

# 모바일 기반의 3차원 객체 검색과 자세 추정을 위한 외형 기반의 인덱스 구축 및 검색 기법\*

탁윤식, 황인준  
고려대학교 전자전기전파공학과  
e-mail : {life993, ehwang04}@korea.ac.kr

## A Shape-based 3D object retrieval and pose estimation scheme for the mobile environment

Yoon-Sik Tak, Eenjun Hwang  
School of Electrical Engineering, Korea University

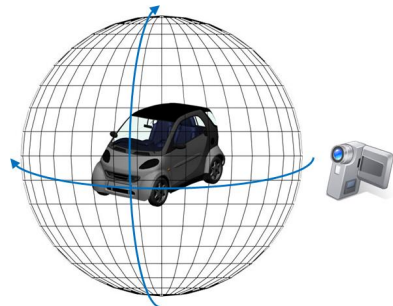
### 요 약

3차원 객체 검색 및 자세 추정 기법은 의료, 보안 등의 다양한 산업 영역에서 매우 중요한 이슈 중 하나로써 연구되고 있다. 정확한 객체 검색 및 자세 추정을 위해서는 객체의 가능한 모든 영상 정보를 사용하여야 하기 때문에 많은 연산시간이 걸리게 되고, 특히 객체의 정확한 자세를 추정하기 위해서는 높은 CPU의 성능과 큰 메모리 공간을 필요로 한다. 이러한 제약으로 인해, 3차원 객체 검색 및 자세 추정은 상대적으로 하드웨어의 성능이 낮은 모바일 장치에서 실행되기 어려웠다. 따라서, 본 논문에서는 모바일 장치에서도 효과적으로 객체 검색 및 자세 추정이 가능하도록 하기 위한 클라이언트-서버 환경에서의 객체의 외형 기반 인덱스 구축 및 검색 기법을 제안한다. 제안된 기법의 주요 특징은 i) 모바일 장치의 하드웨어 환경을 고려하여 비교적 적은 수의 객체의 영상을 바탕으로 한 객체 검색 및 후보 자세 예측과 ii) 모바일 장치에서의 검색 결과와 많은 수의 객체 영상을 기반으로 한 서버에서의 정확한 자세 추정이다. 실험 결과에서는 제안된 기법들을 통해, 빠른 시간 내에 정확한 객체 검색 및 자세 추정이 가능함을 보였다.

### 1. 서론 및 관련연구

임의의 각도에서 촬영한 객체의 단일 영상을 통해서 정확하게 객체를 인식하고 자세를 추정하는 능력은 로봇 비전, 보안 및 3차원 객체 검색 등의 컴퓨터 비전 영역에서 매우 중요한 요소이기 때문에 컴퓨터에 이러한 능력을 부여하기 위한 연구가 활발히 진행되고 있다. 이렇게 다양한 각도의 영상을 사용한 3차원 객체 검색 및 자세 추정을 위한 다양한 기법들이 제안되었다. 이러한 기법들의 예로써, [1]에서는 실시간 객체 검색 및 자세 추정을 위한 probabilistic boosting network 기법을 제안하였고, [2]에서는 최소 제곱 회기 분석 방법을 통한 객체 자세 추정 기법을 제안하였다. [3]에서는 color cooccurrence histogram 기반의 객체 자세 추정 기법을 제안하였다.

일반적으로 다양한 각도에서 촬영한 객체의 영상으로부터 얻을 수 있는 객체의 정보는 매우 다르기 때문에 다양한 각도에서 촬영한 객체의 정보로부터 임의의 한 각도에서의 객체의 형상을 추론하기는 기술적으로 매우 어렵다. 따라서, 임의의 각도에서 촬영된 영상만으로 정확한 3차원 객체를 인식하고 자세를 추정하기 위해서는 그림 1과 같이 해당 객체의 모든 가능한 각도에서 촬영한 영상 정보를 데이터베이스에

