try {

BufferedImage img = ImageIO.read(file.getInputStream()); // 取文件流构造Image对象

BufferedImage imageR = null ; //截取后的图像

int widthR = img.getWidth(null); // 得到源图宽

int heightR = img.getHeight(null); // 得到源图宽

int widthW = 100; //最后生成图片大小

int heightW = 100; //最后生成图片大小

int widthSub=0; //截取图片方法的位移大小

int heightSub=0; //截取图片方法的位移大小

BufferedImage imageS =new BufferedImage(widthW, heightW,

BufferedImage.TYPE\_INT\_RGB);;

//生成原图画布

if((widthR>heightR)&&(heightR>100)){

widthSub=(widthR-heightR)/2;

imageR=img.getSubimage(widthSub, heightSub, heightR, heightR);

imageS.getGraphics().drawImage(imageR, 0, 0, widthW, heightW, null); // 绘制缩小后的图

ImageIO.write(imageS, "jpeg", tagFile); //生成文件

}else if((widthR<heightR)&&(widthR>100)){

heightSub=(heightR-widthR)/2;

imageR=img.getSubimage(widthSub, heightSub, widthR, widthR);

imageS.getGraphics().drawImage(imageR, 0, 0, widthW, heightW, null); // 绘制缩小后的图

ImageIO.write(imageS, "jpeg", tagFile);

}else{

imageR=new BufferedImage(widthW, heightW,

BufferedImage.TYPE\_INT\_RGB);

imageR.getGraphics().drawImage(img, 0, 0, widthW, heightW, null); // 绘制缩小后的图

ImageIO.write(imageR, "jpeg", tagFile);

}

CommunityActivity.fileCopy(path, path1);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}