Int. J. Nonlinear Anal. Appl.
11 (2020) Special Issue, Winter and Spring 2020, No. 4, 53-61
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2020.4500

Investigation of the Effect of Noise on Tracking Objects using Deep Learning

Mohammad Eshaghian*

Department of Computer Engineering and Information Technology , Payame Noor University (PNU), P.O. Box, 19395-3697, Tehran, Iran.

(Communicated by Madjid Eshaghi Gordji)

Abstract

Nowadays, tracking objects has become one of the basic needs of security systems. Deep learning based methods has dramatically improved results in tracking objects. Meanwhile, the quality of the videos captured by camera is effective on the accuracy of the trackers. All images captured by camera inevitably contain noise. The noise is usually created due to various reasons such as the underlying media, weather condition, and camera vibrations in the wind and so on. This paper deals with the issue. In this paper, tracking objects is performed by Yolu 3 architecture in deep learning. Cycle spinning method is also employed to eliminate noise.

Keywords: Noise, Object Tracking, Deep learning, Wavelet transform, Cycle spinning.
2010 MSC: 76T20

1. Introduction

Increasing public and personal camera surveillance, controlling and tracking objects and people is a challenging task. Therefore, tracking moving objects using effective methods has attracted lots of attention. Different methods in tracking objects are introduced, from which the network of artificial neurons is the most important. In this regard, a deep learning architecture is trained based on training data and then objects are tracked in videos. The architecture used in this paper in order to predict objects is Yolo3. The model is trained on Coco. The data are in 80 classes of different objects including humans, animals such as cats, dogs, and so on.
Noise is actually often neglected in images and tracking. The noise in images can have different causes, but have negative effect on performance in any condition. This issue is considerably obvious in Fig. 1.

*Corresponding Author: Mohammad Eshaghian
Email address: m_eshaghian@pnu.ac.ir (Mohammad Eshaghian*)

Received: January 2020 *Revised:* May 2020

Figure 1: Tracking accuracy in noisy and noise-free images

As shown in Fig. 1, the accuracy of tracking has fallen dramatically. Given this, one could conclude that a preprocessing step in tracking objects can greatly increase the accuracy of the tracking by eliminating the potential noise. In this paper, two main issues are addressed: object tracking and noise elimination.
Rest of the paper is organized as follows: In this section, the related literature is reviewed. In Section 2, added noise is discussed. In section 3, the proposed approach is presented, and finally in section 4, the results of the experiments are presented.

2. Literature review

In this section, a set of noise removal methods along with a number of recent researches in the field of object tracking are introduced.
Lots of noise removal methods as well as image quality enhancement methods are presented. Most techniques are performed using the values of the neighbors; such as average value filter in which neighboring pixels are selected, and the center element of the neighborhood takes the average value. The median value filter is performed the same, but the difference is that the median value of a window is placed in the center element. The main issue in such techniques is the storage of original frame or photo information. There are other techniques in which a deep learning architecture takes the pixels of a noisy photo and outputs a photo with the same size as the input from which the noise is removed. An example of this is given in [1]. This technique predicts the degraded values of the image using artificial neural networks and affects the output. The method used in this paper is called the cycle spinning. A signal is generated from a photo using a wavelet transform, then the image is reconstructed with rotational shifts and the noise is eliminated. This method has three basic steps, which includes wavelet transform of the noisy photo, finding the hard or soft threshold, and at last signal conversion. In the first step, a white mask of noise is randomly selected from the pixels of each frame, is applied to the entire frame. Then, wavelet transform is applied, subsequently the cycle spinning with optimal shift values is applied to them. The results show that the method not only increases the accuracy of detection and classification of objects, but also identifies more objects [2-4]. In [5], deep neural networks is used to track objects, focusing on leveraging existing data to efficiently track objects. Meanwhile, similarity learning has been introduced. In [6], a deep neural network is introduced that employs probability distribution to capture video frames and directly compute the tracking accuracy. In [7], focus is taken on regions detected in traceability. This paper shows that different regions can have different detection accuracy. Therefore, efforts have been made to select the best region based on the tracking accuracy. In [8], it is shown that neural networks can be used as a tool to extract features for object tracking. By investigating the different designs of the

