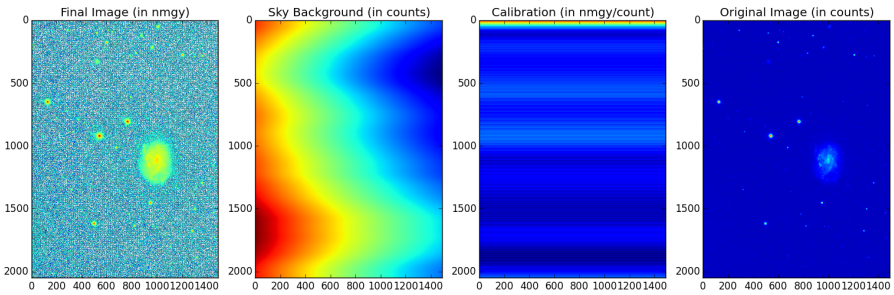
```
        """
        rectangular = []
        bound = self.getBound()
        cropping = []

        for file in self.file_list:
            text = self.getFITS(file)
            r_bound = [round(bound[0]), round(bound[1])]
            c_bound = [round(bound[2]), round(bound[3])]

            if type == 'nano':
                matrix = text[0].data
                cropping = matrix[r_bound[0]:r_bound[1], c_bound[0]:c_bound[1]]

            if type == 'count':
                data = text[2].data[0]
                xinterp = data[1]
                yinterp = data[2]
                matrix = data[0]
                sky_img = self.back_interpolation(xinterp, yinterp, matrix)

                matrix = text[0].data
                new_matrix = matrix / text[1].data + sky_img
                cropping = new_matrix[r_bound[0]:r_bound[1], c_bound[0]:c_bound[1]]

            rectangular.append(cropping)
        return rectangular
```

The getRec.rectangular() method has been improved like this. According to https://acrider.wordpress.com/2015/01/20/extracting-the-data-number-image-dn_image-from-sdss/ , we use calibration to transform between nanomaggies and count images. Sky-image is added in the final result. We still need to improve the interpolation method. Interpolation is used to transform the [256*192] sky-img into [2048, 1489] full image.



nano sky_img calibration final