

对象存储OSS

图片服务手册



# 图片服务手册

## 介绍

### 图片服务介绍

阿里云OSS图片处理服务（Image Service，简称 IMG），是阿里云OSS对外提供的海量，安全，低成本高可靠的图片处理服务。用户将原始图片上传保存在OSS上，通过简单的 RESTful 接口，在任何时间、任何地点、任何互联网设备上对图片进行处理。图片处理服务提供图片处理接口，图片上传请使用OSS上传接口。基于 IMG，用户可以搭建出跟图片相关的服务。

### 图片服务基础功能

图片服务提供以下功能：

- 获取图片信息
- 图片格式转换
- 图片缩放、裁剪、旋转
- 图片添加图片、文字、图文混合水印
- 自定义图片处理样式
- 通过管道顺序调用多种图片处理功能

### 图片服务(IMG)与对象存储(OSS)的关系

- IMG不上传图片，IMG处理的图片是OSS用户上传到Bucket里的Object。也就是说用户可以通过OSS和IMG都可以访问同一张图片。
- IMG只处理来自于OSS的图片。
- IMG和OSS有同样的区域，不同的访问域名。

#### 注意

- Bucket和Object见OSS基本概念
- IMG的图片上传请参考图片上传
- IMG的访问域名请参考图片访问域名
- OSS的访问域名请参考OSS访问域名

## 基本概念

### 基本概念

Object（对象、文件）

在IMG中，用户操作图片的基本数据单元是Object。即OSS对应的Object，单个Object（即每张图片）允许的最大大小是20MB。

Object 命名规范：

1. 使用 UTF-8 编码。
2. 长度必须在 1-1023 字节之间。
3. 不能以 "/" 或者 "\" 字符开头。

## Channel（频道）

Channel是 IMG上的命名空间，也是计费、权限控制、日志记录等高级功能的管理实体。IMG名称在整个图片处理服务中具有全局唯一性，且不能修改。一个用户最多可创建 10 个 Channel，但每个Channel中存放的object的数量没有限制，存储容量每个Channel最高支持2PB。目前Channel跟OSS的Bucket相对应，即用户只能创建与自己在OSS上Bucket同名的Channel。

Channel 命名规范：

1. 只能包括小写字母，数字，短横线(-)。
2. 必须以小写字母或者数字开头和结尾。
3. 长度必须在 3-63 字节之间。

## Style（样式）

图片处理服务提供用户将图片的处理操作和参数保存成一个别名，即样式。一系列操作，利用样式功能后，只需要用一个很短的URL就能实现相同的效果。

- 一个Channel下面有多个样式，目前一个Channel允许最多有50个样式。
- 样式适应于Channel 下面的object图像变化操作。假如在A Channel下面有样式，名称为abc，样式内容是100w.jpg（按宽缩略成100，保存成jpg格式）那么A Channel下面所有的object都能使用样式abc，实现缩略成 100w.jpg的效果。
- 样式的作用范围只在一个Channel下，即A Channel不能使用B Channel 的样式。

Style 命名规范：

1. 长度为1-63个字符。
2. 可以包含数字、大小写字母、下划线(\_)、短横线(-)以及小数点(.)

样式访问示例

## 处理字符串

图片服务定义了处理字符串，包含两个部分转换参数、转换格式：

- 转换参数由一个或多个键值对（以"\_"连接）组成，“值”在前“键”在后，“值”为数字类型，“键”为一位字母。
- 转换格式是一种特殊的转换参数，用户指定转换格式，图片服务对原图处理并返回用户期望的图片文件格式。（支持格式是: jpg，jpeg，webp，png，bmp）。

## 分隔符

图片处理服务使用通过URL来访问处理的图片。所以需要分隔符来区分一些关键字段。不要在使用的图片文件名称中包含图片处理服务设定的分隔符。不然会导致解析出错的问题。

分隔符名称	分隔符	含义
处理分隔符	@	区分Object名称跟处理字符串
样式分隔符	@ !	区分Object跟样式内容，详见样式
管道分隔符		区分多种操作，详见管道

## 数据中心及访问域名

图片服务的数据中心和OSS的数据中心相对应。用户在OSS的某个数据中心上创建一个Bucket，然后选择开通图片服务，那么对应的Channel也属于该数据中心。开通的数据中心及域名见访问域名

## 示例

下面以一个例子来说明上面提到的概念

```
http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h.jpg
```

以图片访问的URL为例子  
 image-demo：用户的频道的名字，即Channel  
 img-cn-hangzhou.aliyuncs.com：图片杭州地区访问域名，即Endpoint  
 upload.jpg：待处理的图片的原图名字，即Object  
 @：处理分隔符  
 100w\_100h.jpg：处理字符串  
 100w\_100h：将原图进行处理的参数，即转换参数  
 jpg：将原图根据参数处理后的保持的格式，即转换格式

## 限制

1. 保存的最终格式只支持jpg, png, bmp, webp (注意gif默认保存成jpg)。
2. 文件的大小限制在20MB。
3. 对缩略后的图片的大小有限制，目标缩略图的宽与高的乘积不能超过4096 \* 4096，而且单边的长度不能超过4096 \* 4。
4. 管道目前限制在4个。

其他限制可以参考文档其他参数的详细描述。

## 访问域名

# IMG开通Region和Endpoint对照表

目前，IMG有以下几个数据中心(Region)对公众提供服务，各区域的访问域名(Endpoint)设置如下：

Region中文名称	Region英文名称	Endpoint
------------	------------	----------

杭州数据中心	oss-cn-hangzhou	img-cn-hangzhou.aliyuncs.com
青岛数据中心	oss-cn-qingdao	img-cn-qingdao.aliyuncs.com
北京数据中心	oss-cn-beijing	img-cn-beijing.aliyuncs.com
深圳数据中心	oss-cn-shenzhen	img-cn-shenzhen.aliyuncs.com
上海数据中心	oss-cn-shanghai	img-cn-shanghai.aliyuncs.com

上传图片需要的OSS访问域名请参见OSS数据中心及访问域名

## 接入图片服务

### 图片URL规则

#### 图片URL构成规则

图片服务都是使用标准的HTTP的GET请求来访问的，所有的处理参数也是编码在URL中的。

#### 直接获取原图

用户有两种方式访问图片，分别是：

- 三级域名
- 自定义域名

### 三级域名访问的URL

```
http://channel.<endpoint>/object
```

这里的endpoint指的是用户图片所在的Region的图片服务的访问域名，关于访问域名请参考访问域名，object为用户所关联的channel上存储的原图片。

### 自定义域名访问的URL

另外一种方式是通过用户绑定的图片服务域名，也就是自定义域名来访问，形式如下：

```
http://userdomain/object
```

其中userdomain为用户开通图片服务绑定的自定义域名，这个域名会关联到一个channel，这里假设用户自定义域名userdomain已经CNAME到channel.endpoint这个三级域名上。如何绑定图片服务的自定义域名可以参考开通图片服务。object为用户所关联channel上存储的原图片。

#### 通过处理参数访问原图

如果用户对原图进行一定的处理然后返回，同样有两种形式，URL的格式如下：

通过三级域名访问  
http://channel.<endpoint>/object@action.format

channel：用户的IMG频道  
endpoint：用户的Channel所在数据中心的访问域名  
object：用户上传在OSS上的图片文件  
action：用户对图片做的处理  
format：用户指定处理后的图片格式

通过用户自定义域名访问  
http://userdomain/object@action.format

一个典型的URL的例子如下：

http://bucket.endpoint/object@100w\_100h\_90Q.jpg 三级域名  
http://userdomain/object@100w\_100h\_90Q.jpg 自定义域名

其中的@100w\_100h\_90Q为转换的具体action，jpg为转换的format，合在一起的100w\_100h\_90Q.jpg称为转换字符串，用来指定对于目标图片的操作，通过指定转换字符串，生成并返回另一张转换处理后的图片。

一个典型的转换字符串，如“@100w\_100h\_90Q.jpg”，代表需要一张宽（w）100px、高（h）100px、绝对质量（Q）90%、jpg格式的图片。

@ 120w\_120h\_90Q .jpg

初始符
转换参数
转换格式

### 通过样式访问原图

为了简化使用，用户可以将特定的处理方法保存为样式，这样以后调用同样的处理方法只需要指定某个样式即可。使用样式来进行图片处理的URL形式如下：

http://userdomain/object@!style

@!是目前支持的样式分隔符，style 是样式的名称

如：http://image-demo.img.aliyuncs.com/example.jpg@!pipe1

其中pipe1 是样式名称。

## 关键词

## 关键词

### 顺序无关

转换参数中键值对是循序无关的，即"120w\_120h\_90Q"和"90Q\_120w\_120h"都能取到想要的图片，系统会对参数按照本规范以下定义的顺序重新排序后再处理。（由于参数的顺序不同有时会表达不同的语义，如"100w\_100h\_2x"表达的是“先缩放到100\*100，再放大2倍”，即得到200\*200的图片；而"2x\_100w\_100h"按照字面顺序理解是“先放大2倍再缩放到100\*100”，即得到100\*100的图片，为了避免这样的理解误差，同时简化处理方式，IMG会对参数按照文档中出现的顺序排序后处理。上例中的"2x\_100w\_100h"会被理解为“100w\_100h\_2x”，得到200\*200的图片。）

### 覆盖处理

如果转换参数中出现多个相同“键”，后面定义的覆盖前面定义。如“120w\_120h\_240w”等同于“120h\_240w”。

### 冲突处理

见每个参数中关于冲突的说明。

### 长边与短边

关于“长边”和“短边”的定义需要特别注意，它们表达的是在缩放中相对比例的长或短。“长边”是指原尺寸与目标尺寸的比值大的那条边；“短边”同理。如原图400 \* 200，缩放为800 \* 100，（ $400/800=0.5$ ， $200/100=2$ ， $0.5 < 2$ ），所以在这个缩放中200那条是长边，400是短边。

### URL安全的Base64位编码

在图片处理服务里会有很多参数需要变成Base64位编码，参考[RFC4648](#)。注意这里的URL 安全Base64位编码只是用于在水印操作某些特定参数(文字水印的文字内容，文字颜色，文字字体及图片水印的水印object)里，不要将其用来签名字符串(Signature)的内容。编码的格式是：

- 先将内容编码成Base64结果;
- 将结果中的加号“+”替换成中划线“-”;
- 将结果中的斜杠“/”替换成下划线“\_”;
- 将结果中尾部的“=”号全部保留;

以Python为例子

```
import base64
input='wqy-microhei'
print(base64.urlsafe_b64encode(input))
```

## 快速开始

## 图片快速开始

## 基于控制台快速开始：

1. 登录OSS控制台，开通OSS。
2. 创建一个Bucket。
3. 上传和下载图片。

具体见控制台快速开始文档。

## 快速了解图片的上传下载

开始使用SDK之前，请先参考用户手册关于上传下载文件的功能介绍。

IMG是使用RESTful API来操作的，所有的请求都是标准的HTTP请求。

- 图片服务中的图片的上传等同于OSS的上传。OSS提供了多种类型的上传文件的方法，如使用单次PUT请求完成的简单上传、使用网页表单直接上传的表单上传、以及用于大文件上传的分片上传。
- 图片服务的下载可以通过浏览器等发送HTTP GET请求图片的URL即可获得图片。

