

PAPER • OPEN ACCESS

Control and preprocessing of graphic data for effective dynamic object recognition

To cite this article: O A Pakhomova and O Ja Kravets 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **537** 052002

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the **collection** - download the first chapter of every title for free.

Control and preprocessing of graphic data for effective dynamic object recognition

O A Pakhomova and O Ja Kravets*

Voronezh State Technical University, Voronezh, Russia

* E-mail: csit@bk.ru

Abstract. The features of detecting static objects are considered. The methods of Sparse Feature Propagation and Dense Feature Aggregation based on the neural network approach of multi-frame end-to-end learning are described. The methods are designed to increase the efficiency of dynamic object recognition. The general structure of the video analytics system is presented. The method of increasing the accuracy of the obtained recognition results and increasing the probability of detecting high-speed objects by the built-in motion detection module is proposed.

1. Introduction

Nowadays, the level of scientific and technological progress entails the targeted development of machine vision systems as one of the most important effective tools for interaction between technology and humans. A key aspect in the field of technical vision is the problem of automatic recognition of images of dynamic objects, such as people, animals, cars and aeroplanes.

Pattern recognition involves determining which particular class the source data belongs to by extracting some of their characteristic features from the total mass of irrelevant data. A single frame of an object of observation is often insufficient for unambiguous identification because information about its characteristics may be incomplete. This is due to problems of illumination, terrain and image resolution, as well as the direct configuration of the surveillance camera. Modern recognition systems solve the problem of detecting objects and their supervision throughout the entire observation period. The accumulated recognition result is refined with each new frame and the probability of unambiguous identification of the object is increased [22].

2. Features of detecting static objects

In recent years, significant progress in the field of detecting static objects using deep neural networks has been observed [1]. A machine learning algorithm based on convolutional networks is one of its varieties.

In convolutional neural networks, an image is viewed, in general, through a window of a much smaller size and moving to the right and down. Inside this window, you search for some important features. For example, a vertical or horizontal line. What the convolutional neural network considers to be an important feature is determined during the training. The locations of these features are shown on feature maps. A certain combination of features in a specific area may indicate the existence of a more complex feature.



For example, your first feature map might look for curves. The second feature map can show a combination of circles referring to the first curve map. The third feature map can indicate lines. The result feature map can detect a bicycle from previous maps [2].

Recently, a common methodology of constructing network architecture has been used more often.

The idea is based on two steps:

1. The extraction of a feature map set (maps of lines, circles, etc) $F = Nfeat(I)$ over the whole input image I via the convolutional basis of the neural network. Detailed information on the convolutional basis of the neural network is presented in [1, 3, 4, 5, 6, 7, 8]. The calculation $Nfeat(I)$ is relatively time-consuming, and therefore, its application to the entire video sequence is impractical.

2. Generating the result of the neural network detection (complex feature maps, e.g. the bicycle map considered above) $Y = Ndet(F)$ upon the feature maps F by performing region classification and bounding box regression over either sparse object proposals [9, 10, 11, 12, 13, 14, 15] or dense sliding windows [16, 17, 18, 19] via a multi-branched sub-network. $Y = Ndet(F)$ is randomly initialized and co-trains with $Nfeat(I)$. The calculation is generally not so complicated or time-consuming.

3. Problems of dynamic object detection

Significant problems arise in the direct application of the video detectors described above. Firstly, the use of recognition algorithms for all frames entails unacceptable computational costs. Secondly, the recognition accuracy is weaker because of reduced visibility in video clips due to, for example, the blurring of images or video defocusing.

Recent articles dedicated to dynamic object detection [20, 21] describe fundamentally multi-frame end-to-end learning. In particular, the approach of eliminating frame-to-frame redundancy is used in [21] to reduce the expensive computational function and to increase calculating speed.

The Sparse Feature Propagation method [21] is based on the concept of key frames (figure 1). The basic idea is that a similar appearance among neighbouring frames results in a similar complexity of the calculated function $Nfeat(I)$. Accordingly, its execution is not required for the entire video sequence. It is needed only for sparse key frames (e.g., for every tenth frame). Moreover, the feature maps on any non-key frame i are propagated from its preceding key frame k by per-pixel feature value warping W and bilinear interpolation.

