UAV Tracking with Proposals Based on Optical Flow

Min Jia[1], Zheng Gao[1(✉)], Zhisong Hao[2], and Qing Guo[1]

[1] School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China
alexzgao@outlook.com
[2] The 54th Research Institute of China Electronics Technology Group Corporation, Shijiazhuang 050000, Hebei, China

Abstract. UAV tracking is aimed to infer the location of the object from the videos captured by an aerial viewpoint. The challenges mainly focus on fast motion, scale variation and aspect ratio variation. The region proposal in image detection can detect the object candidates in the image, which can be leveraged to find the optimal location of the object. In this paper, a tracking algorithm using Farneback optical flow is proposed to provide object proposals for correlation filter for robust tracking under aerial scenarios. The Farneback flow estimates the motion of the object between adjacent frames and an improved FAST detector is adopted to detect the keypoints that contain the local patterns of the object from the last frame. The object proposal is obtained by computing translations of the keypoints. The final proposal is determined by computing the bounding box that encloses the keypoints. A correlation filter from KCF is used to detect the object on the proposal. The quantitative evaluation results on OTB100 show the advantage of the proposed tracker to state-of-the-art trackers in accuracy, especially under fast motion.

Keywords: UAV tracking · Farneback · FAST · Correlation filter

1 Introduction

Generic object tracking has made significant progress these years. The tracking-by-detection framework focuses on applying image detection techniques to tracking systems and achieves great success. Since the feature detection has accomplished superior results in object representation, the tracking algorithm based on feature detector tends to perform better in accuracy than traditional trackers, which are developed by estimating the motions of the object between two adjacent frames [1]. The traditional feature detectors (e.g., HOG, LINEMOD) are proposed to represent the local features in the image and perform incredibly well on feature representation tasks, which contributes to the improvement of feature detection.

In order to construct feature detectors beyond manual design, some researchers begin to explore the study of learning neural networks from the training dataset to represent the objects. The application of neural network accounts for the success in image detection and classification in recent years [2]. Inspired by the work related to

M. Jia et al. (Eds.): WiSATS 2019, LNICST 281, pp. 497–505, 2019.
https://doi.org/10.1007/978-3-030-19156-6_46

neural network, the literatures that discuss deep tracking algorithms built upon deep learning have attracted more attention recently [3]. Held et al. [4] first propose GOTURN, a tracking method based on deep regression network that can run in real time. GOTURN uses the deep regression network to learn the connection between the object appearance and the object location by estimating the motions, which is completely different from the methods with convolutional neural network (CNN) for feature detection and classification. Since the classification network has excellent performance for the description of the object, Bertinetto et al. [5] adopts an offline trained CNN for feature detection by computing convolutional features.

However, there are only few literatures related to UAV tracking. Compared to generic object tracking, the main challenges are different. Mueller et al. [6] argue that for UAV tracking, the primary challenges include fast motion, scale variation and aspect ratio variation.

Zhu et al. [7] introduce the idea of region proposal in image detection to object tracking to deal with fast motion. A search strategy for object location beyond local search is proposed to detect the object candidates and the detection is performed on the proposals. They extract the edge map for the current frame first and then find the bounding boxes that enclose the contours based on Edge Boxes. The object candidates are in the closed boundaries. However, the object proposal strategy based on contours faces two major problems: (1) it is likely to produce an edge map with lots of false proposals that do not contain the object especially when the textures of the background are diverse; (2) the objects with similar aspect ratio to the object of interest will seriously interfere the detection on the proposal boxes, such as tracking for pedestrians on the street.

Generally, the object can be represented with some keypoints. The keypoint contains local structures of the object and is widely used in many fields like face recognition, pose estimation, gesture detection, etc. The local keypoint reflects the tree-structured structure of the object and connection of different parts. Therefore, we would like to track the trajectories of the keypoints along the adjacent frames and estimate the motions of the keypoints to infer the object proposals.

The main work in this paper is to combine optical flow and correlation filter in [8] for UAV tracking. The optical flow is used to compute motions of the keypoints and estimate the object proposals. The correlation filter is performed on the object candidates for accurate detection of the object.

The main contributions of our work are summarized as follows. First, we leverage the Farneback optical flow for object candidate estimation. The flow keeps tracks of the keypoints inside the object and provides information of potential location of the object, which is beyond the local search strategy broadly used in the current trackers. Second, a FAST detector with local threshold is proposed to detect the local strong keypoints inside the bounding box. The keypoints in the previous frame is employed to predict the locations in the current frame based on the flow. The rectangle that encloses the keypoints indicates the proposal of the object. Note that the proposed tracker is called OFT in this paper.

