

[과제3]

Hough Transform

2024-05-01 (수)

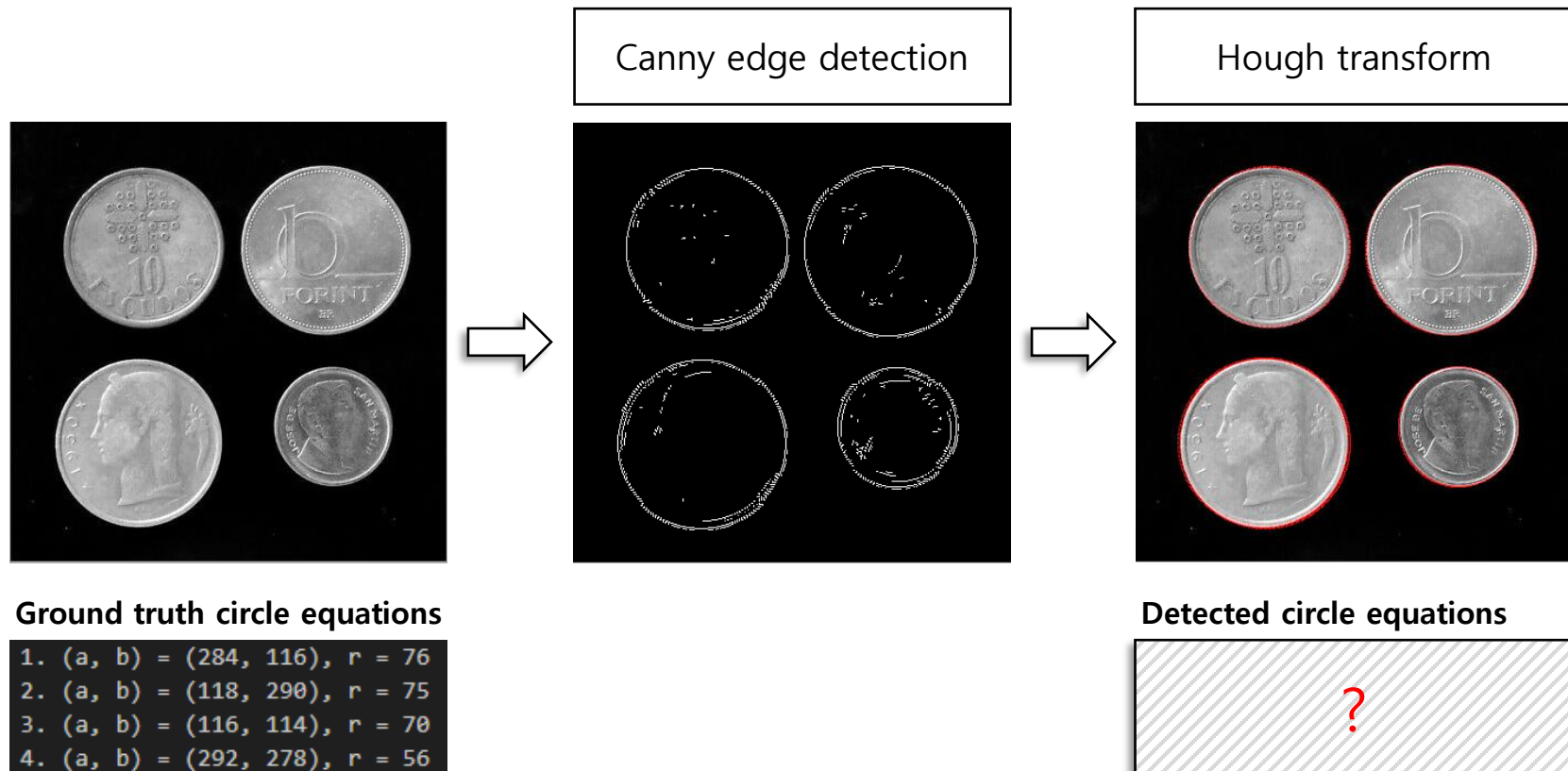
@ 2024-1 컴퓨터비전

Contents

- 과제 개요
- 캐니 에지 검출 알고리즘 (Canny edge detection algorithm)
- 허프 변환 (Hough transform)
- 과제 세부 설명

