Comparison with old cropping API

The selection API was introduced to cope with deficiencies of the older :ref:`CROP API <crop>`, that was designed to control simple capture devices. Later the cropping API was adopted by video output drivers. The ioctls are used to select a part of the display were the video signal is inserted. It should be considered as an API abuse because the described operation is actually the composing. The selection API makes a clear distinction between composing and cropping operations by setting the appropriate targets.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]selection-api-vs-crop-api.rst`, **line 9**); *backlink*
>
> Unknown interpreted text role "ref".

The CROP API lacks any support for composing to and cropping from an image inside a memory buffer. The application could configure a capture device to fill only a part of an image by abusing V4L2 API. Cropping a smaller image from a larger one is achieved by setting the field `bytesperline` at struct :c:type:`v4l2_pix_format`. Introducing an image offsets could be done by modifying field `m_userptr` at struct :c:type:`v4l2_buffer` before calling :ref:`VIDIOC_QBUF <VIDIOC_QBUF>`. Those operations should be avoided because they are not portable (endianness), and do not work for macroblock and Bayer formats and mmap buffers.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]selection-api-vs-crop-api.rst`, **line 18**); *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]selection-api-vs-crop-api.rst`, **line 18**); *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]selection-api-vs-crop-api.rst`, **line 18**); *backlink*
>
> Unknown interpreted text role "ref".

The selection API deals with configuration of buffer cropping/composing in a clear, intuitive and portable way. Next, with the selection API the concepts of the padded target and constraints flags are introduced. Finally, struct :c:type:`v4l2_crop` and struct :c:type:`v4l2_cropcap` have no reserved fields. Therefore there is no way to extend their functionality. The new struct :c:type:`v4l2_selection` provides a lot of place for future extensions.

> **System Message: ERROR/3** (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]selection-api-vs-crop-api.rst`, **line 29**); *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]selection-api-vs-crop-api.rst`, **line 29**); *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]selection-api-vs-crop-api.rst`, **line 29**); *backlink*
>
> Unknown interpreted text role "c:type".

Driver developers are encouraged to implement only selection API. The former cropping API would be simulated using the new one.