



딥러닝 기반 질병 사진 분석 도구 개발

# 캡스톤디자인1 최종발표

김태훈 교수님 | InnoMedics | 20222030 양예은 | 20222014 함승희 |





# TABLE OF CONTENTS

01 팀원 소개



02 연구 배경



03 연구 상황



04 연구 성과



05 차후 계획

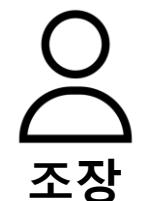




CHAPTER 01



# 팀원 소개



조장

**20222030 양예은**

- 모델 분석 및 모듈화
- MFC 및 모델 연동

**20222014 함승희**

- MFC 설계 및 개발



## 02 연구 배경





CHAPTER 02



# 연구배경

POINT. 01

## 원인 및 문제제기

- 병리 연구를 진행할 때 선행작업으로서 진행되는 병리사진 분석 작업이 수작업으로 진행
- 연구자 별 주관에 의한 기준으로 판단하여 객관성이 부족
- 한마리의 쥐를 통한 SEM 이미지 생성이 평균적 열장 연구를 위한 실험쥐는 수십마리 → 수백장의 이미지를 수작업 분석 (보름정도)

POINT. 02

## 필요성

- 라벨링에 일관성을 제공할 수 있음
- 수작업의 번거로움을 줄일 수 있으며 작업 시간을 효과적으로 축소



## 03 연구 상황





CHAPTER 03



# 연구 상황 - 연구 계획

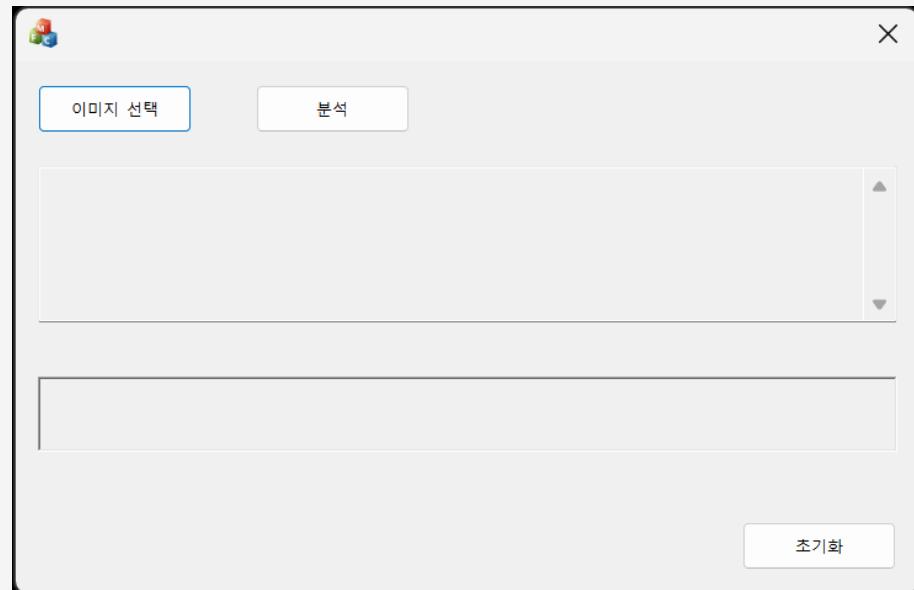




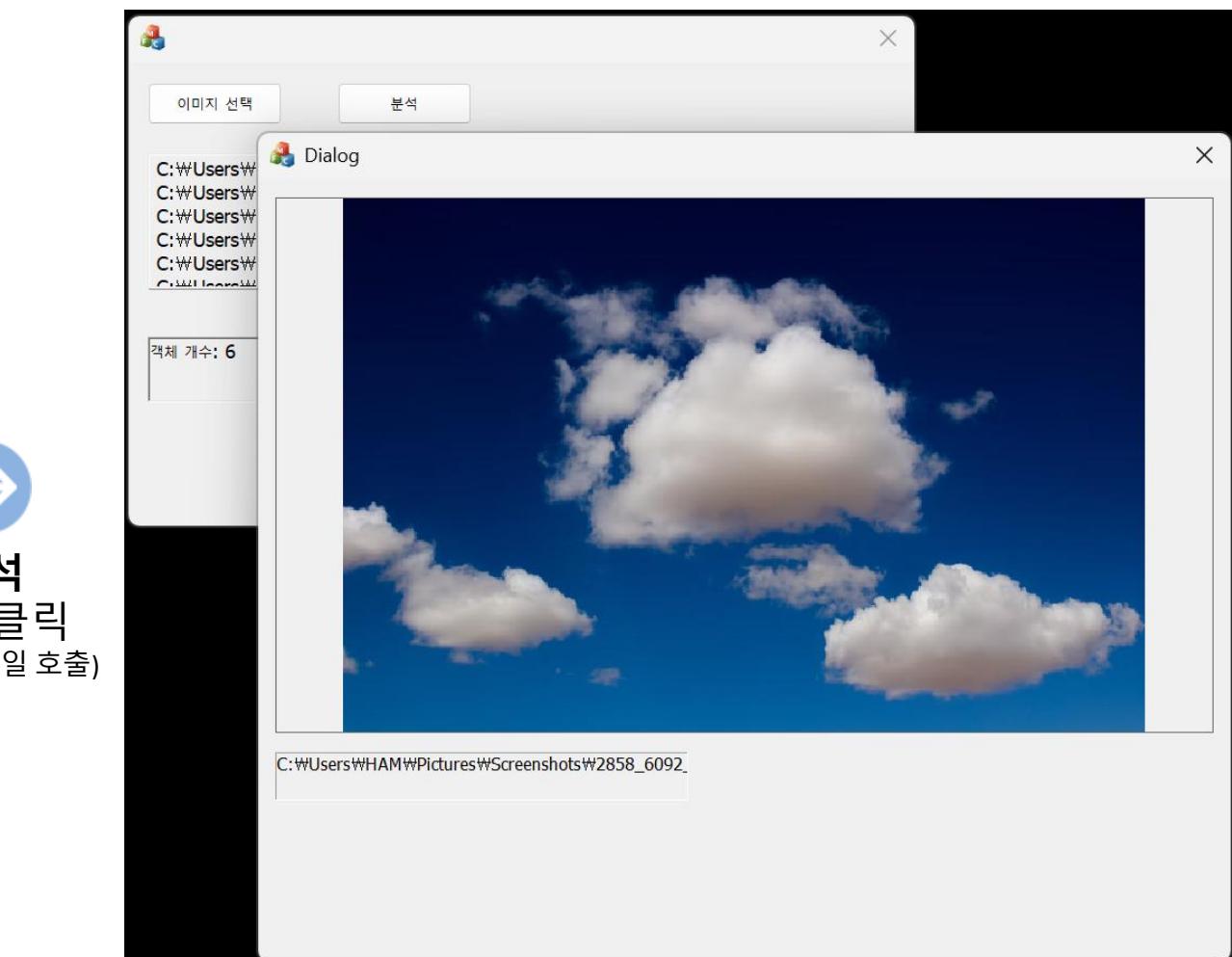
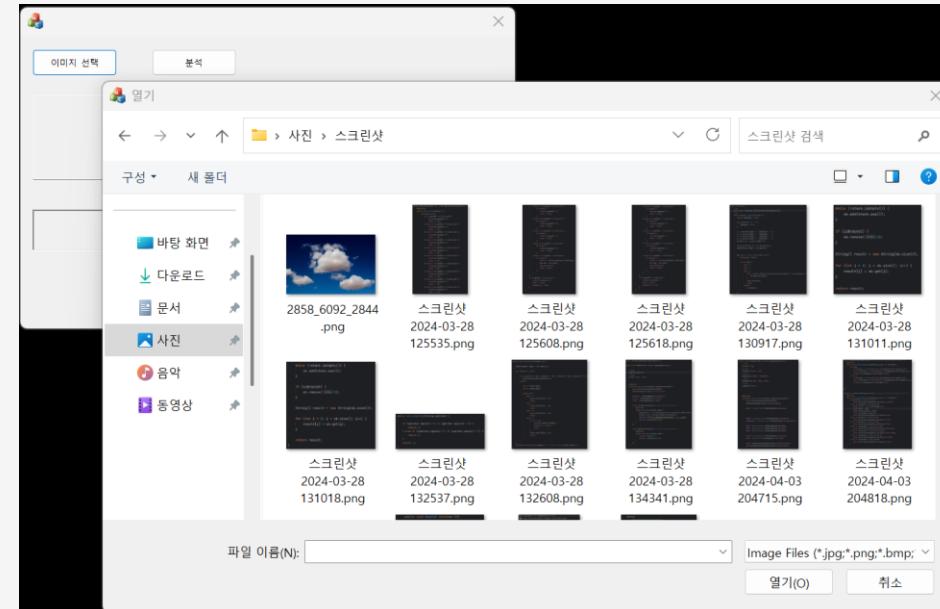
CHAPTER 03



