

애니메이션 작업에 사용되는 이미지 포맷 연구

권동현

초 록

애니메이션 작업에 있어 컴퓨터는 꼭 필요한 도구가 되었다. 그러나 컴퓨터와 그에 따른 소프트웨어들의 특성을 잘 파악하지 못할 경우 노력에 상응하는 결과물을 만들지 못할 수도 있다. 이미지 포맷의 부정확한 이해에서 이러한 경우가 발생하기도 한다. 대다수 작업자들은 제작과정 속에서 이미지 포맷을 습관적으로 선택하는 경우가 많으나, 이미지 포맷은 분명한 기능적 차이를 가지고 있으며 그 효율적 사용법도 다르다. 그러므로 보다 효율적인 작업을 위해 애니메이션 작업에 주로 사용되는 여러 종류의 이미지 포맷들을 특성을 파악하는 것이 필요하다.

먼저 오늘날 통신, 데이터저장 등 컴퓨터 전반에서 널리 쓰이는 데이터 압축의 두 종류인 손실 압축과 비손실 압축의 원리와 이미지표현의 서로 다른 구조인 비트맵 방식과 벡터 방식의 차이점, 24비트 트루 컬러와 8피트 알파채널 등을 살펴보았다. 이러한 특성들을 토대로 2D, 3D, 합성, 편집 등 여러 형태의 애니메이션 작업 간에 사용되는 이미지 포맷들의 기능적 차이점과 그에 따른 장단점을 분석하였으며 실험을 통하여 JPG 파일이 컴퓨터 작업 중에 적은 용량을 차지할 것이라는 생각이 잘못된 것임을 증명하였다.

결론적으로 용량, 기능, 화질을 고려할 때, 편집, 합성, 3D, 2D를 막론하고 가장 효율적인 포맷으로는 TIF 포맷을 제안하며 학교 교육에서는 어도비사의 교육용 프로그램을 많이 사용하므로 호환성과 기능이 우수한 PSD 포맷을 제안하였다. 본 논문이 학교 교육에서나 실무 작업에서 이미지 포맷을 선택함에 있어 기준이 되기를 기대한다.

주제어 : 손실 압축, 비손실 압축, TIF, PSD, JPG, 알파채널, 이미지 포맷

I. 서 론

1. 연구 배경

오늘날 애니메이션 작업에 있어서 컴퓨터 작업

은 필수불가결의 요소가 되었다. 하드웨어와 소프트웨어를 바탕으로 모델링과 렌더링을 거쳐 제작되는 Full 3D Animation은 물론 한 장 한 장 직접 손으로 그리는 전통적인 2D Animation에 이르기까지 컴퓨터의 능력을 사용하지 않는 작업공정은 없다고 단언 할 수 있다. 이와 같은 디지털 의존적인 작업은 수정 및 복제의 편리성에 기인하여 작업의 효율성을 극대화 시켜주는 장점이 있는 반면

디지털의 여러 특성들을 파악하지 못할 경우 작업의 노력에 상응하는 성과를 보장하지 못하는 결과가 발생하기도 한다. 반면에 이러한 디지털의 특성을 잘 활용한다면 보다 경제적이고 합리적인 작업 공정을 구축 할 수 있어 산업현장의 가장 중요한 원칙중 하나인 비용, 시간대비 보다 높은 수준의 작업을 할 수 있는 기반이 될 수 있다.

컴퓨터를 이용한 애니메이션 제작에서 고려해야 할 기술적 사항들은 매우 많다. 그 중 2D 이미지를 사용함에 있어서 주의해야 할 점은 아날로그와 디지털의 변환 간에 용량대비 화질을 유지하는 것과 두 이미지를 합성함에 있어 알파채널이라고 부르는 투명도 설정부분이다. 이는 본 연구자가 96년부터 2001년까지 5년간의 실무 제작을 함에 있어 가장 주의를 기울인 부분이며 본 논문은 이와 같은 경험의 결과를 정리한 것이다.

본 논문은 컴퓨터 그래픽 소프트웨어의 구조 자체를 연구하는 것이 아니라 사용자 입장에서 알아야 할 눈높이 기본 원리 설명과 프로덕션 제작 경험의 결과로 나온 애니메이션 작업에 적합한 이미지 포맷을 정리하는 것으로 앞으로 다른 연구자나 교육현장에서 참고하여 작업의 효율성을 높일 수 있게 하려는 것임을 전제[前提]하고자 한다.

2. 연구의 범위와 및 방법

디지털의 기반이 되는 컴퓨터는 이미 애니메이션 뿐 아니라 인간의 DAN 분석¹⁾과 같은 기초과

학 분야에서부터 건축, 자동차 설계와 같은 산업현장에서는 물론 미술, 음악, 영화처럼 인간의 감각적 부분까지 두루 널리 쓰이는 만능의 도구로서 오늘 이 시대를 대표하는 문명의 이기[利器]이다²⁾. 그러므로 이 방대한 컴퓨터의 응용 범위 속에서 자칫 방향을 잘못 설정할 경우 본 논문이 지향하는 바인 애니메이션 제작에 있어서의 관점이 아닌 컴퓨터와 관련된 여러 분야를 의미 없이 인용하거나 그래픽 소프트웨어의 기초적인 원리를 다루는 공학적 논문으로 변질될 우려가 있다. 그러므로 원리설명이나 전문용어 사용에 있어 프로그램 개발자가 아닌 사용자의 관점을 견지[見地]하여 애니메이션 제작이나 교육에 몸담고 있는 분들이 참고하기에 무리가 없도록 하였다.

연구 방법은 먼저 이미지 포맷을 변환하고 저장함에 있어 가장 중요한 원리인 데이터 압축에 관한 기초적인 사항을 정리하였으며 실제 애니메이션 제작에 주로 사용되는 소프트웨어들과 그 소프트웨어에서 필요한 기본기능들, 특히 합성과 관련된 부분에 대해 알아보았다. 그리고 이에 호환되는 2D 이미지 포맷들을 특성과 장단점을 비교 분석하여 가장 효율적인 이미지 포맷을 선정하는 방법을 취하였다.

보건원(National Institutes of Health/NIH)에서 시작되었다. 컴퓨터의 도움 없이는 30억에 달하는 염색체 전부를 해독하는 것은 불가능한 것으로 전 세계적인 단체의 경쟁과 연합으로 2001년 2월에 전체 인간 게놈 지도를 완성했다고 공식 발표했다. 「브리태니커 백과사전 2007년판 CD HX」, 검색어: 인간게놈계획

1) 인간 게놈 프로젝트(Human Genome Project): 인간을 구성하고 인간으로서의 기능을 발휘하게 하는 청사진인 모든 유전정보를 완전히 해독한다는 계획으로 1989년 1월 미국국립

2) 컴퓨터 응용분야를 일반 사업분야, 공공 개인분야, 정보통신 분야, 기타 분야로 나누기도 한다. 신동일의, 「컴퓨터 사이언스. PC와 인터넷을 중심으로」, 사이텍미디어, 2003, p.1-22.

