Image Cropping, Insertion and Scaling -- the CROP API

Note

The CROP API is mostly superseded by the newer :ref:`SELECTION API <selection-api>`. The new API should be preferred in most cases, with the exception of pixel aspect ratio detection, which is implemented by :ref:`VIDIOC_CROPCAP <VIDIOC_CROPCAP>` and has no equivalent in the SELECTION API. See :ref:`selection-vs-crop` for a comparison of the two APIs.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 11**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 11**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 11**); ***backlink***

Unknown interpreted text role "ref".

Some video capture devices can sample a subsection of the picture and shrink or enlarge it to an image of arbitrary size. We call these abilities cropping and scaling. Some video output devices can scale an image up or down and insert it at an arbitrary scan line and horizontal offset into a video signal.

Applications can use the following API to select an area in the video signal, query the default area and the hardware limits.

Note

Despite their name, the :ref:`VIDIOC_CROPCAP <VIDIOC_CROPCAP>`, :ref:`VIDIOC_G_CROP <VIDIOC_G_CROP>` and :ref:`VIDIOC_S_CROP <VIDIOC_G_CROP>` ioctls apply to input as well as output devices.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 29**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 29**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst`, **line 29**); ***backlink***

Unknown interpreted text role "ref".

Scaling requires a source and a target. On a video capture or overlay device the source is the video signal, and the cropping ioctls determine the area actually sampled. The target are images read by the application or overlaid onto the graphics screen. Their size (and position for an overlay) is negotiated with the :ref:`VIDIOC_G_FMT <VIDIOC_G_FMT>` and :ref:`VIDIOC_S_FMT <VIDIOC_G_FMT>` ioctls.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 33); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 33); *backlink*

Unknown interpreted text role "ref".

On a video output device the source are the images passed in by the application, and their size is again negotiated with the :ref:`VIDIOC_G_FMT <VIDIOC_G_FMT>` and :ref:`VIDIOC_S_FMT <VIDIOC_G_FMT>` ioctls, or may be encoded in a compressed video stream. The target is the video signal, and the cropping ioctls determine the area where the images are inserted.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 40); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 40); *backlink*

Unknown interpreted text role "ref".

Source and target rectangles are defined even if the device does not support scaling or the :ref:`VIDIOC_G_CROP <VIDIOC_G_CROP>` and :ref:`VIDIOC_S_CROP <VIDIOC_G_CROP>` ioctls. Their size (and position where applicable) will be fixed in this case.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 47); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 47); *backlink*

Unknown interpreted text role "ref".

Note

All capture and output devices that support the CROP or SELECTION API will also support the :ref:`VIDIOC_CROPCAP <VIDIOC_CROPCAP>` ioctl.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 54); *backlink*

Unknown interpreted text role "ref".

Cropping Structures

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 64)

Unknown directive type "kernel-figure".

```
.. kernel-figure:: crop.svg
```

```
:alt:    crop.svg
:align:  center

Image Cropping, Insertion and Scaling

The cropping, insertion and scaling process
```

For capture devices the coordinates of the top left corner, width and height of the area which can be sampled is given by the `bounds` substructure of the struct :c:type:`v4l2_cropcap` returned by the :ref:`VIDIOC_CROPCAP <VIDIOC_CROPCAP>` ioctl. To support a wide range of hardware this specification does not define an origin or units. However by convention drivers should horizontally count unscaled samples relative to 0H (the leading edge of the horizontal sync pulse, see :ref:`vbi-hsync`). Vertically ITU-R line numbers of the first field (see ITU R-525 line numbering for :ref:`525 lines <vbi-525>` and for :ref:`625 lines <vbi-625>`), multiplied by two if the driver can capture both fields.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 74); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 74); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 74); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 74); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 74); *backlink*

Unknown interpreted text role "ref".