2 Correlation Filter

The correlation filter is meant to solve ridge regression problem with DFT operation for high speed. The correlation filter is used to perform convolution operation to compute the similarity to the object for the test sample. The training samples are generated by cyclic shifts of the base sample and construct a circulant matrix, which can be expressed with DFT matrix. Supposed that the set of samples is denoted by $\{\boldsymbol{x}_i\}$, and the regression labels are $\{y_i\}$. The optimal solution to the ridge regression problem is found by solving the following loss function:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \|X\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{w}\|_2^2 \tag{1}$$

where X is the circulant matrix composed of the positive sample (i.e., base sample) and negative samples (i.e., shifted samples from the base sample) as the rows, $\boldsymbol{w}$ is the correlation filter, and λ is a regularization parameter that balances the residual error and generalization. The regression function $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}$ outputs the detection score for the test sample $\boldsymbol{x}$, which decides the label of the sample. The work in [8] proves that the solution to (1) is closed-form with $\boldsymbol{w} = (X^HX + \lambda I)^{-1}X^H\boldsymbol{y}$. However, the closed-form solution requires inversion operation, which is expensive in matrix operation and time consuming. Since the circulant matrix can be expressed as a function over its DFT transformation and DFT matrix, the matrix X can be diagonalized as follows:

$$X = F\mathrm{diag}(\hat{\boldsymbol{x}})F^H \tag{2}$$

where $\widehat{\boldsymbol{x}}$ is the expression of $\boldsymbol{x}$ in Fourier domain, and F represents the DFT matrix with $\widehat{\boldsymbol{x}} = \sqrt{n}F\boldsymbol{x}$. The closed-form solution can be solved efficiently with (2):

$$\hat{\boldsymbol{w}} = \frac{\hat{\boldsymbol{x}} \odot \hat{\boldsymbol{y}}}{\hat{\boldsymbol{x}}^* \odot \hat{\boldsymbol{x}} + \lambda} \tag{3}$$

where $\odot$ performs the product in an element-wise way. The optimal solution is computed efficiently in Fourier domain and the real solution is obtained via inverse DFT. To improve the classification performance in non-linear space, the kernel trick is used to obtain a non-linear function $f(z) = \boldsymbol{w}^T\phi(z) = \boldsymbol{\alpha}^T\phi(X)\phi(z) = \sum_i \alpha_i\kappa(z, \boldsymbol{x}_i)$, where $\boldsymbol{w} = \sum_i \alpha_i\phi(\boldsymbol{x}_i) = \phi(X)^T\boldsymbol{\alpha}$ is expressed in kernel space, $\phi(z)$ maps the original space of z to a high-dimensional space, $\kappa(z, \boldsymbol{x}_i)$ performs the dot-product $\kappa(z, \boldsymbol{x}_i) = \phi(z)^T\phi(\boldsymbol{x}_i)$. Thus the optimal solution can be performed over $\boldsymbol{\alpha}$ instead of $\boldsymbol{w}$:

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \left\|\phi(X)\phi(X)^T\boldsymbol{\alpha} - \boldsymbol{y}\right\|_2^2 + \lambda\left\|\phi(X)^T\boldsymbol{\alpha}\right\|_2^2 \tag{4}$$

The solution to (4) can be written as:

$$\hat{\boldsymbol{\alpha}} = \frac{\hat{\boldsymbol{y}}}{\hat{\boldsymbol{k}}^{\boldsymbol{xx}} + \lambda} \tag{5}$$

where $\boldsymbol{k}^{\boldsymbol{xx'}}$ is composed of elements $k_i^{\boldsymbol{xx'}} = \kappa(\boldsymbol{x'}, \boldsymbol{x}_i)$, $\boldsymbol{x}_i$ is the i-th row of X, and $\boldsymbol{k}^{\boldsymbol{xx'}}$ is the DFT transformation of $\boldsymbol{k}^{\boldsymbol{xx'}}$. The regression function in Fourier domain can be expressed as:

$$\widehat{f}(\boldsymbol{z}) = \widehat{\boldsymbol{k}}^{\boldsymbol{xz}} \odot \widehat{\alpha} \tag{6}$$

The regression function outputs a 2-dimensional response map and the maximum value in the map corresponds to the location of the object.

The correlation filter is evaluated on the neighboring region surrounding the tracked location from the last frame. However, the local sampling strategy cannot deal with fast motion. In this paper, the optical flow is adopted to compute the motions and infer the object proposals.

