

스마트 시대의

멀티미디어 기본 이론부터
최신 기술 동향까지



멀티미디어

Chapter 05. 이미지

목차

1. 이미지의 개요
2. 디스플레이 장치의 이미지 표현
3. 이미지의 압축과 저장
4. 이미지 처리
5. 이미지와 그래픽 편집 소프트웨어

1.1 이미지의 개요

■ 이미지의 개념

- '이미지(Image)'는 영상, 심상, 그림 등으로 번역됨
- 이미지는 시각 이미지와 청각 이미지로 구분됨
 - 시각 이미지 : 이미지에 대한 지식이 없어도 그 자체를 직접적으로 경험할 수 있음
 - 청각 이미지 : 언어나 음악으로 표현되어 대상과 일정한 거리를 유지하게 만들

시각적 이미지에 비해 간접적

→ 모든 이미지는 전달하고자 하는 정보를 구체화함으로써 내용을 보다 선명하게 인식할 수 있음



▲ 이미지가 주는 메시지와 정서적인 반응

1.2 멀티미디어 환경에서 이미지

■ 멀티미디어 환경에서 이미지

- GUI 기반의 윈도우 운영체제가 시각적인 요소를 화면에 배치하면서 시작됨
- 멀티미디어 환경을 구성하는 여러 데이터 중 이미지는 특히 중요한 위치를 차지
- 텍스트로 전달하기 어려운 내용을 **그림이나 그래프**로 쉽고 명확하게 전달이 가능함
- 인간이 받아들이는 정보의 80% 이상이 시각을 통해서 인지함
- 입력된 이미지는 기억장치에 저장되고 처리(전송, 복사 등)됨
- 이미지는 **픽셀(Pixel)** 단위로 표현됨
- 효율적인 처리와 저장을 위해 이미지 **압축**(Image Compression)이라는 과정을 거침

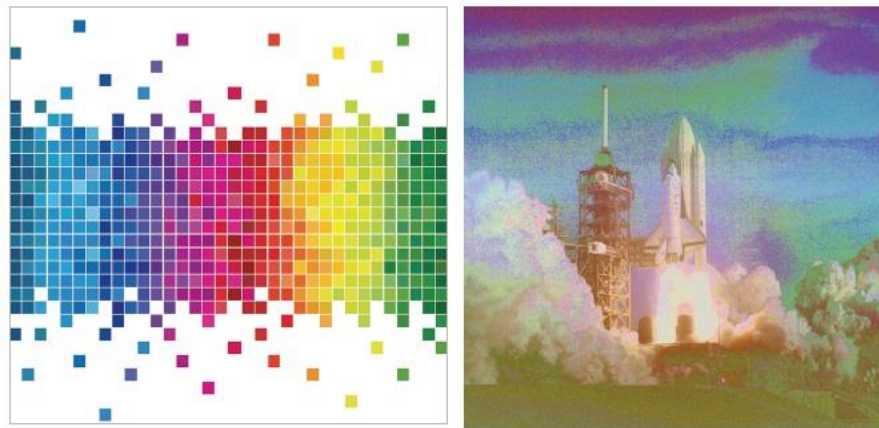


그림 5-2 픽셀로 표현된 이미지의 색상

2.1 디스플레이 장치의 이미지 표현

■ 픽셀과 해상도

- 해상도(Resolution)란 디스플레이 장치나 인쇄물에서 이미지의 정밀도를 나타내는 지표
- 단위는 인치당 픽셀 수를 사용
- 하나의 픽셀에는 이미지를 표현하기 위한 색상(Color) 정보가 저장됨
- 디스플레이 장치의 해상도는 **PPI(Pixel Per Inch)**, 인쇄물의 해상도는 **DPI(Dot Per Inch)**를 사용

■ 픽셀과 이미지의 관계

- 인치당 픽셀의 개수가 많을수록 이미지는 더욱 선명해짐
 - 픽셀 수가 많을수록 메모리 용량을 많이 차지하게 되어 컴퓨터 속도가 느려짐
- 원래 이미지가 손상되지 않도록 이미지 사용 목적에 맞게 적절한 해상도를 적용하는 것이 바람직함

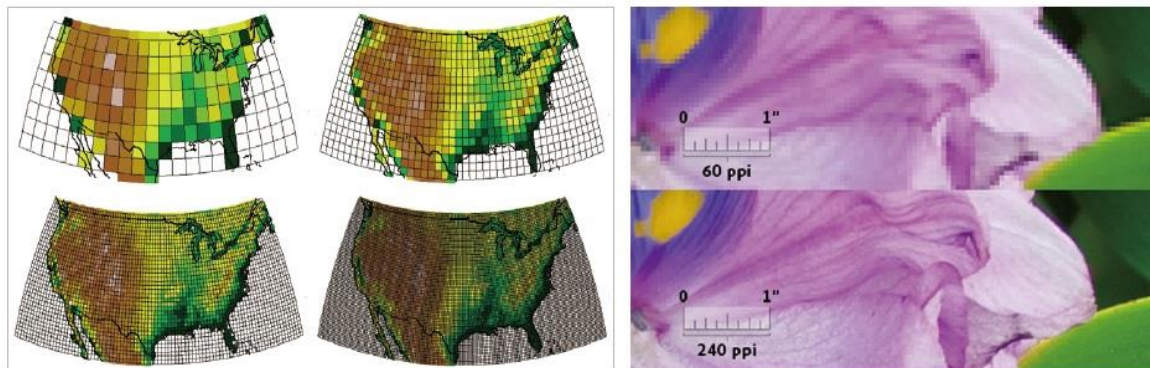


그림 5-3 픽셀과 해상도와의 관계 : 해상도가 높을수록 더 선명하다

2.1 디스플레이 장치의 이미지 표현

■ 픽셀을 구성하는 비트

- 픽셀의 색상 수는 픽셀을 구성하는 비트의 수로 결정됨

표 5-1 픽셀을 구성하는 비트의 개수와 색상의 종류

비트수	색상의 종류	기타
1	$2^1 = 2$	흑백
2	$2^2 = 4$	팔레트
4	$2^4 = 16$	팔레트
8	$2^8 = 256$	팔레트
16	$2^{16} = 65,536$	하이컬러(RGB 각 5비트)
24	$2^{24} = 16,777,216$	트루컬러(RGB 각 8비트)
32	$16,777,216 + 8\text{비트 알파채널}$	트루컬러+알파채널

