이미지 데이터

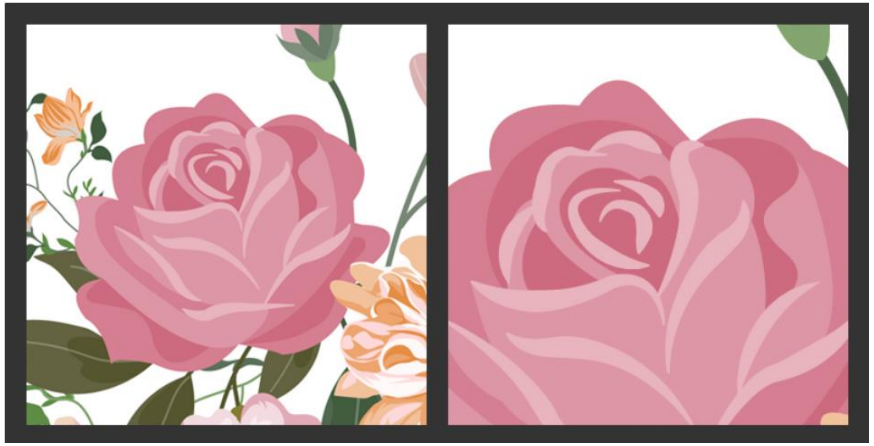
이미지 데이터의 표현방식

비트맵(Bitmap)

- 점(Pixel,화소)로 이미지를 표현
- 이미지를 확대하면 계단현상 생김
  -> 안티앨리어싱 처리함
- 파일형식: BMP, GIF, JPG, PNG 등

백터(Vector)

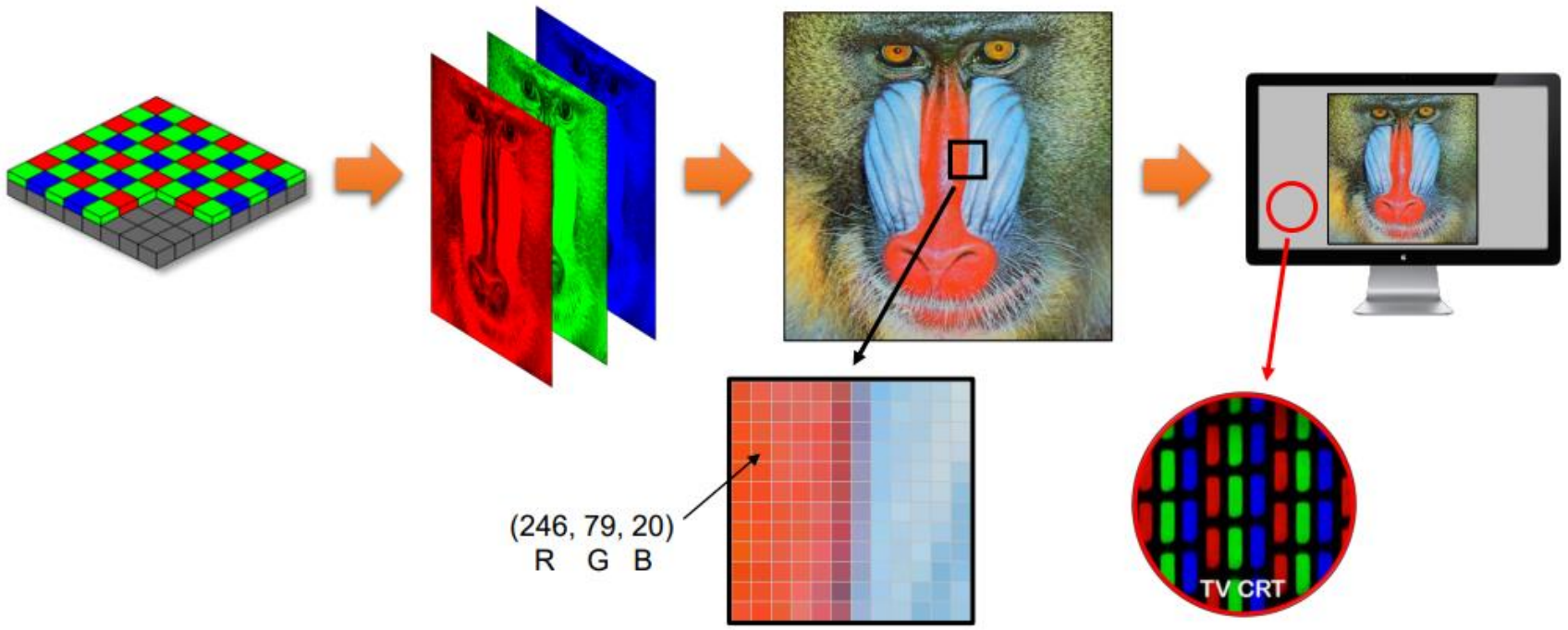
- 점과 점을 연결하는 직선, 곡선을 수학적 원리로 그려 표현
- 이미지 확대해도 계단 현상 없음
- 파일형식: AI, WMF, SGV, EPS 등



