Low Altitude and Low Speed Uav Identification based on Hybrid Model

Kai Luo[1, a], Guibin Zhu[1], Ying Li[1], Wentao Wang[2]

[1]Communication NCO school Army Engineering University of PLA, China

[2]National Key Laboratory on Blind Signal Processing, Chengdu, China

[a]jy03667148@163.com

Abstract. The identification of low-altitude and low-speed uav (Unmanned aerial vehicle) is a hot issue in the field of computer vision. Most problems can be solved based on traditional identification methods, but there is a blind spot for low-altitude uav. By contrast, the method based on deep learning can better solve this problem, but in the case of noise and motion blur, the processing effect of CNN and other methods is poor. In order to solve this problem, we put forward a new model. On the basis of the original residual model, we adopt the convolution kernel of different sizes and combine Inception block with multi-scale convolution group. As the low-altitude uav still has the speed, it needs multi-scale convolution to expand the accepted features for identification. Specifically, the residual connection can accelerate our training and meet the real-time requirements. The enlarged convolution kernel can accept more features to satisfy our identification in the case of noise, and multiple ways of convolving kernel concatenate can satisfy our identification in the case of motion blur in the flying uav. Therefore, this paper aims to train a uav with multi-scale convolution kernel, which can effectively identify low-altitude and slow-flying uav under the condition of noise and motion blur. Experimental results show that this method is feasible.

Keywords: Uav identification, Motion blur, Image denoising.

1. Introduction

In recent years, object detection has become a very hot topic in the field of computer vision, or has become the main image task we deal with, most of the time is closely related to our object detection. In the field of non-deep learning approach,

Harr[1]feature, Hog[2]feature, DPM[3]feature, etc. are extracted and then detected by classifiers such as SVM[4].SVM method is effective in classification of small orders of magnitude, but in the case of complex objects, determining the kernel function is still a big problem .In recent years, the object detection algorithm based on deep learning has stood out one after another. For example, the RCNN [5] algorithm has achieved the most basic object detection, which also shows that the deep learning algorithm has a good application for object identification.

The algorithm based on deep learning has achieved great success in the field of computer vision. In image denoising, many algorithms have been proposed one after another. WNNM is a very good algorithm, which improves the performance and achieves better performance under the premise of ensuring efficiency. In the field of motion blur, relevant CNN [6]and other algorithms have also achieved good results. In the same way, the deconvolution [7] also has a wide range of applications in motion blur.

This article puts forward a kind of based on the residual network expansion of convolution kernels to enhance the identification algorithm, inspired by the residual network, the residual unit was improved, we will image noise and image motion blur is the same as the part of our image characteristics, with different sizes of convolution kernels to study our noise and the characteristics of motion blur. The reason for us to do is that, first of all, the basic model we choose is YOLO [8], because our previous algorithms all use classifiers to perform detection, and the real-time performance is bad. With YOLO, you can directly optimize detection end-to-end, and the unified architecture is very fast. The basic YOLO can reach 45FPS.Secondly, the training method and structure of CNN have changed a lot. Batch normalization and residual network can accelerate the training process of the network and achieve real-time in the identification process. Thirdly, we use

the concatenation of multi-scale convolution and multi-scale convolution to replace the fixed convolution kernel of 3*3 size to avoid the single network structure. The network is all set to be trainable, so as to avoid that the function loss cannot be further reduced due to the parameter solidification of previous training. In the experiment, we compared three methods: YOLOv3[9], and two mixed models. Our results show that the modified model has better performance in the face of images with noise and motion blur.

In general, this paper mainly has the following three major contributions. First, it proposes a complex mixed model to identification, which is based on multi-scale convolution kernel. Experiments show that it has better performance and speed than the existing methods.

Second, the network also proves that in the case of noise and motion blur, multiple convolution kernels can be used to learn their features, and the identification function can be better realized without changing the network depth.

Third, we compared the method of adding network depth with the method of using different convolution kernels concatenated to realize identification, and found that the latter has better effect.

The rest of this article is organized as follows. The second part summarizes some methods of image identification, image denoising and motion blur processing in recent years. In the third part, the problems and methods of our research are described in detail. The experimental results of mixed models and YOLOv3 are compared. Finally, we summarize in the fifth part.