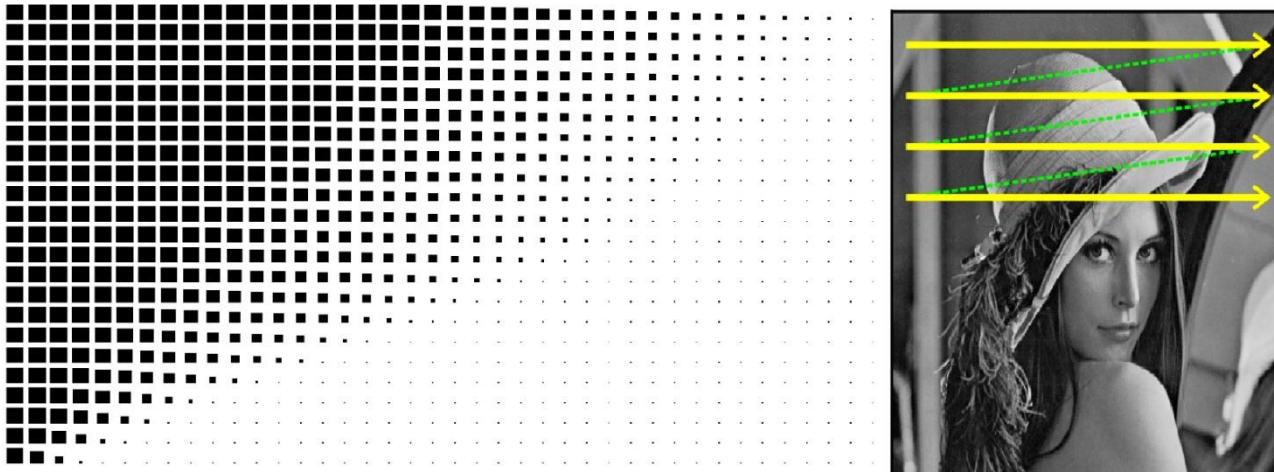
:: 여기서 잠깐 디지털 카메라와 컴퓨터의 픽셀 비교

디지털 카메라는 필름 대신 CCD(Charge Coupled Device)나 CMOS(Complementary Metal Oxide Semiconductor)라는 이미지 센서를 이용하여 영상을 저장한다. 렌즈를 통해 들어오는 영상에 대한 빛을 수치적인 계산 과정을 통해 이미지 픽셀로 변환시킨다. 반대로 컴퓨터의 모니터는 빛의 3원색인 빨강, 녹색, 파랑(RGB)의 디지털 신호를 통해 각각의 픽셀에 색상을 나타낸다.

2.2 비트맵 방식과 벡터 방식

■ 비트맵 방식

- 이미지를 픽셀 단위로 분해하여 각 픽셀의 색상과 위치를 저장하는 방식
- 화면에 표시되는 이미지와 메모리에 저장되어 있는 이미지의 비트 대 비트 형태가 일치
- 이미지를 저장하는 데 필요한 메모리 용량이 크고 디스플레이 부분의 하드웨어가 복잡함
- 레스터(Raster) 이미지라고도 함
 - 래스터는 수평 주사선을 구성하는 연속적인 픽셀의 집합을 의미



▲ 래스터 방식의 개념(왼쪽)과 래스터 스캔의 형태(오른쪽)

2.2 비트맵 방식과 벡터 방식

■ 비트맵 방식

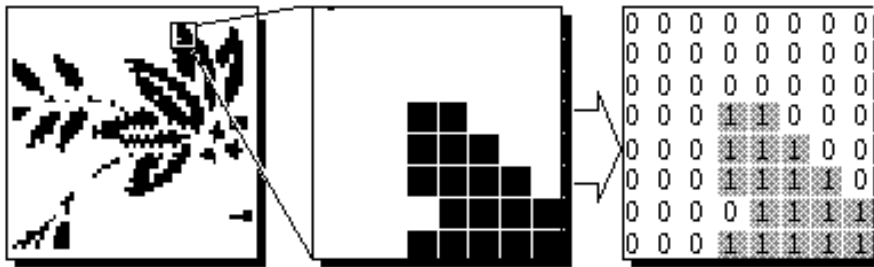
- 이미지를 표현하는 구조가 단순하기 때문에 화면에 이미지를 나타내는 속도가 벡터 이미지에 비해 빠름
- 2차원 이미지의 각 픽셀에 대한 정보는 $I(r,c)$ 로 표현(I 는 명도(Brightness), r 과 c 는 각각 행과 열을 의미)
- 사진이나 회화 이미지를 표현할 때 많이 사용
- 각각의 픽셀은 하나의 색상을 표현하기 위해 적색(Red), 녹색(Green), 청색(Blue)의 값을 적절히 배합하여 색을 표현
- 윈도우의 비트맵 확장자는 .bmp
- GIF, JPEG, PNG, TIFF, PCT, PCX 등으로 저장된 파일은 모두 비트맵 방식임
- 비트맵 방식의 대표적인 이미지 처리 프로그램으로는 포토샵, 페인터, 코렐 페인터 등이 있음

2.2 비트맵 방식과 벡터 방식

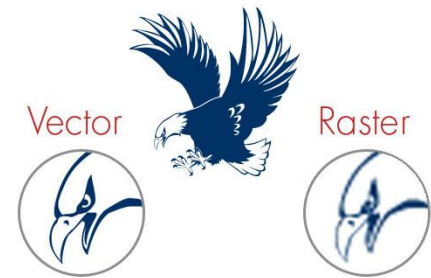
■ 비트맵 방식의 단점

- 각각의 픽셀에 명암과 색상을 정보를 모두 저장해야 하기 때문에 파일의 용량이 증가함
- 이미지를 확대하거나 축소할 경우 이미지의 외곽선 부분이 계단 모양으로 변형됨

→ 최근에는 이미지의 외곽선을 따라 점의 크기를 다르게 하여 외곽선 부분이 매끄럽게 보이도록 하는 기술
RET 사용



▲ 비트맵 이미지의 확대



2.2 비트맵 방식과 벡터 방식

■ 벡터 방식

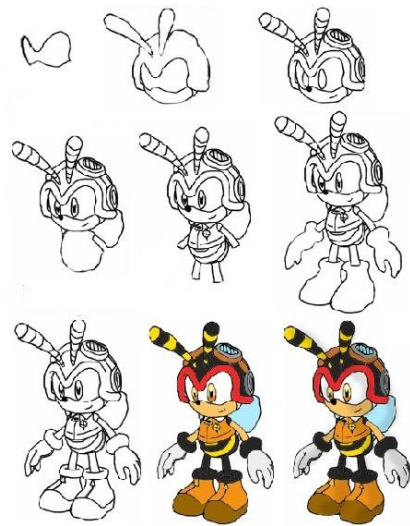
- 이미지를 수학적 공식으로 표현
- 이미지 객체의 위치와 기울기를 산술적 데이터로 기록하고, 외곽선을 만들고, 그 내부에 색상이나 패턴을 적용시켜 표현
- 간단한 도형, 글자, 로고, 캐릭터 디자인에 주로 사용됨
- 평면 또는 공간상에 특정한 좌표 값을 가지고 표현됨
- 정해진 공식에 따라 처리되기 때문에 파일 크기가 비트맵 이미지에 비해 작음
- 파일 크기와 이미지의 화질(선명도)을 손상시키지 않고 확대, 축소, 회전 등과 같은 다양한 방법으로 조작 가능
- 대표적인 벡터 드로잉 프로그램은 일러스트레이터, 플래시, 코렐드로우, 프리핸드 등이 있음

■ 벡터 방식의 단점

- 사진과 같은 복잡한 이미지를 표현하기 힘들
- 확대할 때마다 외곽선을 매끄럽게 만들기 위해 연산과정을 거치기 때문에 처리 속도가 늦음

2.2 비트맵 방식과 벡터 방식

■ 벡터 방식



▲ 벡터 방식으로 이미지를 표현하는 과정

2.2 비트맵 방식과 벡터 방식

■ 비트맵 방식과 벡터 방식의 차이점

표 5-2 비트맵 방식과 벡터 방식의 차이점

	비트맵 방식	벡터 방식
프로그램 구분	이미지 프로세싱 계열	드로잉 계열
이미지 처리 방식	픽셀로 이미지 표현	수학적 함수로 계산하여 이미지 표현
장점	사진, 회화 등 복잡한 이미지 표현 가능	<ul style="list-style-type: none">• 파일의 크기가 작음• 확대하거나 축소해도 이미지의 변형 없음
단점	<ul style="list-style-type: none">• 이미지 크기에 비례하여 파일 용량 증가• 이미지를 확대하거나 축소하면 계단현상 발생	복잡한 이미지 표현이 어려움
프로그램 종류	포토샵, 페인터, 코렐 페인터 등	일러스트레이터, 플래시, 코렐드로우, 프리핸드 등

