

TRACKING MOTION-BLURRED TARGETS IN VIDEO

Shengyang Dai, Ming Yang, Ying Wu, Aggelos K.Katsaggelos

Department of EECS, Northwestern University
2145 Sheridan Road, Evanston, IL 60208

{sda690, mya671, yingwu, aggk}@ece.northwestern.edu

ABSTRACT

Many emerging applications require tracking targets in video. Most existing visual tracking methods do not work well when the target is motion-blurred (especially due to fast motion), because the imperfectness of the target’s appearances invalidates the image matching model (or the measurement model) in tracking. This paper presents a novel method to track motion-blurred targets by taking advantage of the blurs without performing image restoration. Unlike the global blur induced by camera motion, this paper is concerned with the local blurs that are due to target’s motion. This is a challenging task because the blurs need to be identified blindly. The proposed method addresses this difficulty by integrating signal processing and statistical learning techniques. The estimated blurs are used to reduce the search range by providing strong motion predictions and to localize the best match accurately by modifying the measurement models.

1. INTRODUCTION

Many emerging applications require tracking targets in video. In real situations, motion blurs are not uncommon in captured video data. They are undesirable for image analysis because the imperfectness of the target’s appearances is likely to jeopardize image features, such as image gradients, sum-of-square-differences (SSD), and color histograms, and thus invalidates the image matching model (or the measurement model) in tracking. Although deblurring methods that improve the image quality have been widely investigated in the literature [1, 2], these studies are often based on the assumption that the entire image is subject to the same global motion blur (this is for the case when there is camera motion). Unfortunately, in real applications, we observe more complicated and challenging situations where motion blurs are only present in parts of the image. These local motion blurs can be produced by 1) fast movements of the targets, or 2) insufficient lighting that reduces the shutter speed of auto-exposure cameras.

Studies on motion blur detection or estimation of the point spread function (PSF) can be traced back to 1970s. For example, motion blurs and out-of-focus blurs may be identified by locating the zero-crossings of the cepstrum [3] and the bispectrum [4]. In the more recent studies, blurs can be estimated

by mapping the low frequency components to the high frequencies based on a learned VQ codebook [5], or employing the discrete periodic Radon transform for efficient computation [6].

Although motion blur degrades image qualities, we can also take advantage of it. If the motion can be estimated from the blur, it can facilitate many motion-related tasks, such as, optical flow computation [7], motion segmentation [8], and tracking [9]. Note that in the aforementioned works, a global motion blur is assumed in stead of a local one.

This paper presents a novel method to deal with local motion blurs in visual tracking without deblurring, by integrating signal processing and statistical learning techniques. This learning-based scheme differentiates motion-blurred image patches from non-blurred patches and estimates the local motion blur mask to facilitate tracking. A support vector machine (SVM) classifier is employed to discriminate blurs after aligning the patches by steerable filters. Then those identified blurred patches are clustered to obtain a more accurate estimation of the motion direction. During the tracking process, blurred templates of the target are synthesized according to the blur direction and used for matching.

2. THE PROBLEM

It is reasonable to assume that the target’s motion is linear between two successive frames, so the PSF $h(x, y)$ can be modelled as a rectangular pulse whose orientation and width correspond to the direction and severity of the blur. Regardless of the orientation, the PSF is

$$h(x, y) = \begin{cases} 0 & y \neq 0, -\infty \leq x \leq \infty \\ 1/l & y = 0, -l/2 \leq x \leq l/2, \end{cases} \quad (1)$$

where l represents the strength of the blur. The blurred image $I_{blur}(x, y)$ is generated by convoluting the PSF and the original image $I(x, y)$, that is,

$$I_{blur}(x, y) = I(x, y) * h(x, y) + n(x, y), \quad (2)$$

where $n(x, y)$ represents the additive noise. In the frequency domain, the rectangular PSF corresponds to a *sinc* function, which provides valuable information for blur estimation. It is very difficult to detect the zero-crossings associated with

the *sinc* function. However, the associated strip patterns that indicate the possible presence of motion blurs are generally adequately detectable. Thus an important issue here is to locate the strip patterns, or simply the dominant directions in the frequency domain.