II. 이미지 데이터 압축

1. 데이터 압축의 원리

데이터 압축의 기원은 1832년 전신용으로 발명한 모스 부호이나 지금과 같은 데이터 압축이 보급된 것은 실용적 데이터 압축 알고리즘이 개발되었기 때문이다³⁾. 오늘날과 같은 대용량 저장매체가 없던 초기 컴퓨터 사용 세대⁴⁾에는 데이터 압축 기술이 저장 공간의 부족 해소에도 일조하였으나 데이터통신이 발달함에 따라 그 범용성과 실용성은 더욱 높아지게 되었고 현재는 이미지 표현과 원활한 동영상구현과 같은 시각정보 전달에서부터 다량의 데이터 전달 및 보관에 이르기까지 디지털 콘텐츠의 가장 핵심적인 기술로 자리 잡고 있다⁵⁾.

전원이 입력된 것과 그렇지 않은 것을 0과 1로 구현하는 원리로 개발된 전자계산기는 수많은 기술을 접목하여 오늘날 컴퓨터로 진화하였으며 지금까지 무어의 법칙⁶⁾을 따라 그 계산 능력과 속도

는 기하급수적으로 증가하고 있다. 그럼에도 불구하고 기본 원리는 최초의 방식과 변함없이 여전히 모든 데이터 구조는 0과1로 이루어져 있다. 데이터 압축이란 바로 이 반복된 0과1의 숫자를 분석하여 공통된 부분을 줄임으로써 전체적인 부호의 양을 감소시키는 작업이다. 특히 이미지와 관련한 데이터 압축을 encoding, 다시 복원하는 과정을 decoding이라 하며 그 기능을 하는 일종의 프로그램을 각각 encoder, decoder라 부른다.

데이터 압축 기술은 1952년 미국 MIT공대의 데이비드 허프만이 발표하여 압축의 기본이 된 허프만 방식이 대표적이며 이후 ZIP, ARJ, RAR, LZW 등 여러 방식으로 발전되고 있다. 그 세부적인 원리구현의 이해보다 오히려 사용자의 입장에서 이미지 데이터 압축의 사용 방법과 종류 및 특성을 파악하는 것이 실제 애니메이션 제작에 더 도움이 될 것이다.

2. 데이터 압축의 필요성

컴퓨터 처리용량이 늘어남에 따라 컴퓨터 운영체제도 MS-DOS와 같은 CUI(텍스트 유저 인터페이스:Character User Interface)에서 Windows와 같은 직관적 방식인 GUI(그래픽 사용자 인터페이스:Graphical User Interface)로 발전해 왔다. 운영체제의 발전에 따라 컴퓨터가 다루는 정보도 텍스트 방식에서 이미지, 사운드, 영상 과 같은 멀티미디어 형태를 띠게 되었으며 인터넷의 활성화로 데이터의 이동성은 더욱 부각되고 있다⁷⁾.

3) 1977년에서 78년에 J.Ziv와 A.Lempel이 발명한 2종류의 데이터 압축 알고리즘에서부터 비롯되었다고 한다. 박지완, 「문서 데이터 압축 알고리즘 입문」, 성안당, 1995, p.1

4) 최초 Microsoft MS-DOS 사용 메모리는 1Mbyte의 반밖에 되지 않는 640kbyte였으나 2008년 Microsoft Vista 권장 메모리는 2Byte로 최초에 비해 4천배 정도 된다.

5) 박성태외, 「정보통신개론」, 생능출판사, 2001

6) 무어의 법칙 (Moore's law): 인텔의 공동 설립자인 고든 무어가 1965년에 경험적인 관찰에 바탕을 두고 내놓은 것으로 반도체 집적회로의 성능이 24개월마다 2배로 들어난다는 법칙이다. 이종명, 「과학이 세상을 지배한다」한림원.

7) 신동일의, 「컴퓨터 사이언스. PC와 인터넷을 중심으로」, 사

텍스트 데이터와 멀티미디어 데이터의 크기차이는 현격하다. 예를 들어 최근에는 잘 사용하지 않지만 우리에게 친숙한 3.5인치 디스켓 한 장에는 200자 원고지 5000천장의 글을 저장 할 수 있다⁸⁾. 하지만 1.44Mbyte인 디스켓에는 대부분의 용량이 약 3Mbyte이상인 일반적인 MP3 한곡도 온전히 저장할 수 없으며 영상의 기본 사이즈인 SD급 이미지 한 장을 겨우 저장 할 수 있는 크기다⁹⁾. 이같이 텍스트데이터와는 비교할 수 없이 큰 멀티미디어 데이터를 다루기 위해서는 데이터 압축 기술이 꼭 필요하다.

3. 비손실 압축

인터넷 사용 중 프로그램을 다운로드하여 컴퓨터 설치하는 과정을 예를 들어 보면 다운로드 받은 하나의 파일이 설치될 때는 용량이 늘어나면서 여러 개의 파일로 나뉘게 된다. 이때 최종적으로 설치된 프로그램은 개발자가 만든 것과 온전히 동일하며 하나의 정보라도 틀릴 경우 프로그램은 실행되지 않는다. 또 다른 예로서 워드프로세서 파일 10개를 압축 프로그램으로 묶은 뒤 이동시켜

압축을 풀면 이전과 동일한 파일이 생성 된다. 이 같이 압축 보관 후 다시 복원했을 때 온전히 동일한 상태를 유지하는 압축방식을 가역[可逆](reversible)압축, 또는 데이터의 잡음(오류)이 허용되지 않기 때문에 무잡음(noiseless) 압축 이라고도 하며¹⁰⁾ 현재는 일반적으로 데이터의 손실이 없으므로 비손실(lossless) 압축이라고 부른다. 대표적으로 데이터 압축에 많이 사용되는 ZIP¹¹⁾ 방식과 이미지 압축에 사용되는 LZW¹²⁾ 방식이 있다.

4. 손실 압축

앞서 설명하였듯이 멀티미디어 파일은 매우 용량이 크므로 압축이 꼭 필요하다. 그러나 문자, 도형, 일반 데이터 등을 비손실 압축했을 때 평균 압축률은 2대1정도 밖에 되지 않아 더 큰 압축이 필요하였다. 1974년 텍사스 대학의 라오 교수를 비롯한 3명의 연구진이 오늘날의 멀티미디어 혁명을 가능케 한 발명인 이산코사인변환 (DCT:Discrete Cosine Transform)이라는 새로운 직교변환에 관한 논문을 발표하였다¹³⁾. 이것은 인간의 시각세포가 비슷한 색을 잘 구별하지 못하는 일종의 착시현상

이택미디어, 2003, p.6-17.

