

上传视频封面

接口描述

- 1、通过分类id或者视频id上传视频封面图

2、接口支持https协议

接口URL

http://api.polyv.net/v2/video/upload-cover-image

请求方式

POST

接口约束

- 1、接口同时支持HTTP 、HTTPS ， 建议使用HTTPS 确保接口安全， 接口调用有频率限制， [详细请查看](#)
- 2、请求参数vids（视频id），catelds（分类id）：当传了vids参数，以vids为准，当vids不传，以catelds为准，两个参数不能同时为空。
- 3、请求参数imageFile（图片文件），imageUrl（图片http链接），imageBase64（图片数据的base64形式）：图片上传形式，三者任选其一即可，传了多个则按imageFile，imageUrl，imageBase64顺序选取第一个不为空的形式，**注意**：图片格式仅支持png、jpg、jpeg、gif、bmp，且图片大小不得超过5MB。

请求参数描述

参数名	必选	类型	说明
userid	true	String	保利威点播账户id，可以参考【 获取密钥 】获取,获取路径：官网->登录->点播（API接口）

参数名	必选	类型	说明
ptime	true	Long	当前时间的毫秒级时间戳，3分钟内有效
sign	true	String	签名，为40位大写的SHA1值，生成签名的secretkey密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
vids	true	String	视频id，例如： 1b448be3235552e5fafb16489eb01839_1，多个视频id用英文逗号隔开
catelds	false	String	分类id，例如： 1615536384688，该参数获取自【 获取分类及子分类 】，多个分类id用英文逗号隔开
imageFile	false	File	图片文件
imageUrl	false	String	图片链接，支持http、https协议
imageBase64	false	String	图片数据的base64形式

示例

```
http://api.polyv.net/v2/video/upload-cover-image
```

表单参数：

```
vids=1b448be32336069fd567bffc04f3e137_1%2C1b448be323a6a1a6c0c237856f555e88_1&imageUrl=https%3A%2F%2Fd.1anrentuku.com%2Fdown%2Fpng%2F1410%2Fseo-icons%2Fpenguin.png&sign=25108B8B497AD6FE6F1DB979A7FA617E10CE2252&cateIds=1615536384688&userid=1b448be323&ptime=1620286028736
```

响应参数描述

参数名	类型	说明
code	Integer	响应状态码，200为成功返回，非200为失败【详见 全局错误说明 】
status	String	响应状态文本信息
message	String	响应描述信息，当code为400或者500的时候，辅助描述错误原因
data	Object	响应成功时返回图片的大图和小图链接【详见 data字段说明 】，响应失败时返回空

data字段说明

字段	类型	说明
imageUrlSmall	String	图片的小图http链接，例如： http://img.videocc.net/uimage/1/1b448be323/first_image/645824f1-6697-4d14-b343-1ca5fdd4f574_s.png
imageUrlBig	String	图片的大图http链接，例如： http://img.videocc.net/uimage/1/1b448be323/first_image/645824f1-6697-4d14-b343-1ca5fdd4f574_b.png

Java请求示例

快速接入基础代码请[下载相关依赖源码](#)，[点击下载源代码](#)，下载后加入到自己的源码工程中即可。测试用例中的**HttpUtil.java** 和 **VodSignUtil.java** 都包含在下载文件中。

强烈建议您使用[点播Java SDK](#)完成API的功能对接，点播Java SDK 对API调用逻辑、异常处理、数据签名、HTTP请求线程池进行了统一封装和优化。

```

private static final Logger log = LoggerFactory.getLogger(VodVideoPosterTest.class);
/**
 * 上传视频封面
 */
@Test
public void testUploadPoster() throws Exception, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String secretKey = super.secretKey;
    String userid = super.userId;
    String ptime = String.valueOf(System.currentTimeMillis());
    //业务参数
    String url = "http://api.polyv.net/v2/video/upload-cover-image";
    String vids = "1b448be32336069fd567bffc04f3e137_1,1b448be323a6a1a6c0c237856f555e88_1";
    String cateIds = "1615536384688";
    String filePath = getClass().getResource("/file/").getPath() + "poster.png";
    File imageFile = new File(filePath);
    String imageUrl = "https://d.lanrentuku.com/down/png/1410/seo-icons/penguin.png";

    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("userid", userid);
    requestMap.put("ptime", ptime);
    requestMap.put("vids", vids);
    requestMap.put("cateIds", cateIds);
    requestMap.put("imageUrl", imageUrl);
    Map<String, File> fileMap = new HashMap<>();
    fileMap.put("imageFile", imageFile);
    requestMap.put("sign", VodSignUtil.getSign(requestMap, secretKey));

    String response = HttpUtil.postFile(url, requestMap, fileMap, Constant.UTF8);
    log.debug("测试上传视频封面,{}", response);
    //do somethings
}

```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```

{
  "code": 200,
  "status": "success",
  "message": "success",
  "data": {
    "imageUrlSmall": "http://img.videocc.net/uimage/1/1b448be323/first_image/645824f1-6697-4d14-b343-1ca5fdd4f574_s.png",
    "imageUrlBig": "http://img.videocc.net/uimage/1/1b448be323/first_image/645824f1-6697-4d14-b343-1ca5fdd4f574_b.png"
  }
}

```

异常示例

签名不正确

```
{
  "code":400,
  "status":"error",
  "message":"the sign is not right",
  "data":""
}
```

时间戳过期

```
{
  "code":400,
  "status":"error",
  "message":"ptime is too old.",
  "data":""
}
```

内部错误, 或传了非法参数程序转换异常

```
{
  "code": 500,
  "status": "fail",
  "message": "undefined error",
  "data": null
}
```