Attention when using annotation tool labelImg

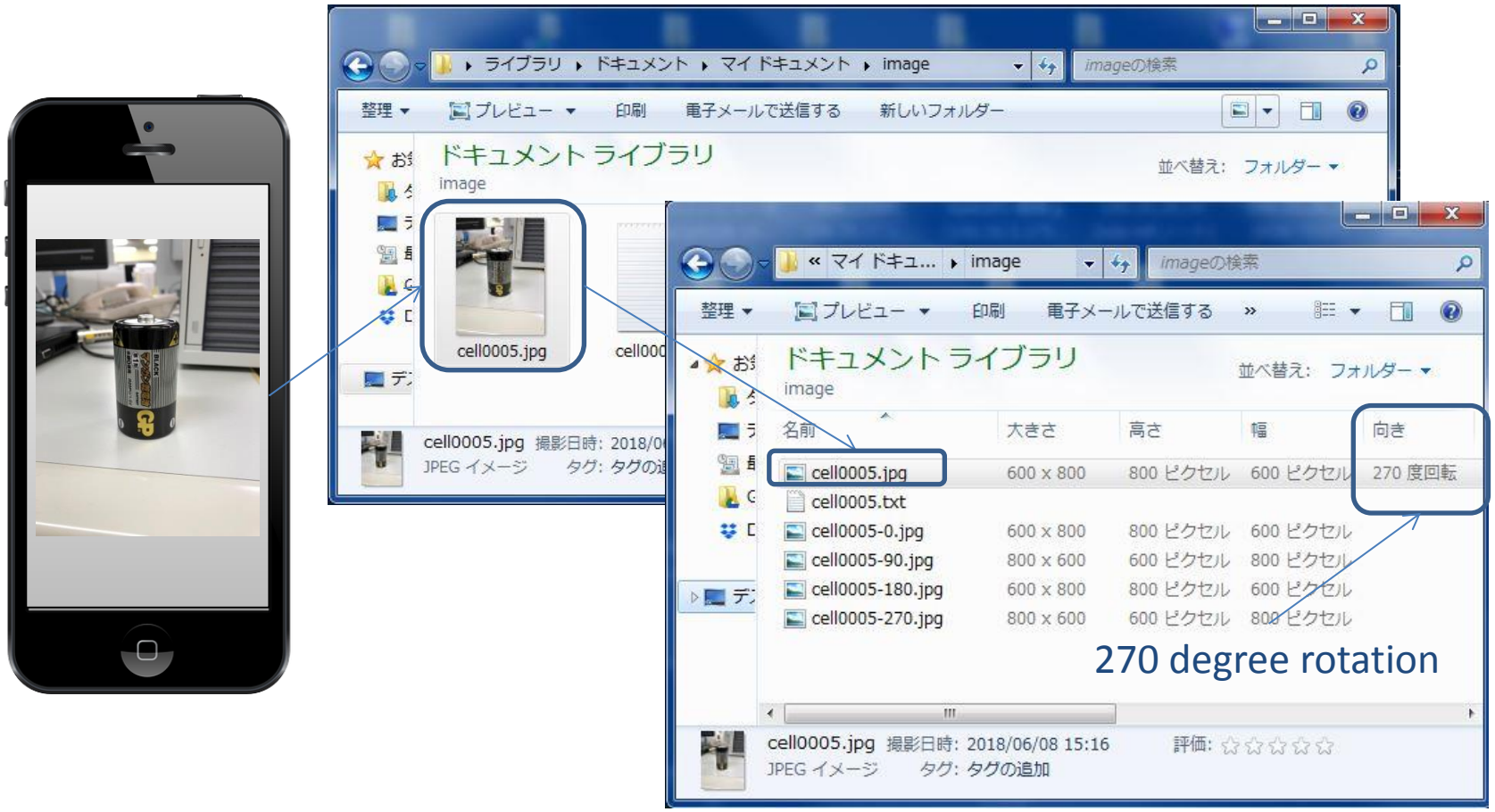
Delete Exif information in advance

2018.10.19

keides2

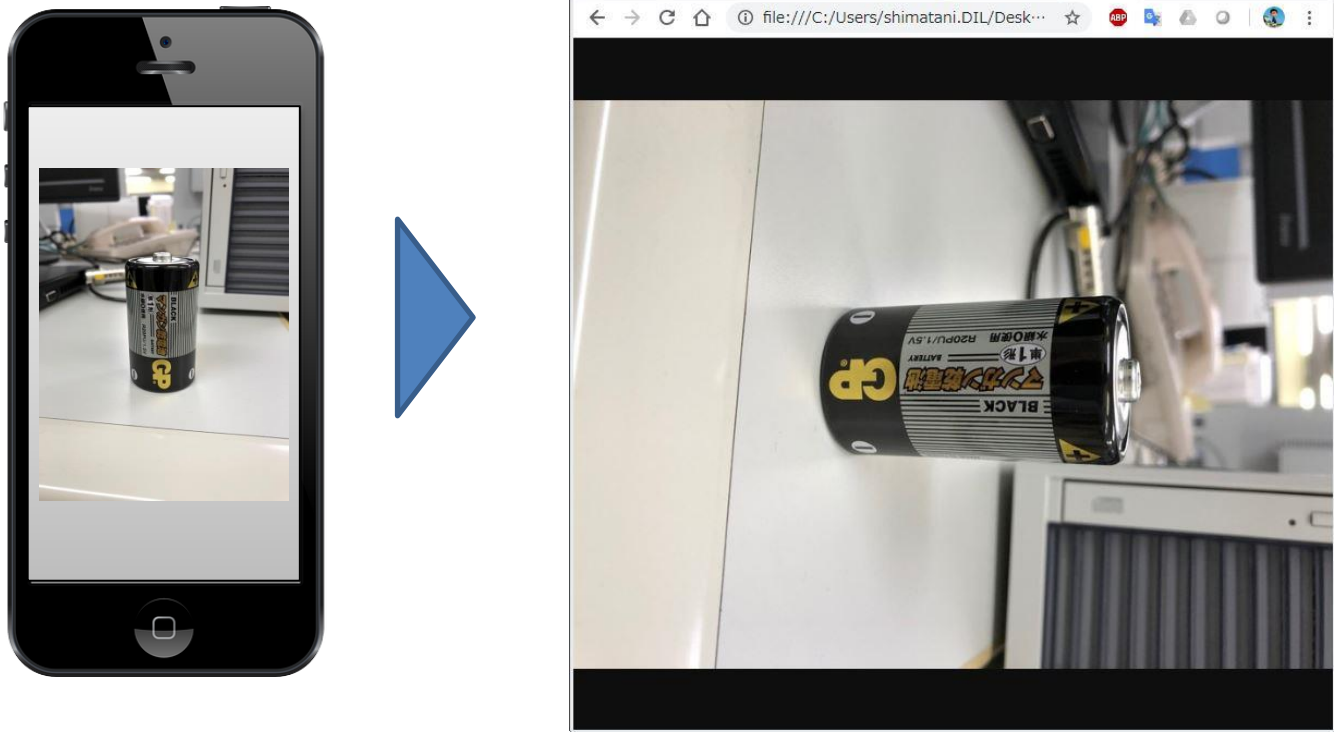
Image taken with smartphone contains