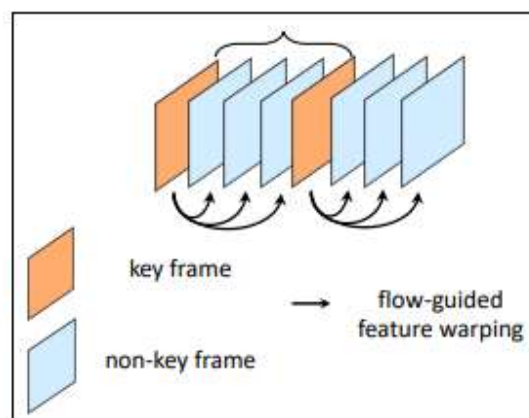


Figure 1. Sparse Feature Propagation method.

The intra-frame pixelwise motion is recorded in a two-dimensional motion field $M_{i \rightarrow k}$. where the two-dimensional motion field is part of a frame (matrix of pixels), the approximation to which can be seen on nearby frames.

The propagation from key frame k to frame i is denoted as

$$F_{k \rightarrow i} = W(F_k, M_{k \rightarrow i}), \quad (1)$$

W - feature warping function. Then the detection network (see section 2) $Y = N \det(Nfeat(F_{k \rightarrow i}))$ works on $F_{k \rightarrow i}$, the approximation to the real feature F_i , instead of computing a result based on $Nfeat(F_i)$. However, the propagated features are only approximate and error prone. This can impair the accuracy of recognition.

The Dense Feature Aggregation method is performed in [20] to improve the quality and accuracy of recognition. The motivation is that deep features or functions will deteriorate due to the presence of blurring or overlapping objects in some images. On the other hand, they can be improved by aggregating from neighbouring frames.

Feature maps set $Nfeat$ (see section 2) is calculated on all frames. For any frame i , the feature maps of all frames (see equation 2) in the time window $[i - r, i + r]$, ($r = 2 \sim 12$ frames) are first deformed into a frame i as equation 1, forming a features map set $\{F_{k \rightarrow i} \mid k \in [i - r, i + r]\}$.

The aggregated feature maps F^- at frame i :

$$F_i^-(p) = \sum_{k \in [i-r, i+r]} W_{k \rightarrow i}(p) * F_{k \rightarrow i}(p), \forall p \quad (2)$$

Weight $W_{k \rightarrow i}$ is adaptively calculated as the similarity between propagated feature maps $F_{k \rightarrow i}$ and real feature maps F_i . The weight is normalized at every location p (point of frame) over near frames $\sum_{k \in [i-r, i+r]} W_{k \rightarrow i}(p) = 1$.

Thus, the main difference from the method [21] is that the propagation of features occurs over all frames. In other words, each frame is considered as key. Compared to a single frame detector, the aggregation in equation 2 significantly improves performance and improves detection accuracy by about 3 mAP (mean Average Precision for object detection, see section 7) points, especially for fast-moving objects (about 6 mAP points). However, the operating time is about 3 times slower due to the repetition of the flow estimate and the aggregation of objects by dense consecutive frames.

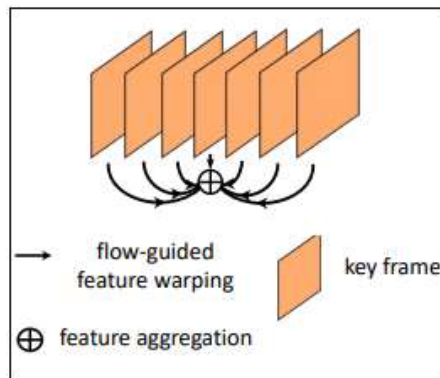


Figure 2. Dense Feature Aggregation method.

4. Video analytics system and its structural element interaction

Video surveillance is a process that is performed using optical-electronic devices for visual control and automatic image analysis. There are quite a number of practical applications:

- photo/video fixation of traffic violations;
- shooting an object in motion;
- systems that allow objects to be classified;
- systems such as "Safe City".

Generally, the common structure of a video analytics system includes a set of video cameras, image capture and recognition modules, a database and knowledge base, as well as a decision-making system. Figure 3 demonstrates the interaction structure of the main modules.