To identify global blurs, images can be divided into small patches so as to suppress the noise by averaging the estimated blurs. However, the situation is more complicated when detecting and identifying local motion blurs, because some parts of the image are non-blurred regions, so then we can not simply apply the averaging scheme. Thus, a critical issue here is to determine if a small image patch is motion blurred or not. We need to answer some important questions: (1) is the information in the individual image patch sufficient to determine if it is blurred? (2) how to suppress the influence of noise and other factors such as texture? (3) how to detect and estimate the blur? and (4) how can the blur be used for tracking?

To answer the first question, only one small image patch itself may not be able to determine the blur. Since it can provide some rough estimates, the final decision can be made by integrating a number of patches. For the second question, there are many factors that may influence the estimation result, such as the additive noise, texture regions, and block boundary effects due to compression. Since it is difficult to model these factors, we employ a statistical learning approach (a support vector machine) to discriminate blurs. For the third question, we employ steerable filters to estimate the dominant direction in the frequency domain.

When the target is blurred due to fast motion, the predetermined measurement model of the tracker is generally unable to obtain good matches and is likely to fail the tracker. At this point, once the blur is detected and identified, we can synthesize various blurred templates of the targets with different blur strengths, and search for the best matches of the blurred templates so as to obtain more accurate tracking results.

3. PROPOSED APPROACH

The proposed approach to tracking motion-blurred targets is illustrated in Fig. 1. The target of interest T is initialized in the first frame. For each image frame f_k , the mean-shift tracker [10] that matches the target's color-histograms is applied first. If its matching score is low, we assume that blur occurs and we need to estimate the blur parameters (direction θ_b and strength l_b). The blur identification procedure consists of three steps: blur mask detection with SVM, blur direction estimation with steerable filters, and blur strength estimation. Then, the mean-shift algorithm is applied again with synthesized blurred templates $T(\theta_b, l_b)$.

3.1. Steerable filters

Steerable filters [11] can extract the dominant direction of a small image patch. They are much more computationally efficient than the Radon transform. The filter in any given direction can be expressed as a linear combination of a small

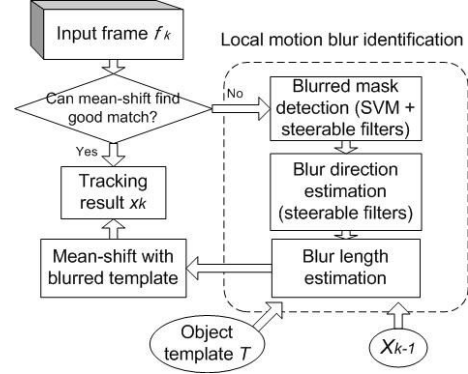


Fig. 1. The proposed approach.

set of basis filters. So we only need to compute the convolution with a small number of filters, and infer the responses on other directions by a simple linear combination. The filter in direction θ can be expressed as

$$f^\theta(x, y) = \sum_j k_j(\theta) f^{\theta_j}(x, y), \quad (3)$$

$$k_j(\theta) = \frac{1}{3}[1 + 2 \cos(2(\theta - \theta_j))], \quad (4)$$

where $\theta_j = j\pi/3$, $j = 0, 1, 2$, and f^{θ_j} is the basis filter in direction θ_j . After scaling of x and y , it can take the rotated version of the following even and odd filters respectively

$$f_e(x, y) = 0.9213(2x^2 - 1)e^{-(x^2+y^2)}, \quad (5)$$

$$f_o(x, y) = (-2.205x + 0.9780x^3)e^{-(x^2+y^2)}. \quad (6)$$

To extract the strip direction in the frequency domain, only the even filter is required due to the symmetry property. To extract the dominant direction of an image patch in the image domain, both even and odd filters are needed, since both a strong image edge and a strip may produce a dominant direction. The final direction can be chosen as

$$\theta_b = \arg \max_{\theta} ((f_e^\theta(x, y))^2 + (f_o^\theta(x, y))^2). \quad (7)$$

3.2. Blur detection based on SVM

A Support Vector Machine (SVM) [12] is a powerful statistical learning algorithm. It can learn a non-linear discrimination function from a set of labelled training data for pattern classification or data regression. This function can minimize the training error, and at the same time, guarantee the generalization ability of the classifier by maximize the margin between the so-called *support vectors*.