2.3 이미지 표현을 위한 컬러 모델

■ 컬러 모델의 개요

- 인간의 시각 체계 : 인간의 눈에서 인지하는 형태
 - 조도(Illumination) : 빛 에너지의 물리적인 척도로 **빛의 밝기 또는 세기**를 의미
 - 휘도(Brightness) : **물체에서 반사되는 빛의 세기**로 반사체 표면의 밝기를 나타내는 양을 의미

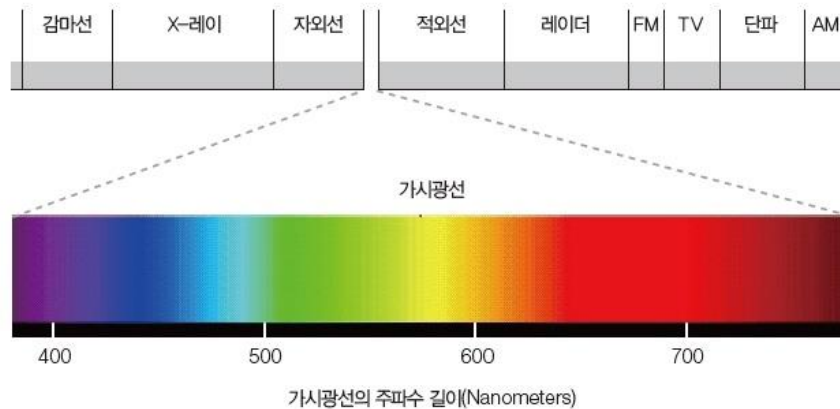
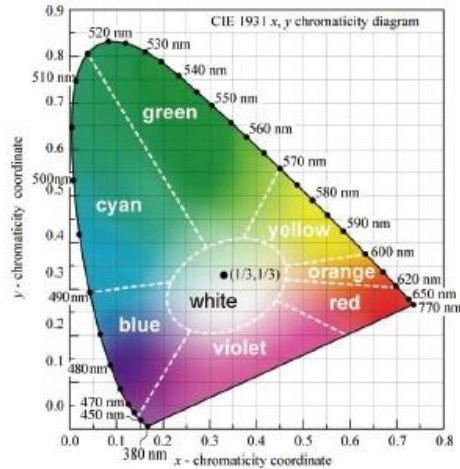


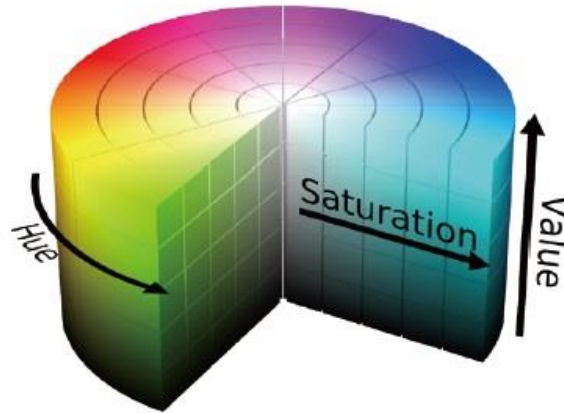
그림 5-8 빛의 주파수 대역과 가시광선의 스펙트럼

- 컬러 모델(Color Model)은 컴퓨터 환경에서 효율적인 컬러 영상 처리를 목적으로 국제조명위원회에서 개발
- 컬러 모델은 컴퓨터에서 각각의 색상에 대해 규정하고 재현할 수 있는 색 공간(Color Space)을 정의

2.3 이미지 표현을 위한 컬러 모델



(a) 2차원의 색도



(b) 3차원의 색 공간

그림 5-9 색의 상대적인 위치와 상호관계를 나타내는 색 체계

■ 대표적인 컬러 모델

- 가산 혼합과 감산 혼합의 원리로 색을 표시하는 RGB와 CMYK 모델
- 색상, 채도, 명도와 같은 색의 삼속성을 기반으로 하는 HSV(HSB), HSL 모델

2.3 이미지 표현을 위한 컬러 모델

■ RGB 컬러 모델

- 빛의 3원색인 빨간색(Red), 녹색(Green), 파란색(Blue)으로 색을 정의
- 세가지 색을 혼합하여 원하는 색을 만들며, 색을 섞을수록 밝아져서 가산 혼합(Additive Color Mixing)이라고 함
- 주로 컬러 TV, 컬러 모니터 등에 이용됨
- 컴퓨터 화면을 구성하는 픽셀은 빨간색, 녹색, 파란색 각각에 한 바이트를 할당하여 모두 24비트로 구성됨

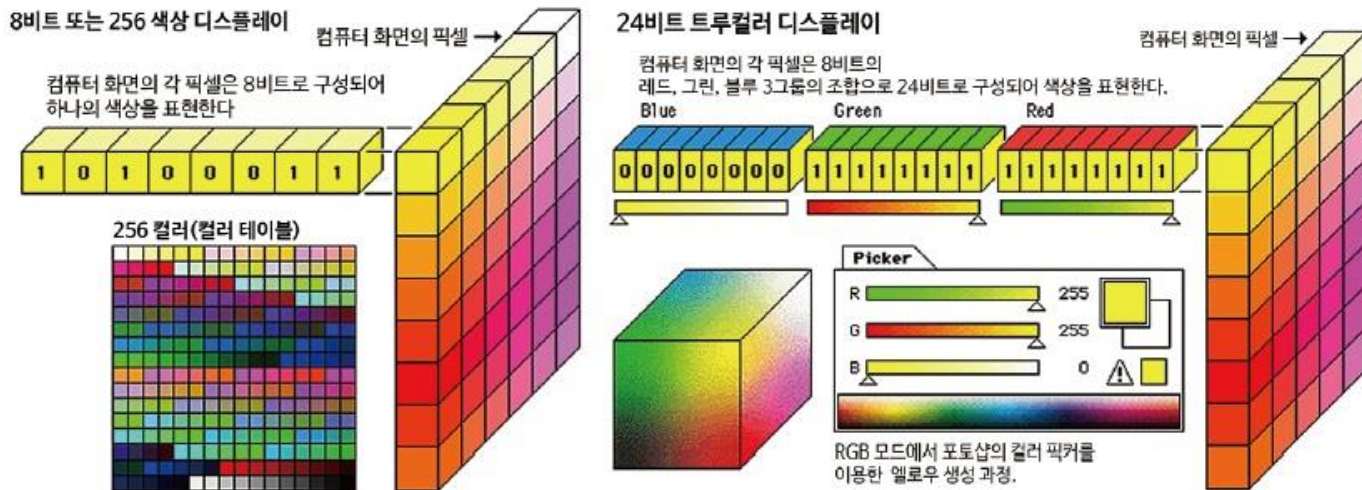


그림 5-10 8비트의 팔레트 컬러와 24비트의 트루컬러

2.3 이미지 표현을 위한 컬러 모델

■ CMYK 컬러 모델

- 인쇄 매체에서 사용되는 것으로, 청록색(Cyan), 선홍색(Magenta), 노란색(Yellow), 검은색(Black)으로 색을 정의
- 물감이나 잉크 같은 안료가 섞일 때 나타나는 색의 혼합을 이용하는 방식
- 색을 섞을수록 어두워지기 때문에 감산 혼합(Subtractive Color Mixing)이라고 함
- CMY 컬러 모델에 **K가 추가된 모델**(이론적으로 모든 색의 표현은 CMY만으로도 가능함)
- CMYK 컬러 모델은 컬러 프린터, 인쇄, 페인팅 등에서 이용