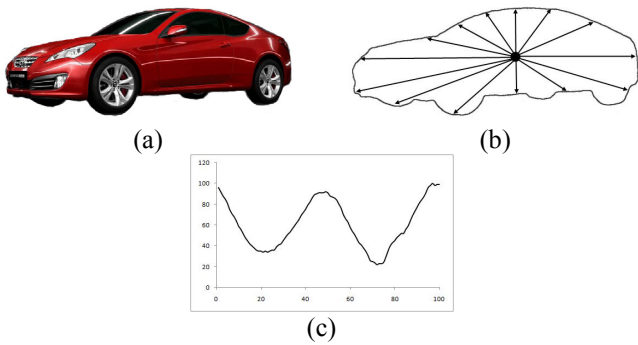


(그림 1) 객체의 다양한 시점에서의 영상 추출

포함하고 있어야 한다. 이렇게 많은 양의 영상 정보를 사용한 검색을 위해서는 높은 CPU 성능과 많은 메모리 공간 및 저장공간을 필요로 하기 때문에, 성능이 뛰어난 데스크탑 및 서버 등의 환경에 적합하였다. 하지만, 단일 영상을 통한 객체 인식 및 자세 추정 기술은 이동형 로봇과 현장에서 바로 사용하기 위한 모바일 장치들과 같이 비교적 낮은 사양의 환경에서 사용되어야 하는 경우가 많다. 따라서, 검색의 정확도를 유지하면서도 이러한 저 사양의 모바일 환경에 적용할 수 있는 기법에 대한 연구가 필요하다.

본 논문에서는 이러한 목적을 위해, 클라이언트-서

\* 이 논문은 2008년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임 (KRF-2008-313-D00858)



(그림 2) 거리 곡선 추출 과정 (a) 원본 이미지 (b) 외곽선 검출 및 거리 계산 (c) 거리 곡선

버 기반의 새로운 외형 기반의 3 차원 객체 검색 및 자세 추정 기법을 제안한다. 모바일 장치는 상대적으로 하드웨어의 성능이 낮기 때문에 객체의 모든 가능한 영상을 고려한 검색이 어렵다. 따라서, 객체로부터 영상을 추출하는 구간을 크게 하여 적은 수의 영상만을 사용하여 검색에 사용함으로써 모바일 장치에 적합한 검색이 이루어지도록 한다. 하지만, 영상을 추출하는 구간을 크게 할 경우, 상당수의 객체 자세 변화에 따른 영상 정보가 없기 때문에 정확한 객체의 자세를 추정하기 어렵다. 이러한 정확도를 보정하기 위해, 상대적으로 하드웨어의 성능이 좋은 서버에 객체의 영상 정보 추출 구간을 작게 하여 객체 자세 변화에 따른 많은 영상 정보를 저장하고, 모바일 장치에서 검색된 결과에 포함된 객체의 자세를 보다 정확하게 추정할 수 있도록 한다.

본 논문의 구성은 다음과 같다. 2 장에서는 영상들로부터 인덱스 구성 및 검색에 사용될 특징정보로써 거리 곡선을 추출하는 과정에 대해서 설명한다. 3 장에서는 모바일 장치에서의 3 차원 객체 검색 및 해당 객체의 후보 자세 검색 기법을 소개하고 4 장에서는 모바일 장치의 검색 결과를 사용한 서버에서의 객체 자세 추정 기법을 설명한다. 5 장에서는 다양한 실험 결과를 보이며, 6 장에서는 본 논문을 결론짓는다.

## 2. 외형 정보 추출

본 논문에서는 질의 영상과 3 차원 객체들로부터 추출한 영상들간의 유사한 정도를 계산하기 위한 특징정보로써 객체의 중심점과 외곽선과의 거리를 계산한 거리 곡선을 사용한다. 이러한 거리 곡선을 데이터베이스에 저장된 3 차원객체들의 영상들 및 질의 영상으로부터 추출하기 위한 과정은 그림 2 에 잘 나타나 있으며, 주요 과정은 다음과 같다.

- 1) 먼저 canny edge detection 알고리즘을 통해 영상으로부터 객체의 외곽선을 검출한다.
- 2) 검출된 외곽선을 기반으로 객체(외곽선 내부)의 중심점을 계산한다. 이러한 중심점은 외곽선 내부 픽셀들의 위치의 평균 값을 사용한다.
- 3) 계산된 중심점과 외곽선과의 거리를 계산하고 외곽선을 따라 이동하면서 계산된 값들을 연결하여

거리 곡선을 구한다.

