제 3 장 RADIANCE 프로그램

3.1. RADIANCE 프로그램의 개요

3.1.1. RADIANCE 프로그램의 개요 및 기본 환경

RADIANCE 프로그램은 미국 국립 Lawrence Berkeley Laboratory(LBL)의 조명연구팀이 개발한 프로그램으로서 1987년 이래로 버클리의 캘리포니아 대학 건축공학과에서 사용되어오고 있는 조명 렌더링 및 시뮬레이션 프로그램이다. RADIANCE 프로그램은 UNIX환경의 워크스테이션급 컴퓨터에서 적합하게 실행될 수 있으며, C언어로 컴파일링 되어있다. [표 3.1]은 RADIANCE 프로그램이 실행될 수 있는 사양과 RADIANCE 프로그램에서 사용할 수 있는 파일로 변환할 수 있는 파일 형식을 보인 것이며, [표 3.2]는 GI(Graphics Interface)를 지원하는 환경을 보인 것이다. 아래와 같이 이전까지의 사용환경은 UNIX를 OS로 사용하는 워크스테이션급 컴퓨터에서 실행되었으나, 현재 PC의 Windows 환경에서 사용할 수 있도록 개발되어있는 상태이다.

[표 3.1] RADIANCE 프로그램의 환경

실행 가능한 환경	변환 가능한 파일 형식
• Sun3, Sun 4 workstarion • DECstation running ULTRIX • Silicon Graphics IRIS • Mac II running A/UX • IBM running AIX	• GDS Things File • IES Luminaire Data • Sun 8 and 24-bit Rasterfiles • Architrion Text File • AutoCAD DXF (next release) • Targa 8, 16, 24 and 32-bit images

[표 3.2] RADIANCE 프로그램의 GI(Graphics Interface) 지원환경

GI 지원환경
• X11 8-bit color or greyscale and 24-bit color displays • X10 8-bit color or greyscale displays • SunView 8-bit color or greyscale • News color or greyscale • AED 512 color graphics terminal

RADIANCE 프로그램은 무료로 배포되고 있어서 인터넷 사용자는 쉽게 이 프로그램을 얻을 수 있으며, 개발자 및 사용 경험이 풍부한 사용자들로부터 많은 도움을 받을 수 있다. 또한 전세계적으로 인터넷을 이용한 사용자 그룹이 형성되어 있어 사용상의 문제에 대한 토론, 정보와 자료의 교환이 이루어지고 있다. 이러한 상황으로 미루어보아, 앞으로 RADIANCE 프로그램이 미국의 연구개발 그룹과 고도의 설계를 수행하는 그룹들에게 표준으로 될 가능성이 크다고 할 수 있다(김훈, 1997).

3.1.2. RADIANCE 프로그램의 기본 개념

RADIANCE 프로그램은 역광선추적기법(Backwards Ray-tracing Technique)을 기초로 한다. 이것은 광선이 자연적으로 진행하는 방향의 반대 방향으로 추적하여, 실제 광선이 발생한 광원의 활동을 예측하는 것을 의미한다. 그 과정은 눈으로부터 시작하여 공간의 대상물들의 표면들을 따라 모든 물리적 상호작용을 계산하여 광원까지의 광선을 쫓아가게 된다(Ward, 1994).

이러한 개념은 2장에서 설명한 몬테카를로 방법과 광선추적기법에 기본 바탕을 두고 있다. 실제 공간에서 광원으로부터 반사된 광선은 결국 흡수될 때까지 반사, 투과되는데 우리가 실제로 보게되는 장면은 이러한 광선들이 망막에 영상을 만든 것이다. 이처럼 망막에 영상이 맺히게되는 것은 공간내에 존재하는 전체 광선중 일부분만이 재실자의 눈에 들어옴으로써 이루어지는 것이므로, 눈으로부터 주변환경

을 거쳐 광원으로까지의 광선을 역으로 추적함으로써, 광원으로부터 나온 광선들의 거동을 확인하여 빛환경을 가시화할 수 있게 된다(Crone, 1992). 다음은 RADIANCE에 기본을 이루는 개념들을 정리하여 보인 것이다(Compagnon, 1997).

RADIANCE 프로그램은 대상물의 3차원 기하학적인 영역에 의한 Scene 파일을 기본으로하는데, 이러한 3차원 영역은 표면들로 이로워져 있으며 공간의 비어있는 상태가 된다. 이러한 공간의 표면은 각각 정의된 방향을 갖고 있으며, 이러한 방향은 데카르트 좌표계(XYZ좌표)에 기초한다. 기본적으로 X축은 동쪽을 향하고, Y축은 북쪽을 향하며, Z축은 천장을 향한다. 일반적으로 X축을 따라 공간의 주요 표면(예를 들어, 대상실의 벽과 같은 수직평면)을 위치시키고, 그에 따라 나머지들을 정렬하는 방식으로 진행하게 된다. 좌표는 길이 단위로 주어지게 되며, 대상공간에서 길이는 상대적으로 작용하게 된다. 각 표면은 오른손 법칙에 따라 좌표의 입력순서가 시계반대방향일 때 사용자가 보는 면(바깥쪽 면)으로 인식된다(반대의 경우에는 안쪽면이 되는 것).

RADIANCE 프로그램 내부에서 사용되는 광선은 [W/m^{2}sr]로 표현되는 복사에 의해 이루어진다. 이러한 복사는 red, green, blue의 세가지 주요 색채에 의해 세가지 채널(RGB)로 나누어 지게 된다. 그에 따라 총 복사값은 세가지 채널에 의해 공급된 Rr, Rg, Rb의 합으로 구성된다.

$$R = 0.263 \cdot Rr + 0.655 \cdot Rg + 0.082 \cdot Rb \text{ [W/m}^2\text{sr]} \qquad \textbf{(식 3.1)}$$

($\because$ 0.263 + 0.655 + 0.082 = 1)

휘도 L값은 위 (식 3.1)에서 얻어진 총복사값 R에 의해 (식 3.2)와 같이 변환된다.