8) 1.44 Mbyte = 1.44 * 1024 * 1024 Byte = 1,509,950 Byte 즉, 대략 151만자 정도의 영문자를 저장할 수 있으며 한글은 2Byte로 구현됨으로 2로 나누면 76만자 정도 된다. 이를 200자 원고지 한 장에 150자정도 글을 쓴다고 할 때 원고지 5000장 (영문일 경우 1만장) 이상을 디스켓 단 한 장에 담을 수 있다. 100장 묶음의 두께를 1 cm 이라고 한다면, 50 cm 두께에 해당하는 글자로 가득 찬 한 엄청난 양의 종이 봉치가 된다.
9) 오늘날 영상 매체의 기본인 DVD와 방송화질 중 HD 이전 사양인 SD급 화질은 모두 가로720*세로480의 이미지 사이즈를 기본으로 한다. 720*480의 이미지를 비손실 압축으로 저장했을 때 1031kbyte(약1.03Mbyte)의 용량이 나온다.

10) 식송우언저, 박지환역, 「문서 데이터 압축 알고리즘 입문」, 성안당, 1995, p.5

11) 1989년 필 케츠가 그의 회사 PKWare에서 발명한 것이다. 처음에는 pkzip.exe로 압축하고, pkunzip.exe로 압축을 푸는 방식으로 사용법이 복잡해 국내에서는 사용자가 많지 않았으나, 97년경 Microsoft에서 Windows의 공식지원을 밝히면서 범용적으로 사용하게 되었다. [온라인], wikipedia, [http://en.wikipedia.org/wiki/ZIP_\(file_format\)](http://en.wikipedia.org/wiki/ZIP_(file_format))

12) LZW(Lempel-Ziv-Welch)는 1978년 렘펠과 지브가 공개한 LZ78 알고리즘을 웰치가 1984년에 개선한 비손실 데이터 압축 알고리즘이다. 이 방식은 예를 들어 이미지에서의 압축과 같이 다른 플랫폼에 적용하기 쉽도록 고안되었다. [온라인], wikipedia, <http://en.wikipedia.org/wiki/Lzw>

13) 정재창, 「전자신문」, 1995년 6월 16일 기사

을 이용한 것으로, 이미지를 여러 개의 블록으로 나눈 후 색의 변화를 주파수로 변환 계산하여 변화값이 큰 블록은 정보량을 늘리고, 변화 값이 크지 않은 블록은 정보량을 줄여서 저장하여 데이터량을 획기적으로 줄이는 방법이다. 이 같이 소리, 이미지와 같은 원래의 데이터를 똑같이 복원할 필요가 없이 조금 틀려도 상관없는 성질을 이용하는 압축법을 비가역[非可逆](irreversible)압축, 또는 유잡음(noisy) 압축, 현재는 주로 손실(loss) 압축이라고 부른다. 대표적 손실 압축으로 인터넷에서 자주 접하는 그림파일인 JPG 포맷이 있다¹⁴⁾.

III. 애니메이션 작업에 사용되는 이미지 포맷 종류

1. 비트맵 방식과 벡터 방식

2D 이미지 포맷은 크게 두 가지로 나눌 수 있다. 비트맵(bitmap)이미지 또는 레스터(raster)이미지라 부르는 방식은 여러 개의 픽셀(pixel)들의 조합으로 이루어진 이미지를 말한다. 디지털카메라로 촬영한 그림과 같이 가로픽셀과 세로픽셀을 가지며 각각의 픽셀이 Red, Green, Blue,의 값을 8비트, 즉 28인 256단계로 계산하여 24비트, 약 천6백만 개의 색¹⁵⁾을 표현하므로 화려하고 복잡한 이미

지를 표현하는데 적합하다. 단점으로는 아래에서 설명할 벡터방식의 이미지에 비해 비교적 이미지 용량이 크며 그림의 크기를 확대할 그림이 흐릿해지며 심할 경우 보이지 않던 픽셀마저 보이게 되어 이미지를 손상시킨다. Adobe Photoshop, Corel Painter, Corel Paintshop pro 등이 이 방식을 사용하는 대표적 프로그램이다¹⁶⁾.

또 한 가지 방식은 벡터(vector)방식이라고 부르며 이미지의 외곽선을 몇 개의 정점으로 정의하여 각각의 곡률과 거리 값으로 계산하여 연결한 후 그 내부를 여러 가지 색으로 채워 이미지를 구현하는 방식이다. 장점으로는 각 정점의 위치와 그에 따른 외곽선의 모양은 상대적으로 계산할 수 있으므로 그림의 크기를 아무리 확대해도 이미지에 전혀 손상이 없으며. 여러 정점의 위치와 연결된 곡률 값, 선두께, 채워진 색 등의 정보만이 필요하므로 데이터의 크기가 매우 작다. 그러나 외곽선을 추출해야만 하므로 아주 복잡한 형태나 일정한 패턴이 아닌 다양한 색을 가진 그림을 만들기가 어렵다는 단점이 있다. 이 방식을 사용하는 대표적 프로그램으로 Adobe Illustrator, Corel Draw, Autodesk Autocad 등이 있다¹⁷⁾.

인터넷환경이 발전하면서 홈페이지에서 움직이

개와 같이 나열했을 때 인간의 눈으로 색의 경계선을 구별하지 못한다고 하여 실제칼라(Truecolor)라고 부른다. 포토샵 경우 기능 향상으로 색의 정보를 8비트를 16비트, 32비트로 높일 수 있으며 이때 저장용량은 평균 두 배씩 늘어난다.
16) 최근에는 비트맵 기반의 소프트웨어에서도 벡터방식을 사용할 수 있도록 성능이 향상되는 추세이다. 그러나 프로그램의 다양한 기능을 사용하기 위해서는 벡터의 성질을 비트맵으로 전환시켜야만 하는 경우가 많다.

17) Autocad는 2차원 설계프로그램이었으나 현재는 3차원 소프트웨어의 대명사로 발전하였다. 3D 프로그램은 모두 벡터방식으로 작업이 이루어지며 렌더링 이미지는 비트맵방식이지만 최근에는 벡터방식 렌더링도 지원하는 추세이다.

14) JPEG는 손실 압축과 비손실 압축을 모두 지원하나 일반적으로 사용하는 방식은 손실 압축 방식이다.

