

# Improved CS-RANSAC Algorithm Using K-Means Clustering

Seunghyun Ko<sup>†</sup> · Ui-Nyoung Yoon<sup>†</sup> · Jumabek Alikhanov<sup>\*\*</sup> · Geun-Sik Jo<sup>\*\*\*</sup>

## ABSTRACT

Estimating the correct pose of augmented objects on the real camera view efficiently is one of the most important questions in image tracking area. In computer vision, Homography is used for camera pose estimation in augmented reality system with markerless. To estimating Homography, several algorithm like SURF features which extracted from images are used. Based on extracted features, Homography is estimated. For this purpose, RANSAC algorithm is well used to estimate homography and DCS-RANSAC algorithm is researched which apply constraints dynamically based on Constraint Satisfaction Problem to improve performance. In DCS-RANSAC, however, the dataset is based on pattern of feature distribution of images manually, so this algorithm cannot classify the input image, pattern of feature distribution is not recognized in DCS-RANSAC algorithm, which lead to reduce it's performance. To improve this problem, we suggest the KCS-RANSAC algorithm using K-means clustering in CS-RANSAC to cluster the images automatically based on pattern of feature distribution and apply constraints to each image groups. The suggested algorithm cluster the images automatically and apply the constraints to each clustered image groups. The experiment result shows that our KCS-RANSAC algorithm outperformed the DCS-RANSAC algorithm in terms of speed, accuracy, and inlier rate.

**Keywords :** RANSAC, CS-RANSAC, K-Means Clustering

## K-Means 클러스터링을 적용한 향상된 CS-RANSAC 알고리즘

고 승 현<sup>†</sup> · 윤 의 녕<sup>†</sup> · Jumabek Alikhanov<sup>\*\*</sup> · 조 근 식<sup>\*\*\*</sup>

## 요 약

이미지를 기반으로 하는 증강현실 시스템에서 가상의 객체를 실제 영상에 저작할 때 생기는 이질감을 줄이기 위해서는 실제 영상에 저작된 가상객체의 방향과 위치에 대해 정확하게 추정을 해야 하며, 이때 호모그래피를 사용한다. 호모그래피를 추정하기 위해서는 SURF와 같은 특징점을 추출하고 추출된 특징점들을 통해 호모그래피 행렬을 추정한다. 호모그래피 행렬의 추정을 위해서 RANSAC 알고리즘이 주로 사용되고 있으며, 특히 RANSAC에 제약 조건 만족 문제(Constraint Satisfaction Problem)와 여기에 사용되는 제약조건을 동적으로 적용하여 속도와 정확도를 높인 DCS-RANSAC 알고리즘이 연구되었다. DCS-RANSAC 알고리즘에서 사용된 이미지 그룹 데이터는 수동적인 방법을 통해 직관적으로 분류되어 있지만 특징점 분포 패턴이 다양하지 않고, 이미지들을 정확하게 분류하기가 어려워서 이로 인해 알고리즘의 성능이 저하되는 경우가 있다. 따라서 본 논문에서는 K-means 클러스터링을 적용하여 이미지들을 자동으로 분류하고 각 이미지 그룹마다 각기 다른 제약조건을 적용하는 KCS-RANSAC 알고리즘을 제안한다. 제안하는 알고리즘은 머신러닝 기법인 K-means 클러스터링을 사용하여 전처리 단계에서 이미지를 특징점 분포 패턴에 따라 자동으로 분류하고, 분류된 이미지에 제약조건을 적용하여 알고리즘의 속도와 정확도를 향상시켰다. 실험 결과 본 논문에서 제안하는 KCS-RANSAC이 DCS-RANSAC 알고리즘에 비해 수행시간이 약 15% 단축되었고, 오차율은 약 35% 줄어들었으며, 참정보 비율은 약 14% 증가되었다.

**키워드 :** RANSAC, CS-RANSAC, K-Means Clustering

## 1. 서 론

증강현실은 실제 영상위에 컴퓨터로 제작한 가상의 객체를 이질감 없이 증강시켜 실제로 존재하는 사물처럼 보이게 하는 기술로 최근 이 기술에 대한 연구가 활발히 진행되고 있다. 이러한 증강현실 기술은 교육, 의료, 군사, 게임, 정비 등 여러 분야에 적용되어 있으며, 특히 비행기 정비와 같이 다양하고 복잡한 정보를 정확하게 제공해야 하는 산업분야에서 사용자에게 보다 정확하고 직관적인 정보를 제공할 수 있는

※ 이 논문은 2016년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2015-R1A2A2A03006190).