In our approach, an SVM is trained to classify blurred and non-blurred small image patches in order to implicitly handle the noise and other influences that are difficult to model explicitly. This approach is completely different from most existing blur estimation methods that are based on spectral analysis and are thus limited by noise. Images with a linear motion blur generally show strip patterns in the frequency domain. In theory, the blur direction and strength can be estimated by extracting the direction and width of the strips.

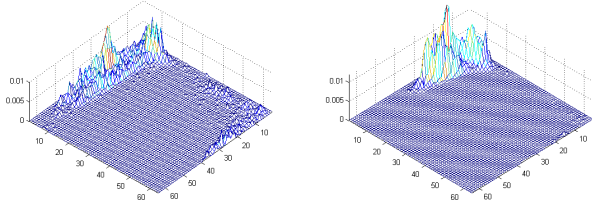


Fig. 2. 2D color histograms for the same object without and with motion blur (frame #1 and #70 in the sequence of Fig. 5(a)).

However, there are two difficulties here: (1) the strip patterns are not apparent when the patches are small such that the noise may override the strip patterns, and (2) the window effect produces edges in the frequency domain that generate undesirable patterns.

Although it is difficult to model these influences, they can be learned from the training examples. To train an SVM classifier, we collect negative (or non-blurred) training data that are patches from non-blurred images, and positive (or blurred) training data that are synthesized motion-blurred image patches. These data are aligned by rotating the patches according to their dominant directions extracted by steerable filters on a fixed scale (4 in our experiments). The direction saliency of a patch p is defined as

$$S(p) = \frac{\max_{\theta} ((f_e^{\theta}(x, y))^2 + (f_o^{\theta}(x, y))^2)}{\min_{\theta} ((f_e^{\theta}(x, y))^2 + (f_o^{\theta}(x, y))^2)}. \quad (8)$$

Patches with $S(p) \leq S_{thr}$ are not included, because they usually represent homogeneous regions and are ambiguous to classify. We set $S_{thr} = 2$. The size of each patch is 16×16 pixels, 5000 non-blurred patches are randomly collected from the Corel database, while 5000 blurred patches are synthesized on random patches selected in the same database. The blur strengths are randomly chosen from 5 to 32 pixels. Each patch is represented by its 256 DFT coefficients. The average classification accuracy of 5-fold cross validation is 90.5%, which is enough for direction estimation in our experiments.

3.3. Blur estimation for tracking

As mentioned in Sec. 1, motion blurs influence most low-level image features and thus fail many tracking algorithms. The mean-shift tracker [10], which searches for the best matching of the kernel weighted color histograms of the target by gradient descent, is used in this work. When motion blur is present, adjacent color pixels will be blended and generate new colors such that the color appearance of the target may be greatly changed (as shown in Fig. 2) to fail the mean-shift tracker. Fortunately, once we can estimate the local motion blur direction, we can synthesize various motion-blurred templates of the target with different blur strengths along the estimated motion direction. Then we can utilize the color histograms of the blurred templates to obtain the matches by mean-shift.

In our experiment, the blurred patches are identified on the region which is obtained by enlarging the tracking result in

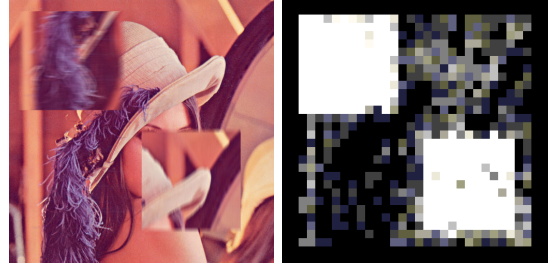


Fig. 3. Local motion blur detection and identification.