2. Related Work

We briefly review the algorithm of object identification. Ross Girshick et al, on the basis of R-CNN, and after improvement, work out that Fast-RCNN [10] will complete classification and box regression simultaneously. In R-CNN, used the linear SVM classification after the completion of the Bounding box regression. In Fast-RCNN, SoftMax classification and box regression are synchronized. But it's still slow. Then, Faster-RCNN proposed by s. Ren, k. He et al. introduced RPN and automatically extracted Region Proposal. The speed was increased by 10 times, basically meeting the demand of real-time. The development of YOLOv3 based on YOLO [11] not only uses the Batch Normalization to expand the input dimension but also USES the structure of SSD [12] to speed it up. The advantages of SSD and FPN [13] are absorbed to form a mixed network.

In our network, we used Residual learning. Residual learning has multiple functions: first, it solved the problem of gradient disappearance caused by too many layers in the network; second, our training speed has been greatly improved; at the last, it can make our model convenient for training with more sparse weights.

While Inception block and Resnet are the same. They were developed from Lenet-5[14], VGG [15] and other methods. As we have realized, the best way to improve network performance is to increase the depth and width of the network, which also means bringing a huge number of parameters.

In order to maintain the sparsity of the network structure, the computational performance of dense block matrix can also be used. Inception block module [16] is adopted mainly to approximate sparse structure into several dense block sub-matrices, so as to reduce parameters and make more effective use of computer resources.

3. Method

3.1 Mixed Convolution Group

Multi-scale convolution can extract different features. 3*3, 5*5 and 7*7 are able to extract different features in a same region. First, connecting these features in depth can not only increase the dimension of the network, but also improve the generalization ability of the network. Inspiration comes from Inception block. However, based on different tasks, we have made many changes. First of all, we have changed the size of the original convolution kernel and replaced it with the convolution kernel of 3*3, 5*5 and 7*7 instead of 1*1.This will help us to learn more features, we remove the pooling

layer and use more filters on the convolution kernel of 3*3 and 5*5.With a ratio of 2:2:1, this combination balances the parameters.

3.2 A Combination of Different Structural

In order to improve the performance of our network and satisfy the identification in both the noisy environment and the environment with motion blur, we are bound to expand our acceptance domain to obtain the context information. Although Resnet[17] can meet the requirements of our context information, but its drawback is obvious, because the convolution kernels that have 1 * 1 and 3 * 3 this two modes, therefore, for small objects detection effect is not ideal, along with motion blur and noise in our environment, to identify under the background of a certain unmanned aerial vehicle (uav) is particularly difficult. Therefore, the general idea of our design is to modify the partial convolution of 1*1,3*3 on the basis of our original Resnet, so that we can not only maintain the traditional point, but also identify the uav in a complex background. Figure 1 below represents the main structure of the traditional Resnet

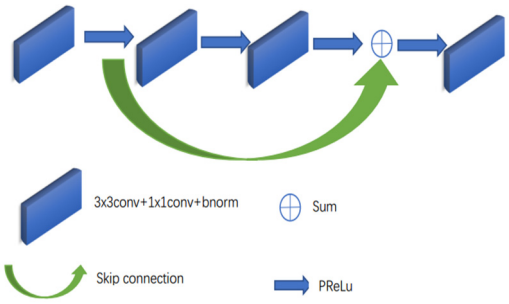


Fig 1. The Main Structure of the Traditional Resnet

below represents the main structure of the traditional Resnet, it can be seen that the traditional network structure only uses the convolution kernel of 3*3. Use identity mapping when block output dimensions are consistent and linear projection when they are inconsistent.

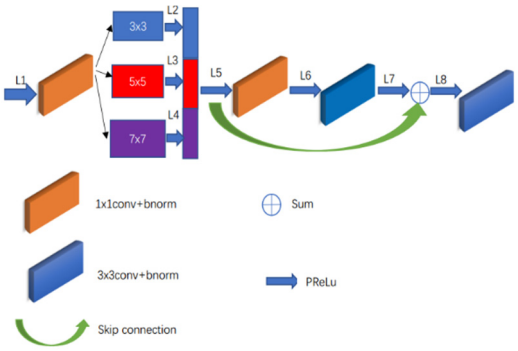


Fig 2. Our Proposed Model

we combine Inception block with the Residual block network structure, and adopt convolution kernels of different sizes, which can expand the scope of our acceptance field, so that the features of noise and motion blur can also be extracted through training.

Let's make a simple explanation of the network module. First, our input is convolved to get $L1$.