과제 개요

- 주어진 영상에서 다수 개의 '원' 검출
 - 캐니 에지 검출 알고리즘 구현하여 에지 영상 추출
 - 에지 영상에 대해 허프 변환을 이용하여 '원' 검출



캐니 에지 검출 알고리즘

CANNY EDGE DETECTION ALGORITHM

캐니 에지 검출 알고리즘

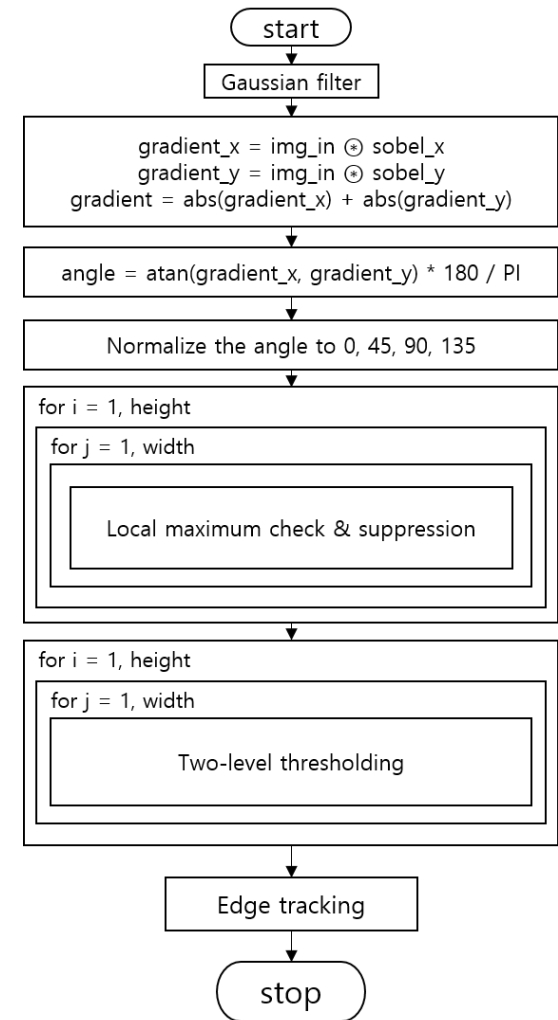
- 캐니 에지 검출 알고리즘[*] (Canny edge detection algorithm)
 - 존 캐니가 1986년에 고안한 알고리즘으로, 영상 내에서 경계선을 검출하는 알고리즘
 - 경계선 검출의 세 가지 목적을 만족하도록 설계
 - 1. 낮은 에러율
 - 경계선 검출 시 에러율을 낮추는 것
 - 검출된 경계선이 실제 경계선일 확률 최대화
 - 2. 경계선의 이동 최소화
 - 검출된 경계선을 실제와 비교했을 때 검출된 경계선의 이동 최소화
 - 3. 하나의 경계선만 검출
 - 한 개의 경계점이 여러 개로 검출되는 것 최소화

[*] J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-8, pp. 679-698, 1986

캐니 에지 검출 알고리즘

■ 캐니 경계선 검출 알고리즘 과정

- ① 잡음 제거를 위한 가우시안 필터 적용
- ② 밝기값의 기울기, 크기, 방향 계산
 - Sobel 필터 등을 사용하여 영상의 가로축과 세로축 미분 결과를 구해 기울기의 크기와 방향 계산
- ③ Non-maximum suppression 단계
 - 얇은 경계선으로 검출하는 과정
- ④ Two-level threshold 단계
 - 두 임계값을 사용하여 강한 경계선과 약한 경계선 구분
- ⑤ 경계선 추적을 통한 잘못된 경계선 제거
 - 최종 경계선 판단

