A Real-Time Two-Stage Detector for Static Monitor Using GMM for Region Proposal

Yingping Liang[1], Yunfei Ma[3], Zhengliang Wu[1], and Mingfeng Lu[1,2(✉)]

[1] Beijing Institute of Technology, Beijing 100081, China
lumingfeng@bit.edu.cn
[2] Beijing Key Laboratory of Fractional Signals and Systems, Beijing Institute of Technology, Beijing 100081, China
[3] Zaozhuang University, Shandong 277100, China

Abstract. CNN-based object detectors have been widely exploited for vision tasks. However, for specific real-time tasks (e.g. object detection on static monitor), the enormous computation cost makes it difficult to work. To reduce the computation cost for object detection on static monitor while inheriting high accuracy of CNN-based networks, this paper proposals a method with a two-stage detector using Gaussian mixture model for region proposal. We test our method on MOT16 datasets. Compared with original models, the two-stage detectors equipped with Gaussian region proposal achieve a better performance with the mAP increased by 0.20. We also design and train a light-weight detector based on our method, which is much faster and more suitable for mobile and embedded device with little drop in accuracy.

Keywords: Deep learning · Computer vision · Intelligent monitoring

1 Introduction

VID (object detection from video) has become a challenge task in recent years. Compared to image object detection, static videos are highly redundant, containing a large amount of temporal locality (that is, similar at different times) and spatial locality (that is, they look similar in different scenes), and frame by frame processing is time-consuming and computationally expensive. So making full use of the timing context can solve the problem of a large amount of redundancy between consecutive frames in the video and improve the detection speed and the detection quality. Although deep learning methods excel on some very large datasets for image detection, there is still a big gap for the application of specific tasks.

The one-stage method (e.g. SSD [1], YOLOv3 [2]) has achieved high efficiency by densely sampling on feature maps over different scales and ratios but suffers from low accuracy; the two-stage method (e.g. Faster-RCNN [3], FPN [4]) has achieved high accuracy by using two-stage structures to describe features and regress bounding box

Beijing Natural Science Foundation (L191004).

S. Shi et al. (Eds.): AICON 2020, LNICST 356, pp. 414–423, 2021.
https://doi.org/10.1007/978-3-030-69066-3_36

parameters but suffers from low efficiency. And one of the key points to improve the detection speed of two-stage methods is to improve the structure for extracting regions of interest.

Traditional methods [5, 6] for video tracking have fewer parameters and low latency, suitable for mobile and embedded vision applications, but lack the ability to recognize categories and have some other problems (see Fig. 4). Some CNN methods utilize optical flow vectors for local area to detect moving target, which is heavily influenced by background. For methods using Convolutional LSTM [7] Network, in order to integrate features across time, ConvLSTM is used to obtain temporal information, which is computationally expensive. The mainstream idea is to combine the context information and tracking information between frames. However, lack of datasets is one of the obstacles for training an end-to-end detector for applications.

It inspires us to think: can we use an efficient mixture model for region proposal and an lightweight convolutional neural network trained on image detection datasets for classification and cascade regression?

Thus we use a pixel to pixel rather than region to region approach by an efficient adaptive Gaussian mixture model for the static video background subtraction. The mixture model presents a set of N models for each pixel. As the pixel value at each point follows a separate non-stationary temporal distribution, the mixture model generates the parameters using an adaptive learning rate for each model at every frame and selects the number of components (N) simultaneously [8]. The upgrade process is achieved by a variant of the maximum a posterior (MAP) solution. While updating, the Gaussian mixture model also produces the region of interest for moving objects as the foreground, which is highly memory and time efficient.

In the CNN part, we follow the framework of recent two-stage detection network with an extra Context Enhancement Module [9] to combine global context with local information and use a lightweight mobilenet as backbone. In the mixture model, we have got the separation of foreground and background and coarse boxes for locations. Thus the RPN module for filtering and adjusting anchors in some modules can be abandoned in the detection part.

In the next section we will review some of the prior work in background subtraction and video object detection. Section 3 describes the upgrade algorithm for Gaussian mixture model and the structure of CNN-based detection part. Section 4 shows the training details and Sect. 5 describes the test results on the MOT dataset without extra training on the training video.

2 Related Work

2.1 Moving Object Tracking

A moving object can be detected if the background of the video is known. To build the model of the background, a well-known process is to use a Gaussian mixture model (GMM) to estimate the probability distribution of pixels in real time [5]. Thus the foreground is detected by observing the local regions which do not fit the distributions. The standard equation for GMM estimation uses a fixed number of components at each pixel.

