

---

# 3D 애니메이션 제작 공정에 있어서 효율적인 이미지 시퀀스 포맷

## Effective Image Sequence Format in 3D Animation Production Pipeline

---

김 호

중앙대학교 첨단영상대학원

Ho Kim(hkeffort@nate.com)

---

### 요약

3D 애니메이션 작업의 결과물을 출력하는데 있어서 동영상 파일로의 렌더링이 가능하지만 대부분의 프로덕션에서는 Image Sequence 렌더링 방식을 택하고 있다. 이러한 Image Sequence 렌더링 방식은 최종 영상물 제작을 위한 합성에 있어서 중요한 과정이다. 이러한 Image Sequence 렌더링 방식에는 다양한 Format의 이미지 형태들이 사용되는데 그 대표적인 것으로서 TGA format을 들 수가 있다. 하지만 TGA format 방식의 이미지가 이러한 과정에서 최적인가에 대한 의문점이 제기되어지고, 3D 프로그램마다 고유의 이미지 format을 가지고 있다는 점에서 이에 대한 연구가 필요하다. 본 연구에서는 Alias사의 Maya를 중심으로 이러한 Image Sequence format에 대한 압축률, 선명도, 합성 시 필요한 Alpha Channel의 보존성, Z-Depth에 관한 Information에 관하여 연구하고자 한다. 이 연구의 결과는 3D 애니메이션을 Pipeline에 Image Sequence format에 대한 가이드 라인을 제공해주는데 목적이 있다.

■ 중심어 : | 3D 애니메이션 | Image Sequence Format |

### Abstract

In 3D animation rendering process, Although we can render the output as a movie file format, most productions use image sequences in their rendering pipelines. This Image Sequence rendering process is extremely important step in final compositing in movie industries. Although there are various type of making image rendering processes, TGA format is one of most widely used bitmap file formats using in industries. People may ask TGA format is most suitable for in any case. As we know 3D softwares have their own image formats. so we need to testify on this.

In this paper, we are going to focus on Alias' 3D package software called MAYA which we will analyze of compressing image sequence, Image quality, supporting Alpha channels in compositing, and Z-depth information. The purpose of this paper is providing to 3D Pipeline as a guideline about effective image sequence format.

■ keyword : | 3D Animation | Image Sequence Format |

---

## 1. 서론

### 1. 연구 배경 및 목적

디지털 환경 속에서의 이미지 제작에 있어서 효율성을 높일 수 있는 적용방법을 연구하고자 한다. 각 프로그램 개발 회사 및 연구소에서 고유의 압축 방식을 개발한 이미지 파일들이 다른 프로그램들과의 호환성을 높이며, 저장 공간을 절약과 동시에 인터넷상에서 파일 공유의 시간을 단축 시켜준다. 일반적으로, 영상 편집을 하기 위해서, 640x480이상의 해상도를 가지고 있어야 하며 압축 비율은 6:1 이상을 권장하고 있다. 하지만 실제로 편집된 동영상을 TV에서 보려면 720x480 이상인 NTSC 표준 규격을 지원해야하고, 방송용으로 사용되려면 720x486인 D1 규격과 2:1 이상의 압축률을 지원해야만 한다는 것이 규격화되어있다. (유럽에서는 PAL 방식으로 규격화 되어있다.) 일반적으로 많은 사용자들은 자신의 영상을 어떤 형식의 파일로 사용하는 것이 적절하고 효과적인지 정확하게 모르고 대부분 이미지를 아웃풋 하는 경우가 많다. 이러한 과정에서 파일크기와 이미지 품질을 추측하여 사용할 뿐 여러 가지 다른 이미지 포맷으로 렌더링을 했을 때 그 차이에 대한 자료를 명확하게 제시해주는 근거는 없었다. 일반적으로 정의할 수 있는 가이드라인은 가장 이미지의 압축이 효과적인과 동시에 최상의 이미지 품질을 나타내는 형식의 포맷일 것이다. 현재 대부분의 사용자들의 이미지 압축에 대한 견해는 이미지 사이즈 크기를 측정 하는 것으로 간주되어있지만, 이미지 품질에 대한 문제는 조금 더 주관적인 관점에서 볼 수가 있을 것이다. 일반적으로 압축방식은 크게 Lossy(불가역) 압축방식과 Lossless(무손실) 압축방식으로 크게 나누어진다. Lossy(불가역) 압축방식은 이미지를 반복해서 저장할 경우에 이미지 해상도가 현저하게 떨어지는 방식이다. 이와 반대로 Lossless(무손실) 압축방식은 이미지를 반복해서 저장할 하더라도, 이미지 해상도를 그대로 현상유지에 시켜주는 압축방식이다.