3 Proposals Based on Optical Flow

The basic idea of object proposal is to use Farneback optical flow to estimate the motions in the image and decide final proposal based on the keypoints of the object. The keypoints are detected by FAST detector with local threshold and the object proposal is estimated by computing the bounding box enclosing the keypoints.

3.1 Farneback Optical Flow

Farneback uses a quadratic polynomial to formulate the intensity of the pixel in the local region. The approximation over local coordinates $\boldsymbol{x}$ for the neighboring region of one pixel can be written as:

$$I(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{x} + c \tag{7}$$

where $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$, $\boldsymbol{b} \in \mathbb{R}^2$ and c are coefficients of the quadratic polynomial, which can be computed with weighted least square algorithm. The polynomial is a function with respect to the horizontal and vertical coordinates $\boldsymbol{x} = (x, y)$. Since the local region surrounds the point of interest, the estimated coefficients vary when the point in the center changes.

When the point moves to the new position in the next frame by displacement $\boldsymbol{d}$, the approximation can be rewritten as:

$$\begin{aligned} I_2(\boldsymbol{x}) &= I_1(\boldsymbol{x} - \boldsymbol{d}) = (\boldsymbol{x} - \boldsymbol{d})^T \boldsymbol{A}_1 (\boldsymbol{x} - \boldsymbol{d}) + \boldsymbol{b}_1^T (\boldsymbol{x} - \boldsymbol{d}) + c_1 \\ &= \boldsymbol{x}^T \boldsymbol{A}_1 \boldsymbol{x} + (\boldsymbol{b}_1 - 2\boldsymbol{A}_1 \boldsymbol{d})^T \boldsymbol{x} + \boldsymbol{d}^T \boldsymbol{A}_1 \boldsymbol{d} - \boldsymbol{b}_1^T \boldsymbol{d} + c_1 \\ &= \boldsymbol{x}^T \boldsymbol{A}_2 \boldsymbol{x} + \boldsymbol{b}_2^T \boldsymbol{x} + c_2 \end{aligned} \tag{8}$$

where $\boldsymbol{A}_2 = \boldsymbol{A}_1$, $\boldsymbol{b}_2 = \boldsymbol{b}_1 - 2\boldsymbol{A}_1\boldsymbol{d}$, $c_2 = \boldsymbol{d}^T\boldsymbol{A}_1\boldsymbol{d} - \boldsymbol{b}_1^T\boldsymbol{d} + c_1$. If $\boldsymbol{A}_1$ is non-singular, the displacement $\boldsymbol{d}$ can be obtained by:

$$\boldsymbol{d} = -\frac{1}{2}\boldsymbol{A}_1^{-1}(\boldsymbol{b}_2 - \boldsymbol{b}_1) \tag{9}$$

In general case for $\boldsymbol{A}_1$, the translation is solved by minimizing the following objective function:

$$\boldsymbol{d}^* = \arg\min_{\boldsymbol{d}} \|\boldsymbol{A}\boldsymbol{d} - \Delta\boldsymbol{b}\|_2^2 \tag{10}$$

where $\Delta\boldsymbol{b} = \boldsymbol{b}_2 - \boldsymbol{b}_1$. However, the optimization on a single point is unreliable as the motion of the point can be affected by the noise. Therefore, the minimization is expanded to the neighboring region of the point:

$$\boldsymbol{d}(\boldsymbol{x})^* = \arg\min_{\boldsymbol{d}(\boldsymbol{x})} \sum_{\Delta\boldsymbol{x}\in\Omega} w(\Delta\boldsymbol{x})\|\boldsymbol{A}(\boldsymbol{x} + \Delta\boldsymbol{x})\boldsymbol{d}(\boldsymbol{x}) - \Delta\boldsymbol{b}(\boldsymbol{x} + \Delta\boldsymbol{x})\|_2^2 \tag{11}$$

where Ω represents the neighboring region of the point $\boldsymbol{x}$, $w(\Delta\boldsymbol{x})$ denotes a 2-dimensional weight function that controls the influence of the points in the local region on the objective function. The solution to (11) can be expressed as:

$$\boldsymbol{d}(\boldsymbol{x}) = \left(\sum w\boldsymbol{A}^T\boldsymbol{A}\right)^{-1}\sum w\boldsymbol{A}^T\Delta\boldsymbol{b} \tag{12}$$

3.2 FAST Detector with Local Threshold

The FAST detector is improved by adopting a local threshold strategy. The local threshold is aimed to detect the strong keypoints in the local region. FAST tries to find the pixels whose intensity is brighter or darker than the pixels surrounding it. The candidate pixel $\boldsymbol{c}$ is compared with local pixels that lie on the line of a circle with radius 3. The total number of the local pixels is 16. Suppose that the intensity of one pixel is $I(\boldsymbol{c})$ and the local pixel around it is $I(\boldsymbol{c} \rightarrow \boldsymbol{p})$. If there are N contiguous pixels with greater or smaller intensities than $I(\boldsymbol{c})$ by a threshold T, the candidate pixel is decided as a keypoint.