■ CMYK 컬러 모델을 사용하는 이유

- CMY를 혼합하여 순수한 검은색을 만드는 것이 불가능하기 때문
- CMY 컬러 모델의 세 가지 잉크를 종이에 뿌리면 건조 시간이 오래 걸려 인쇄 속도가 느려짐
- 문자를 출력하기 위해서 세 가지 잉크의 분사 위치가 매우 정확하게 이루어져야 함
- 검은색 잉크의 가격이 컬러 잉크보다 훨씬 싸기 때문에 비용을 절감할 수 있음

2.3 이미지 표현을 위한 컬러 모델

■ RGB vs CMYK

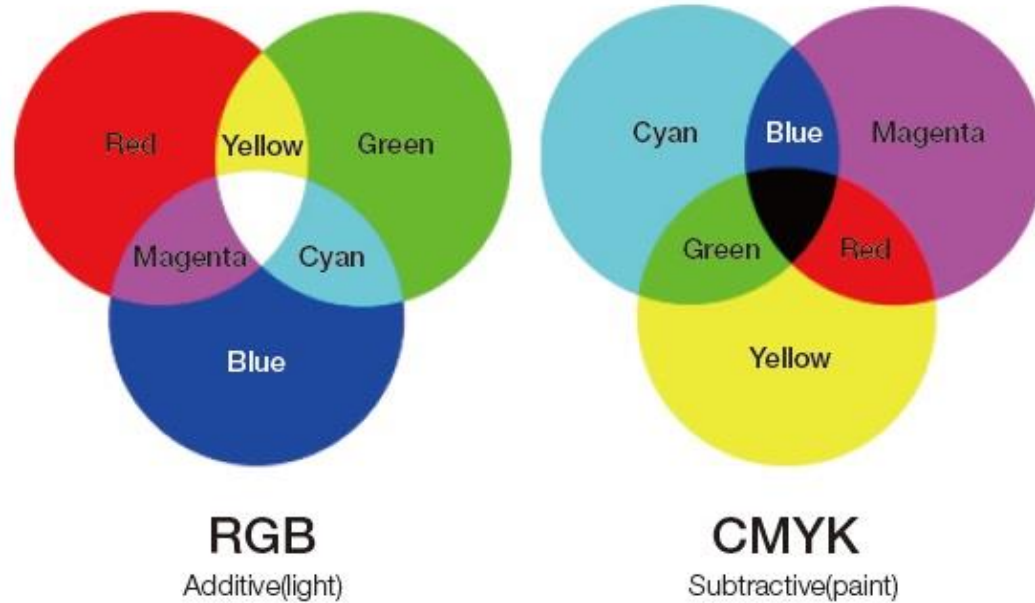


그림 5-11 RGB 컬러 모델과 CMYK 컬러 모델

2.3 이미지 표현을 위한 컬러 모델

■ HSV와 HSL 컬러 모델

- HSV, HSL 컬러 모델은 사람이 감성적으로 색을 인식하는 방법과 가장 가깝게 구성됨
- HSV, HSL 컬러 모델은 이미지의 색상, 채도, 명도 정보를 이용하여 색을 표현
 - 색상(Hue) : 해당하는 색의 원색을 나타냄
 - 채도(Saturation) : 색의 순수한 정도 즉, 진하기 정도를 나타냄
 - 명도(Value, Brightness) : 색깔의 밝은 정도를 나타냄

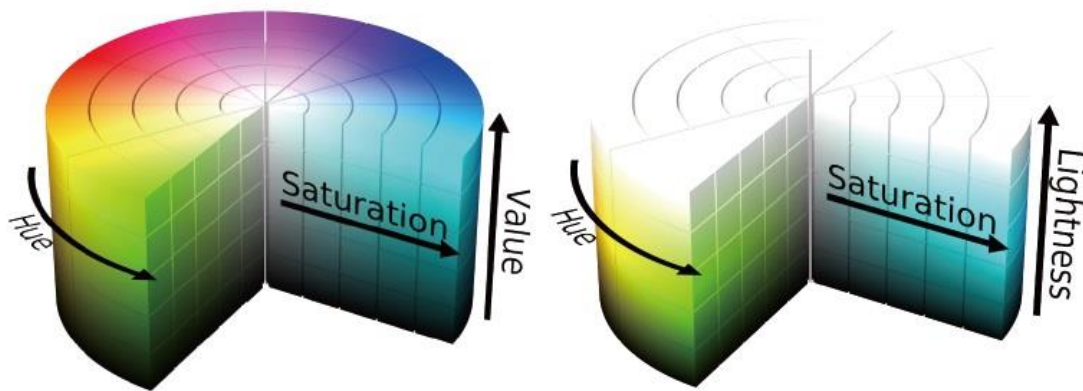


그림 5-12 HSV 컬러 모델과 HSL 컬러 모델

2.3 이미지 표현을 위한 컬러 모델

■ HSV 컬러 모델

- 원통 또는 원뿔 모양의 좌표계를 사용하여 표현
- 색상은 둥근 원으로 $0^{\circ} \sim 360^{\circ}$ 사이의 일정한 각도로 분할하여 표현
- 채도는 원통 또는 원뿔 중심에서부터 바깥쪽의 수평 거리로 표현
- 명도는 원통 또는 원뿔의 세로축을 의미

■ HSL 컬러 모델

- HSV 컬러 모델과 유사한 모델로, 원뿔 두 개를 서로 이어 붙여 색 체계를 나타냄
- 명도를 고정시키고 채도를 변경함
- 밝은 색과 어두운 색이 색 공간에서 서로 대칭적으로 표현됨

2.3 이미지 표현을 위한 컬러 모델

■ RGB와 HSL 색 편집 대화상자



그림 5-13 윈도우에서 제공하는 RGB와 HSL 색 편집 대화상자

2.3 이미지 표현을 위한 컬러 모델

■ 인덱스 컬러(Indexed Color) 모델

- 256가지 이하의 색상을 사용하여 이미지를 표현
- 각 픽셀은 8비트의 색상 정보를 가지는데, VGA 그래픽 카드가 256 색상만 지원할 때 컬러를 표현하기 위해 사용
- 1,677만 개의 색상 집합을 가진 RGB 컬러 모델(픽셀당 24비트)에서 256개의 색상만 제한하여 사용
- 색상의 표현 범위가 좁은 대신 파일의 용량이 작음
- 인덱스 컬러 모델을 사용하는 대표적인 이미지 파일 포맷은 GIF 포맷
- TIF, JPG, PNG 포맷에서도 파일의 용량을 줄이기 위해 사용



그림 5-14 단계별 컬러의 비교

2.3 이미지 표현을 위한 컬러 모델

■ 24비트 컬러를 인덱스 컬러로 변환하는 방법

- 인접한 색상을 선택하는 방법
 - 팔레트에 정확한 색상이 존재하지 않을 경우 해당 색상과 유사한 색상을 선택하여 나타내는 방법
 - 정확한 색상이 중요하지 않은 경우에 사용
- 디더링(Dithering) 기법
 - 팔레트에 존재하지 않는 색을 존재하는 컬러 패턴을 이용하여 가장 유사하게 표현하는 기법
 - 제한된 색의 도트를 조합하여 음영 또는 색상을 나타냄(분홍색의 경우 빨간색과 흰색의 도트 조합으로 만들)
 - 도트의 조합으로 구성되어 다소 거칠게 보인다는 단점이 있음