15) Red, Green, Blue가 각각 8비트씩을 가지므로 $8 \times 3 = 24$ 비트가 된다. $256 \times 256 \times 256 = 16,777,216$, 1천6백만 개의 색은 무지

는 이미지 제작을 위해 개발된 프로그램인 플래시(Flash)를 애니메이션 제작에서도 사용하면서 벡터 포맷의 파일을 비트맵으로 전환하지 않고 그 특성을 그대로 살려 애니메이션 제작에 사용하게 되었다. 벡터 포맷을 활용한 제작은 벡터 포맷 자체의 적은 용량과 명령어를 적용한 움직임으로 최종 용량을 줄여 인터넷에서 전송부하를 줄이기 위한 목적으로 최초 사용되었다. 그러나 벡터 포맷의 기능을 활용한 플래시 애니메이션은 벡터 포맷의 단점인 화려한 색과 복잡한 레이아웃을 표현하는데 한계를 가지고 있어 온전한 벡터 포맷의 기능만을 활용한 애니메이션을 제작하는 경우는 극히 찾아보기 힘들며 대부분 비트맵 포맷을 주로 사용하고 있다¹⁸⁾. 플래시 애니메이션이 가지는 다른 소프트웨어와의 차별 점은 벡터 포맷의 사용보다는 명령어를 구조를 적용하여 보는 이와 상호작용이 가능한 콘텐츠를 만들어 낼 수 있으므로 인터넷 환경에 적합한 것으로 볼 수 있다.

2. 각 이미지 포맷의 특성

현재 프로덕션이나 가정에서 일반적으로 사용되는 이미지 포맷은 매우 다양하나 세부적 특징은 그다지 뚜렷하게 구분되지 않으므로, 모든 이미지 포맷을 나열하는 것 보다 중요하고 대중적인 포맷을 알아보는 것이 더 본 논문의 취지에 적합하다. 가장 대중적 이미지 처리 소프트웨어인 포토샵에서 읽고 쓸 수 있는 포맷이라면 널리 사용되면서

18) 최근의 인터넷 트렌드인 UCC동영상은 대부분 FLV 확장자를 가진 플래시동영상이다. 그러나 FLV 포맷의 내부구조는 비트맵방식인 Quicktime의 형태를 차용하고 있다.

도 기능적으로 중요한 포맷으로 선정하여도 무방할 것이다. 최신버전인 CS3로 알아보았다.

확장자	특징
PSD, PDD	Photoshop 고유 포맷, 비손실 압축지원, 인지도 매우 높음
3DS, OBJ	3D 프로그램인 파일 ¹⁹⁾
BMP, RLE, DIB	윈도우 공식 지원 포맷, 화질 대비 용량 큼
TIF, CRW, NEF, RAF, ORF	Camera raw, 디지털 카메라 데이터 포맷. 비손실 압축 지원
CIN, SDPX, DPX, FIDO	Cineon 디지털 필름시스템 전용 포맷, 채널당 10비트 칼라 저장 가능
GIF	컴퓨터 통신 초기 개발, 손실압축, 256칼라, 애니메이션기능 투명도 적용으로 널리 사용
EPS	벡터, 비트맵 모두 가능한 포맷, 문서출판에 주로 사용
FLM	Premiere와 Photoshop 호환용 Adobe사 전용무비 포맷, 실제 사용 빈도 떨어짐
JPG, JPEG, JPE	현재 인터넷상에서 가장 널리 사용. 손실 포맷이지만 압축률 매우 높음, 24비트 칼라사용
PSB	Adobe 이미지레디 포맷. 사용빈도 매우 낮음
PDF, PDP, PDE, PDP, AI	Adobe에서 개발한 문서전용 파일. 디지털출판에 주로 사용
AI3, AI4, AI5, AI6, AI7, AI	Generic EPS, 어도비사의 대표적 벡터방식 일러스트레이터의 기본 포맷. 호환성 뛰어나함.
PCX	컴퓨터초기 Zsoft사의 PC Paintbrush와 IBM페인트와의 파일교환을 위해 만들어진 포맷.
PCD	코닥 포토CD디스크 전용 포맷
RAW	이미지를 데이터 구조로 변환한 포맷.
PCT, PICT	맥킨토시 표준 그래픽 이미지 포맷.
PXR	3D Animation으로 유명한 pixar의 전용 포맷.
PNG	인터넷용 이미지 포맷, 인터레이스 방식 지원, 24비트 이미지, 비손실 압축 지원
SCT	scitex사 컴퓨터와 파일 교환을 위한 포맷
TGA, VDA, ICB, VST	Truevision사에서 제공하는 Targa와 Vista 비디오 보드 전용 포맷. 현재 널리 사용
TIF, TIFF	모든 컴퓨터에서 널리 사용됨, LZW 비손실 압축 지원, 투명도 알파채널지원
WBMP, WBM	모바일 장치용 그래픽 포맷

[표1] 포토샵에서 읽을 수 있는 포맷의 종류와 설명

[표1]과 같이 포토샵 CS3에서 작업 가능한 이미지 포맷 중 애니메이션 작업 간에 주로 접하는 비트맵 포맷의 종류와 그 특징들을 살펴본다.

19) 포토샵에서 3D파일을 부르면 3D레이어로 인식되어 기본 웨이딩 된 오브젝트를 직접 돌려볼 수 있다. 단 작업을 위해서는 비트맵으로 변환시키는 과정이 필요하다.

1) PSD 포맷

어도비사에서 개발한 포토샵 전용 포맷으로 레이어, 마스크, 텍스트, 이펙트, 벡터 등 포토샵에서 지원하는 모든 기능을 담는다. 별다른 옵션이 없이 기본적으로 비손실 압축방법으로 저장되며 어도비 계열의 소프트웨어에서 모든 기능들이 완벽 호환됨으로 그 범용성이 매우 크다.

2) GIF 포맷

GIF 포맷은 1987년 미국의 통신 회사였던 컴퓨서브(Compu Serve)가 모뎀 등 네트워크를 통해 전송할 목적으로 제정한 그림 파일 압축 규격으로, Graphics Interchanges File의 약자다. GIF 포맷은 LZW 비손실 압축 방식을 사용하나 파일 용량을 줄이기 위해 색상 수를 8비트인 256색으로 제한하였다. 이 후 인터레이스²⁰⁾ 표시 기능과 투명색 지정 기능, 애니메이션 기능 추가 되어 지금도 인터넷상에서 많이 사용되는 포맷이다.

3) JPG, JPEG 포맷

JPEG는 Joint Photographic Expert Group의 약자로 손실과 비손실 압축을 지원하지만 우리가 흔히 사용하는 JPG 포맷은 DCT(Discrete Cosine Transform, 이산코사인변환)이라는 수학적변환공식에 기반을 둔 손실 압축 방식을 이용하는 것이다²¹⁾. 이는 24비트의 색을 이미지 손상을 크게 느끼지 못함에도 매우 높은 압축을 이룰 수 있어

256색으로 표현의 한계가 많았던 GIF 포맷을 대신하여 인터넷공간에서의 대표적 포맷으로 세대교체를 하고 있다. DOS시절 파일 확장명 제약으로 JPEG를 줄인 말이 JPG다.