FAST uses a global threshold for the detection of all points in the image. In this paper, we adopt a local threshold strategy to detect the points whose intensity is considerably different from the surrounding pixels. Assuming that the candidate pixel is close to the pixels in the eight-neighbor region. If the difference between the candidate pixel and the eight-neighbor pixels is high, the candidate pixel can be affected by the noise. Thus the difference in intensity in the neighboring region can be used to control the strength of the threshold. The pixel that is heavily affected by the noise corresponds to high threshold; otherwise, the pixel is detected with small threshold. The proposed local threshold is expressed as follows:

$$T = (k\sigma_{3\times3})^{\left(1-\frac{C\left(\max_i |e_i| / \sum_i |e_i|\right)-1}{C-1}\right)} = (k\sigma_{3\times3})^{\frac{C}{C-1}\left(1-\left(\max_i |e_i| / \sum_i |e_i|\right)\right)} \tag{13}$$

where $C = 8$ is the number of the pixels in the eight-neighbor region, I_i represents the pixels in the region. $e_i = I_c - I_i$ is computed to evaluate the difference between the candidate pixel I_c and its neighboring pixels I_i and $\sigma_{3\times3} = \sum_i |e_i|/C$ is the average difference. k is a constant to adjust the strength of the threshold. Moreover, the confidence term $1 - \left(C\left(\max_i |e_i| \Big/ \sum_i |e_i|\right) - 1\right) \Big/ (C-1)$ evaluates the distribution of the intensity difference in the neighboring region and varies from 0 to 1. If the differences are close to each other, the distribution item approaches 0; otherwise, it approaches 1. If the distribution is not uniform and some differences are larger than the others, which means the candidate pixel is considerably brighter or darker than its neighboring pixels, the confidence term decreases and the threshold reduces accordingly.

4 Performance Evaluation

The evaluation is performed on OTB100 to compare the performance between the proposed tracker and 2 state-of-the-art methods. The evaluated methods include high-speed tracking with kernelized correlation filters (KCF) [8], learning to track at 100 fps with deep regression networks (GOTURN) [4]. KCF trains a correlation filter on HOG feature space. The difference between the proposed method and KCF is that the object proposal strategy is adopted to handle fast motion. GOTURN learns to estimate the motions of the object along the frames and predict its location with a deep regression network.

4.1 Evaluation Criteria

The comparisons in terms of precision and success are performed to evaluate the performance of the trackers. The one-pass evaluation (OPE) in [6] is used as the evaluation metric. OPE means that the tracker runs on test video without restarting during process and the average precision or success rate is computed to evaluate the tracker.

The first metric is precision rate, which is defined as the Euclidean distance in pixels between the center of the tracked location and the ground truth bounding box.

The second one is success rate. The overlap score of the tracked location in one frame is defined as:

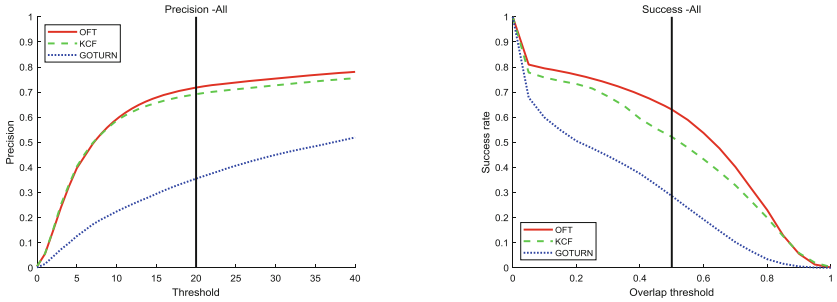
$$s = area(ROI_T \cap ROI_G)/area(ROI_T \cup ROI_G) \tag{14}$$

where ROI_T denotes the tracked bounding box and ROI_G represents the ground truth bounding box. $\cap$ and $\cup$ are the intersection and union operation of these two boxes respectively and $area(\bullet)$ means the area of the box in pixels. Given an overlap threshold τ, the percentage of the frames whose overlap score is larger than τ is defined as the success rate.