$L1$ is convoluted to get $L2$

$$L2 = H_{\text{conv1}}(L1) \tag{1}$$

$H_{\text{convN}}(\bullet)$ the convolution of N*N

$$L3 = H_{\text{conv5}}(L1) \tag{2}$$

$$L4 = H_{\text{conv7}}(L1) \tag{3}$$

$$L5 = H_{partDense}(L2, L3, L4) \tag{4}$$

$H_{partDense}(\bullet,\bullet,\bullet)$ represents the concatenation of three networks

$$L6 = H_{\text{conv1}}(L5) \tag{5}$$

$$L7 = H_{\text{conv3}}(L6) \tag{6}$$

$$L8 = L7 + L5 \tag{7}$$

The above mentioned is the first network module we proposed, Inception block-Residual block.

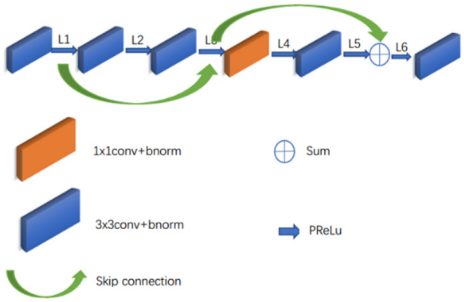


Fig 3. Mixed Model

We are inspired by the Dense block and Residual block, constructing Mixed model. Here is an explanation of the Mixed model

$L1$ is convoluted to get $L2$

$$L2 = H_{\text{conv3}}(L1) \tag{8}$$

$$L3 = H_{partDense}(L1, L2) \tag{9}$$

$H_{partDense}(\bullet,\bullet)$ represents the concatenation of two networks

$$L4 = H_{\text{conv1}}(L3) \tag{10}$$

$$L5 = H_{\text{conv3}}(L4) \tag{11}$$

$$L6 = L5 + L1 \tag{12}$$

The above is our improved network module.

3.3 Comparison of Model Methods

In the same training set, we compared different object detection algorithms, including traditional YOLOv3 and mixed models. Similarly, in order to compare the influence of simply changing the size of Residual block convolution kernel on our recognition effect, we also adjusted the mixed model.

Table 1. Mixed model

	Mixed model	Model Construct		
		Filters	*Size*	*Output*
	Convolution	32	3*3	256*256
	Convolution	64	3*3/2	128*128
1x	Convolution	32	3*3	
	Convolution	32	3*3	
	Dense			128*128
	Convolution	32	1*1	
	Convolution	64	3*3	
	Residual			128*128
	Convolution	128	3*3/2	64*64
2x	Convolution	64	3*3	
	Convolution	64	3*3	
	Dense			64*64
	Convolution	64	1*1	
	Convolution	128	3*3	
	Residual			64*64
	Convolution	256	3*3/2	32*32
8x	Convolution	128	3*3	
	Convolution	128	3*3	
	Dense			32*32
	Convolution	128	1*1	
	Convolution	256	3*3	
	Residual			32*32
	Convolution	512	3*3/2	16*16
8x	Convolution	256	3*3	
	Convolution	256	3*3	
	Dense			16*16
	Convolution	256	1*1	
	Convolution	512	3*3	
	Residual			16*16
	Convolution	1024	3*3/2	8*8
4x	Convolution	512	3*3	
	Convolution	512	3*3	
	Dense			8*8
	Convolution	512	1*1	
	Convolution	1024	3*3	
	Residual			8*8
	AvgPool		Global	
	Connected		1000	
	Softmax			

The extractor presented in this table is constructed from Dense block and Residual block.

Table 2. Proposed Model

	Mixed model	Model Construct		
		Filters	*Size*	*Output*
	Convolution	32	3*3	256*256
	Convolution	64	3*3/2	128*128
	Convolution	32	3*3	
	Convolution	32	5*5	
	Convolution	16	7*7	
	Concatenate			128*128
1×	Convolution	32	1*1	
	Convolution	64	3*3	
	Residual			128*128
	Convolution	128	3*3/2	64*64
	Convolution	64	3*3	
	Convolution	64	5*5	
	Convolution	32	7*7	
	Concatenate			64*64
2×	Convolution	64	1*1	
	Convolution	128	3*3	
	Residual			64*64
	Convolution	256	3*3/2	32*32
	Convolution	128	3*3	
	Convolution	128	5*5	
	Convolution	64	7*7	
	Concatenate			32*32
8×	Convolution	128	1*1	
	Convolution	256	3*3	
	Residual			32*32
	Convolution	512	3*3/2	16*16
	Convolution	256	3*3	
	Convolution	256	5*5	
	Convolution	128	7*7	
	Concatenate			16*16
8×	Convolution	256	1*1	
	Convolution	512	3*3	
	Residual			16*16
	Convolution	1024	3*3/2	8*8
	Convolution	512	3*3	
	Convolution	512	5*5	
	Convolution	256	7*7	
	Concatenate			8*8
4×	Convolution	512	1*1	
	Convolution	1024	3*3	
	Residual			8*8
	AvgPool		Global	
	Connected		1000	
	Softmax			

The extractor presented in this table is constructed from Inception block and Residual block.

