

## 지형에서의 음영공간을 통과하는 최적경로 생성

임인선\*, 구자영\*\*

### A generation of Optimal Path Passing through Shadow Volume in Terrain

In-Sun Im\*, Ja-Young Koo\*\*

#### 요 약

전파가 미치지 않는 3차원 지형에서의 음영공간을 항공기가 안전하게 목적지까지 비행하는 최단 경로를 구하고자 하는 것이다. 여기서 비교적 넓은 음영공간을 다각형 형태의 볼록 다각형으로 분할하고, 분할된 볼록 다각형 내에서는 가중치 값에 따라 중간 목적지까지의 경로를 찾고, 그 중간 목적지를 시작점으로 하여 인접한 다각형에서 다시 목적지를 찾아가는 과정을 분할된 다각형 내에서 반복해서 최종 목적지까지의 최단 경로를 찾아가는 방법을 제안 하였다. 구현은 3차원 실 지형 상에서 전파가 미치지 않는 음영공간상에 Graph Growth 알고리즘의 임계값을 적용한 수정된 알고리즘을 이용하여 최단 경로를 탐색하였다. 실험에 의해 본 논문에서 제안한 방법이 효과적인 비행 궤적을 생성하였다.

#### Abstract

I intend to seek the shortest path which an aircraft safely flies to its destination through the shadow volume of three dimensional terrain which propagation does not extend. Herein, I propose a method in which a relatively wide shadow volume is divided with a polygon type of block polygon, a path up to the intermediate destination is sought in the divided block polygons according to the weight, and then, the shortest path up to the final destination is sought by repeating courses for seeking a destination again from the adjacent polygon based on the starting point of the intermediate destination within the divided polygon. The shortest path was searched by using the revised algorithms which apply to a critical angle from graph growth algorithms on the shadow volume which propagation on the actual three dimensional terrain does not extend to. The experiment shows that a method proposed in this paper creates effective flying curves.

▶ Keyword : 음영공간(Shadow Volume), 최단경로(Shortest Path)

---

• 제1저자 : 임인선  
• 접수일 : 2007.10.31, 심사일 : 2007.11.28, 심사완료일 : 2007.12.15.  
\* 단국대학 대학원 재학 \*\* 단국대학교 정보컴퓨터학부 교수

### 1. 서론

3차원 공간상에서 최단 경로 문제는 로봇의 이동이나 항공기의 비행경로와 같이 출발지에서부터 목적지까지 장애물에 부딪히지 않고 이르는 여러 경로 중에서 가장 효율적인 최단 경로를 선택하는 것이다. 이와 같은 분야의 연구는 일반적으로 정밀한 알고리즘을 개발하거나 이를 분석하는 것이 매우 복잡해지므로 휴리스틱 알고리즘을 개발하고 이를 실험적으로 보완하는 방법이 주로 사용된다. 한편, 알고리즘의 접근 방법은 최적 경로문제에서 발생하는 기하학적인 면들을 고려하는데 특히 계산 기하학 분야에서 개발된 여러 가지 도구들을 사용하여 알고리즘을 개발, 분석하는 것이다[1,2,3].

본 논문에서는 특정한 몇 개의 지점에서 전파를 송신했을 때 전파가 수신되지 않는 공간을 음영공간이라 하는데 이 음영공간 내에서 항공기가 안전하게 목적지까지 최단 경로로 비행할 수 있도록 경로를 구하고자 하는 것이다. 전파가 수신되지 않는 음영공간이 다각형 형태를 이루는데 이 다각형을 블록 다각형으로 분할하고, 분할된 블록 다각형 내에서는 노드와 링크에 x, y, z 방향의 안전 가중치 값을 부여해 주어 중간 목적지를 찾고 그 중간 목적지를 인접한 다각형에서는 출발지로 하여 다시 최종 목적지를 찾아가는 과정을 분할된 블록다각형 내에서 반복한다면 최종 목적지까지의 경로는 최단 경로가 된다. 최단 경로를 구하는 알고리즘 중에서 기존의 Floyd 알고리즘은 모든 노드에서 출발하여 출발한 노드를 제외한 모든 노드를 도착점으로 하는 즉 All-to-All 방식으로 설계된 최단 경로 알고리즘이고, Dijkstra 알고리즘이나, Graph Growth 알고리즘은 One-to-All 방식으로 설계되었기 때문에 출발 노드에서 다른 모든 노드까지 최단 경로를 탐색한다. 즉 목적 노드가 출발 노드와 가깝다 하더라도 최단 경로를 알아내기 위해서는 전체 노드를 탐색해야 하는 단점이 있다.[5]