## 快速体验图片服务的功能

图片服务功能体验请参考<http://oss-demo.aliyuncs.com/image/index.html>

## 用户鉴权

### 用户签名验证

如果用户需要不经过任何授权，通过浏览器即可匿名访问图片服务来处理图片，需要在创建Bucket的时候将Bucket的权限设置为公共读。

创建Bucket

- 控制台：创建Bucket
- SDK：Java SDK-Bucket中新建Bucket
- API：Put Bucket

设置Bucket权限

- 控制台：创建Bucket权限设置
- SDK：Java SDK-Bucket中设置Bucket ACL
- API：Put BucketACL

默认创建的Bucket权限是私有读写，Object默认继承Bucket的权限。如果用户需要通过图片服务访问私有的Object，需要进行身份鉴权。

图片服务的用户鉴权方式和OSS是一致的，有两种鉴权方式：

- 在Header中包含签名
- 在URL中包含签名

具体可参考在Header中包含签名和在URL中包含签名。

## 在URL中包含签名

假定用户绑定的域名www.test.com对应的频道名字为image-demo，object名字为example.jpg，转换字符串为100w.jpg。

首先需要计算Signature字段，计算方法如下：

```
Signature = base64(hmac-sha1(AccessKeySecret,
    VERB + "\n"
    + Content-MD5 + "\n"
    + Content-Type + "\n"
    + Expires + "\n"
    + CanonicalizedOSSHeaders
    + CanonicalizedResource))
```

- AccessKeySecret表示签名所需的密钥
- VERB表示HTTP请求方法，例如PUT，GET，DELETE等
- Content-MD5表示请求内容数据的MD5值，对于图片处理服务这里为空字符串
- Content-Type表示请求内容的类型，对于图片处理服务这里为空字符串
- Expires表示授权给用户URL签名过期时间
- CanonicalizedOSSHeaders表示HTTP中的ObjectMeta组合，对于图片处理服务这里为空字符串
- CanonicalizedResource 表示用户想要访问的OSS资源，在图片处理服务中这项的组成，格式为/channelname/object@处理参数。
- 构建CanonicalizedResource的方法如下：
  - i. 将CanonicalizedResource置成空字符串("")
  - ii. 放入要访问的图片服务资源："/channelname/object"（无Object则不填）
  - iii. 在结尾添加处理参数：@处理参数，（无处理参数则不填）。此时CanonicalizedResource例子如：/channelname/object@100w.jpg。
  - iv. 如果涉及样式管理操作，那么将这些查询字符串及其请求值按照字典序，以&分割，添加到CanonicalizedResource中。此时的CanonicalizedResource例子：  
/channelname?style&styleName=YourStyleName

例子中的CanonicalizedResource为/image-demo/example.jpg@100w.jpg。

注意：上例中的转换字符串可以是简单缩略，文字水印，图片水印、管道和样式（样式的分隔符是@!）

这里需要注意的是，使用URL签名中Expires和CanonicalizedResource不能为空。

最后生成在URL签名，必须在参数后包含OSSAccessKeyId、Expires、Signature这三项，具体生成方法可以参考OSS的API文档中的在URL中包含签名，上文的例子生成的URL签名如下：

```
http://www.test.com/example.jpg%40100w.jpg?OSSAccessKeyId=j4y55h3z88ihxxhlr9nhjjs&Expires=1392949804&Signature=IDBJ09e8Ow4GaPRM1yIf7pIH/CI%3D
```

## 在Header中包含签名

除了在URL中包含签名之外，还可以在HTTP请求的Header中包含签名，签名是由Authorization这个Header指定的，具体的构成规则如下：

```
"Authorization: OSS " + AccessKeyId + ":" + Signature
Signature = base64(hmac-sha1(AccessKeySecret,
    VERB + "\n"
    + Content-MD5 + "\n"
    + Content-Type + "\n"
    + Date + "\n"
    + CanonicalizedOSSHeaders
    + CanonicalizedResource))
```

- AccessKeySecret表示签名所需的密钥
- VERB表示HTTP请求方法，例如PUT，GET，DELETE等
- Content-MD5表示请求内容数据的MD5值，对于图片处理服务这里为空字符串
- Content-Type表示请求内容的类型，对于图片处理服务这里为空字符串
- Date表示此次操作的时间，且必须为HTTP1.1中支持的GMT格式
- CanonicalizedOSSHeaders表示 http中的object meta组合，对于图片处理服务这里为空字符串
- CanonicalizedResource构造方法请参考上文URL签名中的CanonicalizedResource的生成方法。

需要注意的

- Date和CanonicalizedResource不能为空
- 如果请求中的Date时间和OSS服务器的时间差正负15分钟以上，OSS图片处理服务将拒绝该服务，并返回HTTP 403错误。

## 使用SDK处理图片

### 使用OSSSDK访问图片服务

本节主要介绍如何使用OSS的Python SDK去获取private bucket的图片处理服务，因为图片处理服务都是GET操作，所以使用OSS Python SDK主要以Get Object 为主。

以Python SDK为例,在Python SDK里有一个get\_object操作，传入的参数一般是bucket，object。

#### OSS

获取bucket: image-demo，object：example.jpg

```
bucket = 'image-demo'
object = 'example.jpg'
self.oss.get_object(bucket, object)
```

#### 图片服务

##### 简单缩略

获取bucket: image-demo , object: example.jpg ,

转换字符是 : 100w\_100h.jpg

```
bucket = 'image-demo'
object = 'example.jpg'
query = '100w_100h.jpg'
object = object + '@' + query
self.oss.get_object(bucket, object)
```

## 图片水印

获取bucket: image-demo , object: example.jpg ,

转换字符是 : watermark=1&object=cGFuZGEucG5n&t=90&p=5

```
bucket = 'image-demo'
object = 'example.jpg'
query = 'watermark=1&object=cGFuZGEucG5n&t=90&p=5'
object = object + '@' + query
self.oss.get_object(bucket, object)
```

## 文字水印

获取bucket: image-demo , object:example.jpg

转换字符是 : watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ

```
bucket = 'image-demo'
object = 'example.jpg'
query = ' watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ '
object = object + '@' + query
self.oss.get_object(bucket, object)
```

## 样式

获取bucket: image-demo , object:example.jpg

样式名 : pipe1

```
bucket = 'image-demo'
object = 'example.jpg'
style = ' pipe1 '
object = object + '@!' + style
self.oss.get_object(bucket, object)
```

## 管道

获取bucket: image-demo , object:example.jpg

管道操作: 200w.jpg|watermark=1&object=cGFuZGEucG5n&t=90&p=5

```
bucket = 'image-demo'
object = 'example.jpg'
```

```
query = ' 200w.jpg|watermark=1&objectcGFuZGEucG5n&t90&p5'
object = object + '@' + query
self.oss.get_object(bucket, object)
```

## 图片上传

## 图片上传

图片服务处理的图片来自于OSS，所以图片的上传实际是往OSS上同名Bucket上传的。所有的上传请参考OSS用户手册中的上传文件

假如用户需要使用杭州的图片服务，域名为img-cn-hangzhou.aliyuncs.com **上传前提条件**：

- 1 同区域OSS存储空间(Bucket)。例如叫oss-sample，杭州的OSS访问域名为oss-cn-hangzhou.aliyuncs.com。
- 2 通过控制台或者SDK上传图片。例如上传logo.png到oss-sample。

## 控制台上传图片

- 控制台：上传文件

## SDK上传图片

- SDK：Java SDK-Object中PutObject

## 注意事项

- 必须是同区域的OSS和IMG
- 必须是同名的Bucket和Channel
- 必须使用OSS的域名上传

## 图片缩放

### 单边固定缩略

#### 单边固定缩略

可以对图片某一边（宽或高）进行固定到一个长度，另外一边按照比例进行调整。

参数

名称	描述	取值范围
w	指定目标缩略图的宽度	1-4096
h	指定目标缩略图的高度。	1-4096
l	目标缩略图大于原图是否处理。 值是1, 即不处理, 是0, 表示处理	0/1, 默认是0

### 注意事项

- 对缩略后的图片的大小有限制，目标缩略图的宽与高的乘积不能超过 $4096 * 4096$ ，而且单边的长度不能超过 $4096 * 4$ 。
- 如果只指定宽度或者高度，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。

### 示例

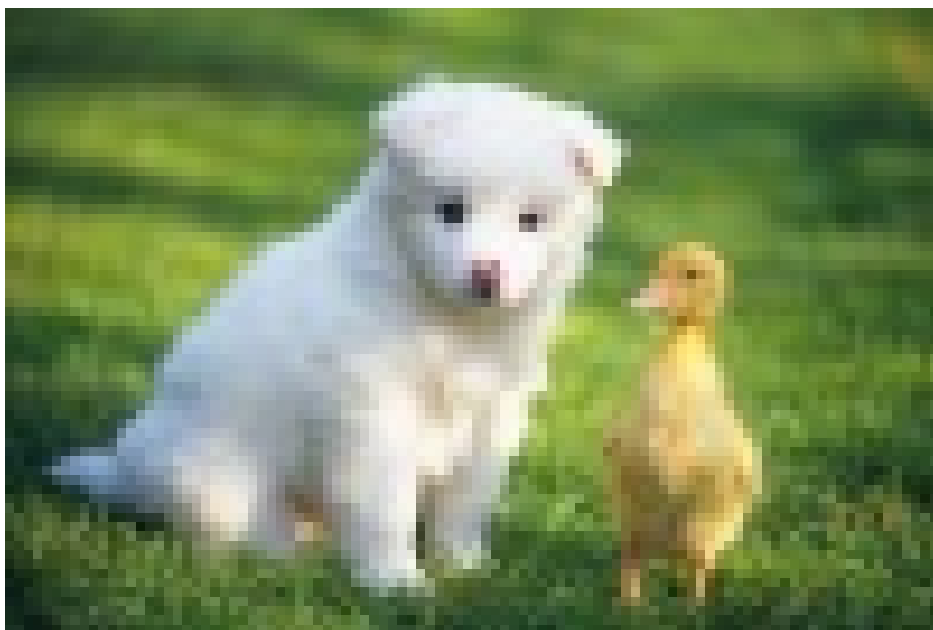
- 将图缩略成高度为100，宽度按比例处理。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h>



- 将图缩略成宽度为100，高度按比例处理。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w>



- 将图缩略成宽度为500，高度按比例处理，如果目标缩略图大于原图不处理。

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@500w\\_1l](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@500w_1l)



## 指定宽高缩略

### 指定宽高缩略

可以对图片指定宽或高原图，按照长边短边进行调整。

#### 参数

名称	描述	取值范围
----	----	------

w	指定目标缩略图的宽度	1-4096
h	指定目标缩略图的高度。	1-4096
e	缩放优先边，默认值：0：长边（默认值）由于图片缩放过程中，原图尺寸与缩放尺寸不一定是相同比例，需要指定以长边还是短边优先进行缩放，如原图200 * 400（比例1:2），需要缩放为100 * 100（比例1:1）。长边优先时，缩放为50 100；短边优先时(e=1)，缩放为`100 200`，若不特别指定，则代表长边优先	0表示按长边优先，默认值 1表示按短边优先
l	目标缩略图大于原图是否处理。如果值是1, 即不处理，是0，表示处理	0/1, 默认是0

### 注意事项

- 对缩略后的图片的大小有限制，目标缩略图的宽与高的乘积不能超过4096 \* 4096，而且单边的长度不能超过4096 \* 4。
- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。

### 示例

- 将图缩略成宽度为100，高度为100，按长边优先

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w\\_0e](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w_0e)



- 将图缩略成宽度为100，高度为100，按短边优先

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w\\_1e](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w_1e)



## 强制宽高缩略

### 强制宽高缩略

可以强制指定目标缩略图的高度和宽度，忽略原图的宽高比。注意这可能会导致图片变形。

#### 参数

名称	描述	取值范围
w	指定目标缩略图的宽度	1-4096
h	指定目标缩略图的高度。	1-4096
e	缩放优先边，如果是强制缩略，值是:2	2（强制缩略）
l	目标缩略图大于原图是否处理。如果值是1，即不处理，是0，表示处理	0/1，默认是0