3.4 Comparison with Traditional Filtering Restoration Methods

We put the image after the traditional filtering restoration into the model of the noiseless training set for identification and comparison, and the results are compared with our model respectively.

We adopt mean filtering (AMF) [18]:

$$f(x,y)=\frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t) \tag{13}$$

The size of the rectangular window in the formula is S_{xy}, The arithmetic mean is the average value of the disturbed image $g(s,t)$ in window S_{xy}.

Inverse filtering recovery [19]:

Assuming that $f(x,y)$ represents the ideal image input and $g(x,y)$ is the degraded image, In the frequency domain through Fourier transform we get that:

$$G(u,v)=F(u,v)H(u,v)+N(u,v) \tag{14}$$

For inverse filtering, the estimated formula is:

$$\widehat{F}(u,v)=G(u,v)+\frac{N(u,v)}{H(u,v)} \tag{15}$$

In order to avoid $H(u,v)$ too small, we add some restrictions in the inverse filtering and only restore in the finite neighborhood near the far point.

Wiener filtering [20]:

$$G(f)=\frac{1}{H(f)}+\left[\frac{|H(f)|^2}{|H(f)|^2+\frac{N(f)}{S(f)}}\right] \tag{16}$$

S and n represent the power spectrum of signal and noise respectively, which is difficult to get in normal times. Therefore, we replace them with K value.

Constrained least square method (CLS) [21]:

$$\mathrm{g}(x,y)=H\left[f(x,y)\right]+\eta(x,y) \tag{17}$$

The objective function is:

$$C=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\nabla^2 f(x,y)\right]^2 \tag{18}$$

Constraints:

$$\left\|g-H(\widehat{f})\right\|^2=\|\eta\|^2 \tag{19}$$

Finally, the minimum expression of F is:

$$\widehat{F}(u,v)=\left[\frac{H^*(u,v)}{|H(u,v)|^2+\gamma|P(u,v)|^2}\right]G(u,v) \tag{20}$$

4. Experiments

We can get many unexpected good results by expanding the training set., however, our computational resources are limited, therefore, we select 1000 more clear and first goal larger images

in network training. And then, we put in 400 to join the motion blur pictures for training, finally, we put in 500 to join the noise of images for training, observing model under the condition of the different training set about performance difference. For training purposes, we cropped the image to 416*416.

4.1 Training Details

Since Inception block structure is adopted, the number of introduced parameters is larger than the original, so we adopt the method of deep connection and reduce the number of convolution kernels for setting. Also, in order to optimize the structure of our network, we first network model based on the original training, is divided into two steps, the first step in the training of the underlying network first, because our resources are limited, at first all training, will produce very big loss. If using the same learning rate, greatly increase the training of our time. The second part, all the network parameters are set for the training, this step is training again after adjusting the network, therefore, is mainly a fine adjustment on network.

The platform we used was tensorflow-gpu1.10, and a Nvidia 1080Ti GPU. We trained 500 epochs to get this result, and it took almost a day to train a model of a particular level of recognition drone, and our datasets were basically from publicly available datasets on the Internet. By adding gaussian noise and motion blur to the original 1000 images, we obtained 1000 images respectively. ,400 and 500 images were added with gaussian noise and training blur respectively to compare the test effect of the model, and the remaining images were used to verify the model. By increasing the gaussian noise with a variance of 0.01 and increasing the motion blur with a displacement of 21 pixels.