본 논문에서는 일반적으로 많이 사용되는 Image Sequence 파일 포맷을 사용해 그 차이점을 분석하여 일반 사용자들이 3D 애니메이션을 제작 파이프라인에 있어서 여러 논란이 되고 있는 압축 포맷의 선택 문제에 대

한 해결방안을 제시하고자 한다.

### 2. 파일의 종류와 특징

이미지는 텍스트 다음으로 일찍이 사용되어 온 미디어이다. 이미지를 저장하기 위해서는 매우 다양한 파일 포맷이 있다. 다양하게 파일 포맷이 사용되는 것은 각각의 포맷이 각각 다른 용도에 적합하도록 만들어졌기 때문이다. 예를 들어 벡터 그래픽 정보만 저장을 필요로 하는 프로그램에서는 불필요한 픽셀의 정보를 저장을 하지 않아도 되며, 흑백 영역의 정보만 저장을 필요로 하는 프로그램에는 색상 정보 저장은 불필요하다. 이렇게 새로 필요로 의해 개발된 파일 포맷들은 초기에 다른 프로그램들과의 호환성 문제가 발생을 하였지만, 이 문제점을 해결하기 위해 다양한 프로그램들은 여러 가지 파일 포맷을 읽을 수 있는 기능을 포함하고 있다. 단순히 RGB 정보 및 Alpha 채널을 저장하는 기능으로부터 이미지의 파일 사이즈를 최대한으로 적게 만드는 압축 포맷 등 다양한 방식의 파일 종류가 있는 것을 알 수 있다. 사용자가 적은 파일 양을 대상으로 이미지를 저장할 경우 파일 크기가 문제가 되지는 않지만, 수백, 수천 장과 같이 많은 양의 데이터를 저장 할 경우, 이를 처리하고자 하는 데이터 디스크 용량이 문제가 된다. 다른 한편으로는 웹 상에서 이미지를 전송할 때에도 파일 포맷의 압축 유형에 따라 전송 속도의 문제점이 발생 할 수 있다. 일반적으로 파일 포맷들은 각각 장점과 단점을 가지고 있으며 사용자들은 파일 종류의 특성을 잘 이해하고 그들의 작업에 걸 맞는 파일 종류를 선택 하여야 한다. 3D 소프트웨어 중 현재 CG 작업에 가장 많이 사용하고 있는 Autodesk사의 3D 프로그램인 Maya는 다음과 같은 이미지 포맷 종류를 지원을 한다. (als, avi, cin, dds, eps, gif, jpeg, iff, png, yuv, rla, sgi, pic, tga, tif, bmp) 아래 [그림 1]은 Maya에서 지원되는 파일에 관한 Compression 방식, Image Sequence, Alpha Channel, Z-Depth 정보에 관한 도표이다.

Format	Extension	Compression	Alpha Channel	Image Sequence	Z-Depth Information
Adobe Illustrator	.ai	Lossless	None	None	None
Alias PIX	.ais	Lossless	Support	Support	Support
AVI	.avi	Lossy	None	Support	None
Cineon	.cin	Uncompress	None	Support	None
Encapsulated PostScript	.eps	Lossy	None	None	Support
GIF	.gif	Lossless	None	Support	None
JPEG	.jpg	Lossy	None	Support	Support
Macromedia Flash	.swf	Lossless	None	None	None
Maya IFF	.iff	Lossless	Support	Support	Support
Maya16 IFF	.iff	Lossless	Support	Support	Support
MacPaint	.pntg	Lossless	None	None	None
Photoshop	.ps	Lossless	Support	None	None
PNG	.png	Lossless	Support	Support	None
Quantel	.yuv	Uncompress	Support	Support	None
QuickDraw	.pict	Lossless	None	None	None
QuickTime Image	.mov	Lossless	None	None	None
RLA	.rla	Lossless	Support	Support	Support
Scalable Vector Graphics	.svg	Lossless	None	None	None
SGI	.sgi	Lossless	Support	Support	Support
SGI16	.sgi	Lossless	Support	Support	Support
SGI Movie		Lossless	None	None	None
SoftImage	.pic	Lossless	Support	Support	Support
Targa	.tga	Lossless	Support	Support	Support
Tiff	.tif	Lossless	Support	Support	Support
Windows Bitmap	.bmp	Uncompress Lossless	None	Support	Support