#### 注意事项

- 此操作会导致图变形。
- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。

#### 示例

- 将图强制缩略成宽度为100，高度为100

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w\\_2e](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w_2e)



## 按比例缩放

### 按比例缩放

可以通过指定一个比例百分比参数，让图片按照指定的比例进行缩略或者放大。

#### 参数

名称	描述	取值范围
p	倍数百分比。小于100，即是缩小，大于100即是放大。	1-1000

#### 注意:

- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。
- 如果参数p跟w, h 合用时，p将直接作用于w, h (乘以p%) 得到新的w,h，如100w\_100h\_200p 的作

用跟200w\_200h 的效果是一样的。

- 如果对图片进行倍数放大，单边的最大长度不能超过4096 \* 4。

### 示例

- 将图按比例放大两倍。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@200p>



- 将图按比例缩略到原来的1/2

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@50p>



## 缩略后填充

### 缩略后填充

先把图按短边优先缩略，然后再用指定颜色填充剩余区域

#### 参数

名称	描述	取值范围
w	指定目标缩略图的宽度	1-4096
h	指定目标缩略图的高度。	1-4096
e	缩略优先边，这里必须指定值为4	4
bgc	指定填充的背景颜色。默认不指定，为白色填充。参数格式： : <red>-<green>-<blue>bgc 如：100-100-100bgc red,green,blue指定一个颜色	Red, green, blue[0-255]

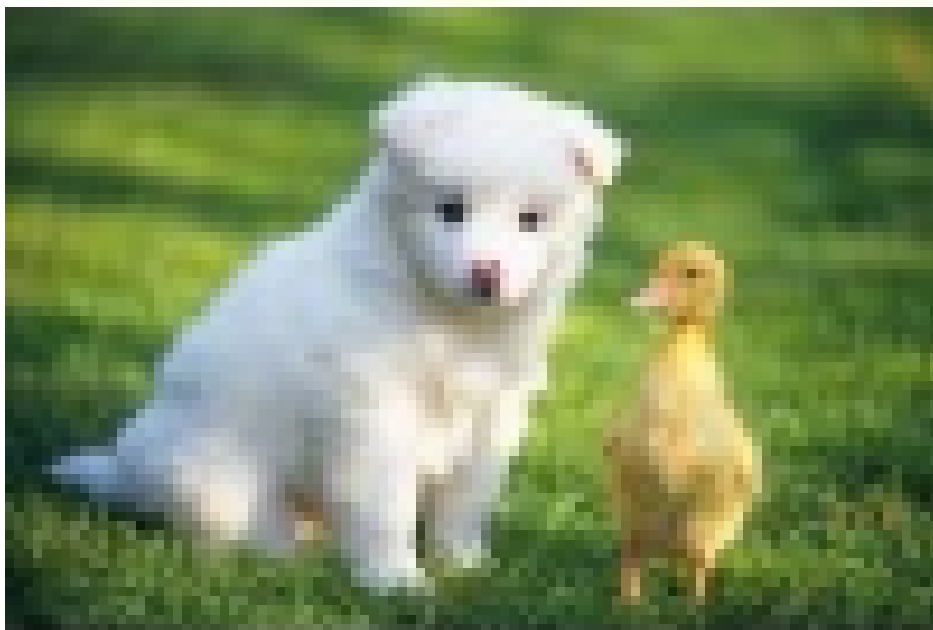
#### 注意事项

- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。
- bgc 由红绿蓝三原色这三个参数指定生成对应颜色。

#### 示例

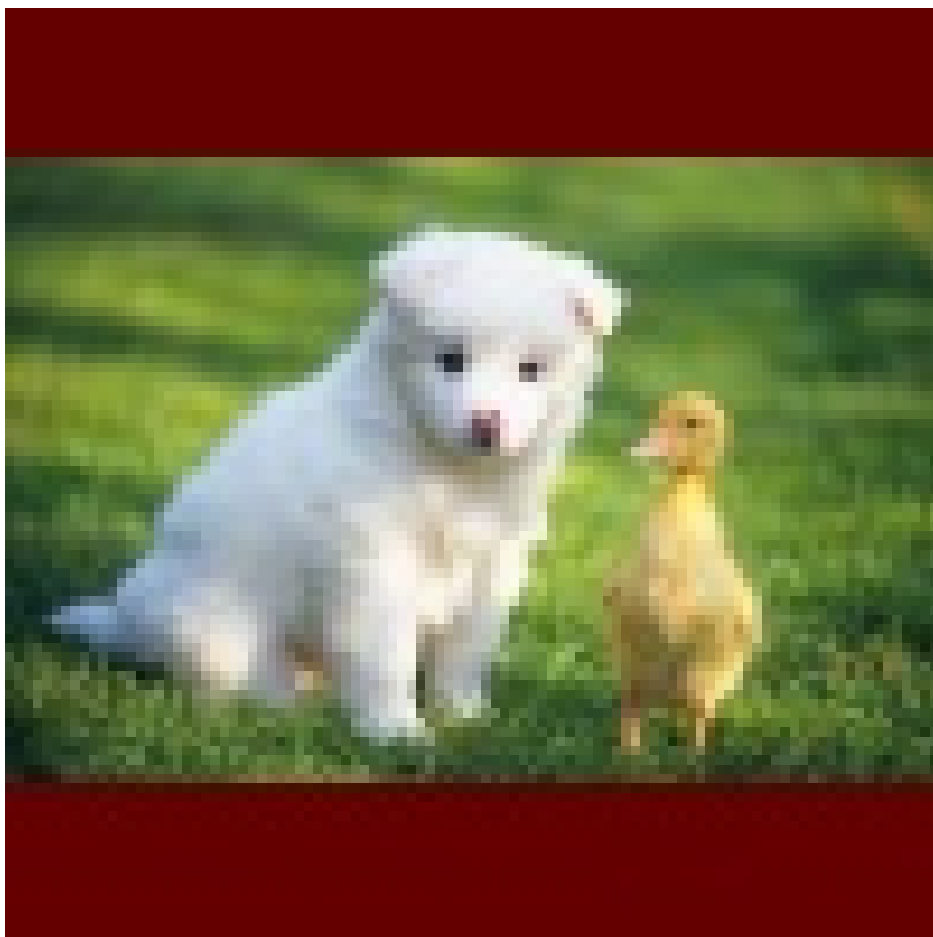
- 将图按短边缩略到100x100, 然后按白色填充

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_4e](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_4e)



- 将图按短边缩略到100x100, 然后按红色填充

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_4e\\_100-0-0bgc](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_4e_100-0-0bgc)



## 图片裁剪

### 自动裁剪

#### 自动裁剪

自动裁剪表示图片先按短边缩略，然后从缩略的目标图片裁剪出中间部分得到对应指定高度和宽度的目标缩略图。

#### 参数

名称	描述	取值范围
w	指定目标缩略图的宽度	1-4096
h	指定目标缩略图的高度。	1-4096
e	缩放优先边，这里指定按短边优化	1
c	是否进行裁剪。如果是想对图进行自动裁剪，必须指定为1	0, 1
l	如果目标缩略图大于原图是否处理，值是1, 即不处理，是0，表	0/1, 默认是0

	示处理	
--	-----	--

### 注意事项

- 自动裁剪从按短边优先缩略的图中间进行裁剪，如果想从裁剪出图的左边部分或者右边部分。即不指定裁剪参数C, 然后再利用管道实现
- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。

### 示例

- 将图自动裁剪成宽度为100，高度为100的效果图

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w\\_1e\\_1c](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w_1e_1c)



- 将图片按短边裁剪然后，裁剪出左半部分。

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w\\_1e|0-0-100-100a](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w_1e|0-0-100-100a)



## 区域裁剪

### 区域裁剪

可以将图片分成多个区域，按照区域进行裁剪。

#### 参数

名称	描述	取值范围
rc	<p>用户可以指定对某一个区域进行裁剪。在这里把图片分成 9 个区域。</p> <p>参数格式：&lt;width&gt;x&lt;height&gt;-&lt;pos&gt;rc.jpg</p> <p>width 指的是裁剪的宽度[0, 4096]</p> <p>Height 指的是裁剪的高度[0, 4096]</p> <p>pos指的是裁剪区域,取值范围是[1,9],默认是左上角,区域数值对应表见下图</p> <p>如果想裁剪左上角，宽度是100,高度是200的区域，参数是：100x200-1rc</p> <p>如果想裁剪左上角，宽度是</p>	width, height的范围是[1,4096]

	100,高度是图片的原高度 参数是：100x0-1rc 或者100x-1rc 如果高度或者宽度不填，或者参数是0，或者参数大于原图。默认是按原图的高度或宽度返回。	
--	---	--

区域数值对应表

1 左上	2 中上	3 右上
4 左中	5 中部	6 右中
7 左下	8 中下	9 右下

#### 注意事项

- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。
- 如果从起点开始指定的宽度和高度超过了原图，将会直接裁剪到原图结尾。

#### 示例

- 裁剪原图左上区域宽度100高度200的区域

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100x200-1rc>



## 内切圆

### 内切圆

用户可以将图片只保存圆形图案，如果图片的最终格式是png, webp, bmp等支持透明通道的图片，那么图片非圆形区域的地方将会以透明填充。如果图片的最终格式是jpg。那么非圆形区域是以白色进行填充。

#### 参数

参数	描述	取值
ci	从图片取出圆形区域 参数格式：[radius]-[type]ci	radius : [1, 4096] 如果radius能指定圆的半径。 但是圆的半径不能超过原图的最小边的一半。如果半径超过。 圆的大小仍然是原图的最大内切圆。 type: [0, 1] 0:表示图片最终大小仍然是原图大小 1: 表示图片最终大小是能包含这个圆的最小正方形

#### 注意事项

- 如果图片的最终格式是png, webp, bmp等支持透明通道的图片，那么图片非圆形区域的地方将会以透明填充。如果图片的最终格式是jpg。那么非圆形区域是以白色进行填充。推荐使用保存成png格式
- 指定半径大于原图最大内切圆的半径。圆的大小仍然是图片的最大内切圆。

#### 示例

- 裁剪半径是100, 保持圆是原来大小

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100-0ci>



- 裁剪半径是100, 保存圆是能包含圆的最小正方形, 格式是png

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100-1ci.png>



- 裁剪半径是1000, 保存圆是能包含圆的最小正方形, 格式是png

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@1000-1ci.png>



## 圆角矩形

### 圆角矩形

可以保存图片是圆角矩形，并可以指定圆角的大小。

#### 参数

参数	描述	取值
2ci	从图片取出圆形区域 参数格式：[radius]-2ci	radius：[1, 4096] radius指定圆角的半径。 但是生成的最大圆角的的半径不能超过原图的最小边的一半。

#### 注意事项

- 如果图片的最终格式是png, webp, bmp等支持透明通道的图片，那么图片非圆形区域的地方将会以透明填充。如果图片的最终格式是jpg. 那么非圆形区域是以白色进行填充。推荐使用保存成png格式

- 指定半径大于原图最大内切圆的半径。圆角的大小仍然是图片的最大内切圆。

#### 示例

- 裁剪圆角半径是30, 格式是jpg

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@30-2ci>



- 图片先自动裁剪成100x100, 然后保存成圆角半径是10,格式是png

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_1e\\_1c\\_10-2ci.png](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_1e_1c_10-2ci.png)



## 索引切割

### 索引切割

将图片分成x，y轴，按指定长度(length)切割，指定索引(index)，取出指定的区域。

#### 参数

参数	描述	取值
ic	参数格式：[length]x-[index]ic 或[length]y-[index]ic length是切割长度 index 是表示块数。（0表示第一块） 其中x表示按x轴，水平线切割。 y 表示按y轴，垂直线切割	length:[1,切割边边长]，单位px。如果超出切割边的大小，返回原图。 index:[0,最大块数)。如果超出最大块数，返回原图。

