

사진 촬영 분포를 기반으로 한 여행 경로 추천 시스템

임동균*, 박명진*, 문연수*, 장승호*, 국찬호*, 박재욱*, 이용규*
*동국대학교 컴퓨터공학과-서울
e-mail:okdk1201@dongguk.edu

A Travel Route Recommendation System Based on the Photograph Shooting Statistics

Dong Guen Lim*, Myung Jin Park*, Yeon Su Moon*, Seung Ho Jang*,
Chan Ho Kuk*, Jae Wook Park*, Yong Kyu Lee*
*Dept of Computer Science & Engineering, Dongguk University-Seoul

요 약

기존의 웹 지도 서비스는 방문 횟수가 많은 장소를 알기 어렵고, 사용자에게 여행 경로를 추천하는 기능 또한 찾기 어려웠다. 따라서 본 논문에서는 사진 촬영 분포를 기반으로 한 여행 경로 추천 시스템을 제안한다. 사진이 많이 촬영된 곳이 여행객이 많이 방문한 곳이며, 유명한 장소일 것이라고 가정하여 사진 촬영 분포를 기반으로 여행 경로를 추천하고자 한다. 여행 경로를 추천하기 위해 사진 데이터의 위치 값을 수집하고, 사진 데이터의 위치 값을 기반으로 사진 촬영 분포를 시각화하여 지도 위에 나타낸다. 또한, 여행 지역 내 사진이 많이 촬영된 장소를 유명한 장소로 선정하여 이를 경유하는 여행 경로를 추천한다. 사용자는 시스템을 통해 유명한 장소를 쉽게 인식할 수 있고, 편리하게 여행 경로를 계획할 수 있다.

1. 서론

구글 지도[1]와 다음 지도[2] 같은 기존 웹 지도 서비스는 사용자에게 POI(Point Of Interest) 검색과 출발지에서 도착지까지의 최적 경로를 제공하는 기능이 있다. 그러나 기존 웹 지도 서비스에서는 사용자가 경로에 경유지를 추가하고 싶은 경우, 출발지와 목적지 사이의 경유지를 선별해야 하고, 직접 경유지간 가장 적절한 경로를 구성해야 하는 단점이 있다.

본 논문은 사진 촬영 분포를 기반으로 경유지를 선정하고, 선정한 경유지를 모두 방문할 수 있는 최적 경로를 제공하는 시스템을 제안한다.

위치 기반 사진 서비스에 축적된 사진 데이터를 기반으로 사진이 많이 촬영된 장소가 유명한 장소라고 가정하고, 웹 지도 위에 사진 촬영 분포를 보여줌으로써, 사용자가 유명한 장소를 한눈에 인식할 수 있도록 도와준다. 그리고 사용자가 여행 경로 추천 기능을 사용하면 자동으로 선정된 복수의 경유지와 이를 모두 방문할 수 있는 여행 경로를 제공받을 수 있다. 궁극적으로 사용자가 여행 일정을 좀 더 쉽고 편리하게 계획할 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 관련 연구, 3절에서는 사진 촬영 분포를 활용한 여행 경로 추천 시스템의 구성과 동작 흐름, 4절에서는 설계 및 구현, 5절에서는 구현 결과 및 성능 분석에 대해서 설명하고, 마지막으

로 6절에서는 본 논문의 결론으로 끝을 맺는다.

2. 관련연구

2.1 정보 시각화

정보 시각화는 방대한 양의 데이터를 그래프, 차트, 다이어그램과 같은 도표로 나타내어 직관적이고 효율적으로 제공할 수 있으며, 데이터 안에서 의미 있는 정보, 경향 그리고 특징 등을 쉽게 파악할 수 있다[3].

본 논문에서는 정보 시각화를 위해 Google Maps JavaScript API v3[4]에서 제공하는 HeatmapLayer를 사용하였다. 이를 통하여 비정형적이고 매우 광범위한 사진의 위치정보를 한눈에 볼 수 있다.