그림 1. Maya에서 지원되는 파일 포맷

### 3. 연구방법 및 범위

본 연구 방법은 디자인 진흥원에 있는 CG Technical Director 재교육 수강중인 3D 유저 총 30명 중 현 실무에서 Maya 프로그램을 사용하는 20명을 선정해 최종적으로 렌더링 하여 아웃풋 할 당시, 가장 많이 사용하는 Image Sequence 파일 종류를 조사해본 결과 tga, iff, tiff, rla, sgi, jpeg 파일 중 tga 파일의 사용빈도가 가장 높은 것으로 나타났다. 본 연구에서 설문조사를 통해 사용 빈도가 낮은 이미지 포맷은 본 연구에서 제외하기로 하였다.

본 연구 방법은, Autodesk사의 3D 프로그램인 Maya 를 사용해, 동일한 애니메이션 파일을 설문조사를 통해 3D 유저들에게 사용 빈도가 높은 이미지 타입으로 같은 Maya 파일인 maya Binary(\*.mb)파일을 tga, iff, tiff, rla, sgi, jpeg Image Sequence 파일로 Batch Rendering을 하여, 렌더링 된 Image Sequence의 파일 사이즈 및 시간 소요를 비교 분석을 하였다. 최종으로 렌더링이 된 이미지 사이즈는 NTSC D1 규격인 720x486 사이즈, 30 fps(frame per second)로 설정을 하고, Image Sequence 파일에 Z-Depth 정보가 있는 경우와 없는 경우의 파일 사이즈를 비교하여 어떤 Image Sequence 파일 포맷이 3D 애니메이션 제작 파이프라인에 더 적합한가에 대해서 연구하고자 한다.

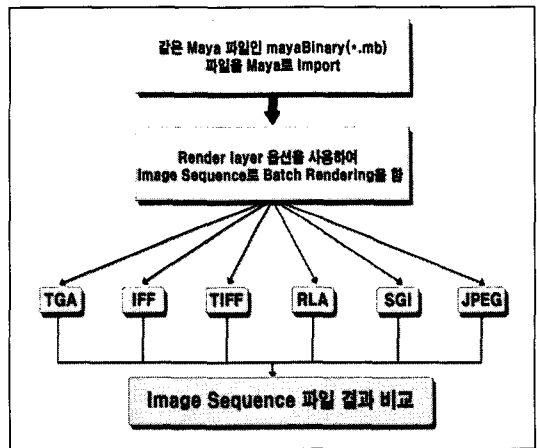


그림 3. Image Sequence 비교 Process

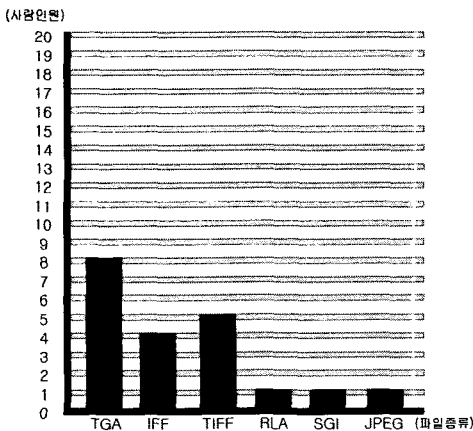


그림 2. 파일 선호도 그래프

Component	PC
CPU	Intel Dual Xeon3.4GHz, 800MHz
Motherboard	Asus NCT-D Motherboard
RAM	4.0 GB DDR2 PC2-3200 (ECC-Registered)
Hard Drive	250GB, 7200RPM, Serial ATA(SATA)
Video Card	PNY Quadro FX 3500 256MB GDDR3
OS System	Windows XP2

