

# 숫자 검출 및 인식

-컴퓨터 비전 미니 프로젝트



컴퓨터공학부 안희성

# INDEX

1. 프로젝트 정의

2. 프로젝트 처리과정

절차 알고리즘

전처리

인식

3. 프로젝트 진행상황

4. 프로젝트 계획

5. 진행상의 문제점

6. Q n A

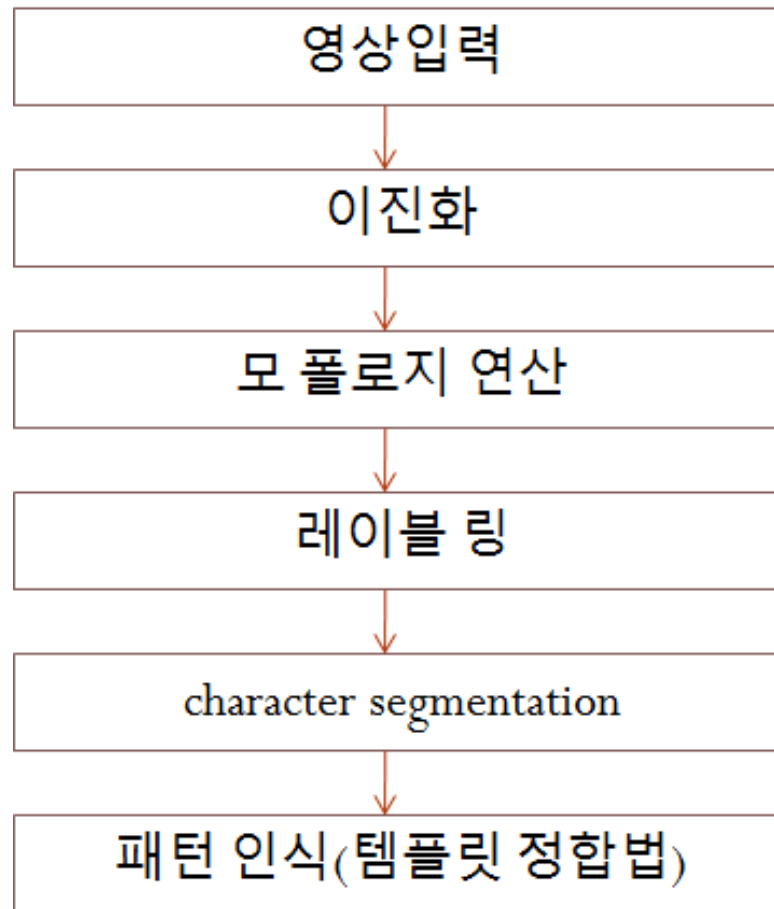
# 프로젝트 정의

# 프로젝트 정의

Open CV와 MFC를 사용해서 패턴인식의 한 분야인 숫자 검출 및 인식 프로그램을 인쇄되거나 손으로 쓴 문자를 Cam으로 판독·식별하여 컴퓨터가 이해할 수 있는 코드로 전환할 수 있게 만들어봄으로써 영상처리 기법을 이해하고 배워 본다.



# 절차 알고리즘



- ① Gray스케일의 1채널 이미지로 변환
- ② 임계 값을 기준으로 임계 값을 초과하는 픽셀과 임계 값 이하의 픽셀을 구분
- ③ 단순 임계치 방법, 평균 이진화, 반복 이진화, 적응 이진화, OTSU 등이 있다



