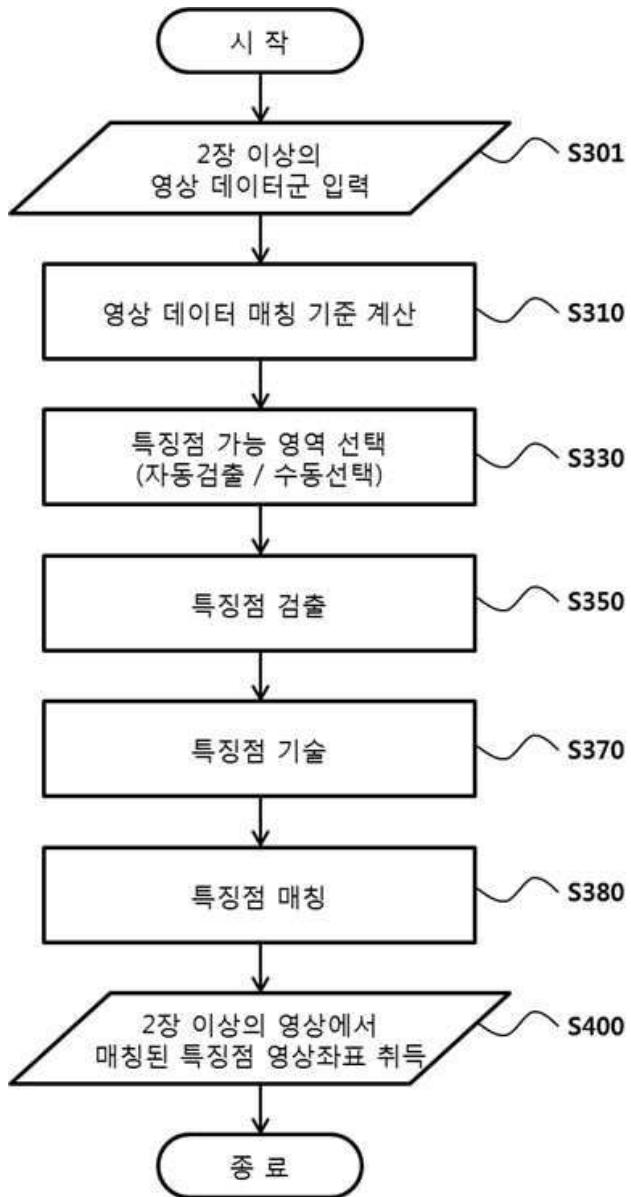
도면8a



도면8b



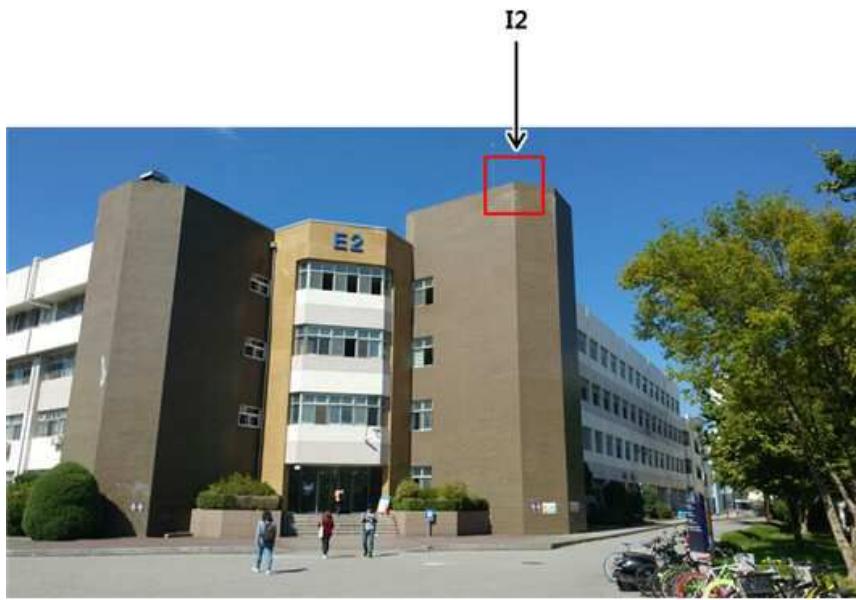
도면9



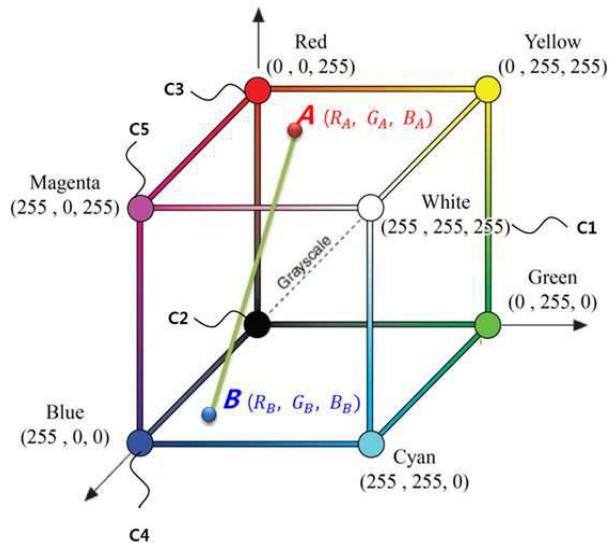
도면10a



도면10b



## 도면11



## 도면12

블록 합

	0	1	2	3	4	5	6	7	8	9
0	6	5	4	6	8	9	10	12	15	17
1	6	7	10	12	19	20	23	25	28	30
2	15	20	40	55	60	90	91	94	100	120
3	50	30	40	35	60	90	70	10	13	15
4	125	124	122	120	119	125	130	135	137	135
5	125	124	120	119	118	124	112	134	140	140
6	101	103	99	95	80	60	70	80	100	110
7	90	80	70	60	50	20	10	12	15	30
8	110	124	80	56	27	29	40	60	32	33
9	35	36	57	48	59	66	75	83	12	25

적분 영상(Integral Image : ii)

	0	1	2	3	4	5	6	7	8	9
0	6	11	15	21	29	38	48	60	75	92
1	12	24	38	56	83	112	145	182	225	272
2	27	59	113	186	273	392	516	647	790	957
3	77	139	233	341	488	697	891	1032	1188	1370
4	202	388	604	832	1098	1432	1756	2032	2325	2642
5	327	637	973	1320	1704	2162	2598	3008	3441	3898
6	428	841	1276	1718	2182	2700	3206	3696	4229	4796
7	518	1011	1516	2018	2532	3070	3586	4088	4636	5233
8	628	1245	1830	2388	2929	3496	4052	4614	5194	5824
9	663	1316	1958	2564	3164	3797	4428	5073	5745	6400

원 영상(Source image : S)

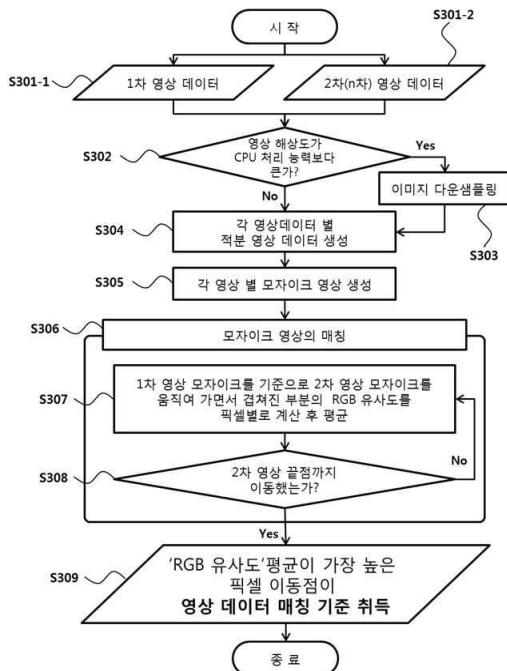
ii1

ii2

ii3

ii4

## 도면13



## 도면14

입력 : RGB 값을 가지고 있는 모자이크화된 A영상, B영상  
출력 : 영상 데이터 매칭 X, Y 기준

```

1: Static int 모자이크영상매칭기준(A영상, B영상){
2:   m_w = 모자이크 영상의 폭;
3:   m_h = 모자이크 영상의 높이;
4:   for (int x_shift=-m_w/2; x_shift<=m_w/2; x_shift++) {
5:     for (int y_shift=-m_h/2; y_shift<=m_h/2; y_shift++) {
6:       M_N=0;
7:       RGB유사도_합계=0;
8:       for (int Ys = 0; Ys < m_h; Ys++) {
9:         for (int Xs = 0; Xs < m_w; Xs++) { //Xs, Ys : A영상 픽셀점
10:           Xt = Xs + x_shift; //Xt, Yt : B영상 픽셀점
11:           Yt = Ys + y_shift;
12:           if (Xt >= 0 && Xt < m_w && Yt < m_h) {
13:             RGB유사도 = A영상(Xs, Ys)와 B영상(Xt, Yt)의 RGB유사도 계산
14:             M_N=M_N+1;
15:             RGB유사도_합계= RGB유사도_합계 + RGB유사도;
16:           }
17:         }
18:       }
19:       RGB유사도_평균 = RGB유사도_합계 / M_N;
20:       if (RGB유사도_평균 > 이전까지_최대값){
21:         이전까지_최대값=RGB유사도_평균;
22:         영상 데이터 매칭 X 기준 = x_shift;
23:         영상 데이터 매칭 Y 기준 = y_shift
24:       }
25:     }
26:   }
27:   return 영상 데이터 매칭 X,Y기준;
28: }
  