The top left corner, width and height of the source rectangle, that is the area actually sampled, is given by struct :c:type:`v4l2_crop` using the same coordinate system as struct :c:type:`v4l2_cropcap`. Applications can use the :ref:`VIDIOC_G_CROP <VIDIOC_G_CROP>` and :ref:`VIDIOC_S_CROP <VIDIOC_G_CROP>` ioctls to get and set this rectangle. It must lie completely within the capture boundaries and the driver may further adjust the requested size and/or position according to hardware limitations.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 86); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]crop.rst`, line 86); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 86); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 86); *backlink*

Unknown interpreted text role "ref".

Each capture device has a default source rectangle, given by the `defrect` substructure of struct :c:type:`v4l2_cropcap`. The center of this rectangle shall align with the center of the active picture area of the video signal, and cover what the driver writer considers the complete picture. Drivers shall reset the source rectangle to the default when the driver is first loaded, but not later.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 95); *backlink*

Unknown interpreted text role "c:type".

For output devices these structures and ioctls are used accordingly, defining the *target* rectangle where the images will be inserted into the video signal.

Scaling Adjustments

Video hardware can have various cropping, insertion and scaling limitations. It may only scale up or down, support only discrete scaling factors, or have different scaling abilities in horizontal and vertical direction. Also it may not support scaling at all. At the same time the struct :c:type:`v4l2_crop` rectangle may have to be aligned, and both the source and target rectangles may have arbitrary upper and lower size limits. In particular the maximum `width` and `height` in struct :c:type:`v4l2_crop` may be smaller than the struct :c:type:`v4l2_cropcap`.`bounds` area. Therefore, as usual, drivers are expected to adjust the requested parameters and return the actual values selected.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 111); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 111); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 111); *backlink*

Unknown interpreted text role "c:type".

Applications can change the source or the target rectangle first, as they may prefer a particular image size or a certain area in the video signal. If the driver has to adjust both to satisfy hardware limitations, the last requested rectangle shall take priority, and the driver should preferably adjust the opposite one. The :ref:`VIDIOC_TRY_FMT <VIDIOC_G_FMT>` ioctl however shall not change the driver state and therefore only adjust the requested rectangle.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 123); *backlink*

Unknown interpreted text role "ref".

Suppose scaling on a video capture device is restricted to a factor 1:1 or 2:1 in either direction and the target image size must be a multiple of 16 Ã— 16 pixels. The source cropping rectangle is set to defaults, which are also the upper limit in this example, of 640 Ã — 400 pixels at offset 0, 0. An application requests an image size of 300 Ã— 225 pixels, assuming video will be scaled down from

the "full picture" accordingly. The driver sets the image size to the closest possible values 304 Ã— 224, then chooses the cropping rectangle closest to the requested size, that is 608 Ã— 224 (224 Ã— 2:1 would exceed the limit 400). The offset 0, 0 is still valid, thus unmodified. Given the default cropping rectangle reported by :ref:`VIDIOC_CROPCAP <VIDIOC_CROPCAP>` the application can easily propose another offset to center the cropping rectangle.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]crop.rst, line 131); *backlink*

Unknown interpreted text role "ref".

Now the application may insist on covering an area using a picture aspect ratio closer to the original request, so it asks for a cropping rectangle of 608 Ã— 456 pixels. The present scaling factors limit cropping to 640 Ã— 384, so the driver returns the cropping size 608 Ã— 384 and adjusts the image size to closest possible 304 Ã— 192.

Examples

Source and target rectangles shall remain unchanged across closing and reopening a device, such that piping data into or out of a device will work without special preparations. More advanced applications should ensure the parameters are suitable before starting I/O.

Note

On the next two examples, a video capture device is assumed; change `V4L2_BUF_TYPE_VIDEO_CAPTURE` for other types of device.

