

# Introduction to cv\_lite

---

## Introduction to cv\_lite

[What is the CV\\_Lite Library?](#)

[Why use the CV\\_Lite library?](#)

[Additional Notes](#)

## What is the CV\_Lite Library?

---

The `cv_lite` module is a lightweight image processing module based on `opencv` for specific tasks. It provides accelerated methods for some common tasks, complementing the methods in the `image` module of `opencv`. It is important to note that it is not an `opencv` library and only provides accelerated versions of some tasks.

## Why use the CV\_Lite library?

---

The K230 uses the OpenCV-based image processing library by default. Using the OpenCV image processing library does not fully utilize the powerful processing performance of the K230 chip. Using the `cv_lite` library, on the other hand, allows for higher frame rates and better detection results. Below is a frame rate comparison of some common functions.

Note: Data sourced from Canaan Technology's official documentation.

For inputs with the same resolution, we compared the processing efficiency of OpenMV processing an RGB565 image and `cv_lite` processing an RGB888 image. The resulting frame rate comparison results are shown in the table below. The following frame rates are only compared when processing a fixed scene. The actual frame rate may be affected by scene complexity, such as the number of circles. Please refer to the specific scene for accuracy.

<b>Task</b>	<b>Input Resolution</b>	<b>CV_Lite Processing Frame Rate (fps)</b>	<b>OpenMV Processing Frame Rate (fps)</b>
Grayscale image find_blobs	480x640	90	57
Color image find_blobs	480x640	80	44
Grayscale image find_circles	480x640	24	1.2
Color image find_circles	480x640	24	1.2
Grayscale image find_rectangles	480x640	40	8
Color image find_rectangles	480x640	38	4.6
Grayscale image find_edges	480x640	57	11
Color Image find_edges	480x640	53	Supports grayscale images only
Grayscale Image Binarization	480x640	90	90
Color Image Binarization	480x640	90	40
Color Image Mean Filter	480x640	26	19
Color Image Gaussian Filter	480x640	12	4

In addition to the above optimizations, cv\_lite also adds interfaces for performing morphological operations, white balance, exposure adjustment, and histogram statistics on RGB888 images using software processing:

Task	Input Resolution	cv_lite Processing Frame Rate (fps)
Erosion	480x640	90
Dilation	480x640	32
Opening	480x640	31
Closing	480x640	32
Morphological Gradient	480x640	12
Top Hat Transform	480x640	12
Black Hat Transform	480x640	12
Grayscale World White Balance	480x640	47
White Patch White Balance	480x640	22
Exposure Adjustment	480x640	65
RGB888 Image Histogram Statistics	480x640	77

## Additional Notes

Here are explanations of common formats in the `cv_lite` module:

- Input data format: `u1ab.numpy.ndarray` type, typically obtained using `image.to_numpy_ref()`.
- Converting `u1ab.numpy.ndarray` back to the `image` instance type is typically accomplished using `img = image.Image(image_width, image_height, image.GRAYSCALE, alloc=image.ALLOC_REF, data=np_data)`. Note that the image type and data size meet the requirements.
- The previous two steps do not reallocate memory; the same memory block is used, which is not very time-consuming.
- RGB888 format data can be used with OpenMV's `image` module to convert it using `to_rgb565()`.

For a complete API reference, please refer to Section 2 of this chapter, 2.cv\_lite API Reference Manual.