```

도면15



## 도면16



<b>S307-1</b>														
165.167.146	149.152.153	101.98.80	125.117.102	143.131.119	142.130.218	147.105.88	88.85.78	65.93.39	83.226.125	92.139.185	88.136.184	77.97.82		
147.175.176	151.156.155	92.78.61	137.124.106	141.128.110	139.127.109	114.103.85	87.82.69	54.63.71	78.90.105	74.95.112	101.139.176	107.134.154		
148.175.176	128.132.128	76.71.60	130.124.108	137.124.105	136.123.103	118.101.82	84.76.63	87.70.79	59.71.88	59.71.88	86.107.126	100.132.143		
166.164.160	146.141.137	87.74.57	138.123.93	134.123.98	134.123.99	113.102.81	85.78.62	45.57.64	62.78.90	62.74.82	67.78.84	88.93.94		
129.80.26	49.50.41	76.78.48	106.104.71	126.116.85	132.122.88	112.102.80	87.80.62	33.59.42	87.66.75	73.80.86	76.80.84	94.101.108		
91.88.81	97.91.93	122.118.91	86.99.81	47.46.36	117.106.84	116.103.88	104.89.86	28.28.27	8.13.12	81.40.28	66.58.57	77.84.93		
154.149.144	89.87.100	154.146.139	150.149.135	125.120.114	137.123.119	134.126.111	116.110.96	91.93.85	90.91.87	99.103.95	115.126.119	99.103.114		

<b>S307-2</b>														
90.111.193	69.111.191	70.111.191	70.111.191	68.111.190	66.111.190	68.124.192	63.124.193	63.122.189	60.121.187	59.122.188	58.121.186	50.111.185		
73.127.196	73.127.196	72.127.196	83.129.184	86.116.150	73.124.181	65.127.194	65.127.193	64.127.192	62.126.192	61.126.191	60.126.191	60.125.190		
76.131.203	75.131.203	54.133.192	148.140.190	137.191.121	88.94	41.64.89	61.116.173	65.131.197	68.133.196	63.131.196	62.130.196	88.125.184		
61.137.209	83.119.206	126.155.192	146.137.127	139.127.117	89.70.68	31.34.96	41.40.99	142.140.126	199.171.117	131.131.127	145.139.122	120.123.122		
164.182.220	236.241.245	210.205.204	243.154.123	184.127.116	98.60.58	34.35.35	50.44.39	159.138.101	112.114.106	109.110.100	108.105.92	96.83.67		
179.179.172	135.137.132	120.123.115	141.132.125	135.127.117	50.91.90	37.37.36	50.44.39	162.141.103	217.138.214	209.202.198	168.171.149	96.88.73		
166.170.171	224.226.226	175.166.159	137.129.118	152.124.114	39.40.40	39.39.36	52.47.41	158.197.97	130.140.142	137.147.150	134.141.143	99.87.69		

<b>S307-3</b>														
76	75	73	76	74	78	74	72	79	81	92	92	76		
76	79	67	78	84	78	72	70	69	79	81	90	84		
77	82	67	93	96	84	82	78	70	71	71	82	90		
77	79	63	92	96	80	74	86	68	62	77	76	88		
34	24	46	84	92	77	75	87	61	82	89	90	91		
64	81	94	85	67	79	74	79	56	19	35	54	94		
92	48	92	94	98	65	66	76	81	81	82	83	89		

도면17



(1) (-13,-7)



(2) (-13,0)



(3) (-13,+7)



(4) (0,-7)



(5) (0, 0)



(6) (0,+7)



(7) (+13,-7)

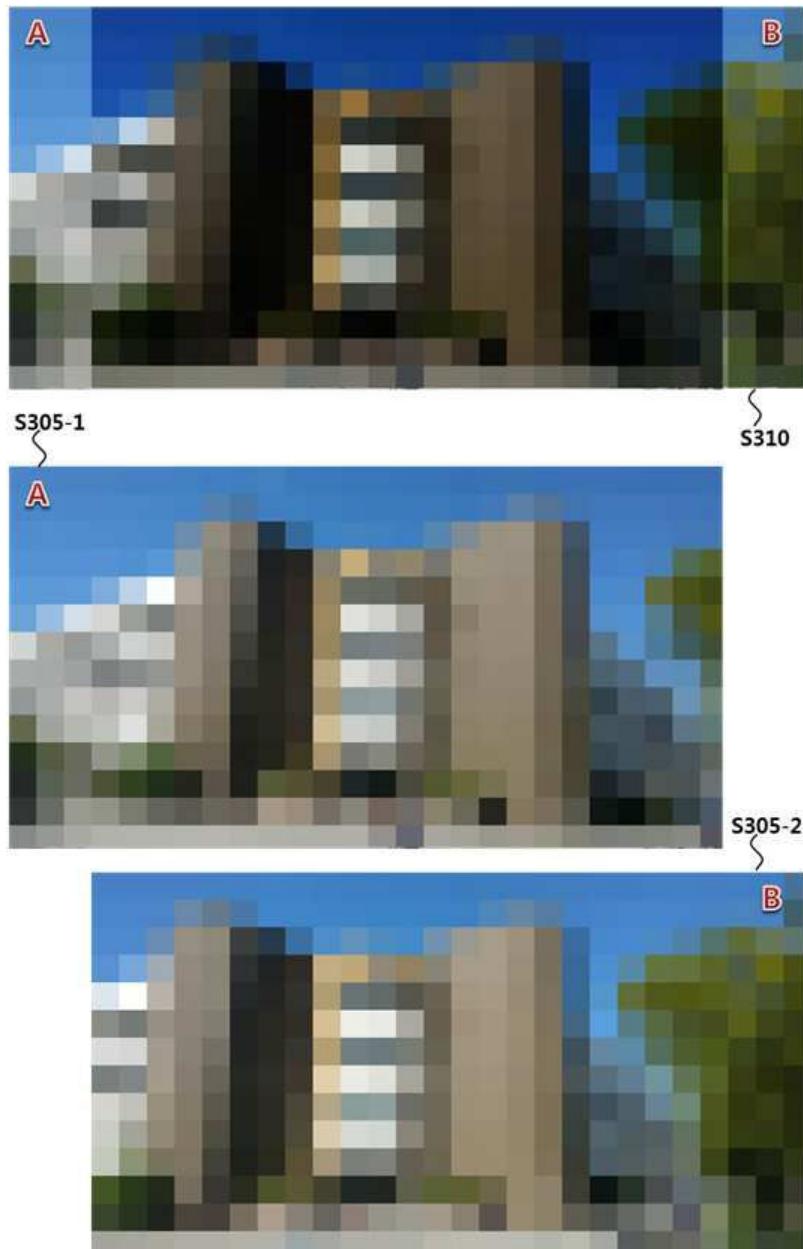


(8) (+13, 0)

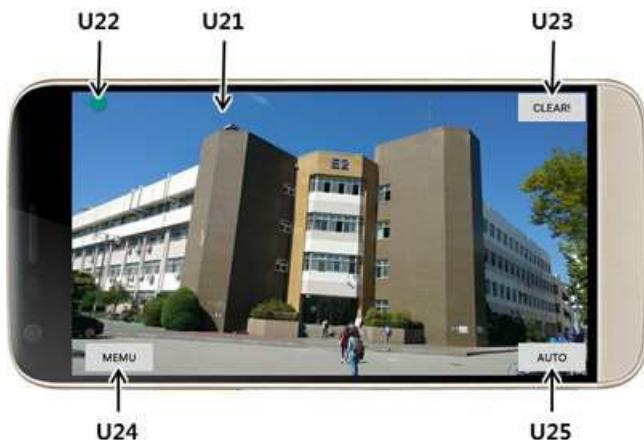


(9) (+13,+7)