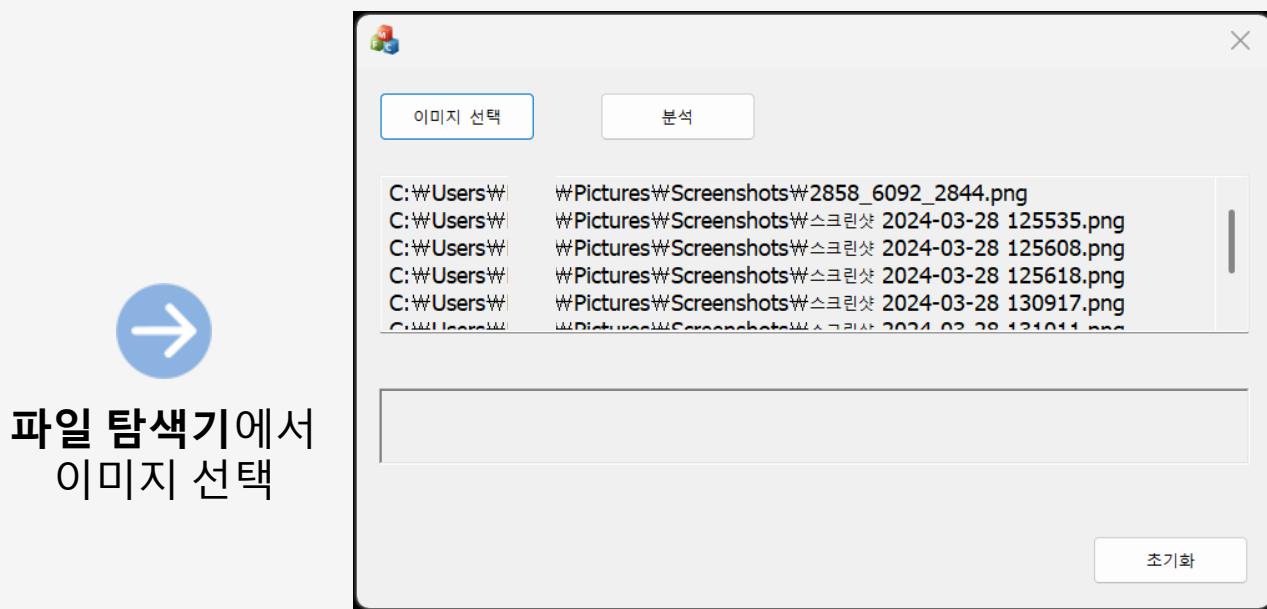
# 연구 상황 - 시스템 구성도



→  
이미지 선택  
버튼 클릭



→  
분석  
버튼 클릭  
(파이썬 파일 호출)



