

Report #1

제목 : NxN Mosaic Button

제출일 : 2016. 12. 12

학과 : 디지털정보공학과

학번 : 201300712

이름 : 김윤기

1. 문제 정의

- 20x20 개의 버튼이 있는 GUI 프로그램을 작성
- 버튼을 누르면 해당 버튼의 이미지가 Mosaic 이미지로 바뀌고 다시 누르면 원래의 이미지가 된다.
- 사용하는 이미지 파일은 자신만의 고유한 이미지 파일로 한다.
- 이미지 파일의 사이즈가 버튼과 맞지 않으면 조정 한다.

2. 설계 결과

initialize() 로 이미지를 받고 MosaicImage()으로 20X20의 타일로 나눈다.

```
tiles[i].addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent event) {  
        // TODO Auto-generated method stub  
        for(int j = 0; j<SIZEj++){  
            Color bcol = getAverageColor(imageData[j]);  
            if( event.getSource() == tiles[j]){  
                if(Flag[j] == 0){  
                    tiles[j].setIcon(new  
ImageIcon(OneColorBufferedImage.getBufferedImage(bWidth, bHeight, col)));  
                    Flag[j] = 1;  
                }  
                else{  
                    tiles[j].setIcon(new ImageIcon(imageData[j]));  
                    Flag[j] = 0;  
                }  
            }  
        }  
    }  
});
```

actionListener() 는 event가 발생했을 때 어떠한 작동을 해 줄것인지 결정한다. 여기서는 입력이 들어오면 Flag가 변화하고 모자이크 동작을 수행한다. Flag는 타일에 입력이 들어온 것을 저장해주는 array이다.

oneColorBufferedImage는 한 이미지의 RGB값을 리턴한다. 이것을 getAverageColor으로 모자이크시킨다.

이제 실행을 시키면 20X20 개의 타일로 구성된 이미지가 뜬다. 어떠한 타일을 누르면 그 부분만 모자이크 처리되어 출력이 되고, 다시 누르면 원상태로 돌아오게 된다.

3. 결론 및 소감

처음 과제는 mosaic를 하는 프로그램이었다. 이 프로그램은 기능적인 측면으로 크게 세 부분으로 나눌 수 있다. 먼저 이미지를 받아와서 타일로 나누는 코드, actionListener를 이용하여 버튼이 눌리는 동작을 나타내는 코드, 마지막으로 어떠한 이미지를 모자이크시키는 코드이다. 가장 구현이 어려웠던 부분은 역시 actionListener를 이용하는 부분이었다. 이미지를 받는 부분과 모자이크시키는 부분은 svn 코드에 기본적인 틀이 짜여져 있어서 어렵지 않았지만, actionListener는 타일의 버튼에서 어떠한 이벤트가 발생할 때 어떤 프로그램이 동작하도록 설계를 해주는 이벤트 처리 클래스이다. 버튼이 누르고, 다시 누르면 되돌아오도록 하는 동작을 구현하기가 어려웠다. 그래서 Flag라는 변수를 이용, 타일의 버튼의 현재 상태를 담아두는 역할을 하도록 하고 구현을 하였다. 그리고 OneColorBufferedImage 라는 다른 source file의 class를 불러와서 이용하는 것이 자유롭다는 것을 알았다.