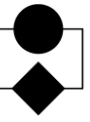




Pattern Recognition

Summary



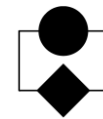


1. 기말고사

- 일시 : 12월 11일(월) 오후 4시
- 방법 : 대면 시험

2. 성적처리

- 평가 방법 : 상대평가
- 평가 기준 : 출석 (20%), 중간 (35%), 과제 (10%), 기말 (35%)



멀티미디어

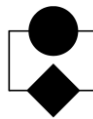


Multi (여러가지) + Media (매체)



여러가지 정보를 표현하고 전달(분배) 위한 다양한 형태의 매체

- 공감각적 매체 : 시각, 청각, 촉각 등
- 문자(Text), 그림(Image), 소리(Sound), 애니메이션, 비디오 등
- 종이, 사진, CD/DVD 등과 같은 정보를 전달하는 도구(장치)를 표현할 때도 Media라는 단어를 사용



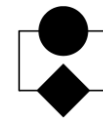
신호처리(Signal Processing)

- 디지털 또는 아날로그 신호를 처리하는 방법 혹은 알고리즘
- 각종 정보를 가공하여 어떠한 부가가치를 만들어 내는 작업



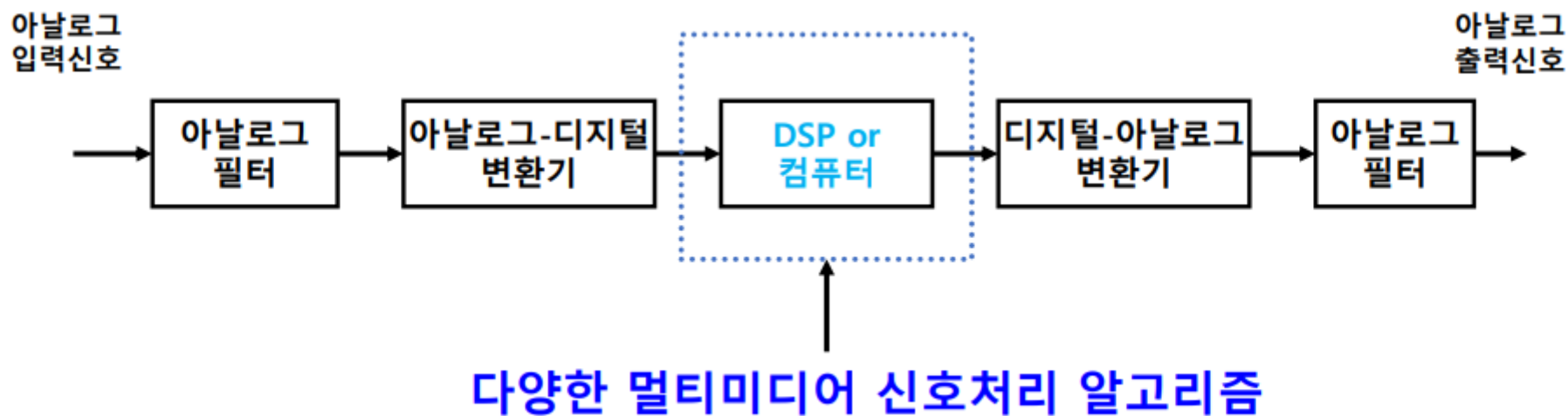
디지털 신호처리(Digital Signal Processing)

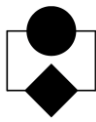
- 디지털 기술의 발전에 따라 아날로그 신호를 디지털 컴퓨터 또는 디지털 집적회로를 이용하여 수치적으로 처리하는 것
- **멀티미디어 신호처리**의 기반이 됨
- 샘플링 된 신호와 데이터들을 수치로 표현하여 처리하는 것
- 신호에서 원하는 정보를 추출, 전달, 저장하거나 혹은 시스템을 관측, 제어할 수 있도록 신호에 어떠한 가공을 디지털 형식으로 처리하는 것을 의미



멀티미디어 신호처리(Multi-media Signal Processing)

- 다양한 멀티미디어(음성, 오디오, 텍스트, 영상, 동영상 등)에 대한 디지털 신호처리를 의미

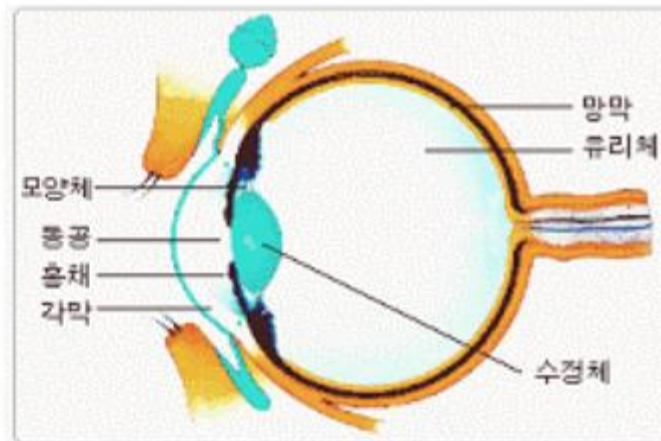




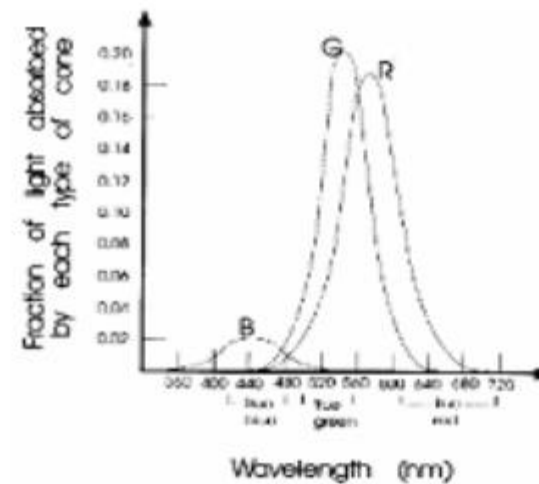
영상 이론

✓ 눈의 구조

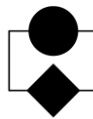
- 보는(Seeing) 동안 눈이 수행하는 2가지 역할
 - To recognize details of a scene(the spatial resolution of the scene)
 - To recognize changes in a scene(the temporal resolution of a scene)
- 카메라 렌즈에 해당하는 수정체를 통해 들어온 빛이 망막에 상을 맺음
 - 망막 : 원추형 모양의 원추체(cone) 시세포, 막대 모양의 간상체(rod) 시세포



눈의 구조



추상체(cone)의 빛의 파장별 색 반응도

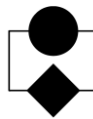


영상 이론



망막의 감광기관(Photoreceptor)의 2가지 형태

- 간상체(Rod)
 - 흑백 구분, 빛의 강도(intensity of light)에 민감
- 원추체(Cone)
 - 색을 구분
 - Red, Green, Blue 등의 색에 민감한 여러 종류의 Cone들이 있음
- 망막(Retina)에는 약 120×10^6 개의 간상체와 8×10^6 개의 원추체가 있음
 - 사람의 눈은 색(color)에는 둔감한 특징을 가짐
 - 특히, 색의 변화에는 더욱 둔감

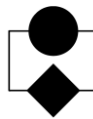


영상 이론



컬러모델(Color model)

- 어떤 특정 상황에서 컬러의 특징을 설명하기 위한 방법
- 하나의 컬러 모델을 사용하여 컬러의 모든 성질을 설명하기는 불가능
 - 일반적으로 컬러의 특성을 표현하기 위해 여러 종류의 컬러 모델을 정의하여 사용
- 서로 다른 영상처리 시스템은 서로 다른 색좌표계를 사용
- 색좌표계는 보통 세가지 요소를 사용하여 색을 표현
 - 각각의 요소를 하나의 축으로 하는 3차원 좌표 시스템에 대응
 - 컬러는 3차원 좌표 시스템 내에 존재하는 하나의 점으로 표시
 - RGB(Red, Green, Blue)모델, YCbCr 모델, HSI(Hue, Saturation, Intensity) 모델 등



