Kernel-based metric for performance evaluation of video infrared target tracking

Jianguo Ling,[a,b] Erqi Liu,[b] Haiyan Liang,[b] and Jie Yang[a]
[a]Shanghai Jiaotong University, Institute of Image Processing and Pattern Recognition,
No. 800 Dongchuan Road, Shanghai 200240, China
E-mail: lingjianguo76@sjtu.edu.cn
[b]China Aerospace Science and Industry Corporation, Institute of the Second Academy, Beijing, 100854, China

Abstract. A kernel-based metric measuring tracking reliability that is based on discriminative components of a kernel target model and kernel mutual information is presented. The discriminative components of the kernel target model are selected by computing the log-likelihood ratios of class-conditional sample densities of these components from a target region and background sampled region. The components selection process is embedded in a metric with kernel mutual information of the target regions of the initial frame and current frame in video infrared target tracking for online evaluation of the tracking reliability. Experimental results have shown that the metric can effectively characterize target tracking results as good or bad.
[DOI: 10.1117/1.2207810]

Subject terms: infrared target tracking; kernel target models; components selection; kernel mutual information; tracking reliability metrics.

Paper 050969LR received Dec. 12, 2005; revised manuscript received Mar. 4, 2006; accepted for publication Mar. 31, 2006; published online Jun. 2, 2006.

1 Introduction

Tracking reliability evaluation of a tracking algorithm is an important issue because it can guide the design of a good tracker. A variety of algorithms for measuring reliability are presented to improve the robustness of the tracking process.[1–4] Several feature-points-based metrics are proposed in Ref. 1 for analysis of partial and total occlusion in video tracking. Erdem et al. introduced other metrics based on the color and motion differences.[2] However, these feature-points and color-based metrics are not fit for evaluating the tracking performance of video infrared target tracking because the extracted feature points and color information of the target region are not reliable in infrared images. The infrared sequences are extremely noisy due to rampant systemic noise or color noise sources incurred by the sensing instrument and the noise from the environment.[5] The aim of this letter is to design a proper metric to evaluate the performance quantitatively of infrared target tracking while utilizing the intensity values information discriminatively and avoiding extracting the feature points of the target region with a kernel-based method.

2 Tracker Evaluation Metric

A kernel-based target tracking approach, such as mean shift algorithm,[6] is a commonly used method in the tracking field. Let $\{x_i\}_{i=1\ldots n}$ be the normalized pixel locations in the target region with center c in the current frame. The function $b:R^2\rightarrow\{1\ldots m\}$ (m-bin histogram is used) associates to the pixel at location x_i the index $b(x_i)$ of its bin in the quantized feature space. The kernel density estimation of the feature $u=1\ldots m$ in the target region is computed as[6]

$$q_u = C\sum_{i=1}^{n} k\left(\left\|\frac{x_i - c}{h}\right\|^2\right)\delta[b(x_i) - u], \tag{1}$$

where δ is the Kronecker delta function, C is the normalization constant, $k(\bullet)$ is the common profile used in corresponding feature domain, and h is the kernel bandwidth. Thus we have the target model

$$q = \{q_u\}_{u=1\ldots m}, \quad \sum_{u=1}^{m} q_u = 1. \tag{2}$$

We can obtain the target candidates in the same way, and the target location in the current frame can be obtained by optimizing the similarity function of the target model and target candidates.

It is unavoidable that some background parts exist in the located target region when we don't use a contour-based method in which tracking is achieved by evolving the contour frame to frame.[7] To evaluate the tracking performance, we seek discriminative components of the tracking model. The selected components of the tracking model are the components that can best describe the tracked target. A rectangular set of pixels covering the target is chosen to represent the target pixels, and an outer surrounding ring set of pixels is chosen to form the sampled background. Given a certain feature u, let q_u and o_u be kernel density estimation values of feature u for pixels in the target region and background sample, respectively. The log-likelihood ratio of the feature u is given by[8]

$$L(u) = \log\frac{\max(q_u, \xi)}{\max(o_u, \xi)}, \quad u = 1\ldots m, \tag{3}$$

where ξ is a small value (we set it to 0.001) that prevents dividing by zero or taking the log of zero. Based on the log-likelihood ratio, we select the components q_u of the tracking model when