도면18



도면19a



도면19b



도면19c



도면20

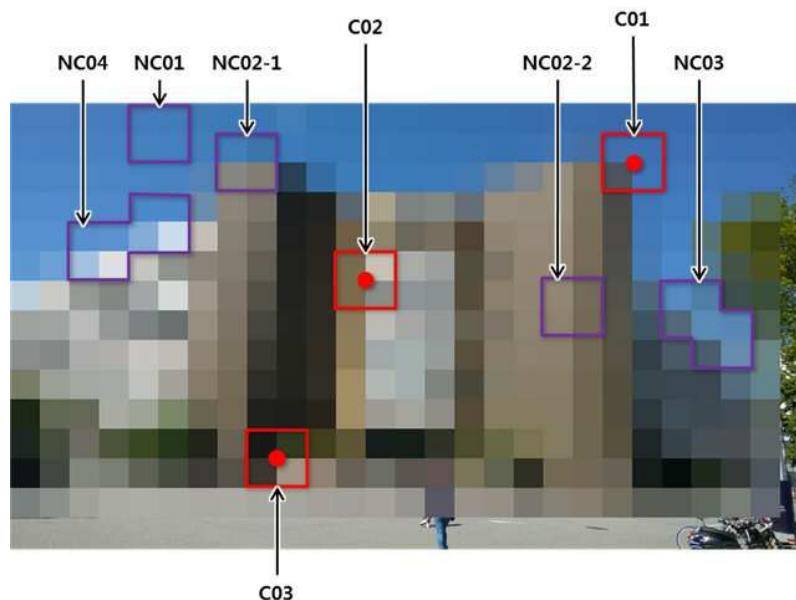
**입력 :** 사용자의 터치  
**출력 :** 사용자 터치 부분에서 특징점을 검출하여 거리나 좌표 표시

```

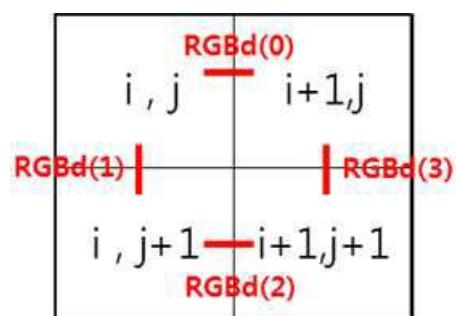
1 : public boolean onTouchEvent(){                                // 터치가 시작되었을 때
2 :     case ACTION_DOWN:                                         // 터치 시작점의 screen x,y 좌표 취득
3 :         t_x, t_y = event.getX(), event.getY();                // 터치 시작점의 screen 좌표를 이미지 평생좌표로 변환
4 :         p_x, p_y = Screen_to_image(t_x,t_y);                  // 터치영역이 왼쪽일 때 오른쪽에 확대 영상
5 :         if(t_x>screen_width/2) R_control_view(p_x,p_y);        // 터치영역이 오른쪽일 때 왼쪽에 확대 영상
6 :         else L_control_View(p_x,p_y);                         // 터치영역에 따른 영상 확대 방향에 따라 확대 영상
7 :         break;
8 :     case ACTION_MOVE:                                         // 터치 후 이동 하였을 때
9 :         t_x, t_y = event.getX(), event.getY();                // 터치점의 screen x,y 좌표 취득
10:        p_x, p_y = Screen_to_image(t_x,t_y);                 // 터치점의 screen 좌표를 이미지 평생좌표로 변환
11:        R_control_view (p_x,p_y) or L_control_View(p_x,p_y); // 이동방향에 따라 확대 영상 이동
12:        break;
13:     case ACTION_UP:                                           // 터치가 끝났을 때
14:         L or R_control_view().close();                         // 확대 영상을 닫음
15:         t_x, t_y = event.getX(), event.getY();                // 터치점의 screen x,y 좌표 취득
16:         p_x, p_y = Screen_to_image(t_x,t_y);                 // 터치점의 screen 좌표를 이미지 평생좌표로 변환
17:         e_Ax, e_Ay, e_d=특징점검출(p_x,p_y,101,101); // 터치점을 기준으로 101X101영역 특징점 검출
18:         show_Main_View_Mark(e_Ax,e_Ay);                      // e_Ax, e_Ay지점에 특징점 표시
19:         e_Bx,e_By = 특징점 매칭(B영상, e_Ax<700, e_Ay>0, e_d); // B영상에서 특징점 매칭
20:         Tx,Ty,Tz=좌표결정(e_Ax,e_Ay,e_Bx,e_By);          // A,B두영상 영상좌표 이용 대상을 좌표 파악
21:         length_t=거리계산(Ax,Ay,Az,Tx,Ty,Tz);            // A영상 위치에서 대상을 T까지 거리 계산
22:         show_Main_View_length(e_x,e_y,length_t);             // A영상에 대상을까지의 거리 표시
23:         break;

```

도면21a



도면21b



## 도면22

(a)

i-1,j-1	i, j-1	i+1,j-1	i+2,j-1
i-1,j	i, j	i+1,j	i+2,j
i-1,j+1	i, j+1	i+1,j+1	i+2,j+1
i-1,j+2	i, j+2	i+1,j+2	i+2,j+2

(b)

i-1,j-1	i, j-1	i+1,j-1	i+2,j-1
i-1,j	i, j	i+1,j	i+2,j
i-1,j+1	i, j+1	i+1,j+1	i+2,j+1
i-1,j+2	i, j+2	i+1,j+2	i+2,j+2

(c)

i-1,j-1	i, j-1	i+1,j-1	i+2,j-1
i-1,j	i, j	i+1,j	i+2,j
i-1,j+1	i, j+1	i+1,j+1	i+2,j+1
i-1,j+2	i, j+2	i+1,j+2	i+2,j+2

(d)

i-1,j-1	i, j-1	i+1,j-1	i+2,j-1
i-1,j	i, j	i+1,j	i+2,j
i-1,j+1	i, j+1	i+1,j+1	i+2,j+1
i-1,j+2	i, j+2	i+1,j+2	i+2,j+2

## 도면23

	$i-1, j-2$	$i, j-2$	$(+1), j-2$	$(+2), j-2$	
$i-2, j-1$	$i-1, j-1$	$(1)$	$(+1), j-1$	$(+2), j-1$	$(+3), j-1$
$i-1, j-2$	$(0)$	$i-1, j$	$i, j$	$i+1, j$	$i+2, j$
$(+2), j-1$	$(2)$	$i, j-1$	$i+1, j$	$i+1, j+1$	$(+2), j+1$
$(+2), j-2$	$i-1, j+2$	$i, j+2$	$(+1), j+2$	$(+2), j+2$	$(+3), j+2$
	$i-1, j+3$	$i, j+3$	$(+1), j+3$	$(+2), j+3$	

	$i-1, j-2$	$i, j-2$	$i+1, j-2$	$i+2, j-2$	
$i-2, j-1$	$i-1, j-1$	$i, j-1$	$i+1, j-1$	$i+2, j-1$	$i+3, j-1$
$i-2, j$	$i-1, j$	$i, j$	$i+1, j$	$i+2, j$	$i+3, j$
$i-2, j+1$	$i-1, j+1$	$i, j+1$	$i+1, j+1$	$i+2, j+1$	$i+3, j+1$
$i-2, j+2$	$i-1, j+2$	$i, j+2$	$i+1, j+2$	$i+2, j+2$	$i+3, j+2$
	$i-1, j+3$	$i, j+3$	$i+1, j+3$	$i+2, j+3$	

	$i-1,j-2$	$i,j-2$	$(+1,j)-2$	$(+2,j)-2$	
$i-2,j-1$	$i-1,j-1$	$i,j-1$	$(+1,j)-1$	$(+2,j)-1$	$(+3,j)-1$
$i-2,j$	$i-1,j$	$i,j$	$i+1,j$	$i+2,j$	$i+3,j$
$i-2,j+1$	$i-1,j+1$	$i,j+1$	$i+1,j+1$	$i+2,j+1$	$i+3,j+1$
$i-2,j+2$	$i-1,j+2$	$i,j+2$	$i+1,j+2$	$i+2,j+2$	$i+3,j+2$
	$i-1,j+3$	$i,j+3$	$i+1,j+3$	$i+2,j+3$	

	$i-1j-2$	$i,j-2$	$(i+1)j-2$	$(i+2)j-2$	
$(i-2)j-1$	$i-1j-1$	$i,j-1$	$i+1j-1$	$i+2j-1$	$(i+3)j-1$
$i-2j$	$i-1j$	$i,j$	$i+1j$	$(i-2)j$	$(i+3)j$
$(i-2)j+1$	$i-1j+1$	$i,j+1$	$i+1j+1$	$(i-2)j+1$	$(i+3)j+1$
$i-2j+2$	$i-1j+2$	$i,j+2$	$i+1j+2$	$i+2j+2$	$(i+3)j+2$
	$i-1j+3$	$i,j+3$	$i+1j+3$	$i+2j+3$	

	$i-1,j-2$	$i,j-2$	$(+1,j-2)$	$(+2,j-2)$	
$i-2,j-1$	$i-1,j-1$	$i,j-1$	$(+1,j-1)$	$(+2,j-1)$	$(+3,j-1)$
$i-2,j$	$i-1,j$	$i,j$	<del><math>i+1,j</math></del> $\textcircled{1}$	$(+2,j)$	$(+3,j)$
$i-2,j+1$	$i-1,j+1$	$i,j+1$	$i+1,j+1$	$(+2,j+1)$	$(+3,j+1)$
$i-2,j+2$	$i-1,j+2$	$i,j+2$	$(+1,j+2)$	$(+2,j+2)$	$(+3,j+2)$
	$i-1,j+3$	$i,j+3$	$(+1,j+3)$	$(+2,j+3)$	

	$i-1,j-2$	$i,j-2$	$i+1,j-2$	$i+2,j-2$	
$i-2,j-1$	$i-1,j-1$	$i,j-1$	$i+1,j-1$	$i+2,j-1$	$i+3,j-1$
$i-2,j$	$i-1,j$	$i,j$	$i+1,j$	$i+2,j$	$i+3,j$
$i-2,j+1$	$i-1,j+1$	$i,j+1$	$i+1,j+1$	$i+2,j+1$	$i+3,j+1$
$i-2,j+2$	$i-1,j+2$	$i,j+2$	$i+1,j+2$	$i+2,j+2$	$i+3,j+2$
	$i-1,j+3$	$i,j+3$	$i+1,j+3$	$i+2,j+3$	

	$i-1,j-2$	$i,j-2$	$j+1,j-2$	$i+2,j-2$	
$i-2,j-1$	$i-1,j-1$	$i,j-1$	$i+1,j-1$	$i+2,j-1$	$i+3,j-1$
$i-2,j$	$i-1,j$	$i,j$	$i+1,j$	$i+2,j$	$i+3,j$
$i-2,j+1$	$i-1,j+1$	$i,j+1$	$i+1,j+1$	$i+2,j+1$	$i+3,j+1$
$i-2,j+2$	$i-1,j+2$	$i,j+2$	$i+1,j+2$	$i+2,j+2$	$i+3,j+2$
	$i-1,j+3$	$i,j+3$	$i+1,j+3$	$i+2,j+3$	