In [6], a effective recursive unsupervised learning method of finite mixture models is proposed. This method suggests an efficient method for maximizing probability distribution and parameter estimation. By using a prior as a bias for maximally mixture models, the method enables us to estimate the parameter and select the number of components. This method is also applicable to Gaussian mixture models in version tasks, which is much more efficient than RPN method for region proposal.

2.2 CNN-Based Detectors

CNN-based two-stage detectors gains excellent performance on large datasets in recent years. And the two-step methods have a more accurate detection than one-step method, by producing more features to describe object and two-step cascade regression. R-CNN uses selective search method for searching candidate region. Faster-RCNN [3] uses Region Proposal Network (RPN) to generate region of interest. And R-FCN [11] designs a fully convolutional structure and a position sensitive RoI pooling module for region proposal. And in the detection part, an extra subnet is usually added to achieve more accurate regression of object boxes and prediction of object categories.

There are also some real-time detectors with less parameters. Light-Head R-CNN [10] designs a lightweight detection network for lower latency and less computation cost, but still reserves the problems of mismatch between a fast backbone and a computational expensive detection network. SSDLite for mobile device adopts the MobileNet as backbone and a one-stage method similar to SSD. By widely using depth separable convolution with residuals, SSDLite generates a larger receptive field and reduces the computation cost but makes poor performance on accuracy. ThunderNet [9] utilizes DWConv to reduce the computation complexity in RPN net and a Special Attention Module (SAM) to enhance the foreground feature.

Fully utilizing the context between frames is also one of the keys for the improvement of state of the art detectors on video tasks. As superior models perform well on GPUs, their performance on mobile and other resource-restrained platforms will be greatly reduced. However, few work focuses on a particular application for target detection in static surveillance video. In this paper, we propose a method combining moving object tracking method and CNN-based detector for the surveillance of static scene.

3 Approach

3.1 Moving Region Proposal

The basic algorithm of Moving Region Proposal follows the formulation of a recursive unsupervised learning method of finite mixture model (see Fig. 1). For every frame in the video, the background frame can not be directly captured as there appears foreground beyond the background. And the background is not always the same on time series (e.g. A car can be either the foreground while moving or a part of background).

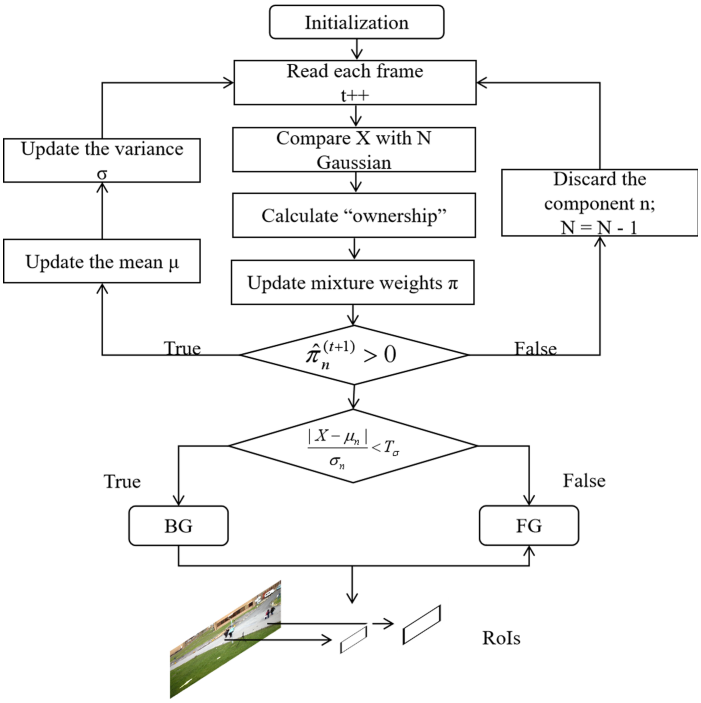

Fig. 1. The program frame and flow chart of moving region proposal.

A Gaussian mixture model with N is defined as:

$$\vec{\theta} = \{\pi_1, \pi_2, \ldots, \pi_N, \vec{\theta}_1, \vec{\theta}_2, \ldots, \vec{\theta}_N\} \tag{1}$$

where π_n is the weight of the n th component in mixture model and $\vec{\theta}_n = \{\mu_n, \sigma_n\}$ is the mean value and standard deviation.

