



KANGWON NATIONAL UNIVERSITY

# 컴퓨터비전 실습

1주차 | OpenCV

CVMILAB @ KNU

# TA 정보

- TA. 박진오, 이재훈
- 소속: 컴퓨터비전 & 의료영상처리 연구실
- 연락처 & 문의 방법
  - 이루리 Q&A 게시판 이용
- 과제 제출 방법
  - 이루리 업로드
  - 기간 : 다음주 수요일 오후 11시 59분까지 (기간 지날 시 0점)
- 출석
  - 동영상 시청 기간 : 매주 월~일
  - 실습 : 직접 호명
  - 실습실 자리 고정

- 오픈소스 영상처리 라이브러리
  - Linux, Windows, Android 등 다양한 플랫폼에서 지원함
  - **C++(실습)**, Python, Java 등의 언어로 제공
- 특징
  - 최신 논문 알고리즘이 주기적으로 업데이트됨
  - 이미지 읽기/저장, 연산 등에 대한 기본함수 제공

# 실습 환경

1. IDE: Windows Visual Studio 2022 Community (vs2019 사용 가능 / vs for mac 불가능)
2. OpenCV 4.5.5
3. Language: C/C++
4. 운영체제: 64비트 Windows 환경 (Windows 10 이상, **64비트 필수**)

※ linux 및 macos 관련 설명은 제공하지 않습니다.

# Windows 실습 환경 구축 1-1. Visual Studio 설치

## Visual Studio 2022 Community 다운로드 후 설치 (Enterprise 등 타 버전 사용가능)

※ 기존 Visual Studio 2022가 설치되어 있는 경우, 실행중인 Visual Studio를 모두 종료하고 시작 메뉴에서 “Visual Studio Installer”를 선택해 실행하고 “수정”을 클릭 한 뒤에, 1-2번으로 동일하게 진행합니다.

### Visual Studio 제품군을 만나보세요

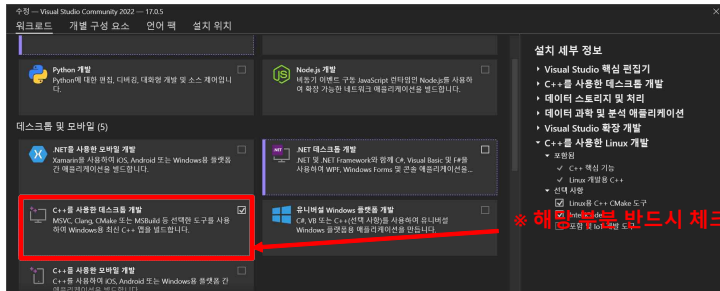
The screenshot displays the Visual Studio download page with three main product cards:

- Visual Studio** (version 17.1): Described as the most powerful IDE for Windows, macOS, and Linux. A dropdown menu is open, showing 'Community 2022' as the selected option, which is highlighted with a red box and a red arrow.
- Visual Studio for Mac** (version 8.10): Described as the most powerful IDE for macOS. A 'Visual Studio for Mac 다운로드' button is visible.
- Visual Studio Code** (version 1.62): Described as the most powerful IDE for Windows, macOS, and Linux. A 'Visual Studio 코드 다운로드' button is visible.

At the bottom, there is a note: "어떤 도구가 가장 적합한지 잘 모르시겠나요? 도움을 드릴 수 있습니다." and a section for "개발자 머신 OS" with buttons for Windows, macOS, and Linux.

# Windows 실습 환경 구축 1-2. Visual Studio 설치

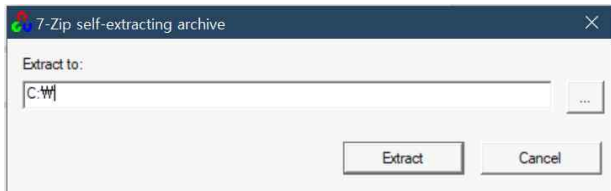
“C++를 사용한 데스크톱 개발” 체크한 뒤 설치 진행



# Windows 실습 환경 구축 2-1. OpenCV 설치

아래 링크를 클릭하여 OpenCV(opencv-4.5.5-vc14\_vc15.exe)를 다운로드하여 실행합니다. 그 후 압축 해제할 곳을 지정하여(C:\권장), 해당 위치에 압축을 해제하면 설치가 완료됩니다. [Download OpenCV from SourceForge.net](#)

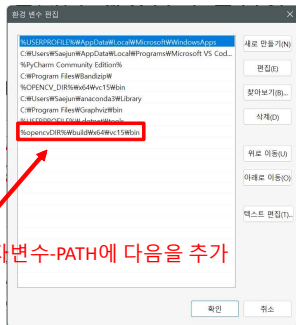
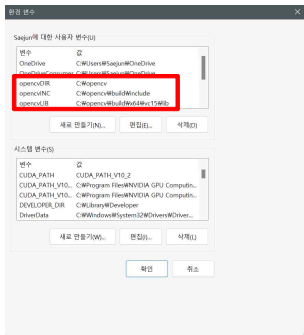
(뒤의 과정 ④에서 추가 설정을 진행합니다.)



# Windows 실습 환경 구축 2-2. OpenCV 설치

C:\opencv 폴더에 설치가 되었음을 확인하고, PATH에 디렉토리를 추가합니다.