rotation information of the image

- Image rotation information is Orientation in Exif information



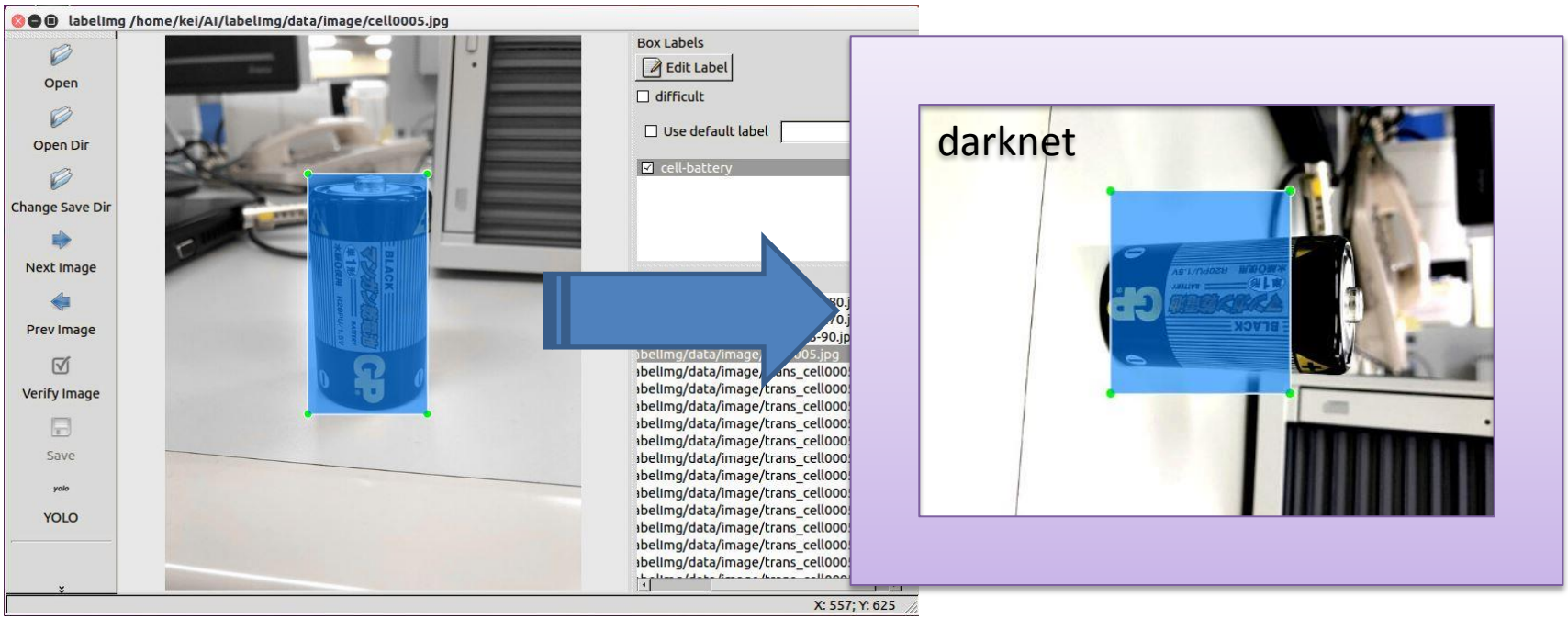
Display on a browser such as Chrome that does not reflect Exif information