#### 注意事项

- 如果指定的索引大于切割后范围，将返回原图。

#### 示例

对图片x轴按100平分，取出第一块。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100x-0ic>



对图片y轴按100平分，取出第一块。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100y-0ic>



对图片x轴按100平分，取出第一百块，仍然是原图

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100x-100ic>



## 高级裁剪

### 高级裁剪

可以通过指定起始横坐标，纵坐标及裁剪的宽度和裁剪的高度对图进行高级裁剪。

#### 参数

名称	描述	取值范围
a	参数的类型：x-y-width-length 如：100-50-200-150a 一共四个参数，每个参数之间以“-”隔开。第一个参数表示起始点x坐标（以左上角为原点），第二个参数表示起始点y坐标，	width, height的范围是1-4096

	<p>第三个参数表示要裁剪的宽度，第四个参数表示要裁剪的高度。如100-50-200-150a 表示从点(100, 50) 裁剪大小为(200, 150)的图片。</p> <p>注意：可以将第三个参数，第四个参数置为0, 表示裁剪到图片的边缘。如100-50-0-0a 表示从点(100, 50) 裁剪到图片的最后</p>	
--	---	--

## 注意事项

- 如果不指定格式，原图将默认转换成jpg格式，如果原图是png, webp, bmp可能会导致图出现变形。详细可以查看质量变换及格式转换。
- 如果指定的起始纵横坐标大于原图，将会返回错误：BadRequest, 错误内容是：Advance cut's position is out of image.
- 如果从起点开始指定的宽度和高度超过了原图，将会直接裁剪到原图结尾。

## 使用示例

- 裁剪图从起点(100, 50)到图的结束

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100-50-0-0a>



- 裁剪图从起点(100, 50)到裁剪100x100的大小

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100-50-100-100a>



## 图片旋转

### 旋转

#### 旋转

可以对处理后的图片进行按顺时针旋转

#### 参数

名称	描述	取值范围
r	默认值：0(表示不旋转)	[0, 360]

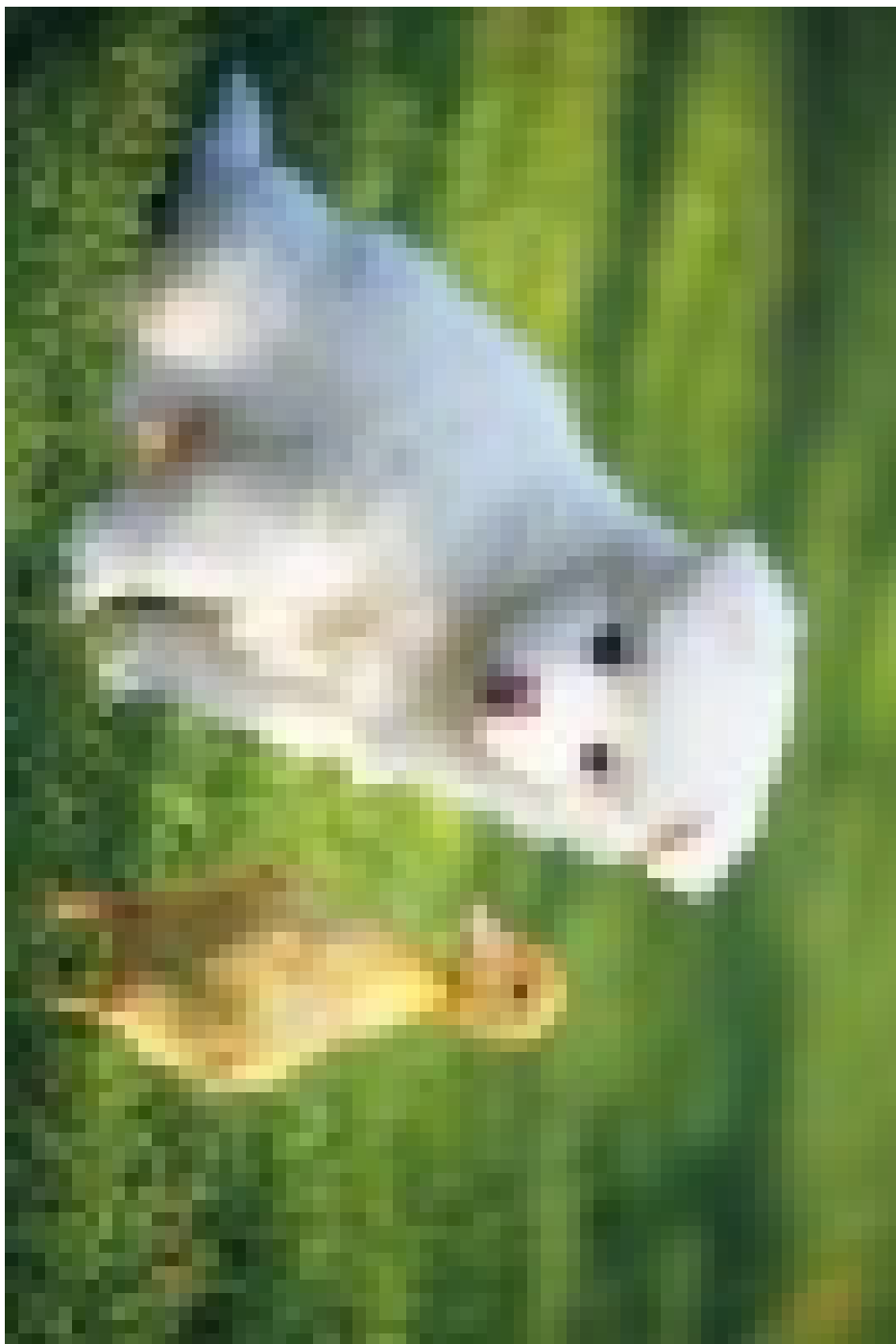
#### 注意事项

- 旋转后的图可能会导致图的尺寸变大。
- 旋转对图的尺寸有限制，图片的宽或者高不能超过4096。

#### 示例

- 将原图缩略成宽度为100,高度为100,按顺时针旋转90度

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_90r](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_90r)



- 将原图按顺时针旋转270度

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@270r>



## 自适应方向

### 自适应方向

某些手机拍摄出来的照片可能带有旋转参数（存放在照片exif信息里面）

#### 参数

名称	描述	取值范围
o	进行自动旋转	[0, 2]

	<p>0：表示按原图默认方向，不进行自动旋转</p> <p>1：表示根据图片的旋转参数，对图片进行自动旋转，如果存在缩略参数，是先进行缩略，再进行旋转。</p> <p>2：表示根据图片的旋转参数，对图片进行自动旋转，如果存在缩略参数，先进行旋转，再进行缩略</p>	
--	--	--

## 注意事项

- 如果采用缩略旋转1,可能会导致图片最终的宽度和高度跟指定的参数不符。
- 进行自适应方向旋转，必须要求原图的宽度和高度必须小于4096。
- 如果原图是没有旋转参数，加上1o, 2o参数不会对图有影响。

## 示例

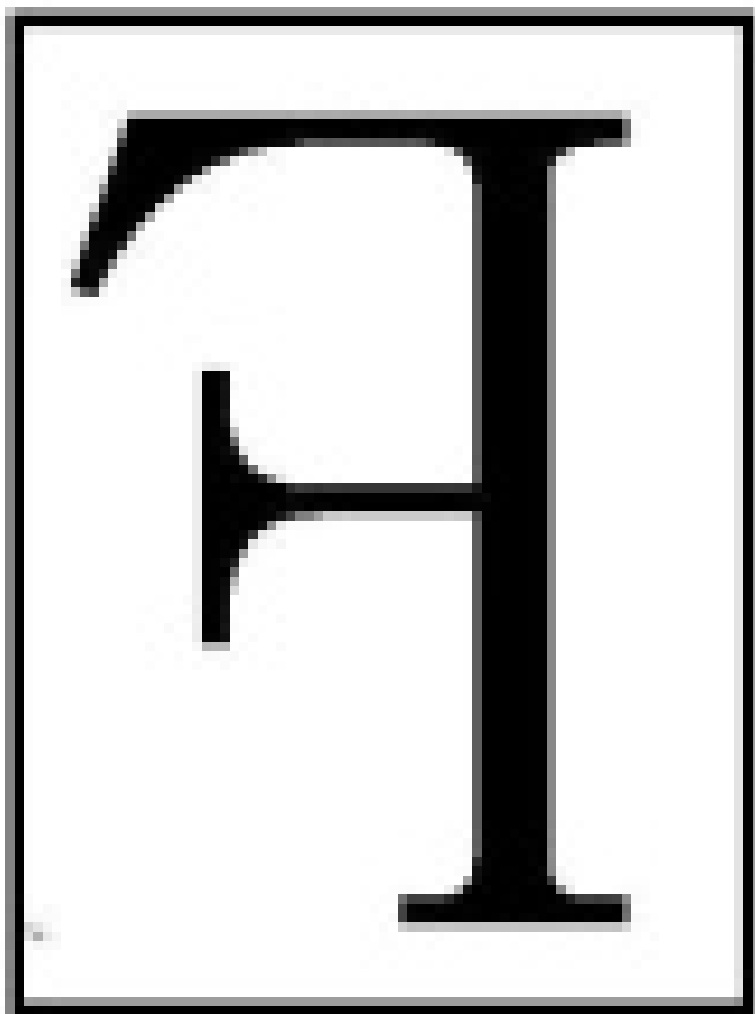
- 将图缩略成宽度为100，对图片不做自动旋转处理

<http://image-demo.img-cn-hangzhou.aliyuncs.com/f.jpg@100w.jpg>



- 将图缩略成宽度为100，对图片进行自动旋转1

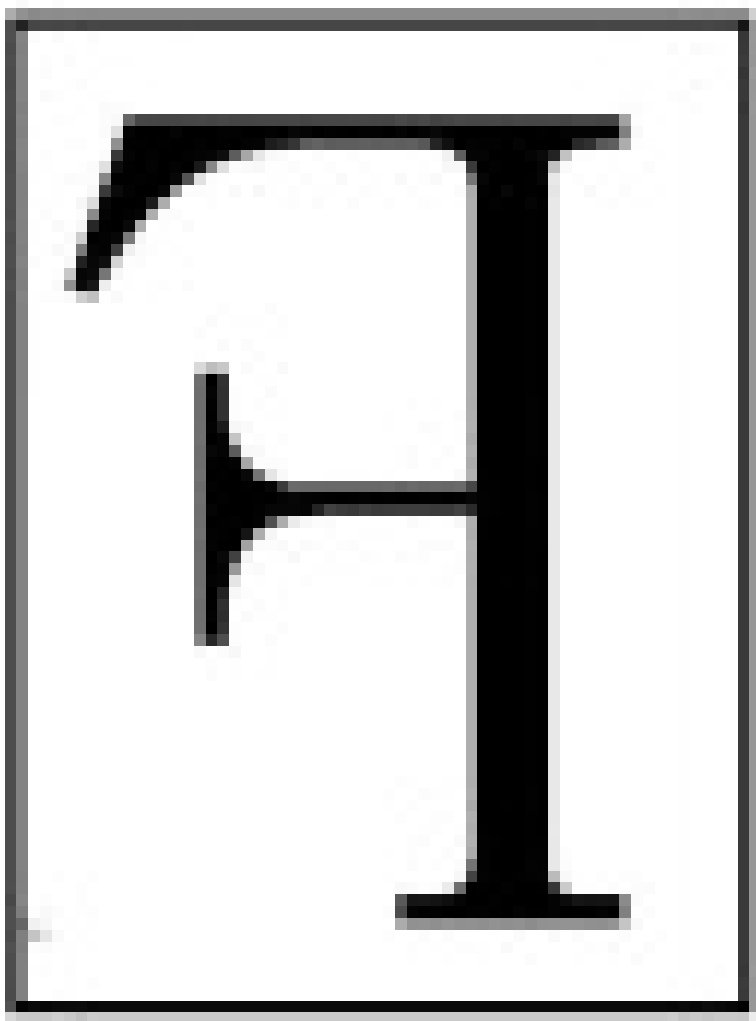
[http://image-demo.img-cn-hangzhou.aliyuncs.com/f.jpg@100w\\_2o.jpg](http://image-demo.img-cn-hangzhou.aliyuncs.com/f.jpg@100w_2o.jpg)