디지털 영상신호처리



디지털 영상신호처리의 태동

- 1960년대 달표면을 찍은 위성사진의 화질 개선을 위해 디지털 컴퓨터를 사용하면서 시작



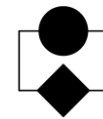
디지털 영상처리

- 실제 영상을 조작하는 것
- 전형적으로 2차원 데이터로 제한되지만, 일부 분야(의학)의 경우 3차원 데이터도 수행



컴퓨터 그래픽스

- 컴퓨터 프로그램에 의한 합성영상을 생성하는 것
- 2차원과 3차원 물체를 가지고 진행



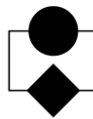
디지털 영상 표현



공간 해상도

[공간 해상도에 따른 전체 영상 화소수]

화소수	라인수	영상전체 화소수
20	15	300
40	30	1,200
80	60	4,800
160	120	19,200
320	240	76,800
640	480	307,200



영상의 파일 크기



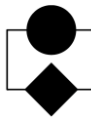
M X N (예 : 640 X 480)

- 흑백 영상, 각 픽셀당 8비트(256 밝기 값) : $640 \times 480 \times 8(\text{bits}) = 640 \times 480 \text{ bytes}$
- 컬러영상 : $640 \times 480 \times 8 \times 3\text{bits}$



영상 헤더(Header) 정보

- 영상 프로세서에게 영상 정보를 제공
- ID 비트들, 영상 해상도, 화소 당 비트 수, 컬러 맵 데이터, 저장 기술, 영상에 대한 설명 등
 - ID 비트 : 어떤 파일 형식이 사용되었고, 어떤 개정판을 사용했는지 등을 기호화



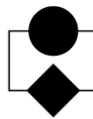
디지털 영상신호처리



사무자동화(O.A.)

현상이 생기지 않고 영상 신호를 변환 후, 주파수 영역 동시에 이용할 수 있다는 장점이 있다. 또한, 압축된 데이터의 분할이 가능하고 영상 부호화를 위한 최적의 웨이블릿 계수를 부호화 하는 가장 고전적이고 효율적인 EZW 영상 부호화 방식이다[5]. 이 방식은 웨이블릿의 간의 상관성에 대해 zerotree 기법을 도입하여 부호화 다양한 응용 분야에서 만족할만한 특성을 제공하고 있다. 따라 계층적 트리에 속하는 모든 계수값을 반복적으로 부호화 하는데 많은 연산을 수행하게 된다는 단점이 있다. 비트플레인 형태로 계수를 처리하는 SPIHT 방식도 2에서는 EZW 보다는 좋은 결과를 나타내지만 각각의 계수를 정렬하여 부호화 하기 때문에 연산 횟수가 많게 된다. 대상이 되는 단위가 하나의 웨이블릿 변환 계수이며 각각 부가 정보가 큰 편이다.

OCR에서의 자동 문자 분리의 예



디지털 영상 파일 형식



현재 가장 널리 사용되는 영상파일 형식은 **JPG 와 BMP 영상 파일**

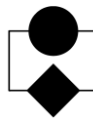


네트워크 브라우저에 의해 처음으로 승인된 영상 형태

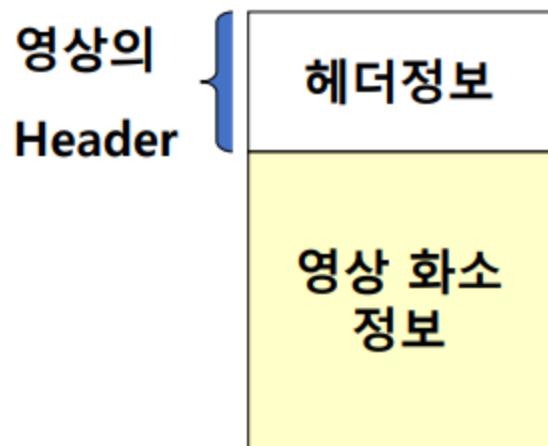
- WWW와 HTML 생성 언어와의 역사적 연관 때문에 8비트 **GIF 형태**는 가장 중요한 파일 형식 중의 하나

[Macromedia Director File Format]

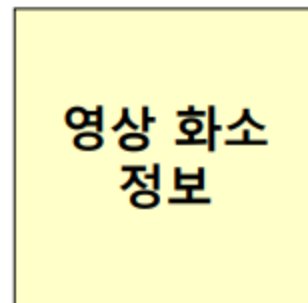
File Import					File Export		Native
Image	Palette	Sound	Video	Anim.	Image	Video	
.BMP, .DIB, .GIF, .JPG, .PICT, .PNG, .PNT, .PSD, .TGA, .TIFF, .WMF	.PAL .ACT	.AIFF .AU .MP3 .WAV	.AVI .MOV	.DIR .FLA .FLC .FLI .GIF .PPT	.BMP	.AVI .MOV	.DIR .DXR .EXE



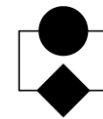
디지털 영상 파일 형식



BMP, JPG, GIF 등과 같은 헤더정보가 있는
영상 파일구조



RAW 영상 파일구조
(헤더정보 없음)



디지털 영상 파일 형식



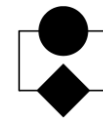
JPEG

- 영상압축에서 가장 기본이 되는 압축표준
- 정지영상압축 표준으로 널리 사용되며, 현재 JPEG2000으로 좀 더 보완됨
- 현재 가장 중요하고, 보편적인 파일 형태



BMP

- 영상 데이터를 압축하거나 다른 어떠한 기법도 사용하지 않고 데이터를 그대로 저장
- 단점 : 파일이 상당히 커짐
- 장점 : 파일구조가 간단, 데이터의 입출력 속도가 빠름

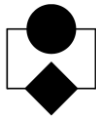


Color 영상 획득



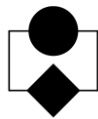
Color맵 또는 파레트

- 몇몇 그래픽스 디스플레이 시스템들은 화소 당 1 또는 2 바이트의 저장 공간만을 할당
- 화소 당 3 바이트를 요구하는 Color영상을 디스플레이 할 때 문제 발생 → color map으로 해결
- 16.7만 (224) Color → 256 (28) 또는 65,53 (216) Color로 줄이기 위한 참조 테이블임



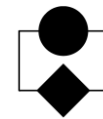
24Bits Color 영상과 Index Color 영상

- 640x480의 24bits Color 영상은 압축을 하지 않을 경우 921.6 Kbyte(=640x480x3)의 저장공간이 필요
- 24bits Color 영상이 각 화소마다 특수효과 정보를 나타내는 값을 저장하기 위한 여분의 bytes를 포함
 - 실제로는 32비트의 영상으로 저장
- 공간적인 문제로 인하여 많은 시스템은 화면 영상을 8비트의 Color 정보(소위 256 Color)만을 사용하도록 만들 수 있음
 - Color 정보를 저장하기 위한 Color 참조표 (Look Up Table) 사용



8Bits Index Color 영상의 Color LUT(Look Up Table)

- 8Bits Color 영상에서 사용된 개념은 각 화소에 대해 단지 인덱스나 코드값 만을 저장
- 값 25를 저장한 하나의 화소는 단지 칼라 참조표(LUT)에서 25번째 열을 의미
- 영상파일은 단지 각 RGB 값을 표현
 - 현재 저장된 8Bits 값이 각각의 인덱스가 된다는 것을 영상의 헤더 정보에 저장
- LUT를 팔레트(**palette**)라는 용어로 사용



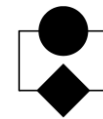
Index Color Image

인덱스	R	G	B
0	5	4	5
1	10	14	26
2	25	34	23
3	43	33	43
.	.		
.	.		
.	.		
254	178	120	48
255	233	120	35

256 컬러(8비트)의 팔레트 예

120	25	254	210	221	. . .
55	2	89	25	210	. . .
253	35	98	35	98	. . .
.
.
.
.