영상 표현 방법

영상(Image)

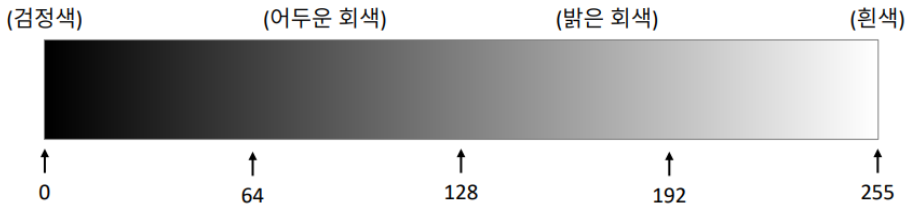
- 픽셀(pixel)이 바둑판 모양의 격자에 나열되어 있는 형태 (2차원 행렬)
- 픽셀: 영상의 기본 단위, picture element, 화소(畵素)



영상 표현 방법

그레이스케일(grayscale)영상

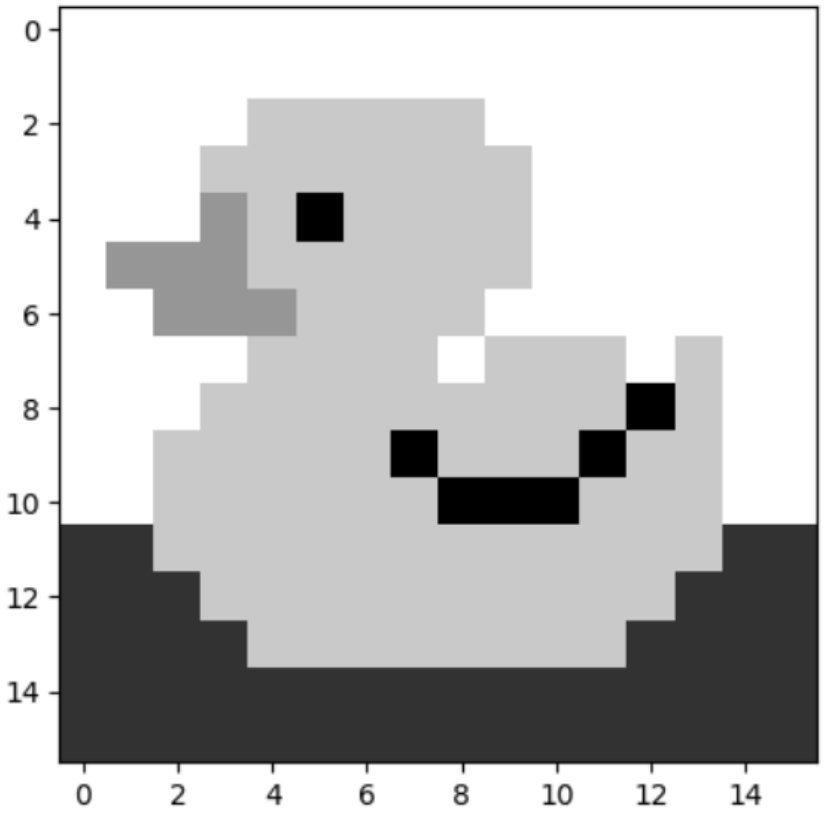
- 흑백 사진처럼 색상 정보가 없이 오직 밝기 정보만으로 구성된 영상
- 밝기 정보를 0~255까지 256 단계로 표현
  (0-> 검정, 255-> 흰색)