여기서 Graph Growth 알고리즘의 단점을 보완하기 위해 임계값을 부여하여 어떤 목적지까지 탐색한 결과치보다 많은 경우에 저장하지 않도록 임계값을 적용하는 수정된 알고리즘을 적용하고자 한다.

본 논문의 구성은 다음과 같다. 제2장에서는 전파의 음영 공간을 추출하는 방법에 대해서, 제3장에서는 음영공간의 블록분할 및 그래프 모델링에 대해서 논하고, 제4장에서는 음영공간 내에서의 최단 경로 알고리즘에 대해서, 제5장에서는 실시간 상의 음영지역 내에서 Graph Growth 알고리즘의 수정된 알고리즘을 적용하여 항공기의 최단 경로를 구현한 결과를 보이고, 제6장에서는 결론을 서술한다.

### 2. 전파에 의한 음영공간의 추출

몇 개의 기지국에서 일정한 크기의 출력으로 전파를 송신하여 거리와 장애물에 의해 전파의 손실이 발생되는데 이때 수신되는 전파의 크기가 일정 값 이하의 전계 강도 값에 의해 형성되는 음영점 등의 집합을 연결한 공간이 음영공간이다..

주어진 지형 정보가  $Z=H(x,y)$  라는 고도 함수에 의해서 표현 된다고 가정하면, 음영점이 검사될 지점  $q = [q_x, q_y, q_z]^T$  과 전파를 송신하는 기지국의 위치  $v = [v_x, v_y, v_z]^T$  가 주어질 때, 선분  $vq$  상의 어떠한 점  $w = [w_x, w_y, w_z]^T$  에서도  $w_z > H(w_x, w_y)$  를 만족하는 것이다. 이를 검사하기 위한 알고리즘은 다음과 같다.

- 1)  $a = 0$   
 $\Delta a = d/D$  //  $q = [q_x, q_y, q_z]^T$  와  $v = [v_x, v_y, v_z]^T$  // 의 xy 평면상의 거리,  
 // 즉  $[v_x, v_y, v_z]^T$  와  $[q_x, q_y, q_z]^T$  의 거리  
 // d : 가시성 검사를 위한 xy 평면상의 거리  
 증분

2) 다음을 계산한다.

$$\begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} = (1 - \alpha) \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} + \alpha \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} + \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}$$

3)  $h_w = H(w_x, w_y)$  값을 다음과 같이 쌍일차 보간에 의해서 구한다.

$$h_w = (1-r_x)(1-r_y)H(n_x, n_y) + r_x(1-r_y)H(n_x + 1, n_y) + (1-r_x)r_yH(n_x, n_y + 1) + r_x r_y H(n_x + 1, n_y + 1)$$

4)  $w_z < h_w$  이면 return (FALSE)  
 (q를 비가시점으로 결정)

5)  $a = a + \Delta a$  ;  $a \geq 1$ 이면 return (TRUE)  
 (q를 가시 점으로 결정)

6) 단계 2)로 간다.

기지국 v로부터 음영점들의 집합을 Coverage(v)라고 표  
 기하며, 기지국들의 집합  $v = \{v_1 \dots v_n\}$  으로 부터의  
 음영점들의 집합을 Coverage(v) = U Coverage(v<sub>i</sub>)로 나  
 타낸다. 여기서 Coverage(v)를 극대화하는 v를 구하는 것이  
 며, U 은 합집합 연산자이다.