	$i-1, j-2$	$i, j-2$	$(i+1)j-2$	$(i+2)j-2$	
$(i-2)j-1$	$i-1, j-1$	$i, j-1$	$(i+1)j-1$	$(i+2)j-1$	$(i+3)j-1$
$(i-2)j$	$(i-1)j$	$i, j$	$(i+1)j$	$(i+2)j$	$(i+3)j$
$(i-2)j+1$	$(i-1)j+1$	$i, j+1$	$(i+1)j+1$	$(i+2)j+1$	$(i+3)j+1$
$(i-2)j+2$	$(i-1)j+2$	$(i, j+2)$	$(i+1)j+2$	$(i+2)j+2$	$(i+3)j+2$
	$(i-1)j+3$	$(i, j+3)$	$(i+1)j+3$	$(i+2)j+3$	

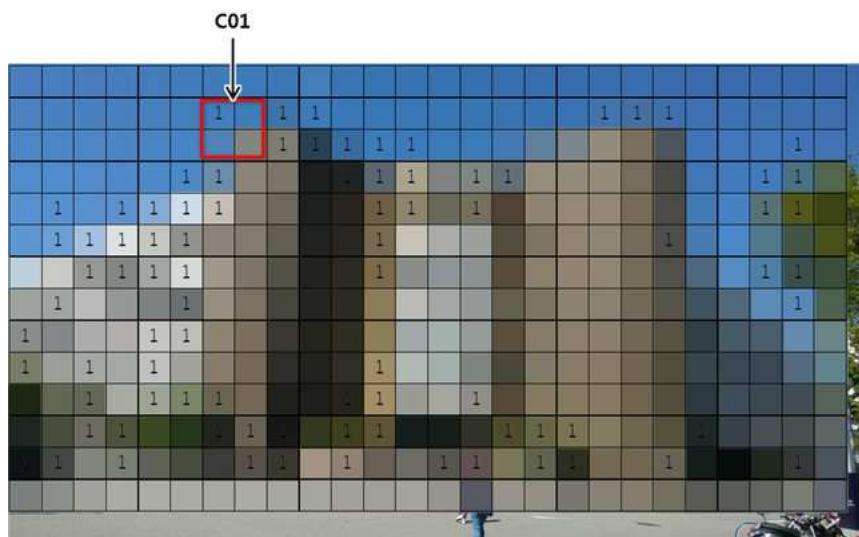
## 도면24

입력 : RGB값을 가지고 있는 모자이크 영상 배열, 기준값  
 출력 : 특징점 가능 영역 배열 : 모자이크 영상과 같은 좌표에 0(특징점 없음), 1(특징점 가능) 표시

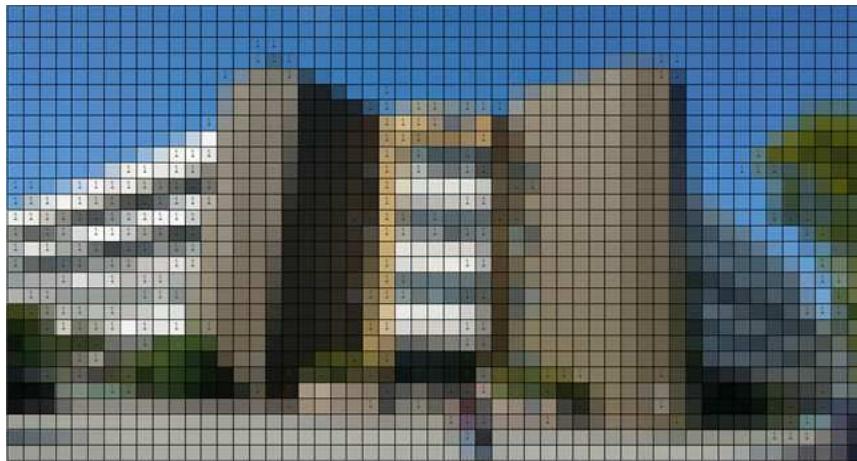
```

1: static int[][] 특징점가능영역검출(int[][][] 모자이크 영상, float 기준값 ){
2:     int m_w = 모자이크 영상 폭;
3:     int m_h = 모자이크 영상 높이;
4:     int[][] 특징점가능영역 = new int[m_w][m_h];
5:     for (int j=0 ; j<m_w-1 ; j++){
6:         for (int l=0 ; l<m_h-1 ; l++){
7:             // 수직, 수평 경계 분석
8:             RGBd[0]=RGB유사도((i,j)픽셀, (i+1,j)픽셀);
9:             RGBd[1]=RGB유사도((i,j)픽셀, (i,j+1)픽셀);
10:            RGBd[2]=RGB유사도((i,j+1)픽셀, (i+1,j+1)픽셀);
11:            RGBd[3]=RGB유사도((i+1,j+1)픽셀, (i+1,j)픽셀);
12:            if ((RGBd[0]>기준값 && RGBd[2]>기준값) || (RGBd[1]>기준값 && RGBd[3]>기준값))
13:                {특징점가능영역[i][j]=1;} // 0:특징점가능영역이 아님
14:            else
15:                {특징점가능영역[i][j]=1;} // 1:특징점가능영역
16:            // 1대각 분석 (가로, 세로 1픽셀씩 이동한 경우) 분석
17:            if (특징점가능영역판단[i][j]==1 && i>0 && j>0 && j<m_h-2 && i<m_w-2){
18:                도면 22와 같은 4가지 RGB유사도 계산
19:                if ((0)>기준값 && (1)>기준값 && (2)>기준값 && (3)>기준값) 특징점가능영역[i][j]=0
20:            }
21:            // 2대각 분석 (가로, 세로 중 2픽셀씩 이동한) 경우 분석
22:            if (특징점가능영역판단[i][j]==1 && i>1 && j>1 && j<m_h-3 && i<m_w-3){
23:                도면 23과 같은 8가지 RGB유사도 계산
24:                if ((0)>기준값 && (1)>기준값 && (2)>기준값 && (3)>기준값) 특징점가능영역[i][j]=0
25:            }
26:        }
27:    }
28:    return 특징점가능영역;
29: }
```

