

裁剪窗口

简要描述:

- 图像裁剪框

产品介绍:

- 图像裁剪 el-copper

使用场景:

- 图像裁剪

参数: *标注为常用

参数名	必选	类型	说明	默认
*img	否	string	图片路径	""
*title	否	string	裁剪框标题	' 图像裁剪'
*details	否	string	裁剪详情	' 图片尽量不要超过 3M...'
*outputtype	否	string	允许图片类型	"png jpg jpeg"
*fixednumber	否	Array	裁剪框宽高比例 fixed 的参数 无需传入（根据你是否传入 fixednumber 自动开启）传入格式 [1,1]	[]
*autocropwidth	否	Number	裁剪框宽度	200
*autocropheight	否	Number	裁剪框高度	200
maximgsize	否	Number	上传图片最大尺寸	2000
enlarge	否	Number	图片根据截图框输出比例倍数	1
size	否	Number	裁剪生成图片的质量	1
canmove	否	Boolean	上传图片是否可以移动	true
canmovebox	否	Boolean	是否允许拖动裁剪框	true
fixedbox	否	Boolean	固定截图框大小不允许改变	false
autocrop	否	Boolean	是否默认生成截图框	True
centerbox	否	Boolean	截图框是否被限制在图片里面	false

返回方法:

方法名	参数	说明	必选
*getcropperimg	data	返回裁剪图片裁剪	是
*closecropper	data	关闭弹窗	是
*updateimg	data	更改图片（当你在父组件更改图片和上传图片（input file）时使用）	否

使用说明

1 引入

```
<!-- css -->
<link rel="stylesheet" href="https://unpkg.com/element-ui/lib/theme-chalk/index.css">
<link rel="stylesheet" type="text/css" href="el-cropper.css">

<!-- js -->
<script type="text/javascript" src="vue.js"></script>
<script src="el-cropper.js"></script>
```

使用说明：在页面中引入即可

2 注册组件

```
<button @click="showcropper">上传照片</button>
<el-cropper
  :img="obj.img"
  v-if="obj.showhidden"
  v-on:getcropperimg='getcropperimg'
  v-on:updateimg='updateimg'
  @closecropper="hidecropper">
</el-cropper>
```

3data 数据

```
data() {  
  return {  
    obj: {  
      img: '',  
      showhidden: false,  
      autocropwidth: 300,  
      title: '上传照片',  
      details: '宝贝详情图片上传有:主图图片上传和宝贝描述里的图片上传。 打开宝贝详情编辑页面 ,先把宝贝的主图图片上',  
      fixednumber: [1, 1]  
    },  
  },  
}
```

4 方法

```
methods: {  
  
  // 上传照片 弹出框显示  
  showcropper() {  
    this.obj.showhidden=true;  
  },  
  
  // 弹出框隐藏(在一个页面时使用多次, 隐藏绑定同一个方法即可)  
  hidecropper() {  
    this.obj.showhidden = false;  
  },  
  
  // (父组件更改图片时加上即可)  
  // 使用场景: 父组件中的更改图片或者上传按钮为<input file >可以在  
  // onchange 事件中修改传入的图片  
  updateimg: function(data) {  
    console.log(data.img);  
    this.img = data.img  
  },  
  
  // 裁剪后的图片路径 base64  
  getcropperimg: function(data) {  
    console.log(data.img);  
    return data.img;  
  },  
},
```

5 同一页面使用多次

1, 绑定参数不同

```
<div id="app">
  <!-- 第一次使用 -->
  <button @click="showcropper">上传照片</button>
  <el-cropper
    :img="obj.img"
    :title="obj.title"
    :autocropwidth="obj.autocropwidth"
    :details="obj.details"
    :fixednumber="obj.fixednumber"
    v-if="obj.showhidden"
    v-on:getcropperimg='getcropperimg'
    v-on:updateimg='updateimg'
    @closecropper="closecropper">
  </el-cropper>
  <!-- 第二次使用 -->
  <button @click="showcropper1">上传视频</button>
  <el-cropper
    :img="obj2.img"
    :title="obj2.title"
    :autocropwidth="obj2.autocropwidth"
    :details="obj2.details"
    v-if="obj2.showhidden"
    @getcropperimg='getcropperimg'
    @updateimg='updateimg'
    @closecropper="closecropper">
  </el-cropper>
</div>
```