### 3. 음영공간의 블록분할 및 그래프 모델링

#### 3.1 음영공간의 블록분할

다각형 내에서의 최단 경로에 관해서는 많은 연구가 이루어  
 졌다. 이런 연구의 목적은 다각형의 노드의 개수가 n이라 할  
 때, 시간 복잡도가 n에 관한 작은 차수의 다항식인 최단경로  
 알고리즘을 구하는 것이다. 대부분의 방법들은 볼록다각형이  
 나, 삼각 분할을 기반으로 하여 최단 경로를 구하는데 다각형  
 의 볼록다각형 분할에 관해서는 선형 시간의 알고리즘이 알려  
 져 있다.

볼록 다각형이란 다각형에서 어느 변을 연장해도 그 다각  
 형을 자르지 않는 사각형 또는 모든 내각이 180도 보다 작은  
 다각형을 말하는데, 다각형의 볼록 분할을 위해서는  
 Schachter[4]의 방법을 사용했다.

#### 3.2 그래프 모델링

이 방법은 Voronoi tessellation의 Dual인 Delaunay  
 tessellation에 근거한 것이다.

Voronoi tessellation은 다각형의 꼭지점을 중심으로 그  
 점으로부터 가장 가까운 점들의 집합으로 공간을 분할한 것이  
 다. 이때 하나의 영역을 Voronoi cell 이라 하고 인접한  
 Voronoi cell을 Voronoi neighbour 이라고 한다.

Voronoi neighbour들의 중심점을 연결한 것이 Delaunay  
 tessellation이다. 볼록 분할은 부분적 Delaunay tessellation  
 을 구하는 것으로서 비 볼록 꼭지점에서 내각을 분할하여 볼록  
 다각형들로 나누는 것이다. 하나의 비 볼록 꼭지점 V'를 분할  
 하는 알고리즘은 다음과 같다.

1. 각 다각형의 정점 중에서 반사 정점(Reflex vertex)을 찾  
 는다.
2. 반사 정점 V에 가장 가까운 가시 정점(Visible vertex)  
 V'을 찾는다.
3. 만약 V'가 V의 Inner cone안에 들어오면 V와 V'를 연결  
 하고 return한다.
4. V에서 출발하는 Delaunay edge들을 구해서 Inner cone  
 안에 들어오는 것이 있으면 그 중 하나를 연결하고(그림 1)  
 그렇지 않으면 Inner cone의 좌우에 있는 두 개의  
 Delaunay edge를 연결한다.
5. return한다.

(그림 1)은 Inner Cone 안에 Delaunay edge가 있는 경  
 우이며, (그림 2)는 Inner Cone 안에 Delaunay edge가 없  
 는 경우이다.

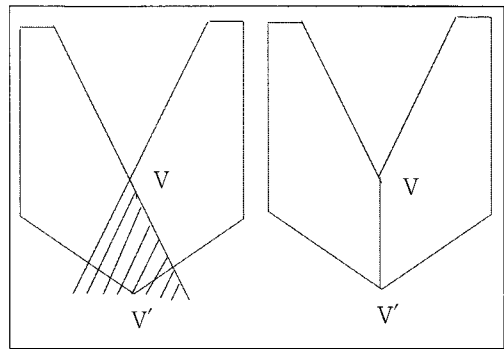


그림 1. Inner Cone 안에 Delaunay edge가 있는 경우  
 Fig 1. A case with a Delaunay edge in the Inner cone

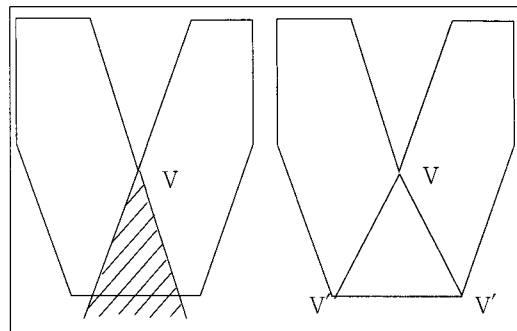


그림 2. Inner Cone 안에 Delaunay edge가 없는 경우  
 Fig 2. A case Without a Delaunay edge in the Inner cone