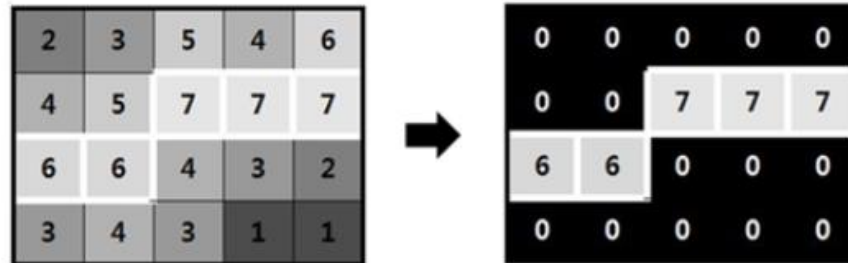


캐니 경계선 검출 알고리즘 순서도

캐니 에지 검출 알고리즘

■ Non-maximum suppression 단계

- 가우시안 필터링으로 인해 경계선이 두껍게 검출되는 문제 해결 목적
- 현재 픽셀의 방향성에 따라 0도 또는 45도 또는 90도 또는 135도 방향에 위치한 인접 픽셀과 비교하여, 현재 픽셀이 비교 범위에서 최댓값인 경우만 경계 후보로 유지



■ Two-level threshold 단계

- 높은 임계값과 낮은 임계값을 사용하여 경계선 구분 과정

$$pixel > High_{Threshold}$$

: 강한 경계선

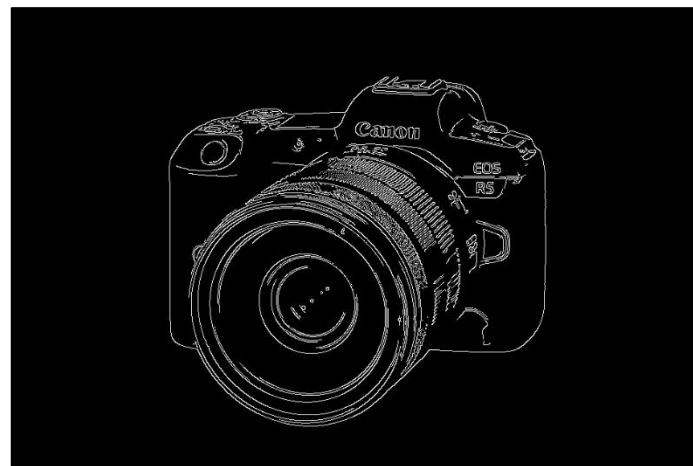
$$low_{Threshold} < pixel < High_{Threshold}$$

: 약한 경계선

$$low_{Threshold} > pixel$$

: 경계선 아님

캐니 에지 검출 알고리즘

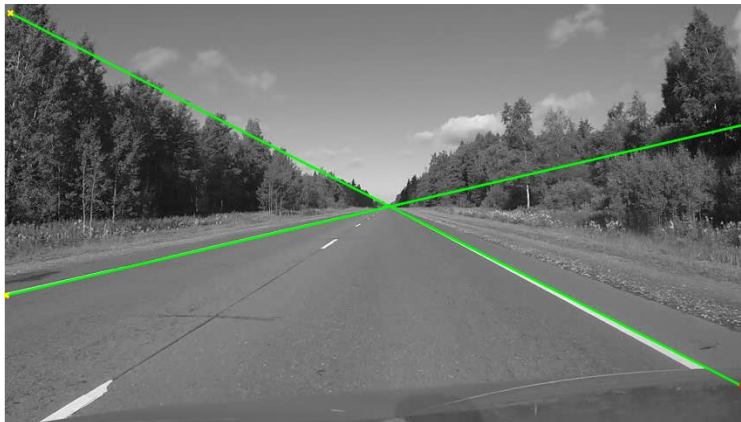


허프 변환

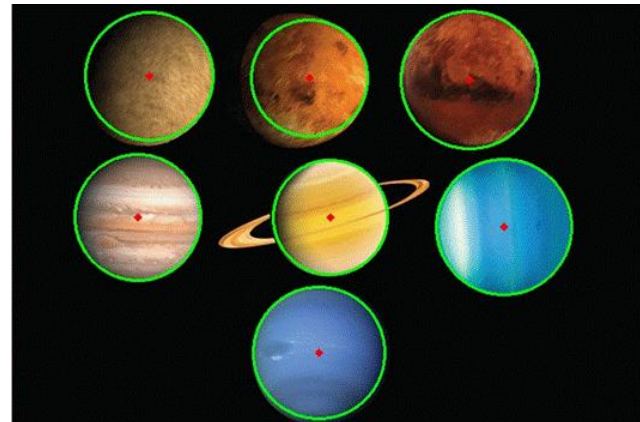
HOUGH TRANSFORM

허프 변환

- 허프 변환[*] (Hough transform)
 - 영상 내 특징을 추출하는 알고리즘
 - 직선, 원, ...
 - 영상 내의 특징들에 대해 모델을 가정하고, 모델의 파라미터를 파라미터 공간 (허프 공간)에서 투표를 통해 찾는 알고리즘
 - 영상 내 노이즈가 존재해도 높은 검출 성능을 보임



허프 변환 직선 검출 예시



허프 변환 원 검출 예시

[*] P. V. C. Hough, "Method and means for recognizing complex patterns," U.S. Patent 3 069 654, Dec. 18, 1962.