※ 이 논문은 2016년도 한국정보처리학회 추계학술발표대회에서 '머신러닝 기법을 적용한 CS-RANSAC 알고리즘'의 제목으로 발표된 논문을 확장한 것이다.

† 준 회 원 : 인하대학교 컴퓨터공학과 박사과정

\*\* 준 회 원 : 인하대학교 컴퓨터정보공학과 석사

\*\*\* 중 심 회 원 : 인하대학교 컴퓨터정보공학과 교수

Manuscript Received : December 21, 2016

First Revision : January 23, 2017

Accepted : January 23, 2017

\* Corresponding Author : Geun-Sik Jo(gsj@inha.ac.kr)

이점을 가진다[1-4]. 이와 같은 정보를 이질감 없이 사용자에게 제공하기 위해서는 가상의 객체를 원하는 위치에 정확하게 증강시켜야 하고, 이를 위해 증강시킬 대상을 정확히 인식하고 추적하는 기술이 요구된다. 이러한 추적 및 객체 인식 기술은 크게 센서기반, 컴퓨터시각기반으로 나눌 수 있다.

첫 번째, 센서를 기반으로 하는 기술은 음향, 관성, 광학, 기계센서들을 이용하는 기술로 각 센서들 고유의 장단점을 이용해서 객체를 빠른 속도로 인식 및 추적을 할 수 있지만 정확도가 떨어지는 단점이 존재한다.

두 번째, 컴퓨터의 시각을 기반으로 하는 기술은 카메라와 영상처리 알고리즘을 사용하는 기술로 비마커기반(Markerless), 마커기반(Marker) 두 방법으로 나누어진다. 마커기반 기술의 경우에는 마커를 사용하여 실제 구현이 쉬운 반면 적용할 수 있는 범위가 제한적이고, 증강시키고자 하는 대상에 마커를 설치해야 하는 현실적이지 않은 단점이 있다. 그러나 비마커기반 기술의 경우에는 실제 환경에서 흔히 볼 수 있는 특징점들(선, 점, 텍스처, 엣지 등)을 기반으로 카메라의 방향을 정하는 방식을 사용한다.

비마커 기반 증강현실 추적 기술 중 하나인 호모그래피는 점, 선, 엣지 등을 이용하여 카메라의 방향, 자세를 추정하여 실제 영상위에 가상의 객체를 정확하고 직관적으로 증강하여 사용자에게 보다 정확한 정보를 제공할 수 있다.

이러한 호모그래피를 구하기 위해서는 주로 RANSAC (RANDOM Sample Consensus) 알고리즘이 사용되며, 이 알고리즘을 이용해 호모그래피를 효율적으로 추정할 수 있다. RANSAC 알고리즘은 SURF (Speeded Up Robust Features)[5]와 같은 특징점들을 임의로 선택하는 랜덤 샘플링방식으로 호모그래피를 계산하는 방식이다. 이때 선택된 특징점들의 분포가 선형 혹은 군집을 이루는 경우 부정확한 호모그래피를 추정하게 되는 문제가 발생한다. 이러한 문제점을 해결 하기위해 특징점들을 샘플링하는 과정에서 제약 조건 만족 문제를 적용한 CS-RANSAC (Constraint Satisfaction problem-RANSAC) 알고리즘이 연구 되었다[6, 7].

그러나 CS-RANSAC 알고리즘은 대부분의 이미지 그룹에서 좋은 성능을 보이지만 랜덤샘플링에 의해 특징점들을 선택하기 때문에 부정확한 호모그래피를 추정 할 수 있는 특징점들을 다시 선택되어 결국에는 알고리즘의 성능이 저하되는 문제가 발생한다. 이 문제를 해결하기 위해 이미지들의 특징점 분포 패턴에 따라 분류된 그룹에 각기 다른 단위격자 크기를 동적으로 적용한 DCS-RANSAC (Dynamic CS-RANSAC) 알고리즘이 연구 되었다[12].

DCS-RANSAC 알고리즘은 이미지들의 특징점 분포 패턴에 따라 나누어진 이미지 그룹에 제약 조건 중 하나인 단위격자 크기를 동적으로 적용시킨 알고리즘이다. 그러나 알고리즘에서 사용된 이미지 그룹들은 특징점 분포 패턴에 따라 수동적인 방법을 통해 직관적으로 분류되어 특정 이미지들의 특징점 분포 패턴을 반영하지 않아 이미 분류된 이미지 그룹에 속하지 않은 이미지로 알고리즘을 실행 할 경우에는 성능이 저하되는 경우가 발생한다.