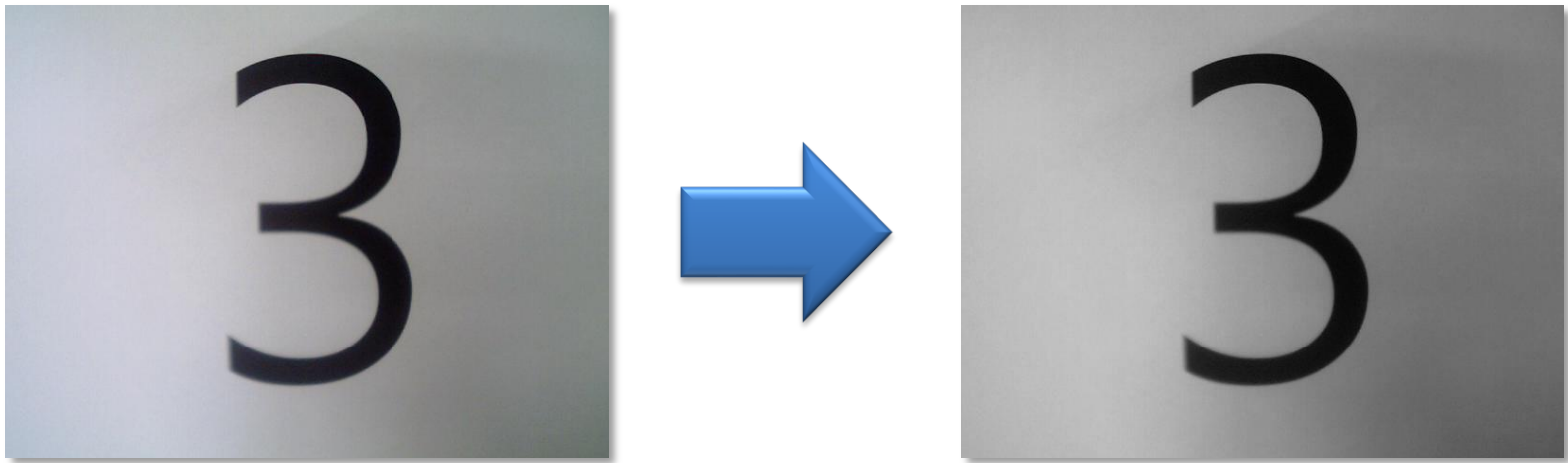
원영상  
OSTU



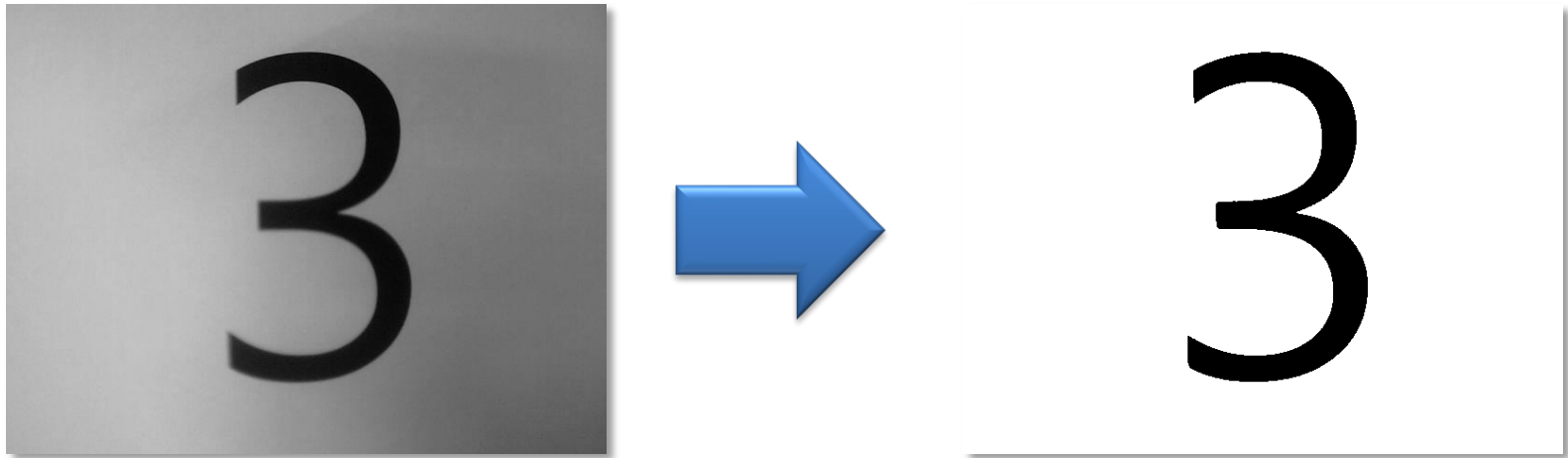
그레이 스케일



단순 임계치



- Cam에서 입력 받은 이미지를 RGB 컬러공간에서 GRAY 컬러공간으로 변환



- GRAY 이미지에 Threshold를 적용

## 모폴로지 연산

### ① 침식(erosion) 연산

이진영상에서는 객체영역을 흰색으로 볼 때 이들 영역이 줄고 검은색 영역이 늘어나며, 그레이 영상(또는 컬러영상)에서는 밝은 영역이 줄고 어두운 영역이 늘어나는 쪽으로 연산 결과가 변한다.