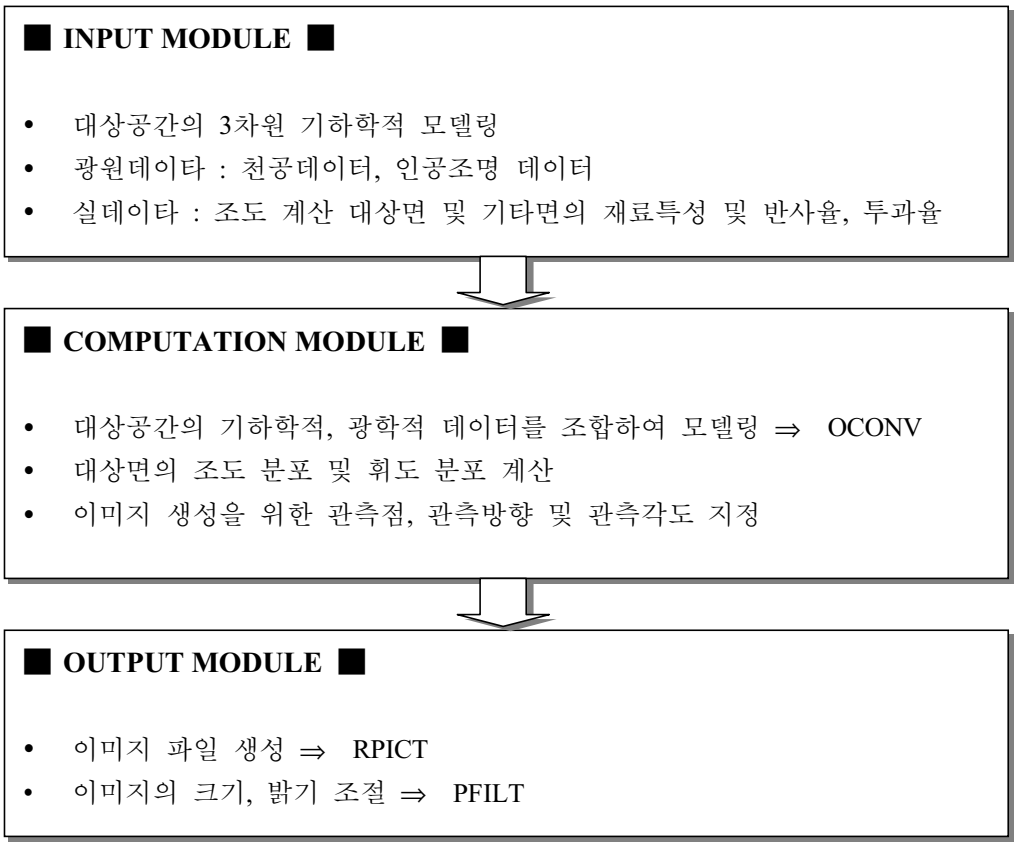
$$L = 179R = 47.1Rr + 117.2Rg + 14.7Rb \qquad \textbf{(식 3.2)}$$

이러한 색채널에 의해 광선을 모델링하는 방법은 빛의 스펙트럼 분포가 좁은 파장대역에서 많은 채널을 사용할 경우에 비교적 정확하지 못한 결과를 낳을 수도 있으나, 재료나 천공에 대한 색을 표현할 경우에는 유용한 결과를 얻을 수 있다(예, CIE XYZ좌표계).

3.2. RADIANCE 프로그램의 구성요소

3.2.1. RADIANCE 프로그램의 기본 요소

이 프로그램은 다수의 독립된 프로그램 모듈로 구성되어 있어서 3차원 공간의 기하학적 모델링에 필요한 도구, 광원 및 재료의 광학적 특성을 모델링하는데 필요한 도구, 계산결과를 가시화하여 이미지 파일을 작성 및 변환하는 도구, 눈부심 현상을 분석하여 가시화하는 도구 등 빛환경의 모델링 및 분석, 평가에 필요한 거의 모든 기능을 제공하고 있다(Crone, 1992; Ward, 1998). [그림 3.1]은 RADIANCE 프로그램의 3가지 기본모듈을 보인 것이다.



[그림 3.1] RADIANCE 프로그램의 기본모듈

입력모듈에는 대상공간의 3차원 기하학적 요소와 천공 및 인공조명을 모델링하기 위한 광원데이터, 그리고 공간에 사용된 재료 및 반사율, 투과율을 입력한다. 계산 모듈에서는 입력모듈서의 데이터를 모두 조합하여 모델링하여 대상면의 조도 및 휘도분포를 계산하고, 이미지 생성을 위한 관측점, 관측방향, 관측각도를 지정한다. 이러한 지정에 따라 이미지 파일을 생성하고 크기 및 밝기를 조절하여 원하는 이미지를 출력한다.

3.2.2. RADIANCE 프로그램의 구성 파일

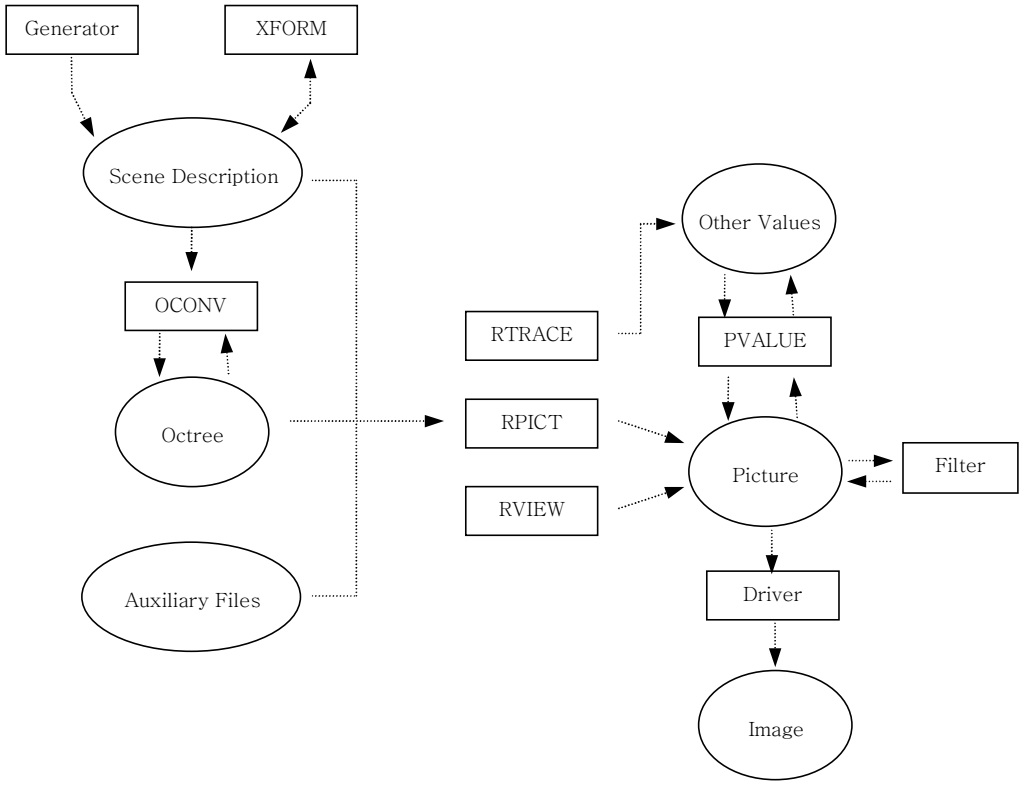
[표 3.3]은 위의 과정에 따른 RADIANCE 프로그램의 내부의 파일 형식을 보이고 있다. 각각의 텍스트 파일은 사용자가 직접 작성하여 적용되는 파일이며, 바이너리 파일은 RADIANCE 프로그램내에서 시뮬레이션 및 변환된 파일이다. 텍스트 파일외의 바이너리* 파일들을 생성하는데에 사용된 모든 명령어와 매개 변수의 목록을 보기 위해서는 GETINFO 명령을 사용한다.