256 컬러(8비트)의 팔레트 인덱스를 이용한 영상저장 예



BMP

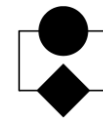
파일헤더(BITMAPFILEHEADER)
영상헤더(BITMAPINFOHEADER)
팔레트 정보(RGBQUAD)

BMP 파일의 헤더



영상데이터

(영상데이터가 거꾸로 저장되어 있음)

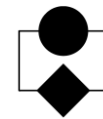


영상 포인트 처리(Image Point Processing)



사칙연산

사칙연산	수식	비고
덧셈	$Output[m][n] = Input[m][n] + k1$	Output[m][n]: 출력영상 Input[m][n]: 원본영상 k1 ~ k4: 임의의 상수 값
뺄셈	$Output[m][n] = Input[m][n] - k2$	
곱셈	$Output[m][n] = Input[m][n] * k3$	
나눗셈	$Output[m][n] = Input[m][n] / k4$	



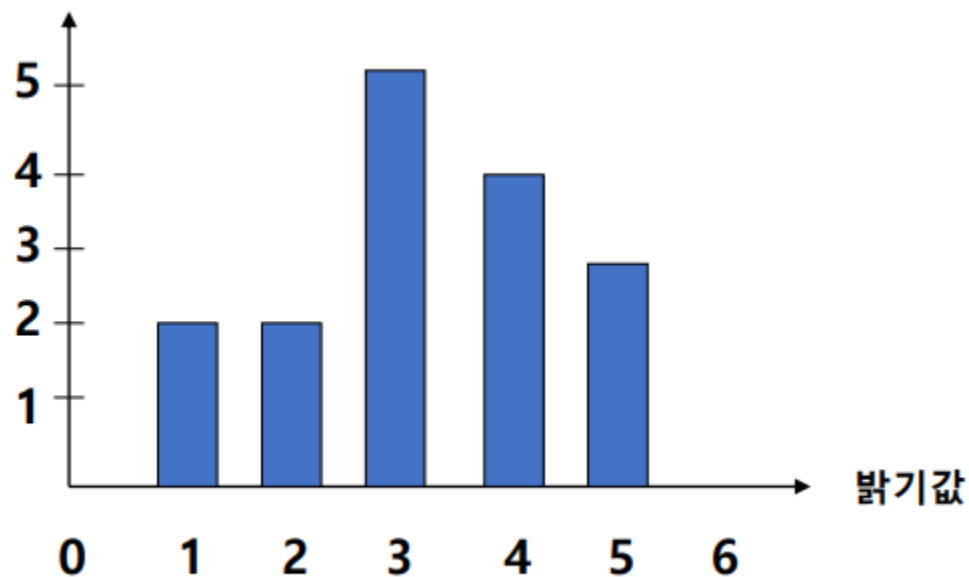
영상 히스토그램(Image Histogram)

- ✓ 영상의 명암값 프로파일을 확인하기 위해 사용
- ✓ 영상의 구성(명암 대비, 명암 값 분포)에 대한 정보를 제공

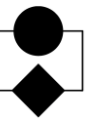
5	4	3	3
5	4	4	5
2	1	3	3
3	1	4	2

4x4 영상

빈도수(frequency)



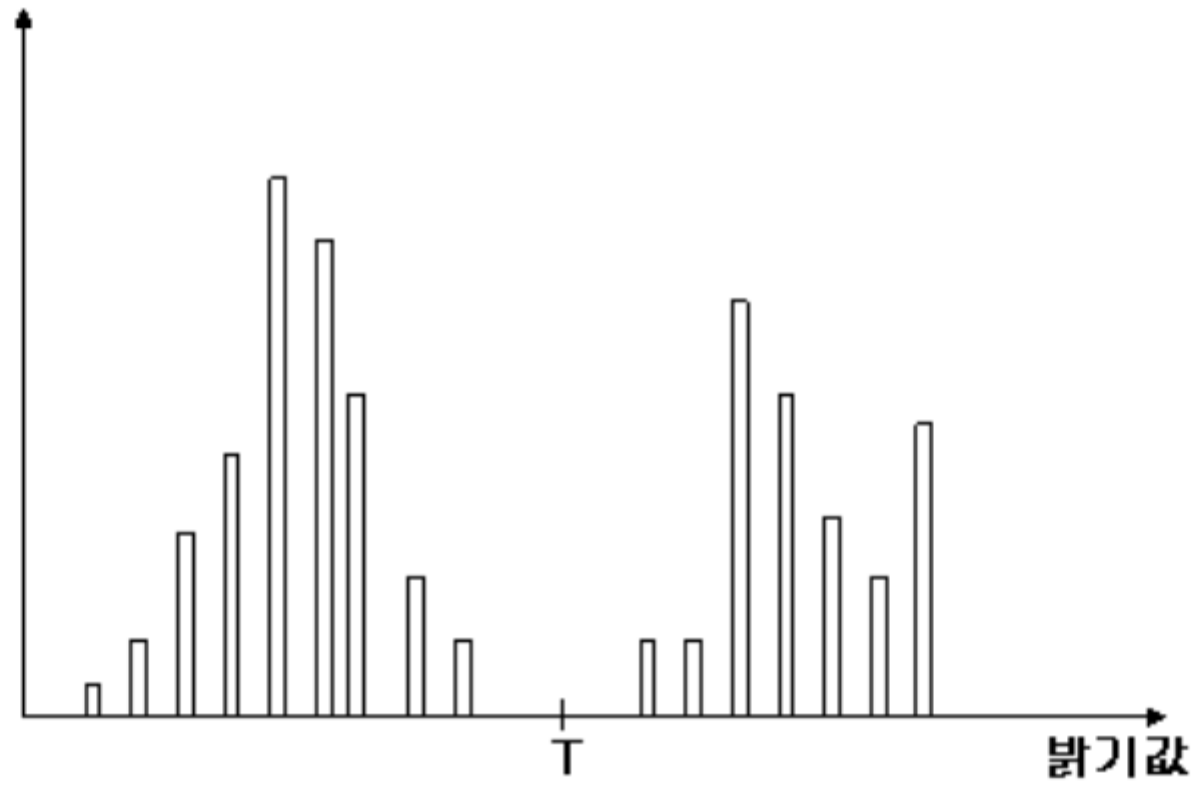
16화소(4X4) 영상에 대한 영상 히스토그램



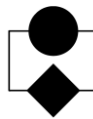
임계값(Threshold) 결정



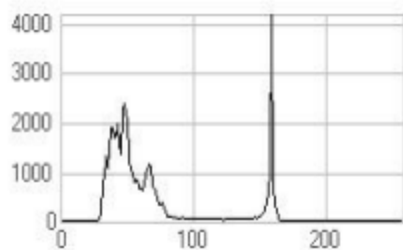
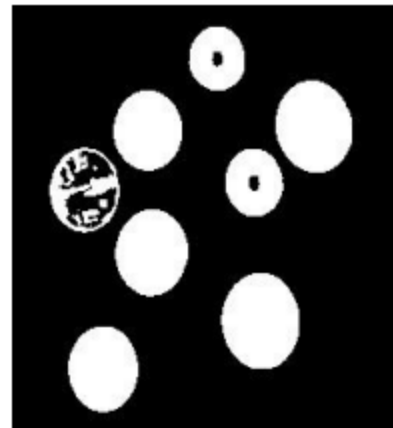
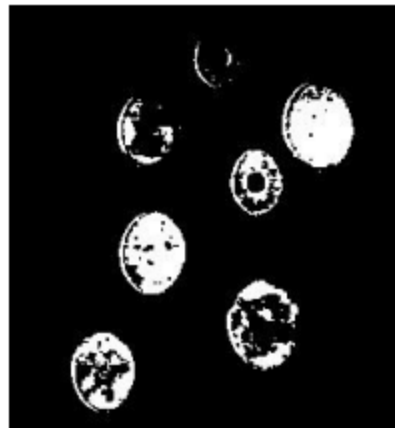
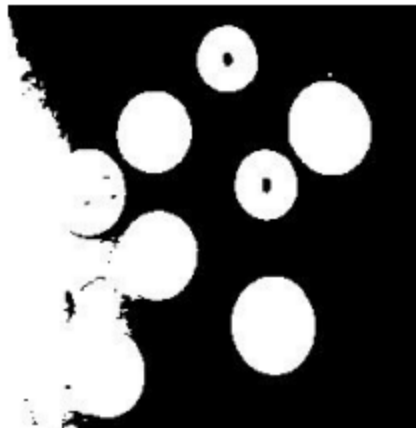
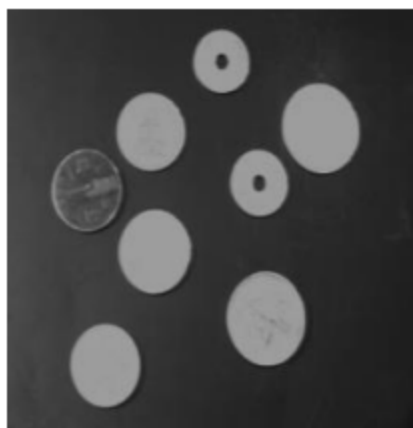
두 모드(Bimodal) 히스토그램의 시각적 분석에 의한 임계값 결정