### ② 팽창(dilation) 연산

침식, 팽창연산은 잡영을 제거하는데 주로 사용된다.

### ③ 열기(opening) 연산

침식연산을 먼저 수행하고 팽창연산을 수행하여 잡영들을 제거할 수 있다.

### ④ 닫기(closing) 연산

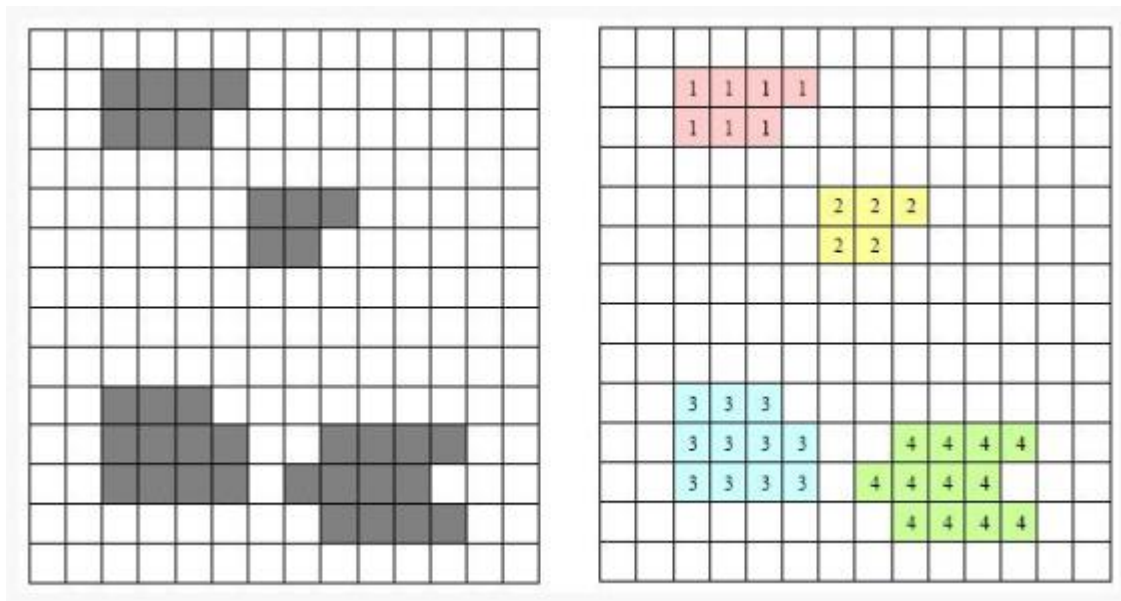
팽창연산을 먼저 수행하고 침식연산을 수행하여 구멍들을 제거할 수 있다.

### ⑤ Top-Hat (Black-Hat) 연산

이와 같이 열림 연산(또는 닫힘 연산)을 수행한 후 원본 영상과 열림 연산(또는 닫힘 연산)의 결과의 차이를 구하면 영상의 텍스처 정보(또는 윤곽선 정보)를 얻을 수 있다.



인접한 화소에 모두 같은 번호(Label)를 붙이고 연결되지 않은 다른 성분에는 다른 번호를 붙이는 일  
이진화된 영상의 픽셀수의 범위를 정하여 원하는 부분을 얻을 수 있다



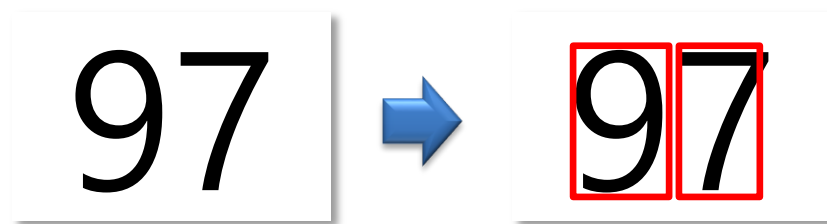
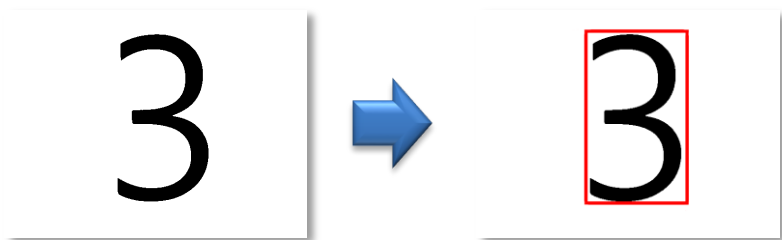
- 모폴로지한 이미지에 레이블링(라벨화)를 적용