[표 3.3] RADIANCE 프로그램의 파일구성

파 일 형 식	확장자	파일타입
• Scene - materials and geometry definitions	.rad	Text
• Numeric data tables	.dat	Text
• Functions	.cal	Text
• Internediate data to calculate glare indices	.gla	Text
• Progress reports	.log	Text
• Parameters defining a RADIANCE project	.rif	Text
• Octree – compiled version of a scene	.oct	Binary*
• RADIANCE picture	.pic	Binary*
• Ambient illuminance values	.amb	Binary*
• TIFF formet picture	.tif	Binary

3.2.3. RADIANCE 프로그램의 기본 흐름

[그림 3.2]는 RADIANCE 프로그램의 구성 및 데이터 처리과정을 보여주는 것이다. 사용자는 RADIANCE 프로그램에서 제공하는 Generator(기본적인 텍스트 파일 편집기)를 이용하여 대상공간의 기하학적 자료를 입력하여 *.rad(Scene file) 파일을 작성한다. 같은 형상이 다수 존재할 경우, 하나의 형상을 모델링하여 XFORM 명령에 의해 여러곳에 복사하고 회전시킬 수도 있다. 그러나, ACAD2RAD 또는 TORAD와 같은 보조 변환 프로그램을 이용할 경우 3차원으로 작성된 AutoCAD 파일로부터 대상실의 기하학적 자료를 갖는 *.rad 파일을 자동 생성할 수 있게 되는데 이러한 경우 사용자는 텍스트 편집기를 이용하여 대상물의 재료특성의 모델링만 추가적으로 해주면 된다.



[그림 3.2] RADIANCE 프로그램의 흐름도

위와 같은 방법에 의해 대상공간의 기하학적, 광학적 모델링이 완료되면 OCONV 명령을 실행하여 광선추적기법을 적용하기 위한 octree 파일, 즉 *.oct 파일로 변환한다. 그리고 RVIEW, RPICT, RTRACE 등의 명령에 의한 광선추적 결과로부터 조도 분포 및 휘도 분포 값을 계산하고, 그 결과로부터 이미지 파일인 *.pic 파일을 생성한다. *.pic 파일은 PFILT 명령과 같은 그래픽 필터를 이용하여 이미지의 크기나 밝기를 조절하고 화면에 이미지를 가시화하게 된다. 이미지의 생성을 위하여 사용자는 RVIEW, RPICT 명령에 관측점, 관측방향 및 관측각도 등을 지정하는 매개 변수를 입력하여야 한다.

3.3. RADIANCE 프로그램의 주요 실행 명령

[표 3.4]는 지금까지 설명한 RADIANCE 프로그램의 구성요소와 기본 흐름에서 사용되는 RADIANCE 프로그램에 포함된 주요 실행 명령들의 목록과 설명을 보인 것이다.

[표 3.4] RADIANCE 프로그램 목록

명령어	내용
AEDIMAGE	AED 512 colror graphics을 위한 화면 출력명령
ACAD2RAD*	AutoCAD 프로그램에서 작성한 공간을 RAD 파일로 변환
ARCH2RAD	Architrion text 파일을 RAD 파일로 변환
BGRAPH	Batch graph 설정
CALC	계산기
CNT	Index counter
DAYFACT	작업면상에 유입되는 조도와 주광율을 계산
FALSECOLOR*	등고선 형태의 RADIANCE 이미지 파일 생성
FINDGLARE	현휘의 발생위치 계산
GENBOX*	육면체 형태 생성
GENPRISM	각기둥 형태 생성
GENREV	자유로운 곡면 형태 생성
GENSKY*	외부천공 모델링
GENSURF	함수값에 따른 면 형태 생성
GENWORM	나선 형태 생성
GENINFO*	RADIANCE 파일의 정보 확인
GLARE	현휘와 시각적 안정성 계산
GLARENDX	Glare index 계산
IES2RAD*	IES 파일을 RADIANCE 프로그램에 맞는 파일로 변환
LAMPCOLOR	조명원의 색분포 계산
MKILLUM*	RADIANCE 프로그램의 조도 계산
MGF2RAD	Materials and Geometry Format 파일을 RAD 파일로 변환