4) PNG²²⁾ 포맷

PNG 포맷은 Portable Network Graphics의 약자로 기존 GIF의 라이선스와 구조적 문제를 개선하기 위해 1996년에 W3C²³⁾에서 인증한 포맷이다. GIF와는 달리 24비트 칼라와 8비트 투명도를 지원한다. 비손실 압축 포맷이며 인터레이스, 기능이 있다. 애니메이션 기능은 없지만 기능이 추가된 JNG, MNG 포맷이 있다.

5) BMP 포맷

마이크로소프트 윈도우 지원 비트맵 포맷²⁴⁾으로 개발 초기 역사를 같이 한 IBM의 OS/2에서도 공식 지원한다. 24비트 칼라를 표시할 수 있었으나 2001년 10월 발표된 윈도우XP에 와서 알파 채널을 포함한 32비트 포맷을 지원하였다. 일반적으로 데이터를 압축하지 않고 사용되지만, 최근들어 비손실 RLE 압축 방식도 지원한다.

6) TIF, TIFF 포맷

TIFF는 Tagged Image File Format의 약자로 1980년대 엘더스(Aldus)사와 스캐너 제조사들이 일반적인 파일 형식을 사용하여 저장기 위하여 개

20) 다운로드 속도가 느린 인터넷상에서 먼저 받은 데이터를 일정간격의 가로줄로 점차적으로 이미지를 보여주는 방식.

21) 김명찬외, 「새로운 비손실 압축 기법에 관한 연구」, 『창원대학교산업기술연구소 논문집』, 제14호, pp57-68

22) 관련된 모든 규격을 개발하고 관리하는 공식홈페이지는 <http://www.libpng.org/pub/png/>

23) W3C(World Wide Web Consortium)는 월드 와이드 웹을 위한 표준을 개발하고 장려하는 조직으로 팀 버너스 리를 중심으로 1994년 10월에 설립되었다. <http://www.w3.org/>

24) 윈도우즈 공식지원 벡터 포맷은 WMF확장자를 가진다.

발되었다. DOS 시절 파일 확장명 제약으로 TIFF를 줄인 말이 TIF다. 1992년 엘더스사를 어도비사가 합병함으로 포토샵의 PSD 포맷과 같이 의 레이어 기능이 추가되는 등 기능이 높아지고 있다. LZW 압축이 가능하여 RLE 압축을 사용하는 다른 비손실 포맷에 비해 더 높은 압축률을 가지며 8비트의 투명도의 알파채널도 포함하고 있어 합성작업에 유연하게 사용된다.

7) TGA 포맷

TGA 포맷은 트루비전(True vision)사의 방송용 장비인 타가보드를 위하여 개발된 그래픽 포맷이다. 24비트 컬러와 8비트 투명도 까지 저장 가능하며, 비손실 RLE 압축을 지원한다. 1989년 업데이트 이후 변화가 없었지만 컴퓨터 역사 초기 그래픽 작업을 TV화면으로 볼 수 있도록 변환 해주었던 타가보드의 높은 위상으로 현재도 2D 및 3D 애니메이션 분야에서 인지도가 높다.

2. 2D Animation 작업에 사용되는 이미지 포맷

2D 애니메이션 제작방법은 전통적인 한 장 한 장 그림을 그리고 셀룰로이드에 라인을 복사하여 뒷면에 채색한 후 배경과 함께 촬영하는 고전적 방식에서 최초의 라인 드로잉부터 마지막 리코딩 작업까지 온전히 디지털로 작업하는 단계로 발전하였다. 그러나 업계 및 교육현장에서 가장 많이 사용하는 방식은 한 장 한 장의 라인 드로잉을 애니메이션 전용 용지에 전통적 방식으로 그린 다음

고속스캐너로 컴퓨터에 입력 하는 방식이다. 이 같은 제작과정에서는 스캐너 입력이후의 채색, 배경 합성, 편집, 리코딩까지의 모든 과정이 소프트웨어 속에서 이루어진다.

이 처럼 소프트웨어 속에서 진행될 경우 그 소프트웨어에 최적화 된 전용 포맷²⁵⁾²⁶⁾이 사용되므로 범용 이미지 포맷을 사용할 필요가 없다. 범용 이미지 포맷이 사용되는 곳은 드로잉 데이터를 스캐너로 입력할 때와 배경으로 그린 그림을 스캐너로 입력할 때 2가지 밖에 없으며. 편집을 마친 후 더빙과 같은 후반작업을 위해 아날로그 테이프 출력 시 일반적인 레코딩 시스템을 사용한다면 이 때도 범용 이미지 포맷을 사용하기도 한다.

현재 애니메이션 업체에서 입출력 간에 사용하는 포맷은 대부분 TGA 또는 TIF 포맷을 사용하고 일반적으로 인터넷상에서 많이 사용되는 JPG나 BMP, PNG 포맷 등은 사용하지 않고 있다. JPG 포맷은 앞서 설명에서와 같이 손실 포맷이므로 작업 간에 소스로 사용하기에는 부적합하며 BMP 포맷은 화질대비 용량이 크고, PNG 포맷은 아직 소프트웨어에서 지원하는 경우가 많지 않고 인지도가 낮기 때문이다.

3. 3D Animation 작업에 사용되는 이미지 포맷

25) Peps 3.05버전의 전용 포맷들/ pxs: 스캔 받은 이미지 파일, pxa: 컬러 이미지 파일, pxi: full color 이미지 파일, pxv: Vector 이미지 파일, pxm: sheet 작업파일

26) Toonz 4.3의 전용 포맷들/ tzi: 스캔받은 이미지 파일, tzu: 컬러 확장 이미지 파일 (클린업), tzp: 컬러 이미지 파일 (painting), scn: 스캔 셋업파일, cln: 클린업 셋업파일, casm: 최종렌더링을 위한 어리소스 파일, xsh: X-sheet 셋업 파일

3D 소프트웨어에서 2D 이미지를 가져오는 경우는 모델링을 위한 4면도 기초 드로잉, 3차원 물체에 색을 입히는 Mapping 이미지, 렌더링을 위한 배경 이미지로 크게 나눌 수 있으며 2D 이미지를 내보내는 경우는 최종 편집을 위해 초당 30장의 그림을 한 장씩 내보내는 시퀀스 방식이 있다.

이미지를 부를 때, 3D 소프트웨어들은 대부분의 이미지 포맷들을 부를 수 있으며 심지어 포토샵과 같이 복잡한 구조를 가지고 있어도 대중적으로 사용된다면 읽을 수 있는 기능이 추가되고 있다. 또한 작업 간에 사용되는 이미지는 맵 소스는 소프트웨어 상에서 연속 배치, 칼라 조정, 거리에 따른 심도 표현 등, 재창조되어 렌더링 되는 경우가 많으므로 이미지 압축여부에 따른 편차가 그리 크지 않은 편이다. 그러므로 특수한 포맷을 사용하기보다 작업자의 취향에 따라 범용적인 포맷을 사용하는 경우가 대부분이다.