The capture module provides two-way interaction with video cameras. Stop/start commands, get/set settings for exposure, gain and frames per second (fps) among others are served at the entrance to the video camera. The requested settings and video are transmitted at the output. There are several ways to process the video from cameras: the RTSP, ONVIF/PSIA, HTTP standards, and the SDK as a set of tools or interaction interfaces (API) presented by the developers of video surveillance cameras.

Consider the input and output parameters of the recognition module using the example of AutoCode Traffic VMS technology developed by the Ukrainian company Video Internet Technologies based on the neural network approach and actively used in Russia and other CIS countries for the recognition of number plates [22].

Module input data:

1. Settings:

- ROI (Region of interest) covering only part of the image;
- on/off dynamic mode, viewing the image in connection with others;
- the time interval required to decide the recognized object as lost (out of sight);
- the minimum observation time required to make a decision about object recognition;
- threshold in probability (take into account objects that are estimated above a certain given level of probability);
- on/off filter by the number of unrecognized characters;
- number of unrecognized characters allowed;
- return period of recognition results.

2. Video stream in Grayscale format

Module output data:

- image on which the decision was made;
- coordinates of the number plate;
- recognition result as a string of characters;
- probability with which the decision was made;
- flag that informs with a positive value that the object of observation is lost and the current result belongs to the best of all the intermediate ones.

Thus, the recognition module considers all intermediate results in relation to one another and, after losing the object, makes a final decision on the recognized image.

The output data of the recognition module enter the general decision-making system. At this stage, the imposition of a number of conditions is usually performed. This allows it to be determined whether an object belongs to a certain class. For example, traffic enforcement violations or objects located in the forbidden zone.

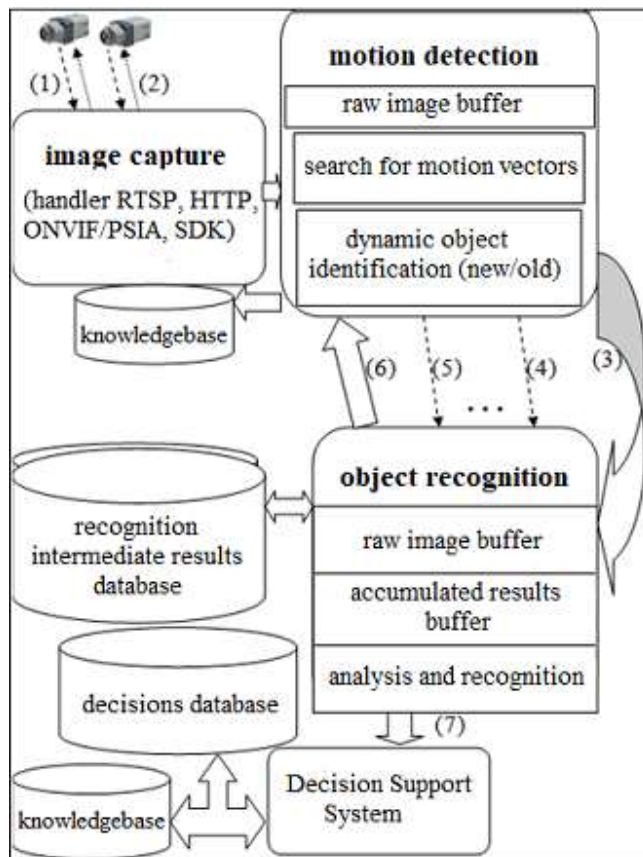


Figure 3. Distributed system structure: (1) - video data streams; (2) - management commands; (3) - a message about the loss of an object from visibility; (4) - the flow of messages about the first object; (5) –the flow of n-object messages; (6) - the command to stop tracking the i-object; (7) - recognition results.

5. Advanced video analytics system

Video analytics systems often refer to real-time systems. Accordingly, a rigid time frame is imposed on the recognition subsystem. The number of frames processed per unit of time is directly proportional to the probability of detecting objects moving at high speed.

To reduce the computational cost of the search and the number of false positives when maintaining the object of observation, it is proposed to embed the motion detection module into the video analytics system (figure 3).

Video data from surveillance cameras are being input to the built-in detection subsystem. The output data which are then being input to the recognition module are informational message streams, which are of two types of data structure:

1. The structure of the detection result, including the components:

- image on which the existence of the object was decided on;
- area in which the object was detected;
- identifier of the real-world object necessary to relate to the previous results of its detection;

- the time the frame was created.