허프 변환

■ 직선 검출

– 직선의 방정식 (공간 좌표계)

- $y = a \times x + b$

→ 파라미터: (a, b)

• 공간 좌표계 직선의 방정식의 문제점

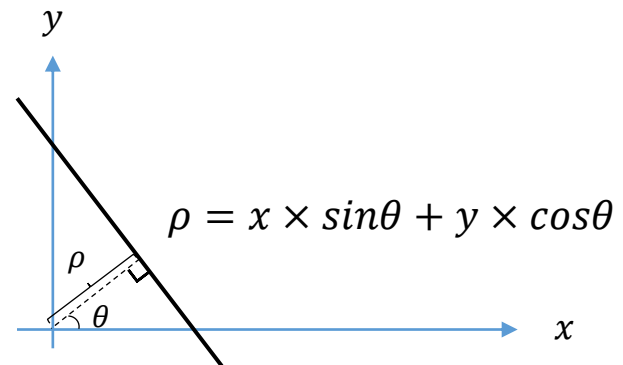
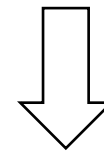
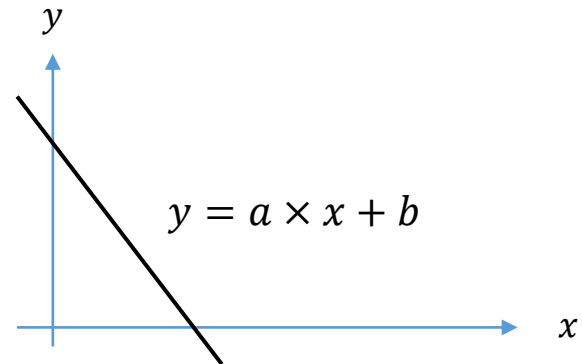
➢ 파라미터의 공간이 무한대

➢ a 의 범위: $[-\infty, +\infty]$

– 직선의 방정식 (극좌표계)

- $\rho = x \times \sin\theta + y \times \cos\theta$

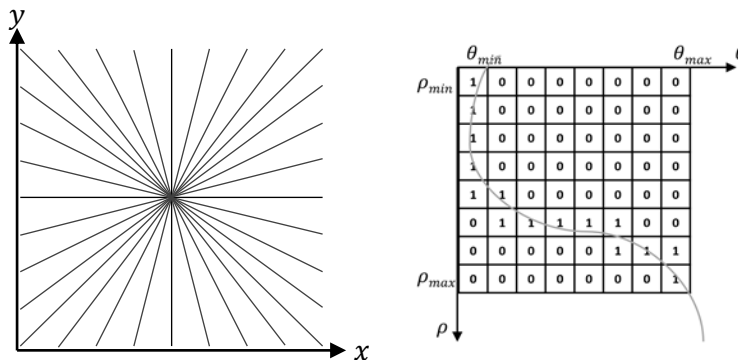
→ 파라미터: (ρ, θ)



허프 변환

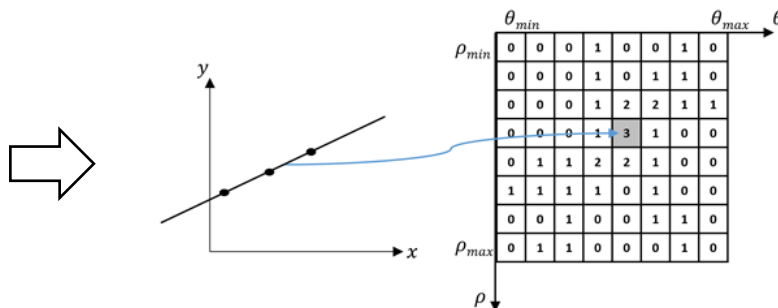
■ 직선 검출

- 영상의 한 특징점을 거리와 각도 공간으로 변환
 - 등 간격 샘플링
- 투표 (Voting)를 통해 직선의 파라미터를 찾고, 이를 통해 직선 검출



