Power System Technology
ISSN:1000-3673

Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

Object Tracking based on Compressive Sensing Using Gabor Filters

Reza abdi payamani[1]
[1]MSc in Electrical Engineering (Electronics), Shahid Chamran University of Ahvaz
Reza.a.payamani@gmail.com

Abstract

The development of efficient and effective appearance models for object tracking is challenging due to factors such as camera angle changes, illumination variations, occlusion, and motion blur. Existing online tracking algorithms often update their models with samples from observations in the current frames. When these adaptive appearance models are data-dependent, there is insufficient data for online algorithms to learn at the beginning. Moreover, online tracking algorithms frequently face the problem of drift. Due to self-learning, there is a probability of adding misallocated samples to the appearance model, which degrades it. In this paper, we propose a simple yet effective tracking algorithm with an appearance model based on feature extraction using a data-independent Gabor filter. The proposed appearance model non-adaptively preserves the structural feature space of the original image of the object. A Gabor filter is designed to effectively extract features for the appearance model when applied to the image. This filter extracts both foreground and background features. The tracking task is formulated as a binary classification problem, utilizing a Support Vector Machine (SVM) classifier with online updates. The proposed tracking algorithm operates in real-time and outperforms other advanced methods in terms of accuracy and efficiency on challenging sequences. The algorithm was tested on six datasets named David, Bolt, Pedestrian, Goat, Cyclist, and Chase. The results from these experiments demonstrate the superiority of this method compared to other approaches.

Keywords: Compressive Sensing, Sparse, Feature Extraction, Object Tracking, Gabor Filter.

Introduction

One of the primary objectives of image processing and computer vision is to enable computers and robots to replicate basic tasks akin to human vision in understanding motion and scenes. To achieve this goal, extensive efforts have been made in the field of object tracking, a crucial and challenging topic in image processing. In numerous video surveillance applications and, more broadly, in image processing applications, detection and tracking—considered low-level processing tasks—are necessary. The system's output, through detection and subsequently tracking objects, serves as input for higher-level processes such as motion determination and interpretation, object counting, behavior recognition, and more. Object tracking, within a sequence of images or video, aims to find or identify a specific target to automatically recognize and follow it. Tracking can be defined as the task of estimating the trajectory of an object when it moves within a scene. In other words, we aim to determine the object's position in the image

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 Revised: 20-04-2024 Accepted: 13-06-2024

at any given time. The core of object tracking is the robust estimation of the target's motion state (location, orientation, size, etc.) in each frame of the image sequence.

There are three essential stages in video processing to consider in object tracking: detecting the desired moving objects; tracking them from one frame to the next; and using object tracking analysis to recognize their behavior. Object tracking in images or visual tracking is one of the most critical tools for intelligent and advanced video analysis. Therefore, numerous papers have been presented to develop tracking algorithms for various scenarios [1]. Visual tracking, due to its extensive practical applications, especially efforts to reduce computation and thus execution time, minimize computer memory usage, and maintain high accuracy, continues to attract many researchers, with growing interest in[2] the field.
In 1975, the mean shift method was first introduced by Fukunaga and Hostetler, initially used for analyzing a complex feature space and identifying clusters of different shapes within it . This algorithm was initially applied to image segmentation tasks but was adapted for object tracking by Comaniciu and Meer in 2002. In this approach, the target object is identified in the image based on its features, the algorithm is applied, and it calculates the object's center of mass, weighted mean of features, and other statistical parameters. When the object moves in the next frame, the algorithm is re-applied to the image iteratively without re-extracting features or re-identifying the object, automatically locating the object in the current frame.
In 2002, a continuous adaptive mean shift method, a parameter-free approach for finding local maxima in a probability density distribution using gradient ascent, was employed . This algorithm starts by selecting an appropriate color model for the object to be tracked and obtaining the histogram of the object's color type. This histogram is stored for identifying the object in subsequent frames. [3]Each pixel in the image is then assigned a specific probability based on this histogram to indicate its relevance to the object.
In 2004, state-space-based trackers were introduced by Okuma and colleagues, demonstrating that tracking an object involves defining a model that effectively tracks such movements . The most significant trackers in this category are those based on Kalman filters and particle filters[4] **.**
In [5], object trackers based on optical flow were introduced. Generally, these trackers assume that an object maintains constant illumination intensity while moving within the image. This means that as the object moves from one side of the image to the other, its brightness does not change. Optical flow-based methods are highly diverse and have various classifications. They have been used in numerous applications, including video compression (some MPEG encoders are based on optical flow) [6], obtaining the three-dimensional shape of moving objects using a single camera [7, 8], deriving global image motion in satellite images [8], and eliminating hand shake effects in video cameras.
In [9-15], tracking methods based on compressive sensing theory were employed. This theory was first introduced by Tao and Candes in 2005 [9, 10]. Furthermore, Hanke and colleagues utilized this approach for visual tracking in 2011 [11]. In 2012, Zhang and colleagues applied this method for real-time tracking [12], and in 2015, they proposed adaptive real-time tracking using this method [13]. Additionally, Zhang and colleagues used this approach for fast tracking in 2014 [14]. Zhao and his team also implemented tracking using compressive sensing theory under the title of adaptive scale compressive tracking in 2016 [15]. Compressive sensing theory