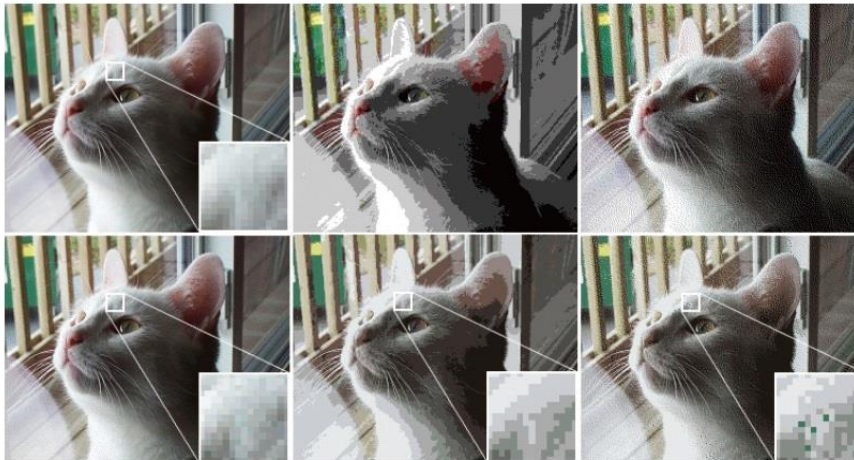


그림 5-15 원본 사진과 웹 컬러 팔레트를 사용해 디더링을 수행한 결과 [01]

3.1 이미지의 압축과 저장

■ Raw 이미지 포맷

- 디지털 카메라 또는 스캐너의 이미지 센서가 촬영한 정보에 최소의 처리 과정만 거친 데이터를 의미
- **전혀 가공되지 않은 상태**이며 그래픽 편집기 또는 인쇄 등에 사용할 수 없음
- 이미지 센서에 의해 감지된 빛의 세기에 대한 정보만 가지고 있음
- 파일의 확장자는 디지털 카메라 제조사에 따라 여러 가지가 사용됨
- **사진을 찍으면 이미지 센서에 의해 감지된 이미지 정보가 JPEG 파일로 변환됨**
- Raw 파일을 이미지로 변환하기 위해서는 사용자가 별도의 처리 과정을 거쳐야 함



▲ Raw 이미지와 JPEG 이미지의 비교

3.1 이미지의 압축과 저장

■ Raw 이미지의 장점

- JPEG 이미지보다 **화질이 좋음**
- Raw 이미지는 압축을 하지 않거나 무손실 압축을 사용하므로 이미지를 **항상 원본 형태로 유지**할 수 있음
- Raw 변환 소프트웨어를 사용하면 섬세한 제어가 가능
- Raw 파일은 12비트나 14비트의 명암 정보를 가지고 있고 그림자, 밝은 부분, 채도가 깊은 색에 더 정확성을 제공함
- 사용자가 원하는 대로 색 공간을 설정할 수 있음

■ Raw 이미지의 단점

- JPEG 파일보다 일반적으로 2~6배 정도 **파일 용량이 큼**
- 이미지를 메모리 카드에 저장할 때 JPEG 보다 시간이 많이 걸림
- 표준 Raw 포맷이 별도로 지정되어 있지 않음
- 다른 표준 포맷에 비하여 아직 활성화가 되지 않았기 때문에 Raw 포맷을 지원하는 **별도의 소프트웨어가 필요함**

3.2 비트맵 이미지 포맷

■ 비트맵 이미지 포맷

- 마이크로소프트사에서 개발
- 윈도우 OS의 그래픽, 이미지를 저장하는 기본 포맷으로 사용됨
- 확장자는 .bmp
- 압축되지 않은 비트맵 이미지를 저장
- 이미지를 저장하는 파일 형식 중 구조가 가장 단순해 빠른 이미지 처리를 위한 전용 포맷으로 사용
- 이미지를 확대하거나 축소하기 어려움
- 비트맵 이미지 포맷은 모든 픽셀의 컬러 값을 그대로 저장함
- 파일을 저장할 때 다른 포맷으로 저장하는 것보다 파일 용량이 매우 커짐
- 비트맵 이미지 파일은 비트맵 파일 헤더, 비트맵 정보 헤더, 비트맵 팔레트, 이미지 픽셀 데이터 등으로 구성됨

3.2 비트맵 이미지 포맷

■ 비트맵 이미지 포맷



그림 5-17 비트맵 이미지의 압축 결과 - 왼쪽 위부터 1비트, 4비트, 8비트, 16비트, 24비트, GIF

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 표준

- 높은 화질, 적은 용량, 자유로운 사용성을 만족하는 디지털 이미지 표준에 대한 요구로 탄생
- 현재 웹사이트, 인쇄, 출판, 광고 등에서 가장 보편적으로 사용되는 이미지 포맷
- RGB, CMYK 컬러 모델을 지원
- 이미지 데이터를 압축, 저장하여 이미지의 이동, 복사, 전송, 저장, 재생 능력을 향상시킴
- 품질 저하가 발생하는 손실압축 방식을 사용하지만, 손실 없이 최대 25:1까지 압축이 가능
- 손실을 감수하면 최대 100:1까지도 가능함
- 적은 파일 용량으로 품질이 우수한 이미지를 표현할 수 있기 때문에 사진 압축에 많이 사용됨
- 선이나 문자, 세밀한 격자 등이 다수 포함된 이미지는 GIF나 PNG 같은 비손실 압축 표준을 사용하는 것이 바람직함
- 압축률과 화질을 향상시킨 'JPEG 2000'이 발표되었으나 JPEG에 밀려 널리 사용되지는 못함
 - 일부 운영체제가 지원하지 않고, 별도의 프로그램을 설치해야 하는 불편함

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 표준



그림 5-19 JPEG 표준의 압축률과 화질의 관계 [03]

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 표준



JPEG



JPEG 2000

3.3 JPEG(JPG) 이미지 포맷

∴ 여기서 잠깐 이미지 데이터의 용량을 감소시키는 방법

이미지 데이터의 용량을 감소시키는 방법으로 다음과 같이 세 가지를 들 수 있다. 앞의 두 가지 방법은 픽셀 개수와 비트 개수를 감소시키기 때문에 이미지의 물리적인 변형이 발생한다.

- ① 하나의 픽셀을 구성하는 비트 개수를 감소시키는 방법
 - ② 이미지를 구성하는 픽셀 전체 개수를 감소시키는 방법
 - ③ 압축 기법을 사용하여 파일 용량을 감소시키는 방법
-

■ JPEG 표준 압축 알고리즘

- 무손실 압축(Lossless Compression) 방식과 손실 압축(Loss Compression) 방식을 모두 지원함
 - 무손실 압축 방식 : 압축된 데이터를 다시 복원했을 때 이전의 데이터와 모든 비트가 일치함
 - 손실 압축 : 저장 용량을 감소시키기 위해 압축 과정의 손실을 감수하고 사용하는 방법
 - 사람의 눈에 민감한 밝기 정도를 나타내는 '휘도(음영)' 정보는 유지시키고, '색상(Low-Pass)'
 - 부분만을 중점적으로 압축

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 알고리즘을 사용하여 압축하는 과정



그림 5-21 JPEG 압축 알고리즘과 복호화 알고리즘

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 알고리즘을 사용하여 압축하는 과정

⊖ 색상 변환(Color Transformation)