원본 영상

에지 영상



직선 검출 결과

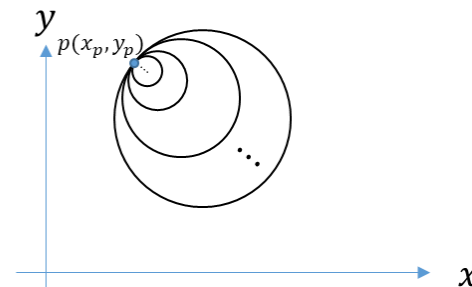
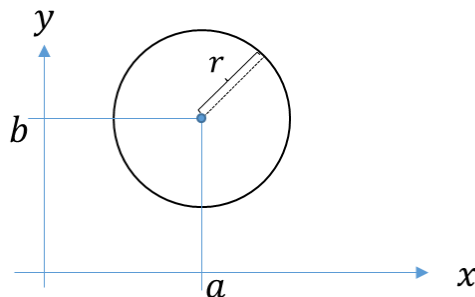
허프 변환

■ 원 검출

- 원 방정식: $(x - a)^2 + (y - b)^2 = r^2 \rightarrow$ 파라미터: (a, b, r)
 - (a, b) : 원 중심, r : 반지름
- 투표를 통해 영상 내 원의 파라미터를 찾기 위해 (x, y) 평면에서 (a, b, r) 공간으로 변환

$$\begin{array}{l} x - a = r \times \cos\theta \\ y - b = r \times \sin\theta \end{array} \Rightarrow \begin{array}{l} a = x - r \times \cos\theta \\ b = y - r \times \sin\theta \end{array}$$

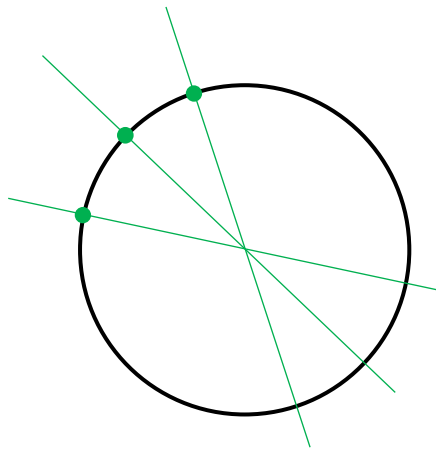
- (x, y) 평면 상의 각 후보 픽셀들 (특징점)에 대해 (a, b, r) 공간에서 투표 후, (x, y) 평면으로 변환하여 영상 내 원 검출
 - 3차원의 accumulator를 사용하기 때문에 시간 및 메모리 부담이 큼