명령어	내용
NORMPAT	RADIANCE 이미지에 사용된 형식 일반화
OBJVIEW	RADIACEN object 보기
OCONV*	Scene 파일로부터 Octree 파일 생성
PCOMB	RADIANCE 이미지 합성
PCOMPOS	RADIANCE 이미지 조절
PFILT*	RADIANCE 이미지의 필터링
PINTERP	이미지에 다른 이미지를 삽입 또는 적출
PROTATE	이미지의 회전
PSIGN	텍스트로부터 이미지 생성
PVALUE	다른 형식으로부터(혹은 으로) RADIANCE 이미지 변환
RA_BN	Barneyscan 이미지로부터(혹은 로) RADIANCE 이미지 변환
RA_PIXAR	PIXAR 그림으로부터(혹은 으로) RADIANCE 이미지 변환
RA_PPM	Poskanzer Portable Pixmap으로부터(혹은 으로) RADIANCE 이미지 변환
RA_PR	Pixrect rasterfile로부터(혹은 로) RADIANCE 이미지 변환
RA_PR24	24-bit rasterfile로부터(혹은 로) RADIANCE 이미지 변환
RA_RGBE	RADIANCE 이미지의 실행길이 변환
RA_T16*	16-bit Targa 파일로부터(혹은 로) RADIANCE 이미지 변환
RA_T8*	8-bit Targa 파일로부터(혹은 로) RADIANCE 이미지 변환
RA_TIFF*	TIFF 이미지 파일로부터(혹은 로) RADIANCE 이미지 변환
RAD	RADIANCE 이미지의 렌더링
RCALC*	기록 계산기
RPICT*	RADIANCE 이미지 파일 생성
RTRACE*	RADIANCE 프로그램의 광선추적
RVIEW*	RADIACNE 이미지의 상호작용 계산
THF2RAD	GDS 파일로부터 RADIANCE로 변환
XFORM*	Scene 파일의 회전, 이동 등의 변환
XIMAGE*	X Window System에서 RADIANCE 이미지의 화면출력

위에서 살펴본 명령들 중 본 연구에서 이용된 주요 명령어(*로 표식)들의 기본적인 내용은 다음과 같다.

3.3.1 공간의 3차원 모델링을 위한 명령 및 파일 작성

공간을 3차원 모델링하기 위해서는 대상공간의 기하학적인 요소를 포함하는 파일, 공간내부에 사용된 재료의 특성을 포함하는 파일, 그리고 야간시 조명특성을 나타내는 파일 등일 필요하게 된다.

1) 공간의 3차원 기하학적 요소

공간을 3차원의 기하학적인 형태로 모델링하기 위한 RADIANCE 프로그램의 주체가 되는 *.rad 파일 즉, Scene 파일은 크게 4가지 기본 데이터형식으로 구성된다.

— 주석(comments)

Scene 파일의 도입부에 위치하며, 프로그램 설명을 위해 사용된다. 문장의 처음부분에 #기호를 사용하여 프로그램 언어가 아닌 주석으로 사용되는 것이다.

형식) # all lines beginning with character

예) # scene.rad 파일의 이름명시

This is Material file. 파일의 내용명시

— 초기 형식(Primitives)

파일의 주요 내용부분이며, 프로그램의 주체가 된다. 아래와 같은 형식으로 입력되며 [표 3.5]는 아래의 형식에 위치한 각각의 기호를 설명한 것이다.

형식) modifier type identifier

N S_1 S_2 S_3 S_M

0

M R_1 R_2 R_3 R_M

예) **void** plastic red_paint

0

0

5 1 .02 .02 .03 0

red_paint sphere red_ball
0
0
4 1 1.5 0 2

[표 3.5] rad 파일의 주요 형식

이 름	내 용
Modifier	이전에 정의된 identifier를 위치시키거나 identifier가 정의되지 않았을 경우에는 'void'를 사용
Type	입력되는 도형의 형식이나 파일의 형식
Identifier	위의 형식에 따라 설정된 이름
N	문자형 인수의 수
M	실수 인수의 수
S_X	특정 초기 형식을 위한 문자형 인수
R_X	특정 초기 형식을 위한 실수 인수

— 외부 프로그램의 호출(calls to external programs)

Scene 파일 내부에서 다시 다른 프로그램을 호출하여 실행시킬 때 사용하며, 호출된 프로그램의 출력은 Scene 파일의 문법에 맞는 형식에 의해 지정되어야 한다. 명령파일의 입력값이 길어질 경우, '\'기호를 사용하여 여러 줄로 연장할 수 있다.

형식) ! program_name parameters

예) ! genbox cinderblock fireplace 1.5 2 1.5 -i

— 이름의 변경(identifier aliases)

조건(modifier)에 의해 identifier를 다른 이름으로 변경한다.

형식) modifier alias new_identifier old_identifier

예) Alberta_image **alias** Alberta_photo Photo_paper

2) 인공광원의 모델링

인공조명을 시뮬레이션하기 위해서는 조명원의 배광분포곡선에 따라 각도별 광도값을 *.rad 파일로 작성하면 되지만, 조명기구를 생산하는 회사에서 배포하는 IES 파일 형식을 이용하면, IES2RAD 명령을 통하여 쉽게 조명파일을 작성할 수 있다.

IES2RAD 명령은 아래와 같은 형식으로 실행시키며, 그에 따라 데이터 파일(*.dat)과 Scene 파일(*.rad)이 생성된다.

형식) ies2rad -t lamp_type -c r g b input_filename

예) **ies2rad** -t default -c .225 .75 0 ies02

3) CAD 파일의 변환

대상공간의 3차원적 요소와 재료특성은 앞에서 설명한 바와 같이 *.rad 파일로 작성하면 되는데, 3차원 기하학적 형태의 경우에는 ACAD2RAD.lsp나 TORAD.lsp 파일에 의한 각 명령등에 의해 CAD 파일을 변환하여 사용할 수 있다.

각각의 명령은 다음과 같은 형식으로 AutoCAD 프로그램에서 불러들여서 실행시키게 된다.

예) command: (load “c:/lisp 파일이 위치한 directory/**acad2rad**.lsp”)

command: (load “c:/lisp 파일이 위치한 directory/**torad**.lsp”)

각각의 명령에 따라 ACAD2RAD 명령의 경우에는 모든 3차원 요소를 하나의

rad 파일로 변환을 하고, TORAD 명령의 경우에는 CAD 에서 지정된 layer 별, 혹은 각 color 별로 별도의 파일을 작성한다.

[그림 3.3]은 AutoCAD 프로그램에서 TORAD 명령을 실행하였을 때 입력을 받아들이기 위한 명령들을 보인 것이다.