The precision plot shows the center location errors over the frames. The success plot shows the success rates at different overlap thresholds from 0 to 1.

4.2 Quantitative Comparisons

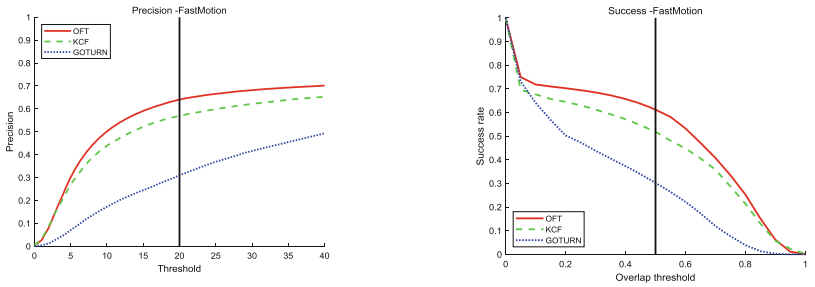
The overall OPE plots on all videos from OTB100 are given in Fig. 1. The proposed OFT is the top tracker in terms of both precision and success. The precision results at 20 pixels and success results at overlap threshold 0.5 are reported to evaluate the quantitative performance. The top two trackers are OFT and KCF. OFT achieves 0.7188 in precision and 0.6306 in success and thus outperforms KCF with precision 0.6918 and success 0.5219 by 2.7% and 10.87%, respectively. The comparisons in precision and success indicate that the proposed proposal strategy contributes to the improvement of the accuracy.



a) Precision b) Success

Fig. 1. Overall performance in OPE.

The evaluation results on fast motion challenge are shown in Fig. 2. Compared to the results of the overall performance, OFT accomplishes greater improvements for fast motion. The results for OFT are 0.6405 in precision and 0.6116 in success, compared to KCF with 0.5690 in precision and 0.5187 in success. The gap between OFT and KCF confirms that the object proposal can deal with fast motion effectively.



a) Precision b) Success

Fig. 2. OPE under fast motion.

5 Conclusion

Although the generic tracking has made significant progress in recent years, the UAV tracking for videos captured from aerial viewpoints still faces challenges such as fast motion, scale variation and aspect ratio variation. Inspired by the idea of region proposal in image detection, an object proposal strategy is proposed to infer the object candidates based on Farneback flow. First, the Farneback flow is computed to find the connections between two adjacent frames by estimating the motions in the image. Then the keypoints that represent the local structures of the object are detected with a FAST detector. The FAST detector is improved with a local threshold based on the intensity distribution in the neighboring region around the candidate point. The keypoints inside the object bounding box from the last frame are used to find the movements in the next frame and decide the box for object candidate, which predicts the object proposal. A correlation filter is adopted to detect the object location on the proposals. The evaluation is performed on OTB100 to compare the proposed method and the state-of-the-art trackers, which demonstrates that the proposed tracker achieves superior results in accuracy.

Acknowledgement. This work was supported by the National Natural Science Foundation of China (No. 61671183, 61771163 and 91438205) and the Open Research Fund of State Key Laboratory of Space-Ground Integrated Information Technology (No. 2015_SGIIT_KFJJ_TX_02).

References

1. Li, P., Wang, D., Wang, L., Lu, H.: Deep visual tracking: Review and experimental comparison. Pattern Recognit. **76**, 323–338 (2018)
2. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. In: 2018 IEEE Conference Computer Vision Pattern Recognition, Salt Lake City, UT (2018)
3. Choi, J., Chang, H.J., Yun, S., Fischer, T., Demiris, Y., Choi, J.Y.: Attentional correlation filter network for adaptive visual tracking. In: 2017 IEEE Conference Computer Vision Pattern Recognition, Honolulu, HI, pp. 4828–4837 (2017)

4. Held, D., Thrun, S., Savarese, S.: Learning to track at 100 FPS with deep regression networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 749–765. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_45
5. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: Fully-convolutional siamese networks for object tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 850–865. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_56
6. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for UAV tracking. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 445–461. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_27
7. Zhu, G., Porikli, F., Li, H.: Beyond local search: tracking objects everywhere with instance-specific proposals. In: 2016 IEEE Conference Computer Vision Pattern Recognition, Las Vegas, NV, pp. 943–951 (2016)
8. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-Speed Tracking with Kernelized Correlation Filters. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 583–596 (2015)