주로 사용되는 포맷은 인터넷에서 구한 자료들일 경우 JPG, BMP, GIF, PNG 포맷 등을 그대로 사용하며 때로는 포토샵과 같은 이미지 소프트웨어를 통해 TIF, TGA 포맷 등으로 변환시켜 사용하기도 한다. UV좌표를 이용한 매핑소스와 같이 소프트웨어 상에서 직접 생성한 이미지는 합성을 위해 알파채널이 꼭 필요하므로 알파채널을 포함하면서 널리 사용되며 비손실 압축을 지원하는 TIF 포맷과 TGA 포맷이 많이 사용된다.

4. 편집, 합성 작업에 사용되는 이미지 포맷

2D, 3D를 비롯하여 그 외 여러 실험적 기법이

나 스톱모션 애니메이션 까지 모든 애니메이션 작업은 음향, 특수효과 등 후반작업 과정을 거쳐서 완성된다. 이러한 후반작업을 위한 소프트웨어는 매우 많지만 기본적인 기능은 작업소스를 실시간 재생으로 타이밍을 확인하여 컷을 구성하도록 하는 편집에 중점을 둔 것과 작업소스의 보정작업 기능, 여러 이미지들을 중첩, 부분삭제, 부분추가를 통한 새로운 이미지 합성과 같은 이펙츠 기능에 중점을 둔 것이 있다.

특히 이미지 보정을 위해서는 처음부터 가지고 있는 색의 정보량이 중요하므로 저장하는 과정에서 이미 많은 정보들을 생략한 JPG 포맷과 같은 손실압축 방식은 색 보정 작업에서 사용하지 않는 것이 좋다. 또한 배경과 캐릭터의 중첩과 같은 알파채널의 투명도를 사용하는 작업과 같은 경우 알파채널을 가지고 있지 않은 파일은 한 장씩 수작업으로 투명부분을 생성해야 하므로 매우 어려운 과정을 거치게 된다. 때문에 알파채널을 지니고 있으면서 인지도가 높은 TIF, TGA 포맷을 주로 사용하며 어도비 소프트웨어 간에는 레이어별로 다중 알파채널을 가지는 PSD 포맷이 가장 좋다. PNG 포맷도 인터넷 홈페이지제작에 편리하도록 투명도를 가지고 있는데 최근에는 여러 이펙츠 소프트웨어에서 PNG 포맷을 읽고 쓸 수 있는 기능이 추가되고 있어 합성작업에 사용될 수 있는 가능성이 높아지고 있다.

IV. 이미지 포맷 특성 비교

하여 데이터를 보충하였다.

1. JPG 파일에 대한 오해

파일을 용이하게 저장하기 위해 여러 방법의 압축을 사용하는 것을 앞서 살펴보았다. 특히 학교 교육 중에 학생들의 작업과정을 살펴보면 JPG파일을 많이 사용하는 것을 볼 수 있는데 이는 다른 이미지 포맷들과 비교하여 적은 용량이므로 대량의 파일을 소프트웨어에서 불러서 사용하여도 컴퓨터에 무리가 없으리라 생각하기 때문이다. 하지만 실제로 JPG파일이 컴퓨터 작업에 사용될 때 어느 정도 하드웨어 능력을 사용하는지 알아볼 필요가 있다.

테스트 이미지는 윈도우XP를 설치했을 때 기본으로 제공되는 [그림.1] “수련.jpg”를 작은 이미지의 예제로 사용하였으며 큰 이미지의 예제로는 인터넷에서 무료로 제공되는 컬러 프린터 및 모니터 테스트용 그림인 “Reference Print printer.jpg”²⁷⁾ [그림.2]를 사용하였다. 각각 하드디스크 저장 시 용량과 포토샵을 최초 실행시켜 그림을 불러왔을 때 윈도우 작업관리자에 나타난 RAM 사용 변화를 통해 알아보았으며 윈도우 작업관리자에서 나타나는 RAM 용량은 Mbyte 단위이므로 오차를 줄이기 위해 각 5회의 테스트 후 평균값을 사용하였다. 압축 용량별 테스트를 위해 포토샵의 저장기능을 활용하여 대, 중, 소 3종류로 압축된 이미지를 비교하였으며 포토샵의 캐쉬 기능으로 RAM이 사용되지 않도록 매번 새롭게 포토샵을 실행하여 RAM 사용량을 체크했다. 비교대상으로 TIF 포맷을 선정



[그림.1] “수련.jpg” 해상도: 800X600



[그림.2] “Reference Print printer.jpg” 해상도: 2100X1500

형 태	하드디스크 저장용량	포토샵에 부른 후 RAM 증가량
그림1. JPG 원본	82 Kbyte	2 Mbyte
그림1. JPG 압축 0	50 Kbyte	2 Mbyte
그림1. JPG 압축 6	107 Kbyte	2 Mbyte
그림1. JPG 압축 12	344 Kbyte	2 Mbyte
그림1. TIF 무압축	1,433 Kbyte	2 Mbyte
그림2. JPG 원본	4,472 Kbyte	35 Mbyte
그림2. JPG 압축 0	446 Kbyte	35 Mbyte
그림2. JPG 압축 6	977 Kbyte	35 Mbyte
그림2. JPG 압축 12	4,488 Kbyte	35 Mbyte
그림2. TIF 무압축	24,543 Kbyte	35 Mbyte

[표.2] 압축 용량별 RAM사용 테스트 결과

27) <http://www.colour-science.com/>에서 구할 수 있다.

[표.2]에서와 같이 결과는 너무나 단순명료하다. 두 개의 그림 모두 각각 압축상태에 따라 하드디스크 저장용량은 큰 차이가 있었으나 포토샵작업을 위해 불러들 때에 차지하는 RAM 사용량은 모두 동일하였다. 즉 하드디스크 저장용량은 작업의 효율성에 아무런 관계가 없으며 오직 이미지의 가로×세로 크기만이 영향을 끼치는 것을 볼 수 있다. 이는 압축의 기본원리가 데이터를 저장할 때나 이동할 때만 사용되며 RAM과 CPU에서 처리될 때는 사용되지 않기 때문이다. 즉 손실, 비손실을 따지지 않고 압축된 모든 데이터는 작업을 위해서 RAM에 불러들일 때는 다시 무압축 상태로 바뀌게 되기 때문이다. 이 같은 결과로 볼 때 JPG 포맷과 같은 손실 압축 포맷은 저장할 때와 인터넷을 통한 데이터 이동에만 유용하며 정확한 작업을 위해서는 매우 좋지 않은 포맷임을 알 수 있다. 특히 기술발전으로 인하여 10만원 내외로 700Gbyte의 하드디스크를 구매할 수 있는 요즈음과 같은 경우라면 저장 공간의 부족으로 작업에 어려움을 당하는 경우는 거의 찾아보기 어려우므로 데이터를 저장할 때 최대한의 정보를 담은 포맷을 사용하는 것이 보다 유리하다.