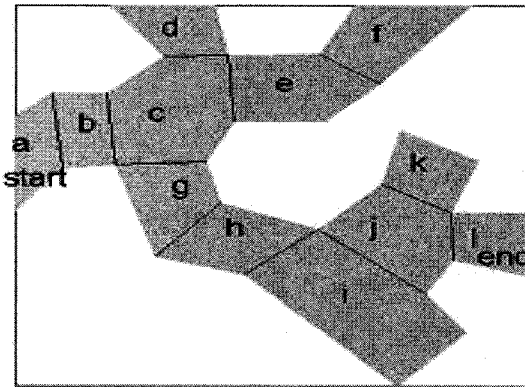


그림 3. 볼록 분할의 예  
Fig 3. An example of convex decomposition

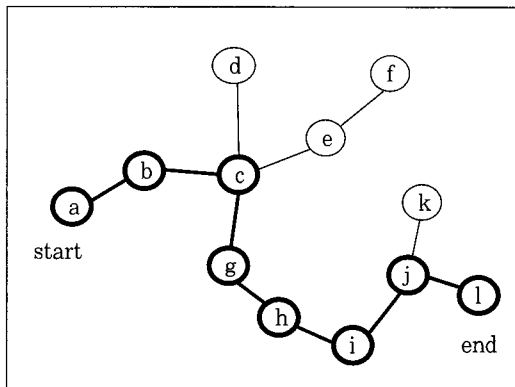


그림 4. 인접그래프와 최단경로  
Fig 4. Adjacency graph and shortest path

#### 4. 음영공간 내에서의 최단경로 알고리즘

어떤 지역의 최단 경로를 탐색하는 데는 많은 시간이 소요 되므로 일정 범위의 지역을 볼록 다각형으로 분할하고 분할된 볼록 다각형 내에서 최단 경로 구하는 알고리즘을 적용하여 전체적인 최단 경로를 구하고자 하는데, 처음 시작점과 마지막 목적지는 알고 있다고 가정하면, 각 분할된 다각형 내에서 최저 고도 값 이상인 상태에서 수정된 Graph Growth 알고리즘을 이용하여 최단 경로를 찾는다.

```

Input : s(start node), t(target node)
Output : shortest path lists
Procedure ShortestPath_Algorithm(s,t)
    begin
        Queue_Initialization(Q);
        for i=1 to n do
            d(s,i) = ∞;
            Threshold = 0;
            d(s,s) = 0;
            Queue_Insertion(Q,s);
        While (Queue ≠ Null) do
            begin
                Queue_Removal(Q,i);
                for each j adjacent to i do
                    if d(s,j) > d(s,i) + link(i,j) then
                        begin
                            d(s,j) = d(s,i) + link(i,j);
                            if d(s,j) < Threshold then
                                if i = t then
                                    Threshold = d(s,i);
                                else
                                    Queue_Insertion(Q,j);
                                end
                            end
                        end
                    end
            end
    end
    
```

그림 5. Graph Growth의 수정된 알고리즘  
Fig 5. Modify algorithms of Graph Growth

본 논문에서 적용하고자 하는 그림 5의 Graph Growth의 수정된 알고리즘의 기본적인 개념은 처음에 출발 노드에서 목적지 노드까지의 경로 값을 임계값으로 줄여서 목적지 노드가 검색되면 현재까지의 값을 임계값과 비교하여 비용이 임계값 이상인 경우에는 더 이상 탐색하지 않도록 큐에 삽입하지 않고, 비용이 임계값보다 적으면 기존의 임계값을 대체한다.(8)

Graph Growth 알고리즘은 수행을 마치는 순간 모든 노드들의 최단 경로를 찾게 되지만, 본 논문에서 적용한 Graph Growth 수정된 알고리즘은 임계값 이하로 검색되는 노드들에 대해서만 최단 경로를 찾게 된다. 그림 6은 Graph Growth의 수정된 알고리즘의 방문 노드 수를 비교해보기 위한 노드와 링크를 구성한 그림이다. 그림 6의 그래프에서 시