# character segmentation

인식

숫자를 한 단위로 분할하는 과정

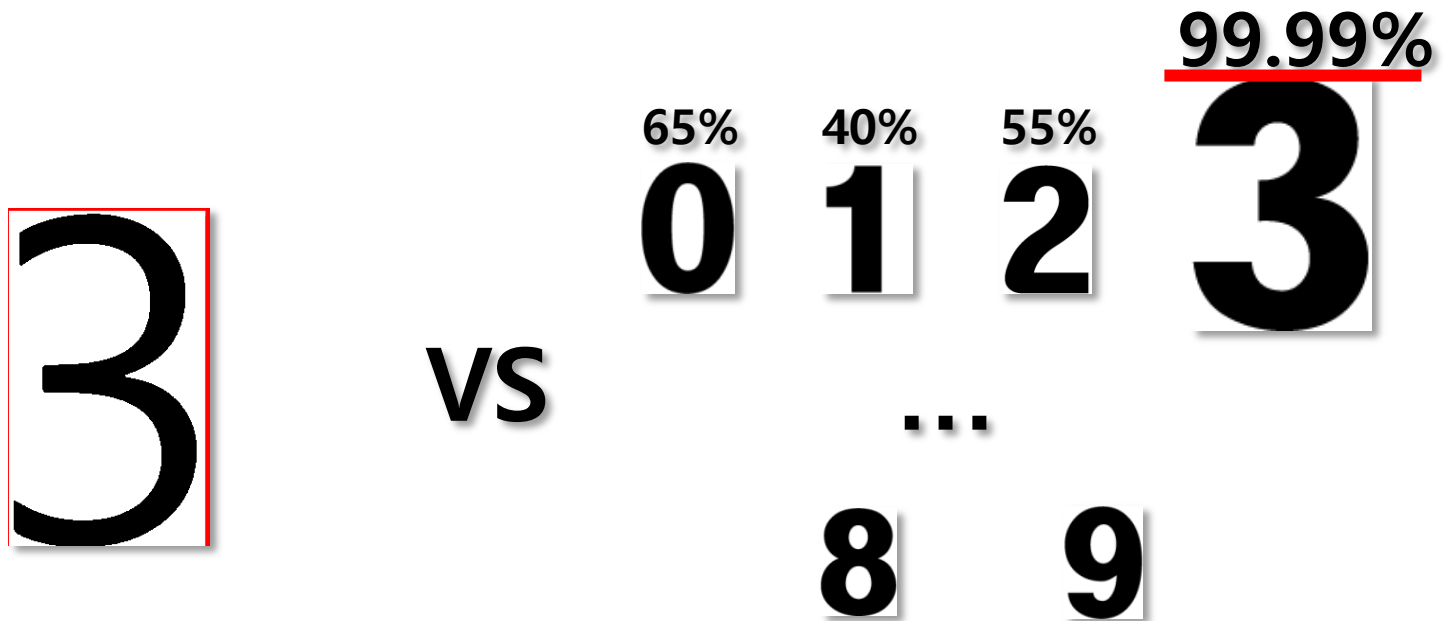
숫자(문자)인식은 한 글자 단위로 인식 하기 때문에 한 글자 영역을 구분해 낸다.