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

suggests that if the feature space dimensions are sufficiently large, these features can be mapped randomly to a lower-dimensional feature space that contains enough information to reconstruct the original high-dimensional features. Therefore, in tracking problems, this point is crucial since reducing the feature space dimensions significantly impacts processing speed.
Despite advancements in tracking, there are still issues in this field, as the performance of existing algorithms is not always satisfactory. For instance, illumination changes can dramatically affect the local or global appearance of the target, and when background clutter resembles the object, it can result in significant tracking errors. Figure 1 shows some examples of these challenges that complicate the design of an efficient tracking algorithm. By considering assumptions such as smooth object motion without sudden movements or constant velocity or acceleration, the tracking problem can be simplified. Moreover, prior knowledge about the number, size, and shape of objects can be used to simplify the problem; however, these assumptions can only be applied in certain applications, preventing the development of a comprehensive algorithm..

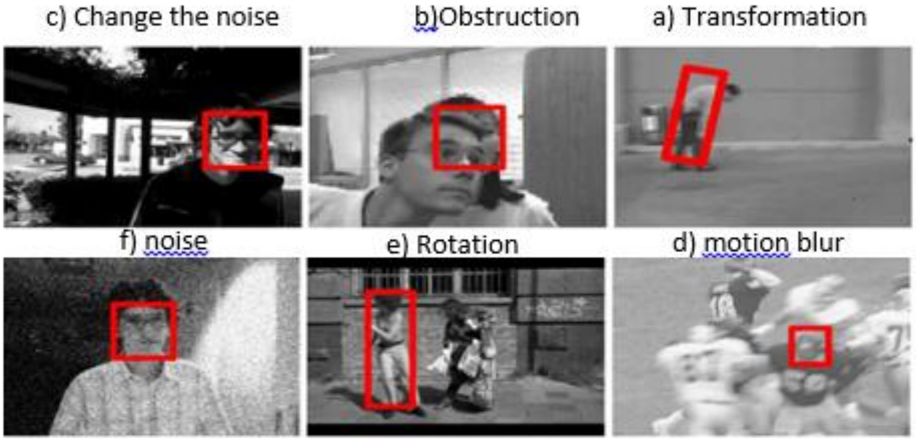

Figure (1): Examples of Tracking Challenges [16].