- 4) 마지막으로 객체의 크기 보정을 위해 영상들로부터 계산된 모든 거리 곡선들을 같은 최대 값을 가지도록 표준화 시킨다.

## 3. 유사 3 차원 객체 및 후보 자세 검색 기법

본 장에서는 모바일 장치에서의 단일 영상 질의를 통한 유사 객체 검색 및 후보 자세 검색 기법에 대해서 설명한다. 모바일 장치는 CPU, 메모리 공간 및 저장 공간 등의 한계로 인해, 많은 양의 카메라 관점을 변화시킨 영상을 통한 유사 3 차원 객체의 검색 및 자세 추정이 어렵다. 따라서, 모바일 장치에서는 영상을 추출하기 위한 카메라 관점의 변화폭을 크게 함으로써 비교적 적은 양의 영상만을 사용하여 객체 검색 및 후보 자세를 추정하도록 한다. 이렇게 카메라 관점을 변화시키며 객체의 영상을 추출하는 과정에서, 객체의 대칭성에 따라서 반복되는 외형의 패턴이 발생된다. 예로써 ‘구’의 경우 모든 카메라의 관점에서 동일한 외형을 가지며, ‘정육면체’의 경우 45 도 간격으로 반복되는 외형 패턴을 가진다. 이런 패턴 유형에 따라 카메라의 수평 이동과 수직이동에 따라 각각 4 개의 외형 반복 패턴을 찾을 수 있다. 이러한 반복되는 영상 패턴을 제거함으로써 검색의 정확도에 영향을 주지 않으면서도 검색 속도를 월등하게 빠르게 할 수 있다. 이러한 외형 반복 패턴에 따른 객체의 분류 방법에 대해서는 [4]에서 설명하고 있다.

이렇게 중복된 영상 패턴을 제거하고 남은 영상들을 효과적으로 질의 영상과 비교하기 위해서는 적합한 인덱스 구조를 사용하여야 한다. 본 논문에서는 객체들로부터 추출한 영상들로부터 계산된 거리 곡선들을 푸리에 변환을 통해 계산된 계수들을 사용하여 R-Tree 기반의 인덱스 구조를 생성한다. 사용되는 인덱스 구조의 생성 과정은 다음과 같다.

- 1) 각각의 객체들의 영상들로부터 추출한 거리 곡선으로부터 계산된 푸리에 계수 값을 인덱스 공간에 나타낸다.
- 2) 각각의 개체 별로 인덱스 공간상에서 인접한 영상들을 푸리에 계수 값의 경계 값을 설정하며 순차적으로 그룹핑 한다.
- 3) 최종적으로 모든 개체의 영상들이 하나의 루트 노드에 포함되도록 2)에서 형성한 개체의 그룹들을 순차적으로 다시 그룹핑 한다.

이러한 3 차원 객체들로부터 추출한 영상들을 통해 인덱스를 구축하는 방법에 대한 자세한 설명은 [5]에 잘 나타나 있다. 이렇게 구성된 R-Tree 기반의 인덱스를 사용해서 효과적인 객체 검색 및 후보 자세 예측을 위해, 기존연구로써 [5]에서는 GEMINI Framework [6] 기반의 k-NN 알고리즘을 제안하였다. GEMINI Framework 는 거리 곡선과 유사한 타임시리즈 검색에서 ‘no false dismissal’을 가상적으로 보장하며, 아래 경계(low bound) 값을 가지는 함수들을 계층적으로 정의함으로써 순차적인 비교에 비해 빠른 검색이 가능하