2.2 최적 경로 제공

방문할 장소가 여럿일 경우 경유지를 고려한 최적 경로를 제공하기 위한 연구[5]가 진행되고 있고 관련 기능을 제공하는 웹 서비스로 구글 지도와 다음 지도 등이 있다. 하지만 사용자가 경유지를 일일이 선택해야 하는 불편함이 있으며, 자동으로 경유지를 선정하여 최적 경로를 제공하는 서비스는 찾기 어렵다.

본 논문에서는 경유지를 사용자가 직접 선택하지 않고, 위치 기반 사진 서비스의 사진 정보를 활용하여 사진 촬영 분포를 나타내고 사진 촬영이 많이 된 장소를 경유지로 선정한다는 점에서 기존의 관련 연구와 차별된다.

* 본 연구는 미래창조과학부 및 정보통신산업진흥원의 서울어코드 활성화 지원사업의 연구결과로 수행되었음 (NIPA-H1807-2014-1004)

3. 사진 촬영 분포 기반 여행 경로 추천 시스템

3.1 시스템 구성도

시스템 구성도는 그림 1과 같이 서버 컴퓨터와 3대의 수집 컴퓨터로 구성된다.

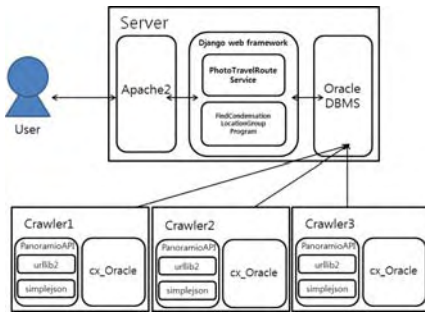


그림 1. 시스템 구성도

서버 컴퓨터에는 웹 서버 Apache2, RDBMS Oracle이 있고, Python 기반의 웹 어플리케이션 프레임워크 Django를 사용하여 본 시스템에 해당하는 웹 어플리케이션 PhotoTravelRouteService와 사진이 많이 촬영된 위치를 찾아주는 FindCondensationLocationProgram을 작성한다.

각 수집 컴퓨터에는 Python 언어로 작성한 수집기(crawler)가 있으며 수집기는 PanoramioAPI와 cx_Oracle 라이브러리를 사용하여 위치정보태그 기반의 사진 서비스인 파노라미오(Panoramio)[6]에 게시된 사진의 위치정보를 수집하고 저장한다.

3.2 동작 흐름

동작 흐름은 그림 2와 같으며, 사진의 위치정보는 수집 컴퓨터에 의해서 사진에 수집이 되었다고 가정한다.

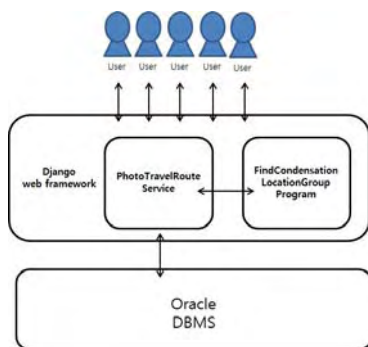


그림 2. 동작 흐름

- 1) 사용자가 웹 브라우저를 통해 시스템에 접근하면 시스템은 PhotoTravelRouteService의 사진 촬영 분포 및 경로를 표시하는 지도가 포함된 메인화면을 사용자에게 전송한다.
- 2) 사용자에게 메인 화면이 전송되면, 사용자가 보고 있는 지도의 좌측 하단 좌표와 우측 상단 좌표를 시스템에 전송한다. 시스템은 이를 PhotoTravelRouteService

에게 넘기고 PhotoTravelRouteService는 DBMS에 해당 좌표 범위 내 사진 촬영 위치를 질의한다.

- 3) PhotoTravelRouteService가 DBMS로부터 좌표 범위 내 사진 촬영 위치정보를 얻으면, 이를 FindCondensationLocationGroupProgram에게 넘겨준다. FindCondensationLocationGroupProgram는 경유지를 선정하며, 선정된 그룹 내 좌표들의 평균값으로 대표 좌표를 구하고 이를 PhotoTravelRouteService에게 넘겨준다.
- 4) PhotoTravelRouteService는 사진 분포를 표시할 좌표 리스트와 유명한 장소로 선정된 좌표 리스트를 사용자에게 전송한다.
- 5) 메인 화면에서는 사진 분포를 히트맵으로 표시하고 유명한 장소를 마커로 표시한다. 사용자가 여행 경로 추천 기능을 사용하면 출발지와 도착지 내 유명한 장소를 경유지로 포함한 최적의 경로를 생성한다.