그림 4. Image Sequence 렌더링에 사용된 PC 사양

## II. 본론

### 1. 합성

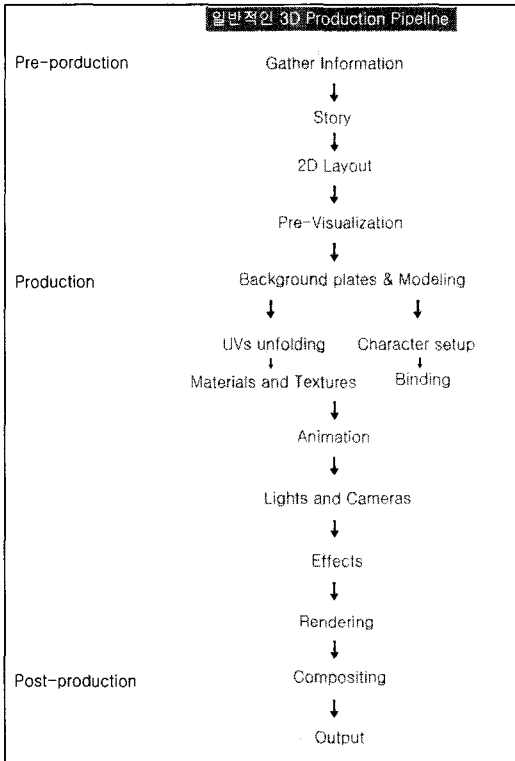


그림 5. 일반적인 3D Production Pipeline

[그림 5]에서 보듯이 Image Sequence를 만드는 단계는 Post Production 단계 중 렌더링에 포함이 된다. 보통 많은 제작 스튜디오에서 렌더링을 할 때, Image Sequence 방식을 채택하는 이유는 최종적으로 영상의 이미지 품질을 높이기 위하여 합성과정에 요구 되는 과정이다. 요즘 같은 디지털 시대에서 합성은 다양한 편집 프로그램을 사용하여 여러 개의 동영상 이미지들 또는 정지된 이미지들을 다시 재조합하여 하나의 Image Sequence 파일을 만들어낸다. 1984년에 처음으로 루카스 필름에서 일하는 두 명의 과학자가 합성이란 용어를 형식적으로 정의하였다. “경험상 우리는 컴파일 시간을 절약하기 위해 긴 소스 코드를 짧은 몇 개의 모듈로 나누었다. 어떤 루틴에서 실수하더라도 그에 해당하는 모듈만을 다시 컴파일하면 되기 때문에 프로그램 전체를 다

시 불러들이는 데 걸리는 시간이 짧다. 비슷하게, 채색이나 디자인에서 작은 실수를 하더라도 전체 이미지를 다시 컴파일 할 필요가 없다[1].” 이렇게 독립적인 하나의 Image Sequence 파일로 만들어 작업하는 주된 이유는 제작 시간 단축과 부가적으로 합성 할 경우 각 레이어의 색 조정 및 효과를 줄 수 있기 때문이다. Maya에서 지원하는 Render layer 옵션을 사용하여 각각의 오브젝트의 독립된 레이어를 하나의 Image Sequence 파일로 렌더링을 해 줄 수 있다. 예로, Maya에서는 카메라 옵션 중에 Depth of field 옵션을 사용하여 카메라의 물체의 초점을 조절을 할 수 있는 기능이 있다. 그렇지만 아쉽게도 아직 Maya에서 Depth of field 계산과정에 있어서 렌더링의 시간 요소가 너무 오래 걸리는 단점이 있다. 그러기 때문에 아래의 [그림 6]와 같이 렌더링이 된 Image Sequence를, 합성 프로그램으로 가져가 합성을 하여 특정 레이어에 초점을 흐리게 해줄 경우, 수동적으로 이미지를 제어하는 방식이 훨씬 시각적으로 보다 정확하게 초점을 조절 해줄 수 있다.

