

Opening Exercise

Write a method that turns pixels with an average intensity less than 85 to **GREEN**, pixels with an average intensity less than 170 to **RED**, and all other pixels to **BLUE**.

What Is Wrong With This Answer?

```
public void exercise01()
{
    for ( Pixel p : this.getPixels() )
    {
        double averageIntensity = p.getAverage();
        if ( averageIntensity < 85 )
            p.setColor( Color.green );
        if ( averageIntensity < 170 )
            p.setColor( Color.red );
        if ( averageIntensity < 256 )
            p.setColor( Color.blue );
    }
}
```

Guarded Action

manipulate a pixel
only if
it satisfies a particular condition

the **if** statement

```
if ( condition is true )  
    take the action
```

Alternative Action

Idea

manipulate a pixel

in one way

if it satisfies a particular condition

or in another way

if it satisfies a particular condition

Alternative Action

Implementation

the **if-else** statement

```
if ( condition is true )  
    take one action  
else  
    take the other action
```

Range Selection

Using nested **if-else** statements to find ranges:

```
if ( value is in the first range )  
    take the first action  
else if ( value is in the second range )  
    take the second action  
...  
else  
    take the final action
```

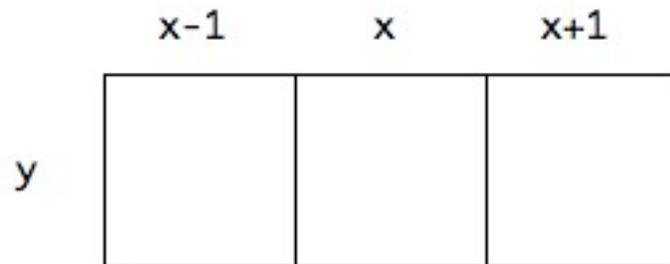
One Hallmark of Good Design

"When making a design choice, always have at least two alternatives. That way, you can at least know that you are not doing the worst possible thing."

— paraphrased from Kent Beck

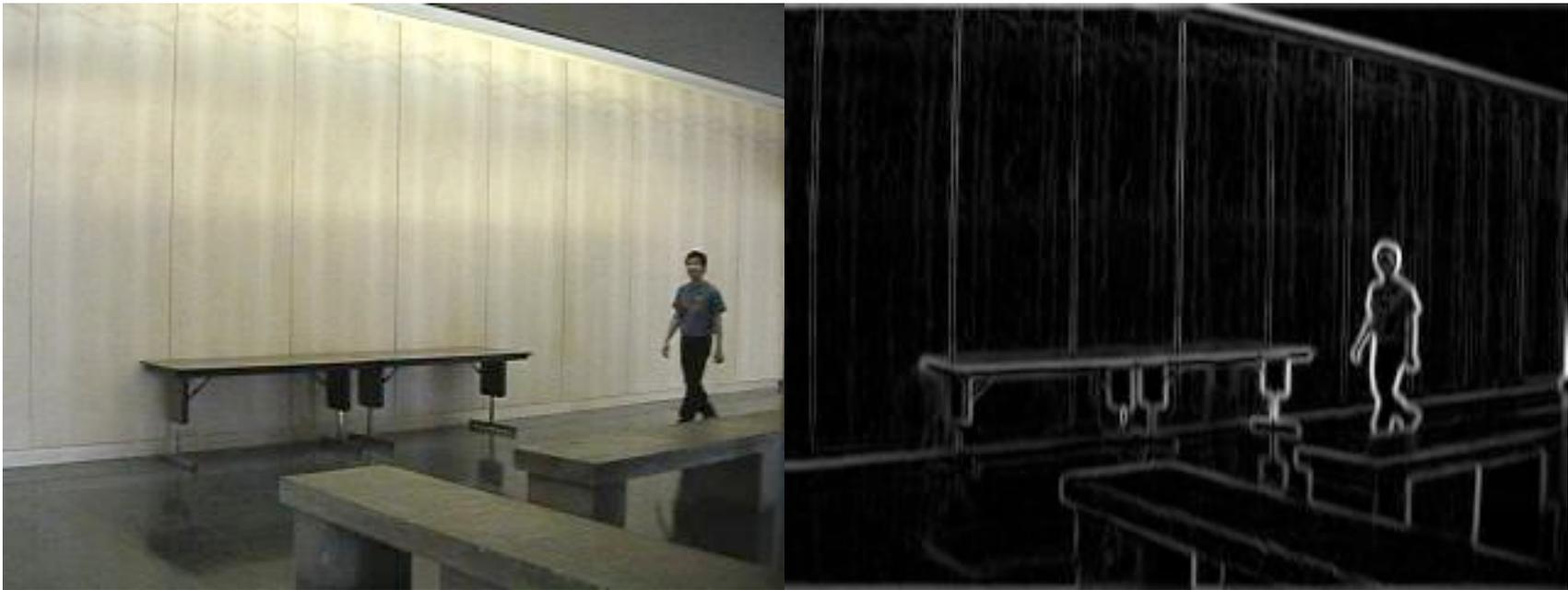
Finding Objects in Images

Boundaries appear when neighboring pixels have very different colors.



This is the task of **edge detection**.

An Application of Edge Detection



computer vision

Boolean Conditions

```
if ( condition is true ) ...
```

simple boolean expression

```
x < y          p.colorDistance(epsilon)
```

conjunction (**and**)

```
x < y  &&  p.colorDistance(epsilon)
```

disjunction (**or**)

```
x < y  ||  p.colorDistance(epsilon)
```

Homework 2

What are the high-level operations?

- insert the image
- draw a box (several times)
 - draw a horizontal line
 - draw a vertical line

Design your solution in this way:

- create an empty method for each operation
- design, implement, and test each method one at a time