4.2 The Training Results

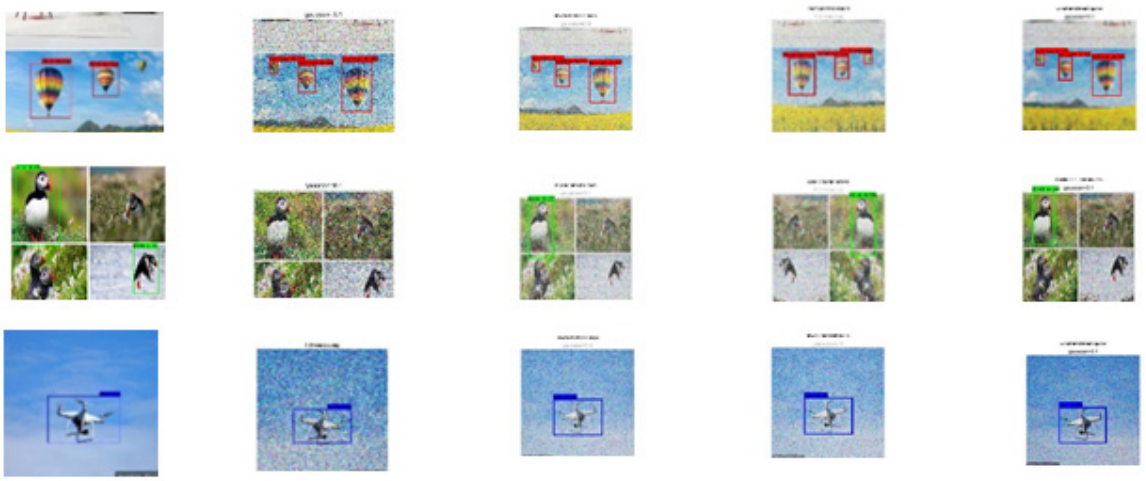
We first restore the image with the traditional method, and then test it with the traditional model whose training set without blur image

We used σ=0.1, or motion blur of displacement pixel 21.

The image restored after filtering is shown in the figure below:

(a)Original (b)Gaussian (c)Mean filtering (d)Inverse filtering (e)CLSFig

Fig 4. Filtered and Restored Images



(a)Original (b)Gaussian (c)Mean filtering (d)Inverse filtering (e)CLS

Fig 5. Use Traditional Model to Detect the Filtered Image Objects Respectively

It can be seen that the image recognition rate after processing is unstable. And no method can improve the detection accuracy on all kinds of objects. Test with the model of the same original picture, and the results are shown as follows:

1000 pictures were selected, among which 800 were trained, 100 were verified and 100 were tested. The training included 600 uav pictures, 150 birds and 50 balloons.

The model is selected as traditional model(Module use Residual block):

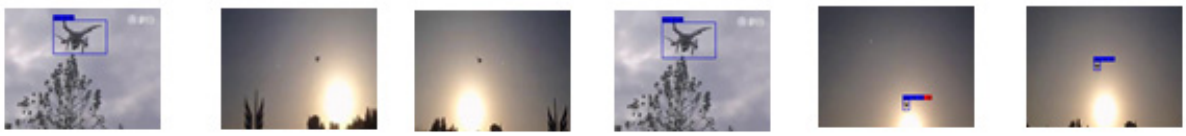


Fig 6. Proposed Model and Traditional Model are Used to Detect Objects Respectively in the Original Image

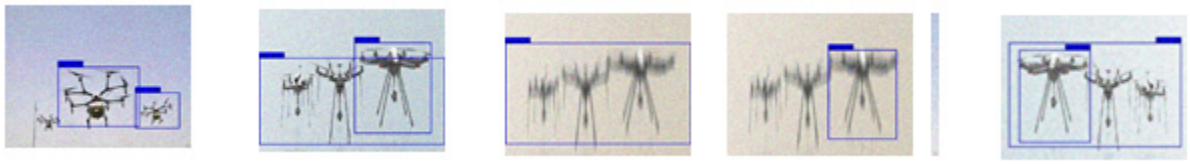


Fig 7. Proposed Model and Traditional Model Are Used to Detect Objects Respectively in the Gaussian Image

Fig 8. Proposed Model and Traditional Model are Used to Detect Objects Respectively in the Motion Blur Image

Table 3. AP(average percision)

Datasets	***classes***	***AMF***	***IHM***	***CLS***	***Winner***	***Original***
Original 1000	uav	0.25	0.34	0.37	-	0.31
	bird	0.09	0.13	0.11	-	0.12
	ballon	0.03	0.02	0.01	-	0.02
Motion bulr 1400	uav	-	-	-	0.16	-
	bird	-	-	-	0.1	-
	ballon	-	-	-	0.02	-

Arithemetic mean filter (AMF) Inverse harmonic mean (IHM) Constrained least squares (CLS) Winner filter (Winner)