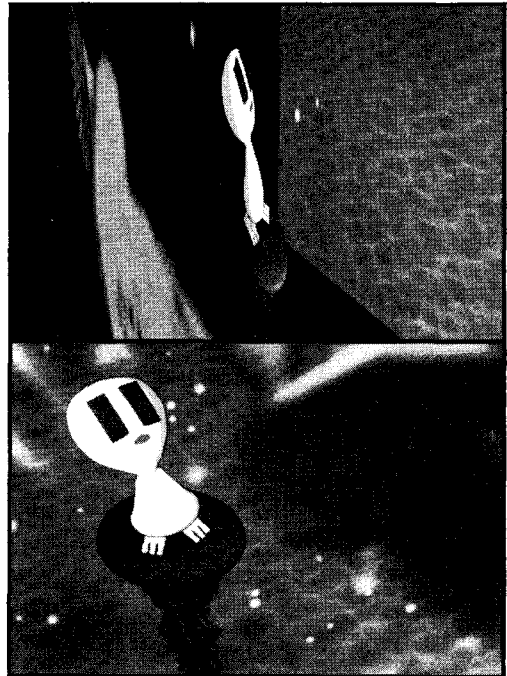


그림 6. 레이어의 구성

## 2. 압축 방식

많은 사람들이 파일 타입을 결정할 당시, 가장 최우선적으로 선택하는 주된 요소가 파일에 어떠한 이미지 콘텐츠가 들어가는지 여부에 따라 어떤 파일 포맷으로 저장할지 결정하게 된다. 여기에 압축률이 높음과 동시에 높은 이미지 품질이 가장 주된 목적이다. 이미지를 압축하는 방식을 잠시 살펴보면, 대부분의 파일 포맷을 보면 각 채널당 8Bit의 색상 정보를 지원한다. 이는 “24bit 이미지” (빨강채널[8bit], 초록채널[8bit], 파랑채널[8bit])로 나타나며, 파일에 따라 각 채널당 16Bit, 32Bit 채널을 지원해 주는 파일도 있지만, 해당 편집 프로그램에서 각 채널당 16Bit, 32Bit를 지원해 주지 않을 경우 최종적으로 8Bit의 색상 채널로 변환을 시켜 주어야 하기 때문에 파일 타입의 선택은 유의해야 할 요소 중 하나이다. 24bit 이미지는 Alpha 채널을 소유하고 있지 않지만, 32bit 이미지는 Alpha 채널을 소유하고 있어 일반적으로 이미지 합성이 필요한 경우에 사용되어진다.

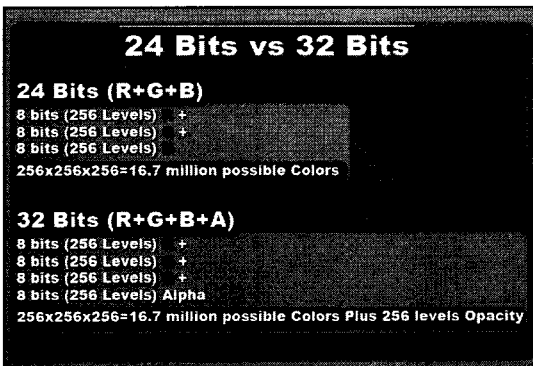


그림 7. 24Bits & 32Bits 이미지 비교

Jpeg 같은 Lossy(불가역) 압축 방식은 손실 압축 방법으로 압축률은 매우 좋지만, 대신 한번 압축하면 압축 이전 상태로 복원이 불가능하다는 단점이 있다. Jpeg로 압축하는 과정에서는 사용자가 압축률을 임의로 정해줄 수 있는데, 압축률이 낮을수록 화질이 좋아지고, 압축률이 높으면 화질은 떨어진다. Alpha 채널을 Jpeg 파일 같은 경우에는 지원을 하지 않아서 이미지를 합성할 경우 Matte, Mask, Keying 과 같은 다른 방법으로 합성방법이 필요로 한다. 이 밖에 Lossless(무손실) 압축 방식과

일 중, [그림 8]와 같이 RLE (Run-Length Encoding) 압축 방식을 지원하는 파일 포맷들이 있다. RLE 압축 방식에 있어서도 이미지의 형태에 따라서 압축률이 달라지는 것을 볼 수 있다. RLE의 기본 원리는 같은 색상이나, 중복되는 데이터가 많을 경우, 반복되는 횟수 정보를 부호화 시켜서, 데이터의 코드가 연속되어 있는 그래픽 이미지를 효율적으로 압축을 해준다. tiff, tga 파일 같은 경우에는 RLE 방식을 지원해 주어서 3D 애니메이션과 같이 단일 색상의 백그라운드 가 있는 경우에는 보다 압축률이 효과적으로 적용이 된다. 이미지를 저장할 때는 어떤 압축 기법을 사용하는 이미지의 품질과 압축률은 서로 반비례하게 되어있다. 압축 효율이 높으면 이미지의 질은 낮아지고, 반대로 압축률이 낮으면 이미지 품질은 좋아지게 된다. 사용자들은 본인의 작업에 맞는 효율적인 이미지 압축률을 선택하여 이미지를 저장하는 것이 현명한 방법이다.