영상 표현 방법

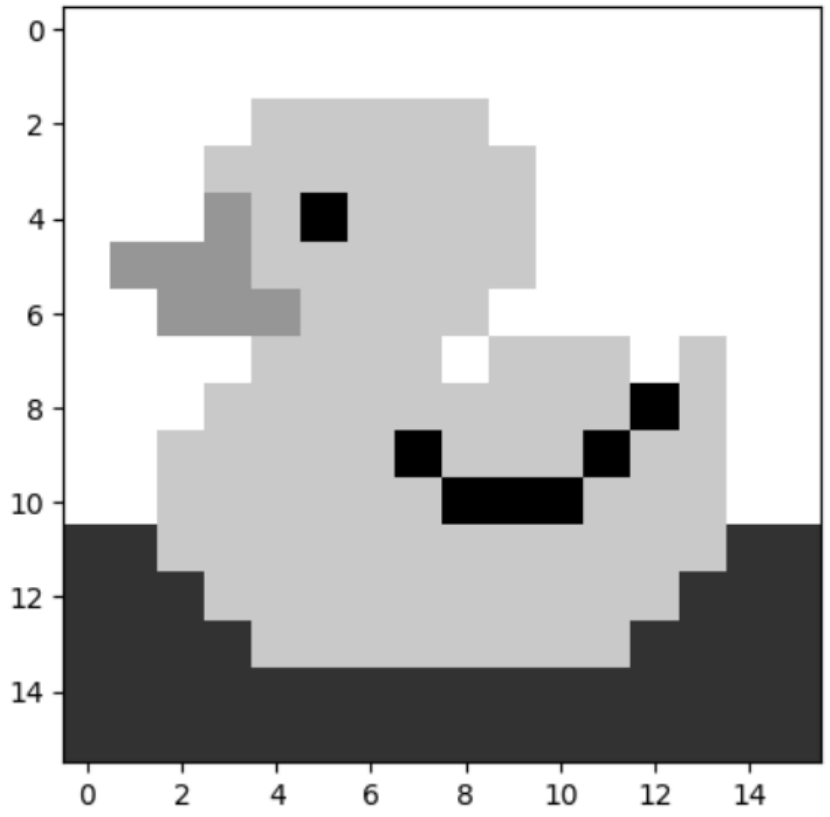
그레이스케일(grayscale)영상

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
1	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
2	255	255	255	255	201	201	201	201	201	255	255	255	255	255	255	255
3	255	255	255	201	201	201	201	201	201	201	255	255	255	255	255	255
4	255	255	255	150	201	0	201	201	201	201	255	255	255	255	255	255
5	255	150	150	150	201	201	201	201	201	201	255	255	255	255	255	255
6	255	255	150	150	150	201	201	201	201	255	255	255	255	255	255	255
7	255	255	255	255	201	201	201	201	255	201	201	201	255	201	255	255
8	255	255	255	201	201	201	201	201	201	201	201	201	0	201	255	255
9	255	255	201	201	201	201	201	0	201	201	201	0	201	201	255	255
10	255	255	201	201	201	201	201	201	0	0	0	201	201	201	255	255
11	50	50	201	201	201	201	201	201	201	201	201	201	201	201	50	50
12	50	50	50	201	201	201	201	201	201	201	201	201	201	50	50	50
13	50	50	50	50	201	201	201	201	201	201	201	201	50	50	50	50
14	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
15	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50