본 논문에서는 머신러닝 기법인 K-means 클러스터링 알

고리즘을 사용하여 입력된 이미지들을 특징점 분포 패턴에 따라 자동으로 분류하는 방법을 제안한다. 제안하는 알고리즘에서는 전처리 단계에서 이미지들에서 SURF 특징점을 추출한 후 특징점들을 K-means 클러스터링 알고리즘을 통해 클러스터링 하여 이미지 그룹들을 특징점 분포 패턴에 따라 분류하고 입력된 이미지들은 클러스터에 의해 이미 분류된 이미지 그룹에 자동으로 분류한다.

본 논문의 전체 구성은 다음과 같다. 2장에서는 RANSAC 알고리즘, K-means clustering 알고리즘에 대해 기술하고 RANSAC 알고리즘을 사용하여 호모그래피 행렬을 구할 때, 알고리즘의 랜덤 샘플링과정에서 발생하는 문제점을 보완한 다른 연구들에 대해 기술한다. 3장에서는 K-means clustering을 적용한 향상된 CS-RANSAC 알고리즘 방법을 제안하고, 4장에서는 실험 및 분석을 통하여 본 연구의 타당성을 검증한다. 실험을 통해 제안된 알고리즘의 성능을 기존 알고리즘의 성능과 비교하여 분석한다. 마지막으로 5장에서는 결론 및 향후 연구에 대해 논의한다.

## 2. 배경 지식 및 관련 연구

### 2.1 K-means clustering Algorithm

K-means clustering 알고리즘은 비지도 학습방법 알고리즘 중 하나이며 주어진 데이터를 K개의 클러스터 수로 분류하는 알고리즘이다. N개의 데이터를 입력받으면 입력된 데이터를 N보다 작거나 같은 K개의 그룹으로 나누며 이 과정에서 거리 기반의 그룹 간 비유사도와 같은 비용 함수를 최소화하는 방식으로 이루어지며 같은 그룹 내의 유사도는 증가하고 다른 그룹간의 유사도는 감소한다[11]. 여기서 N개의 데이터는 데이터베이스에 저장된 이미지들이고 K개의 클러스터 수는 이미지들의 특징점 분포패턴에 따라 분류된 이미지 그룹의 수이다.

### 2.2 CS-RANSAC

호모그래피를 추정하기 위해 사용되는 방법 중에 널리 사용되고 있는 RANSAC 알고리즘은 구하고자 하는 수학적 모델을 계산하기 위해 필요한 최소한의 데이터를 랜덤으로 샘플링하여 모델을 계산하고, 나머지 데이터에 의해 그 모델을 평가하는 방법으로 충분한 횟수를 반복하여 가장 지지도가 높은 모델을 구하는 방법이다[8]. RANSAC 알고리즘의 전처리 단계에서 실시되는 샘플링은 임의의 특징점 들을 랜덤 샘플링을 통해서 선택하게 된다. 선택된 특징점 중에서는 특징점 들의 분포가 선형 혹은 특정역역에 모이는 군집을 이루는 경우에는 잘못된 호모그래피를 추정하게 되어 알고리즘의 성능이 저하가 될 수 있다. 이와는 반대로 선택된 특징점들이 넓게 분포된 경우에는 정확한 호모그래피를 추정할 수 있다. 이 문제를 해결하기 위해서 제약 조건 만족 문제를 적용한 CS-RANSAC 알고리즘이 제안되었다.

CS-RANSAC 알고리즘은 이미지를 NxN의 단위격자로 나눈 후 선택된 특징점이 포함된 단위격자와 같은 열과 행에 위치한 단위격자들을 배제하고 나머지 특징점들을 사이에서 호모그래피를 구하기 위한 특징점들을 선택하도록 하여 선형과

군집 문제를 해결하였다. 이를 위해 특징점 분포 패턴에 따라서 분류된 이미지 그룹에 제약 조건 중 하나인 단위격자를 동일한 크기(17×17)를 적용시켜 호모그래피를 추정한다. 그러나 CS-RANSAC 알고리즘은 특징점 분포 패턴에 의해 분류된 모든 이미지 그룹에 동일한 단위격자 크기(17×17)를 적용하여 특정 그룹에서는 성능이 저하되는 문제가 발생한다[13].