작 노드를 2로 하고 목적 노드를 13으로 설정된 예를 들어 설명하면, Graph Growth 알고리즘은 어떠한 최단 경로를 구성하더라도 노드 탐색 횟수는 27번이다. 그 방문 경로는 노드 2→1→3→7→6→4→8→12→5→9→13→11→17→10→14→16→22→18→13→14→15→19→21→23→20→24→25이다.

그러나 본 논문에서 수정된 알고리즘을 사용하면 목적 노드 13의 최단 경로를 위한 탐색 횟수는 14번이며 그 방문 경로는 노드 2→1(비용:2)→3(비용:7)→7(비용:5)→6(비용:6)→4(비용 :13)→8(비용:10)→12(비용:8)→5(비용:15)→9(비용:15)→13(비용:16)→11(비용:11)→17(비용:13)→13(비용:10)이다.

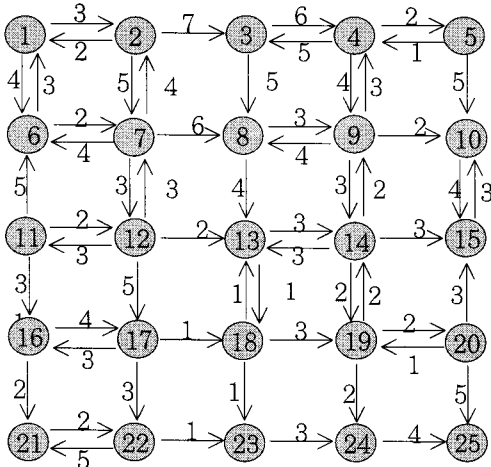


그림 6. 노드의 구성  
Fig 6. Composition of Node

더 이상 노드를 방문해도 비용이 10이하인 경우가 없으므로 방문을 종료한다.

이때 방문해서 찾은 최단 경로의 노드는 2→7→12→13이며, 비용은 10이다.

### 5. 구현 결과

본 논문에서 All-to-All 방식의 Floyd 알고리즘과 One-to-All 방식인 Dijkstra 알고리즘, Graph Growth 알고리즘을 실행한 후 비교한 결과 GIS의 실 지형에서는 Graph Growth 알고리즘이 빠르게 수행되었으며, Graph Growth 알고리즘에서 임계값을 주어 수정한 알고리즘의 실

행시간이 가장 빠른 것으로 탐색 횟수에서 알 수 있었다. 본 논문의 구현을 위하여 전파 예측을 GIS (Geographic Information System)에 적용한 음영 공간 내에서 항공기의 최단 경로 추적을 Graph Growth 알고리즘을 수정한 알고리즘(그림 5)으로 2차원 및 3차원 공간상에서 구현하였다. 지리적인 위치는 경기도 남부지역(경도 127° 8' 11", 위도 37° 13' 1")을 중심으로 약 1 : 12만의 축척의 지도로, 약 32.4Km 반경 내에서 4개의 기지국에서 전파를 발사하여 전파가 미치지 않는 3차원 음영공간이 구성되었다. 다각형으로 구성된 음영공간 내에서 고도는 지상으로부터 항공기의 최저 고도인 약 450피트(137미터) 이상을 이격하여 비행할 수 있도록 하여 최단 경로 추적 시스템을 구현 하였다.[7]

(그림 7)은 경기도 남부의 지역으로 시가지, 도로, 등고선 등이 표기된 2차원 지도이며, (그림 8)은 (그림 7)의 3차원 지도로 기울기 65°, 회전각 30°를 주었으며, 기지국 4개의 위치는 <표 1>과 같으며, 도곽의 해상도는 경도 1초는 31m이며, 위도 1초는 22m이다.