- Since I guess darknet does not reflect exif information, if you display an image with Exif information using Chrome which does not reflect Exif information as well, it looks like the figure below.
- I guess darknet is watching similar images.



When you create a bounding box with labelImg

- Even if I make a bounding box in the state of the left figure, I guess that darknet understands as shown in the right figure.
- At this time, learning will be done without correct annotation.

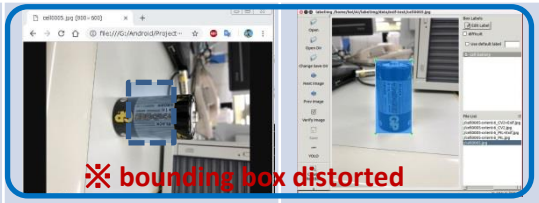
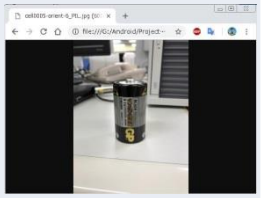
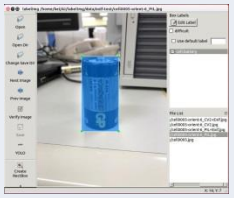
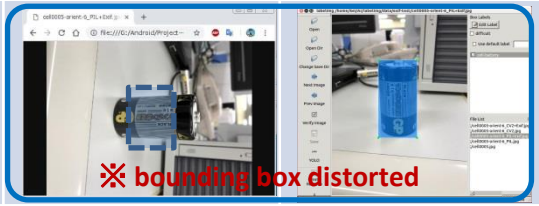
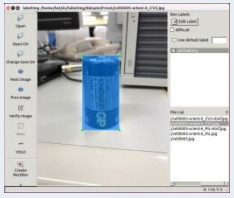


* Since the annotation coordinate format of darknet expresses the center and height / width of the bounding box as a ratio to the image size, if the aspect ratio of the image changes, the bounding box also becomes distorted

Therefore, before annotating with labelImg, delete Exif information in advance

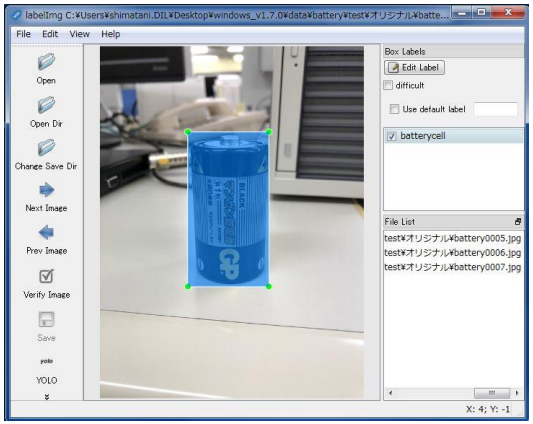
- In advance,
  - Before annotation or before learning
  - If you want to increase N by using Yolo-img-x28_windows.py, you can do after annotation is executed
  - Because it increases the image by N in the direction as seen with labelImag, deletes Exif information and saves it
- Script storage location
  - https://github.com/keides2/android-yolo-v2/tree/tts-JP/scripts/labelImg
  - exif-test.py
    - You can check the difference between handling of library PIL and OpenCV images
  - yolo-img-x28_windows.py
    - You can generate 24 sets of images and annotation files from one set of images and annotation files.