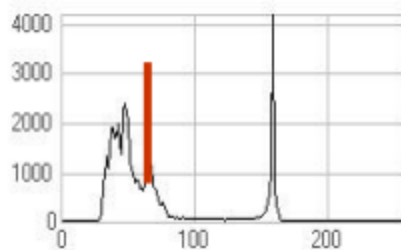
시각적 분석에 의한 임계값 결정



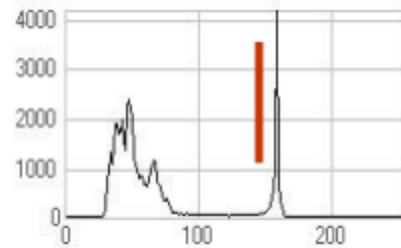
히스토그램 기반의 이진화(Binarization)



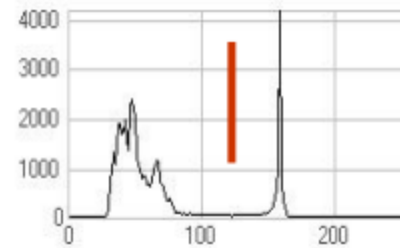
(a) 원영상



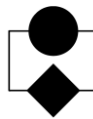
(b) T가 작을 때



(c) T가 너무 클 때



(d) 적당한 T



임계값 결정 방법(Gonzalez, Woods)

1. 임계값 T 의 초기값을 추정한다.

(제안된 추정 값은 영상에서 밝기의 최소값과 최대값 사이의 중간점 또는 평균점)

2. 임계값 T 를 기준으로 영상을 분할하고, 밝기값이 T 보다 작은 화소들로 구성된 영역($G1$)과 밝기값이 T 보다 큰 화소들로 구성된 영역($G2$)의 화소들로 나눈다.

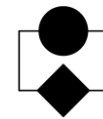
(T 는 $G1$, $G2$ 어디에 두어도 상관없음)

3. 영역 $G1$ 과 $G2$ 에 대하여 화소들의 밝기의 평균값(μ_1, μ_2)을 계산한다.

4. 연속적으로 다음과 같은 새로운 임계값을 계산한다.

$$T = \frac{\mu_1 + \mu_2}{2}$$

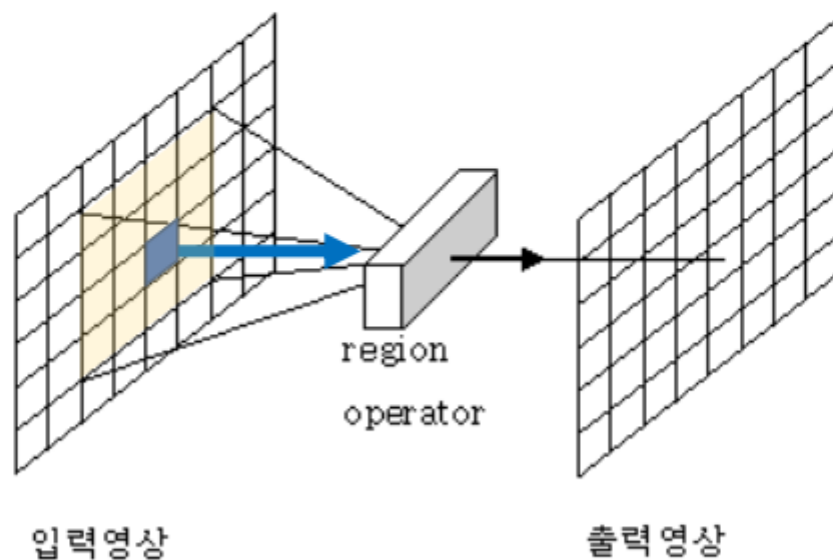
5. 연속적인 반복에서 경계 값의 변화가 미리 정의된 오차 \mathcal{E} 보다 작을 때 까지 단계 2에서 4까지 반복한다.



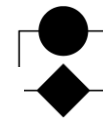
영상 영역 처리(Image Area Processing)



영상신호 영역 처리는 출력 화소를 만들기 위해서 입력 화소 뿐만 아니라, 입력 화소 주위에 있는 화소들을 사용하여 컨볼루션(Convolution)을 통하여 이루어짐



영상 영역 처리 개념



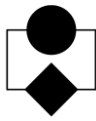
영상 영역 처리(Image Area Processing)



컨볼루션 (Convolution)

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad 3 \times 3 \text{ 컨볼루션 마스크}$$

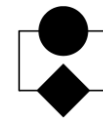
- 입력 화소의 근방에서 화소의 가중 합으로 표현
- 행렬로 표현되는 가중치는 컨볼루션 마스크(convolution mask), 컨볼루션 커널(convolution kernel)로 부름
- 가중 행렬의 크기는 항상 그 중심을 결정할 수 있도록 기수(odd)로 사용
- 중심점의 위치는 출력 화소의 위치에 대응
- 이미지 스무딩(smoothing), 샤프닝(sharpening), 에지(Edge) 검출 등과 같은 영상처리에 효과적으로 사용



영상 영역 처리(Image Area Processing)

✓ 컨볼루션 (Convolution)

$$\begin{aligned} I'5 = & M_1 * I_1 + M_2 * I_2 + M_3 * I_3 \\ & + M_4 * I_4 + M_5 * I_5 + M_6 * I_6 \\ & + M_7 * I_7 + M_8 * I_8 + M_9 * I_9 \end{aligned}$$

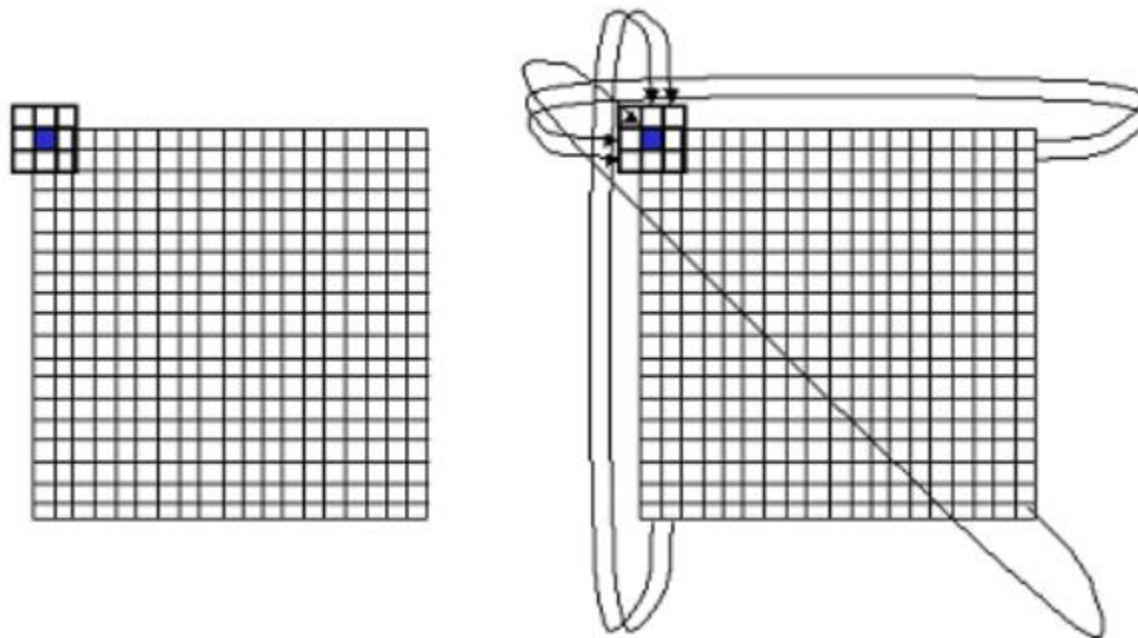


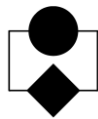
Convolution



영상 영역 밖 부분에 대한 처리

- 컨볼루션 창이 중심이 (0,0)이 될 때
- 해결책
 - Zero-padding: 컨볼루션 창의 비어있는 셀(cell)은 0으로 처리
 - Warp-around convolution: 영상의 모서리가 서로 연결되어 있는 것처럼 처리