By considering assumptions such as smooth object motion without sudden movements or the assumption of constant velocity or acceleration, the object tracking problem can be simplified. Additionally, prior knowledge about the number, size, and shape of objects can be used to simplify the problem. While these assumptions help in simplifying the problem, they hinder the development of a robust and efficient algorithm applicable across various scenarios.
Therefore, this paper presents a robust tracking method that integrates the tracking problem with Gabor filters and a Support Vector Machine (SVM) classifier to create the appearance model. Our goal is to develop a robust tracking algorithm that not only provides logical answers to tracking queries but also addresses existing challenges.
This paper is organized as follows: In Section 2, the overall design and details of the proposed method for achieving a robust and efficient tracking algorithm using Gabor filters and SVM are presented. This method is applied to various videos, each containing different tracking challenges across multiple scenarios, and the results are discussed in Section 3. Finally, the conclusion is provided in Section

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

2. Proposed Method

In this section, we present a robust tracking method that integrates the tracking problem with Gabor filters and a Support Vector Machine (SVM) classifier to create the appearance model. Gabor filters have garnered significant attention because they can approximate the characteristics of certain cells in the visual cortex of some mammals [17]. Additionally, these filters possess desirable localization properties in both the spatial and frequency domains, making them suitable for texture segmentation problems. Gabor filters are widely used in various applications such as texture segmentation, object detection, document analysis, edge detection, network recognition, image coding, and representation [18, 19].

A Gabor filter can be considered as a sinusoidal plane of specific frequency and orientation modulated by a Gaussian envelope. These filters are available in one-dimensional and two-dimensional forms, with the two-dimensional filters being utilized in this paper. Gabor filters are highly effective in feature extraction, and by tuning their parameters, one can achieve optimal results in feature extraction along borders or edges for the tracking problem. Therefore, due to the effective performance of Gabor filters in feature extraction, it was deemed necessary to include them in the proposed algorithm. The application of this filter on the image, along with other algorithm steps, is illustrated in Figure (2**).**

Figure (2): Structure of the Proposed Algorithm

The workflow of the algorithm is as follows: First, an image is captured from the desired video or film. After applying necessary preprocessing steps, the Gabor filter is employed to extract image information, which constitutes the image features. These features are then fed into an SVM classifier to construct the appearance model for tracking. To address changes in size or color of the target object, the application of the object image on the filter bank has been utilized. Moreover, the most crucial step in designing the filter bank is to have filters with the following properties [20]:

This model is single•

- Features at the low level such as colors and lines
- Features at the middle level such as shapes and texture regions
- Features at the high level such as activities and semantic knowledge
- Can be used in both perceptual and conceptual levels
- Resistant to object rotation and size change (enlargement/shrinkage)
- Provides a complete perceptual understanding of the object

The resulting feature matrix should be sufficiently small [20, 21].

For this purpose, we use the Gabor filter bank as described below.

$$g(x,y,\lambda,\theta,\psi,\sigma,\gamma) = \exp(\frac{x'^2+\gamma^2 y'^2}{2\sigma^2})\exp(i\left(2\pi\frac{x'}{\lambda}+\psi\right))\frac{x'^2+\gamma^2 y'^2}{2\sigma^2})\exp(i\left(2\pi\frac{x'}{\lambda}+\psi\right))$$

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

(3-2)
Real

g(x,y,λ,θ,ψ,σ,γ) = exp($\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}$)cos $\left(2\pi\frac{x'}{\lambda}+\psi\right)$ $\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}$)cos $\left(2\pi\frac{x'}{\lambda}+\psi\right)$

(3-3) Hypothetical

g(x,y,λ,θ,ψ,σ,γ) = exp($\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}$)sin $\left(2\pi\frac{x'}{\lambda}+\psi\right)$ $\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}$)sin $\left(2\pi\frac{x'}{\lambda}+\psi\right)$ (3-4)

There where

$x' = xcos\theta + ysin\theta$ $x' = xcos\theta + ysin\theta$ (3-5)

و

$y' = -xsin\theta + ycos\theta$ $y' = -xsin\theta + ycos\theta$ (3-6)

The filter bank is applied to the image in 4 scales and 4 orientations, resulting in 16 feature maps with the same size as the input image. Then, each feature map is divided into 16 regions, and the average feature value for each region is calculated. This process generates the desired filter bank, as illustrated in Figure (3) [20, 21].