- 이미지의 RGB 컬러 모델을 YIQ 컬러 모델로 변환하는 과정
- YIQ 컬러 모델은 컬러 텔레비전에서 사용하는 모델로 색상보다 밝기에 더 민감한 시각을 고려한 원리
- 이 과정에서는 정보 손실이 발생하지 않음

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

그림 5-22 RGB 컬러 모델을 YIQ로 변환하는 공식

⊖ 다운 샘플링(Down Sampling)

- 전 단계에서 변환된 YIQ 컬러 모델을 YCbCr 컬러 모델로 변환하는 과정(Y는 밝기, Cb, Cr은 색상을 나타냄)
- 이 과정에서 Y 성분은 그대로 유지하고 색상 정보인 I와 Q 값을 감소시킴
- 이미지 크기는 1/4. 또는 1/2 크기로 감소되며 손실이 발생함

3.3 JPEG(JPG) 이미지 포맷

■ JPEG 알고리즘을 사용하여 압축하는 과정

⊗ DCT(Discrete Cosine Transformation) 진행

- 2차원 이미지 공간의 컬러 정보를 2차원의 주파수 정보로 변환하는 과정
- 중요하지 않은 부분이 손실되기 때문에 전체 데이터 용량이 감소함
- 이미지를 구성하는 64개(8×8) 픽셀 블록을 샘플링 함

④ 양자화(Quantizaion)

- DCT 변환으로 발생한 64개의 DCT 계수에서 불필요한 고주파 잡음들을 제거하는 과정
- 미리 정의된 상수로 나누고 그 결과를 정수값으로 반올림함
- 데이터 압축이 가장 큰 동시에 데이터 손실이 가장 많이 발생함

⑤ 부호화(Encoding)

- 양자화 과정을 거친 데이터를 부호화 알고리즘으로 다시 압축하는 작업
- 모든 데이터에 대해 나타날 확률을 따져 서로 다른 길이의 코드를 배정함
(자주 나타나는 데이터의 코드 길이가 짧음)

3.4 GIF 이미지 포맷

■ GIF(Graphics Interchange Format) 이미지 포맷

- 온라인 전송을 위해 만들어졌으며 무손실 압축 방식인 LZW(Lempel-Ziv-Welch) 알고리즘을 사용함
- 1.5:1~2:1의 비율로 이미지를 압축
- 이미지 품질의 손상 없이 파일 용량을 원본의 **최대 40%까지 감소**시킬 수 있음
- 사진의 경우 압축 효과가 크지 않으나 일러스트레이터로 제작된 그래픽 파일은 압축 효과가 큼
- 웹페이지에서 간단한 작업으로 다양한 효과를 줄 수 있고 특별한 플러그인을 요구하지 않아 여러 환경에서 쉽게 사용
- 1989년 새로운 기능을 추가한 GIF89a 형식을 발표, 현재 두 개의 GIF 버전을 구분하지 않고 사용함



3.4 GIF 이미지 포맷

■ GIF 이미지 포맷의 단점

- 최대 256가지 색상을 지원하는 이미지만 구현할 수 있음
- 압축 기술의 특허 문제로 널리 사용하기에는 한계가 있음
 - 정밀한 사진보다 단순한 색상으로 이루어진 선, 아이콘 등에 사용하면 유용함

■ GIF89a 형식에 새로 추가된 기능

- 인터레이싱 기능
 - 하나의 화면을 짝수 줄의 화면과 홀수 줄의 화면으로 분리하여 교대로 주사하여 재생하는 방법
 - 이미지를 낮은 해상도에서부터 시작하여 서서히 선명하고 뚜렷한 이미지로 표현함

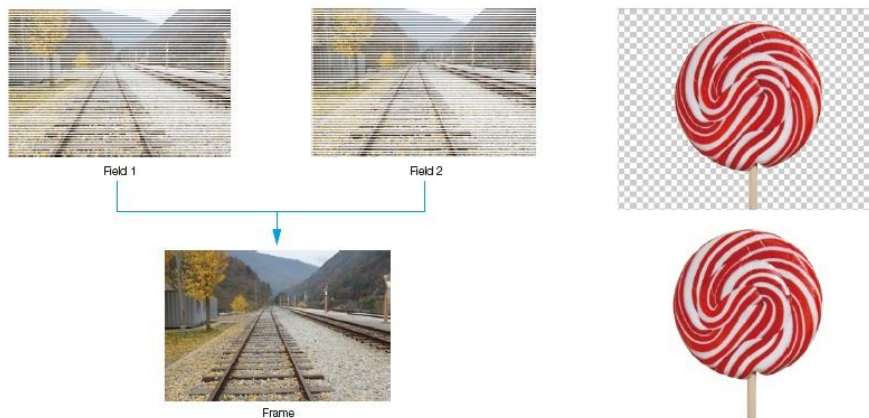


그림 5-24 인터레이스 GIF와 투명 GIF

3.4 GIF 이미지 포맷

■ GIF89a 형식에 새로 추가된 기능

- 투명 기능
 - 특정한 색을 투명으로 지정하여 다른 색의 배경 위에 겹쳐서 표현하는 기능
 - 이미지가 해당 페이지 위에 떠 있는 것 같은 시각적인 효과를 낼 수 있음
- 애니메이션 기능
 - 이미지를 구성하는 개체가 살아 움직이는 것처럼 이미지를 재생하는 기술
 - 하나의 파일에 여러 장의 이미지 프레임들을 저장해서 순서대로 번갈아 가며 재생시켜 움직이는 것처럼 보이게 함
 - 간단하게 만들 수 있는 장점, 이미지의 움직임, 유연성, 재생 속도 등이 다소 떨어지는 단점이 있음



그림 5-25 GIF 애니메이션의 예

3.5 PNG 이미지 포맷

■ PNG(Portable Network Graphics) 이미지 포맷

- GIF 형식을 대체하기 위해 1995년에 개발
- 24비트의 트루 컬러(True Color)를 지원하고 무손실 압축 방식을 사용하기 때문에 이미지를 손상시키지 않음
- PNG 포맷은 GIF보다 압축 효율이 10~30% 정도 향상되어 GIF보다 적은 용량으로도 이미지 표현이 가능
- GIF의 특징인 인터레이싱(Interlacing) 기능과 투명 기능도 제공
- JPG와 GIF의 장점을 모두 갖추고 있으며, 화질은 BMP 정도의 높은 수준을 유지
- 파일 용량이 JPEG나 GIF보다는 크다는 단점이 있음



그림 5-26 GIF, PNG, JPEG 형식의 파일 용량 비교 [06]

3.5 PNG 이미지 포맷

■ PNG 포맷의 특징

- GIF보다 압축률이 더 높음
- GIF처럼 단색 투명층이 아니라, 투명도 자체를 조절할 수 있는 부드러운 투명층을 지원함
- 인터레이싱 이미지 재생 기능이 제공되며, GIF 형식보다 더 빠르게 나타남
- 인터넷에서 이미지 표시를 염두에 두고 개발되었기 때문에 CYMK 등과 같은 색 공간은 지원하지 않음
- 애니메이션 기능을 지원하지 않음

4.1 이미지 처리

■ 이미지의 획득과 처리

- '이미지 처리'는 이미지에 대한 사람들의 이해를 돕기 위해 컴퓨터를 이용하여 이미지를 생성, 조작, 개선, 해석, 분석, 인식하는 기술뿐만 아니라 이미지 데이터에 포함되어 있는 왜곡이나 잡음을 제거하고 이미지로부터 정보를 도출하는 과정도 포함됨