得到的目标效果图宽度是100，高度是127。

- 将图缩略成宽度为100，对图片进行自动旋转2

[http://image-demo.img-cn-hangzhou.aliyuncs.com/f.jpg@100w\\_1o.jpg](http://image-demo.img-cn-hangzhou.aliyuncs.com/f.jpg@100w_1o.jpg)



得到的目标效果图宽度是79, 高度是100。

## 图片效果

### 锐化

#### 锐化

可以对处理后的图片进行锐化处理，使图片变得清晰。

#### 参数

名称	描述	取值范围
sh	表示进行锐化处理。取值为锐化参数，参数越大，越清晰。	[50, 399] 为达到较优效果，推荐取值为

		100
--	--	-----

## 示例

- 对原图进行锐化处理，锐化参数为100。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100sh>



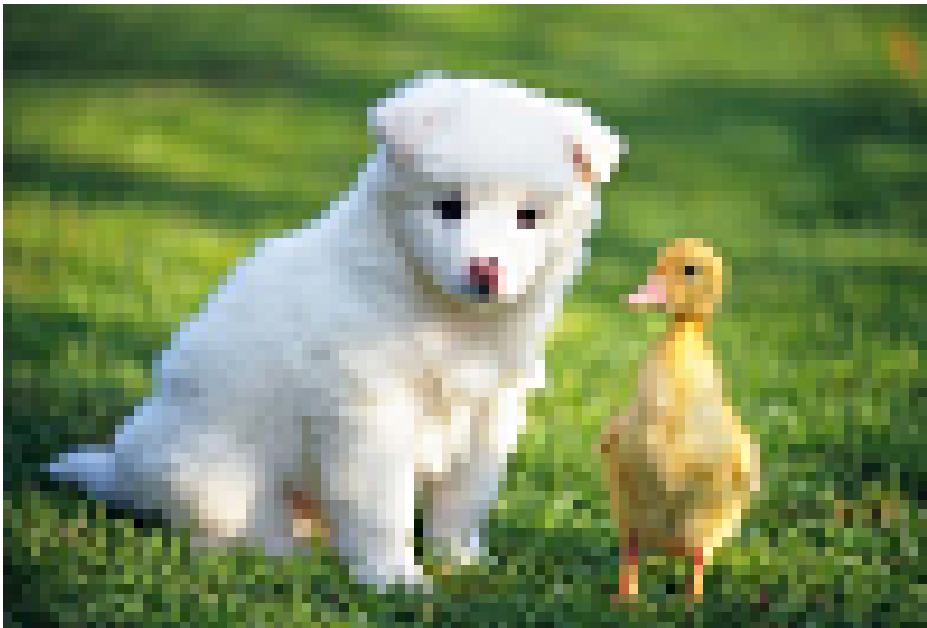
- 对原图进行锐化处理，锐化参数为50。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@50sh>



- 将原图缩略成高度100, 宽度100, 并进行锐化操作, 保存成png格式

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_100sh.png](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_100sh.png)



## 模糊效果

### 模糊效果

可以对图片进行模糊操作。

#### 参数

参数	描述	取值
bl	参数格式：[radius]-[sigma]bl radius是模糊半径 sigma是正态分布的标准差 如：3-2bl 模糊半径是3,标准差是2	radius取值在 [1,50] , radius越大, 越模糊 sigma取值 [1,50] , 越大, 越模糊

#### 示例

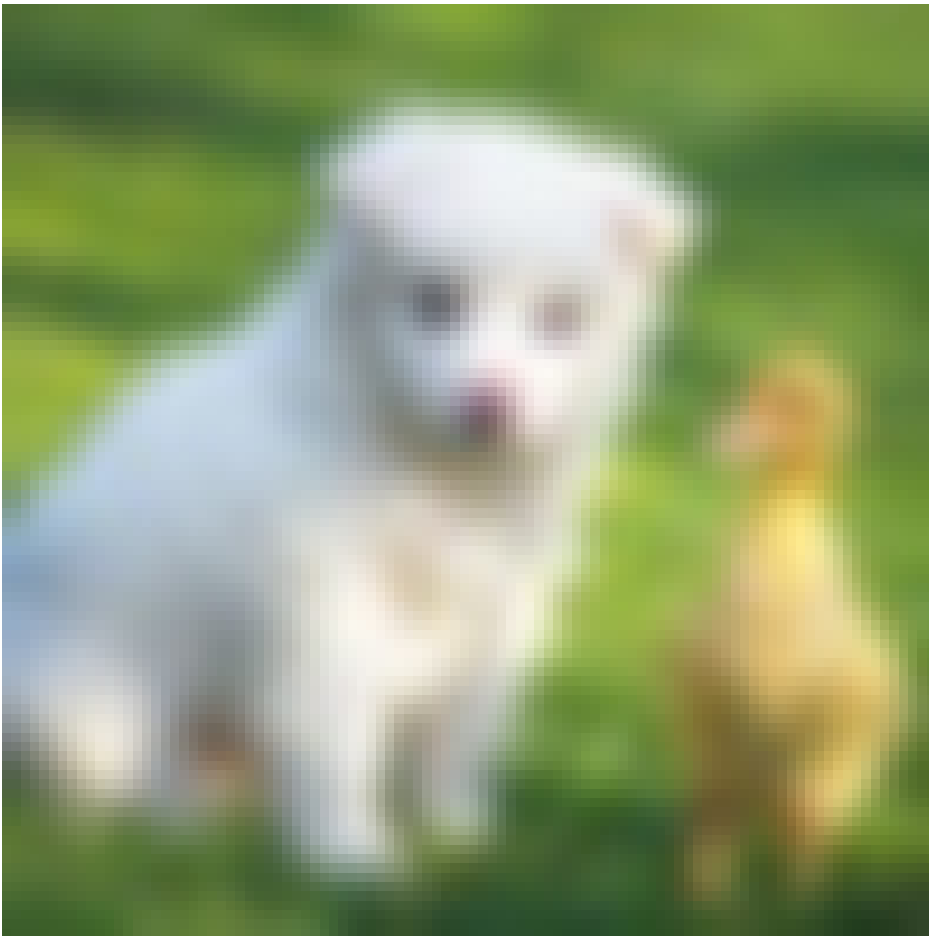
- 对图片进行模糊半径是3, 标准差是2。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@3-2bl>



- 图片先自动裁剪成100x100, 然后对图片进行模糊半径是3, 标准差是2。

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_1e\\_1c\\_3-2bl](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_1e_1c_3-2bl)



## 亮度和对比度

### 亮度和对比度

可以对处理后的图片进行亮度和对比度调节

#### 参数

名称	描述	取值范围
b	亮度调整 0表示原图亮度，小于0表示亮度越低，大于0表示亮度越高	[-100, 100]
d	对比度调整 0表示原图对比度，小于0表示对比越低，大于0表示对比越高	[-100, 100]

#### 示例

- 将原图只进行亮度调整

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@50b>



- 将原图只进行对比度调整

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@-50d>



- 将原图进行亮度和对比度调整

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@-50d\\_50b](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@-50d_50b)



## 图片水印

### 概述

### 概述

水印操作可以在图片上设置另外一张图片或者文字做为水印。

水印类型分成图片水印，文字水印，和文图混合水印。

## 基本参数

### 水印基本参数

图片水印，文字水印，和文图混合水印都可以使用如下参数。

名称	描述	参数类型
t	参数意义：透明度, 如果是图片水印，就是让图片变得透明，如果是文字水印，就是让水印变透明。 默认值：100，表示 100%（不透明） 取值范围: [0-100]	可选参数
p	参数意义：位置，水印打在图的位置，位置如成如下图。 默认值：9，表示在右下角打水 印 取值范围：[1-9]	可选参数
x	参数意义：水平边距, 就是距离图片边缘的水平距离，这个参数只有当水印位置是左上，左中，左下，右上，右中，右下才有意义 默认值：10 取值范围：[0 – 4096] 单位：像素（px）	可选参数
y	参数意义：垂直边距, 就是距离图片边缘的垂直距离，这个参数只有当水印位置是左上，中上，右上，左下，中下，右下才有意义 默认值：10 取值范围：[0 – 4096] 单位：像素(px)	可选参数
voffset	参数意义：中线垂直偏移，当水印位置在左中，中部，右中时，可以指定水印位置根据中线往上或者往下偏移。 默认值：0 取值范围：[-1000, 1000] 单位：像素(px)	可选参数
watermark	参数意义：选择水印的类型	取值范围 1, 2, 3 1：图片水印 2：文字水印 3：文图混合水印

### 区域数值对应表

1 左上	2 中上	3 右上
4 左中	5 中部	6 右中
7 左下	8 中下	9 右下

#### 注意事项

- 水平边距、垂直边距、中线垂直偏移可以调节不仅可以调节水印在图片中的位置，而且当图片存在多重水印时，也可以调节两张水印在图中的布局。
- 用到的URL安全的Base64位编码可以参考关键词。

#### 示例

- 右下角打上文字水印

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ>



- 右下角打上文字水印，水平边距是 10，垂直边距20

[http://image-demo.img-cn-](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=90&p=9&x=10&y=20)

[hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=90&p=9&x=10&y=20](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=90&p=9&x=10&y=20)



- 右中部分打上水印，水平边距为10，垂直中线偏移为20,透明度为50

[http://image-demo.img-cn-](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=50&p=6&x=10&yoffset=20)

[hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=50&p=6&x=10&yoffset=20](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&t=50&p=6&x=10&yoffset=20)



## 注意

文字水印参数请参考文字水印

## 图片水印

### 图片水印

图片水印就是在原图的基础上加上一张水印图片

### 访问类型

```
@watermark=1&object<encodedobject>&t<transparency>&x<distanceX>&y<distanceY>&p<position>...
```

其中watermark与object两个参数为必填项。文中出现的url\_safe\_base64\_encode指的是URL安全base64编码，请参见关键词。

### 参数

名称	描述	参数类型
object	参数意义：水印图片的object名字(必须编码) 注意：内容必须是URL安全base64编码 encodedObject = url_safe_base64_encode(object) 如object为"panda.png", 编码过后的内容为 "cGFuZGEucG5n"	必选参数

水印位置参数请参考基本参数

### 水印图片预处理

用户在打水印时，可以对水印图片进行预处理，支持的预处理操作有：图片缩放，图片裁剪（不支持内切圆），图片旋转（具体内容请直接查看文档相关章节），但不支持管道操作。还额外支持一个参数：P（大写P），表示水印图片按主图的比例进行处理，取值范围为[1, 100]，表示百分比

## 预处理示例

设置了10P, 当主图是100x100, 水印图片大小就为10x10, 当主图变成了200x200，水印图片大小就为20x20。如果生成的图片大小不一样，而使用相同的水印处理参数，就会导致一些小图，水印图片过大。或者一些大图，水印图片过小。增加P参数，就可以解决这个问题。采用P参数，IMG会根据主图的大小来动态调整水印图片的大小。

如果水印操作是：watermark=1&object=cGFuZGEucG5nQDMwUA&t=90&p=9&x=10&y=10（右下角打水印。水印图片是：panda.png@20P, 表示水印的大小按主图的20%缩放。）