표 1. 기지국의 위치  
Table 1. Position of site

비고	#1	#2	#3	#4
경도	127°02'40"	127°19'49"	127°17'22"	126°56'48"
위도	37°25'08"	37°07'03"	37°20'43"	37°09'16"
고도	573.9m	411.8m	489.8m	188.3m
송신 출력 : 10 Kw				
주파수 : 1000 Mhz				
전계 강도 : 59				

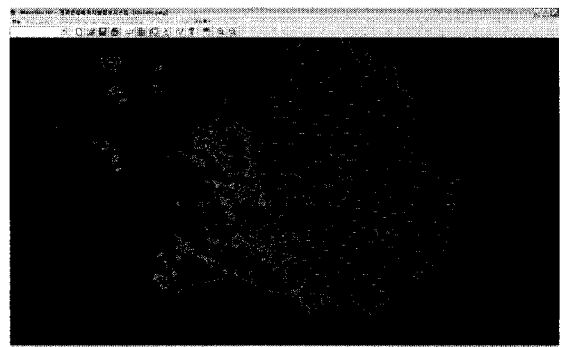


그림 7. 경기도의 지도  
Fig 7. Map of Gyeonggido

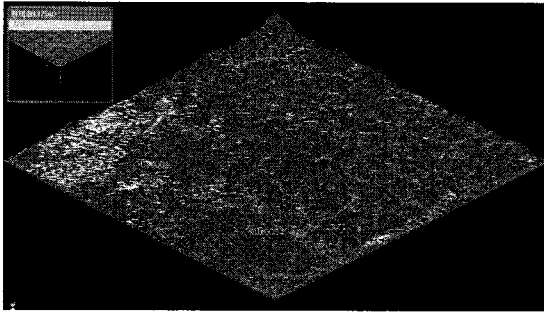


그림 8. 경기도의 3차원 지도  
Fig 8. 3-D Map of Gyeonggi-do

(그림 9)와 (그림 10)은 2차원과 3차원 상에서 표1의 4개의 기지국에서 안테나 높이 5m와 송신 출력과 주파수 및 전계강도 값을 주었을 때의 전파의 반경 및 음영 공간을 표시한 것이다. 여기서 파란색으로 표시된 공간이 전파가 미치는 공간이므로 파란색영역으로 둘러싸인 짙은 검은색부분이 음영공간이다.

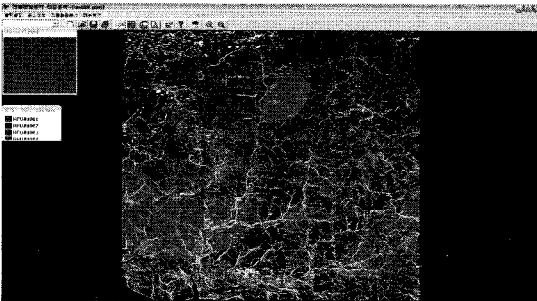


그림 9. 전파의 음영 공간(2차원)  
Fig 9. Shadow volume of Propagation(2-D)

(그림 11)과 (그림 12)는 2차원 및 3차원 상의 전파가 미치지 않는 음영공간을 다각형으로 구성한 다음 블록다각형으로 분할하여 비용을 부여하였다. 비용 부과 방법은 X축 과 Y축에서



그림 10. 전파의 음영 공간(3차원)  
Fig 10. Shadow volume of Propagation(3-D)

음영지역의 증가는 가장 적게 부여하고 외곽으로 갈수록 적은 비용을 부과하였으며(1~5사의의 값), Z축 즉 고도의 비용 부과는 최저 고도를 유지하게 부여하였다. 여기서 흰색 선이 임의의 한 출발 지점에서 목표 지점까지 비행의 최저 고도인 450 ft(약 137m) 이상을 유지하면서 비행하는 경로를 나타낸 그림이다.