Suppose a set of sampled data X and the vector $\vec{Y} = [y_1, y_2, \ldots, y_N]^T$ with only 0 and 1, which stands for the subordinate component of data X, the joint probability density is:

$$p(X, \vec{Y}; \vec{\theta}) = \prod_{n=1}^{N} \{\pi_n p_n(X; \vec{\theta}_n)\}^{y_n} \tag{2}$$

Thus when a new sample of pixel value is added at iteration *t+1*, we get the update equation for estimation of π, μ and σ:

$$\hat{\pi}_n^{(t+1)} = \hat{\pi}_n^{(t)} + (1+t)^{-1}\left(\frac{o_n^{(t)}(X^{(t+1)})}{1 - Nc_T} - \hat{\pi}_n^{(t)}\right) - (1+t)^{-1}\frac{c_T}{1 - Nc_T} \tag{3}$$

$$\hat{\mu}_n^{(t+1)} = \hat{\mu}_n^{(t)} + (1+t)^{-1} \frac{o_n^{(t)}(X^{(t+1)})}{\hat{\pi}_n^{(t)}} (X^{(t+1)} - \hat{\mu}_n^{(t)}) \tag{4}$$

$$\hat{d}_n^{(t+1)} = \hat{d}_n^{(t)} + (1+t)^{-1} \frac{o_n^{(t)}(X^{(t+1)})}{\hat{\pi}_n^{(t)}} ((X^{(t+1)} - \hat{\mu}_n^{(t)}) (X^{(t+1)} - \hat{\mu}_n^{(t)})^T - \hat{d}_n^{(t+1)}), d = \sigma^2 \tag{5}$$

For the task of background subtraction, we have a fix learning rate α to replace the $(1+t)^{-1}$. We also have $c_T = \alpha M/2$ where M stands for the number of parameters per component of the mixture model. See [5] for more details.

Compared with CNN-based two-stage region proposal methods, the Gaussian mixture model has several advantages as follows: (1) having less computational cost; (2) using an adaptive probability distribution to describe background features which is sensitive to background change.; (3) releasing RPN which enables more gradients flow from high-level feature for classification and more accurate regression.

We also set a series of expansion rates to generate mutil-scales RoIs for a better and more stable detection while do not necessarily increase the latency. In order to better utilize the continuous characteristics of videos, we use a simple method for predicting the possible RoIs in the next frame. Suppose the center of RoIs in the previous frame are and the current ones are, we simply predict the next by keeping the same vector difference.

3.2 Light-Weight Detector

To accurate the detection speed and decrease the storage and computation load, we adopt an input with a minimum side length of 300 pixels on region only with mask. As the Gaussian mixture model is sensitive to the moving region, a smaller backbone have little effect on the accuracy of the model while reducing the computational complexity. The main task of the detector is to adjust the coordinates of candidate areas and identify the classification, solving the problems of detection mismatch for local region in Gaussian mixture model (see Fig. 2).

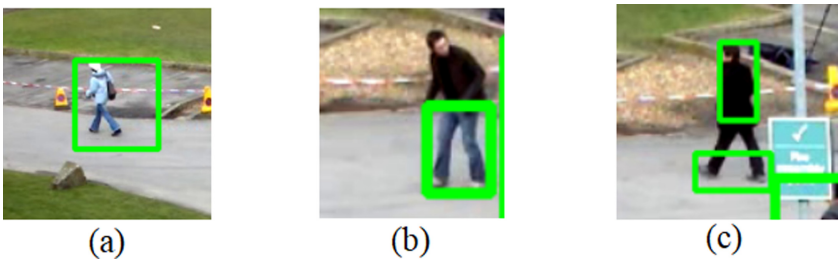

Fig. 2. Some problems in moving objection tracking by GMM method: (a) Expansion of detection frame caused by residual image of moving object. (b) Mismatch for some local statics regions. (c) Discontinuous detection for a single target.

We build a light-weight CNN backbone from MobileNets [12] for its superior performance on mobile and embedded device. By widely using depth-wise separate convolution, the model makes a further compression of the parameters.Generally, a larger receptive field is able to capture more context information which is essential for regression of coordinate values. Thus we set the initial convolutions be 7 × 7 convolutions with 64 channels.