2. The structure of informing of the tracking termination:

- identifier of the real-world object;
- message about the loss of the object.

Moreover, information about each individual object is transmitted by the detection module by an independent parallel stream of data structures.

Thus, the detection module will reduce the computational load of the recognition subsystem by providing the already accumulated information about the objects, delegating to itself the responsibility of detecting the observation area and calculating the motion. Firstly, it will allow the system of video analytics to increase the number of frames analysed in real time and, thereby, increase the percentage of recognized high-speed objects. Secondly, it will allow the accuracy of recognition as a whole to be improved, concentrating on one object of interest.

6. The structure of the motion detection module

Motion detection module includes the following components:

- temporary storage of accumulated queuing for image processing. Buffer size is a custom value. If the limit is reached, each newly arrived image is discarded to prevent unnecessary processing time. Accordingly, as the volume decreases, the speed of motion detection on the video stream increases, but the performance of the recognition module deteriorates due to the lack of sufficient input information to make a decision. On the other hand, as the buffer increases, temporary lags appear, but the accuracy of detection and recognition of a moving object increases;
- knowledge base, the main practical importance of which is in the storage of intermediate results of detecting a dynamic object;
- the subsystem of the search for motion vectors, performing frame-by-frame analysis, identifying key features and drawing up some map of their movement throughout the video sequence. A feature (see section 2) means some part of the object of observation (a block of pixels) that appeared on the key frame and repeats itself on subsequent intermediate frames. The features map shows that the block of interest was first seen on the frame and its further movement on + 1,2,3 ... frames.

7. Experiments

ImageNet VID dataset [23] is a prevalent large-scale benchmark for video object detection. It is used for this experiment. Model training and evaluation are performed on the 3,862 video snippets from the training set and the 555 snippets from the validation set. There are about 25 frames per second for the video snippets. The images are resized to a shorter side of 600 pixels for the image recognition network, and a shorter side of 300 pixels for the flow network. 30 object classes are used for experiment. Stochastic Gradient Descent training is performed (samples are selected randomly).

The metric to measure the accuracy of object detectors (mAP) using the considered methods is calculated and displayed on the graph. It precisely measures the “false positive rate” or the ratio of true object detections to the total number of objects.

Calculating mAP using equations 3:

$$mAP = \frac{1}{|classes|} \sum_{c \in classes} \frac{TP(c)}{TP(c) + FP(c)} \quad (3)$$

Where True Positive $TP(c)$ - a proposal was made for class c and there actually was an object of class c ; False Positive $FP(c)$ - a proposal was made for class c , but there is no object of class c ;

Overall comparison results are shown in figure 4.

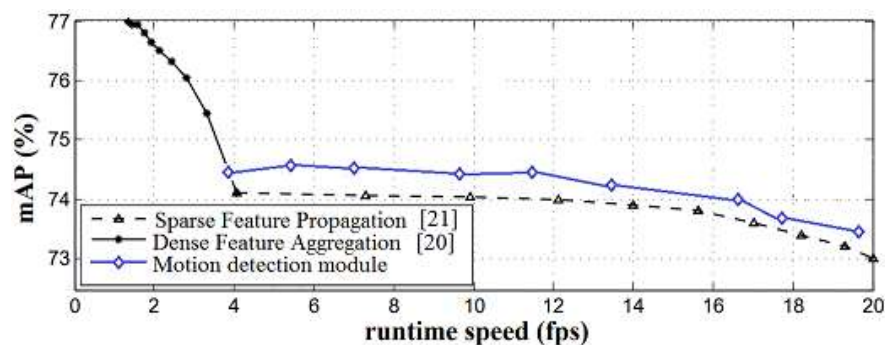


Figure 4. Speed-accuracy trade-off curves for the detection methods.

As for Sparse Feature Propagation [21], by varying key frame duration from 1 to 10, it can achieve a five-fold acceleration speedup with moderate accuracy loss (within 1%). As for Dense Feature Aggregation [20], by varying the temporal window to be aggregated from ± 1 to ± 10 frames, it improves the mAP score by 2.9% but the runtime speed is about three times slower than the per-frame baseline. The method is not able to reach an acceptable processing speed (about 20 fps). As for the motion detection module, the mAP score is higher than Sparse Feature Propagation with a sufficiently high number of frames per second.