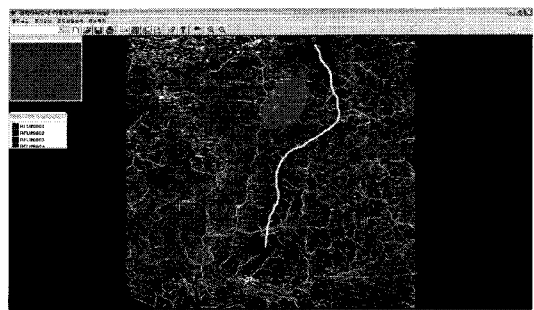


그림 11. 음영공간의 비행경로(2차원)  
Fig 11. Flying path of Shadow volume (2-D)

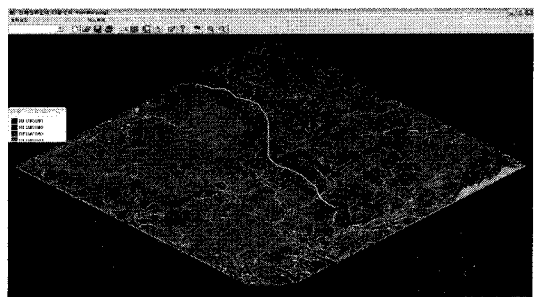


그림 12. 음영공간의 비행경로(3차원)  
(Fig 12) Flying path of Shadow volume (3-D)

## 6. 결론

본 논문에서는 <표 1>과 같이 4개의 기지국에서 일정한 주파수로 송신했을 때 수신이 미치지 못하는 음영 공간상에서 출발점에서 목적지까지의 최단경로의 결과를 얻을 수 있었다. 본 논문에서 최단 경로를 구하기 위해 Graph Growth 알고리즘의 수정된 알고리즘을 다각형 분할 방법을 이용하여 작은 다각형으로 분할하여 적용하였으므로 탐색 횟수 등이 현저히 감소 되었으며, 실제 현실에서와 같이 대단히 넓은 영역에서도 적용이 가능하다. 그러나 전파의 회절, 굴절 등의 고려사항과 송신 출력, 수신 전파의 전계강도 및 항공기의 속도, 곡률 등을 모두 적용하고, 음영지역의 다각형 구성을 세밀하게 적용한다면 더 많은 컴퓨팅 파워를 필요로 한다. 컴퓨터 그래

픽이나, 컴퓨터 비전 분야에서 최단 경로 탐지와 추적에 관한 문제를 해결하기 위해 더 많은 연구가 필요하다.

### 참고문헌

- [1] D. T.Lee and F. Preparata. "Euclidean shortest paths in the presence of rectilinear barriers". Networks, 14:393-410. 1994.
- [2] F. B. Zhan and C. E. Noon, "A Comparison Between Label-Setting and Label-Correcting for Computing One-to-one Shortest Paths", Journal of Geographic Information and Decision Analysis, Vol 4, no. 2, pp.1-11, 2000.
- [3] F. B. Zhan, "Three Fast Shortest Path Algorithms on Real Road Networks : Data Structures and Procedures", Journal of Geographic Information and Decision Analysis, Vol 1, pp.69-82, 1999.
- [4] Bruce Schachter, "Decomposition of Polygons into Convex Sets", IEEE Trans on Computers, Vol. 27, No11, pp 1078- 1081, 1997.
- [5] B. V Cherkassky, A. V . Goldberg and T. Radzik, " Shortest paths algorithms : Theory and experimental evaluation," Mathematical Programming 73, pp129-174, 1996.
- [6] 추연웅, 구자영. "다각형을 통과하는 곡선의 최적화", 정보처리학회 논문지, 제5권 제9호, pp89-95, 1998.
- [7] 임인선. "인터넷상의 동영상에서의 물체 특징점 탐지 및 추적", 컴퓨터정보학회 논문지, 제10권 제1호. 2005.
- [8] 심충섭, 김진석. "One-to-One 최단 경로 탐색 알고리즘의 성능 평가", 정보과학회 논문지, 제29권 제11호. 2002.

### 저자소개

임인선  
단국대학교 대학원



구자영  
1986년 2월 : 한국과학기술원 공학  
박사  
1986년~ 현재 : 단국대학교 정보컴  
퓨터학부 교수