neural network layers for feature extraction, an optimal design for tracking objects is achieved. In [9], overfitting is prevented using ensemble methods added to pre-trained models, then an increase in model performance would be achieved. In [10], a dual SVM-based methods is proposed to increase the speed of tracking objects, which have a positive effect on online tracking.

3. Preliminaries

3.1. Noise Addition

To make the content understandable, noise is added by percentage. By randomly selecting a specific percentage of the total pixels of the frame and whitening the pixels in the frame. This means that the value of the gray pixels are set to 255. At last, a frame containing white noise is produced. An example of the noisy frames is shown in Fig. 2. Fig. 2 (A) shows the original frame, and Fig. 2 (B) shows the same image with 25% white noise.

Figure 2: Noise Addition

3.2. YOLO Algorithm

YOLO algorithm gives a much better performance on all the parameters with a high fps for real-time usage. YOLO algorithm is an algorithm based on regression, instead of selecting the interesting part of an Image, it predicts classes and bounding boxes for the whole image in one run of the Algorithm. To understand the YOLO algorithm, first we need to understand what is actually being predicted. Ultimately, we aim to predict a class of an object and the bounding box specifying object location. Each bounding box can be described using four descriptors:

1. Center of the box (bx, by)
2. Width (bw)
3. Height (bh)
4. Value c corresponding to the class of an object

Along with that we predict a real number pc, which is the probability that there is an object in the bounding box. YOLO doesn't search for interested regions in the input image that could contain an object, instead it splits the image into cells, typically 19x19 grid. Each cell is then responsible for predicting K bounding boxes. An Object is considered to lie in a specific cell only if the center co-ordinates of the anchor box lie in that cell. Due to this property the center co-ordinates are always calculated relative to the cell whereas the height and width are calculated relative to the whole Image size. During the one pass of forwards propagation, YOLO determines the probability that the cell contains a certain class. p_c denotes the probability that there is an object of certain class 'c'.

$$\text{score}_{c,i} = p_c \times c_i \tag{3.1}$$

The class with the maximum probability is chosen and assigned to that particular grid cell. Similar process happens for all the grid cells present in the image. After predicting the class probabilities, the next step is Non-max suppression, it helps the algorithm to get rid of the unnecessary anchor boxes. There are numerous anchor boxes calculated based on the class probabilities. To resolve this problem Non-max suppression eliminates the bounding boxes that are very close by preforming the IoU (Intersection over Union) with the one having the highest class probability among them.

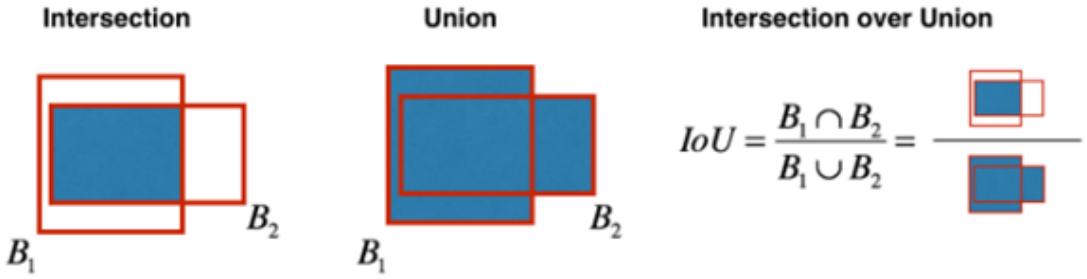

Figure 3: Intersection over Union

It calculates the value of IoU for all the bounding boxes respective to the one having the highest class probability, it then rejects the bounding boxes whose value of IoU is greater than a threshold. It signifies that those two bounding boxes are covering the same object but the other one has a low probability for the same, thus it is eliminated. Once done, algorithm finds the bounding box with next highest class probabilities and does the same process, it is done until we are left with all the different bounding boxes. After this, almost all of our work is done, the algorithm finally outputs the required vector showing the details of the bounding box of the respective class. The overall architecture of the algorithm can be viewed below:

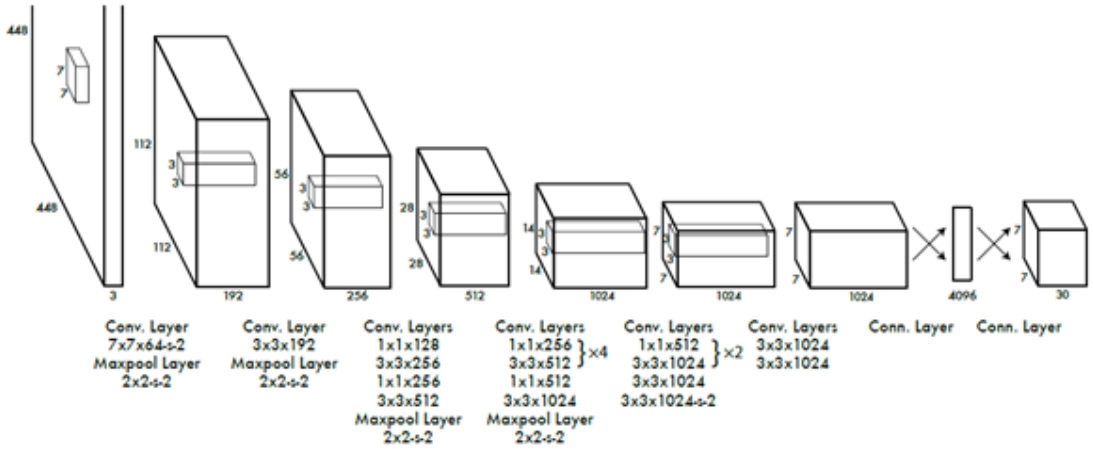

Figure 4: YOLO Architecture

Therefore, the most important parameter of the Algorithm, its Loss function is shown below. YOLO simultaneously learns about all the four parameters it predicts. This was all about the YOLO Algorithm. We discussed all the aspects of Object detection along with the challenges we face in that domain. We then saw some of the algorithms that tried to solve some of these challenges but were failing in the most crucial one-Real time detection (speed in fps). We then studied the YOLO algorithm which outperforms all the other models in terms of the challenges faced, its fast-can work well in real-time object detection, follows a regression approach. Still improvements are being made

in the algorithm. We currently have four generations of the YOLO Algorithm from v1 to v4, along with a slightly small version of it YOLO-tiny, it is specifically designed to achieve an incredibly high speed of 220 fps.

4. Proposed approach

As mentioned earlier, Yolo3 architecture is used to detect and track objects. To speed up the process, weights are previously trained on Cocoa dataset. The architecture was introduced by Microsoft and is now known as the model that can be used to track real-world objects. This model is used with different architectures. The architecture used in this paper has 75 layers of convolution. This architecture is discussed further in [11].
Noise removal is performed in two basic steps of wavelet transform and rotating shift, which are described below.
Step 1: Wavelet tranform
Assume that X is the input frame and N is the noise. Then the noisy frame is denoted by Y and is equal to:

$$Y = X + N \tag{4.1}$$

Then the recovered image as is denoted by X'. According to the proposed method in [2-4], wavelet transform of noisy image is performed in three stages. 1. Wavelet transform of noisy photo. 2. Finding the soft or hard threshold of the resulting signal 3. Apply two-dimensional discrete image conversion to obtain noise-free image. If wavelet transform is denoted by W and the threshold function with $\eta(.)$, then the noise-free image will be as follows:

$$\hat{x} = w^{-1}(\eta(W(y))) \tag{4.2}$$

where eta is the threshold function that can be soft or hard. The hardness of the threshold is affected by the amount of detail retention at the edges and other parts of the image.
Step 2: Cycle spinning
The algorithm has the ability to detect unknown signals based on successive shifts and then find the average value of the results linearly.
If we consider the shift of an image to be $S_{i,\ j}$, and the value of the wavelet transform is denoted by W, then the rotational shift will be equal to:

$$\hat{y} = \frac{1}{k_1 k_2} \sum_{i=1,j=1}^{k_1,k_2} S_{-i,-j} \left(W^{-1} \left(\eta \left(W \left(S_{i,j}(y)\right)\right)\right)\right) \tag{4.3}$$

where k_1 and k_2 are the maximum shifts. Finding the maximum optimal values is mostly based on trial and error. However, some methods have been proposed to compute it [12]. It should be noted that the sigma in the above equation depends on the values of k_1 and k_2, and increasing the values would slow down the shift operation. Therefore, in this paper, an optimal number is .
The general process of the approach is shown in Fig. 5:

5. Experimental results

As mentioned earlier, an attempt is made to obtain more accuracy by considering the computational speed. According to the explanation given in the third section of the paper, Yolo version three is used to perform the experiments. To increase efficiency and accuracy, as well as obtaining quick

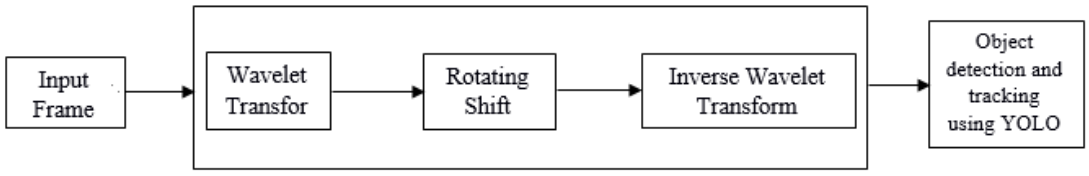

Figure 5: The proposed approach

results, a previously trained model is employed. This model is trained on Cocoa data introduced by Microsoft. This data set belongs to real-world eighty different classes.
In the first step, the video frames are extracted, then a white mask of noise is added to the frames by percentage and randomly. Comparison is made to understand the effect of noise and its removal from frames. The accuracy of object detection is determined by the percentage with the name of the object in all frames. If the probability of object detection is less than 50%, they are not displayed in the figures. Fig. 6 shows that by adding more noise, some objects cannot be detected by the proposed model. The accuracy of the detection is also reduced. By adding 40% of noise, object detection and tracking would reach 0 by the model. In Fig. 6, three consecutive frames are selected from a video and 25% noise is applied to them, then objects inside the frames are detected (Fig. 6- a, b, c). Then, on the same previous frames, the noise level is increased to 40% and attempts are made to detect objects using the presented architecture (Fig. 6- d, e, f).