### 2.3 DCS-RANSAC

위의 성능 저하를 방지하기 위해 입력된 이미지를 특징점 분포 패턴에 따라 분류하고 각 그룹에 알맞은 단위격자 크기를 동적으로 적용한 알고리즘이 DCS-RANSAC이다.

DCS-RANSAC에서는 이미지들을 특징점 분포 패턴에 따라 다섯 가지 그룹으로 분류하고 입력된 이미지를 이미 분류된 이미지 그룹의 특징점 분포패턴을 사용하여 분류하고 각각의 다섯 가지 이미지 그룹에 알맞은 단위격자 크기를 적용하여 랜덤 샘플링에서의 탐색 시간을 줄여 알고리즘의 성능을 높였다.

그러나 알고리즘에서 사용된 이미지 그룹들은 특징점 분포 패턴에 따라 수동적인 방법을 통해 직관적으로 분류되어 특정한 이미지들의 특징점 분포 패턴만을 반영하여 이미 분류된 이미지 그룹에 속하지 않은 이미지를 사용해서 알고리즘을 실행 할 경우에는 성능이 저하되는 경우가 발생한다.

### 2.4 T-RANSAC

T-RANSAC (Triangle-RANSAC)은 RANSAC 알고리즘의 샘플링 과정에서 특징점들의 필터링을 수행할 때 삼각형의 넓이를 이용한다. RANSAC 알고리즘의 고질적인 문제인 특징점들의 선형, 군집 문제를 해결하기 위해 특징점 3개를 선택하여 삼각형을 형성하고 경우의 수를 구한다. 만약 삼각형의 넓이의 합이 일정 임계값을 넘게 될 경우에는 샘플로 선택되고 넘지 않을 경우에는 필터링 하는 방법이다[9].

Fig. 1은 T-RANSAC의 샘플링 과정을 나타낸다. Fig. 1과 같이 네 개의 선택된 특징점 f1, f2, f3, f4 중 3개를 조합하여 Fig. 1의 a, b, c, d와 같이 4개의 삼각형을 만들고 Equation (1)을 사용하여 삼각형의 넓이를 구한 다음에 값이 특정 임계값을 넘을 경우에는 최종 샘플로 선택하게 된다.

$$S(a) + S(b) + S(c) + S(d) > Threshold[S(x) : \text{area of } \Delta] \quad (1)$$

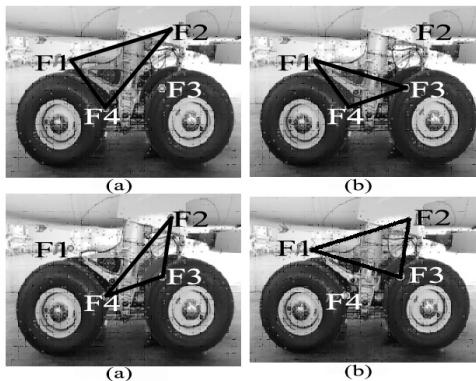


Fig. 1. T-RANSAC Sampling Process

## 3. K-means 클러스터링을 적용한 향상된 CS-RANSAC 알고리즘

### 3.1 KCS-RANSAC 알고리즘의 구조도

본 장에서는 전처리 단계에서 머신러닝 기법인 K-means 알고리즘을 사용하여 입력된 이미지를 특징점 분포 패턴에 따라 자동으로 분류하는 방법을 적용한 K-means clustering CS-RANSAC (이하 KCS-RANSAC) 알고리즘을 제안한다. Fig. 2에서 기존의 DCS-RANSAC 알고리즘에 제안하는 방법이 추가된 KCS-RANSAC의 구조도이다. Fig. 2의 과정 ①과 같이 KCS-RANSAC 알고리즘은 이미지를 K-means 클러스터링 알고리즘과 SURF 특징점들을 사용하여 이미지 그룹을 특징점 분포 패턴에 따라서 자동으로 구분한 후 호모그래피를 계산한다.