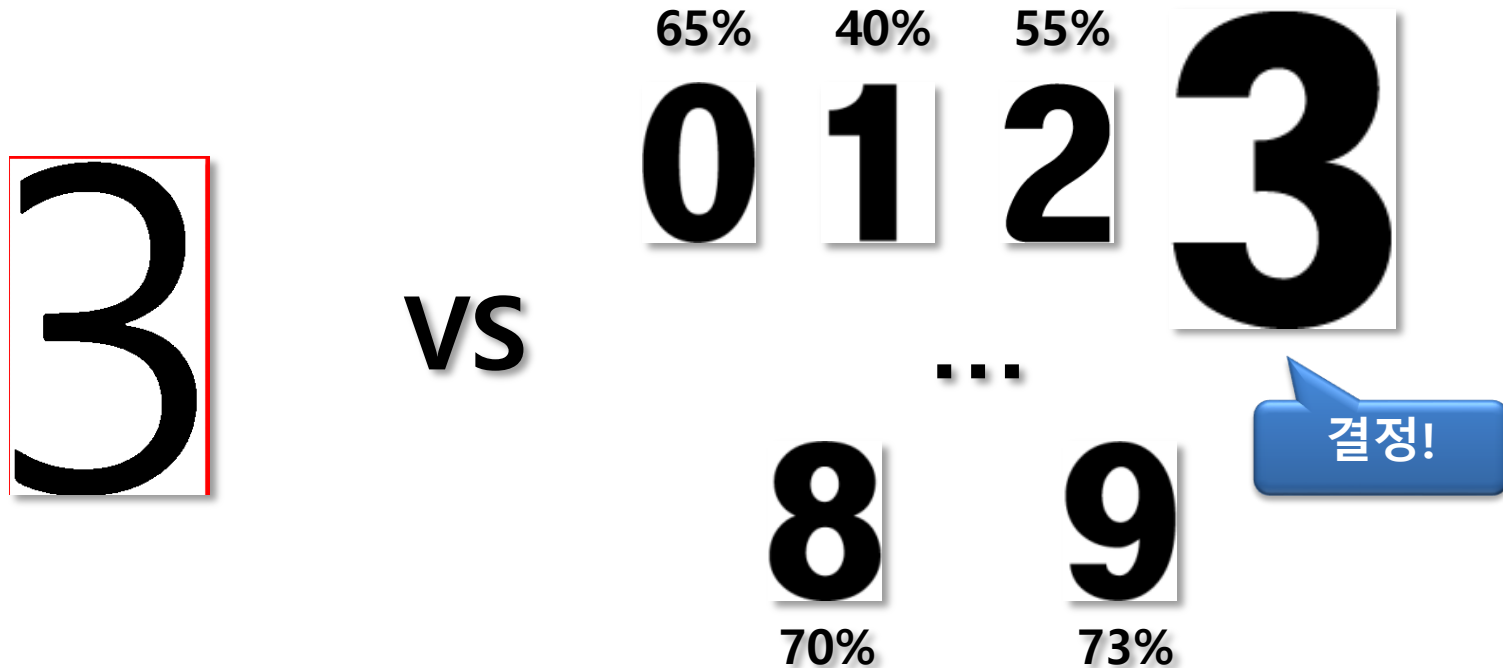
- 이미지에서 숫자 영역만 추출

- 이미지에서 한 단위로 분할

표준 패턴과의 비교를 통해 일치도, 유사도가 가장 높은 표준 패턴을 찾는다

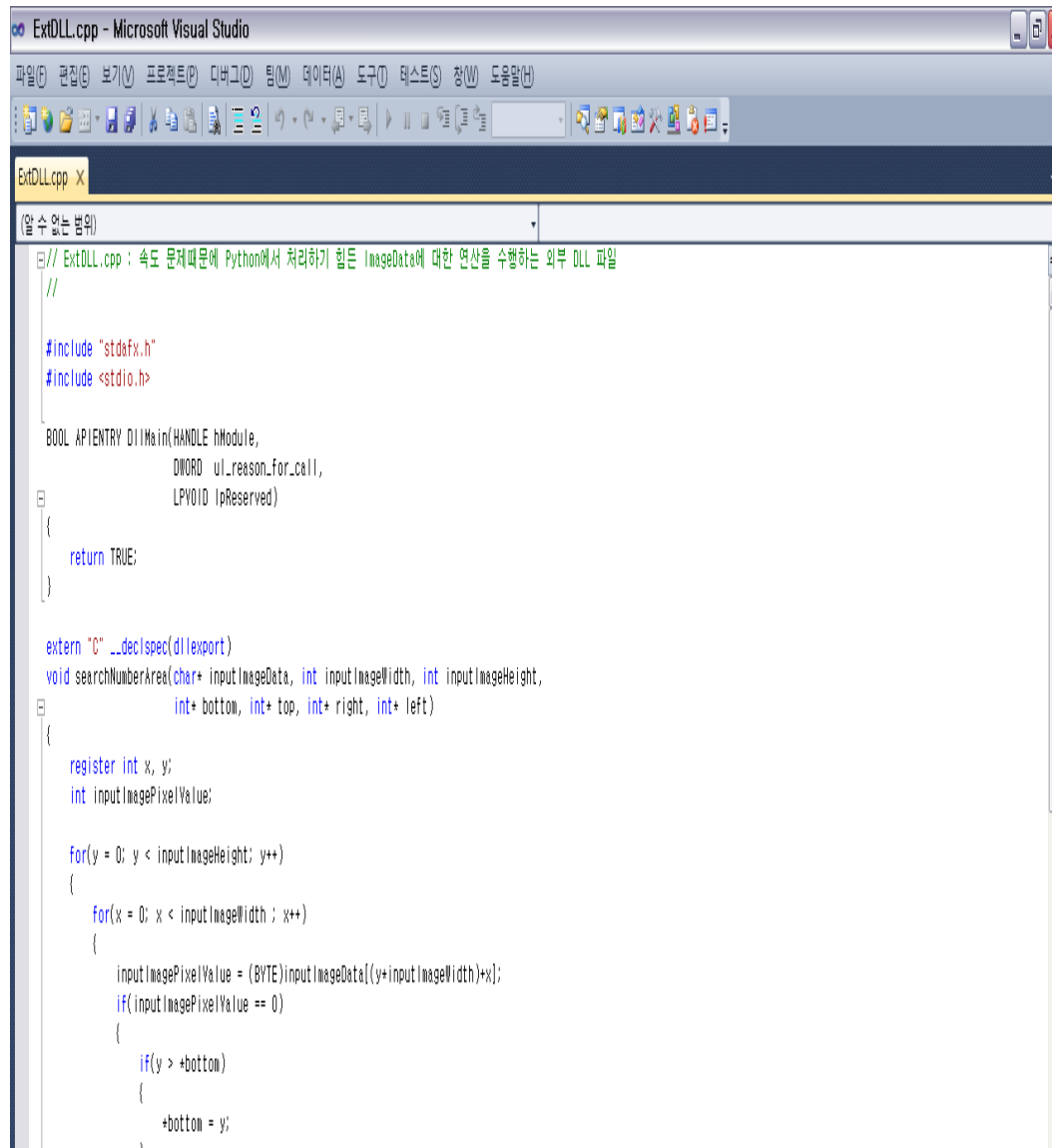


- 입력 영상에서 추출한 숫자 이미지를 기본 숫자 이미지들과 하나씩 비교
- 비교 결과를 토대로 가장 적합한 이미지를 결정



- 비교 결과를 토대로 가장 적합한 이미지를 결정

## 💡 코드 분석 중!



```
ExtDLL.cpp - Microsoft Visual Studio

파일(F) 편집(E) 보기(V) 프로젝트(P) 디버그(D) 팀(T) 데이터(A) 도구(T) 테스트(S) 창(W) 도움말(H)

ExtDLL.cpp x
(알 수 없는 범위)

// ExtDLL.cpp : 속도 문제때문에 Python에서 처리하기 힘든 ImageData에 대한 연산을 수행하는 외부 DLL 파일
//

#include "stdafx.h"
#include <stdio.h>

BOOL APIENTRY DllMain(HANDLE hModule,
                      DWORD ul_reason_for_call,
                      LPVOID lpReserved)
{
    return TRUE;
}

extern "C" __declspec(dllexport)
void searchNumberArea(char* inputData, int inputImageWidth, int inputImageHeight,
                      int* bottom, int* top, int* right, int* left)
{
    register int x, y;
    int inputImagePixelValue;

    for(y = 0; y < inputImageHeight; y++)
    {
        for(x = 0; x < inputImageWidth; x++)
        {
            inputImagePixelValue = (BYTE)inputData[(y*inputImageWidth)+x];
            if(inputImagePixelValue == 0)
            {
                if(y > *bottom)
                {
                    *bottom = y;
                }
            }
        }
    }
}
```

# 프로젝트 계획

	3월~4월 1주	4월 2~3주	5월 1주	5월 2주	5월 3주	5월 4주	5월 5주	6월 1주
주제선정								
자료조사								
코드분석								
설계 및 코딩								
Test 및 검사								

✧ 익숙지 않은 툴과 언어 사용의 어려움..

✧ 코드 이해의 어려움..

✧ 영상처리 대한 지식 부족..



Q & A





Thank you.