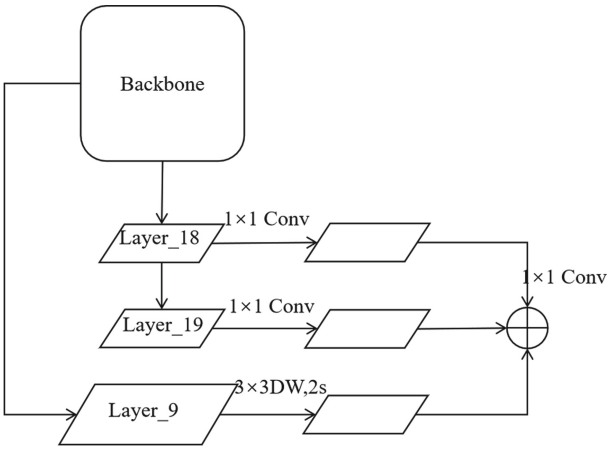

Fig. 3. Context enhance module for mixing up low-level and high-level features.

Besides, low-level features play an import role in regression while high-level features are significant to classification. To better combine the high-level features with low-level features, we design and optimize the Context Enhance Module (see Fig. 3). The key idea of CEM is to merge multi-level context and further enlarge the receptive field. Based on the MobileNet, we preserve the layer 7 as the low-level feature map and use a 3 × 3 depth wise convolutions and a point wise convolutions to enlarge receptive field and expand channels. For layer 12 and 18, we use a point wise convolution to compress the channels to 128. By stacking these there layers, the base feature map for classification and regression comes to a better balance between low-level and high-level features.

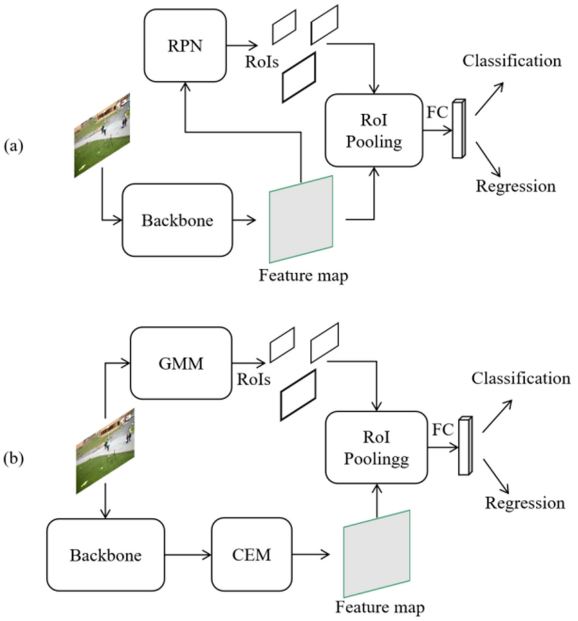

Fig. 4. (a) shows a typical two-stage detector using RPN for region proposal. (b) shows a two-stage detector with mixture model, using our method for optimization in static tasks.

3.3 Classifier and Regressor

A typical two-stage objection detection model uses RPN for region proposal (see Fig. 4). Compared with the RPN with a pre-trained model, we utilize the detection results from mixture model which is more adaptive and sensitive to the foreground. As the coordinate values of RoIs are relatively more accurate and require less adjustment, extra gradients is supposed to flow from the classification module. Thus we also present two adaptive hyper-parameters ω, ω' to estimate the learning weights of classification module and regression module in training losses. Assuming the number of epoches is t, the ω, ω' is defined as follow:

$$\begin{aligned} \omega &= 1 - 0.5 \times e^{-t/T} \\ \omega' &= 0.5 \times e^{-t/T} \end{aligned} \tag{6}$$

For train the classification, we assign a softmax label with a 0.5 weight for background and a 2.0 weight for objects in loss function. For the regerssor we use smooth L1 loss.

4 Implement Details

The moving region proposal using Gaussian mixture model has the same function with RPN, providing regions of interest and making a rough regression forecast. Therefore, we can train our model on conventional object detection datasets (e.g. VOC2012, COCO). Different scales and ratios are used to generate anchors while training. To put weights more on the CNN-based detector for classification and the second regression, we use the hyper-parameter ω to balance the losses while training. As the first regression for roughly adjusting RoIs while training can be achieved by Gaussian mixture model, we also set the weights of RPN losses (both RPN score loss and RPN regression loss) finally converge to a low value by $\omega' = 1 - \omega$. This method generates different RoIs with adjusted scales and ratios, which is closer to scene in reality (Fig. 5).

