**<https://github.com/AlexeyAB/darknet/issues/232>**

[**AlexeyAB**](https://github.com/AlexeyAB)**commented**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#issuecomment-336916198)

|  |
| --- |
| Hi,  In the OpenCV version of Yolo you can keep aspect ratio right now, just replace this code: <https://github.com/opencv/opencv/blob/73af899b7c737677f008b831c8e61eaeb2984342/samples/dnn/yolo_object_detection.cpp#L58-L65>  //! [Resizing without keeping aspect ratio]  cv::Mat resized;  cv::resize(frame, resized, cv::Size(network\_width, network\_height));  //! [Resizing without keeping aspect ratio]  //! [Prepare blob]  Mat inputBlob = blobFromImage(resized, 1 / 255.F);  //! [Prepare blob]  to this:  //! [Prepare blob]  Mat inputBlob = blobFromImage(frame, 1 / 255.F, cv::Size(network\_width, network\_height)); //Convert Mat to batch of images  //! [Prepare blob] |

[](https://github.com/AlexeyAB)

Owner

[**AlexeyAB**](https://github.com/AlexeyAB)**commented**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#issuecomment-336916784)

|  |  |  |
| --- | --- | --- |
| There are at least 3 versions of Yolo:   1. Yolo v1 with fully connected layers: <https://pjreddie.com/darknet/yolov1/> 2. Yolo v2 fully convolutional network [yolo.2.0.cfg](https://github.com/pjreddie/darknet/blob/master/cfg/yolo.2.0.cfg) and [yolo-voc.2.0.cfg](https://github.com/pjreddie/darknet/blob/master/cfg/yolo-voc.2.0.cfg) - that used in my fork: <https://arxiv.org/pdf/1612.08242.pdf> 3. Yolo v2.x with keeping of aspect ratio - current (since 10 Apr 2017): <https://pjreddie.com/darknet/yolo/>   It seems to me it is the old behavior of YOLO (v1) : <https://github.com/pjreddie/darknet/blame/179ed8ec76f329eb22360440c3836fdcb2560330/src/demo.c#L44>  No. Yolo v1 used fully conected layers, file yolo\_demo.c instead of demo.c and had to small accuracy, you can find Yolo v1 here: <https://github.com/AlexeyAB/yolo-windows>  This my fork fully corresponds to the Yolo v2 that uses yolo-voc.2.0.cfg or yolo.2.0.cfg and with accuracy 78.6 mAP (VOC 2007), 73.4 mAP (VOC 2012), **44.0 mAP (COCO - table 5)**: <https://arxiv.org/abs/1612.08242>   * **Yolo v2 released at 17 Nov 2016** (1 year ago): [pjreddie@c6afc7f](https://github.com/pjreddie/darknet/commit/c6afc7ff1499fbbe64069e1843d7929bd7ae2eaa) * resize\_image() replaced by letterbox\_image() with **keeping aspect ratio at 10 Apr 2017** (6 monthes ago): [pjreddie@8d9ed0a#diff-4e71a2cf0098713e52e5dae1dfd56c06L44](https://github.com/pjreddie/darknet/commit/8d9ed0a1d680c8d31e453e2e1cebfda66b357c11#diff-4e71a2cf0098713e52e5dae1dfd56c06L44)   So now with keeping of aspect ratio we can get about **48.1 mAP (COCO)** so it adds about +4.1 mAP for COCO: <https://pjreddie.com/darknet/yolo/>  Why didn't you update this behavior in the same way?  May be I will update it latter. Maybe soon Joseph will release a new version of Yolo with new improvements, and I'll add it all together.  This version of Yolo v2 works a bit worse with different aspects of the training and detection datasets, but it works. Aspect ratio invariance is achieved by using crop that depends on jitter parameter in .cfg-file:  [darknet/src/data.c](https://github.com/AlexeyAB/darknet/blob/5a2efd5e5327c56a362442dce70bb3e46201cb89/src/data.c#L697)  Line 697 in [5a2efd5](https://github.com/AlexeyAB/darknet/commit/5a2efd5e5327c56a362442dce70bb3e46201cb89)   |  |  | | --- | --- | |  | image cropped = crop\_image(orig, pleft, ptop, swidth, sheight); | |

[](https://github.com/iraadit)

Author

[**iraadit**](https://github.com/iraadit)**commented**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#issuecomment-336924359)

|  |
| --- |
| Thank you once again for your completer answer.  If I'm modifying your code to include the keeping of the aspect ratio, would you be interested in a Pull Request? |

**[@iraadit](https://github.com/iraadit) [iraadit](https://github.com/iraadit) closed this**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#event-1295086593)

[](https://github.com/iraadit)

Author

[**iraadit**](https://github.com/iraadit)**commented**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#issuecomment-336926709)

|  |
| --- |
| For OpenCV, looking at the definition of [blobFromImage](https://docs.opencv.org/3.3.0/d6/d0f/group__dnn.html" \l "ga0507466a789702eda8ffcdfa37f4d194), it appears to me that the behaviour is different to kept aspect ratio in pjreddie Darknet:  input image is resized so one side after resize is equal to corresponding dimension in size and another one is equal or larger. Then, crop from the center is performed.  It seems to resize and cut the parts of the image that aren't fitting in a square, instead of adding black margins (letterbox) |

**[@iraadit](https://github.com/iraadit) [iraadit](https://github.com/iraadit) reopened this**[**on 16 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#event-1295102811)

[](https://github.com/AlexeyAB)

Owner

[**AlexeyAB**](https://github.com/AlexeyAB)**commented**[**on 17 Oct 2017**](https://github.com/AlexeyAB/darknet/issues/232#issuecomment-336955485)**•**

**edited**

|  |
| --- |
| Yes, you are right, blobFromImage does it in the different way than Darknet.  But there are trade-off in any cases - below is an example of resizing an image in different ways:   1. blobFromImage(): object size **71 x 43**, keeps aspect ratio, but **part of image is lost (cropped)** 2. letterbox\_image(): **the smallest** object size **48 x 28**, keeps aspect ratio and see whole image 3. resize\_image(): object size **48 x 43**, see whole image but **doesn't keep aspect ratio**   If I'm modifying your code to include the keeping of the aspect ratio, would you be interested in a Pull Request?  Known Yolo problem is a difficult to detect of small objects. And letterbox\_image() has the smallest object size 48 x 28.  So yes, I'll apply your pull request, but I think there should be an if-branch that depends on command line flag, which allows us to use the current resize\_image() version without keeping the aspect ratio, and letterbox\_image() version with keeping the aspect ratio.  For example:   1. Original image: [air](https://user-images.githubusercontent.com/4096485/31623978-dc1fcada-b2a9-11e7-99a1-9d87293fcc74.jpg) 2. Resized (416x416) with keeping aspect ratio - OpenCV blobFromImage(): [air_416x416_cropped](https://user-images.githubusercontent.com/4096485/31624238-b102326a-b2aa-11e7-9db1-399b2888927e.jpg) 3. Resized (416x416) with keeping aspect ratio - Darknet letterbox\_image(): [air_416x416_letterbox](https://user-images.githubusercontent.com/4096485/31624244-b5a8a722-b2aa-11e7-9330-bf7b3feef65a.jpg) 4. Resized (416x416) without keeping aspect ratio - this fork of Darknet resize\_image(): [air_416x416](https://user-images.githubusercontent.com/4096485/31624247-b89e6c14-b2aa-11e7-8992-b2fc020575af.jpg) |