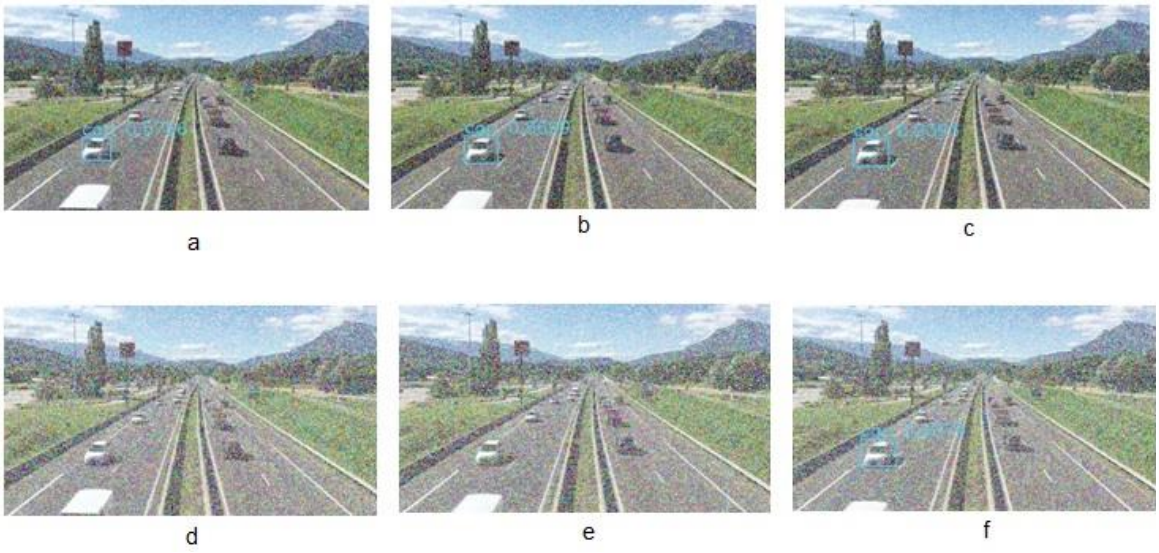

Figure 6: The effect of noise on object detection

Fig. 7 shows the number of detectable vehicles in each frame. The horizontal axis shows the frame number and the vertical axis shows the number of vehicles. 1500 frames have been used to draw this diagram. As can be seen from the figure above, increasing noise from 15% to 40% can have a significant effect on tracking objects in video.
Fig. 8 shows the 1500 frames. The diagram shows the average accuracy of detecting tracked cars in percentage for 15 and 40 percentage of noise levels. First, the accuracy of the vehicle detection in percentage is summed and then divided by the number of vehicles. It is clear that noise addition by 40% has a negative effect on vehicle tracking compared to noise addition by 15%, and most of the diagrams remain at zero.

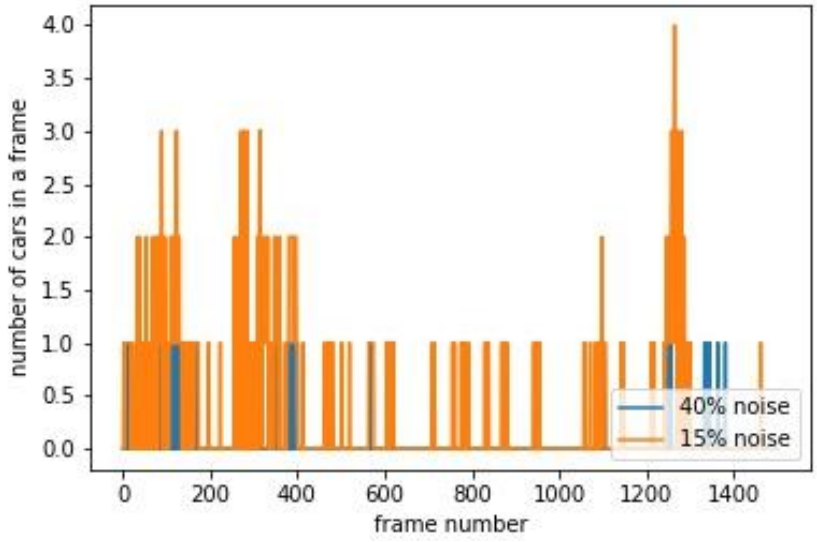

Figure 7: Comparison of the number of vehicles detected for 15and 40 percent applied noise