### 3.2 전처리 과정

KCS-RANSAC의 전처리 과정은 먼저 이미지에서 SURF 특징점을 추출한다. 추출된 특징점을 K-means clustering 알고리즘을 사용하여 특징점 분포 패턴에 따라 클러스터 개수만큼의 이미지 그룹을 생성한 후 클러스터를 이용하여 이미지들을 각각의 이미지 그룹으로 분류한다. 분류된 이미지들을 사용하여 10~30까지의 단위격자 별로 수행시간, 호모그래피 오차율, 그리고 호모그래피의 참정보 포함 비율 값을 구한다. 수행시간과 오차율은 최소가 되는 값을, 참정보 포함 비율은 최대가 되는 값을 갖는 단위격자를 구한 후 수행시간이 최소인 단위격자를 우선시하여 각 그룹별 최적의 단위격자 크기를 구한다. 여기서 구한 단위격자 크기를 각 이미지 그룹에 적용하여 호모그래피를 추정한다.

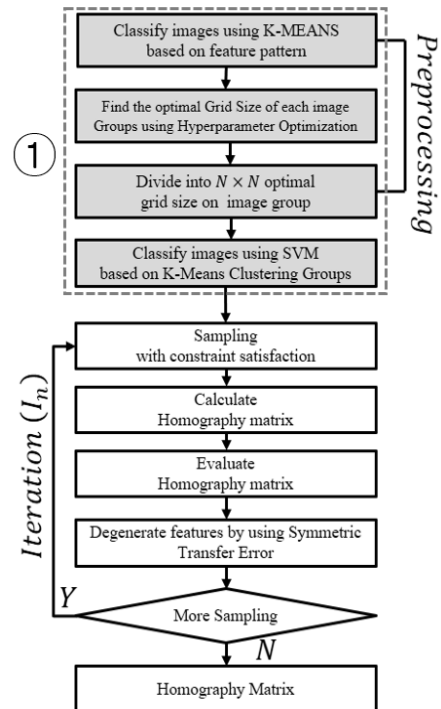


Fig. 2. KCS-RANSAC Sequence for Estimating Homography

### 4. 실험 및 분석

#### 4.1 실험 평가 방법

제안하는 알고리즘의 성능 평가를 위해 UKBench Dataset [10]을 사용한다.

첫 번째 실험은 SURF를 사용해 추출한 특징점을 가지고 K-means 알고리즘을 수행할 때 최적의 클러스터링 값을 구하는 실험이다. 각 이미지 그룹은 특징점 분포 패턴에 따라 클러스터링 수만큼 분류된다. 실험은 클러스터의 개수를 5~15까지 설정하고 K-means 알고리즘을 수행한 후 결과 값에 따라 최적의 클러스터링 값을 구한다.

두 번째 실험은 첫 번째 실험에서 구한 10개의 이미지 그룹에 대하여 우선 KCS-RANSAC의 각 이미지 그룹별로 최적의 단위격자 크기를 구하는 실험을 하였다. 최적의 단위격자 크기를 구하는 방법은 각 이미지 그룹마다 알고리즘의 수행시간, 추정된 호모그래피의 오차율, 추정된 호모그래피의 참정보 포함비율을 구하고 수행시간과 호모그래피 오차율을 최소화하고 호모그래피 참정보 포함 비율 값이 최대가 되는 단위격자 크기를 구한다.

세 번째 실험은 DCS-RANSAC 알고리즘과 KCS-RANSAC의 성능을 비교한다. 비교방법은 알고리즘의 수행시간, 호모그래피 오차율, 호모그래피 참정보 비율을 비교한다.

마지막 실험으로는 제안하는 알고리즘과 RANSAC, CS-RANSAC, T-RANSAC의 성능을 비교하였다. CS-RANSAC의 단위격자 크기는 최적의 결과 값을 갖는 17×17로 하였다[13]. 비교 기준은 위의 두 실험과 마찬가지로 수행시간, 오차율, 참정보 비율을 비교한다.

#### 4.2 K개의 클러스터링 개수

첫 번째 실험은 이미지 그룹을 분류할 때 사용할 클러스터의 수 K 값을 구하기 위한 실험이다.

실험 결과 K의 클러스터의 개수가 Table 1과 같이 10개일 때의 성능이 우수함을 보였다. 비록 K의 클러스터의 개수가 8개일 때의 오차율이 10개일 때 보다 낮게 나오지만 나머지 수행시간과 참 정보 포함 비율을 비교하면 클러스터의 개수가 10개일 때의 성능이 뛰어났다.