如果原图按宽度是400，需要缩略，再打上上述水印的示例：

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@400w|watermark=1&object=cGFuZGEucG5nQDMwUA&t=90&p=9&x=10&y=10>



如果原图按宽度300缩略，再打上上述水印的示例：

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@300w|watermark=1&object=cGFuZGEucG5nQDMwUA&t=90&p=9&x=10&y=10>



访问形式：

参数中的object规则是：

- 1 图片水印原图名字（ 未经过URL安全base64编码的名字 ） + @ + Action
- 2 对第一步的字符串进行URL安全base64编码

如果要指定对水印图片进行预处理，处理参数带在水印Object之后，以@符号连接。如：

- 对panda.png 不进行任何预处理: object = url\_safe\_base64\_encode("panda.png")
- 对panda.png 进行放大2倍: object = url\_safe\_base64\_encode("panda.png@200p")
- 对panda.png 进行缩小一倍，亮度调节成50, 对比度调节成40: object = url\_safe\_base64\_encode("panda.png@50p\_50b\_40d")
- 对panda.png 增加按宽度50缩略，亮度调节成30: object = url\_safe\_base64\_encode("panda.png@50w\_30b")
- 对panda.png 增加按高度20缩略，对比度调节成10: object = url\_safe\_base64\_encode("panda.png@20h\_10d")
- 对panda.png 水印图的大小基于原图的20%进行处理，对比度调节成10: object = url\_safe\_base64\_encode("panda.png@20P\_10d")

示例

- 下面URL的含义，是example.jpg加上水印文件是 :panda.png (panda.png 经过URL安全base64编码后是：cGFuZGEucG5n)

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5n&t=90&p=9&x=10&y=10>



- 对panda.png按宽度是50缩放。那么水印文件是：panda.png@50w（panda.png@50w 经过URL安全base64编码后是:cGFuZGEucG5nQDUwdw）

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5nQDUwdw&t=90&p=9&x=10&y=10>



- 对panda.png按50%的比例缩小。那么水印文件是：panda.png@50p（panda.png@50p经过URL安全base64编码后是：cGFuZGEucG5nQDUwcA）

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5nQDUwcA&t=90&p=9&x=10&y=10>



- 对panda.png，自动裁剪成宽度是40，高度是30大小。那么水印文件是  
： panda.png@40w\_20h\_1e\_1c(panda.png@40w\_20h\_1e\_1c经过URL安全base64编码后是  
： cGFuZGEucG5nQDQwd18yMGhfMWVfMWM )

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5nQDQwd18yMGhfMWVfMWM&t=90&p=9&x=10&y=10>



- 对panda.png. 高级裁剪，从起点(0, 0) 裁剪到(65, 65)的位置。那么水印文件是:panda.png@0-0-65-65a (panda.png@0-0-65-65a经过URL安全base64编码后是  
cGFuZGEucG5nQDAtMC02NS02NWE )

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5nQDAtMC02NS02NWE&t>

=90&p=9&x=10&y=10



## 文字水印

### 文字水印

文字水印就是在原图的基础上加上一段文字内容做为水印。

#### 访问类型

```
@watermark=2&text<encodeText>&type<encodeType>&size<size>&color<encode color>&t<t>&p<p>&x<x>&y<y>&voffset<offset>&y<y>
```

其中watermark与object两个参数为必填项。文中出现的url\_safe\_base64\_encode指的是URL安全base64编码，请参见关键词。

#### 参数

名称	描述	参数类型
text	参数意义：表示文字水印的文字内容(必须编码) 注意：必须是URL安全base64编码 encodeText = url_safe_base64_encode(font Text) 最大长度为64个字符(即支持汉字最多20个左右)	必选参数
type	参数意义：表示文字水印的文字类型(必须编码) 注意：必须是URL安全base64编码 encodeText = url_safe_base64_encode(font Type) 取值范围：见下表（文字类型编	可选参数

	码对应表) 默认值: wqy-zenhei (编码后的值: d3F5LXplbmhlaQ)	
color	参数意义: 文字水印文字的颜色 (必须编码) 注意: 参数必须是URL安全base64编码 EncodeFontColor = url_safe_base64_encode(font Color) 参数的构成必须是: ##### + 六个十六进制数 如: #000000表示黑色。#是表示前缀, 000000每两位构成RGB颜色, #FFFFFF表示的是白色 默认值: #000000黑色 base64编码后值: IzAwMDAwMA	可选参数
size	参数意义: 文字水印文字大小 (px) 取值范围: (0, 1000] 默认值: 40	可选参数
s	参数意义: 文字水印的阴影透明度 取值范围: (0,100]	可选参数

水印位置参数请参考基本参数

#### 文字类型编码对应表

参数值	中文意思	URL安全base64编码后的值	备注
wqy-zenhei	文泉驿正黑	d3F5LXplbmhlaQ==	根据RFC, 可省略填充符=变为 d3F5LXplbmhlaQ
wqy-microhei	文泉微米黑	d3F5LW1pY3JvaGVp	
fangzhengshusong	方正书宋	ZmFuZ3poZW5nc2h1c29uZW==	根据RFC, 可省略填充符=变为 ZmFuZ3poZW5nc2h1c29uZW
fangzhengkaiti	方正楷体	ZmFuZ3poZW5na2FpdGk=	根据RFC, 可省略填充符=变为 ZmFuZ3poZW5na2FpdGk
fangzhengheiti	方正黑体	ZmFuZ3poZW5naGVpdGk=	根据RFC, 可省略填充符=变为 ZmFuZ3poZW5naGVpdGk
fangzhengfangsong	方正仿宋	ZmFuZ3poZW5nZmFuZ3Nvbmc=	根据RFC, 可省略填充符=变为 ZmFuZ3poZW5nZm

			FuZ3Nvbmc
droidsansfallback	DroidSansFallback	ZHJvaWRzYW5zZmFsbGJhY2s=	根据RFC，可省略填充符=变为 ZHJvaWRzYW5zZmFsbGJhY2s

## 示例

- 字体是文泉驿正黑，字体大小是40, 颜色是白色(#FFFFFF), 文字阴影是50, 文字水印内容是：Hello, 图片服务！，水印位置是：右中，水平边距是：10，中线垂直偏移是：20

[http://image-demo.img-cn-](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&s=50&t=90&p=6&x=10&voffset=20)

[hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&s=50&t=90&p=6&x=10&voffset=20](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&s=50&t=90&p=6&x=10&voffset=20)



- 最简单水印：文字内容是：Hello, 图片服务

[http://image-demo.img-cn-](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ)

[hangzhou.aliyuncs.com/example.jpg@watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ)



## 文图混合水印

### 文图混合水印

文图混合水印就是文字，图片并列一起做为水印打在图片上。

#### 访问类型

```
@watermark=3&object<encodeObject>&text<encodeText>&type<encodeType>&size<size>&color<encodecolor>&order<order>&align<align>&interval=<interval>&t<t>&p<p>&x<x>&y<y>
```

其中watermark与object两个参数为必填项。文中出现的url\_safe\_base64\_encode指的是URL安全base64编码，请参见关键词。

#### 参数

文图混合水印，相当于文字水印跟图片水印的混合，并行在一行输出。所以文图混合水印支持文字水印和图片水印的参数。其中object, text是必选参数。

名称	描述	参数类型
order	参数意义：文字，图片水印前后顺序 取值范围：[0, 1] order = 0 图片在前(默认值)； order = 1 文字在前。	可选参数
align	参数意义：文字、图片对齐方式 取值范围：[0, 1, 2] align = 0 上对齐(默认值) align = 1 中对齐 align = 2 下对齐	可选参数
interval	参数意义：文字和图片间的间距 取值范围: [0, 1000]	可选参数

水印位置参数请参考基本参数

## 示例

- 单纯文字水印，文字内容是：Hello, 图片服务! 阴影是50, 位置在右下角，水平边距和垂直边距都是10, 水印透明是:90

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&s=50&t=90&p=9&x=10&y=10>



- 单纯图片水印，图片object 是panda.png，位置在右下角，水平边距和垂直边距都是10, 水印透明是:90

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=1&object=cGFuZGEucG5n&t=90&p=9&x=10&y=10>



- 文图混合水印，文字内容是：Hello, 图片服务! 阴影是50, 位置在右下角，图片object 是 panda.png。水平边距和垂直边距都是10, 水印透明是:90，排版方式是图片前，对齐方式是对齐，间距是10

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=3&object=cGFuZGEucG5n&type=d3F5LXplbmhlaQ&size=40&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=I0ZGRkZGRg&order=0&align=1&interval=10&t=90&p=9&x=10&y=10>



## 格式转换

## 质量变换

### 质量变换

如果图片保存成jpg, 可以支持质量变换。

#### 参数

名称	描述	取值范围
q	决定jpg图片的相对质量，对原图按照q%进行质量压缩。如果原图质量是100%，使用"90q"会得到质量90%的图片；如果原图质量是80%，使用"90q"会得到质量72%的图片。只能在原图是jpg格式的图片上使用，才有相对压缩的概念。如果保存图片是png，那么相对质量就相当于绝对质量。	1-100
Q	决定jpg图片的绝对质量，把原图质量压到Q%，如果原图质量小于指定数字，则不压缩。如果原图质量是100%，使用"90Q"会得到质量90%的图片；如果原图质量是95%，使用"90Q"还会得到质量90%的图片；如果原图质量是80%，使用"90Q"不会压缩，返回质量80%的原图。只能在jpg/png效果上使用，其他格式无效果。如果一个转换url里，同时指定了q和Q，按Q来处理	1-100

#### 注意事项

- 如果不填Q或者q 这两个参数，这样有可能会导导致图片占用空间变大。如明确想得到一个质量固定的图片，请采用Q参数。如果想按原图质量来保存，指定成100q

#### 示例

- 将原图缩略成100w\_100h，相对原图质量的80%的jpg图

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_80q](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_80q)



- 将原图缩略成100w\_100h,绝对质量的80的jpg图

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_100h\\_80Q](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_100h_80Q)



## 格式转换

### 格式转换

可以将图片转换成对应格式(jpg, png, bmp, webp)

#### 参数

名称	描述
----	----

jpg	将原图保存成jpg格式，如果原图是png,webp, bmp存在透明通道，默认会把透明填充成黑色。如果想把透明填充成白色可以指定1wh参数
png	将原图保存成png格式
webp	将原图保存成webp格式
bmp	将原图保存成bmp格式
src	按原图格式返回，如果原图是gif, 此时返回gif格式第一帧,保存成jpg格式，而非gif格式

## 注意事项

- wh只有当原图是四通道(即有透明背景)的png, webp, bmp转换成jpg格式时才有效果。即把原图当中的透明背景以白色填充，如果不指定wh, 那么上述图转换成jpg时，透明背景将会变成黑色。

保存成jpg格式时，默认是保存成标准型的jpg(Baseline JPEG) 如果想指定是渐进式JPEG(Progressive JPEG), 可以指定参数1pr, 详见渐进显示。