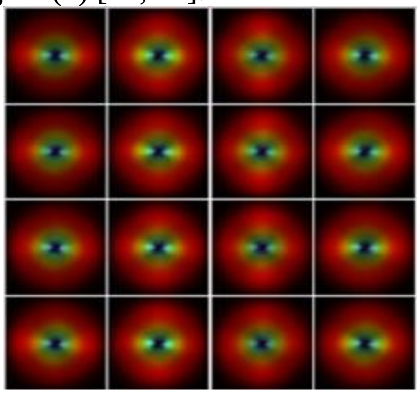

Figure 3: Output of the Designed Gabor Filter Bank

Applying this filter bank to the object image yields 256 real numbers, which are then sent to the SVM classifier. The SVM classifier is highly suitable for creating an appearance model for object tracking. Additionally, one notable advantage of this classifier over other neural network classifiers is its high processing speed, which is crucial in tracking tasks. The faster the processing speed in the algorithm stages, the more optimal the solution will be. In the following, two metrics are proposed for evaluating the tracking algorithm. The first metric is the success rate (SR), which indicates successful tracking if it exceeds 50%. The success rate metric is defined by equation (4-1):

$\mathrm{score} = \frac{\mathrm{area}(\mathrm{ROI_T} \cap \mathrm{ROI_G})}{\mathrm{area}(\mathrm{ROI_T} \cup \mathrm{ROI_G})}$ $\mathrm{score} = \frac{\mathrm{area}(\mathrm{ROI_T} \cap \mathrm{ROI_G})}{\mathrm{area}(\mathrm{ROI_T} \cup \mathrm{ROI_G})}$

(4-1)

That $ROI_T ROI_T$ The desired range for tracking and $ROI_G ROI_G$ The desired range is the reference range. The second criterion is the center location error, which is defined as the Euclidean distance between the center position of the tracked object and the manually labeled

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

reference range. The FCT algorithm has been tested on several datasets and has shown relatively good performance compared to other tracking algorithms. The proposed algorithm has been tested on six datasets named David, Bolt, Pedestrian, Goat, Cyclist, and Pursuit. The results for each dataset are presented in the next section, followed by an explanation of these results. In the third section, the results of this algorithm are compared with the results of other existing methods

.

3- Results

In the second section, details and relationships of the Gabor filter for feature extraction using the proposed method were presented. According to the provided explanations, this filter has very good performance in feature extraction. By adjusting its parameters, an optimal solution can be achieved in the process of feature extraction at edges or boundaries for the tracking problem. In this section, we evaluate the results obtained from applying the proposed method for obstruction and state change, as well as rotation and sudden movement cases.

3-1- Obstruction and State Change

In the pedestrian sequence shown in Figure (4), the target object is obstructed heavily. Additionally, due to low visibility and accuracy for tracking the object, it poses a significant challenge. The proposed algorithm handles obstruction and state changes well, whereas the FCT algorithm [14] loses track of the target object in multiple frames. Figure (4) shows the tracking result in frame 80 of the pedestrian, which, as stated in Tables (1) and (2), achieved a success rate of 88% and a low center location error of 4.

Figure (4): Tracking results for frame 50 of the pedestrian.

Moreover, in the Bolt sequence, multiple objects in a scene with rapid appearance changes due to rapid shape change and movement are displayed. The proposed algorithm shows better performance compared to the FCT algorithm [14] in terms of success rate and center location error. Figure (5) depicts the result of tracking the famous sprinter Bolt, which was successful with a success rate of 99 and a center location error of 9**.**

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

Figure (5): Tracking results for frame 20 of Bolt.

In the sequence of the goat, the proposed algorithm demonstrates better performance in terms of success rate and center location error compared to the FCT algorithm [14]. Figure (6) shows the tracking result with a success rate of 79 and a center location error of 18**.**

Figure (6): Tracking results for frame 60 of the bicycle.

3-2- Rotation and Sudden Movement

In the sequence of the cyclist, when a complete rotation occurs and sudden movement happens, this algorithm outperforms the FCT algorithm [14] in both success rate and center location error metrics. Figure (7) depicts the tracking of the cyclist, with a success rate and a center location error of 40 and 10, respectively.