## 도면25a



## 도면25b



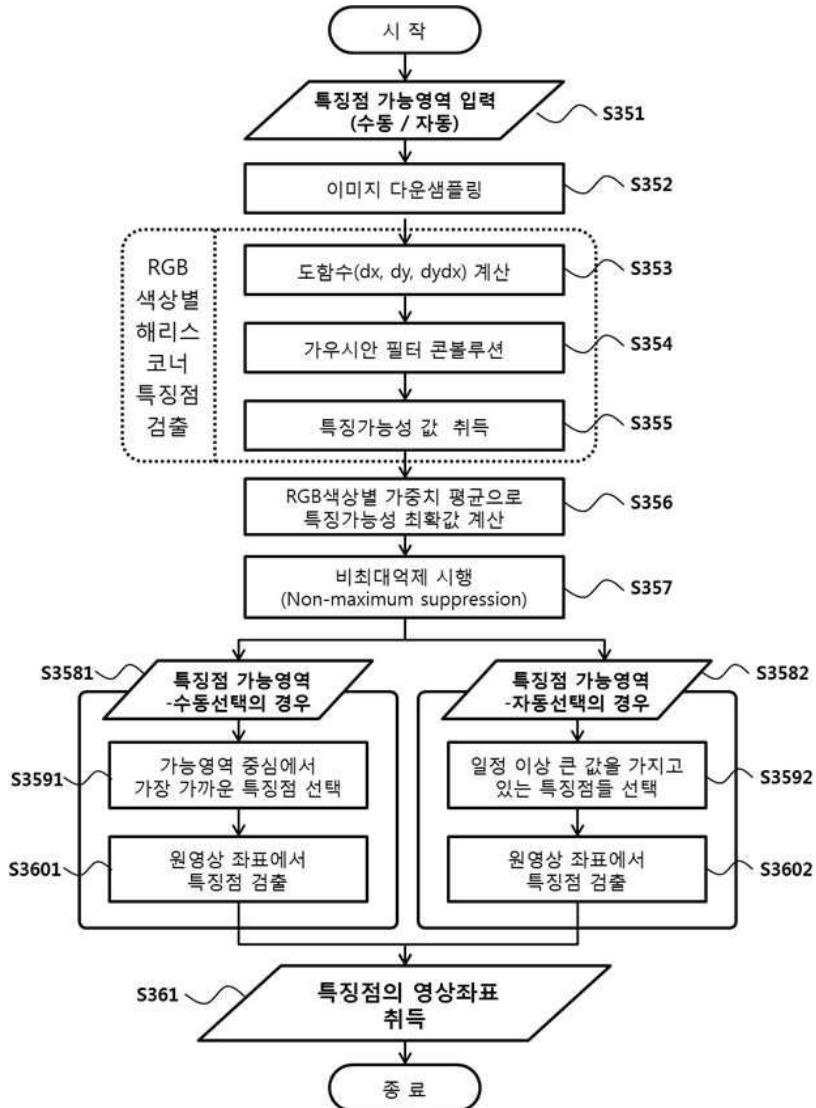
## 도면26

입력 : RGB값을 가지고 있는 영상 배열 : **Source\_mat**  
출력 : 입력영상과 같은 크기를 가지고 있는 RGB 특징 가능성 값 배열

```

1: static float[][] 해리스코너특징점검출 {int[영상폭][영상높이][RGB] Source_mat{
2:     int s_w = Source_mat.length; // 영상 폭
3:     int s_h = Source_mat[0].length; // 영상 높이
4:     float[][] 해리스코너특징점맵 = new float[s_w][s_h]; //결과 값
5:     float[][] dx = new float[s_w][s_h]; // 영상 x축 도함수 결과
6:     float[][] dy = new float[s_w][s_h]; // 영상 y축 도함수 결과
7:     float[][] dydx = new float[s_w][s_h]; // 영상 xy축 도함수 결과
8:     float Gdy2, Gdx2, Gdydx; //컨볼루션 연산결과
9:     float C // 특징 가능성값
10:    float[][] G_F = new float[3][3]; //가우시안 필터
11:    G_F[0][0]=0.0751f ; G_F[1][0] = 0.1238f ; G_F[2][0]=0.0751f;
12:    G_F[0][1]=0.1238f ; G_F[1][1] = 0.2042f ; G_F[2][1]=0.1238f;
13:    G_F[0][2]=0.0751f ; G_F[1][2] = 0.1238f ; G_F[2][2]=0.0751f;
14:    // 영상 도함수(dx, dy, dydx) 배열 생성
15:    for (int RGB=0 ; RGB<3 ; RGB++){
16:        for (int j=1 ; j<s_h-1 ; j++){
17:            for (int i=1 ; i<s_w-1 ; i++){
18:                dx[i][j][RGB] = Source_mat[i-1][j][RGB]+1 + Source_mat[i+1][j][RGB];
19:                dy[i][j][RGB] = Source_mat[i][j-1][RGB]+1 + Source_mat[i][j+1][RGB];
20:                dydx[i][j][RGB] = dy[i][j][RGB]*dx[i][j][RGB];
21:            }
22:        }
23:        // 콘볼루션 연산(가우시안 필터와 도함수) 수행
24:        // 특징가능성 값 계산 C=(pq-r^2)-k(p+q)^2
25:        for (int RGB=0 ; RGB<3 ; RGB++){
26:            for (int j=1 ; j<s_h-1 ; j++){
27:                for (int i=1 ; i<s_w-1 ; i++){
28:                    Gdy2=dy[i-1][j-1][RGB]*2+G_F[0][0]*dy[i][j-1][RGB]*2+G_F[1][0]*dy[i+1][j-1][RGB]*2+G_F[2][0]
29:                        +dy[i-1][j][RGB]*2 * G_F[0][1]*dy[i][j][RGB]*2 +G_F[1][1]*dy[i+1][j][RGB]*2 * G_F[2][1]
30:                        +dy[i-1][j+1][RGB]*2+G_F[0][2]*dy[i][j+1][RGB]*2+G_F[1][2]*dy[i+1][j+1][RGB]*2+G_F[2][2];
31:                    Gdx2=dx[i-1][j-1][RGB]*2+G_F[0][0]*dx[i][j-1][RGB]*2+G_F[1][0]*dx[i+1][j-1][RGB]*2+G_F[2][0]
32:                        +dx[i-1][j][RGB]*2 * G_F[0][1]*dx[i][j][RGB]*2 +G_F[1][1]*dx[i+1][j][RGB]*2 * G_F[2][1]
33:                        +dx[i-1][j+1][RGB]*2+G_F[0][2]*dx[i][j+1][RGB]*2+G_F[1][2]*dx[i+1][j+1][RGB]*2+G_F[2][2];
34:                    Gdydx=Gdyx[i-1][j-1][RGB]+G_F[0][0]*dydx[i][j-1][RGB]*G_F[1][0]*dydx[i+1][j-1][RGB]*G_F[2][0]
35:                        +dydx[i-1][j][RGB]*G_F[0][1]*dydx[i][j][RGB]*G_F[1][1]*dydx[i+1][j][RGB]*G_F[2][1]
36:                        +dydx[i-1][j+1][RGB]*G_F[0][2]*dydx[i][j+1][RGB]*G_F[1][2]*dydx[i+1][j+1][RGB]*G_F[2][2];
37:                    C = (Gdy2+Gdx2-Gdydx+Gdydx) - 0.04f*(Gdy2+Gdx2)*(Gdy2+Gdx2);
38:                    해리스코너특징점맵[i][j][RGB] = C
39:                }
40:            }
41:        }
42:    return 해리스코너특징점맵;
43: }
```

## 도면27



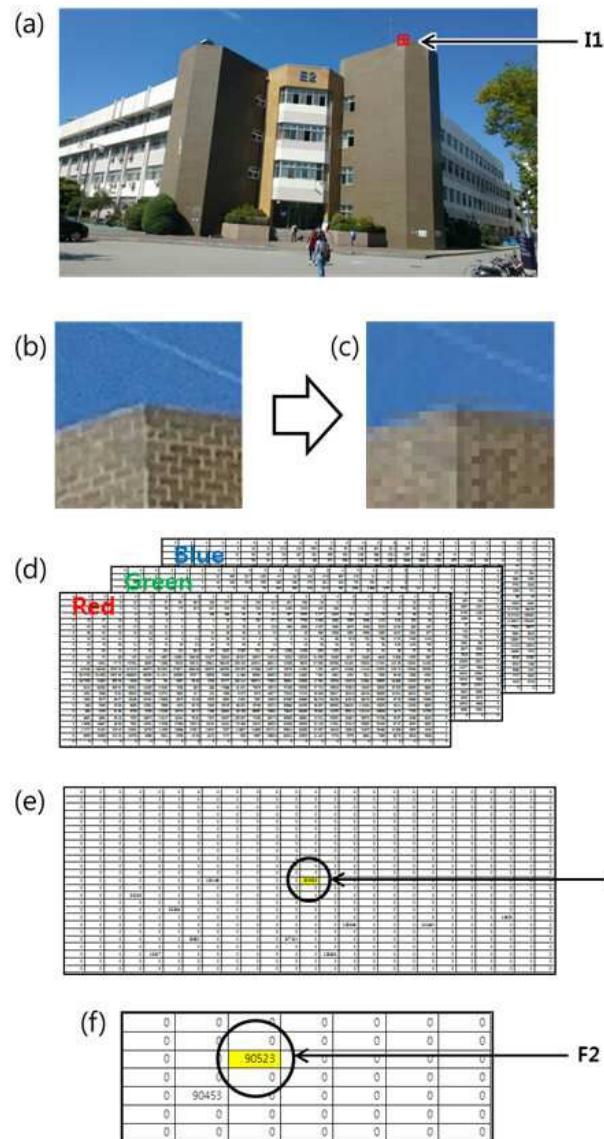
## 도면28

입력 : RGB채널별 특징가능성값 배열: Source\_mat, c\_v: 기준, Red, Green, Blue: 해당영역의 RGB평균  
출력 : 입력영상과 같은 크기를 가지고 있는 비최대억제가 끝난 특징 가능성값 배열

```

1: static float[][] 비최대억제 (float[][][] Source_mat, float c_v, int Red, int Green, int Blue){
2:     int s_w = Source_mat.length; // 영상의 폭
3:     int s_h = Source_mat[0].length; // 영상의 높이
4:     float[][][] RGB_NMS_out = new float[s_w][s_h]; // 결과 값
5:     float max_temp;
6:     float wsum = 1f/Red+1f/Green+1f/Blue; //RGB채널별 가중치를 계산하기 위한 가중치합
7:     float wr = 1f/Red / wsum; // wr:Red채널 가중치
8:     float wg = 1f/Green / wsum; // wg:Green채널 가중치
9:     float wb = 1f/Blue / wsum; // wb:Blue채널 가중치
10:    float[][] com = new float[3][3]; // 비교할 대상
11:    for (int j = 1; j < s_w - 1; j++) {
12:        for (int i = 1; i < s_h - 1; i++) {
13:            // 가중치에 따른 특징가능성값 계산
14:            //Source_mat[][][0]:Red, Source_mat[][][1]:Green, Source_mat[][][2]:Blue
15:            com[0][0]=Source_mat[i-1][j-1][0]*wr+Source_mat[i-1][j-1][1]*wg+Source_mat[i-1][j-1][2]*wb;
16:            com[1][0]=Source_mat[i][j-1][0]*wr+Source_mat[i][j-1][1]*wg+Source_mat[i][j-1][2]*wb;
17:            com[2][0]=Source_mat[i+1][j-1][0]*wr+Source_mat[i+1][j-1][1]*wg+Source_mat[i+1][j-1][2]*wb;
18:            com[0][1]=Source_mat[i-1][j][0]*wr+Source_mat[i-1][j][1]*wg+Source_mat[i-1][j][2]*wb;
19:            com[1][1]=Source_mat[i][j][0]*wr+Source_mat[i][j][1]*wg+Source_mat[i][j][2]*wb;
20:            com[2][1]=Source_mat[i+1][j][0]*wr+Source_mat[i+1][j][1]*wg+Source_mat[i+1][j][2]*wb;
21:            com[0][2]=Source_mat[i-1][j+1][0]*wr+Source_mat[i-1][j+1][1]*wg+Source_mat[i-1][j+1][2]*wb;
22:            com[1][2]=Source_mat[i][j+1][0]*wr+Source_mat[i][j+1][1]*wg+Source_mat[i][j+1][2]*wb;
23:            com[2][2]=Source_mat[i+1][j+1][0]*wr+Source_mat[i+1][j+1][1]*wg+Source_mat[i+1][j+1][2]*wb;
24:            //비최대억제 (중심 (i,j)를 기준으로 8연결방식 비교로 비최대억제 시행)
25:            max_temp = Math.max(com[0][0],com[1][0]);
26:            max_temp = Math.max(max_temp, com[2][0]);
27:            max_temp = Math.max(max_temp, com[0][1]);
28:            max_temp = Math.max(max_temp, com[1][1]);
29:            max_temp = Math.max(max_temp, com[2][1]);
30:            max_temp = Math.max(max_temp, com[0][2]);
31:            max_temp = Math.max(max_temp, com[1][2]);
32:            max_temp = Math.max(max_temp, com[2][2]);
33:            if (com[i][j] == max_temp && com[i][j] > c_v)
34:                RGB_NMS_out[i][j] = com[i][j];
35:            else RGB_NMS_out[i][j] = 0;
36:        }
37:    }
38:    return RGB_NMS_out;
39: }
```