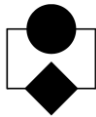


Convolution



저역통과 필터

- 공간 저역통과 필터는 영상의 저주파 성분을 그대로 통과시키고, 고주파 성분은 감쇄되는 효과
- 저역통과 컨볼루션 마스크는 현재 입력화소의 밝기값 주변의 8개의 이웃화소 밝기값과 자기자신의 밝기값을 평균하는 역할
- 고주파 성분이 많은 영상의 경계선영역의 경우 이러한 저역통과 필터 처리로 인하여 급격한 밝기 변화가 완만한 변화로 감쇄되는 현상이 나타남
→ 이러한 시각적 효과를 **영상 흐림화(blurring)**

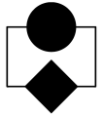


Convolution



고역통과 필터

- 고역통과 필터는 저역통과 필터의 반대 효과를 낸다. 저주파 성분은 감쇄시키는 반면에 고주파 성분은 통과시키는 특징
- 영상에서 에지와 같은 고주파 성분이 있는 부분을 더욱 강조하는 샤프닝(sharpening) 효과
- 원본 영상을 고역통과 필터링 한 경우 영상의 spot성 노이즈(고주파 성분)가 매우 부각되어 나타남
- 고역통과 필터링은 흐릿한 영상(blurring)을 샤프닝 처리하는데도 많이 사용

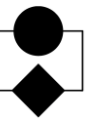


Convolution



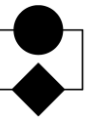
경계선(Edge) 검출

- 영상의 경계선은 사물의 윤곽, 위치, 크기 등을 나타내고, 영상의 밝기와 색이 급격한 변화가 있는 경우 경계선(에지)이 존재



임펄스 잡음(Impulse Noise)





Average Filter를 이용한 잡음 제거

- Average Filter를 이용한 잡음 제거 결과



가우시안 잡음 제거 결과



임펄스 잡음 제거 결과

Median Filter를 이용한 잡음 제거

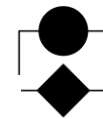
- Median Filter를 이용한 잡음 제거 결과



가우시안 잡음 제거 결과



임펄스 잡음 제거 결과

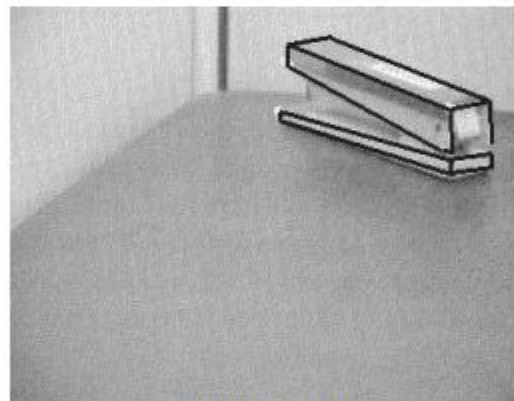


영상 분할(Image Segmentation)

- 영상 분할(segmentation)
 - 영상분할은 배경(background)에서 객체(object)를 추출하는 것을 의미
 - 음성인식에서 특별한 소리만을 인식하는 것도 분할에 해당
 - 영상에서 관심 있는 영역(semantic region)은 객체가 되고 나머지는 배경으로 취급

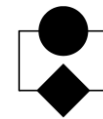


영역분할

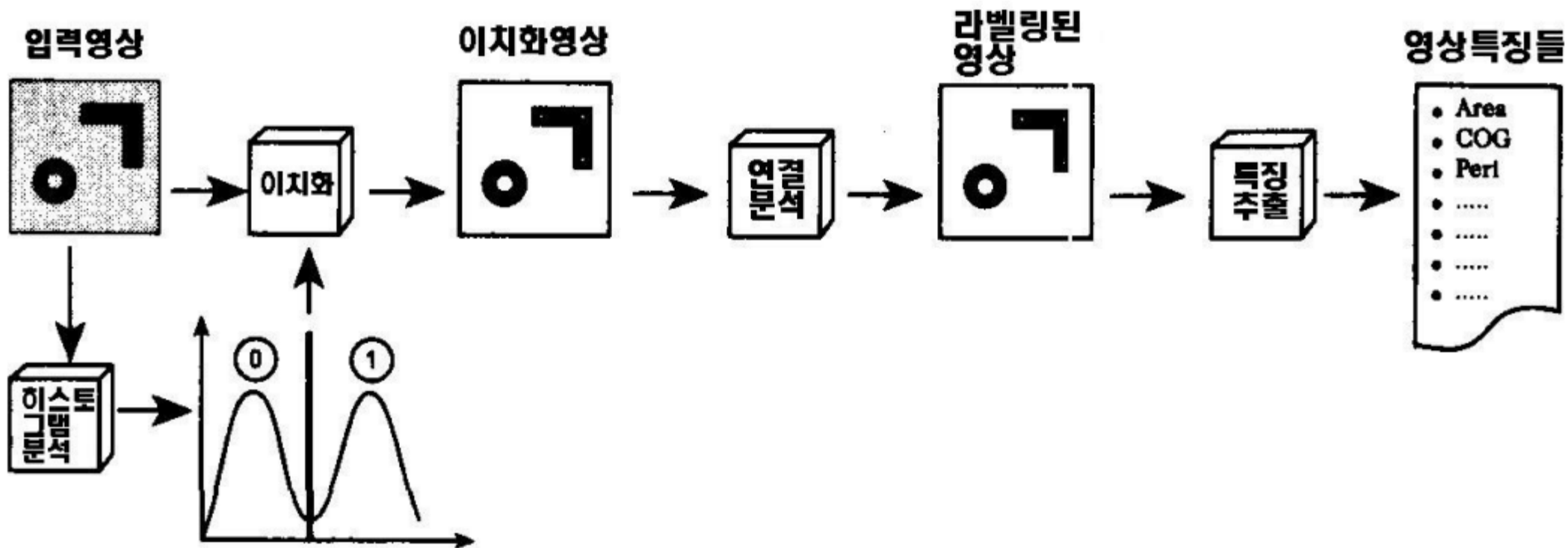


객체인식

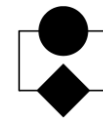
ROI : Region Of Interest



이진 영상을 이용한 객체 인식



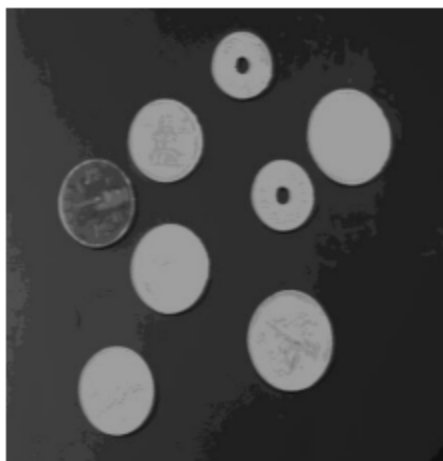
영상 이진화를 이용한 객체 인식 단계



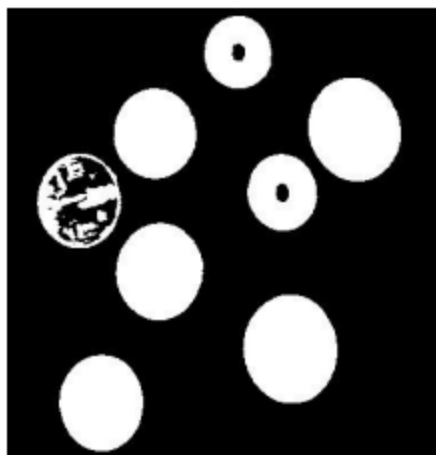
이진 영상을 이용한 객체 인식

- Labeling

- "인접하여 연결"되어 있는 모든 화소에 동일한 번호(Label)을 붙이고, 또 다른 연결 성분에는 다른 번호를 붙이는 작업



(a) 원 입력영상



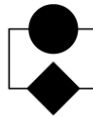
(b) 이진화된 영상



(c) Labeling된 영상



(d) 하나의 Label 영역



이진 영상을 이용한 객체 인식

- Grassfire : 이진화된 영상을 Labeling 하는 가장 간단한 알고리즘
- 자기호출을 가진 재귀호출(recursive call)을 이용
- 과도한 자기호출은 시스템 스택(system stack)을 넘치게(overflow)하기 때문에, 너무 큰 크기의 객체 영역을 Labeling하기에는 적당하지 않음
- 자기호출은 프로그램이 내부적으로 처리하는 많은 명령들로 인해 속도가 저하되는 단점이 있음