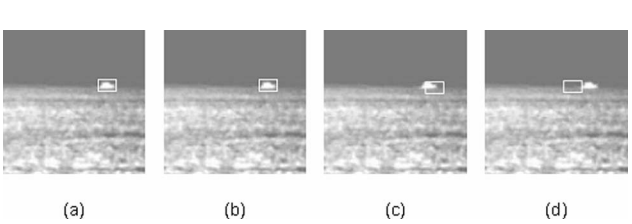

Fig. 1 Ship target in the sea-sky background: (a) initial frame; (b) correct location, E_k=1; (c) only part of the target is located, E_k=0.843; (d) target missing, E_k=0.

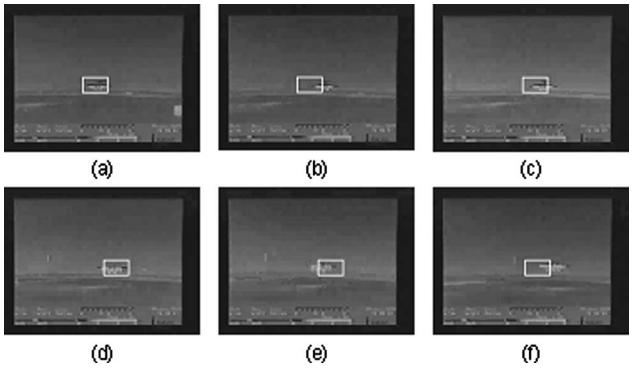

Fig. 2 Plane sequence and its different located target regions: (a) frame 8, E_k=0.935; (b) frame 18, E_k=0.362; (c) frame 32, E_k=0.562; (d) frame 70, E_k=0.904; (e) frame 81, E_k=0.634; (f) frame 95, E_k=0.245.

$$L(u) > \tau, \tag{4}$$

where τ is a threshold determined by our prior knowledge of the target. From Eq. (4), we know that the selected components are the components that can best describe a target. This is because high values of $L(u)$ denote a higher kernel density of feature u than that of the sampled background, and the pixels of feature u in the target region are thus parts of the real target. In order to strengthen the selection process, a background-weighted method of the kernel density estimation of the target region is also used.[6] Therefore, a cost function S_k is defined to embody the lost information of the selected discriminative components of the initial target region during the tracking process:

$$S_k = \frac{N - N_k}{N}, \tag{5}$$

where N is the number of pixels in the target region that construct the selected components in the initial frame and N_k is the number of pixels in the target region that construct these components in frame k. Large values of S_k are an indication of the information decrease of the selected components of the initial target model.

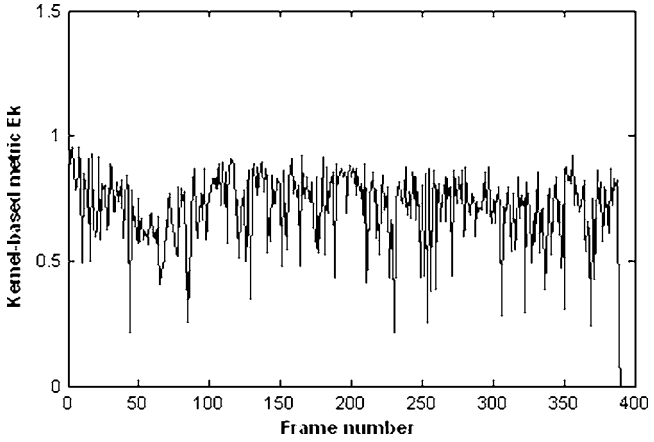

Fig. 3 Values of kernel-based metric against frame number for ship sequence.