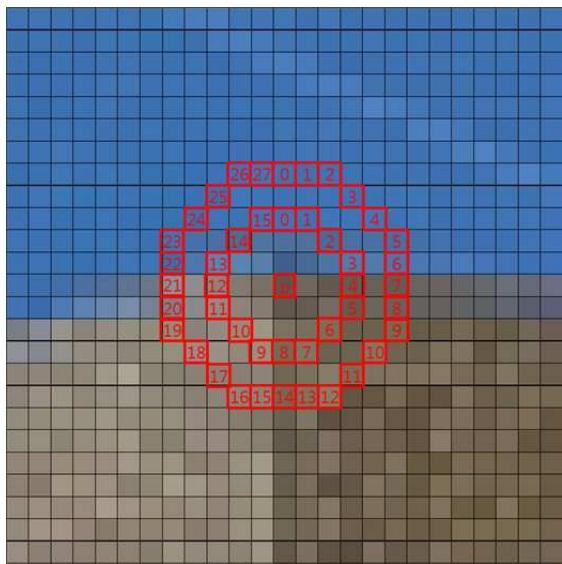
## 도면29



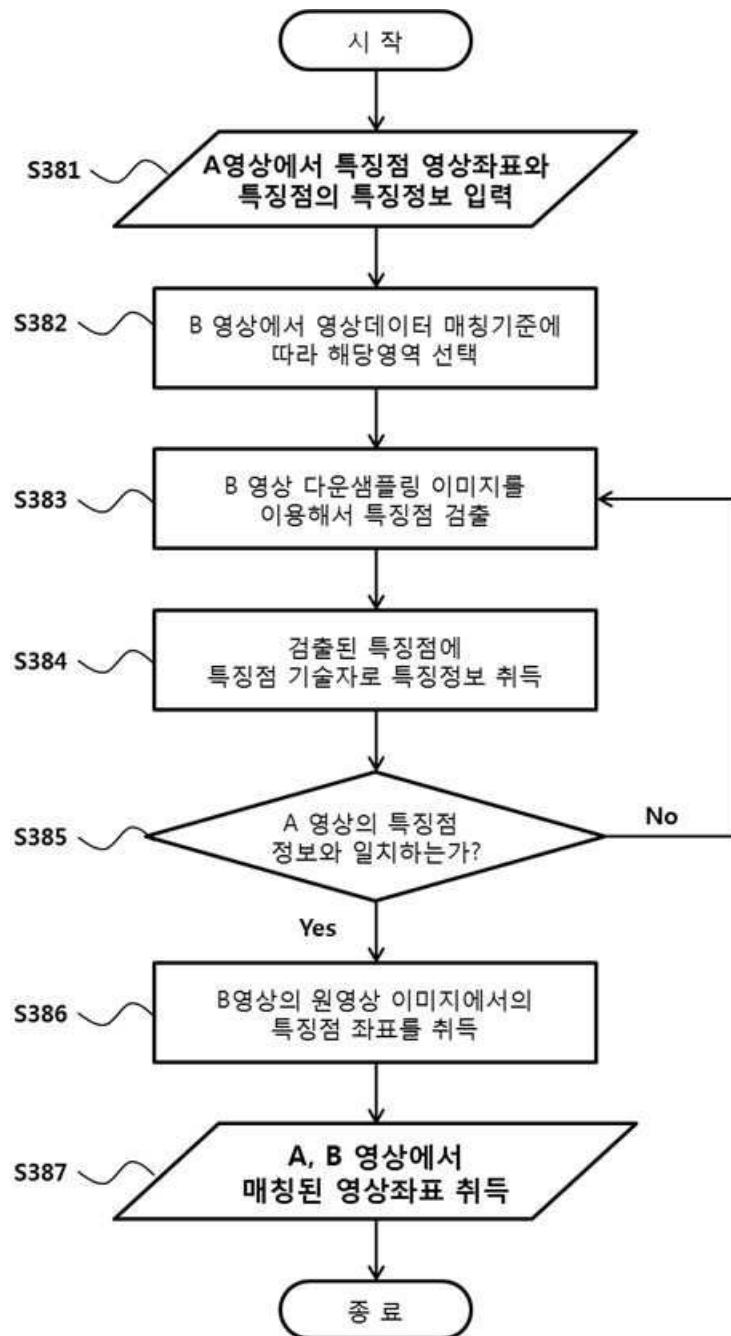
## 도면30a

	i-5	i-4	i-3	i-2	i-1	i	i+1	i+2	i+3	i+4	i+5
j-5				26	27	0	1	2			
j-4			25					3			
j-3		24			15	0	1		4		
j-2	23			14			2			5	
j-1	22		13					3		6	
j	21		12		P			4		7	
j+1	20		11				5		8		
j+2	19		10			6			9		
j+3	18			9	8	7			10		
j+4		17					11				
j+5		16	15	14	13	12					

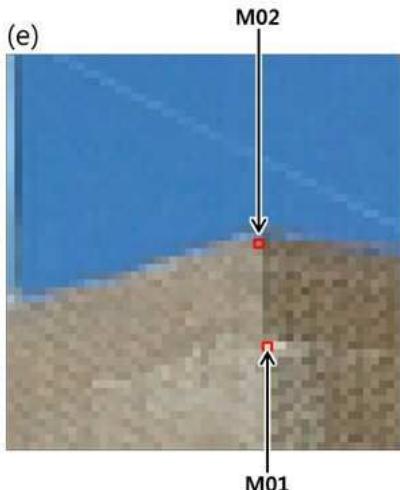
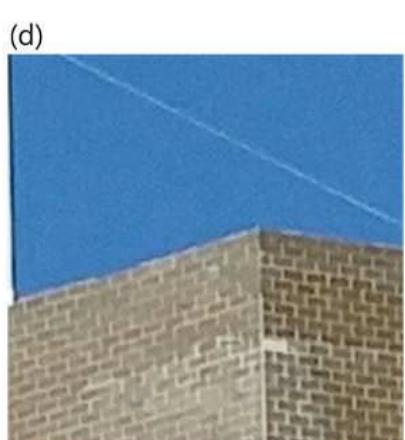
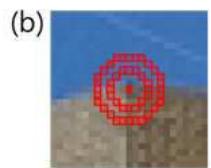
### 도면30b