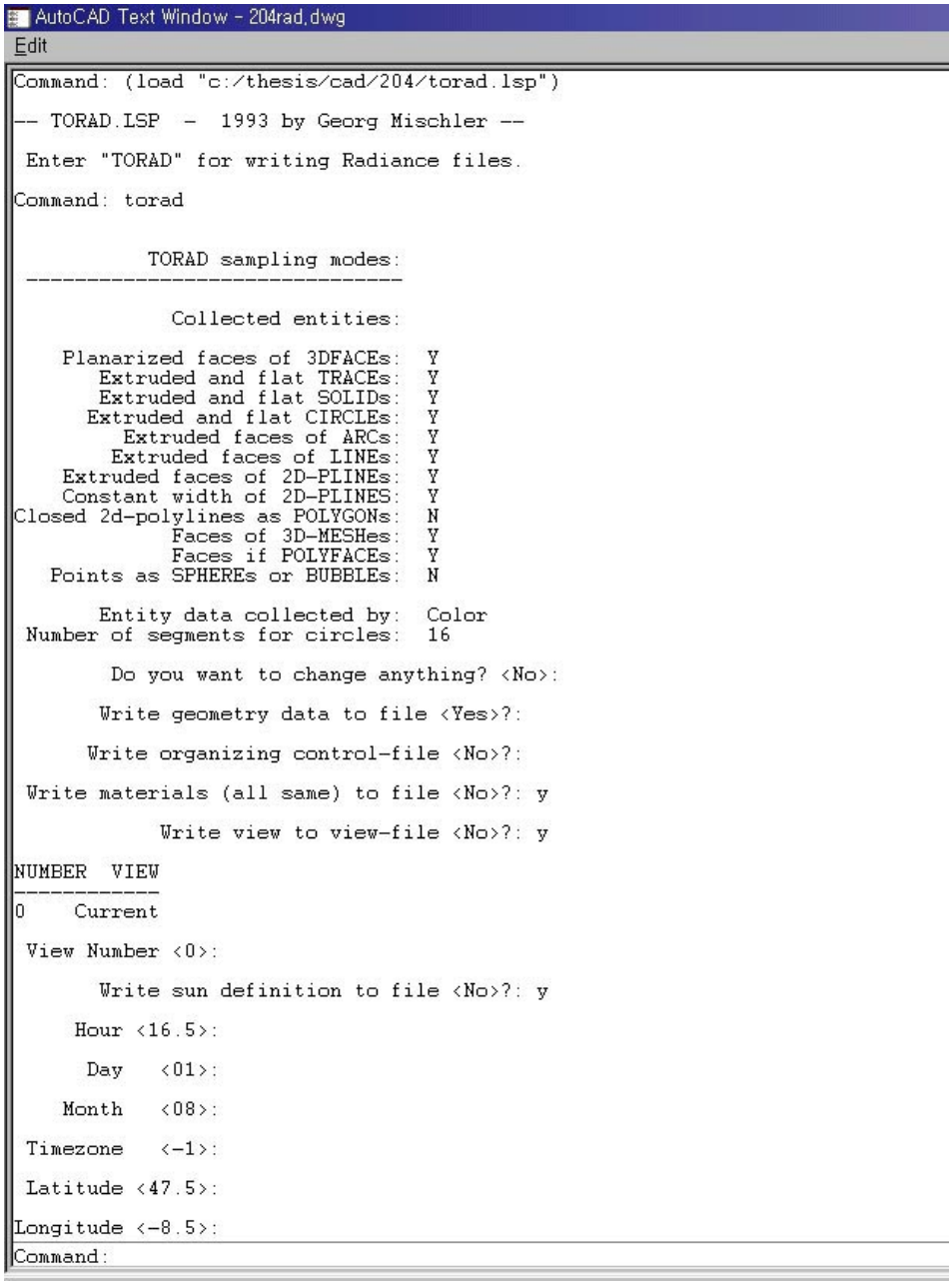
특히 TORAD 명령의 경우에는 재료특성, 관측지점, 외부천공 파일 등을 별도로 작성할 필요없이 위와 같이 AutoCAD 프로그램 내에서 CAD 파일을 통해 일괄적으로 작성할 수 있다.

4) 대상공간의 회전 이동 등의 편집

지금까지 언급한 명령들에 의해 작성된 Scene(*.rad) 파일내의 공간들을 다시 회전시키거나 이동시키고자 할 때는 XFORM 명령을 사용한다. 이 명령은 Scene(*.rad) 파일 내에서 외부 명령의 호출 형식으로 작성하여 실행시킨다. XFORM 명령의 옵션항목 중 –rx, -ry, -rz 항목은 각각 X, Y, Z 축의 회전 각도를 지정하고 –t 항목은 이동시키고자하는 위치의 좌표를 입력한다.

형식) xform –rx degrees –ry degrees –rz degrees \
-t x y z file_name

예) **xform** -rz 90 -t 2 0 0 scene



[그림 3.3] AutoCAD 프로그램내의 TORAD 명령실행 모습

3.3.2. 외부 천공을 모델링하기 위한 명령 및 파일 작성

외부 천공를 모델링하기 위해서는 GENSKY 명령을 사용하는데, skyfunc 를 identifier로 사용하게 된다. 모델링하고자 하는 공간의 수평면조도 E_h에 의해 천공을 계산하게 되는데, 태양복사의 Z축 값인 R_Z를 GENSKY 명령의 –b 옵션의 값으로 입력한다.

CIE overcast sky: $$R_Z = \frac{9}{7} \cdot \frac{E_h}{\pi \cdot 179}$$ **(식 3.3)**

Uniform overcast sky: $$R_Z = \frac{E_h}{\pi \cdot 179}$$ **(식 3.4)**

천공과 지면의 색에 대한 표현은 skyfunc 함수의 복사값에 의해 조절되는데, 3.1 절에서 설명한 것과 같이 R = 1[W/m^{2}sr]이 되는 Rr, Rg, Rb의 값으로 조절한다. 또한 이들 값에 따라 휘도 L[cd/m^2]값이 정해지는데, 휘도값을 정하는 또 다른 변수인 Crn, Cgn, Cbn값은 RADIANCE 프로그램의 계산파일 중 rgb.cal 파일에 의해 계산된다.

$$R_r = \frac{Crn \cdot L}{179} \qquad R_g = \frac{Cgn \cdot L}{179} \qquad R_b = \frac{Cbn \cdot L}{179}$$ **(식 3.5)**