## 示例

将png保存成jpg格式

<http://image-demo.img-cn-hangzhou.aliyuncs.com/panda.png@.jpg>



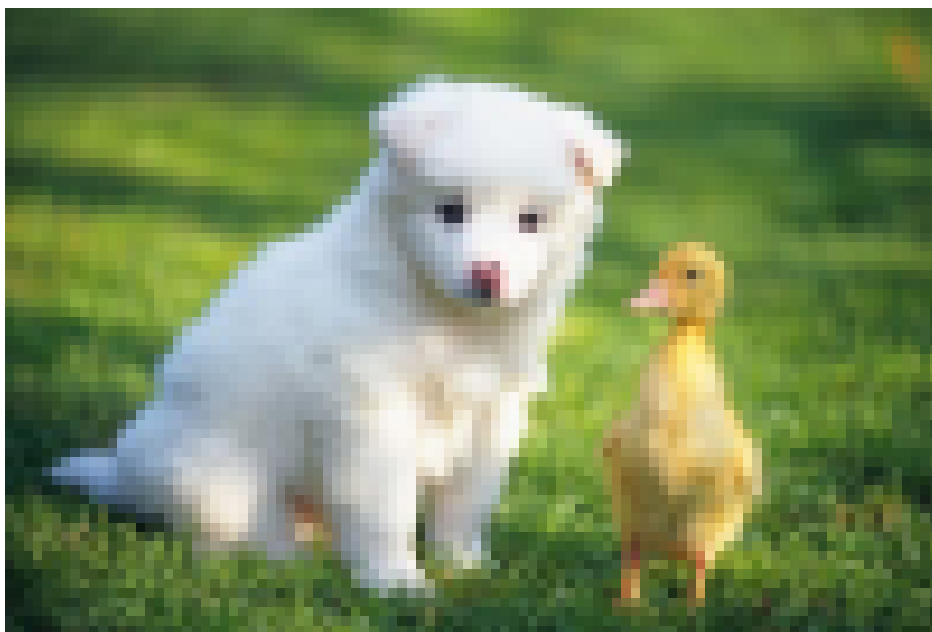
- 将png保存成jpg格式，透明的地方填充成白色

<http://image-demo.img-cn-hangzhou.aliyuncs.com/panda.png@1wh.jpg>



- 将jpg保存成高度为100, 宽度为100的png格式

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h\\_100w.png](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100h_100w.png)



## 渐进显示

### 渐进显示

jpg格式在呈现时有两种方式，一种是自上而下扫描式的，另外一种是先模糊后渐渐清晰（在网络环境比较差时明显）默认保存是第一种，如果指定呈现时是先模糊后清晰的，请使用渐进显示参数。

#### 参数

名称	描述	取值范围
pr	1表示保存成渐进显示的jpg格式 0表示保存成普通的jpg格式	[0, 1]

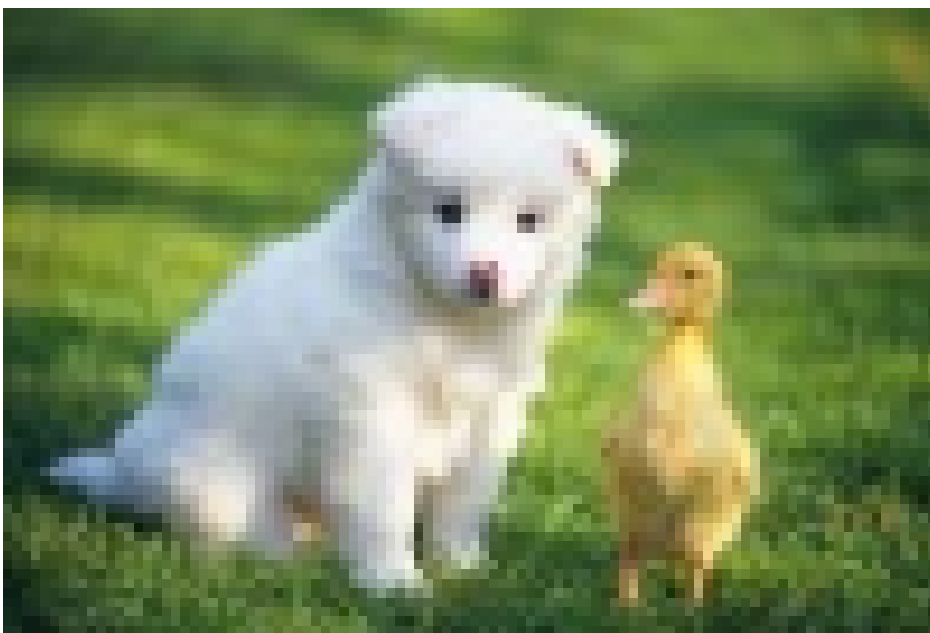
#### 注意事项

- 此参数只有效果图是jpg格式时，才有意义。

#### 示例

- 将图缩略成宽度100，高度100，并且保存成渐进显示的jpg格式

[http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w\\_200h\\_1pr.jpg](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@100w_200h_1pr.jpg)



- 将png图保存成渐进显示的jpg格式

<http://image-demo.img-cn-hangzhou.aliyuncs.com/panda.png@1pr>



## 获取图片信息

### 获取基本信息

#### 获取文件基本信息

获取图片基本信息，是获取图片的宽度、高度和大小 <http://image-demo.img.aliyuncs.com/example.jpg@info>

#### 请求格式

请求URL：@info

#### 返回格式

```
{
  "height": 267,
  "size": 21839,
  "width": 400}
```

- height 表示图片的高度
- size 表示图片的大小

- width 表示图片的宽度

#### 示例

<http://image-demo.img.aliyuncs.com/f.jpg@exif>

```
{
  "height": 267,
  "size": 21839,
  "width": 400}
```

## 获取exif信息

### 获取文件exif信息

exif 是专门为数码相机的照片设定的，可以记录数码照片的属性信息和拍摄数据。 可以通过@exif 来获取，返回格式是Json格式，目前支持返回

- GPSTatitudeRef
- GPSTatitude
- GPSTongitudeRef
- GPSTongitude
- DateTime
- DateTimeOriginal
- DateTimeDigitized
- Make
- Model
- Orientation

等参数

### 注意事项

并不是每一张图片都有exif信息。如果原图没有exif,请求exif信息时，会返回400错误。错误码是：BadRequest 错误内容是：Image has no exif info.

#### 示例

- 没有exif 的例子

<http://image-demo.img.aliyuncs.com/example.jpg@exif>

```
<Error>
<Code>BadRequest</Code>
<Message>Image has no exif info.</Message>
<RequestId>5502D98553F47BFAB7F95B8C</RequestId>
<HostId>image-demo.img.aliyuncs.com</HostId>
</Error>
```

- 有exif 的例子

<http://image-demo.img.aliyuncs.com/f.jpg@exif> 返回事例

```
{
  "Compression": {"value": "6"},
  "DateTime": {"value": "2015:02:11 15:38:27"},
  "ExifTag": {"value": "2212"},
  "FileSize": {"value": "23471"},
  "GPSLatitude": {"value": "0deg "},
  "GPSLatitudeRef": {"value": "North"},
  "GPSLongitude": {"value": "0deg "},
  "GPSLongitudeRef": {"value": "East"},
  "GPSMapDatum": {"value": "WGS-84"},
  "GPSTag": {"value": "4292"},
  "GPSVersionID": {"value": "2 2 0 0"},
  "ImageHeight": {"value": "333"},
  "ImageWidth": {"value": "424"},
  "JPEGInterchangeFormat": {"value": "4518"},
  "JPEGInterchangeFormatLength": {"value": "3232"},
  "Orientation": {"value": "7"},
  "ResolutionUnit": {"value": "2"},
  "Software": {"value": "Microsoft Windows Photo Viewer 6.1.7600.16385"},
  "XResolution": {"value": "96/1"},
  "YResolution": {"value": "96/1"}}
```

## 获取基本信息和exif信息

### 获取基本信息和exif信息

可以通过@infoexif来获取获取文件的基本信息包括宽度，长度，文件大小，格式。并且如果文件有exif信息，就返回exif信息，如果没有exif信息，就只返回基本信息。返回结果是json格式。

#### 请求格式

请求URL + @infoexif

#### 返回格式

json格式

#### 该接口与exif接口的区别

本接口返回是exif信息的基本上增加 FileSize（文件大小），Format（图片格式），ImageHeight(图片高度)，ImageWidth(图片宽度) 并且当图片没有exif信息时，只返回基本信息，当有图片有exif信息，将基本信息跟exif信息一起返回

#### 示例

- 没有exif 的例子

<http://image-demo.img.aliyuncs.com/example.jpg@infoexif>

```
{
```

```
{
  "FileSize": {"value": "21839"},
  "Format": {"value": "jpg"},
  "ImageHeight": {"value": "267"},
  "ImageWidth": {"value": "400"}
}
```

- 有exif 的例子

<http://image-demo.img.aliyuncs.com/f.jpg@infoexif>

```
{
  "Compression": {"value": "6"},
  "DateTime": {"value": "2015:02:11 15:38:27"},
  "ExifTag": {"value": "2212"},
  "FileSize": {"value": "23471"},
  "Format": {"value": "jpg"},
  "GPSLatitude": {"value": "0deg"},
  "GPSLatitudeRef": {"value": "North"},
  "GPSLongitude": {"value": "0deg"},
  "GPSLongitudeRef": {"value": "East"},
  "GPSMapDatum": {"value": "WGS-84"},
  "GPSTag": {"value": "4292"},
  "GPSVersionID": {"value": "2 2 0 0"},
  "ImageHeight": {"value": "333"},
  "ImageWidth": {"value": "424"},
  "JPEGInterchangeFormat": {"value": "4518"},
  "JPEGInterchangeFormatLength": {"value": "3232"},
  "Orientation": {"value": "7"},
  "ResolutionUnit": {"value": "2"},
  "Software": {"value": "Microsoft Windows Photo Viewer 6.1.7600.16385"},
  "XResolution": {"value": "96/1"},
  "YResolution": {"value": "96/1"}
}
```

## 获取图片主色调

### 获取图片主色调

本接口是为了获取图片的平均色调

#### 请求格式

<image-url>@imageAve

#### 返回格式

0xRRGGBB （RR GG BB都是十六进制表示红 绿 蓝三个颜色）

#### 示例

- 可以在浏览器访问

<http://image-demo.img.aliyuncs.com/example.jpg@imageAve>

得到结果

```
{"RGB": "0x5c783b"}
```

原图是：

- <http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg>



0x5c783b对应的颜色RGB(92,120,59)是：



管道

管道  
管道

管道是一种可以实现多种处理任务顺序执行的机制。用户可以通过管道在一次访问中按照顺序完成对图像的不同处理。

## 访问规则

```
<图片URL>@<action1>|<action2>
```

URL 通过@符号后面处理参数（action1, action2）来实现即时云处理，如果有多任务（比如先做缩略，再加上水印）可以用管道来实现，执行顺序按管道指定顺序执行，目前最多支持四级管道。

## 管道的分隔符是“|”

上述表示先对图片URL做处理action1 然后再在上述的基础上做处理action2，然后输出结果。上述action1，action2可以是简单缩略，文字水印，图片水印任意一种。

## 使用示例

- 先对图片做按高度300缩略，然后再加上文字水印，水印内容是：Hello 图片服务！

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@300h|watermark=2&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ>



这个例子由action1( 300h : 按高度是300缩略) 操作完再执行action2(watermark=2&text=SGVsbG8g5Zu-a54mH5pyN5YqhIQ: 文字水印，水印内容是：Hello 图片服务！)处理时间，先对图片执行Action1操作，再执行Action2操作。

- 先对图片做文字水印，水印内容是: Hello, 图片服务！ 水印位置在右下角，然后再对图片做图片水印，水印object是:panda.png, 水印位置在中间。

<http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@watermark=2&text=SGVsbG8g5Zu->

54mH5pyN5YqhIQ&p=9|watermark=1&object=cGFuZGEucG5n&t=90&p=5



## 样式

### 样式访问

#### 样式概要

因为所有对图片的变换都会加在URL后面，会导致URL变得冗长，不方便管理与阅读。图片处理服务提供用户将常见的操作保存成一个别名，即样式（Style）。一个复杂操作，利用样式功能后，只要用一个很短的URL就能实现相同的效果。