Table 4. AP(average percision)

Datasets	***Model***	***Traditional model***	***Mixed model***	***proposed***
Original 1000	uav	0.34	0.40	0.42
	bird	0.13	0.21	0.15
	ballon	0.02	0.02	0.05
gaussian1400	uav	0.31	0.36	0.38
	bird	0.12	0.20	0.16
	ballon	0.02	0.01	0.03
Motion 1400	uav	0.29	0.37	0.40
	bird	0.09	0.12	0.14
	ballon	0.01	0.01	0.02

Test our data set against different models to see the effect of the test. We choose different datasets to test different models. We add gaussian noise to original images (Gaussian 1400). We add motion blur to original images (Motion 1400).

As can be seen from the table, the proposed method does not significantly change the depth, which is better than concatenate directly model. The recognition effect is better than our traditional recognition model, namely the model of Darknet[22]architecture.

5. Conclusion

In this paper, an effective model is designed to identify uav images with noise and motion blur. In particular, it is easy to train a deep and complex convolutional neural network with the help of skip connection and direct concatenation. The combination of a large number of deep learning skills, speed up the training process, improve our recognition rate. At the same time, this paper also makes a comparison between the traditional Darknet model and the model proposed by us, and constructs another model for deepening the network. Different from our model, it uses the combination of Dense block and Residual block. The experimental results also show that the recognition rate of Mixed model is higher than that of the traditional network. However, compared with our proposed method of using Inception block block and Residual block, our proposed model recognition performance is better because we expand the receiving field and can extract more features. However, we still have some work to be further studied. Firstly, because the number of pictures is too small, especially the pictures of our balloons, the recognition rate of our balloons is low. Secondly, in terms of image restoration, can we restore the image before recognition? Third, we add gaussian blur and motion blur, in the practical application there are more complex cases, research how to adapt to more complex cases of the identification problem is a very meaningful topic.

References

[1]. P Viola, M Jones. Rapid object detection using a boosted cascade of simple features[J]. Proc. IEEE CVPR 2001, 905-910, 2001.

[2]. Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005, 1: 886-893. (2016: Google Citation: 14046).

[3]. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350 Pedro F. Felzenszwalb; Ross B. Girshick; David McAllester; Deva Ramanan. Object Detection with Discriminatively Trained Part-Based Models[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence1627 - 1645.

[4]. S. Bascon, S. Arroyo, P. Jimenez, et al. Road-sign detection and recognition based on support vector machines[J]. IEEE Transactions on Intelligent Transportation Systems, 2007, 8(2): 264-278.

[5]. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[6]. Pavel Svoboda, Michal Hradis, Lukas Marsik, Pavel Zemcik.CNN for License Plate Motion Deblurring.2016.

[7]. Patrick Wieschollek, Michael Hirsch, Bernhard Scholkopf, Hendrik P. A. Lensch. Learning Blind Motion Deblurring. The IEEE International Conference on Computer Vision (ICCV), 2017, pp. 231-240.

[8]. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi; You Only Look Once: Unified, Real-Time Object Detection. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788.

[9]. J Redmon, A Farhadi Yolov3: An incremental improvement. Preprint arXiv:1804.02767, 1804.

[10]. Ross Girshick. Fast R-CNN. The IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448.

[11]. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788.

[12]. Wei Liu, Dragomir Anguelov, Dumitru, Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg European Conference on Computer VisionECCV 2016: Computer Vision – ECCV 2016 pp 21-37.

[13]. Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2117-2125.

[14]. Haykin S, Kosko B. Gradient based learning applied to document recognition. Wiley-IEEE Press, 2009; 86(11): 306–351.

[15]. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[16]. Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]//Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[17]. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[18]. Malek-Mohammadi M, Rojas C R, Wahlberg B. A class of nonconvex penalties preserving overall convexity in optimization-based mean filtering[J]. IEEE Transactions on Signal Processing, 2016, 64(24): 6650-6664.

[19]. Yumei W, Ping L, Jun Z. Correction of Field Curved Images by the Polynomial Approximation of the Inverse Filtering Function[J]. ACTA PHOTONICA SINICA, 2003, 32(6): 745-748.

[20]. Gardner W A. Cyclic Wiener filtering: theory and method[J]. IEEE Transactions on communications, 1993, 41(1): 151-163.

[21]. Kim H, Park H. Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method[J]. SIAM journal on matrix analysis and applications, 2008, 30(2): 713-730.

[22]. Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.