4. 설계 및 구현

4.1 데이터베이스 설계

데이터베이스는 지리 정보를 저장하는 LOCATION과 사진 정보를 저장하는 PHOTO 개체(Entity)로 이루어져 있다.

LOCATION은 LocationName, Latitude, Longitude, Count 속성(Attribute)을 가지며, LocationName은 기본 키로 지역명을 저장하고, Latitude는 해당 지역의 위도 값, Longitude는 해당 지역의 경도 값, Count는 해당 지역에서 촬영된 사진의 수를 저장한다.

PHOTO은 PhotoID, Latitude, Longitude, PhotoURL, 속성(Attribute)을 가지며, PhotoID는 기본 키로 수집된 사진의 고유 번호를 저장하고, Latitude는 사진의 위도 값, Longitude 사진의 경도 값, PhotoURL은 사진의 URL를 저장하며, LOCATION의 LocationName을 외래키로 가진다. 그림 3은 LOCATION과 PHOTO의 논리적 모델을 나타낸 것이다.

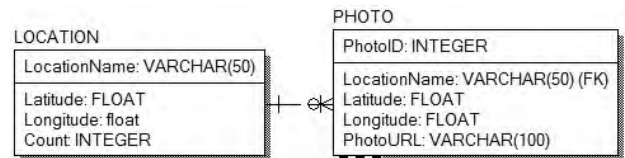


그림 3. LOCATION과 PHOTO의 논리적 모델

LOCATION은 여러 개의 PHOTO를 0개 이상 가질 수 있으므로 관계는 1:N이며, 비식별 관계이다.

4.2 수집기 구현

수집기는 Python 언어로 작성하였으며, 파노라미오[6]에 게시된 사진의 위치정보를 수집한다. 수집기는 PanoramioAPI 클래스를 객체로 생성하여 이를 사용하는

데, PanoramioAPI 클래스는 파노라미오의 웹 API를 호출하고 결과 값을 가져오는 역할을 한다. 다음 그림 4은 PanoramioAPI 클래스의 일부분이다.

그림 4. PanoramioAPI 소스코드

```
class PanoramioAPI(object):
    url = 'http://www.panoramio.com/map/get_panoramas.php'
    def search(self, set=None, limit_from=0, limit_to=20):
        self._params = { 'set': set, 'from': limit_from,
                          'to': limit_to, }
        request = urllib2.Request(url=self._get_url())
        response = urllib2.urlopen(request)
        return json.load(response)
```

PanoramioAPI 클래스에는 search 메소드가 정의되어 있다. search 메소드는 외부로부터 set, limit_from, limit_to의 값을 받아오고, 이 값들을 _params 인스턴스 변수에 저장한다. 파노라미오 API는 set이라는 변수에 0 이상의 정수 값인 파노라미오 사용자 식별자 번호인 user id 값을 넘겨서 해당 사용자가 올린 사진과 사진의 위치 정보와 같은 여러 정보를 획득할 수 있다.

메소드 _get_url을 호출하여 인스턴스 변수 url과 _params를 파노라미오 웹 API 주소 형식에 맞는 주소로 생성하고, urllib2 라이브러리의 Request 메소드와 urlopen 메소드를 사용하여 생성한 주소를 호출한 결과 값을 가져온다. 그리고 simplejson 라이브러리의 load 메소드를 사용하여 결과 값을 JSON 객체로 생성하고 이를 반환한다. 수집기는 반환된 JSON 객체를 파싱하여 user id에 해당하는 사용자가 올린 사진의 위치정보를 수집할 수 있다.

4.3 데이터 시각화

사진이 촬영된 분포를 지도 위에 보여주기 위해서 구글 지도[1]와 Google Maps JavaScript API v3[4]에서 제공하는 HeatmapLayer를 사용한다. 시스템의 메인 페이지에 해당하는 main.html에는 구글 지도와 HeatmapLayer를 생성하는 initialize라는 JavaScript 함수가 있다.