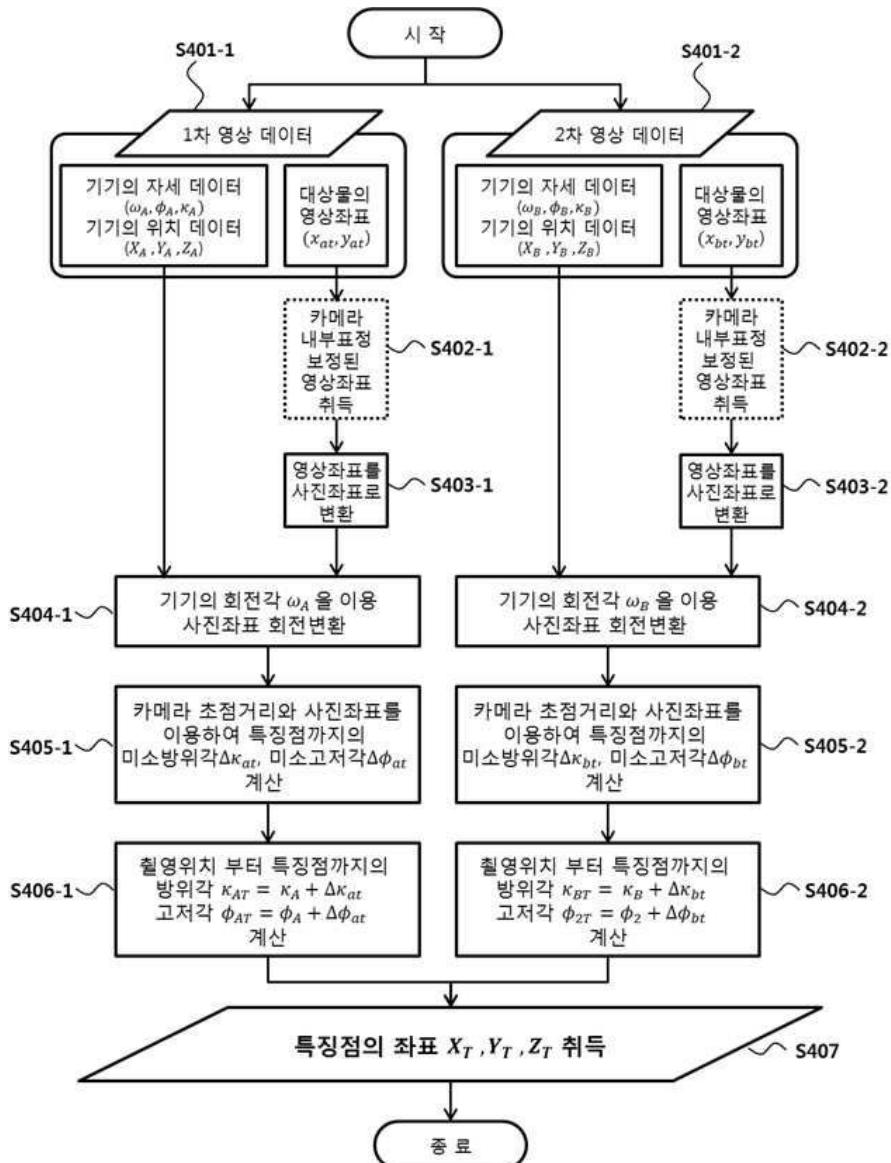
## 도면31



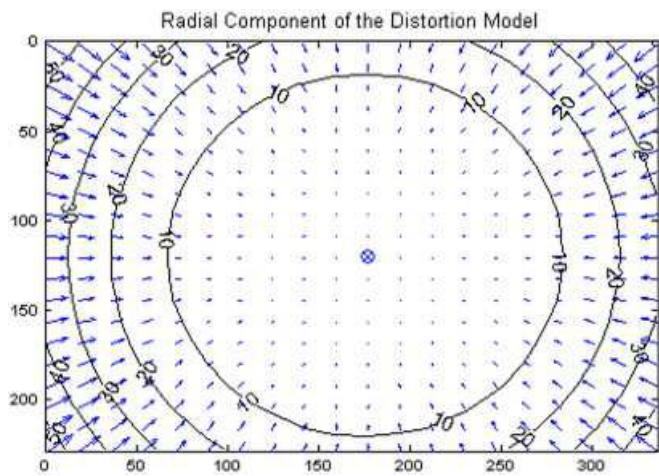
도면32



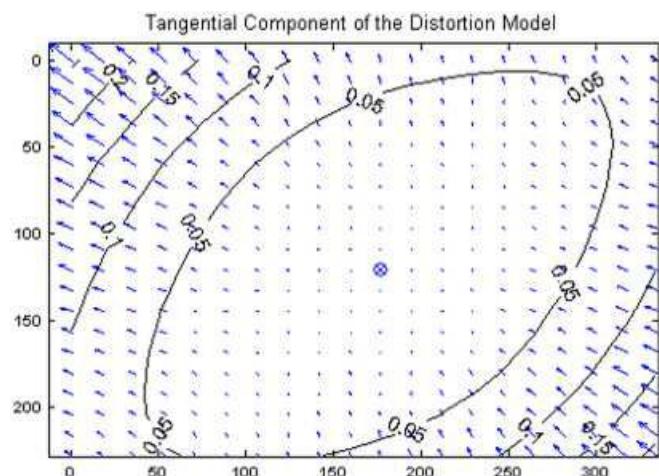
## 도면33



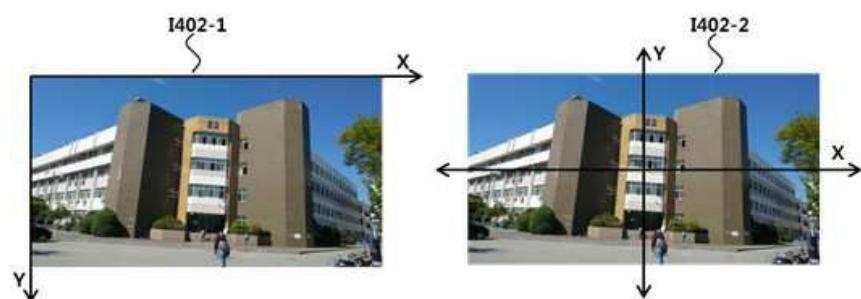
## 도면34a



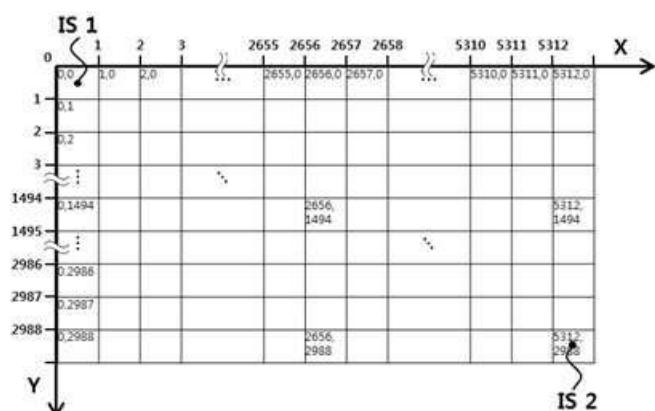
도면34b



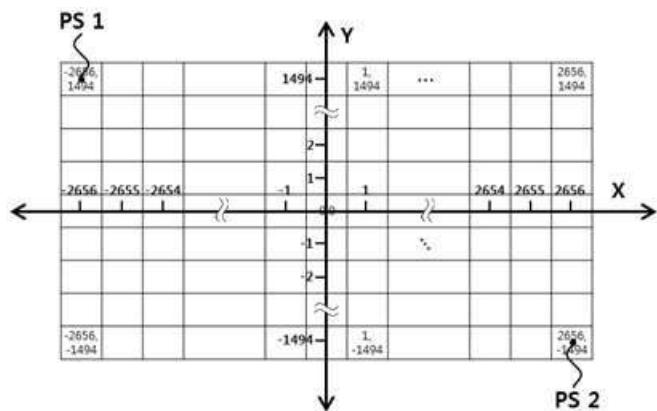
도면35a



도면35b



도면35c



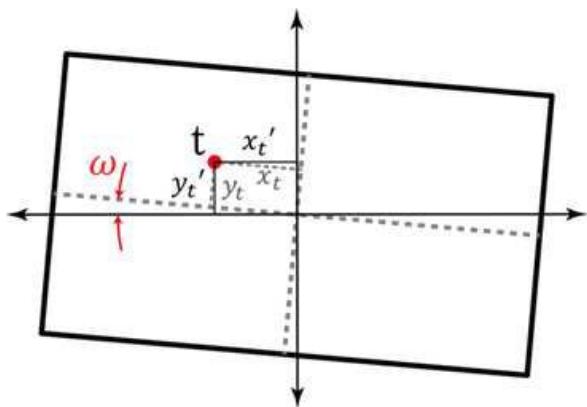
도면36a



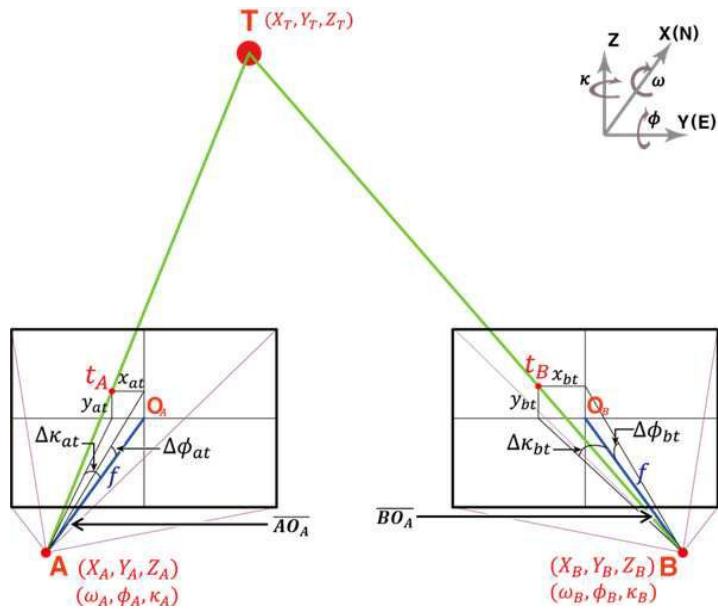
도면36b



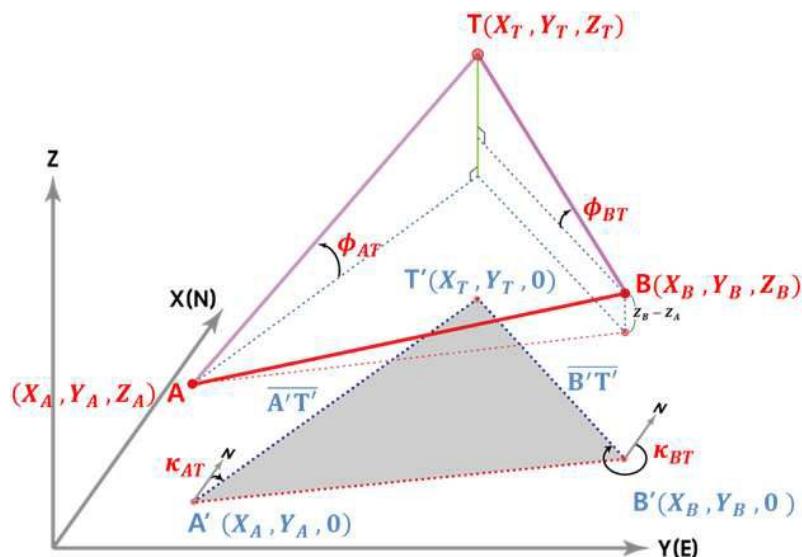