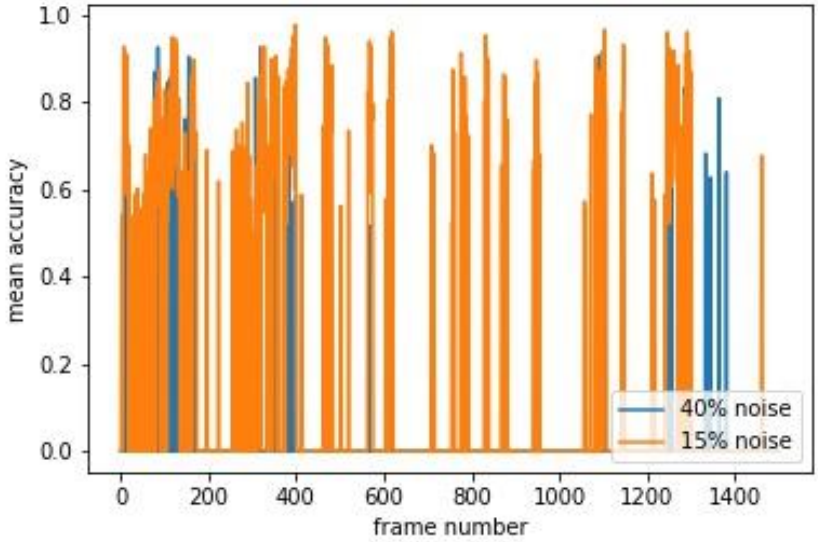

Figure 8: Comparison of object detection accuracy for 15 and 40 % noise

Once all the noisy frames of the video have been traversed, noise removal operation starts. In this step, wavelet transform is performed. The rotational shift then completes the noise removal operation. An example of the de-noised frames are shown in Fig. 9.
As mentioned earlier, fixed frames are used to compare the results. As can be seen, the proposed method has not only increased the accuracy of object detection, but has also been able to detect more objects. Fig. 9 (a) shows a sample frame. Added noise to image is 25%. In this case, vehicles can be detected by human eyes, while the system detect it as a train! After the noise removal operation, Fig. 9 (c) is obtained. The effect of the proposed method can be observerd, according to the figure. All detected objects are assigned to the associated class.
Fig. 9 (d) is also obtained after noise from Fig. 9 (c). In (c), only one vehicle can be detected. While in (d), more vehicles with acceptable percentages are detected and tracked. This is very important in self-driving cars. Because if the algorithm is not able to detect the surrounding vehicles, it can cause an accident.