Fig. 3은 특징점 분포 패턴에 의해 구분된 10개의 그룹이다

#### 4.3 이미지 그룹별 단위격자 수

두 번째 실험은 10개의 이미지 그룹별로 적용할 단위격자의 크기를 결정하기 위한 실험이다. 실험은 본 논문에서 제안하는 KCS-RANSAC 알고리즘을 UKBench Dataset 20쌍에 대해 단위격자의 수(N×N)를 10~30까지 변화시키며 100번 반복 수행하

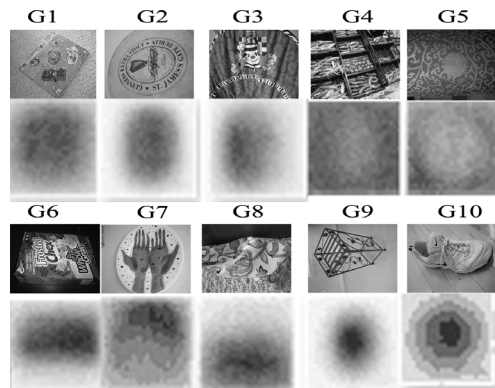


Fig. 3. 10 Image Groups Based on the Feature Distribution

여 평가항목들의 값을 구해 각각 평균을 계산하여 평가하였다.

평가하고자 하는 항목은 KCS-RANSAC 알고리즘의 수행시간과 추정된 호모그래피의 오차율, 그리고 호모그래피의 참정보 포함 비율이다.

실험결과는 Table 2와 같으며 각 이미지 그룹에 최적화된 단위격자크기는 세 가지 평가 항목을 모두 고려하여 구하였다.

Table 2. Optimal Grid Size for Each Image Groups

Group	Optimal Grid Size (<=30)
G1	10
G2	20
G3	10
G4	15
G5	10
G6	22
G7	24
G8	21
G9	30
G10	16

#### 4.4 DCS-RANSAC vs KCS-RANSAC

이번 실험은 제안하는 알고리즘과 DCS-RANSAC 알고리즘의 성능을 동일한 환경에서 각 이미지 그룹별로 2쌍씩 총 20쌍의 이미지를 가지고 알고리즘을 100회씩 수행하여 얻은 결과 값을 평균내서 비교하였다.

Table 3에서 볼 수 있듯이 제안한 방법은 DCS-RANSAC 알고리즘보다 수행속도가 평균 15%, 오차율은 35% 향상하였고, 참정보 포함 비율은 13% 증가한 것으로 확인되었다. 이는 KCS-RANSAC에 머신러닝 기법을 사용하여 특징점 분포 패턴에 따라 이미지를 자동으로 분류하여 다양한 특징점 분포 그룹들의 호모그래피를 추정 할 수 있기 때문으로 볼 수 있다.

Table 1. Optimized Number of Cluster

	5	6	7	8	9	10	11	12	13	14	15
PT	526.6	401.1	253.7	108	197.9	106.9	418.7	195.4	308.5	252.2	1903
ER	218.1	34.29	34.33	28.18	70.32	28.84	69.56	100.5	79.92	53.27	72.18
IR	0.647	0.675	0.656	0.715	0.638	0.728	0.664	0.641	0.616	0.716	0.56

PT : Processing Time (ms), ER: Error Rate (pixel), IR: Inlier Rate (%)

Table 3. DCS-RANSAC vs KCS-RANSAC

	DCS-RS	KCS-RS
PT	64.548	54.8544
ER	31.124	20.2352
IR	66.22	75.0866

PT: processing Time (ms), ER: Error Rate (pixel), IR: Inlier Rate (%)

#### 4.5 제안하는 알고리즘의 성능 비교 및 평가

본 성능비교는 10개의 이미지 그룹에 대하여 RANSAC, CS-RANSAC, T-RANSAC, KCS-RANSAC 네 가지 알고리즘을 동일한 환경에서 각 그룹마다 2쌍의 이미지를 적용하여 알고리즘을 100번 반복수행하여 얻은 값들의 평균값을 비교하였다. 실험결과 KCS-RANSAC의 수행시간, 오차율, 참 정보 포함 비율이 다른 알고리즘보다 우수한 것을 볼 수 있다. Table 4에서 G4의 경우 CS-RANSAC이 KCS-RANSAC 보다 1.93ms 낮은 수행시간을 보이는데 그 이유는 두 번째 실험의 각 그룹별 최적의 단위격자를 구하는 과정에서 수행시간뿐만 아니라 오차율, 참 정보 포함비율 세 가지 값을 고려하여 단위격자크기를 구했기 때문에 G4에서 수행시간이 CS-RANSAC보다 느리지만 나머지 항목인 오차율, 참 정보 포함 비율이 CS-RANSAC보다 우수한 성능을 보여주고 있다.