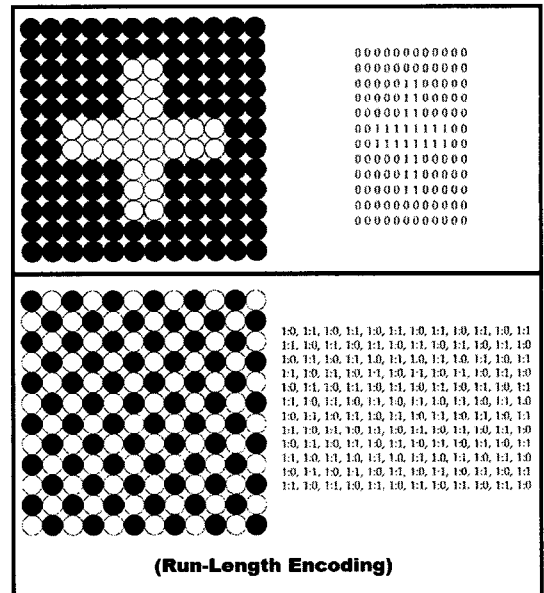


그림 8. Run-Length Encoding 방식[4]

## 3. 파일 렌더링 시간 & 파일 사이즈 비교

[그림 9]와 [그림 10]은 Autodesk회사 제품인 Maya를 사용하여 같은 파일을 7초 (0 frame ~ 210 frame) 동안 Z-depth 정보가 있는 Image Sequence파일과 Z-depth의

정보가 없는 Image Sequence파일로 렌더링을 하였을 때 렌더링의 총 소요 시간과 이미지 파일 사이즈에 대한 결과이다.

File Format	Single File Size	Total File Size	Rendering Time
TGA	1.368KB	281MB	1시간 58분
IFF	142KB	28.9MB	1시간 5분
TIFF	58KB	11.6MB	1시간 48분
RLA	115KB	23.3MB	2시간 11분
SGI	126KB	26.0MB	1시간 4분
JPEG	11KB	2.15MB	1시간 40분

그림 9. Z-depth와 Alpha Channel이 없는 경우 Image Sequence 파일 비교

File Format	Single File Size+(Z-depth Size)	Total File Size	Rendering Time
TGA	1.368KB + (185KB)	333MB	2시간 01분
IFF	273KB	58.2MB	1시간 5분
TIFF	241KB + (185KB)	51.7MB	1시간 48분
RLA	233KB	49.7MB	2시간 11분
SGI	309KB + (185KB)	66.3MB	1시간 5분
JPEG	194KB + (185KB)	41.7MB	1시간 42분

그림 10. Z-depth와 Alpha Channel이 있는 경우 Image Sequence 파일 비교

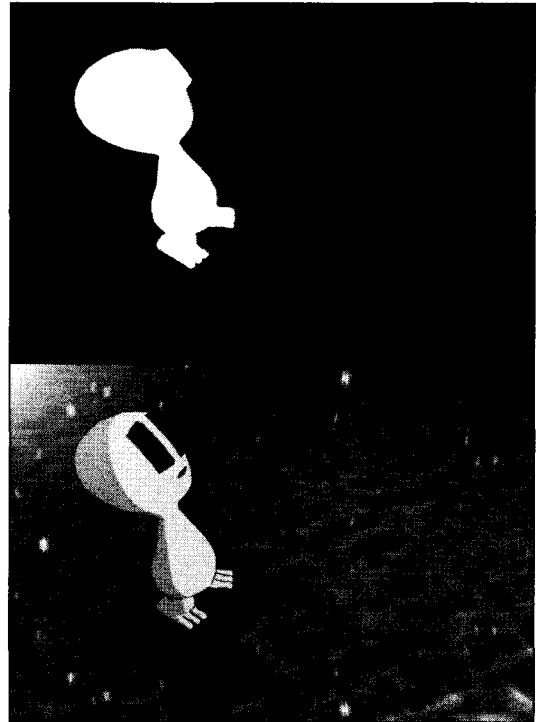
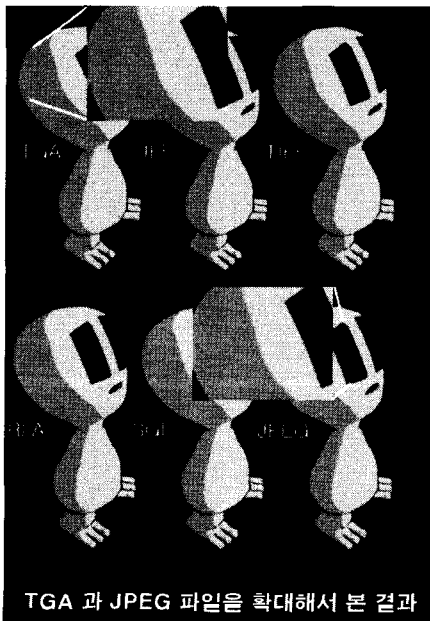