허프 변환

■ 원 검출 : 그래디언트 (gradient) 기반 원 검출

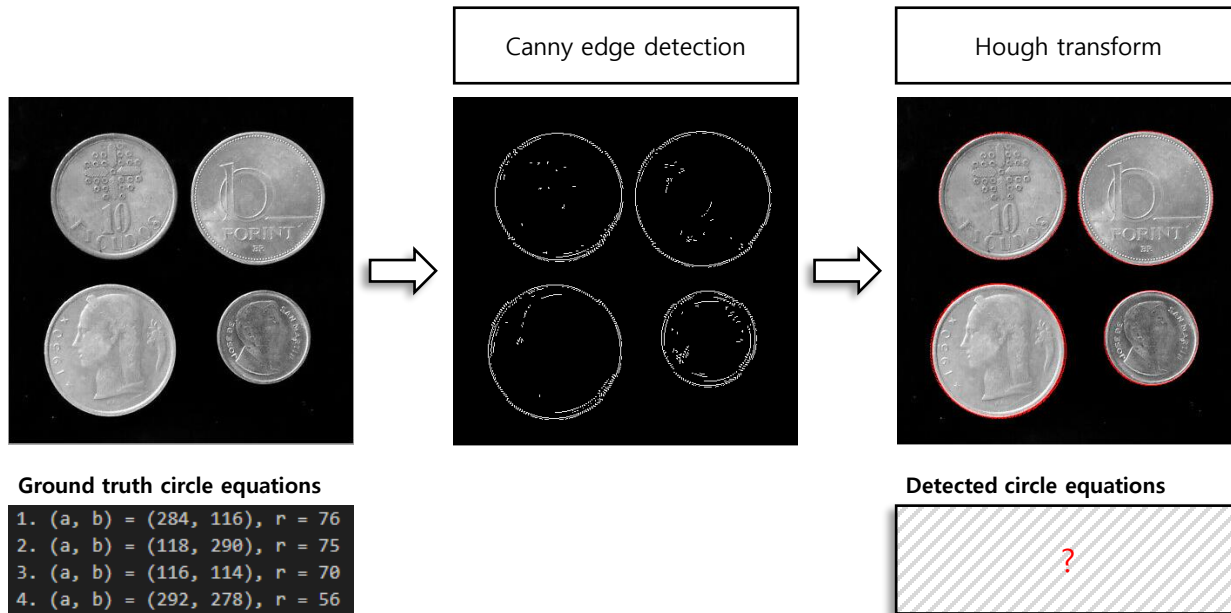
- 원의 방정식으로부터 구할 수 있는 (a, b, r) 3차원 파라미터 공간에서 투표하여 원을 검출하는 방식 → 시간 및 메모리 부담이 큼
- 이를 해결하기 위해,
그래디언트 기반 방법을 통해 두 단계로 원 검출 수행
 - 1. 원의 중심 좌표 검출
 - 에지 영상 추출 간 계산한 그래디언트를 이용하여 투표 방식으로 중심 좌표 검출
 - 2. 검출된 원의 중심으로부터 투표 방식을 통해 반지름 검출



과제 세부 설명

과제 세부 설명

- 주어진 영상에서 다수 개의 '원' 검출
 - 캐니 에지 검출 알고리즘 구현하여 에지 영상 추출
 - 에지 영상에 대해 허프 변환을 이용하여 '원' 검출
 - (a, b, r) 투표 방법
 - 그래디언트 기반 방법
 - 주어진 정답 (영상 내 각 원의 중심 & 반지름)에 가까운 결과를 도출하도록 구현



과제 세부 설명

■ 보고서 관련 사항

- 각 원에 대해 주어진 정답 (영상 내 각 원의 중심 & 반지름)에 가까운 원 검출 결과를 정리하여 작성

		1번 원	2번 원	3번 원	4번 원
Ground truth	원의 중심	(116, 114)	(284, 116)	(118, 290)	(292, 278)
	반지름	70	76	75	56
Hough transform results	원의 중심				
	반지름				

- 원 검출 결과, 연산량, 메모리 사용량, 수행 시간 등의 관점에서 두 허프 변환 원 검출 방법을 비교
 - (a, b, r) 투표 방법
 - 그래디언트 기반 방법

과제 세부 설명

- 테스트 파일
 - 입력 이미지
 - coins_396x400.raw (8-bit grayscale RAW 파일)
 - 정답 원 방정식 텍스트 파일
 - coins_groundTruth.txt
- 구현 환경
 - Visual Studio 2022 버전에서 C++ 언어로 구현
 - OpenCV 함수를 사용하지 않고 캐니 에지 검출 알고리즘과 허프 변환 직접 구현