도면36c



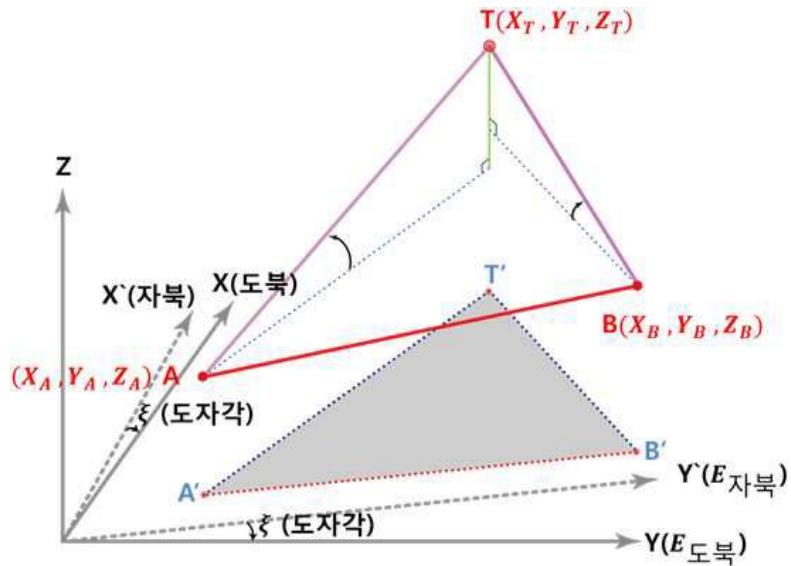
도면37



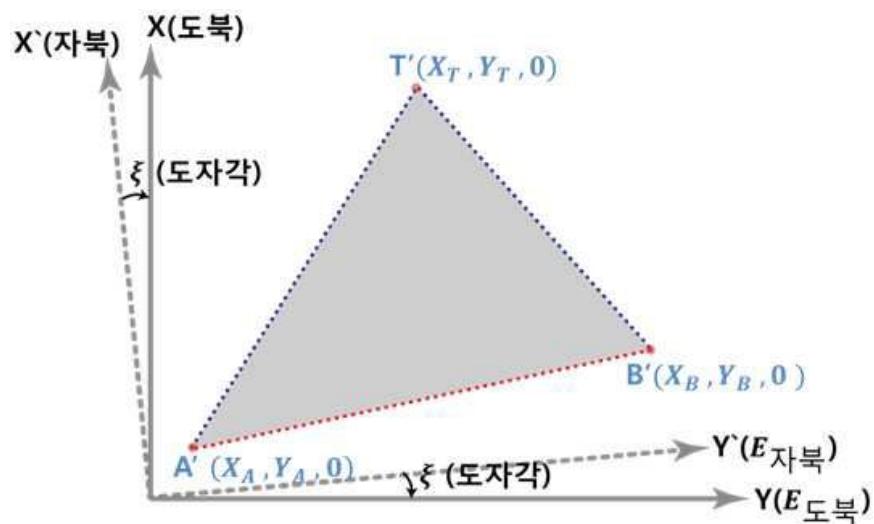
도면38



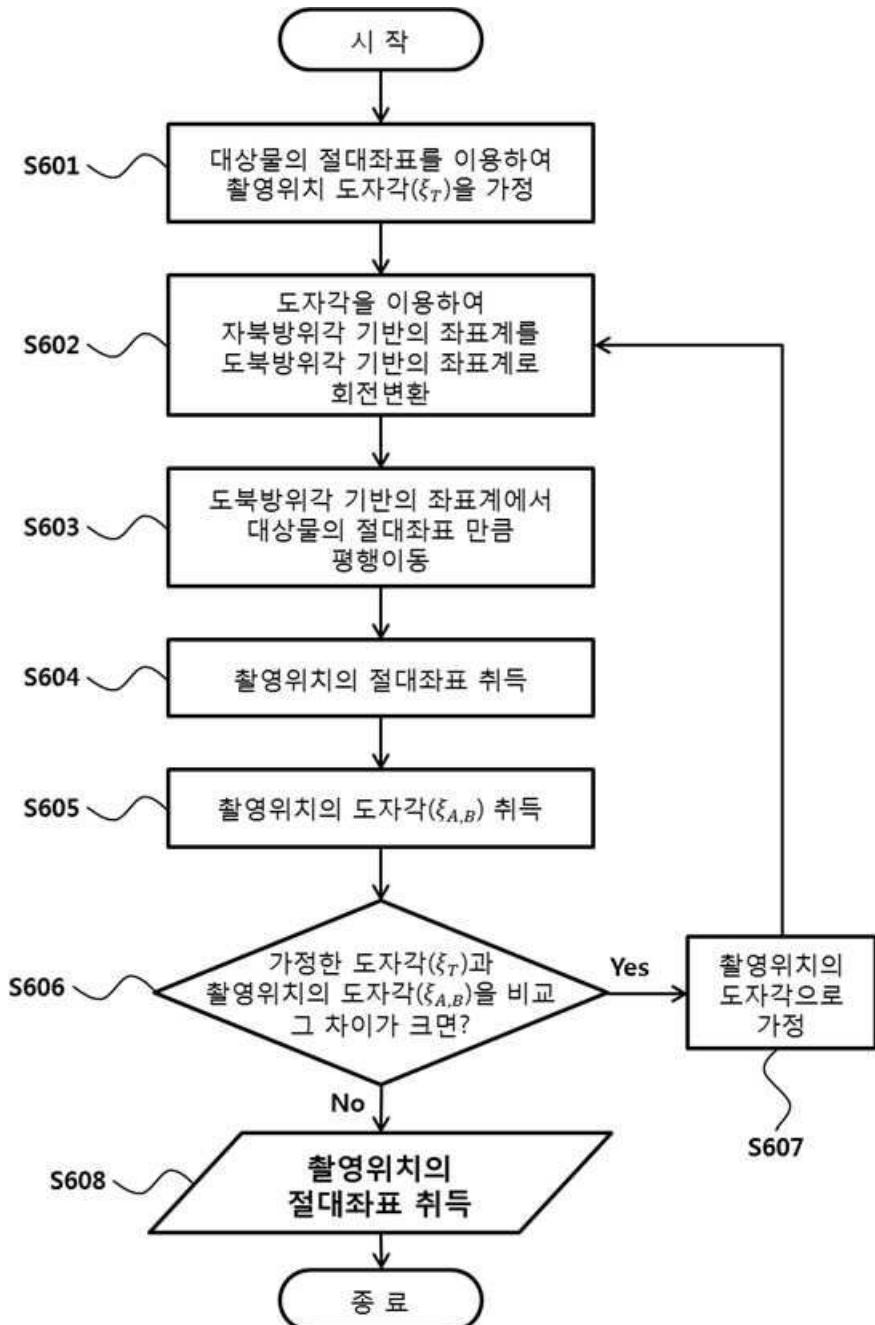
도면39a



도면39b



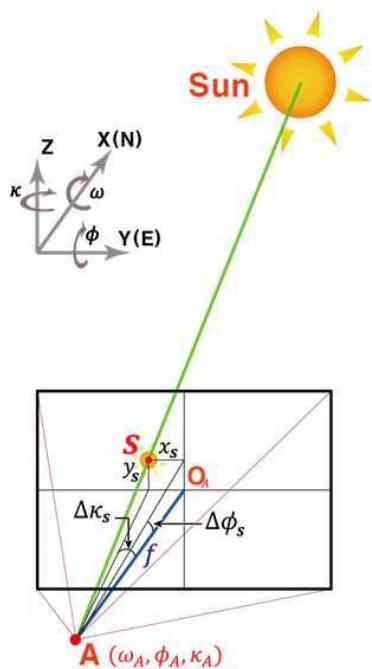
## 도면40



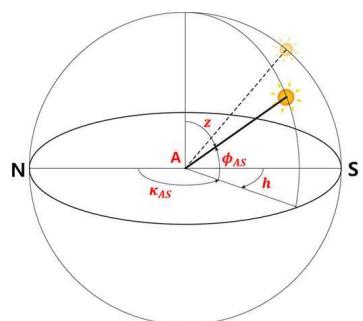
## 도면41a



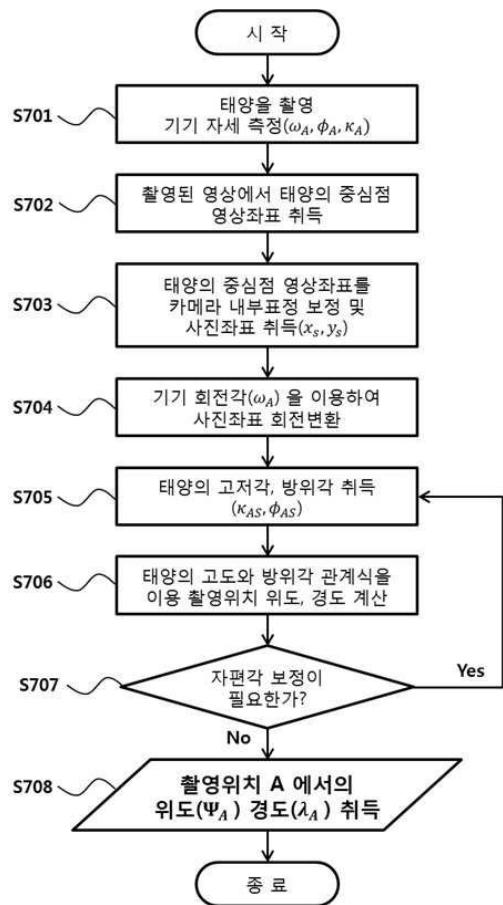
도면41b



도면42



## 도면43



## 도면44a



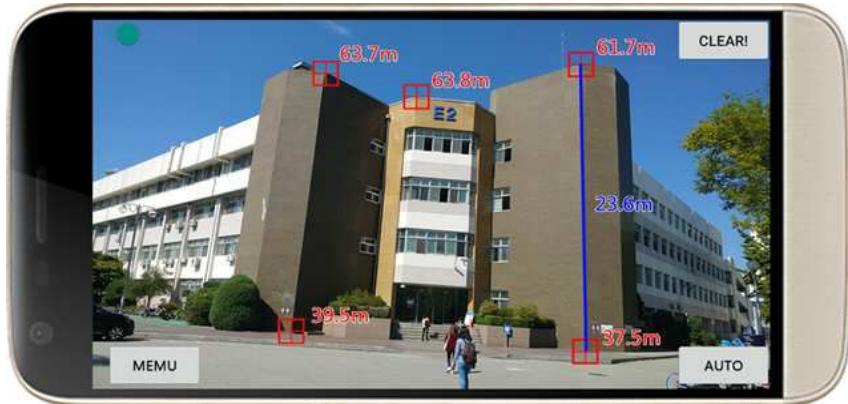
도면44b



도면44c



도면45a



도면45b