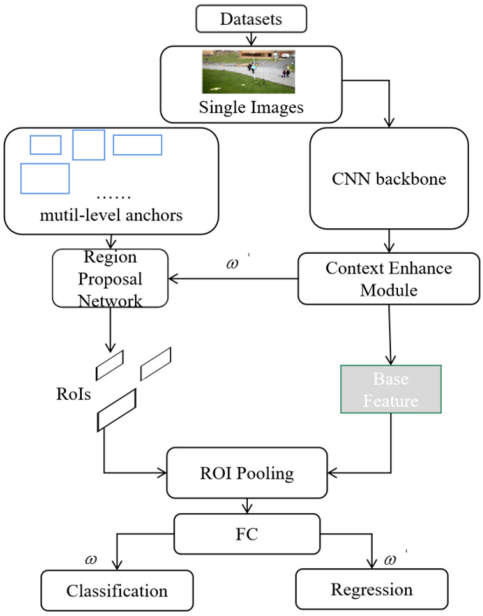

Fig. 5. The framework while training using the method similar to two-stage network. The value next to the arrow line represents the weight in training losses.

Our models are trained on VOC2012 using SGD with a momentum of 0.9. For anchors, we use 4 scales of 32,64,128, and 256, and 3 ratios of 1:1, 1:2, 2:1. By default we set the positive threshold 0.6 and the negative threshold 0.3. Note that unlike large detectors, we use less regularization such as weight decay and random dropout. As mobilenet and the light-weight subnet are small enough that there are few parameters and have less trouble with overfitting. Too much regularization may make it hard to converge. We also use the pre-trained MoblieNet on ImageNet to help the model

converge faster. The network is trained for 40K iterations on VOC2012 dataset. The learning rate drops from initialized 0.001 to 0.0001 by a factor of 0.1 at 30K iterations.

5 Experiments

The MOT15 dataset [13] consists of a set of videos labeled with track boxes. 5 of 11 are static videos which are suitable for the performance testing of out model. The results of accuracy and stability are shown in Table 1.

Table 1. Performance of backbone on complexity, comparison for different networks.

Method	Backboke	FPS	mAP
Faster-RCNN	VGG16	7.0	0.4083
Ours	VGG16	7.4	**0.6156**
Ours*	MobileNet	**21.4**	0.5737

We use method as a drop-in replacement for the Region Proposal Network and backbone in Faster-RNN. This model performs better than the original model in static monitoring tasks with a batter accuracy and a faster rate. We first compared the performance of GMM for region proposal by a raw Faster-RCNN and a Faster-RCNN with GMM. The latter is 0.4 FPS faster than the former with a superior increase of mAP.

We also tested our model combined MobileNet while using GMM for region proposal. By widely using depth-wise convolutional layers in backbone, our model is almost 200% faster than Faster-RCNN with VGG and have much less parameters with little decrease in accuracy. The results is shown in Table 1.

6 Conclusion

In this paper we investigate the effectiveness of read-time detectors in object detection on static monitors and propose a lightweight two-stage method with Gaussian Mixture Model for region proposal and a CNN-based detector. In the backbone, we utilize the MobileNet for its superior performance on embedded device and low latency. For the detection part, we propose a Context Enhance Module to enlarge the receptive field and merge low-level feature and high-level feature. For RoIs, we abandon the Region Proposal Network which is computational expensive and utilize the online training GMM for region proposal, making the detector practical for real-time applications.

References

1. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, Bastian, Matas, Jiri, Sebe, Nicu, Welling, Max (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
2. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. arXiv:1804.02767 (2018)
3. Ren, S., He, K., Girshick, R., et al.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
4. Lin, T.Y., Dollár, P., Girshick, R., et al.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125 (2017)
5. Lee, D.-S.: Effective gaussian mixture learning for video background subtraction. IEEE Trans. Pattern Anal. Mach. Intell. **27**(5), 827–832 (2005)
6. Zivkovic, Z., van der Heijden, F.: Recursive unsupervised learning of finite mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **26**(5), 651–656 (2004)
7. Chen, X., Wu, Z., Yu, J.: TSSD: temporal single-shot detector based on attention and LSTM for robotic intelligent perception (2018)
8. Zivkovic, Z., van der Heijden, F.: Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recogn. Lett. **27**(7), 773–780 (2006)
9. Qin, Z., Li, Z., Zhang, Z.: ThunderNet: Towards Real-time Generic Object Detection (2019). https://arxiv.org/pdf/1903.11752.pdf
10. Li, Z., Peng, C., Yu, G., et al.: Light-head R-CNN: in defense of two-stage object detector. arXiv preprint arXiv:1711.07264 (2017)
11. Dai, J., Li, Y., He, K., et al.: R-FCN: object detection via region-based fully convolutional networks. In: Advances in Neural Information Processing Systems, pp. 379–387 (2016)
12. Howard, A.G., Zhu, M., Chen, B., et al.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
13. Milan, A., Leal-Taixé, L., Reid, I., et al.: MOT16: a benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)