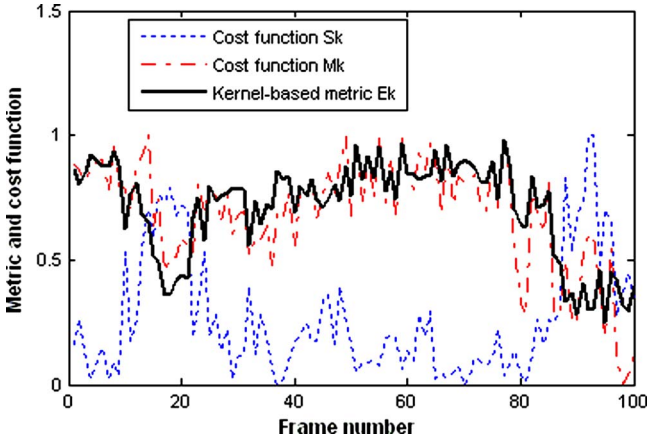

Fig. 4 Values of kernel-based metric and cost functions against frame number for plane sequence.

For two discrete valued random vectors X and Y with marginal probability mass function $p(x), p(y)$ and joint probability function $p(x,y)$, the mutual information between them is defined as

$$I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}. \tag{6}$$

Given the kernel density estimations q_u of feature u and q_v of feature v of the initial and current target region, respectively, the marginal probability mass functions $p(u)$ and $p(v)$ are given by

$$p(u) = q_u, \quad p(v) = q_v, \tag{7}$$

where u and v are the feature values in the quantized feature space. The joint probability $p(u,v)$ between the two kernel density estimations is calculated as

$$p(u,v) = p(u)p(v|u), \tag{8}$$

where $p(v|u)$ is a conditional probability of v while observing u. We place a one-dimensional kernel centered on u and kernel values are used as $p(v|u)$. For example, conditional probability $p(v|u)$ with a Gaussian kernel is given by

$$p(v|u) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left[\frac{(u-v)^2}{2\sigma^2}\right], \tag{9}$$

where σ is the standard deviation of the Gaussian kernel. Here, we define kernel mutual information as

$$I(U,V) = \sum_{u=1}^{m} \sum_{v=1}^{m} p(u,v) \log \frac{p(u,v)}{p(u)p(v)}. \tag{10}$$

Therefore, a cost function M_k is defined based on kernel mutual information to evaluate how much information of the initial target region holds in frame k and it is given by

$$M_k = \frac{I(U,V)}{\max(H_1,H_2)}, \tag{11}$$

where H_1 and H_2 are the entropies of the target regions of the initial frame and current frame, respectively, in the quantized feature space, which are given by

$$H_1 = \sum_{u=1}^{m} p(u)\log u, \quad H_2 = \sum_{u=1}^{m} p(v)\log v, \tag{12}$$

where $\max(H_1,H_2)$ is the maximum information entropy value of the two compared entropies. Because $p(u)$ and $p(v)$ are the marginal probability mass functions, $\max(H_1,H_2)$ is also the maximum of the kernel mutual information. So,

$$0 \le M_k \le 1. \tag{13}$$

A single metric can be obtained to evaluate the tracking performance by combining the information of the discriminative components of the kernel target model in frame k and kernel mutual information cost function defined above as follows:

$$E_k = c_1(\alpha S_k + M_k + c_2), \tag{14}$$

where the constants c_1, α, and c_2 are chosen to satisfy

$$0 \le E_k \le 1. \tag{15}$$

In our work, the constants c_1, α, and c_2 are chosen in the same way as the feature-points-based mutual information metric presented in Ref. 1, that is, $c_1=0.5$, $\alpha=-1$, $c_2=1$. This means that when the tracked target is lost ($S_k=1, M_k=0$), E_k achieves the minimum value 0 while the target is entirely accurate located ($S_k=0, M_k=1$), E_k achieves the maximum value 1. The kernel-based metric E_k is a measure of the tracking performance of a tracking process. A large value of E_k represents a good tracking performance and reliable tracker output in the current frame.

