cropped_img = img[y1:y2, x1:x2]

"how it works explain step by step"

The expression `cropped_img = img[y1:y2, x1:x2]` is a common way to crop or extract a portion of an image using NumPy slicing in Python. Here's a step-by-step explanation of how it works:

1. **Understanding the Image as a NumPy Array**
   In OpenCV, images are represented as multi-dimensional NumPy arrays. For a color image, this array is 3-dimensional: height, width, and channels (color layers). The shape looks like `(height, width, channels)`.
2. **Array Indexing Basics**
   NumPy arrays can be sliced using the syntax `array[start:stop]`, which extracts elements from the `start` index up to (but not including) the `stop` index along that dimension.
3. **Slicing Syntax for Images**
   Since images have height and width dimensions, to crop a rectangular region, you slice the rows (height) and columns (width) of the array like this:
   - `y1:y2` — Selects rows from y1 to y2 (vertical direction).
   - `x1:x2` — Selects columns from x1 to x2 (horizontal direction).
4. **Putting It Together:** `img[y1:y2, x1:x2]`
   Here, `img` is the image array. The slice `y1:y2` picks the rows (pixels vertically), and `x1:x2` picks the columns (pixels horizontally). This extracts a rectangular portion from row `y1` up to (but not including) `y2` and column `x1` up to (but not including) `x2`.
5. **Resulting Cropped Image**
   The result assigned to `cropped_img` is a smaller image that contains only the pixels within this rectangle. The cropped image keeps the same number of color channels as the original.

Visual Example:
If you imagine an image as a grid of pixels, slicing picks a rectangular box inside that grid defined by the top-left corner `(x1, y1)` and the bottom-right corner `(x2, y2)`.

Summary:

- `img[y1:y2, x1:x2]` extracts a rectangular section of the image from pixel rows `y1` to `y2-1` and columns `x1` to `x2-1`.
- This uses NumPy's powerful slicing capability on the image array.
- The order is `[rows, columns]` which corresponds to `[y, x]` coordinates in image terms.

Let me know if you want a code example illustrating this in practice!