예)

```
# CIE overcast sky (Horizontal illuminance: 10000lux)
# Ground reflectance: 0.1
! gensky  1  1  1  -c  -b  22.8634396  -g  0.1

skyfunc  glow  sky_glow
0
0
4  1  1  1  0
```

```
skyfunc_glow   glow   sky
0
0
4   1   1   1   180

skyfunc   glow   ground_glow
0
0
4   1   1   1   0
ground_glow   glow   ground
0
0
4   1   1   -1   180
```

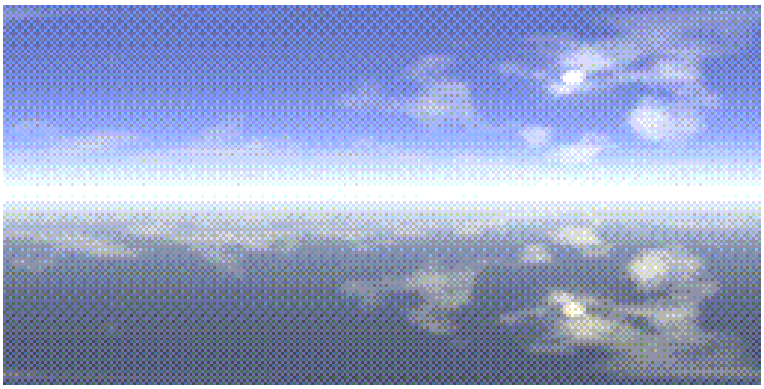
천공과 지면의 입력줄에서 마지막값인 180은 각각 180도로 시뮬레이션하는 것을 의미한다.

청천공의 경우에는 gensky 명령에 모델링하고자 하는 시각과 대상공간의 지리학적 특성을 입력하여, 실행한다.

예) **gensky** 7 1 12 -a 37.97 -o –23.50 -m -30 +s

월 일 시 위도 경도 표준시 태양의 유무

[그림 3.4]은 RADIANCE 프로그램에서 천공과 지면을 각각 180도로 지정하여 시뮬레이션한 그림을 보인 것이다.



[그림 3.4] RADIANCE 프로그램에서 모델링한 정오 천공의 예

3.3.3. 이미지 파일을 생성하기 위한 명령

1) OCONV

이제까지의 과정들을 통하여 작성된 Scene 파일(*.rad)과 Sky 파일, 재료 파일 등을 모두 통합하여 이미지 파일을 생성하기 위한 전단계인 octree 파일(*.oct)을 생성하게 된다. 서두에 언급했듯이 octree 파일은 OCONV 명령에 의해 생성되는데, 명령 형식은 다음과 같다. 각각의 파일 이름들을 사용자가 임의로 정한다.

예) **oconv** scene.rad sky.rad material.rad > scene.oct

2) RVIEW

작성된 octree 파일(*.oct)으로 관측점(X, Y, Z)과 관측방향(dx, dy, dz의 벡터값)을 지정하기 위해 RVIEW 명령을 사용한다 또한 –vh와 –vv 항목을 사용하여 수평각도와 수직각도를 결정한다. RVIEW 명령은 화면에 시뮬레이션된 대상물을 나타내낸 명령으로 이미지 파일을 생성하기 위해 적합한 지점을 선택하고, 시뮬레이션된 상태를 점검하기 위해 사용된다.

형식) rview -vp x y z -vd dx dy dz -vh vh -vv vv \ *.oct

예) **rview** -vp 100 100 10 –vd 1 1 -1 -vh 10 -vv 15 \ scene.oct

3) PFILT

rview 명령만을 사용할 경우 이미지가 모두 백색으로 나타나게 되는데, 이를 위해 대상공간의 광원에 의한 휘도를 나타내주는 노출(exposure)을 준다. 천공의 휘도가 약 10000[cd/m^2]일 경우 최고 휘도는 약 100[cd/m^2] 정도의 값을 갖게 된다. 노출 항목은 이러한 휘도를 RADIANCE 프로그램의 이미지에 적용시키는 것으로 휘도 L 값과 공간을 요소들로 나눈 각 요소에 도달하는 최대 휘도 P값에 의해 결정된다.

$$P = e \cdot \frac{L}{179} \qquad \textbf{(식 3.6)}$$

P는 0 ~ 1 사이의 범위에서 값을 갖지며, 위의 식은 결국, 아래의 식으로 나타낼 수 있다.

$$e = \frac{179}{L_{\max}} \qquad \textbf{(식 3.7)}$$

노출값 e의 초기값은 1로 설정되어 있다. 이러한 노출은 아래와 같은 형식으로 PFILT 명령을 통하여 적용시키게 된다.

형식) pfilt -e exposure_value *.pic > *.pic

예) **pfilt** -e 2 scene.pic > scene1.pic

4) RPICT

위의 과정을 통해 결정된 항목들로 이미지 파일을 생성하는데, RADIANCE 프로그램의 이미지파일은 RPICT 명령을 통해 octree 파일(*.oct)을 *.pic 파일로 계산, 생성된다. –x와 –y 항목은 생성되는 그림의 크기를 결정하는데, pixel 수의 단위로 입력한다.

형식) rpict -vp x y z -vd dx dy dz -vh vh -vv vv \
–x X -y Y *.oct > *.pic

예) **rpict** -vp 100 100 10 –vd 1 1 -1 -vh 10 -vv 15 \
–x 1024 -y 1024 scene.oct > scene.pic

또한 위의 PFILT 명령은 '|' 기호를 통하여 RPICT 명령과 함께 표현할 수 있다.

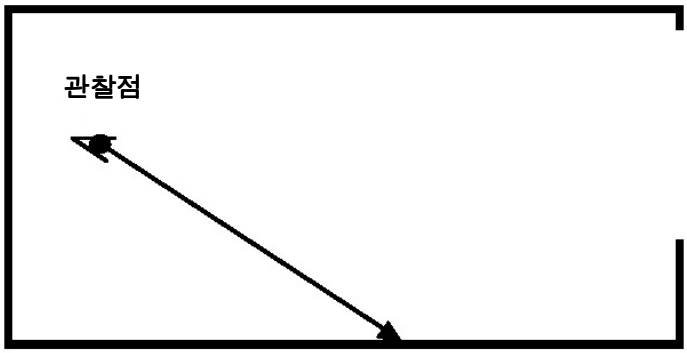
예) **rpict** -vp 100 100 10 –vd 1 1 -1 -vh 10 -vv 15 \
–x 1024 -y 1024 scene.oct | **pfilt** -e 2 > scene.pic

RPICT 명령에는 위의 옵션 항목들 외에도 광선이 반사되고 확산되는 개수 및 광선의 수를 지정할 수 있는데, 이것은 RADIANCE 프로그램의 역광선추적기법(Backwards Ray-tracing Method)의 개념이 사용된 것을 확인할 수 있는 부분이다.