Table 5에서 일부 그룹의 경우에도 CS-RANSAC의 실험 결과가 KCS-RANSAC보다 좋은 결과를 보이는데 이는 위에서 기술 한 것처럼 KCS-RANSAC의 오차율은 CS-RANSAC보다 높은 결과를 보이지만 나머지 수행시간과 참 정보 포함 비율이 CS-RANSAC보다 높은 것을 알 수 있다.

마찬가지로 Table 6에서 G3의 CS-RANSAC의 참 정보 포함 비율이 KCS-RANSAC보다 0.05% 높은 참 정보 포함 비율을 보이는데 이는 수행시간과 오차율과 함께 비교를 해보면 KCS-RANSAC의 수행시간과 오차율이 CS-RANSAC보다 더 우수한 것을 알 수 있다.

KCS-RANSAC의 수행시간은 Table 4~6과 같이 다른 알

고리즘의 평균에 비해 약 30%에서 36.7%까지 평균 34%의 향상을 보였고 오차율은 약 16.1%에서 19.4%까지 평균 17.6%의 향상을 보였고 참정보 비율은 약 0.2%에서 약 1.6%까지 평균 1%의 성능 향상을 보였다. 각 이미지 그룹마다 최적의 단위격자 크기를 적용하였기 때문에 성능을 향상 시킬 수 있었다.

## 5. 결론 및 향후 연구

본 논문에서는 입력된 이미지를 특징점 분포 패턴에 따라 K-means clustering 알고리즘을 사용하여 그룹별로 구별하고 이미지 그룹마다 최적화된 단위격자 크기를 적용시킨 KCS-RANSAC 알고리즘을 제안하였다. KCS-RANSAC의 성능 평가를 위하여 K-means 클러스터링 알고리즘을 사용하여 입력된 이미지를 특징점 분포 패턴에 따라 분류하고 호모그래피 오차율, 수행시간, 참정보 비율을 측정하였다. 이미지 그룹 10개에 대해 동적 제약 조건인 단위격자 크기를 적용했기 때문에 특징점 분포에 강건한 결과를 보였다. 또한 랜덤 샘플링 과정에서 불필요한 탐색 연산을 줄여 알고리즘의 효율성을 향상시켰다. 수행시간 측면에서는 평균 34%의 성능 향상을 보였고 오차율은 17.6%, 참정보 비율은 1%의 성능향상을 보였다.

본 논문의 알고리즘은 증강현실 기술 중에서 비마커 기반의 객체 추적에 사용되는 알고리즘이지만 빠르게 진행되는 영상보다는 정적인 영상에 적합한 알고리즘이다. 특히 항공기 정비와 같이 대부분 정적인 영상에서 HMD (head Mounted Display)를 가지고 작업하는 환경에서는 정확하고 빠른 추적을 통해 가상의 객체를 HMD에 증강시킬 수 있으므로 항공기 정비를 보다 효율적으로 할 수 있게 된다.

본 논문에서는 이미지에서 랜덤 샘플링 과정을 진행 할 때 특징점 분포가 희박한 부분의 특징점이 선택되어 불필요한 연산이 증가하여 오버헤드가 발생하는 문제점이 있다. 따라서 향후 연구에서는 샘플링 단계에서 특징점 분포가 적은 부분의 특징점들을 필터링하여 선택이 되지 않게 하는 연구가 필요하다.

Table 4. Processing Time (ms)

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
RANSAC	81.04	88.69	57.22	127.3	121.6	44.22	38.83	107.1	128.8	75.45
CS-RANSAC	78.4	127.9	46.22	54.02	122.9	35.57	38.09	125.2	120.4	38.49
T-RANSAC	81.86	87.66	59.19	118.9	116.8	46.37	39.92	108.3	126.5	67.5
KCS-RANSAC	47.19	69.1	44.23	55.95	109.3	35.01	34.37	55.5	63.55	36.97

Table 5. Error Rate (pixel)

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
RANSAC	8.349	25.99	7.224	17.53	21.52	17.79	99.89	12.44	7.341	22.16
CS-RANSAC	7.029	19.12	7.678	17.29	23.31	16.42	106.7	7.733	4.816	20.76
T-RANSAC	7.957	24.11	7.546	17.37	20.36	17.16	99.55	11.6	6.822	21.8
KCS-RANSAC	6.704	19.84	6.971	17.07	17.84	16.84	78.68	11.19	4.823	13.77

Table 6. Inlier Rate (%)

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
RANSAC	76.94	71.86	89.83	63.4	45.96	89.03	64.1	85.95	75.11	76.88
CS-RANSAC	77	71.93	90.3	64.38	46.53	89.2	68.67	86.77	77.7	76.88
T-RANSAC	76.96	71.85	89.86	63.47	46.25	89.04	64.66	86.14	75.59	76.9
KCS-RANSAC	77.02	72	90.25	64.55	46.76	89.26	69.31	86.84	77.86	76.9

References

[1] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and Virtual Environments*, Vol.6, Issue 4, pp.355-385, 1997.