2. 포맷에 따른 용량 비교

위 실험의 결과는 색의정보를 최대한 많이 가지는 포맷을 사용하는 것이 효율적임을 알 수 있었다. 그러나 애니메이션 작업에서는 한 장의 이미지만을 사용하는 경우 보다 프레임 별로 순차적으로 증가된 숫자이름으로 저장된 시퀀스 파일과 같이

많은 수의 이미지 들을 동시에 다루는 경우가 많다. 이와 같은 경우 소프트웨어에서 데이터를 부를 때 하드디스크에 무리가 따르게 되며 부르는 시간도 많이 걸린다. 이 같은 원리로 볼 때 적은 량의 데이터가 이미지를 부르거나 저장할 때에는 분명히 효율적이다. 그러므로 같은 이미지가 포맷별로 저장되는 용량의 차이를 알아보는 것은 효율적인 제작공정을 위해 꼭 필요하다. 특히 합성에 필요한 투명도를 나타내는 알파채널은 이미지 포맷의 중요한 기능이므로 테스트 이미지는 알파채널을 포함하고 있는 것으로 한정 하였다.

테스트에 사용된 [그림.3]이미지는 앞에서 사용한 [그림.2]에서 사람 부분만을 800×800의 해상도로 줄인 다음 배경을 지우고 체크무늬를 합성하여 만들었다. 합성을 위해 만든 알파채널은 [그림.4]와 같으며 실험하는 이미지에 포함되어 있었다.



[그림.3] 테스트 이미지 해상도: 900×900



[그림.4] 합성을 위한 알파채널

테스트 결과는 [표.3]과 같다. 이 중 알파채널을 가지고 있지 않은 포맷과 색 정보가 부족한 GIF, 손실압축 포맷들을 제외하고 1번 PSD 포맷과 10번 비손실 압축 TGA 포맷, 13번 비손실 압축 TIF 포맷을 최종 비교 대상으로 선정 하였다. 여기서 용량대비 색의 정보와 기능성이 가장 좋은 것은 TIF 포맷이다. TGA 포맷은 RLE 압축방식을 사용하기 때문에 좀 더 효율이 좋은 TIF 포맷의 LZW 압축에 비해 용량이 크다. PSD 포맷도 비슷한 크기인걸로 미루어볼 때 LZW 압축 알고리즘을 사용하고 있는 것으로 판단해 볼 수 있다. 여기서 유의해야 할 것은 7번 PNG 포맷의 결과다. PNG 포맷은 저장과정 중 알파채널에 대한 다른 옵션이 없이 포토샵의 레이어 투명정도를 그대로 이용하므로 알파채널 생성을 위해 배경을 지워야만 한다. 그러므로 온전한 그림의 정보를 가지고 있지 않지만 대부분 알파채널의 역할이 합성에 필요 없는 배경을 지우는 역할을 하는 것이므로 배경정보를 제외한다면 역시 비손실 LZW 압축을 사용하여 용량대비 화질에서 오히려 TIF 포맷보다 좋은 결과를 보인다.

	포맷	압축	알파채널	데이터 용량	비고
1	PSD	비손실	O	1,474 Kbyte	레이어 1개
2	JPG	손실	X	64 Kbyte	압축퀄리티0
3	JPG	손실	X	739 Kbyte	압축퀄리티12
4	BMP	안함	X	2,374 Kbyte	24bit
5	BMP	안함	O	3,165 Kbyte	32bit
6	GIF	비손실	O	288 Kbyte	256칼라 배경 없음
7	PNG	비손실	O	1,082 Kbyte	배경 없음
8	TGA	비손실	X	1,400 Kbyte	24bit
9	TGA	안함	X	2,374 Kbyte	24bit
10	TGA	비손실	O	1,864 Kbyte	32bit
11	TGA	안함	O	3,165 Kbyte	32bit
12	TIF	안함	O	3,192 Kbyte	
13	TIF	비손실	O	1,470 Kbyte	

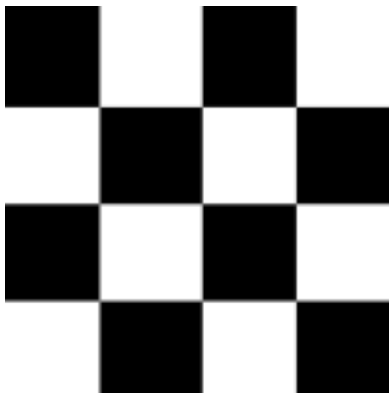
[표.3] 포맷별 저장 용량 테스트 결과

3. 그림에 따른 비손실 압축 용량 비교

데이터 손실이 없는 비손실 압축의 구조상 복잡한 그림은 압축효율이 낮고 단순한 그림은 압축효율이 높다. 테스트를 통하여 확인해 보자. 먼저 사용된 [그림.3]과 체크무늬배경을 따로 저장한 [그림.5]를 서로 비교대상으로 하였다. 실험 목적상 알파채널이 필요 없으므로 두 개의 그림 모두 알파채널이 포함되어 있지 않으며 비손실 압축을 할 수 없는 포맷은 비교에서 제외하였다. 단 참고삼아 JPG 포맷 최고화질을 추가 하였다.



[그림.3] 복잡한 이미지



[그림.5] 단순한 이미지

	저장 포맷	복잡한 이미지	단순한 이미지
1	PSD 압축	1,447 Kbyte	86 Kbyte
2	TIF 무압축	2,401 Kbyte	2,397 Kbyte
3	TIF 압축	1,293 Kbyte	39 Kbyte
4	TGA 무압축	2,364 Kbyte	2,374 Kbyte
5	TGA 압축	1,433 Kbyte	29 Kbyte
6	PNG 압축	927 Kbyte	7 Kbyte
7	JPG 손실압축	739 Kbyte	78 Kbyte

[표.4] 그림정보에 따른 저장 용량 테스트 결과

[표.4]의 2번과 4번을 볼 때 무압축일 경우 이미지의 복잡한 정도와 상관없는 용량인 것을 알 수

있다. 그러나 3번과 5번에서는 이미지의 내용에 따른 압축효율은 매우 차이가 큰 결과를 보이고 있다. 이는 1번 PSD 포맷과 6번 PNG 포맷의 경우에서도 볼 수 있는데 비손실 압축의 특성상 색 정보가 단순할수록 연속된 정보가 많아 압축할 수 있는 여지가 많아지기 때문이다. 이 같은 결과는 몇 가지 색으로 채색을 하는 2D애니메이션 제작공정에서는 매우 유리한 조건이 될 수 있다. 1번 PSD 포맷의 경우 단순한 이미지에서 압축효율이 떨어지는 이유는 포토샵 전용 파일 특성상 많은 부가정보들을 담고 있기 때문이다. 두 이미지 모두에서 최고의 압축률을 가진 파일은 PNG 포맷이다.