이러한 옵션 항목 중 중요한 항목으로 –ab(ambient bounce), -ad(ambient division), -av(ambient value)의 세가지를 들 수 있는데, 아래의 그림들을 통해 역광선추적기법(Backwards Ray-tracing Method)의 개념과 각 항목들의 적용형태를 살펴보았다.

가시화된 이미지를 나타내는 명령인 RPICT 명령에서 각 항목의 기본값은 0으로 지정되어있다. –ab는 상호 반사되는 수를 결정하고, -ab는 광선이 도달된 면에서의 복사 색분포를 결정하며, -ad는 2차 반사되어 나누어지는 광선의 수를 결정한다.

[그림 3.5]는 역광선추적 개념에 따라 광원에서 광선이 발생하는 것이 아니라 재실자의 눈에서 최초의 광선이 발생하는 것을 나타내고 있다.



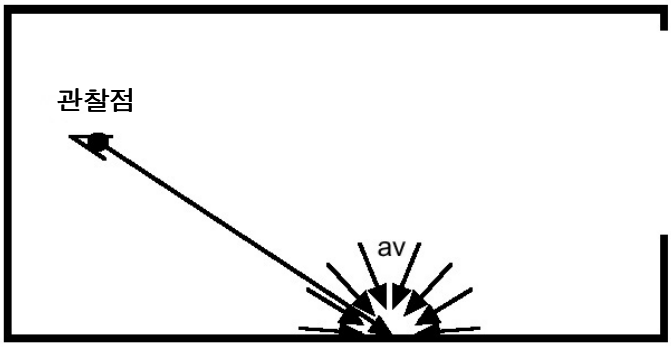
[그림 3.5] 재실자의 눈에서 발생한 최초의 광선

그림과 같은 상태에서 –ab는 0값을, -av는 0(r) 0(g) 0(b)값을 가지게 된다. –av 항목에서의 3가지 복사인 Rr, Rg, Rb 값은 이들 세가지 값의 총합인 R_{amb}에 의해 결정되는데, R_{amb} 값은 대상공간의 작업면 조도[lux]값인 E_{amb}값에 의해 계산된다.

$$R_{amb} = \frac{E_{amb}}{179 \cdot \pi} \qquad \textbf{(식 3.8)}$$

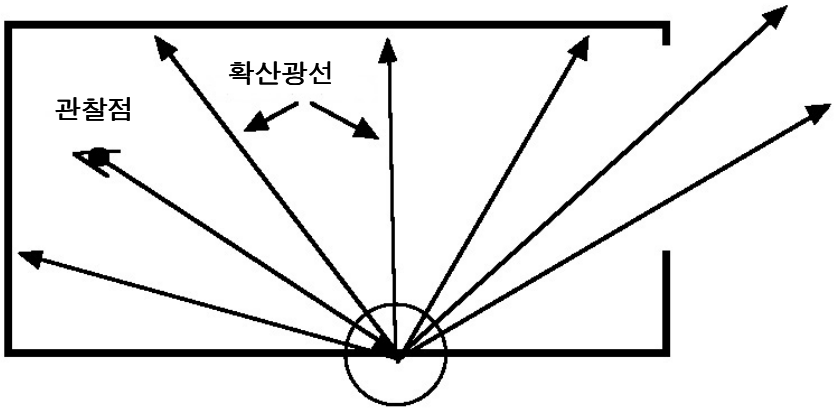
[그림 3.6]은 –av 항목에 의한 값에 의해 결정된 복사의 분포를 보여주는 것이다.

그림과 같은 상태는 –ab는 0값을 –av는 일례로 작업면 조도가 500lux일 경우 .889(r), .889(g), .889(b)의 값을 가지게 된다.



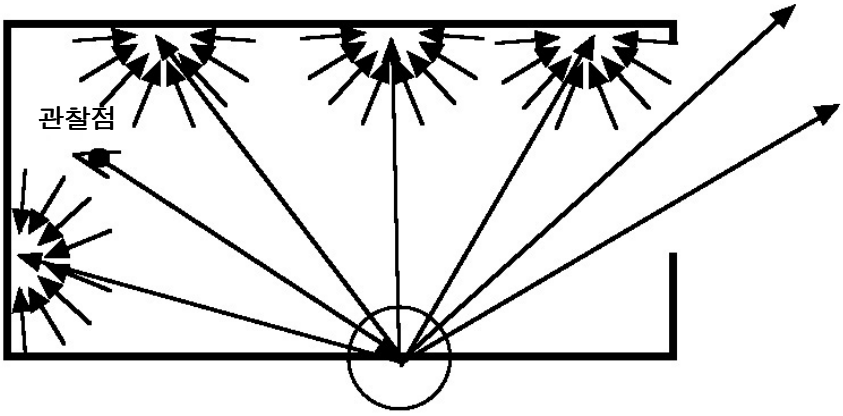
[그림 3.6] 대상공간의 표면에 도달한 광선의 복사분포 결정

[그림 3.7]은 –ad 항목에 의해 나누어진 광선들이 2차적으로 반사되는 것을 보여준다. 표면에 도달한 광선이 지정된 값에 따라 나누어져 다른 표면으로 2차 반사되고 있다.



[그림 3.7] 확산되는 광선의 수에 따라 반사

[그림 3.8]은 위 [그림 3.5], [그림 3.6], [그림 3.7]의 과정을 모두 적용하였을 때의 개념을 보인 것이다.



[그림 3.8] 반사된 광선이 2차적으로 확산될 값 결정

RADIANCE 프로그램은 위의 과정들에 통해 대상공간에 공급되는 자연채광, 혹은 인공조명들의 광원을 모델링하게 된다.

MKILLUM 명령은 이러한 개념을 이용하여 재실자의 눈에서 광선을 발생시켜 조도를 예측하는 명령이다.