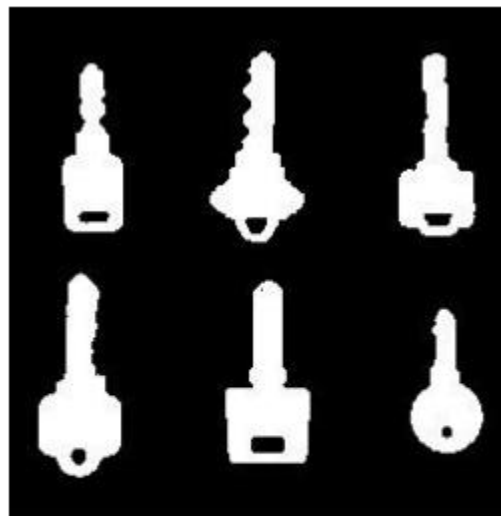
영역 경계 검출

- 영역의 경계추적

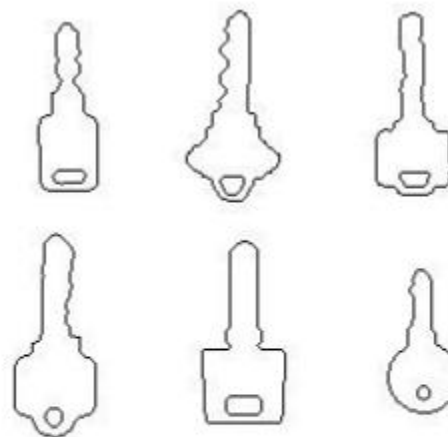
- 이진화된 영상 또는 Labeling된 영상에서 일정한 밝기 값을 가지는 영역의 경계를 추적하여 경계픽셀의 순서화된 정보를 얻어내는 것



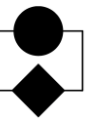
(a) 원 입력영상



(b) 이진화된 영상



(c) 영역경계 추적결과



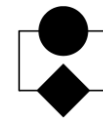
영역 경계 검출

			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			

(a) 이진화된 영상

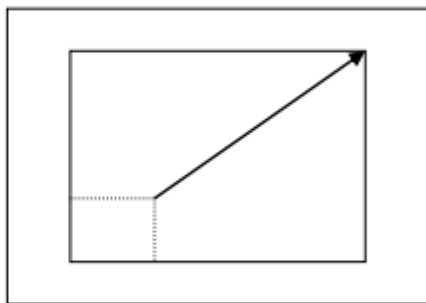
			X	0				
		15	255	255	1			
	14	255	255	255	255	2	3	
	13	255	255	255	255	4		
	12	255	255	255	5			
		11	255	255	6			
		10	9	8	7			

(b) 추적된 경계위치

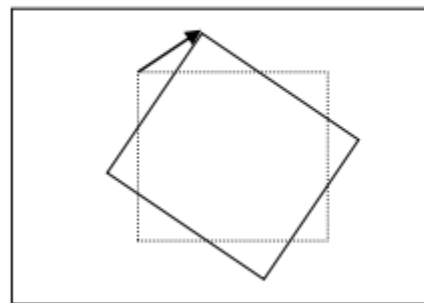


Geometric Transform

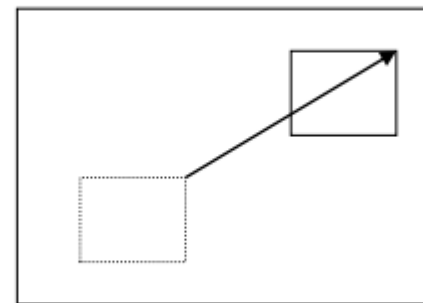
- 기하학적 처리
 - 임의의 기하학적 변환에 의해 화소들의 배치를 변경하는 처리
 - 영상 내에서 화소들을 움직이거나 화소들을 생성(보간법)



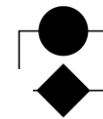
Scaling



Rotation



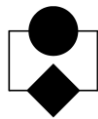
Translation



사상(Mapping)

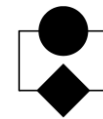
순방향 사상
(Forward Mapping)

역방향 사상
(Reverse Mapping)



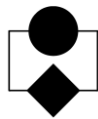
보간법(Interpolation)

- 보간법은 없는 화소 값을 주변 화소값을 사용하여 얻는 방법으로
주변의 화소들을 분석하여 새로운 화소를 생성
 - 인접 화소 보간법(Nearest neighbor interpolation)
 - 양선형 보간법(Bilinear interpolation)
 - 고차 보간법(Higher order interpolation)



Nearest neighbor interpolation

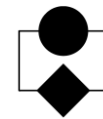
- 개요
 - 생성된 주소에 가장 가까운 원시 화소를 **출력 화소로 할당**
 - 원시 화소에 대해 계산된 분수 주소는 가장 가까운 유효한 화소 주소로 반올림
- 장점
 - 처리 속도가 빠르다
- 단점
 - 계단 현상(Aliasing)이 확연하게 드러남
 - 입력 화소에 대응하는 출력 화소들의 수가 클수록 영상의 질이 떨어짐
 - 결과 영상이 상황에 따라 바뀔 수 있음



Bilinear Interpolation

- 개요

- 새로운 화소를 생성하기 위해 네 개의 **가장 가까운 화소들**에 가중치를 곱한 값들의 합을 사용
- 각각의 가중치는 화소로부터의 거리에 반비례
- 가장 인접한 화소 보간법보다 더 스무딩한 영상을 산출
- 상당히 많은 계산량 요구 : 화소당 3개의 일차 보간법



Scaling

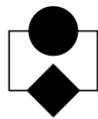
- 영상을 확대, 축소하는 기법

- 확대

- 확대(magnification), 스케일링 업(scaling up), 줌(zooming), 업샘플링(upsampling), 스트레칭(stretching)

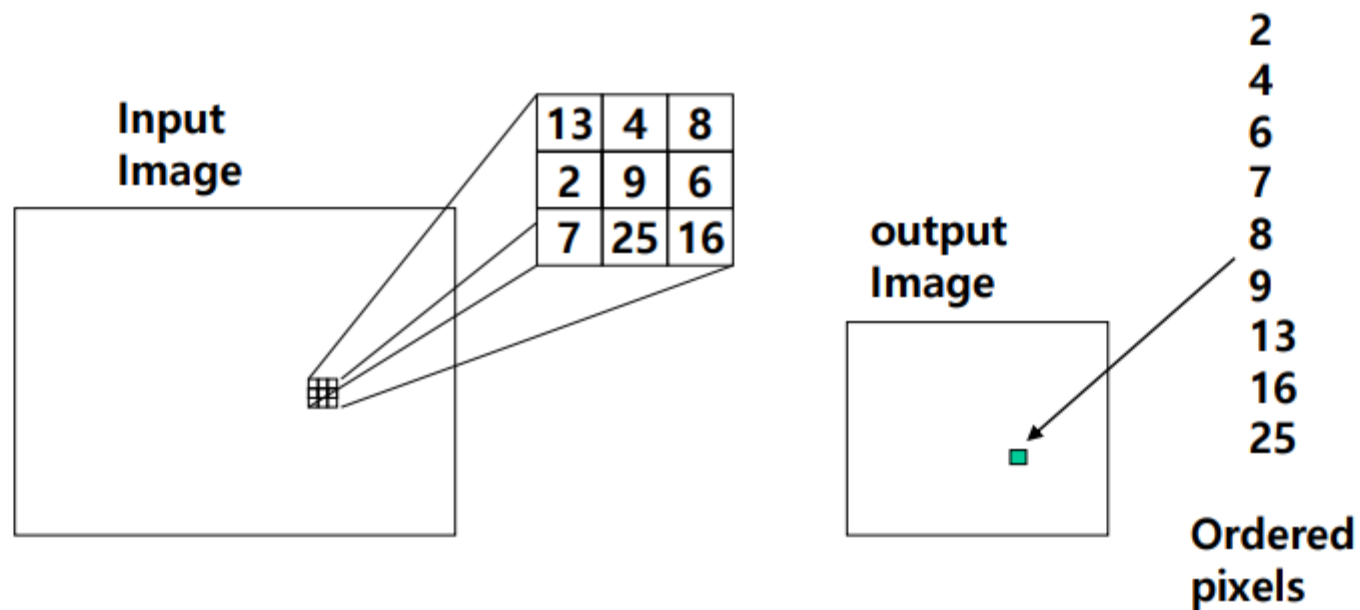
- 축소

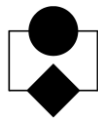
- 축소(minification), 스케일링 다운(scaling down), 데시메이션(decimation), 다운샘플링(down sampling), 쉬링킹(shrinking)



