

图像在网络上的传输方法 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/284/2021_2022__E5_9B_BE_E5_83_8F_E5_9C_A8_E7_c104_284984.htm 图像在网络上传输

，一般都是传输编码压缩后的图像。例如在本地将一张图像编码压缩成JPEG格式后通过网络传输出去，接收端接收到之后再将其解码，用于显示，或者直接存储到存储介质上。当然，如果不考虑传输数据量，或者有一些特殊需求也可以直接将图像对象序列化后传输。本文只讨论前者，关于图像对象序列化可以参考我的另一篇文章《图像序列化》。那么整理一下，这个过程也就分为3个步骤：1、发送端图像编码。

2、以字节流的方式在网络上传输；3、接收端解码。图像编码因为图像编码解码主要目的是针对图像在网络中的传输，所以编码之后的图像不必保存在硬盘上，可以直接放入一个字节数组。

```
public byte[] getCompressedImage(BufferedImage image)...{ byte[] imageData = null. try ...{ ByteArrayOutputStream baos = new ByteArrayOutputStream(). ImageIO.write(image, "jpg", baos). imageData = baos.toByteArray(). } catch (IOException ex) ...{ imageData = null. } return imageData. }
```

图像解码 接收端接收到表示图像数据的字节数组后，对其进行解码，得到图像对象。因为我们在发送端将其编码成JPEG格式，所以可以直接在接收端使用ImageIO对其进行解码。

```
public BufferedImage getDecompressedImage(byte[] imageData)...{ try ...{
```

```
ByteArrayInputStream bais = new
```

```
ByteArrayInputStream(imageData). return ImageIO.read(bais). }
```

```
catch (IOException ex) ...{ return null. } }
```

网络传输 因为图像编码

之后是一个存在于内存中的字节数组，所以可以使用IO流的方式将其发送到网络的接收端，接收端建立链接将其接收。最常用的例如建立 Socket 连接等等。这部分代码讲IO的书里都有，我就不在这里写了。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com