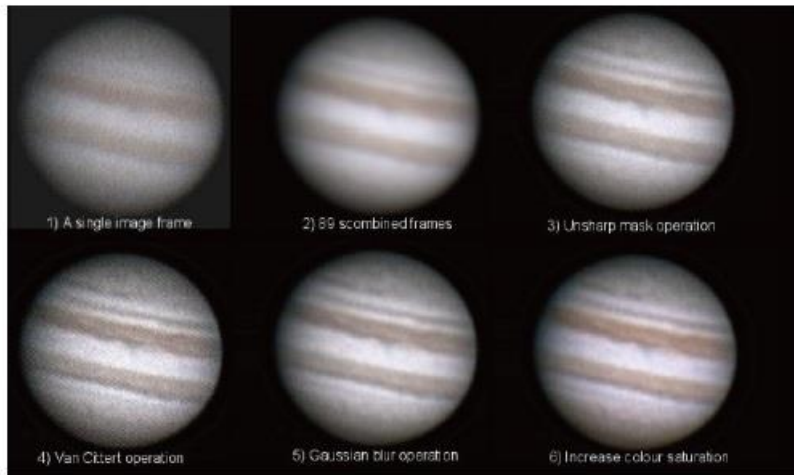


그림 5-27 이미지 처리의 대표적인 사례 : 선명한 우주의 모습 [07]

4.1 이미지 처리

■ 이미지 처리 기법 중 많이 사용되는 기법

- 이미지의 확대, 축소, 회전 등과 같은 기하학적 변환
- 이미지의 명도, 대비 등과 같은 색 보정 작업과 색 다른 색 공간으로의 색 변환
- 둘 이상의 영상을 결합하는 디지털 합성
- 둘 이상의 영상을 맞추는 영상 정합
- 영상의 변형
- 글자 인식, 얼굴 인식 등과 같은 영상 인식
- 영상 분할
- 2차원 물체의 인식

4.1 이미지 처리

■ 대표적인 이미지 처리 기술

- 평활화(Smoothing) : 손상된 이미지 부분을 최소화하여 부드럽게 만드는 기술
- 블러링(Blurring) 현상 제거 기술 : 이미지가 흐리게 나타나는 현상을 제거하는 기술
- 선명화 기술(Sharpening) : 뚜렷한 이미지를 얻기 위한 기술
- 필터링(Filtering) : 각종 잡음을 제거하는 기술
- 대비도 증가 : 이미지의 밝기나 색을 변화시키는 기술
- 윤곽선 추출기법
- 각종 패턴인식(Pattern Recognition) 기법

4.2 히스토그램과 이미지의 평활화

■ 히스토그램(Histogram)

- 이미지의 어두운 정도를 나타내는 개념으로 **이미지를 구성하는 픽셀에 대한 명암 값의 분포를 나타냄**
- 이미지 처리에서 일반적으로 많이 사용됨
- X축 : 이미지에 대한 256 단계(0~255)의 명암 값을 나타냄
- Y축은 각각의 명암 값(level)에 대한 빈도수를 나타냄
- 이미지 처리 과정에서 밝기를 조절함으로써 명도 값의 분포를 넓게 하여 좋은 품질의 이미지를 획득함



그림 5-28 히스토그램 그래프와 이미지의 상태 [08]

4.2 히스토그램과 이미지의 평활화

■ 히스토그램(Histogram)

- 히스토그램 그래프에서 **문턱치(Threshold)** 값을 설정하면 유용한 정보만을 추출할 수 있음
 - 문턱치 값 : 이미지와 배경을 분리할 수 있는 가장 적절한 값을 의미
- 컬러 이미지인 경우는 명암과 RGB 색상에 대해 4개의 히스토그램 그래프로 표현

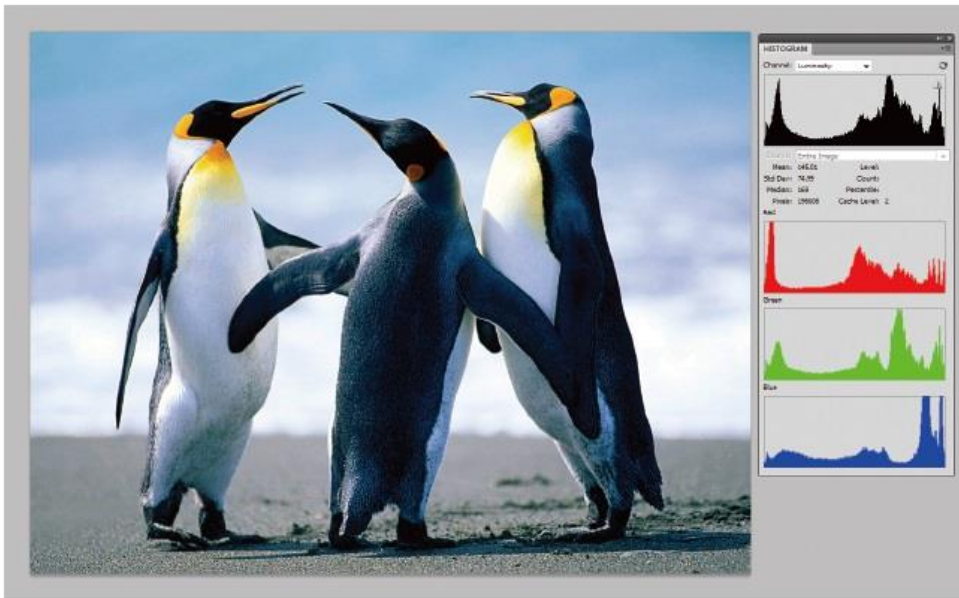


그림 5-29 컬러 이미지에 대한 히스토그램 그래프

4.2 히스토그램과 이미지의 평활화

■ 히스토그램 평활화(Histogram Equalization)

- 문턱치 값을 기준값으로 이미지에 대한 히스토그램 평활화를 수행함
- 이미지에서 명암도에 따른 픽셀의 개수를 고르게 분포시키는 기법
- 이미지 평활화를 위한 공식

$$\text{Histogram Equalization} = \frac{\text{Max GrayLevel}}{\text{TotalPixel 수}} \times \text{누적값}$$

- 히스토그램 평활화를 통해 이미지는 밝고 어두운 부분이 거의 균등하게 분포됨



그림 5-30 흑백 이미지에 히스토그램 평활화를 수행한 결과 [09]

4.3 이미지 필터링

■ 이미지 필터링(Image Filtering)

- 원래의 이미지에 특정한 마스크(Mask)를 필터(Filter)로 사용하여 처리하는 방법
- 3×3 , 5×5 , 7×7 등과 같은 홀수개의 픽셀들을 마스크로 사용함
- 3×3 마스크의 경우에는 중앙의 픽셀과 인접한 8개의 픽셀들을 이용하여 중앙의 픽셀 값을 결정
- **컨벌루션(Convolution) : 이미지에 마스크를 덮어서 연산하는 것**