Example: Resetting the cropping parameters

```
struct v4l2_cropcap cropcap;
struct v4l2_crop crop;

memset (&cropcap, 0, sizeof (cropcap));
cropcap.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;

if (-1 == ioctl (fd, VIDIOC_CROPCAP, &cropcap)) {
    perror ("VIDIOC_CROPCAP");
    exit (EXIT_FAILURE);
}

memset (&crop, 0, sizeof (crop));
crop.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;
crop.c = cropcap.defrect;

/* Ignore if cropping is not supported (EINVAL). */

if (-1 == ioctl (fd, VIDIOC_S_CROP, &crop)
    && errno != EINVAL) {
    perror ("VIDIOC_S_CROP");
    exit (EXIT_FAILURE);
}
```

Example: Simple downscaling

```
struct v4l2_cropcap cropcap;
struct v4l2_format format;

reset_cropping_parameters ();

/* Scale down to 1/4 size of full picture. */

memset (&format, 0, sizeof (format)); /* defaults */

format.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;

format.fmt.pix.width = cropcap.defrect.width >> 1;
format.fmt.pix.height = cropcap.defrect.height >> 1;
format.fmt.pix.pixelformat = V4L2_PIX_FMT_YUYV;

if (-1 == ioctl (fd, VIDIOC_S_FMT, &format)) {
    perror ("VIDIOC_S_FORMAT");
    exit (EXIT_FAILURE);
}
```

```
/* We could check the actual image size now, the actual scaling factor
   or if the driver can scale at all. */
```

Example: Selecting an output area

> **Note**
>
> This example assumes an output device.

```
struct v4l2_cropcap cropcap;
struct v4l2_crop crop;

memset (&cropcap, 0, sizeof (cropcap));
cropcap.type = V4L2_BUF_TYPE_VIDEO_OUTPUT;

if (-1 == ioctl (fd, VIDIOC_CROPCAP;, &cropcap)) {
    perror ("VIDIOC_CROPCAP");
    exit (EXIT_FAILURE);
}

memset (&crop, 0, sizeof (crop));

crop.type = V4L2_BUF_TYPE_VIDEO_OUTPUT;
crop.c = cropcap.defrect;

/* Scale the width and height to 50 % of their original size
   and center the output. */

crop.c.width /= 2;
crop.c.height /= 2;
crop.c.left += crop.c.width / 2;
crop.c.top += crop.c.height / 2;

/* Ignore if cropping is not supported (EINVAL). */

if (-1 == ioctl (fd, VIDIOC_S_CROP, &crop)
    && errno != EINVAL) {
    perror ("VIDIOC_S_CROP");
    exit (EXIT_FAILURE);
}
```

Example: Current scaling factor and pixel aspect

> **Note**
>
> This example assumes a video capture device.

```
struct v4l2_cropcap cropcap;
struct v4l2_crop crop;
struct v4l2_format format;
double hscale, vscale;
double aspect;
int dwidth, dheight;

memset (&cropcap, 0, sizeof (cropcap));
cropcap.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;

if (-1 == ioctl (fd, VIDIOC_CROPCAP, &cropcap)) {
    perror ("VIDIOC_CROPCAP");
    exit (EXIT_FAILURE);
}

memset (&crop, 0, sizeof (crop));
crop.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;

if (-1 == ioctl (fd, VIDIOC_G_CROP, &crop)) {
    if (errno != EINVAL) {
        perror ("VIDIOC_G_CROP");
        exit (EXIT_FAILURE);
    }

    /* Cropping not supported. */
    crop.c = cropcap.defrect;
}

memset (&format, 0, sizeof (format));
```

```
format.fmt.type = V4L2_BUF_TYPE_VIDEO_CAPTURE;

if (-1 == ioctl (fd, VIDIOC_G_FMT, &format)) {
    perror ("VIDIOC_G_FMT");
    exit (EXIT_FAILURE);
}

/* The scaling applied by the driver. */

hscale = format.fmt.pix.width / (double) crop.c.width;
vscale = format.fmt.pix.height / (double) crop.c.height;

aspect = cropcap.pixelaspect.numerator /
     (double) cropcap.pixelaspect.denominator;
aspect = aspect * hscale / vscale;

/* Devices following ITU-R BT.601 do not capture
   square pixels. For playback on a computer monitor
   we should scale the images to this size. */

dwidth = format.fmt.pix.width / aspect;
dheight = format.fmt.pix.height;
```