一个频道（Channel）下面有多个样式，样式的作用范围只在一个频道（Channel）下，目前一个频道（Channel）允许最多有50个样式。

#### 样式访问

##### 访问规则

```
<文件URL>@!StyleName
```

文件URL即是由Channel+Object 组成的URL地址，参见图片URL规则

1. @!是样式的分隔符，在URL带了这个分隔符，图片处理服务会把该分隔符后面的内容当成样式的名称。
2. StyleName表示的是样式的名字。
3. 创建样式、删除样式和修改样式都在前端控制台实现。
4. 当访问的样式在指定频道（Channel）不存在时，将返回NoSuchStyle错误。

## 示例

假如对image-demo这个Channel创建三个样式。

样式名	样式内容
pipe1	150w_150h_1e_1c_0i_100q_1x.jpg  watermark=1&object=cGFuZGEucG5n&t=51 &p=9&x=10&y=10
pipe2	250w_250h_0e_0c_0i_90q_1x.jpg 150w_150h_0 e_1c_1i_90q_1x.jpg
pipe3	180w_180h_1e_1c_0i_0o_90q_1x.jpg  watermark=2&type=d3F5LXplbmhlaQ&size= 25&text=SGVsbG8g5Zu- 54mH5pyN5YqhIQ&color=IzAwMDAwMA&t= 90&p=9&x=10&y=10

对于样式1(pipe1):

- [http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@150w\\_150h\\_1e\\_1c\\_0i\\_100q\\_1x.jpg|watermark=1&object=cGFuZGEucG5n&t=51&p=9&x=10&y=10](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@150w_150h_1e_1c_0i_100q_1x.jpg|watermark=1&object=cGFuZGEucG5n&t=51&p=9&x=10&y=10)



- <http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg!pipe1>



两者是一样的。

对于样式2(pipe2)：

- [http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@250w\\_250h\\_0e\\_0c\\_0i\\_90q\\_1x.jpg|150w\\_150h\\_0e\\_1c\\_1i\\_90q\\_1x.jpg](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@250w_250h_0e_0c_0i_90q_1x.jpg|150w_150h_0e_1c_1i_90q_1x.jpg)



- <http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@!pipe2>



效果是一样的

对于样式3 (pipe3) :

- [http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@180w\\_180h\\_1e\\_1c\\_0i\\_0o\\_90q\\_1x.jpg|watermark=2&type=d3F5LXplbmhlaQ&size=25&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=IzAwMDAwMA&t=90&p=9&x=10&y=10](http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@180w_180h_1e_1c_0i_0o_90q_1x.jpg|watermark=2&type=d3F5LXplbmhlaQ&size=25&text=SGVsbG8g5Zu-54mH5pyN5YqhIQ&color=IzAwMDAwMA&t=90&p=9&x=10&y=10)



- <http://image-demo.img-cn-hangzhou.aliyuncs.com/example.jpg@!pipe3>



效果是一样的

## 样式相关操作

### 样式相关操作

图片服务提供了一些Style相关的操作，用于增删查改样式（Style）。

### Put Style

在某个频道（Channel）下，创建一个样式（Style），将复杂的图片服务的处理参数保存成一个样式，创建样式后，通过样式实现与参数同样的图片处理效果。

#### 请求语法

```
PUT /?style&styleName=YourStyleName HTTP/1.1
Host: ChannelName.img-cn-hangzhou.aliyuncs.com
Date: GMT Date
Authorization: SignatureValue
```

```
<?xml version="1.0" encoding="UTF-8"?>
<Style>
<Content>100w_200h.jpg</Content>
</Style>
```

### 请求参数(Request Parameters)

参数名称	说明	是否必须	取值范围
styleName	待创建的Style的名称	是	参见style命名规则

### 请求元素(Request Elements)

参数名称	说明	是否必须	取值范围
Content	Style对应的内容，表示处理图像所用的参数	是	参见图像处理参数

### 请求示例

```
PUT /?style&styleName=style-example HTTP/1.1
Host: channel-example.img-cn-hangzhou.aliyuncs.com
Date: Thu, 08 Jan 2015 06:17:55 GMT
Authorization: OSS 2onpuorvhikxergnrzmwkn0t:Hyc0UH+CXKQv6ExbZMi+HPn4Gxc=

<?xml version="1.0" encoding="UTF-8"?>
<Style>
<Content>100w_200h.jpg</Content>
</Style>
```

### 返回示例

```
HTTP/1.1 200 OK
x-img-request-id: 54AE211379B222C77F000016
Date: Thu, 08 Jan 2015 06:17:55 GMT
Connection: close
Content-Length: 0
Server: AliyunOSS
```

### 细节分析

1. 如果Channel不存在，返回404 Not Found错误，错误码：NoSuchChannel。
2. 每个Channel下最多能创建50个Style。如果超过50个，Create Style时返回403 Forbidden错误，错误码：AccessDenied，错误消息为：Your style count is exceeded 50.
3. 只有Channel的拥有者才能在该Channel下创建Style。如果试图在一个不属于自己的Channel下创建Style，返回403 Forbidden错误，错误码：AccessDenied。
4. 如果待创建的Style已经存在，则会更新原有的Style。

## List Style

List Style可以获取某个频道（Channel）下的所有样式（Style）的信息。

### 请求语法

```
GET /?style HTTP/1.1
Host: ChannelName.img-cn-hangzhou.aliyuncs.com
```

Date: GMT Date  
Authorization: SignatureValue

### 响应元素(Response Elements)

参数名称	说明
Name	Style名称
Content	Style对应的内容
CreateTime	Style创建时间
LastModifyTime	Style最后修改的时间

### 请求示例

```
GET /?style HTTP/1.1
Host: channel-example.img-cn-hangzhou.aliyuncs.com
Date: Thu, 08 Jan 2015 06:28:02 GMT
Authorization: OSS 2onpuorvhikxergnrzmwkn0t:zDI6c1trJAGHbR8rreyzq6lMq9U=
```

### 返回示例

```
HTTP/1.1 200 OK
x-img-request-id: 54AE237279B222C77F000023
Date: Thu, 08 Jan 2015 06:28:02 GMT
Connection: close
Content-Type : application/xml
Content-Length: 568
Server: AliyunOSS

<?xml version="1.0" encoding="UTF-8"?>
<StyleList>
  <Style>
    <Name>style-example1</Name>
    <Content>400w</Content>
    <CreateTime> Thu, 08 Jan 2015 06:28:02 GMT </CreateTime>
    <LastModifyTime> Thu, 08 Jan 2015 06:28:02 GMT</LastModifyTime>
  </Style>
  <Style>
    <Name>style-example2</Name>
    <Content>400w</Content>
    <CreateTime>Thu, 08 Jan 2015 06:28:02 GMT</CreateTime>
    <LastModifyTime> Thu, 08 Jan 2015 06:28:02 GMT</LastModifyTime>
  </Style>
</StyleList>
```

### 细节分析

1. 如果频道 ( Channel ) 不存在, 返回404 Not Found错误, 错误码: NoSuchChannel。
2. 只有频道 ( Channel ) 的拥有者才能List该频道 ( Channel ) 下的样式 ( Style )。如果试图在一个不属于自己的频道 ( Channel ) 下List Style, 返回403 Forbidden错误, 错误码: AccessDenied。

## Get Style

Get Style可以获取某个样式（Style）的属性信息，包括样式名称、内容，以及创建和最后修改时间。

请求语法：

```
GET /?style&styleNae=YourStyleName HTTP/1.1
Host: ChannelName.img-cn-hangzhou.aliyuncs.com
Date: GMT Date
Authorization: SignatureValue
```

### 请求参数(Request Parameters)

参数名称	说明	是否必须	取值范围
styleName	需要获取的Style的名称	是	参见style命名规则

### 响应元素(Response Elements)

参数名称	说明
Name	Style名称
Content	Style对应的内容
CreateTime	Style创建时间
LastModifyTime	Style最后修改的时间

### 请求示例

```
GET /?style&styleNae=style-example HTTP/1.1
Host: channel-example.img-cn-hangzhou.aliyuncs.com
Date: Thu, 08 Jan 2015 06:20:12 GMT
Authorization: OSS 2onpuorvhikxergnrzmwkn0t:SQe/ZdW92fmFgLEiIwsH4f8YTA8=
```

### 返回示例

```
HTTP/1.1 200 OK
x-img-request-id: 54AB9937B703C78879000167
Date: Thu, 08 Jan 2015 06:20:12 GMT
Connection: close
Content-Type : application/xml
Content-Length: 236
Server: AliyunOSS

<?xml version="1.0" encoding="UTF-8"?>
<Style>
  <Name>jujht9w0d4</Name>
  <Content>erp2g2twla</Content>
  <CreateTime>Thu, 08 Jan 2015 06:20:12 GMT</CreateTime>
  <LastModifyTime>Thu, 08 Jan 2015 06:20:12 GMT</LastModifyTime>
</Style>
```

## 细节分析

1. 如果频道 ( Channel ) 不存在，返回404 Not Found错误，错误码：NoSuchChannel。
2. 如果样式 ( Style ) 不存在，返回404 Not Found错误，错误码：NoSuchStyle。

## Delete Style

Delete Style用来删除某个样式 ( Style ) 。

### 请求语法

```
DELETE /?style&styleNae=YourStyleName HTTP/1.1
Host: ChannelName.img-cn-hangzhou.aliyuncs.com
Date: GMT Date
Authorization: SignatureValue
```

### 请求参数(Request Parameters)

参数名称	说明	是否必须	取值范围
styleName	待删除的Style的名称	是	参见style命名规则

### 请求示例

```
DELETE /?style&styleNae=style-example HTTP/1.1
Host: channel-example.img-cn-hangzhou.aliyuncs.com
Date: Thu, 08 Jan 2015 06:30:20 GMT
Authorization: OSS 2onpuorvhikxergnrzmwkn0t:mK217ZMjVP30w4Q99vYwBEgddqw=
```

### 返回示例

```
HTTP/1.1 204 No Content
x-img-request-id: 54AE23FC79B222C77F000028
Date: Thu, 08 Jan 2015 06:30:20 GMT
Connection: close
Content-Length: 0
Server: AliyunOSS
```

## 细节分析

1. 如果频道 ( Channel ) 不存在，返回404 Not Found错误，错误码：NoSuchChannel。
2. 不管样式 ( Style ) 存不存在，只要删除操作合法，删除成功后都会返回204 No Content。
3. 只有频道 ( Channel ) 的拥有者才能删除该频道下的样式 ( Style ) 。如果试图删除一个不属于自己的频道下的样式，返回403 Forbidden错误，错误码：AccessDenied。

## 错误响应

### 错误响应

当用户访问图片处理服务出现错误的时候，图片处理服务会返回给用户相应的错误码和错误信息，以帮助用户定位与处理问题。

## 图片处理服务错误的响应格式

错误响应的消息体例子

```
<Error>
  <Code>BadRequest</Code>
  <Message>Input is not base64 decoding.</Message>
  <RequestId>52B155D2D8BD99A15D0005FF</RequestId>
  <HostId>userdomain</HostId>
</Error>
```

错误包含以下元素：

- Code: 图片处理服务返回给用户的错误码。
- Message: 图片处理服务给出的详细错误信息。
- RequestId: 用以标识错误请求的唯一UUID，在无法解决问题时候，可以使用此错误ID发送给图片处理服务的工程师去定位错误的原因。
- HostId：用来标识访问的图片处理服务集群。

## 图片处理服务的错误码

错误码	描述	HTTP 状态码
TooManyPipe	管道数目超过限制	400
InvalidArgument	参数错误	400
BadRequest	错误请求	400
MissingArgument	缺少参数	400
ImageTooLarge	图片大小超过限制	400
WatermarkError	水印错误	400
AccessDenied	拒绝访问	403
SignatureDoesNotMatch	签名不匹配	403
NoSuchFile	图片不存在	404
NoSuchStyle	样式不存在	404
NoSuchChannel	频道不存在	404
InternalServerError	服务内部错误	500
NotImplemented	方法未实现	501