**Algorithm 1.  $k$ -NN Search**


---

```

Variable queue : MinPriorityQueue;

1.  queue.push(root); Result = {};
2.  while not queue.IsEmpty() do
3.    top = queue.Pop();
4.    if top.id is in the Result
5.      continue;
6.    else
7.      if top is a sequence with DTW Dist.
8.        add top to Result;
9.        if |Result| =  $k$ 
10.         return Result;
11.       else if top is a leaf node
12.         for each data Fourier Point C in top
13.           queue.push(C, Fourier_Dist(Q,C));
14.       else if top is a Fourier point, C
15.         retrieve a full sequence C from DB;
16.         queue.push(C, LB_Keogh(Q,C));
17.       else if top is a sequence C with LB_Keogh Dist
18.         queue.push(C, DTW(Q, C));
19.       else // top is a non-leaf node
20.         for each child node C in top
21.           queue.push(C, MINDIST(Q,C));

```

---

다. 본 논문에서는 거리 곡선간의 차이 계산을 위해 동적 정합 기법을 사용하였으며, 다양한 아래 경계 함수들을 계층적으로 정의하였다. GEMINI Framework와 아래 경계 함수들에 대한 자세한 설명은 [7]에 잘 나타나 있으며, 제안된 검색 기법은 알고리즘  $k$ -NN Search에 나타내었다.

검색의 효율성을 위해 반복되는 외형 패턴을 제거하였기 때문에,  $k$ -NN Search를 통해 검색된 객체의 후보 자세의 외형과 동일한 외형을 가지는 동일 객체의 제거된 다른 자세들의 영상 정보를 추가적으로 추출하여 그 중 질의 영상과 가장 유사한 영상을 찾아야 한다. 외형정보만으로는 이렇게 동일 외형의 영상들과 질의 영상간의 유사도 비교가 불가능 하기 때문에, SIFT 알고리즘을 통해 판단하도록 한다.

**4. 3 차원 객체 자세 추정 기법**

3장에서 설명한대로, 사용자는 모바일 장치를 사용하여 질의 영상과 유사한 객체 및 해당 객체의 후보 자세를 추정할 수 있다. 하지만, 모바일 장치의 하드웨어 성능의 한계로 인해 적은 양의 카메라 영상만을 검색에 사용하였기 때문에, 질의 영상을 정해진 객체의 위치가 아닌 임의의 자세에서 촬영할 경우 정확한 객체의 자세 추정이 어렵다. 이러한 한계를 극복하기 위해서, 본 논문에서는 하드웨어 성능이 높은 서버에 모바일 장치의 데이터베이스에 저장된 3 차원 객체와 동일한 객체들을 카메라 관점의 변화폭을 작게 하여 객체들로부터 많은 양의 영상들의 인덱스를 생성하여 보다 더 정확한 객체의 자세를 추정할 수 있도록 한다. 제안된 모바일 장치와 서버와의 데이터 교환을 통한 정확한 객체 자세 추정 기법은 알고리즘 Pose Estimation에 나타나 있다.

**Algorithm 2. Pose Estimation (Mobile Result)**


---

```

Variable queue : MinPriorityQueue;
Server_Result = {};

1.  for each Object  $O_i$  in Mobile_Result
2.    root = Find_Object_Root( $O_i$ .id);
3.    queue.push(root);
4.    while not queue.IsEmpty() do
5.      top = queue.Pop();
6.      if top is a sequence with DTW Dist.
7.        if ( $O_i$ .dist  $\leq$  top.dist)
8.          add top to Server_Result; break;
9.      else if top is a leaf node
10.         for each data Fourier Point C in top
11.           if ( $\text{Fourier\_Dist}(Q,C) \leq \text{top.dist}$ )
12.             queue.push(C, Fourier_Dist(Q,C));
13.         else if top is a Fourier point, C
14.           retrieve a full sequence C from DB;
15.           if ( $\text{LB\_Keogh}(Q,C) \leq \text{top.dist}$ )
16.             queue.push(C, LB_Keogh(Q,C));
17.         else if top is a sequence C with LB_Keogh Dist
18.           if ( $\text{DTW}(Q,C) \leq \text{top.dist}$ )
19.             queue.push(C, DTW(Q, C));
20.         else // top is a non-leaf node
21.           for each child node C in top
22.             if ( $\text{MINDIST}(Q,C) \leq \text{top.dist}$ )
23.               queue.push(C, MINDIST(Q,C));
24.         Clear queue;