→  
파일 탐색기에서  
이미지 선택



CHAPTER 03



# 연구 상황 - 모델 분석

```
class Dataset(BaseDataset):
    CLASSES = [ 'x' for x in range(255)]
    CLASSES.append('cell')

    def __init__(...):
        pass

    def __getitem__(self, i):
        # read data
        image = cv2.imread(self.images_fps[i])
        mask = cv2.imread(self.masks_fps[i], 0)

        # extract certain classes from mask (e.g. cars)
        masks = [(mask == v) for v in self.class_values]
        mask = np.stack(masks, axis=-1).astype('float')

        # apply augmentations
        if self.augmentation:
            sample = self.augmentation(image=image, mask=mask)
            image, mask = sample['image'], sample['mask']

        # apply preprocessing
        if self.preprocessing:
            sample = self.preprocessing(image=image, mask=mask)
            image, mask = sample['image'], sample['mask']

        return image, mask

    def __len__(self):
        return len(self.ids)
```

```
train_transform = [
    albu.HorizontalFlip(p=0.5),
    albu.VerticalFlip(p=0.5),
    albu.ShiftScaleRotate(shift_limit=0, scale_limit=(0.3, 0.9), rotate_limit=90, p=0.2, border_mode=cv2.BORDER_REPLICATE),
    albu.RandomBrightnessContrast(brightness_limit=(-0.2, 0.2), contrast_limit=(-0.2, 0.2), p=0.5),
    albu.OneOf(
        [
            albu.CLAHE(p=1),
            # albu.RandomBrightness(p=1),
            # 오류 발생 원인: RandomBrightness → 수정 필요!
            albu.RandomBrightnessContrast(p=1), # ✅ 수정된 부분
            albu.RandomGamma(p=1),
        ],
        p=0.6,
    ),
    albu.OneOf(
        [
            albu.Blur(blur_limit=3, p=1),
            albu.MotionBlur(blur_limit=3, p=1),
        ],
        p=0.6,
    ),
    albu.OneOf(
        [
            # albu.RandomBrightness(p=1),
            # 오류 발생 원인: RandomBrightness → 수정 필요!
            albu.RandomBrightnessContrast(p=1), # ✅ 수정된 부분
            albu.HueSaturationValue(p=1),
        ],
        p=0.6,
    ),
]
return albu.Compose(train_transform)
```

```
def to_tensor(x, **kwargs):
    return x.transpose(2, 0, 1).astype('float32')

def get_preprocessing(preprocessing_fn):
    _transform = [
        albu.Lambda(image=preprocessing_fn),
        albu.Lambda(image=to_tensor, mask=to_tensor),
    ]
    return albu.Compose(_transform)
```