중간값 표현

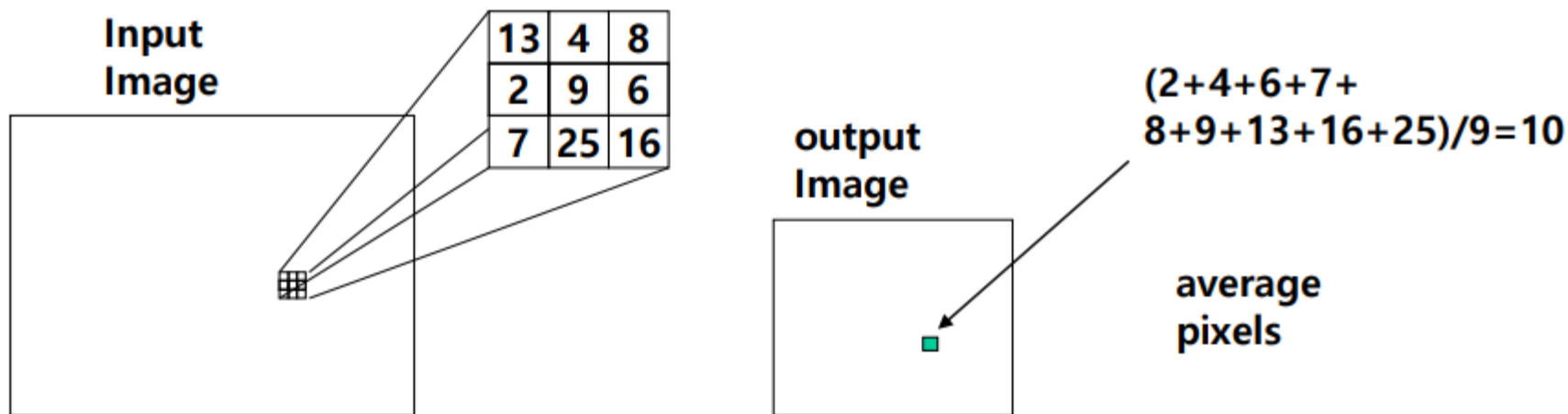
- 화소들의 블록을 그 중간값으로 표현
 - 블록내 화소들을 오름차순으로 정렬, 중간값 선택
 - 블록의 화소들을 정렬하는데 많은 계산 비용 소요





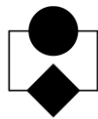
평균값(Average) 표현

- 화소들의 블록을 평균값으로 표현

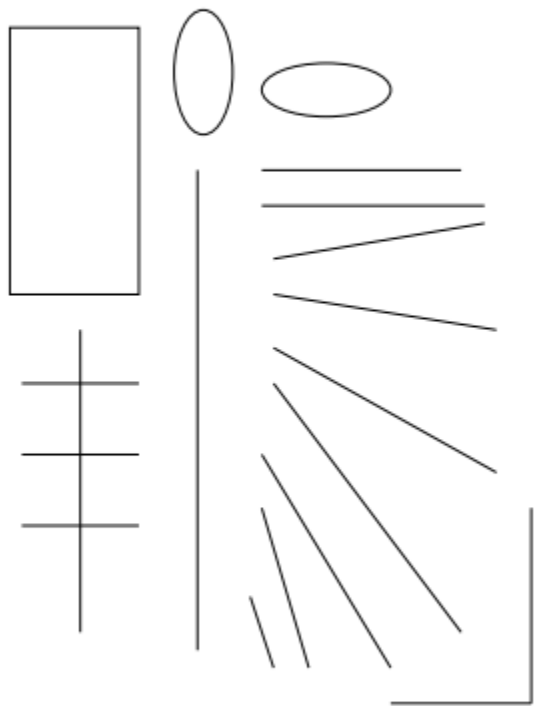


- 미디언 표현 vs 평균 표현

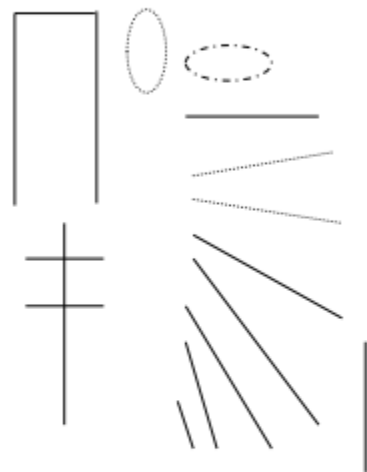
- 미디언 표현: 영상 선명, 처리속도 늦음(정렬)
- 평균 표현 : 스무딩 효과, 처리속도 빠름



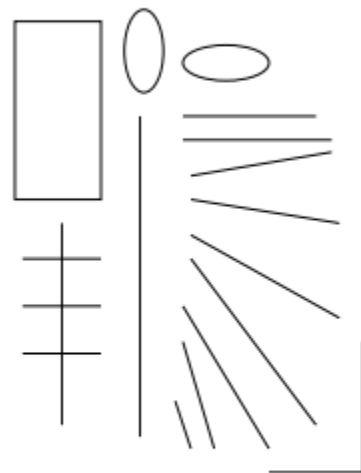
Scaling



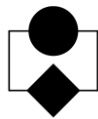
(a) 원래영상



(b) 2에 의한 서브샘플링에
의해 축소된 영상

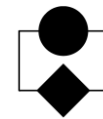


(c) 흐리게 만든 후 2만큼
축소된 영상



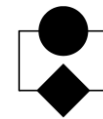
Morphology(형태학)

- ✓ 영상의 형태(Shape)를 분석하고 처리하는 기법
- ✓ 영상의 경계, 블록, 골격 등 형태를 표현하거나 서술하는 데 필요한 영상 요소를 추출하는데 형태학 처리를 활용
- ✓ 영상의 경계 너비가 일정치 않거나 중간에 단절되어 이를 일정하게 하거나 연결할 때 형태학 처리가 필요



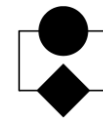
침식(Erosion)

- ✓ 물체의 크기를 그 배경과 관련하여 일정하게 줄여주는 것
- ✓ 물체의 크기는 줄어들고, 배경은 확대됨.
- ✓ 영상의 물체와 배경 사이에 스파이크 잡음이 있을 때 이 잡음을 제거하거나 전체 영상에서 아주 작은 물체를 제거하는 데 응용
- ✓ 영상에서의 돌출부는 감소시키고, 내부 돌출부는 증가시켜서 서로 닿은 물체를 분리할 때도 유용



팽창(Dilation)

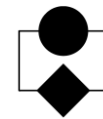
- ✓ 물체 내부의 돌출부는 감소하고 외부의 돌출부는 증가시켜서 물체의 크기를 확장하고 배경은 축소하는 기법
- ✓ 물체 내부에 발생한 구멍과 같은 공간을 채우거나 짧게 끊어진 영역을 연결하는 데
- ✓ 이진 영상에서 팽창연산은 입력 화소가 균일한 곳에서는 변화가 없으나 흑백 화소가 같이 있는 영역에서 동작



팽창(Dilation)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0



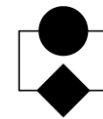
팽창(Dilation)