Handling results of images by library (using exif-test.py)

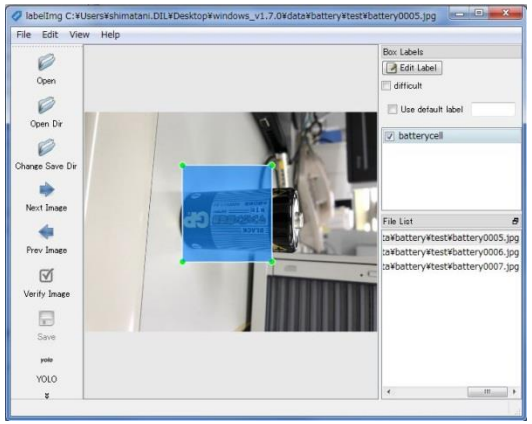
library	Command parameter	File name	Orientation of Exif	Chrome (darknet)	labelImg	Photo viewer	note
-	-	cell0005.jpg（**original**）	270 degree	 ※ bounding box distorted			**darknet misleads at learning**
PIL	img.save()	cell0005-orient-6_PIL.jpg	none				When this command is used to make N increasing tool, darknet does not mislead learning
	img.save(…., exif=exif)	cell0005-orient-6_PIL+Exif.jpg	270 degree	 ※ bounding box distorted			When using this command to create N increasing tool, darknet will mislead at learning
OpenCV	cv2.read(…, cv2.IMREAD_IGNORE_ORIENTATION)	cell0005-orient-6_CV2.jpg	none				When this command is used to make N increasing tool, darknet does not mislead learning
	cv2.read()	cell0005-orient-6_CV2+Exif.jpg	none	 ※ bounding box distorted			When using this command to create N increasing tool, darknet will mislead at learning

Notes on deleting Exif (1/2)

- By simply deleting the Exif information by the following method, the already executed annotation data can not be used
  - Use OpenCV image.read () and read the image without argument cv2.IMREAD_IGNORE_ORIENTATION
  - Using JPEG-EXIF_autorotate



labelImg Screen (before deleting Exif)

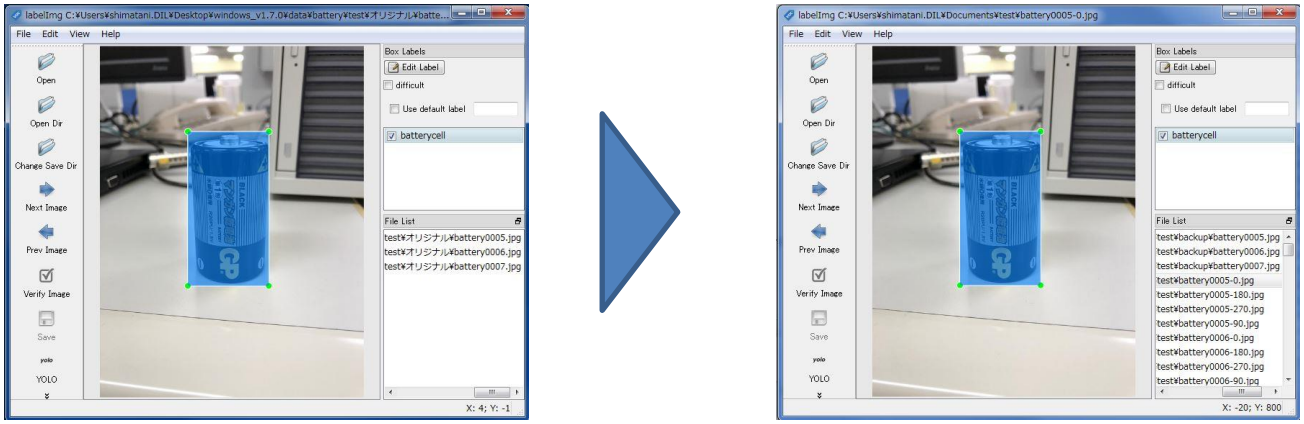


labelImg Screen (after deleting Exif)

- (reference) JPEG-EXIF_autorotate
  - https://savolai.net/software/JPEG-EXIF_autorotate

Notes on deleting Exif (2/2)

- In case of you have already annotated:
- The following tools delete the Exif information after rotating the image based on the Exif information, so you do not need to annotate again.
  - yolo-img-x28_windows.py
  - This tool rotates and converts to increase the image.



labelImg Screen (before deleting Exif)　　labelImg Screen (after deleting Exif)