**一、resize（图片缩放）功能使用指导**

**1、基本原理**

•调用[acldvppCreateChannel](mk:@MSITStore:C:\Users\ascend\Documents\WeChat%20Files\hu1015138540\FileStorage\File\2020-06\应用开发指导.chm::/zh-cn_topic_0238843648.html#ZH-CN_TOPIC_0238843648)接口创建图片数据处理的通道。

•调用[acldvppVpcResizeAsync](mk:@MSITStore:C:\Users\ascend\Documents\WeChat%20Files\hu1015138540\FileStorage\File\2020-06\应用开发指导.chm::/zh-cn_topic_0238843729.html#ZH-CN_TOPIC_0238843729)异步接口，将输入图片缩放到输出图片大小。对于异步接口，还需调用[aclrtSynchronizeStream](mk:@MSITStore:C:\\Users\\ascend\\Documents\\WeChat%20Files\\hu1015138540\\FileStorage\\File\\2020-06\\应用开发指导.chm::/zh-cn_topic_0238843731.html" \l "ZH-CN_TOPIC_0238843731)接口阻塞Host运行，直到指定Stream中的所有任务都完成。

•调用[acldvppCreateResizeConfig](mk:@MSITStore:C:\Users\ascend\Documents\WeChat%20Files\hu1015138540\FileStorage\File\2020-06\应用开发指导.chm::/zh-cn_topic_0238843682.html#ZH-CN_TOPIC_0238843682)接口创建图片缩放配置数据，不支持指定缩放算法，默认缩放算法为“最近邻插值”。

•缩放后输出图片内存根据YUV420SP格式计算，计算公式：对齐后的宽\*对齐后的高\*3/2

**2、基本流程**

**//1.ACL初始化**

const char \*aclConfigPath = "../src/acl.json";

aclInit(aclConfigPath)；

**//2.运行管理资源申请,包括Device、Context、Stream，stream\_是aclrtStream类型**

aclrtSetDevice(deviceId\_)；

aclrtCreateContext(&context\_, deviceId\_);

aclrtCreateStream(&stream\_);

**//3.创建图片数据处理通道时的通道描述信息，dvppChannelDesc\_是acldvppChannelDesc类**型

dvppChannelDesc\_ = acldvppCreateChannelDesc();

**//4.创建图片数据处理的通道。**

acldvppCreateChannel(dvppChannelDesc\_);

**//5.将图片读入内存，inDevBuffer\_表示存放输入图片的内存, inDevBufferSize表示内存大小，输入内存要提前申请**

acldvppMalloc(&inBufferDev, inBufferSize);

**//6. 创建输入图片的描述信息，并设置各属性值，vpcInputDesc\_是acldvppPicDesc类型**

vpcInputDesc\_ = acldvppCreatePicDesc();

acldvppSetPicDescData(vpcInputDesc\_, reinterpret\_cast<char \*>(inBufferDev));

acldvppSetPicDescFormat(vpcInputDesc\_, PIXEL\_FORMAT\_YUV\_SEMIPLANAR\_420);

acldvppSetPicDescWidth(vpcInputDesc\_, inputWidth);

acldvppSetPicDescHeight(vpcInputDesc\_, inputHeight);

acldvppSetPicDescWidthStride(vpcInputDesc\_, decodeOutWidthStride);

acldvppSetPicDescHeightStride(vpcInputDesc\_, decodeOutHeightStride);

acldvppSetPicDescSize(vpcInputDesc\_, decodeOutBufferSize);

**//7. 申请输出内存vpcOutBufferDev\_，内存大小vpcOutBufferSize\_根据计算公式得出**

acldvppMalloc(&vpcOutBufferDev\_, vpcOutBufferSize\_);

**//8. 创建输出图片的描述信息，并设置各属性值，vpcOutputDesc\_是acldvppPicDesc类型**

**//如果抠图的输出图片作为模型推理的输入，则输出图片的宽高要与模型要求的宽高保持一致**

vpcOutputDesc\_ = acldvppCreatePicDesc();

acldvppSetPicDescData(vpcOutputDesc\_, vpcOutBufferDev\_);

acldvppSetPicDescFormat(vpcOutputDesc\_, PIXEL\_FORMAT\_YUV\_SEMIPLANAR\_420);

acldvppSetPicDescWidth(vpcOutputDesc\_, modelInputWidth);

acldvppSetPicDescHeight(vpcOutputDesc\_, modelInputHeight);

acldvppSetPicDescWidthStride(vpcOutputDesc\_, resizeOutWidthStride);

acldvppSetPicDescHeightStride(vpcOutputDesc\_, resizeOutWidthStride);

acldvppSetPicDescSize(vpcOutputDesc\_, vpcOutBufferSize\_);

**//9. 执行异步缩放，再调用aclrtSynchronizeStream接口阻塞Host运行，直到指定Stream中的所有任务都完成**

acldvppVpcResizeAsync(dvppChannelDesc\_, vpcInputDesc\_,vpcOutputDesc\_, resizeConfig\_, stream\_);

ret = aclrtSynchronizeStream(stream\_);

**//10. 解码结束后，释放资源，包括输入/输出图片的描述信息、输入/输出内存**

(void)acldvppDestroyResizeConfig(resizeConfig\_);

resizeConfig\_ = nullptr;

(void)acldvppDestroyPicDesc(vpcInputDesc\_);

vpcInputDesc\_ = nullptr;

(void)acldvppDestroyPicDesc(vpcOutputDesc\_);

vpcOutputDesc\_ = nullptr;

**3、代码流程图**