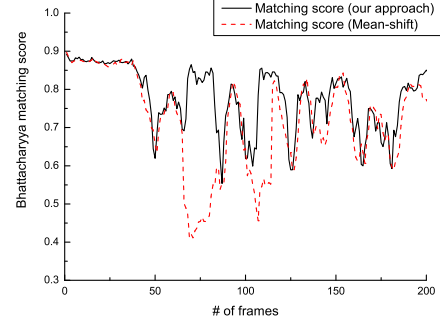


Fig. 4. Matching score comparison with Mean-shift for the sequence of Fig. 5 (a).

the previous frame by 50% in length. We use morphological operations to obtain a spatially consistent blurred mask, and steerable filters to estimate the blur direction of this region. The direction estimation result is likely to be distracted by nearby strong edges without the blur identification modular.

Once the blur direction θ_b is estimated, for a given blur strength l , a new template $T(\theta_b, l)$ can be synthesized (the template size is reduced to exclude the influence from nearby regions). The mean-shift tracker finds the best match for $T(\theta_b, l)$ very efficiently, and provides a similarity measurement denoted by $M_T(\theta_b, l)$. The blur strength l_b is chosen by a line search, the search step in our experiments is set to 4 pixels.

$$l_b = \arg \max_l M_T(\theta_b, l). \quad (9)$$

4. EXPERIMENTS

4.1. Local motion blur detection and identification

Fig. 3 shows the result of local blur detection and identification on synthesized image. The left image contains two regions blurred by different parameters. The right image shows the confidence of the SVM classifier (darker color means lower confidence). The ground truth is $\theta = 0, l = 10$ and $\theta = 90, l = 15$ for two regions respectively. The estimated blur directions of the two regions are $\theta = 5^\circ$ and $\theta = 84^\circ$ respectively, which are very close to the ground truth.

4.2. Tracking with blur estimation

We incorporate the local motion blur detection in a mean-shift tracker. The color histogram has 64×64 bins in the hue-saturation plane. As the object moves quickly or the illumination is not sufficient, the motion blurs are severe and result