```

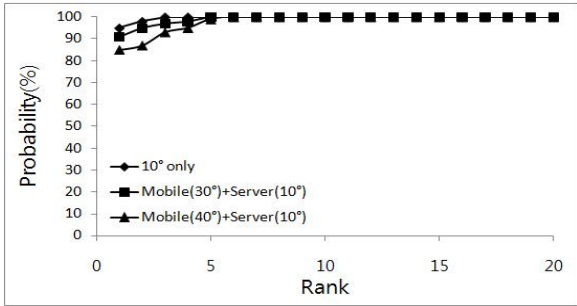
---

- Find\_Object\_Root( $O_i$ .id) returns the root of the index tree from the object whose object id is  $O_i$ .id.

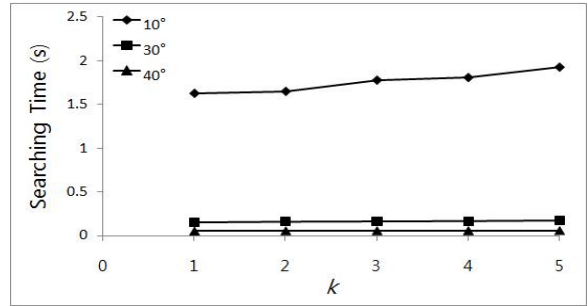
---

- 1) 모바일 장치는 자신의 데이터베이스를 사용한  $k$ -NN 알고리즘의 결과(Mobile\_Result)를 서버에 전송한다. 서버는 전송된 결과에 포함된 객체와 해당 객체의 질의 영상과의 차이 정보를 사용해서 빠른 검색 시간 내에 보다 더 정확한 객체의 자세를 추정한다.
- 2) 서버는 모바일 장치로부터 전송된 결과에 포함된 각각의 객체에 대해서 모바일 장치에서 예측한 자세보다 더 유사한 자세가 서버의 데이터베이스에 있는지 검색한다. 검색 과정은 다음과 같다.
  - 서버에 구축된 인덱스 구조에서 해당 객체의 영상을 모두 포함하는 노드를 검색하여 루트 노드로 설정한다. (Line 2)
  - 인덱스 구조에서 모바일 장치에서 예측한 자세와 질의 영상과의 차이보다 더 작은 차이를 가지는 노드들을 검색하며, 각각의 노드에 맞게 질의 영상과의 차이를 계산한다. (Line 4 - 23)
  - 검색된 자세들 중 최소 값을 가지는 자세를 찾아서 해당 객체의 자세로 추정한다. (Line 6 - 8)

서버는 Pose Estimation 알고리즘을 통해 객체들의 자세를 추정한 후, 3장에서와 마찬가지로 추정된 객체의 자세와 외형이 같은 동일 객체의 다른 자세들의 영상 정보를 추가적으로 추출하여 그 중 질의 영상과 가장 유사한 영상을 SIFT를 통해 판단한다. 최종적으로



(그림 3) 검색 정확도 비교



(그림 4) 검색 시간 비교

로 서버는 검색 결과를 모바일 장치에 전송함으로써 사용자는 모바일 장치에서 계산된 결과보다 정확한 결과를 얻을 수 있다.

## 5. 실험 결과

본 논문에서는 제안된 기법들의 성능을 평가하기 위한 다양한 실험을 실시하였다. 실험을 위해 3 차원 객체들로부터 카메라 관점을 이동시키며 촬영한 약 14,000 개의 이미지들을 사용하였으며, 사용된 시스템은 Intel Core2Duo CPU와 4 GB RAM으로 구성되었다. 본 논문에서 제안된 알고리즘들은 C#을 사용해 구현하였다. 모바일 장치에서는 30° 간격으로 촬영된 카메라 영상을 사용하였으며, 서버에서는 보다 정확한 객체의 자세 추정을 위해 10° 간격으로 촬영된 영상을 사용하였다. 질의 영상은 데이터베이스에 저장된 3 차원 객체와 동일한 실제 물체로부터 임의의 각도에서 촬영된 영상을 사용하였다.