마스크  
필터링이미지  
증강  
albumentation정규화 및  
구조화



## 04 연구 성과

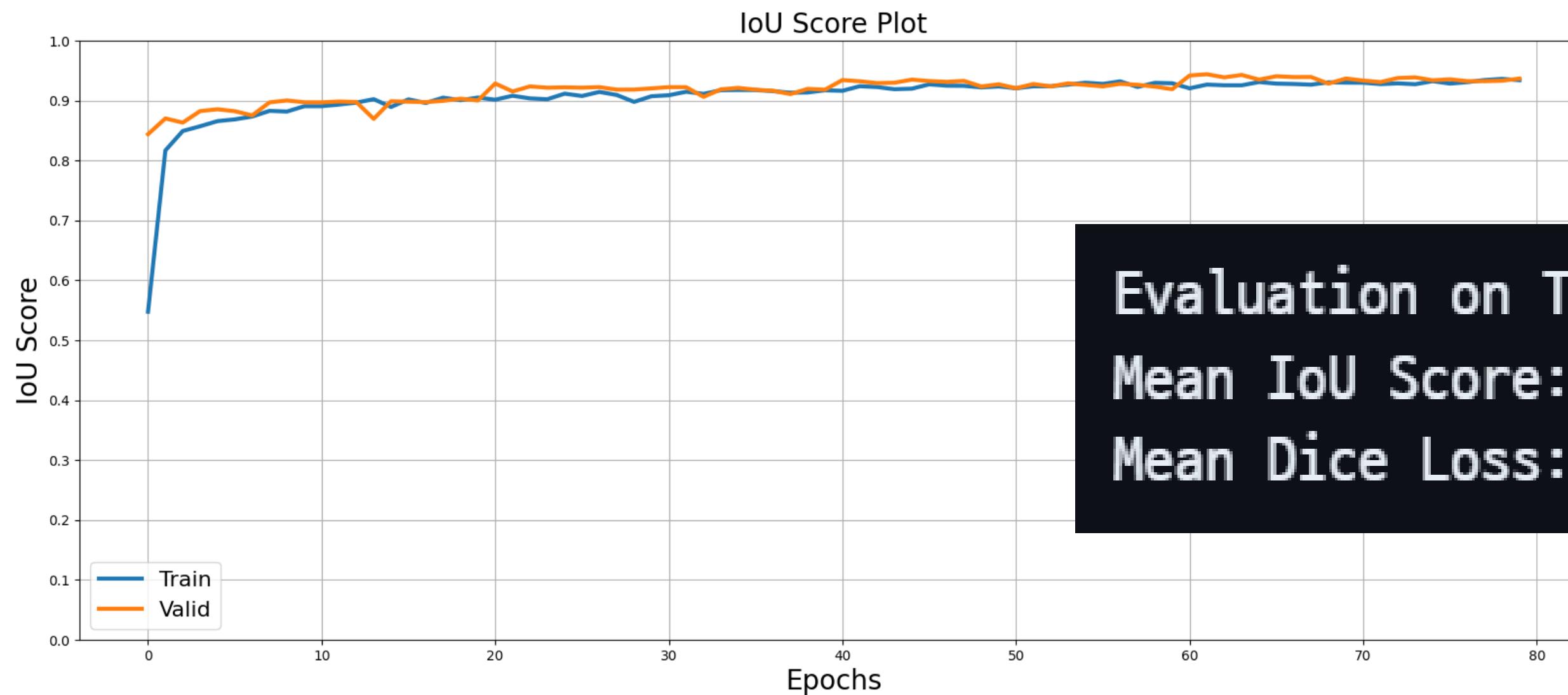




CHAPTER 04



# 연구 성과 - 테스트 결과



Evaluation on Test Data:  
Mean IoU Score: 0.9362  
Mean Dice Loss: 0.0339



CHAPTER 04



# 연구 성과 - 테스트 결과





CHAPTER 04



# 연구 성과 - 데모 영상

The screenshot shows a dark-themed code editor interface. On the left is a sidebar with icons for file operations, search, and other tools. The main area has a terminal tab open, displaying the following text:

```
PS C:\Users\yy\Desktop\school\CAPSTONE\main> python .\processing_module.py
Traceback (most recent call last):
  File ".\processing_module.py", line 196, in <module>
    preprocessor = DataPreprocessor(workspace_path=workspace_path, dataset_path=dataset_path, auto_labeling_path=auto_labeling_path)
TypeError: __init__() got an unexpected keyword argument 'dataset_path'
```

The terminal window title is "main". The status bar at the bottom shows the file name "main" and line count "4".



# 05 차후 계획





CHAPTER 05



# 차후 계획



연구 내용	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월
주제 선정 / 보고서 작성										
MFC 설계 및 개발										
U-Net 모델 분석 및 재구성										
MFC, U-Net 연동										
테스트 및 평가										
문제점 보안, 성능 개선										
프로젝트 결과 보고서 작성										

MFC, U-NET 연동

테스트 및 평가

문제점 보안, 성능 개선

결과 보고서 작성



감사합니다

