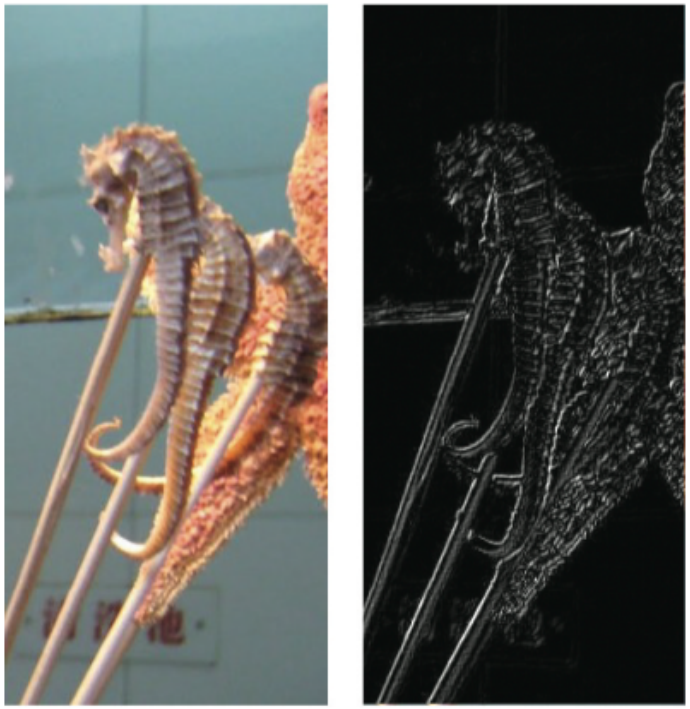
Edge Detection

Edge Detection

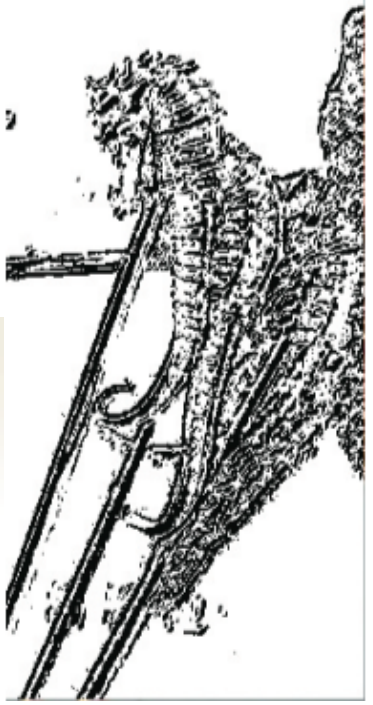
- Blurring is averaging across pixels.
- Edge detection is looking for differences between pixels.
  - We draw lines that our eyes see—where the luminance changes.
- If the pixel changes left-to-right, up and down, then we make our pixel black. Else white.

Edge Detection



```
def edge(source):
  for px in getPixels(source):
    x = getX(px)
    y = getY(px)
    if y < getHeight(source)-1 and x < getWidth(source)-1:
      sum = getRed(px)+getGreen(px)+getBlue(px)
      botrt = getPixel(source,x+1,y+1)
      sum2 = getRed(botrt)+getGreen(botrt)+getBlue(botrt)
      diff = abs(sum2-sum)
      newcolor = makeColor(diff,diff,diff)
      setColor(px,newcolor)
```

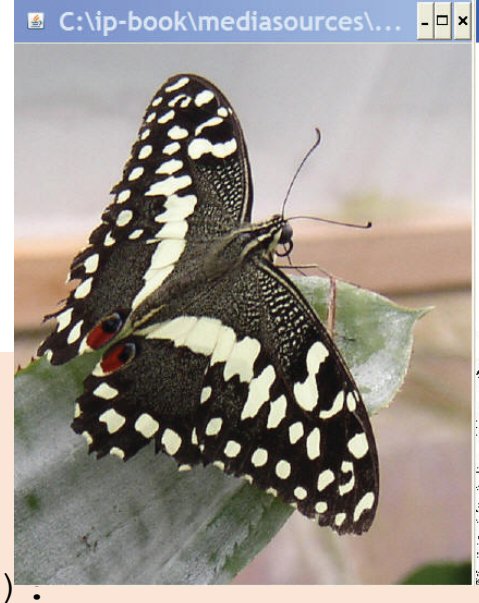
Yet Another Version

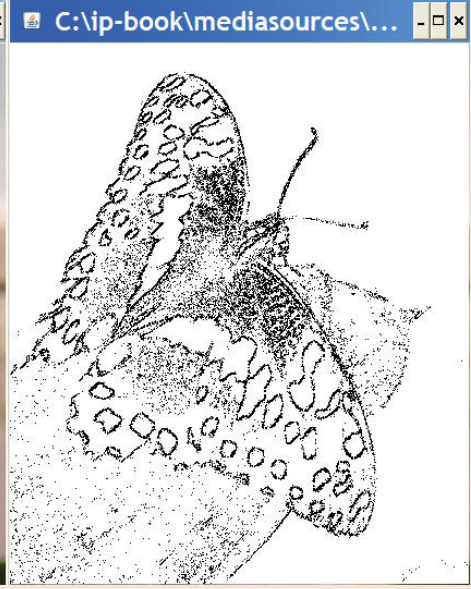


```
def luminance(pixel):
  r = getRed(pixel)
  g = getGreen(pixel)
  b = getBlue(pixel)
  return (r+g+b)/3

def edgedetect(source):
  for px in getPixels(source):
    x = getX(px)
    y = getY(px)
    if y < getHeight(source)-1 and x < getWidth(source)-1:
      botrt = getPixel(source,x+1,y+1)
      thislum = luminance(px)
      brlum = luminance(botrt)
      if abs(brlum-thislum) > 10:
        setColor(px,black)
      if abs(brlum-thislum) <= 10:
        setColor(px,white)
```

More Careful Edge Detection






```
def lineDetect(filename):
  orig = makePicture(filename)
  makeBw = makePicture(filename)
  for x in range(0,getWidth(orig)-1):
    for y in range(0,getHeight(orig)-1):
      here=getPixel(makeBw,x,y)
      down=getPixel(orig,x,y+1)
      right=getPixel(orig,x+1,y)
      hereL=(getRed(here)+getGreen(here)+getBlue(here))/3
      downL=(getRed(down)+getGreen(down)+getBlue(down))/3
      rightL=(getRed(right)+getGreen(right)+getBlue(right))/3
      if abs(hereL-downL)>10 and abs(hereL-rightL)>10:
        setColor(here,black)
      if abs(hereL-downL)<=10 and abs(hereL-rightL)<=10:
        setColor(here,white)
  return makeBw
```

Here we look in all four directions.