V. 결 론

JPG 포맷은 저장 시 적은 용량을 차지하지만 손실압축으로서 색 정보를 많이 손상시켜 저장하므로 인터넷에서 구한 소스파일이 아닌 이상 작업간 주고받는 포맷으로는 되도록 사용하지 않는 것이 좋다. BMP 포맷은 압축이 지원되지 않아 용량이 크므로 데이터를 저장하고 불러들일 때 효율성이 떨어져 역시 사용하지 않는 것이 좋다.

TIF 포맷과 TGA 포맷은 비손실 압축이면서 동시에 합성에 필요한 알파채널이 지원되므로 작업에 매우 적합한 포맷이다. 특히 복잡한 이미지에서는 RLE 방식의 압축을 사용하는 TGA 포맷보다 압축 효율이 높은 LZW 방식을 사용하는 TIF 포맷이 더 좋은 효율을 가진다고 볼 수 있다. 또한 개발이 중단된 TGA 포맷과는 달리 어도비사에서 관리되고 있는 TIF 포맷은 포토샵 레이어 정보를 그

대로 저장 할 수 있어 어도비계열 소프트웨어와 비 어도비 계열 소프트웨어 간의 호환성에서 각 소프트웨어의 기능을 최고로 활용하여 사용할 수 있는 매우 높은 범용성을 가진 포맷이다.

PNG 포맷은 아직 인지도가 높지 않으나 TIF 포맷을 능가하는 비손실 압축률과 제한적이긴 하지만 합성에 필요한 알파채널을 가지고 있으며 인터넷 환경에서의 호환성도 높아 앞으로 포토샵의 레이어 인식과 같은 소프트웨어 지원이 더 강화된다면 차세대 포맷으로 자리 잡을 가능성이 매우 높아 보인다.

PSD 포맷은 TIF 포맷에 버금가는 효율적인 비손실 압축 포맷이면서 알파채널과 제한적인 벡터 포맷 저장기능 등 작업에 매우 유용한 포맷이다. 그러나 이 보다 더 중요한 점은 앞서 다른 파일들과는 달리 어도비 공식 포맷으로서 애니메이션 제작에 많이 사용되는 편집, 이펙트 소프트웨어인 프리미어(premiere)와 애프터이펙츠(aftereffects)에서 레이어와 여러 플러그인을 적용한 정보가 세부적인 값의 손상 없이 그대로 불러 작업 할 수 있는 완벽한 호환성을 보여준다. 예를 들어 포토샵의 기본 포맷인 PSD 파일을 프리미어에서 열어 작업하는 도중 다시 포토샵에서 수정을 하고 저장하면 그 내용이 즉각 프리미어에 반영된다. 포토샵에서 작업한 PSD 포맷의 파일을 애프터이펙츠에서 그대로 불러들이면 각 레이어별로 작업된 상태 그대로 인식되므로 캐릭터의 관절을 여러 레이어로 작업해서 저장한 후 애프터이펙츠에서 불러들인다면 그림을 단순히 움직여서 제작하는 컷아웃(cutout) 애니메이션 제작과 같은 작업에서 매우 효율적인

공정이 될 수 있다. 일러스트레이터와의 벡터호환성도 매우 뛰어나며 벡터 포맷을 이용한 플래쉬(Flash) 애니메이션 제작에도 사용할 수 있다.

이 처럼 여러 경우를 볼 때 PSD 포맷은 일반 작업에서는 물론 어도비계열의 소프트웨어를 많이 사용하는 학교 교육에 있어서 매우 유연한 작업환경을 구성해 주는 아주 뛰어난 포맷으로 추천한다. 마지막으로 이미지 포맷의 특성을 잘 파악하여 프로덕션 작업에서, 또는 학교 교육에서 사용한다면 컴퓨터를 활용한 애니메이션 제작에서 효율적인 작업공정으로 보다 능률적인 작업 환경을 구성할 수 있을 것으로 기대 한다.

참고문헌

- 신동일의, 『컴퓨터 사이언스 PC와 인터넷 중심으로』, 싸이텍미디어, 2003.
- 박성태외, 『정보통신 개론』, 생능출판사, 2001.
- 식송우연, 『문서 데이터 압축 알고리즘 입문』, 성안당, 1995.
- 이종명, 『과학이 세상을 지배한다』, 한림원, 2006.
- 김명찬외, 「새로운 비손실 압축 기법에 관한 연구」, 『창원대학교산업기술연구소 논문집』, 제14호, pp57-68
- 『britannica 백과사전』, 2007년 한글판 CD HX.
[http://en.wikipedia.org/wiki/ZIP_\(file_format\)](http://en.wikipedia.org/wiki/ZIP_(file_format))
- <http://en.wikipedia.org/wiki/Lzw>
- 정채창, 『MPEG와 정보혁명』, 전자신문 1995년 6월 16일

ABSTRACT

The image format research which is suitable in animation work

Kwon, Dong-Hyun

The computer has become an indispensable tool for animation works. However if you don't understand the characteristics of the computer and its software, you might not have the result satisfying your efforts. The incorrect understanding of image format sometimes causes it. Habitually image formats are selected usually for most of works but there is a distinct difference among those image formats while the efficient usages of them are different from each other. For your more efficient work therefore, you need to identify the characteristics of various kinds of image format used mostly for animation works. First I took a look at the theories of the lossy compression and lossless compression, which are two types of data compression widely used in the whole parts of computer world and the difference between bitmap method and vector method, which are respectably different in terms of the way of expressing images and finally the 24 bit true color and 8 bits alpha channel. Based on those characteristics, I have analyzed the functional difference among image formats used between various types of animation works such as 2D, 3D, composing and editing and also the benefits and weakness of them. Additionally I've proved it is wrong that the JPEG files consume a small space in computer work. In conclusion, I suggest the TIF format as the most efficient format for whatever it is editing, composing, 3D and 2D in considering capacity, function and image quality and also I'd like to recommend PSD format which has compatibility and excellent function, since the Adobe educational programs are used a lot for the school education. I hope this treatise to contribute to your right choice of image format in school education and practical works.

Key Word : lossless compress, loss compress, TIF, PSD, JPG, alpha channel, image format

권동현

조선대학교 만화애니메이션학과 전임강사

(100-250) 서울특별시 서초구 방배2동 2626번지 래미안 방배1차 아파트 106동 302호

Tel : 010-6249-1678

msz0099@paran.com