3 Experimental Results

Different tracked regions of a standard mean shift tracker[6] of a 400-frame infrared ship sequence (the size of each frame is 128×128 pixels) and a 100-frame infrared plane sequence (the size of each frame is 160×120 pixels) are evaluated by the kernel-based metric. The intensity space is taken as a feature space and it is quantized into 64 bins. We implement the tracking algorithm with the metric output in VC++6.0 on a Pentium 4 platform and the current implementation of the tracking algorithm with the metric output is capable of tracking at 15 and 17 frames/s of the ship sequence and plane sequence, respectively. The kernel-based metric is adopted properly in this situation to evaluate the tracking process after a top-hat transform preprocessing in the target region. Some representative frames from these sequences are shown in Figs. 1 and 2, respectively. The rectangle shown in the infrared image indicates the located target region. The outputs of the metric of different located target regions represent quantitatively the amount of information of the selected target that the tracker can capture in different frames. The variations of the tracking performance denoted by the proposed metric for various image frames in different sequences are also shown in Figs. 3 and 4.

The variable parameters c_1, α, and c_2 in Eq. (14) are chosen to satisfy the requirement $0 \le E_k \le 1$ and their values are kept constant throughout the experiments. From Fig. 4, we find that the variation of the cost function M_k is almost the same as that of the proposed metric and the cost function S_k has a similar curve to them but with reverse variation because it evaluates the lost information of the selected components of the initial target model during the tracking process. In fact, we can treat the cost functions identically by assigning the variable parameters as $c_1=0.5, \alpha=-1$, and $c_2=1$ in most cases. Notice that for abrupt appearance changes (for example, the size of the tracked target will abruptly increase when one target across another), the metric will be ineffective because the tracker output is not reliable in this situation. Since such abrupt changes are transient, the metric works effectively again after that. As we know, a robust tracker with a proper model update method is less sensitive to the appearance changes and can track the target even though the tracked target model is largely different than the initial target model. Here, N in Eq. (5) andH_1 in Eq. (11), which are computed from the target region of the initial frame, are also updated when a model update method is implemented.

4 Conclusions

This paper has presented a kernel-based metric to evaluate the reliability of the tracking process. The metric is constructed with a kernel method by embodying the information flow of the selected discriminative components of the kernel target model and kernel mutual information of the target regions of the initial frame and current frame. Future research will attempt to design a more suitable kernel target model to complement the kernel-based metric.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments. This work is partially supported by the Aeronautics Science Fund (China) under Grant No. 04F57004.

References

1. E. Loutas, I. Pitas, and C. Nikou, "Entropy-based metrics for the analysis of partial and total occlusion in video object tracking," *IEE Proc. Vision Image Signal Process.* **151**(6), 487–497 (2004).
2. C. E. Erdem, A. M. Tekalp, and B. Sankur, "Metrics for performance evaluation of video object segmentation and tracking without ground-truth," *Proc. ICIP* **2**, 69–72 (2001).
3. C. E. Erdem, A. M. Tekalp, and B. Sankur, "Video object tracking with feedback of performance measures," *IEEE Trans. Circuits Syst. Video Technol.* **13**(4), 310–324 (2003).
4. P. Villegas and X. Marichal, "Perceptually-weighted evaluation criteria for segmentation masks in video sequences," *IEEE Trans. Image Process.* **13**(8), 1092–1103 (2004).
5. J. Wei and I. Gertner, "Discrimination, tracking, and recognition of small and fast moving objects," *Proc. SPIE* **4726**, 253–266 (2002).
6. D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(5), 564–577 (2003).
7. A. Yilmaz, X. Li, and M. Shah, "Contour-based object tracking with occlusion handling in video acquired using mobile cameras," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(11), 1531–1536 (2004).
8. T. C. Robert, Y. X. Liu, and L. Marius, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(10), 1631–1643 (2005).