2, 点击显示事件不同

```
// 上传照片 弹出框显示
showcropper(e){
  this.obj.showhidden=true;
},

// 上传视频 弹出框显示
showcropper1(e){
  this.obj2.showhidden=true;
},
```

3, 隐藏绑定一个方法即可

```
// 弹出框隐藏(在一个页面时使用多次, 隐藏绑定同一个方法即可)
hidecropper(){
  this.obj.showhidden = false;
  this.obj2.showhidden = false;
},
```

- 更多返回错误代码请看源代码

```
• <body>
•   <!-- 同页面使用多次 -->
•     <div id="app">
•       <!-- 第一次使用 -->
•       <button @click="showcropper">上传照片</button>
•       <el-cropper
•         :img="obj.img"
•         :title="obj.title"
•         :autocropwidth="obj.autocropwidth"
•         :details="obj.details"
•         :fixednumber="obj.fixednumber"
•         v-if="obj.showhidden"
•         v-on:getcropperimg='getcropperimg'
•         v-on:updateimg='updateimg'
•         @closecropper="hidecropper">
•     </el-cropper>
•     <!-- 第二次使用 -->
•     <button @click="showcropper1">上传视屏</button>
•     <el-cropper
•       :img="obj2.img"
•       :title="obj2.title"
•       :autocropwidth="obj2.autocropwidth"
•       :details="obj2.details"
•       v-if="obj2.showhidden"
•       @getcropperimg='getcropperimg'
•       @updateimg='updateimg'
•       @closecropper="hidecropper">
•     </el-cropper>
•   </div>
• </body>
•
•
•
•
• <script>
•
•   var app = new Vue({
•     el:' #app',
•     data() {
•       return {
```

```

    obj: {
img:'https://ssl.bdstatic.com/70cFuXSh_Q1YnxGkpoWK1HF6hhy/it/u=3300305952,1328708913&fm=27&gp=
g',
        showhidden: false,
        autocropwidth: 300,
        title:'上传照片',
        details:'宝贝详情图片上传有:主图图片上传和宝贝描述里的图片上传。 打开宝贝详情
页面 , 先把宝贝的主图图片上',
        fixednumber:[1,1]
    },
    obj2: {
img:'https://ss2.bdstatic.com/70cFvnSh_Q1YnxGkpoWK1HF6hhy/it/u=1718395925,3485808025&fm=27&gp=
g',
        showhidden: false,
        autocropwidth: 100,
        // title 不绑定即为默认
    },
    }
},
methods: {

    // 上传照片 弹出框显示
    showcropper(e) {
        this.obj.img =
'https://ssl.bdstatic.com/70cFvXSh_Q1YnxGkpoWK1HF6hhy/it/u=2035447233,1717815544&fm=27&gp=0. jp
        this.obj.showhidden=true;
    },

    // 上传视频 弹出框显示
    showcropper1(e) {
        this.obj2.showhidden=true;
    },

    // 弹出框隐藏(在一个页面时使用多次, 隐藏绑定同一个方法即可)
    hidecropper() {
        this.obj.showhidden = false;
        this.obj2.showhidden = false;
    },

    // (父组件传入 imgurl 调用子组件中的上传图片) 更改图片
    // 使用场景: 父组件中的上传按钮为<input file >可以在 onchange 事件中 修改传入的图片

```

起裁剪框

```
•      updateimg: function(data) {  
•          console.log(data.img);  
•          this.img = data.img  
•      },  
•  
•      // 裁剪后的图片路径 base64  
•      getcropperimg: function(data) {  
•          console.log(data.img);  
•          return data.img;  
•      },  
•  },  
•  })  
•  </script>  
•  </html>
```

```
•  
•
```

```
•
```