그림 5의 코드와 같이 수집한 사진의 위치정보 중 위도와 경도를 LatLng 객체로 생성하여 이를 배열이나 MVCArray[] 객체에 추가한다. 그리고 그림 6와 같이 HeatmapLayer 객체 생성 시 그림 5에서 생성한 데이터를 넘겨준다. HeatmapLayer 객체 생성 후, HeatmapLayer 객체의 setMap 메소드를 호출하여 인자 값으로 구글 지도 객체를 넘겨주면 그림 5에서 LatLng 객체로 생성한 사진 촬영 위치가 구글 지도 위에 분포로 표시가 된다.

그림 5. Heatmap 지리 데이터 입력

```
var photoData = {
    new google.maps.LatLng(37.782551, -122.445368),
    ... }
```

그림 6. HeatmapLayer 생성

```
function initialize() {
    var pointArray = new google.maps.MVCArray(photoData);
    heatmap = new google.maps.visualization.HeatmapLayer({
        data: pointArray});
    heatmap.setMap(map);
}
```

4.4 경유지 선정

사용자가 출발지와 도착지를 지도 위에 입력하면 시스템은 출발지와 도착지 사이에 사진이 많이 촬영된 지역을 기반으로 경유지를 자동으로 선정한다. 사진이 많이 촬영된 지역을 선정하는 방법은 위도, 경도 좌표 중 소수점 아래 넷째 점까지 동일한 좌표들을 찾아낸다. 해당 좌표의 개수들이 많을수록 해당 지역을 유명 여행지로 가정하고, 해당 좌표들의 평균값을 구하여 대표 좌표를 구한다. 이 방식으로 출발지와 도착지 사이에서 사진이 많이 촬영된 지역을 사용자가 원하는 만큼 찾아서 자동으로 경유지를 선정할 수 있다.

4.5 최적 경로 생성

최적화된 경로를 제공하기 위해서 Google Maps JavaScript API[4]를 사용한다.

그림 7는 본 논문에서 구현한 시스템의 소스 코드 일부분이다. 그림 7의 request 변수와 같이 origin 키의 값으로는 출발지를, destination 키의 값은 도착지, waypoints 키의 값으로 이전에 선정한 지역들을 담고 있는 배열, optimizeWaypoints 키의 값을 true로 한다. 그리고 DirectionService 객체가 route 메소드를 호출하면 waypoints에 저장했던 지역을 경유해서 가는 최적의 경로가 생성된다. 이 경로를 DirectionRenderer 객체의 setDirections 메소드의 인자 값으로 넘기고 setDirections 메소드를 호출하면 지도 위에 해당 경로가 출력된다.

그림 7. 여행 경로를 생성하는 소스코드

```
var directionsService = new google.maps.DirectionsService();
var request = {
    origin: start, destination: end, waypoints:
    waypts, optimizeWaypoints: true, travelMode:
    google.maps.TravelMode.DRIVING};
...
directionsService.route(request, function(response,
status) {
    if (status == google.maps.DirectionsStatus.OK) {
        directionDisplay.setDirections(response);
    ... });
```

5. 구현 결과 및 성능 분석

5.1 구현 결과

그림 8와 같이 메인 화면에 주 기능과 관련된 메뉴를 배치하여 사용자가 쉽게 기능을 사용할 수 있도록 하였다.



그림 8. 메인 화면

사용자가 그림 8와 같이 지도에서 출발지와 도착지를 지정하면, 사진 분포 통계치가 상대적으로 높은 곳을 여러 곳 선별해서 나타내고 경로 찾기를 할 경우, 경유지를 고려한 최적 경로를 알려준다. 그리고 여행지를 선택할 경우, 여행지에서 촬영된 사진을 볼 수 있다.

탐색은 사용자가 여행하기 원하는 지역을 쉽게 찾을 수 있도록 도와주는 기능이다.

5.2 성능 분석

본 논문에서 구현하여 얻은 제주도의 사진 분포도의 상위 10개의 순위와 2012년 제주도청 관광지별 관광 인원 조사 결과[7] 상위 10개의 순위 비교는 표 1와 같다.