과제 제출

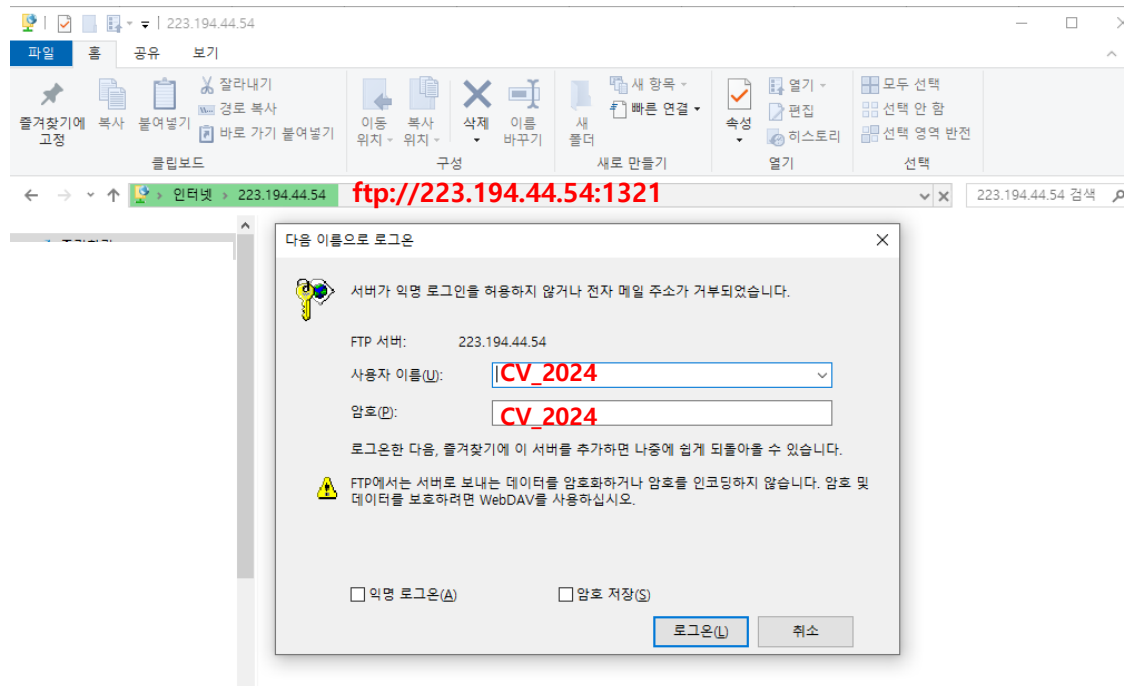
- 제출물 코드, 보고서 압축한 파일 제출 (학번_이름_ver#.zip) (예: 2024123456_홍길동_ver2.zip)
 - 코드
 - 과제 수행한 visual studio project (Debug 폴더 제외)
 - 코드에 주석 작성
 - 보고서
 - 과제 개요, 과제 수행 방법, 결과 분석, 고찰
- 제출처
 - FTP Sever
- 마감일
 - **2022년 5월 28일 (화요일) 23:59:59** (서버 시간 기준)
 - 마감일 이후 ~ 일주일: 채점 점수의 50%만 실제 과제 점수로 반영
 - 일주일 이후: 0점 처리

과제 제출 방법

■ FTP Server

– Windows 파일 탐색기 또는 FileZilla 이용해 서버 접속

- URL: ftp://223.194.44.54:1321
- Username: CV_2024
- Password: CV_2024



과제 제출 방법

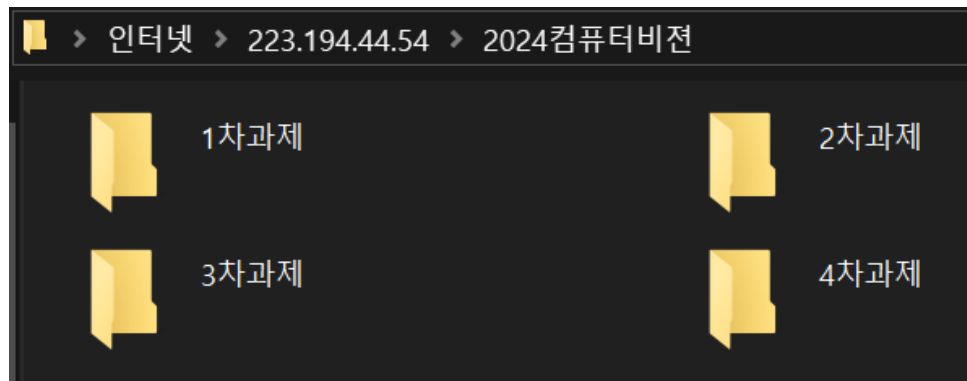
■ FTP Server

– 과제 폴더에 과제 압축 파일 업로드

- 업로드 후 파일 삭제 불가능함 → 과제 수정 시, 새로운 버전으로 업로드
- 마지막 버전의 과제 압축 파일로 과제 채점 예정

➢ Ex. 2024123456_홍길동_ver1.zip
2024123456_홍길동_ver2.zip
2024123456_홍길동_ver3.zip

→ "2024123456_홍길동_ver3.zip" 파일로 과제 채점



END OF PRESENTATION

Q&A