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

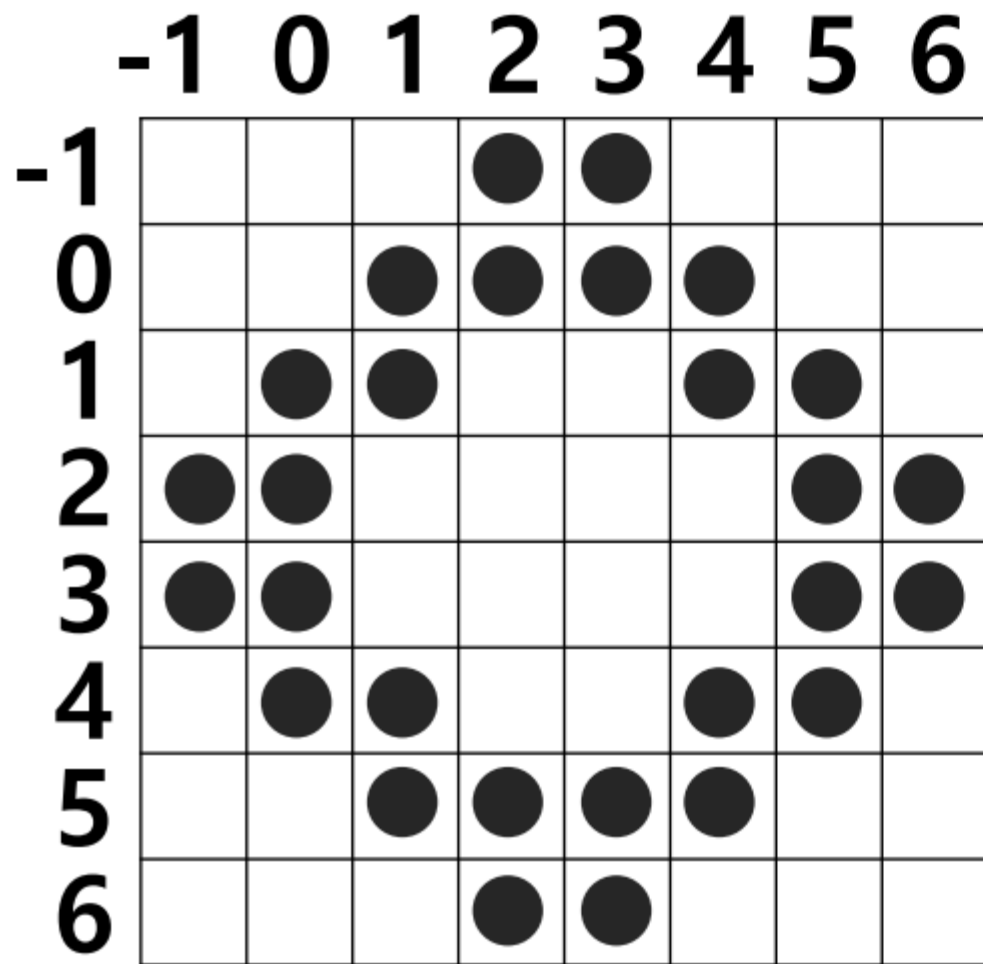


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

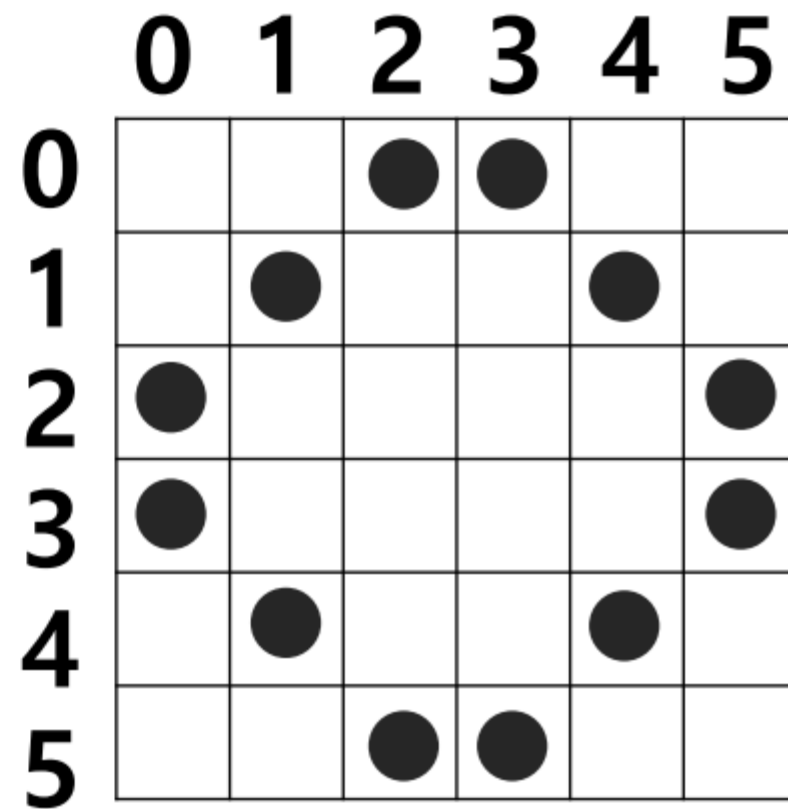




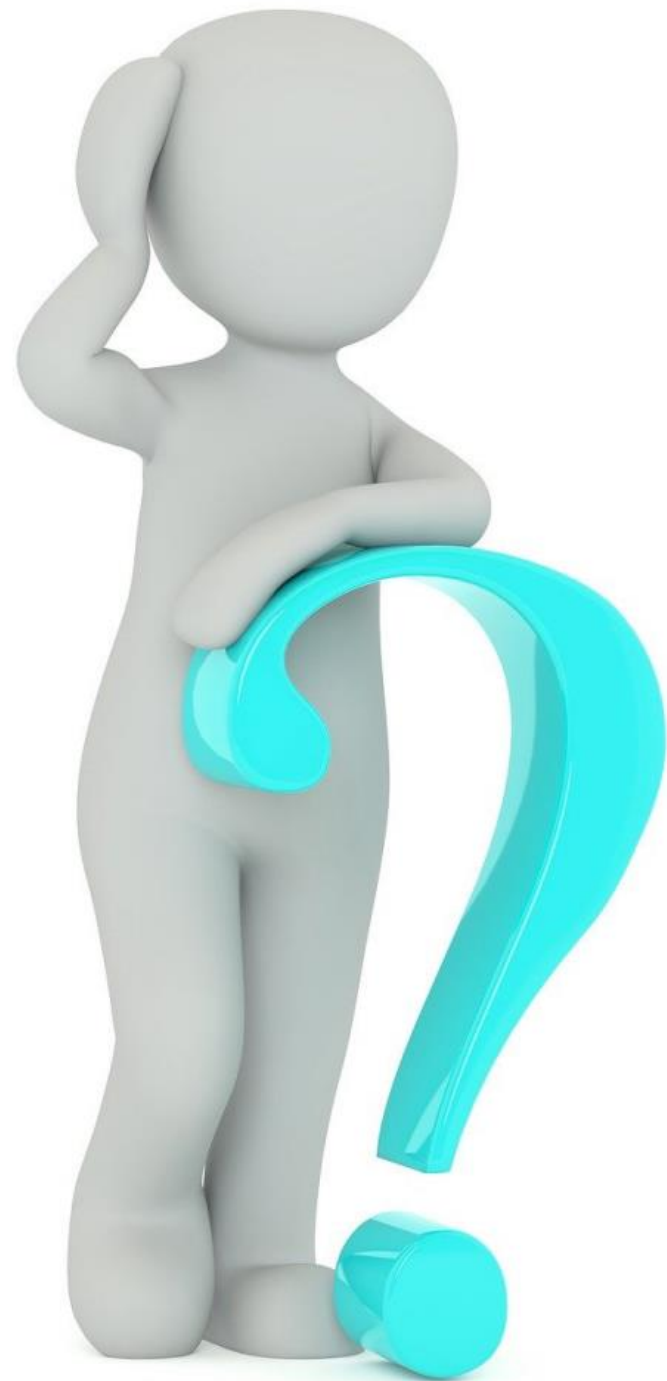
경계 검출



(f) $(A \oplus B) - (A \ominus B)$



(g) $A - (A \ominus B)$





Thank You