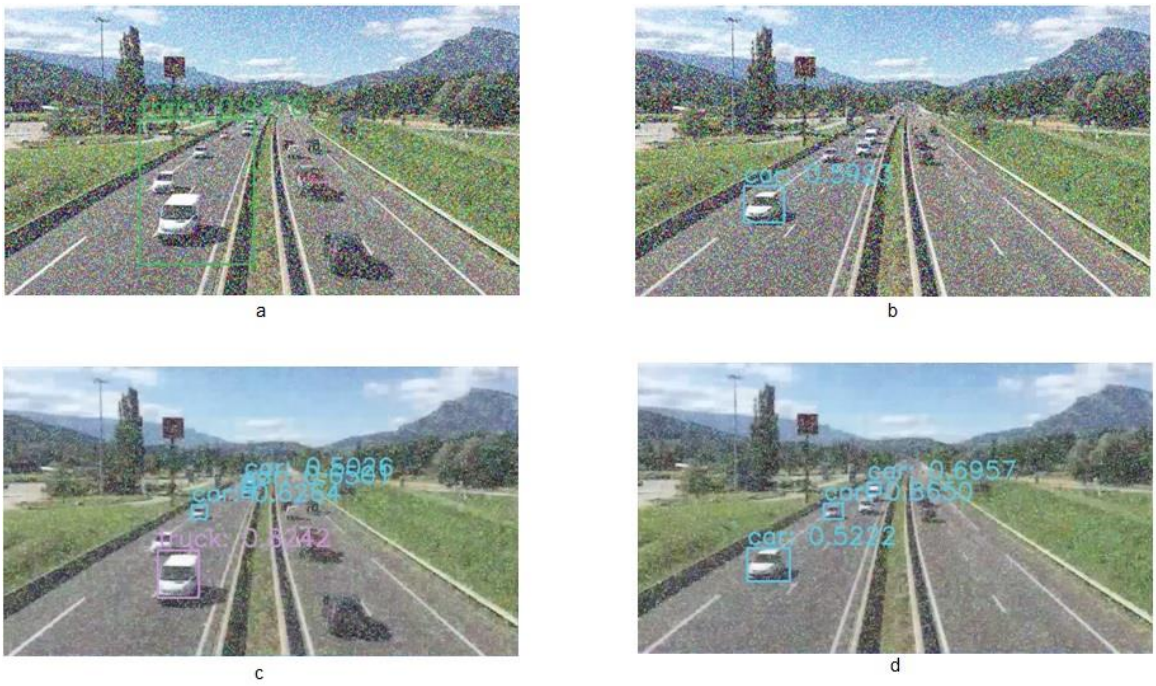

Figure 9: Elimination of noise and its effect on detection accuracy

In the following, two criteria of accuracy and recovery have been used to measure the efficiency of the system provided. The definitions of these two criteria are as follows:

6. Conclusion

In this paper, the effect of white noise on the error rate of neural network in object detection is investigated. By increasing noise, it is observed that the proposed model is not able to detect any object and in most cases the accuracy of detection is reduced. Furthermore, the quality of the frames is increased using the cycle spinning, and the effect on increasing the accuracy of object tracking is observed. The cycle spinning is one of the most efficient methods because it keeps more information in the frames, including the edges.

References

[1] Sudipta Singha Roy , Mahtab Ahmed and Muhammad Aminul Haque Akhand . Noisy image classification using hybrid deep learning methods.
[2] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," Biometrika, vol. 81, pp. 422–455, 1994.
[3] D. Donoho, "De–noising by Soft–Threshold," IEEE Trans. On Information Theory, vol. 41, pp. 613–627, 1995.
[4] I. M. Johnstone and D. L. Donoho, "Adapting to Smoothness via Wavelet Shrinkage," J. Statistical Association, vol. 90, no. 432, pp. 1200-1224, 1995.
[5] Luca Bertinetto , Jack Valmadre , Jo ao F. Henriques , Andrea Vedaldi and Philip H. S. Torr . Fully-Convolutional Siamese Networks for Object Tracking.
[6] Mengyao Zhai , Mehrsan Javan Roshtkhari, Greg Mori . Deep Learning of Appearance Models for Online Object Tracking.

[7] Shuchao Pang , Juan José del Coz , Zhezhou Yu , Oscar Luaces and Jorge Dı´ez .Deep Learning and Preference Learning for Object Tracking: A combined approach.
[8] Lijun Wang, Wanli Ouyang, Xiaogang Wang and Huchuan Lu .Visual Tracking with Fully Convolutional Networks.
[9] Lijun Wang, Wanli Ouyang, Xiaogang Wang and Huchuan Lu .STCT: Sequentially Training Convolutional Networks for Visual Tracking.
[10] Jifeng Ning , Jimei Yang , Shaojie Jiang , Lei Zhang and Ming-Hsuan Yang . Object Tracking via Dual Linear Structured SVM and Explicit Feature Map.
[11] Tsung-Yi Lin , Michael Maire , Serge Belongie m , Lubomir Bourdev , Ross Girshick , James Hays , Pietro Perona , Deva Ramanan , C. Lawrence Zitnick and Piotr Dollár . Microsoft COCO: Common Objects in Context.
[12] S. M. E. Sahraeian, F. Marvasti and N. Sadati, "Wavelet image denoising based on improved thresholding neural network and cycle spinning," to appear in ICASSP2007.
[13] Samuel Dodge and Lina Karam . Understanding How Image Quality Affects Deep Neural Networks.