[2] F. Zhou, H. B. Duh, and M. Billinghurst, "Trends Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR," *IEEE International Symposium on Mixed and Augmented Reality*, pp.193-202, 2008.

[3] F. D. Crescenzo, M. Fantini, F. Persiani, L. D. Stefano, P. Azzari, and S. Salti, "Augmented Reality for Aircraft Maintenance Training and Operations Support," *IEEE Computer Graphics and Applications*, Vol.31, Issue 1, pp.96-101, 2011.

[4] K. S. Lee, A. N. Rosli and G. S. Jo, "A Method for Automatically Creating an Interactive Semantic Video based on AR System," *IEEE International Conference on Systems, Man, and Cybernetics*, in press, 2012.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, L. V., "SURF: Speeded Up Robust Features," *Computer Vision and Image Understanding*, Vol.110, No.3, pp.346-359, 2008.

[6] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol.2, No.3, pp.485-508, 1988.

[7] D. Stricker, "Tracking with Reference Images: A Real-Time and Markerless Tracking Solution for Out-Door Augmented Reality Applications," *International Symposium on Virtual Reality, Archeology, and Cultural Heritage*, ACM, pp.77-82, 2001.

[8] E. Vincent and R. Laganier, "Detecting planar homographies in an image pair," *IEEE International Symposium on Image and Signal Processing and Analysis*, pp.182-187, 2001.

[9] RANSAC [Internet], <http://en.wikipedia.org/wiki/RANSAC>, 2016.

[10] UKbench dataset (Center for Visualization&Virtual Environments) 2016 [Internet], <http://vis.uky.edu/~stewe/ukbench/>.

[11] B. Subbiah and S. Christopher, "Image Classification through Integrated K-means Algorithm," *International Journal of Computer Science Issues*, Vol.9, No.2, pp.518-524, 2012.

[12] D. Chandra, "Automatic Determination of Constraint Parameter for Improving Homography Matrix Calculation in RANSAC Algorithm," *Korea Information Processing Society*, Vol.21, No.1, 2014.

[13] G. S. Jo, K. S. Lee, C. Devy, C. H. Jang, and M. H. Ga, "RANSAC versus CS-RANSAC," *American Association for Artificial Intelligence (AAAI)*, pp.1350-1356, 2015.



고 승 현

e-mail : kosehy@gmail.com  
 2011년 Texas A&M Univ. Computer Engineering with Computer Science Track(학사)  
 2017년 인하대학교 컴퓨터정보공학과 (석사)

2017년~현 재 인하대학교 컴퓨터공학과 박사과정  
 관심분야 : Computer Vision, Deep Learning, Machine Learning, A.I



윤 의 녕

e-mail : entymos@hotmail.com  
 2013년 인하대학교 컴퓨터정보공학과(학사)  
 2015년 인하대학교 컴퓨터정보공학과(석사)  
 2015년~현 재 인하대학교 컴퓨터공학과 박사과정

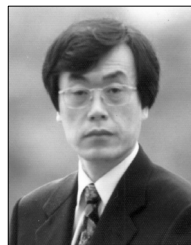
관심분야 : Interactive Video, Augmented Reality, Deep Learning, Semantic Web



Jumabek Alikhanov

e-mail : jumabek4044@gmail.com  
 2014년 Tashkent University of Information Technologies, Software Engineering(학사)  
 2017년 인하대학교 컴퓨터정보공학과 석사

관심분야 : Machine Learning, Deep Learning, Computer Vision, Image Processing, A.I, Data Mining



조 근 식

e-mail : gsjo@inha.ac.kr  
 1982년 인하대학교 전자계산학(학사)  
 1985년 Queens College, CUNY(M.A.)  
 1991년 City University of New York (Ph.D.)

1991년~현 재 인하대학교 컴퓨터정보공학과 교수

관심분야 : AI, Semantic Web, Intelligence Agent System