이미지의 각각의 픽셀에 대해 인접한 주변의 픽셀들의 조합으로 대체하는 작업

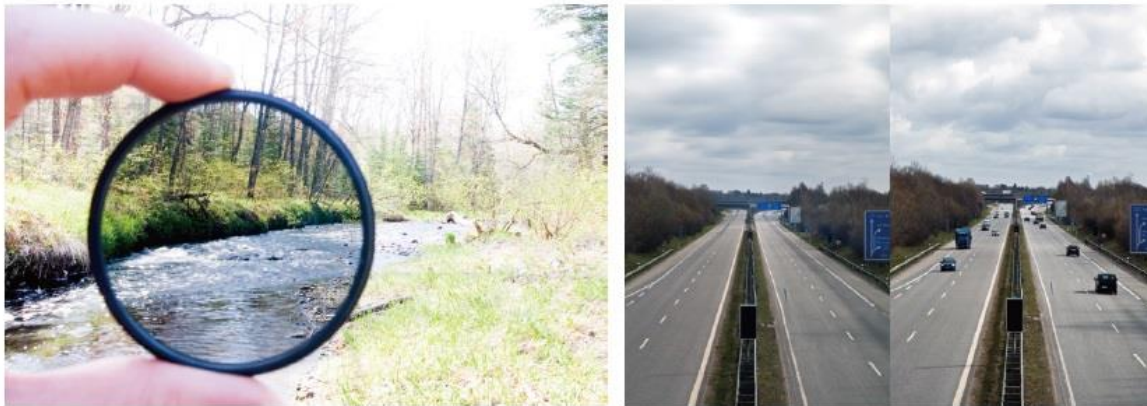


그림 5-31 아날로그 이미지 필터링과 디지털 이미지 필터링 [10]

4.3 이미지 필터링

■ 필터의 종류

- 이미지를 부드럽게 하는 저주파 통과 필터(LPF, Low Pass Filter)
 - 서서히 변하는 영역은 유지하고 급격하게 변하는 영역을 서서히 변경되도록 평균값으로 대체함
- 이미지를 뚜렷하게 하는 고주파 통과 필터(HPF, High Pass Filter)
 - 흐려진 영상을 개선하는데 사용



그림 5-32 원본 이미지(왼쪽 위)와 다양한 이미지 필터링 결과

4.4 이미지 처리 기법과 응용 분야

■ 이미지 처리 기법

표 5-3 이미지 처리 기법

이미지 분할 (Image Segmentation)	<ul style="list-style-type: none">• 이미지에서 사용 목적에 알맞은 영역을 검출하거나 객체의 일부분을 분리해 내는 것• 동질적인 부분과 이질적인 부분을 분리해 냄• 동질성, 대조도, 단색조, 컬러, 질감 등의 특성들을 이용하여 이미지를 분리함• 컴퓨터 비전(Computer Vision), 이미지 분석(Image Analysis), 이미지 처리에서 매우 중요한 기술임
이미지 복원 (Image Restoration)	<ul style="list-style-type: none">• 외부 요인에 의해 손상된 이미지를 대수학적인 방법이나 주파수 영역의 기법 등을 이용하여 원래의 이미지로 만드는 것• 이미지의 질이 저하되는 원인으로는 시간이 흘러 마모된 이미지, 움직임이나 방해물에 의한 것, 획득 과정에서 불필요한 잡음이 삽입된 것, 캡처 장비의 불안정 등을 들 수 있음
이미지 분석 (Image Analysis)	<ul style="list-style-type: none">• 이미지 처리 과정에서 얻은 객체의 정보를 이용하여 디지털 이미지 데이터에 대해서 수치화된 데이터를 추출하는 과정 또는 추출된 데이터를 분류하는 것• 객체의 형태학적인 특성, 색상 특성 및 질감 특성을 추출하기 위해서는 여러 수식을 사용함• 얻어진 데이터를 통계학적으로 혹은 신경망 이론(Neuron Networks)이나 퍼지 이론(Fuzzy) 등을 사용하여 분류함
이미지 특성 검출 (Image Feature Extraction)	<ul style="list-style-type: none">• 이미지의 정확한 분할을 위해서 원하는 객체의 특성을 추출하는 것• 중요한 특징값과 중요하지 않은 특징값들을 판단하여 정확히 적용시켜야 정확한 특성을 검출할 수 있음• 이미지의 특징값은 이미지 전체 또는 이미지의 분할된 영역에 대해서 적용할 수 있음
이미지 분류 (Image Sorting)	<ul style="list-style-type: none">• 이미지 분류는 픽셀 단위의 분류와 영역이나 그룹의 분류로 나뉨• 픽셀 단위의 분류는 기준 화소가 인접한 픽셀의 명도, 색상 등과 같은 유사성을 고려하여 같은 그룹으로 포함함• 영역별 분류는 인접한 그룹과의 유사성 또는 그룹 간의 거리에 의해서 같은 그룹으로 포함함

4.4 이미지 처리 기법과 응용 분야

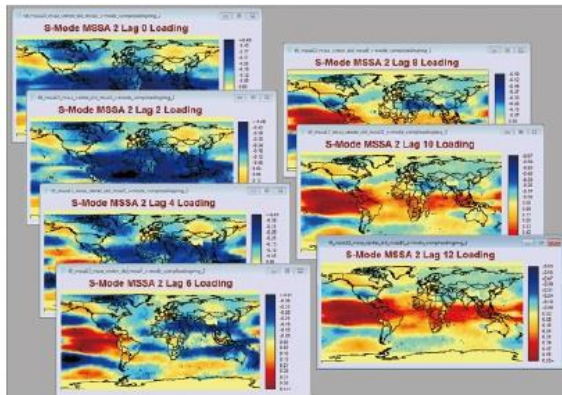
■ 이미지 처리 기법



(a) 이미지 분할



(b) 이미지 복원



(c) 이미지 분석



(d) 이미지 특성 검출

그림 5-33 다양한 이미지 처리 기법 [11]

5.1 이미지와 그래픽 편집 소프트웨어

■ 페인팅 소프트웨어

- 물감으로 그림을 그리는 것과 같은 작업을 할 수 있는 프로그램
- 픽셀 단위로 이미지를 처리하는 **비트맵 방식**을 기반으로 하여 이미지를 확대하면 선명도가 떨어짐
- 픽셀 개수의 조정으로 회화나 사진 같은 사실적인 이미지 표현이 가능함
- 어도비 포토샵(Adobe Photoshop), 코렐 페인터(Corel Painter)등이 대표적임

■ 드로잉 소프트웨어

- 대상의 윤곽을 선으로 작업하는 프로그램
- 페인팅 소프트웨어로 만든 이미지보다 용량을 적게 차지함
- **벡터 방식**이기 때문에 이미지의 왜곡 없이 확대 · 축소가 가능함
- 어도비 일러스트레이터(Adobe Illustrator), 코렐드로우(CorelDraw), 오토 스케치 등이 대표적

5.1 이미지와 그래픽 편집 소프트웨어

■ 이미지 프로세싱 소프트웨어

- 사진, 동영상 등과 같은 디지털 이미지의 편집 작업을 위한 프로그램
- 이미지의 확대, 축소, 회전 등과 같은 기하학적 변환과 색보정, 색변환, 매트 페인팅을 통한 디지털 영상 합성 등의 작업을 할 수 있음
- 포토샵, 포토 스타일러 등이 대표적임



▲ 드로잉 계열의 프로그램과 이미지 편집 프로그램

5.2 포토샵과 일러스트레이터

■ 포토샵

- 어도비시스템즈(Adobe Systems)사에서 개발한 이미지 편집 소프트웨어
- 비트맵 이미지의 편집에 사용됨
- 색상 보정, 필터 효과와 같은 이미지 변형, 오래된 사진 복원, 이미지 합성, 문자디자인, 웹디자인 등에서 많이 활용

■ 일러스트레이터

- 어도비시스템즈(Adobe Systems)사에서 개발한 소프트웨어
- 벡터 이미지의 생성과 편집에 사용됨
- 로고 디자인, 패키지 디자인, 인쇄 디자인 등에 활용됨



그림 5-34 포토샵과 일러스트레이터의 작업 화면