예) mkillum -ab 2 -ad 256 scene.oct > illum.rad

5) FALSECOLOR

RADIANCE 프로그램에서는 일반적인 가시화 이미지뿐만아니라 같은 조도, 혹은 휘도 값을 선으로 나타내는 등고선 형태의 이미지도 생성할 수 있다. [표 3.6]는 FALSECOLOR 명령에 사용되는 옵션인 각 항목들의 내용을 정리한 것이다.

[표 3.6] FALSECOLOR 명령의 항목

항 목	내 용
-I	RPICT 명령을 사용하여 이미지 파일 생성시 같은 -i 항목을 사용하여 생성한 이미지 파일(*.pic) – 대상공간의 표면에 입사되는 복사를 계산하는 것으로 창문등의 투과되는 재료는 계산되지 않는다.
-p	RPICT 명령으로 생성된 일반적인 이미지 파일(*.pic)
-ip	휘도분포를 모델링하기 위해 일반적인 이미지 파일만을 불러들이는 항목
-cl	조도(휘도)분포를 colorline로 표현
-cb	조도(휘도)분포를 colorband로 표현
-s	조도(휘도)분포의 범위 결정
-l	이미지에 나타나는 값의 label – 휘도의 단위인 nit[cd/m^2]가 기본값으로 설정되어 있다.
-n	각 값의 분포 범위를 나누는 수

이러한 이미지는 아래와 같은 형식에 따라 FALSECOLOR 명령을 이용하여 생성된다.

조도분포형식) falsecolor -i input -p picture -cl -s scale \
-l label -n devide_number > *.pic

휘도분포형식) falsecolor -ip picture -cl -s scale -l label \
-n devide_number > *.pic

예) **falsecolor** -i scene1.pic -p scene2.pic –cl \
-s 1000 -n 10 -l LUX > scene_false.pic

6) 이미지 출력 및 변환 명령

앞에서 생성된 이미지 파일들을 확인하기 위해서는 UNIX 환경에서 ximage 명령을 사용한다.

예) ximage –e auto scene.pic

이러한 이미지 파일(*.pic 파일)은 RADIANCE 프로그램의 환경인 UNIX workstion의 환경에 적합한 것으로 PC의 window 환경에서 나타내기 위해서는 일련의 변환과정을 거쳐야 한다. 이미지 파일(*.pic 파일)은 아래의 형식에 따라 각각 TIFF 파일과 GIF 파일로 변환될 수 있다.

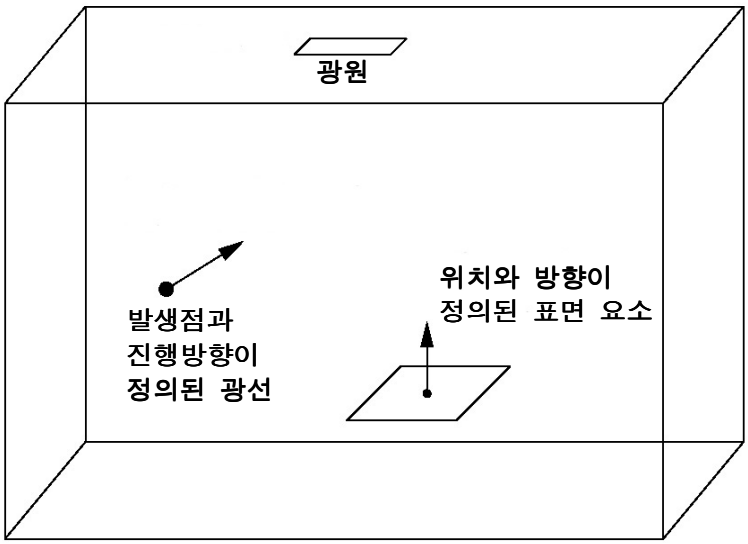
형식) ra_tiff pic_filename tiff_filename
ra_gif pic_filename gif_filename

예) **ra_gif** scene.pic scene.gif
ra_tiff -r material.tif material.pic

특히 RA_TIFF 명령은 –r 항목을 사용하여 PC의 window 환경에서의 이미지 파일을 UNIX workstion의 환경에 적합한 이미지 파일로 변환하는 역변환 과정에도 사용될 수 있다.

3.3.4. 대상공간의 광선 추적 및 조도 계산을 위한 명령

RTRACE 명령은 Octree(*.oct) 파일에 의해 임의의 방향에서 임의의 지점으로부터의 광선을 추적할 수 있는 명령이다. 이것은 특히 수치적인 값(예, 조도)으로 구해질 수 있으며, [그림 3.9]에서와 같은 지점 혹은 표면으로부터의 하나 혹은 다수의 광선들을 모두 추적할 수 있다.



[그림 3.9] 대상공간 내에 발생하는 광선

RTRACE 명령을 실행하기 위해 먼저 이미지 파일의 각 pixel을 계산하기 위해 방향벡터가 계산되어야 한다. 이러한 값들은 RADIANCE 프로그램 내의 pcyl.cal 이라는 계산 파일에 의해 결정된다. RADIANCE 프로그램의 이미지는 수평 pixel의 수(XD)와 수직 pixel의 수(YD)에 의해 결정되며, 이미지의 길이는 AM이라는 매개변수에 의해 고도의 개념으로 결정된다.(예, AM = -10)

- 이미지 파일에서의 광선계산

  형식) cnt YD XD |

  rcalc -f pcyl.cal -e XD = XD; YD = YD; AM = AM |

```
rtrace -x XD -y YD -fac *.oct > *.pic
```

예)

```
cnt 640 480 |
rcalc -e 'xr:640; yr:480' -f unusual_view.cal |
rtrace -x 640 -y 480 -fac scene.oct > unusual.pic
```

- XIMAGE 명령을 사용하여 결정된 위치에서의 조도값 계산

형식)

```
ximge *.pic |
rtrace -h -x x -i *.oct |
rcalc -e 'function'
```

예)

```
ximge scene.pic |
rtrace -h -x 1 -i scene.oct |
rcalc -e '$1=47.4*$1+120*$2+11.6*$3' > view.pic
```

RCALC 명령은 각 파일으로부터 주어진 문법과 관계 정보 들을 계산, 변환하여 주는 것으로 RCALC 명령에서 ' $ '는 문자열이 주어졌을 때 사용하며, ' @ '는 파일이 주어졌을 때 사용한다.