내 PC > 속성 > 우측 “고급 시스템 설정” > 환경 변수 > 사용자 변수에 다음을 추가



사용자 변수-PATH에 다음을 추가



# Windows 실습 환경 구축 2-2. OpenCV 설치

# 고급 시스템 설정 - 환경변수 - 사용자 변수 - 새로 만들기

opencvDIR

C:\opencv

opencvINC

C:\opencv\build\include

opencvLIB

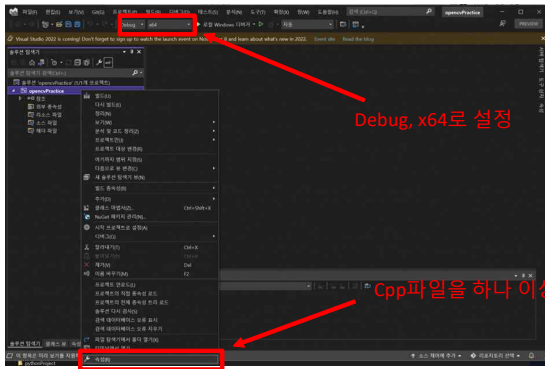
C:\opencv\build\x64\vc15\lib

# 고급 시스템 설정 - 환경변수 - 사용자 변수 - Path - 편집

%opencvDIR%\build\x64\vc15\bin

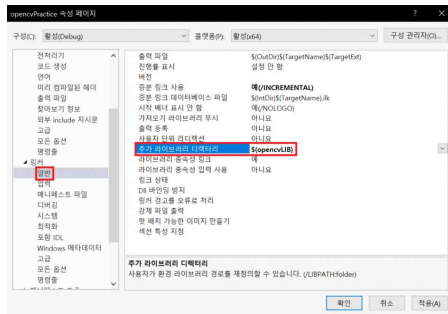
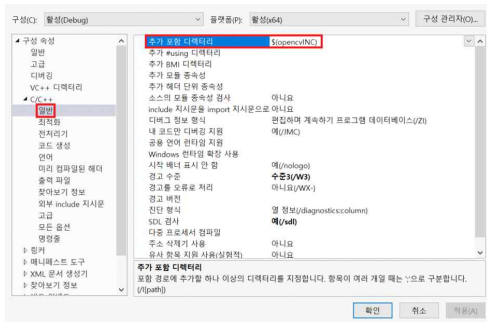
# Windows 실습 환경 구축 3-1. 프로젝트 만들기

C++프로젝트를 하나 생성하고 다음을 수행합니다.



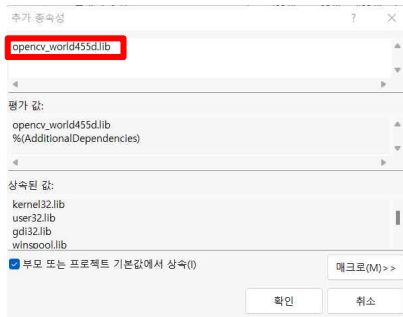
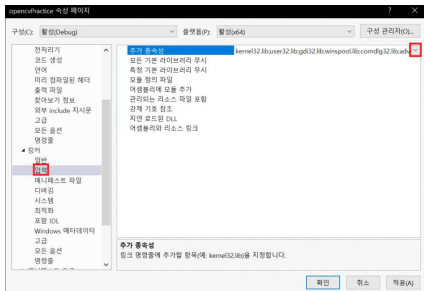
# Windows 실습 환경 구축 3-2. 프로젝트 만들기

- 다음을 수행합니다.
- \$(opencvINC) / \$(opencvLIB)



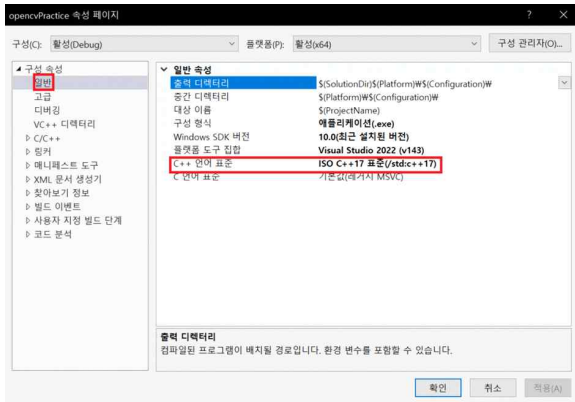
## Windows 실습 환경 구축 3-3. 프로젝트 만들기

- 다음을 수행합니다.
- `opencv_world455d.lib`



# Windows 실습 환경 구축 3-4. 프로젝트 만들기

- 다음을 수행합니다.



# 실습 1-1 | Hello OpenCV

## 문제

“Hello OpenCV와 Opencv의 Version을 출력하고 첨부해드린 Dog.jpg를 읽어서 화면에 출력합니다.

## 코드 및 실행결과

```
#include "opencv2/opencv.hpp"
#include <iostream>

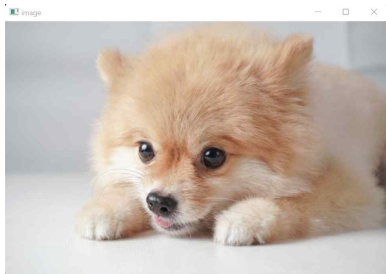
int main(void) {
    std::cout << "Hello OpenCV" << CV_VERSION << std::endl;

    cv::Mat img;
    img = cv::imread("dog.jpg");

    if (img.empty()) {
        std::cerr << "Image load failed!" << std::endl;
        return -1;
    }

    cv::namedWindow("image");
    cv::imshow("image", img);

    cv::waitKey();
    return 0;
}
```



# 실습 1-1 | Hello OpenCV

```
#include <iostream>
#include "opencv2/opencv.hpp"

int main()
{
    std::cout << "Hello OpenCV " << CV_VERSION << std::endl;

    cv::Mat img;
    img = cv::imread("dog.jpg");

    if (img.empty()) {
        std::cerr << "image load failed!" << std::endl;
        return - 1;
    }

    cv::namedWindow("image");
    cv::imshow("image", img);

    cv::waitKey();
    return 0;
}
```