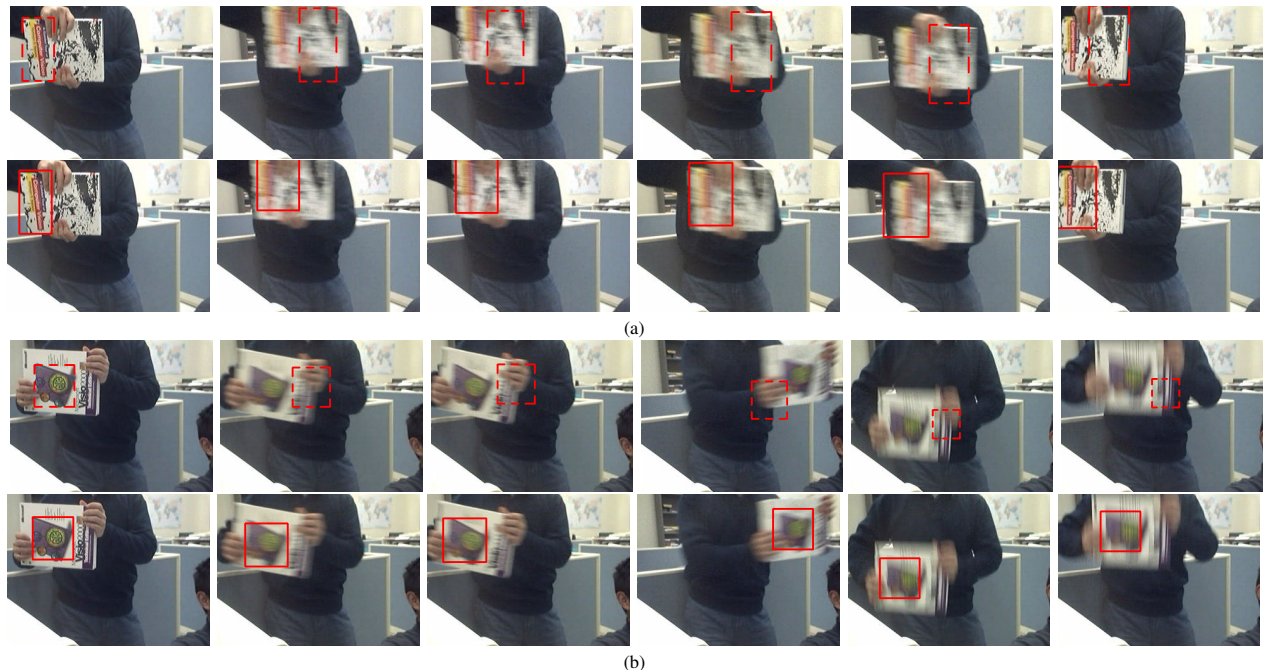


Fig. 5. Tracking and comparison. Two examples are shown, where top/bottom rows in (a) and (b) are the tracking results without/with motion blur estimation (frame # for (a): 1, 69, 70, 105, 183, 190, frame # for (b): 1, 63, 64, 135, 177, 190).

in the blending of the adjacent colors. The target is manually initialized in the first frame. As shown in Fig. 5, the yellow pixels in the book title region are blended with the adjacent red pixels, and the purple and the green ones on the box are merged to generate some yellow pixels. As shown in the first row, the mean-shift tracker is unable to find an exact matching from frame 69 and 63, thus loses track. With the help of local motion blur estimation, our approach achieves much more accurate tracking results, as shown in the bottom rows in Fig. 5 (a) and (b). For sequence (a), Fig. 4 shows that our method can achieve higher matching scores in present of motion blur, thus provide a more accurate tracking result.

5. CONCLUSION

This paper addresses the challenges in tracking motion-blurred targets and proposes a novel learning-based method to estimate local motion blurs for tracking. This method uses an SVM classifier to differentiate blurred and non-blurred image patches to cope with the influence of noise and other non-parametrical factors. The proposed method yields promising results on detecting local motion blurs and achieves encouraging results on tracking blurred targets compared to the state-of-the-art tracking algorithm. The future study includes the incorporation of more complex motion blurs and the reduction of the complexity.

6. REFERENCES

- [1] Deepa Kundur and Dimitrios Hatzinakos, "Blind image deconvolution," *IEEE Signal Processing Mag.*, vol. 13, no. 3, pp. 43 – 64, May 1996.
- [2] Mark R. Banham and Aggelos K.Katsaggelos, "Digital image restoration," *IEEE Signal Processing Mag.*, vol. 14, no. 2, pp. 24 – 41, Mar. 1997.
- [3] Michael Cannon, "Blind deconvolution of spatially invariant image blurs with phase," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 24, no. 1, pp. 58 – 63, Feb. 1976.
- [4] M.Michael Chang, Murat Tekalp, and A. Tanju Erdem, "Blur identification using the bispectrum," *IEEE Trans. Signal Processing*, vol. 39, no. 10, pp. 2323 – 2325, Oct. 1991.
- [5] Ryo Nakagaki and Aggelos K.Katsaggelos, "A VQ-based blind image restoration algorithm," *IEEE Trans. Image Processing*, vol. 12, no. 9, pp. 1044 – 1053, Sept. 2003.
- [6] Daniel P.K. Lun, Tommy C.L.Chan, Tai-Chiu Hsung, David D. Feng, and Yuk-Hee Chan, "Efficient blind image restoration using discrete periodic radon transform," *IEEE Trans. Image Processing*, vol. 13, no. 2, pp. 188 – 200, Feb. 2004.
- [7] Ioannis M. Rekleitis, "Steerable filters and cepstral analysis for optical flow calculation from a single blurred image," in *VI'96*, Toronto, May 1996, pp. 159 – 166.
- [8] Paolo Favaro and Stefano Soatto, "A variational approach to scene reconstruction and image segmentation from motion-blur cues," in *CVPR'04*, Washington, DC, Jun.27 - Jul.26 2004, vol. 1, pp. 631 – 637.
- [9] Hailin Jin, Paolo Favaro, and Roberto Cipolla, "Visual tracking in the presence of motion blur," in *CVPR'05*, San Diego, CA, June 20-26, 2005, vol. 2, pp. 10 – 25.
- [10] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 5, pp. 564 – 577, May 2003.
- [11] William T. Freeman and Edward H. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 9, pp. 891 – 906, 1991.
- [12] Vladimir N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, 1995.