영상 표현 방법

그레이스케일(grayscale)영상



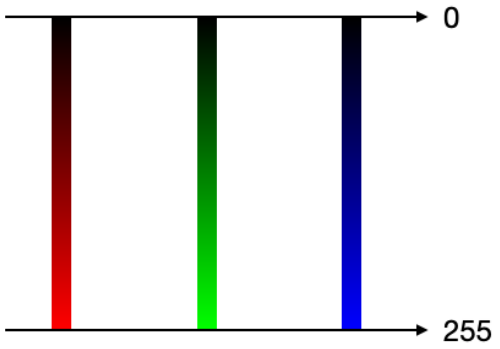

```
import numpy as np
import matplotlib.pyplot as plt

# 16x16 픽셀
duck = np.array([
    [255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255],
    [255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255],
    [255,255,255,255,201,201,201,201,201,255,255,255,255,255,255,255],
    [255,255,255,201,201,201,201,201,201,201,255,255,255,255,255,255],
    [255,255,255,150,201,  0,201,201,201,201,255,255,255,255,255,255],
    [255,150,150,150,201,201,201,201,201,201,255,255,255,255,255,255],
    [255,255,150,150,150,201,201,201,201,255,255,255,255,255,255,255],
    [255,255,255,255,201,201,201,201,255,201,201,201,255,201,255,255],
    [255,255,255,201,201,201,201,201,201,201,201,201,  0,201,255,255],
    [255,255,201,201,201,201,201,  0,201,201,201,  0,201,201,255,255],
    [255,255,201,201,201,201,201,201,  0,  0,  0,201,201,201,255,255],
    [ 50, 50,201,201,201,201,201,201,201,201,201,201,201,201, 50, 50],
    [ 50, 50, 50,201,201,201,201,201,201,201,201,201,201, 50, 50, 50],
    [ 50, 50, 50, 50,201,201,201,201,201,201,201,201, 50, 50, 50, 50],
    [ 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50],
    [ 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50],
], dtype=np.uint8)

print(duck.shape)
plt.imshow(duck, cmap='gray')
plt.show()
```

영상 표현 방법

트루컬러(truecolor) 영상

- 컬러 사진처럼 색상 정보를 가지고 있어서 다양한 색상을 표현할 수 있는 영상
- Red, Green, Blue 색 성분을 256 단계로 표현
  → 256^3 = 16,777,216 색상 표현 가능

흑백 이미지와 마찬가지로 각각의 색은 0~255까지의 숫자로 색의 농도를 표현할 수 있고,
RGB 즉 Red,Green,Blue 순서로 3가지 색의 조합으로 표현해야 한다



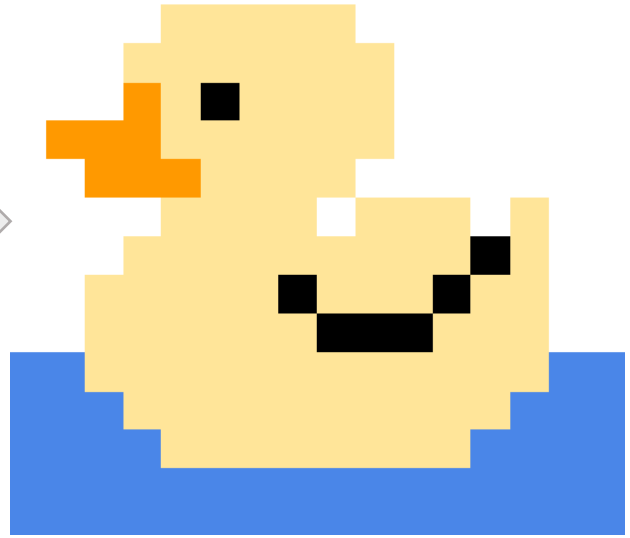

```
Red   : (255,0,0)
Green : (0,255,0)
Blue  : (0,0,255)
Black : (0,0,0)
White : (255,255,255)
```