그림 12. Alpha 값으로 합성된 이미지



TGA 과 JPEG 파일을 확대해서 본 결과

그림 11. 이미지 포맷 파일 확대 비교

Z-depth가 없이 Image Sequence를 렌더링을 할 경우, [그림 11]과 같이 tga, iff, tiff, rla, sgi 파일을 확대하여 이미지의 품질을 비교 하였을 때 이미지 품질 면에 있어서 거의 차이가 없지만 Lossy(불가역) 압축 방식인 jpeg 파일인 경우 이미지의 품질이 많이 떨어지는 것을 볼 수 있다. 물론 [그림 11] 같은 이미지와 같이 Alpha 채널이 이미지 합성에 필요할 경우에는 Alpha 채널을 지원하지 않는 jpeg 파일방식은 부적합한 파일이다. 만약 이 문제를 해결하기 위해서는 jpeg 파일을 편집 프로그램으로 불러들여 Keying을 사용하여 배경에 해당하는 검정색을 제거를 해주어야 한다. Keying 효과는 움직이는 요소와 하나의 다른 배경을 경합하기 위해서 사용되는 합성의 방법중 하나로, 일반적으로 컬러가 주어진 배경에 대해 부여된 사물들을 결합 하는데, 이는 보통 Bluescreen, 또는 Greenscreen 으로 불리는 파랑 또는 초록색의 배경이다. 일반적으로 이 두 개의 색상을 많이 사용하는 이유는 사람의 피부에서는 파란색과 초록색을 찾기가 어렵기 때문이다. 사람의 신체와 배경 사이의 콘트라스트를 두드

러지게 주기 위하여 스크린에서 bluescreen 이나 greenscreen을 사용한다. 이 후에 배경색, 또는 Key 컬러는 제거되거나 Keyed out 될 수 있다. 투명한 영역들을 정의 하는 Key 컬러가 없으면, 움직이는 사물과 정교하게 맞아 떨어지는 하나의 매트를 만들어 내는 것이 불가능하지 않다면, 어렵다. [그림 11] 같은 경우에는 배경색이 Black 이므로 Black keying이 필요로 한다. jpeg 파일은 압축률이 다른 이미지 파일 포맷에 비해 파일의 크기가 가장 작은 장점을 가지고 있다. 반대로 tga 방식은 파일 사이즈가 가장 큰 파일로 나온 결과를 확인 할 수 있다. 이미지에서 Alpha 채널이 자동으로 저장되어지기 때문에 따로 Alpha 채널을 만들 필요가 없이 모든 오브젝트를 하나로 묶어 렌더링을 할 경우 [그림 12]과 같은 결과를 볼 수 가 있다. [그림 10]을 보면 Z-depth 정보가 있을 경우, 파일 사이즈를 비교한 표이다. jpeg, tga, tiff, sgi 파일들이 자체적으로는 Z-depth 정보를 지원하지 않지만, 추가적으로 Z-depth 정보를 iff 파일로 저장할 하기 때문에 기본파일 용량에서 새로 생성된 Z-depth 파일에 관한 정보의 파일 사이즈만 증가 하는 것을 볼 수 있다. 그러나 iff, rla 파일은 Z-depth에 관한 정보를 하나의 동일한 파일에 저장을 하므로 Z-depth에 관한 정보를 하나의 이미지를 사용하여 합성할 수 있는 편리함을 제공한다. 파일 사이즈 측면에서는 Z-depth 정보가 없는 경우와 마찬가지로 jpeg 파일이 용량이 가장 적고, tga 파일이 가장 용량이 큰 것을 결과를 통해 볼 수 있다.