순위까지는 정확히 일치하지는 않지만 성산일출봉, 천제연폭포, 한라산국립공원, 천지연폭포, 산방산 6개가 적중하여 정확도는 60%이다. 이러한 결과는 사진 촬영 분포를 기준으로 하였기 때문에 사람이 자주 모이는 경기장과 공항이 상위권에 속해있어 정확도를 낮추는 요인으로 분석된다. 하지만 경기장과 공항을 제외하여 자연경관만 비교하면 7개 중 6개가 적중함으로 정확도가 85%가 된다.

표 1. 사진 촬영 분포도 순위와 2012년 제주도 관광지별 관광객 순위 비교

순위	사진 촬영 분포도	인원 조사 결과
1	성산일출봉	성산일출봉
2	천제연폭포	천지연폭포
3	제주국제공항	대포주상절리대
4	정방폭포	한라산국립공원
5	한라산국립공원	민속자연사박물관
6	천지연폭포	정방폭포
7	산방산	한림공원
8	제주월드컵경기장	만장굴관광지
9	선인교	천제연폭포
10	제주 경기장	산방산

또한, 찾고자 하는 여행 경로의 범위가 달라질 경우의

성능을 분석하기 위해 출발지는 고정하고 도착지의 위치 값(위도, 경도)을 점진적으로 증가시켜 처리속도의 변화를 측정하였다. 측정 시 도착지 위치 값(위도, 경도)의 소수점 둘째 자리를 1씩 증가시켰다. 측정 결과는 그림 9와 같으며, 도착지가 출발지로부터 멀어져도 처리속도는 큰 차이를 보이지 않음을 알 수 있다.

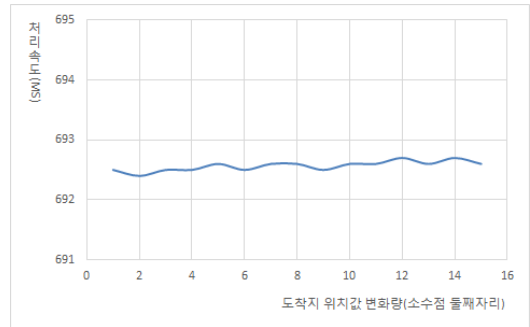


그림 9. 여행 경로 범위 증가에 따른 처리 속도

6. 결론

기존 방법들은 출발지와 목적지까지의 최적 경로만 제공한다는 단점을 가지고 있었다. 본 논문에서 제안하는 시스템에서는 사진 촬영 분포를 기반으로 여행 경로를 추천하여, 사용자는 유명한 지역이 어디인지를 쉽게 인식할 수 있었고, 여행 경로를 편리하게 계획할 수 있었다. 그리고 성능 분석 결과, 사진 촬영 분포도로 얻은 데이터가 관광청의 통계 데이터와 유사함을 알 수 있었다. 또한, 여행 경로의 범위가 다른 경우에도 성능에는 큰 차이가 없음을 알 수 있었다.

사진 촬영 분포가 유명한 지역을 나타내는 지표가 될 수 있으나 사용자가 원하는 여행지가 아닐 수 있다. 따라서 사용자 특성을 고려한 여행 경로 추천을 향후 과제로 남긴다.

참고문헌

- [1] 구글 지도, <https://maps.google.co.kr/>
- [2] 다음 지도, <https://http://map.daum.net/>
- [3] 최광선, 함영경, 김선호, “빅데이터 시각화,” 한국컴퓨터정보학회지, 제21권, 제1호, pp.33-43, 2013.
- [4] Google Maps JavaScript API, <https://developers.google.com/maps/documentation/javascript/reference>
- [5] 임재걸, 이강재, “여행지 최적 경로를 제공하는 웹 시스템의 설계와 구현,” 한국컴퓨터정보학회논문지 제12권, 제5호, pp.19-27, 2007.
- [6] 파노라미오, <http://www.panoramio.com>
- [7] 2012년 제주도 관광지별 관람인원, Islandhttp://stat.kosis.kr/nsieu/view/tree.do?task=branchView&hOrg=218&id=218*MT_OTITLE