- jpg: 3처널(RGB) ➔ (행,열,3)
- png : 4채널 짜리 (RGBA) ➔(행,열,4)
  - RGBA에서 A는 투명도: 0->투명, 255->불투명
  예: [255,255,25,0] , [255,255,25,100] , [255,255,25,255]

영상 표현 방법

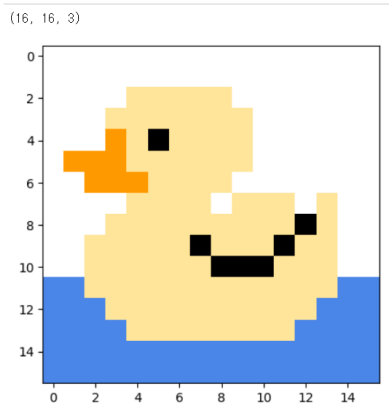
트루컬러(truecolor) 영상

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
1	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
2	[255,2							,229,153]	[255,229,153]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
3	[255,1	[255,255,255]		[255,153, 0]		[255,153, 0]		,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
4	[255,1							,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
5	[255,255,25					,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
6	[255,255,255]	[255,255,255]	[255,153, 0]	[255,153, 0]	[255,153, 0]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]
7	[255,255,255]	[255,255,255]	[255,255,255]	[255,255,255]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,229,153]	[255,255,255]	[255,255,255]
8	[255,255,255]	[255,255,255]	[255,255,255]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[0, 0, 0]	[255,229,153]	[255,255,255]	[255,255,255]
9	[255,255,255]	[255,255,255]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[0, 0, 0]	[255,229,153]	[255,229,153]	[255,229,153]	[0, 0, 0]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]
10	[255,255,255]	[255,255,255]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[0, 0, 0]	[0, 0, 0]	[0, 0, 0]	[255,229,153]	[255,229,153]	[255,229,153]	[255,255,255]	[255,255,255]]
11	[74,134,232]	[74,134,232]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[74,134,232]	[74,134,232]
12	[74,134,232]	[74,134,232]	[74,134,232]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[74,134,232]	[74,134,232]	[74,134,232]
13	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[255,229,153]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]
14	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]]
15	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]	[74,134,232]



영상 표현 방법

```
import numpy as np
import matplotlib.pyplot as plt

duck = np.array([
    [[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,153,  0],[255,229,153],[  0,  0,  0],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,153,  0],[255,153,  0],[255,153,  0],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,153,  0],[255,153,  0],[255,153,  0],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,229,153],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[  0,  0,  0],[255,229,153],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[  0,  0,  0],[255,229,153],[255,229,153],[255,229,153],[  0,  0,  0],[255,229,153],[255,229,153],[255,255,255],[255,255,255]],
    [[255,255,255],[255,255,255],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[  0,  0,  0],[  0,  0,  0],[  0,  0,  0],[255,229,153],[255,229,153],[255,229,153],[255,255,255],[255,255,255]],
    [[ 74,134,232],[ 74,134,232],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[ 74,134,232],[ 74,134,232]],
    [[ 74,134,232],[ 74,134,232],[ 74,134,232],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[ 74,134,232],[ 74,134,232],[ 74,134,232]],
    [[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[255,229,153],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232]],
    [[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232]],
    [[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232],[ 74,134,232]],
], dtype=np.uint8)

print(duck.shape)
plt.imshow(duck)
```



영상 표현 방법

1. 이미지 형식(gif, jpg,jfif, png, wmf 등) 조사

2. 색상표 색상표란(RGB,RGBA, HSV 등)