Conclusion

Currently, there are effective methods for detecting static objects based on convolutional neural networks. However, these mechanisms are not able to perform well the detection of dynamic objects. There are the problems of unacceptable computational costs and loss of recognition accuracy. To eliminate them, it was proposed to embed a motion detection module into the general system of video analytics. The embedded subsystem receives streams of images from video surveillance cameras. At the output, the motion detection module transmits the accumulated information to the recognition subsystem as a series of parallel data structures. The data structures contain detailed information about the time the frame was created, the location of the intended object of observation, and whether the object is new or needs to be considered in conjunction with other results. Thus, it will allow the video analytics system to analyse more information in real time. Accordingly, the number of detected high-speed objects and the accuracy of the output information about the video surveillance object as a whole will increase.

References

- [1] Simonyan K and Zisserman A 2015 Very deep convolutional networks for large-scale image recognition *ICLR* 2-5
- [2] Zeiler M D and Fergus R 2014 Visualizing and Understanding Convolutional Networks *ECCV* 818-33
- [3] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V and Rabinovich A 2015 Going deeper with convolutions *CVPR* 2-5
- [4] Szegedy C, Ioffe S, Vanhoucke V and Alemi A 2016 Inceptionv4, inception-resnet and the impact

- of residual connections on learning *arXiv preprint arXiv:1602.07261*
- [5] Huang G, Liu Z, Weinberger K Q and van der Maaten L 2017 Densely connected convolutional networks *CVPR* 2-5
 - [6] Xception F 2017 Deep learning with depthwise separable convolutions *CVPR* 5
 - [7] Howard A G, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M and Adam H 2017 Mobilenets: Efficient convolutional neural networks for mobile vision applications *arXiv preprint arXiv:1704.04861*
 - [8] Huang J, Rathod V, Sun C, Zhu M *et al* 2017 Accuracy trade-offs for modern convolutional object detectors *CVPR* 1-5
 - [9] Girshick R, Donahue J, Darrell T and Malik J 2014 Rich feature hierarchies for accurate object detection and semantic segmentation *CVPR* 2-5
 - [10] He K, Zhang X, Ren S and Sun J 2014 Spatial pyramid pooling in deep convolutional networks for visual recognition *ECCV* 2-5
 - [11] Girshick R 2015 Fast r-cnn *ICCV* 2-5
 - [12] Ren S, He K, Girshick R and Sun J 2015 Towards real-time object detection with region proposal networks *NIPS* 2-5
 - [13] Dai J, Li Y, He K and Sun J 2016 Object detection via region-based fully convolutional networks *NIPS* 2-5
 - [14] Dollar P, Lin T-Y, Girshick R, He K, Hariharan B and Belongie S 2017 Feature pyramid networks for object detection *CVPR* 2-5
 - [15] Dai J, Qi H, Xiong Y, Li Y, Zhang G, Hu H and Wei Y 2017 Deformable convolutional networks *ICCV* 2-7
 - [16] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y and Berg A C 2016 Single shot multibox detector *ECCV* 2-5
 - [17] Redmon J, Divvala S, Girshick R and Farhadi A 2016 Unified, real-time object detection *CVPR* 2-6
 - [18] Redmon J and Farhadi A 2016 Yolo9000: better, faster, stronger *arXiv preprint arXiv:1612.08242*
 - [19] Goyal P, Lin T-Y, Girshick R, He K and Dollar P 2017 Focal loss for dense object *arXiv preprint arXiv:1708.02002*
 - [20] Zhu X, Wang Y, Dai J, Yuan L and Wei Y 2017 Flow-guided feature aggregation for video object detection *ICCV* 1-8
 - [21] Zhu X, Xiong Y, Dai J, Yuan L and Wei Y 2017 Deep feature flow for video recognition *CVPR* 1-8
 - [22] VIT Company 2015 SDK for number plate recognition https://docs.vitcompany.com/en/wiki/SDK_for_number_plate_recognition
 - [23] Russakovsky O, Deng J, Su H, Krause J, Satheesh S 2015 Imagenet large scale visual recognition challenge *IJCV* 6