### III. 결과 분석

마야에서 지원하는 다양한 Image Sequence 파일 포맷을 사용하여 렌더링을 하였을 때 나온 결과처럼 같은 컴퓨터 사양으로 렌더링을 하였을 때 파일의 렌더링 시간과 이미지의 파일 크기가 각기 다르게 나타나는 것을 실험 결과를 통해 알 수 있다. 설문 조사를 통하여 많은 3D 프로그램 사용자가 tga 이미지 포맷을 선호하지만, Maya에서 Alpha 채널을 사용해 이미지를 합성 할 경우 tiff 파일 포맷이 가장 적은 사이즈로 나와 이미지 합성

할 경우 가장 효율적인 파일인 것을 볼 수 있다. 그리고 Z-depth 정보가 있는 파일인 경우 Alpha 채널을 지원하지 않는 Lossy(불가역) 압축 방식인 jpeg 파일을 제외한 Lossless(무손실) 압축 방식을 지원하는 rla 파일 포맷이 파일 사이즈가 작아 작업 효율성면에 있어서도 효과적인 것으로 나타났다.

### IV. 결론

높은 해상도를 지닌 이미지는 많은 용량의 하드 공간을 필요로 하는 것은 누구나 다 아는 사실이다. 그러므로 3D 애니메이션 제작에 있어서 파일의 압축방식을 선정할 때 이미지 파일 사이즈와 이미지의 품질을 고려해야 한다. 본 연구에서 실행된 결과처럼 Maya에서 다양한 이미지 파일 타입을 지원해주지만, 현재 프로덕션에서 많이 사용되는 이미지 타입은 한정되어 있고 다른 어플리케이션과의 호환성 역시선택에 있어서 중요한 요소 중의 하나이다. 3D 애니메이션 제작에 있어서 파일의 압축률과 화질은 제작환경에 있어서의 경제성과 효율성 및 네트워크 간의 파일 전송 등 제작에 있어서 고려되어야 한다. 3D 애니메이션 제작자들이 압축의 용도에 따라 이미지의 품질과 압축률 중 어느 것을 더 중시할 것인지에 따라 압축 파일을 선택을 하게 될 것이고, 이번 연구는 3D 애니메이션 제작자들이 Image Sequence를 사용해 렌더링을 할 경우 이미지 포맷의 선정에 있어서 효율적인 가이드라인이 될 것이다.

### 참고 문헌

- [1] 레프 마노비치, 서정신, *뉴미디어의 언어*, 생각의 나무, 2004.
- [2] 마샬 맥루한, 박정규, *미디어의 이해*, 커뮤니케이션 북스, 2004.
- [3] 이만재, 이산성, *멀티미디어 교과서*, 안그래픽스, 2002.
- [4] R. Brinkmann, *The Art and Science of Digital*

*Compositing*, Morgan Kaufmann, 1999.

- [5] N. Chapman and J. Chapman, *Digital Multimedia*, Wiley, 2004.
- [6] J. Krasner, *Motion Graphic Design and Fine art animation*, Focal Press, 2004.
- [7] Trish and C. Meyer, *Creator Motion Graphics with After Effects Vol.2: Advanced Techniques*, CMP Books, 2005.
- [7] <http://area.autodesk.com>
- [8] <http://www.highend3d.com>
- [9] <http://www.answers.com>
- [10] <http://en.wikipedia.org>

**저 자 소 개**

김 호(Ho Kim)

중신회원



- 1997년 9월 ~ 2001년 5월 :  
School of Visual Arts, B.F.A.,  
Animation과 졸업
- 2002년 1월 ~ 2003년 12월 : New  
York University, M.S., Digital  
Imaging and Design과 졸업

▪ 2006년 3월 ~ 현재 : 중앙대학교 첨단영상대학원 애  
니메이션 박사과정

<관심분야> : 3D 애니메이션, 모션 그래픽