먼저 제안된 기법의 정확성에 대한 실험을 수행하였다. 제안된 기법과의 비교를 위해 i) 10° 간격으로 촬영된 영상 ii) 30° 간격으로 촬영된 영상(모바일 장치)과 10° 간격으로 촬영된 영상(서버)의 조합, 그리고 iii) 40° 간격으로 촬영된 영상(모바일 장치)과 10° 간격으로 촬영된 영상(서버)의 조합에 따른 검색 정확도를 비교하였다. 그림 3에서는 cumulative match curves (CMC)를 사용해서 비교된 기법들의 정확도를 나타내었다. 이 그래프를 통해서 모바일 장치에 30°와 40° 등의 비교적 넓은 카메라 관점 이동 간격을 가지도록 영상을 촬영하여도 서버에 10° 등의 좁은 간격의 영상 촬영 간격을 가진 영상을 저장할 경우, 10° 간격의 영상 촬영 간격을 가진 경우와 유사한 결과를 얻을 수 있음을 알 수 있다.

다음 실험으로, 영상 추출 간격의 변화에 따른 추출된 영상 수의 변화에 따른 검색 시간을 비교하였다. 비교를 위해, 10°, 30° 그리고 40°의 영상 추출 간격의 변화를 주었다. 그림 4에서는 각각의 영상 추출 간격에 따라 k-NN 검색의 시간 변화를 그래프로써 나타내었다. 그래프를 통해서 영상 추출 간격을 30° ~ 40°로 할 경우, 10°로 할 경우보다 월등하게 검색 속도가 빠름을 알 수 있다. 또한 그림 3과 4를 통해, 30° ~ 40°의 영상 추출 간격에 따른 객체 영상을 추출할 경우, 모바일 장치에서 실행 가능할 정도의 검색 시간을 가질 수 있으며, 서버의 정확도 보정을 통

해 만족할만한 정확도를 가질 수 있음을 알 수 있다.

## 6. 결론

본 논문에서는 모바일 장치에서도 효과적으로 3D 객체 검색 및 자세 추정이 가능하도록 하기 위한 클라이언트-서버 환경에서의 객체 외형 기반 인덱스 구축 및 검색 기법을 제안하였다. 제안된 기법은 모바일 장치의 하드웨어 성능을 고려하여 적은 수의 영상을 기반으로 한 객체 검색 및 후보 자세 검색과 객체 자세 예측의 정확도를 높이기 위한 서버에서의 자세 추정 기법으로 구성된다. 실험을 통해서, 모바일 장치를 위해 영상 추출 간격을 30° ~ 40°와 같이 넓게 할 경우, 10° 등과 같이 많은 양의 영상을 기반으로 할 경우에 비해 월등히 빠른 검색시간을 가질 수 있음을 보였다. 또한, 서버에서의 객체 자세 보정을 통해 높은 정확도를 가질 수 있음을 보였다.

## 참고문헌

- [1] J. Zhang, S.K. Zhou, L. McMilan and D. Comanicu, "Joint Real-time Object Detection and Pose Estimation Using Probabilistic Boosting Network," International Conference on Computer Vision and Pattern Recognition, pp.1-8, 2007
- [2] M. Saito and K. Kitaguchi, "Appearance based object pose estimation using regression models," SICE Annual Conference, pp.1926-1929, 2008
- [3] S. Ekvall, F. Hoffmann and D. Kragic, "Object Recognition and Pose Estimation for Robotic Manipulation using Color Cooccurrence Histograms," International Conference on Intelligent Robots and Systems, pp.1284-1289, 2003
- [4] Y. Tak, K. Kim and E. Hwang, "Real-time 3-D Object Detection and pose estimation using Multi-view Images for the Mobile Environment," ACM Multimedia 2009, submitted
- [5] Y. Tak and E. Hwang, "An indexing scheme for efficient camera angle invariant image retrieval," International Conference on Computer and Information Technology, pp.143-148, 2008
- [6] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," ACM SIGMOD, pp. 419-429, 1994
- [7] E. Keogh and C. Ratanamahatana, "Exact indexing of dynamic time warping," Knowledge and Information Systems, Vol.7, pp. 358-386, 2005