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 Revised: 20-04-2024 Accepted: 13-06-2024

Figure (7): Tracking results for frame 30 of the biker.

The target object in the car sequence in Figure (8) undergoes a sudden 360-degree rotation, and the proposed algorithm demonstrates better performance compared to the FCT algorithm. This type of tracking, conducted on the car, achieves a success rate and error percentage of 85 and 7, respectively.

Figure (8): Tracking results for frame 100 of chasing.

For the David sequence, when the person exits a dark room, their appearance changes due to the change in lighting and posture. The proposed algorithm provides an appropriate response to this change. This tracking also indicates strong performance with a success rate of 99 and a positional error of 9

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 *Revised: 20-04-2024* *Accepted: 13-06-2024*

Figure (9): Tracking results for frame 80.

The software used for this task is MATLAB version 2016a. The tracking results from the six datasets tested with the proposed algorithm are fully collected in Tables (1) and (2). Subtable (1) indicates the best performance metric compared to the success rate of this algorithm with algorithm.

Table (1): Success Rate Performance [14]

Videos	Proposed algorithm	FCT [14]
bolt	95	94
Goat	79	77
David	99	98
Biker	40	35
Chasing	81	79
Pedestrian	85	83

Table (2) represents the average tracking error, which, through experiments on several datasets, shows that the proposed algorithm achieves the best results in most video sequences compared to other methods [14]. Additionally, the proposed tracker is more accurate compared to other trackers.

Table (2) Center Location Error

Videos	proposed algorithm	FCT [14]
bolt	3	10
Goat	14	18
David	8	11
Biker	11	12
Chasing	7	10
Pedestrian	7	7

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 Revised: 20-04-2024 Accepted: 13-06-2024

4- Conclusion and Recommendations

The six datasets tested with the proposed algorithm exhibit excellent quality. These datasets were also tested with the algorithm in [14], which yielded good results against tracking challenges arising from factors such as sudden changes, shape variations, rotations, occlusions, and lighting changes. In comparison with other tracking methods, the proposed algorithm demonstrated superior performance. Furthermore, it was observed that the proposed method significantly outperformed the algorithm in [14] in tracking these multiple datasets. In this paper, we proposed a robust tracking algorithm with a appearance model that preserves the structure of the original image space. Feature extraction is a crucial issue in the field of tracking, and various algorithms have been proposed for this purpose. The proposed algorithm in this paper incorporates a Gabor filter to effectively extract background and foreground features from the target. The Gabor filter, based on the provided descriptions regarding its size and orientation, demonstrates excellent performance in feature extraction at edges, making it the preferred choice for this algorithm over other filters. The tracking task is set up as a binary classification problem, and the classifier used in this algorithm performs better than other classifiers in neural networks due to its high processing speed. Through multiple experiments on various datasets, the proposed algorithm has shown better performance compared to other advanced tracking algorithms and has provided better solutions to most existing challenges.

Reference

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, p. 13, 2006.

[2] G. Bradsky, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, vol. 2, pp. 1-15, 1998.

[3] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *European conference on computer vision*, 2004, pp. 28-39.

[4] Z. Chen, "Bayesian filtering: From Kalman filters to particle filters, and beyond," *Statistics*, vol. 182, pp. 1-69, 2003.

[5] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *International Journal of Computer Vision*, vol. 92, pp. 1-31, 2011.

[6] I. E. Richardson, *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*: John Wiley & Sons, 2004.

[7] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM computing surveys (CSUR)*, vol. 27, pp. 433-466, 1995.

[8] T. Brox and J. Malik, "Large displacement optical flow: descriptor matching in variational motion estimation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, pp. 500-513, 2011.

[9] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE transactions on information theory*, vol. 51, pp. 4203-4215, 2005.

[10] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE transactions on information theory*, vol. 52, pp. 5406-5425, 2006.

[11] H. Li, C. Shen, and Q. Shi, "Real-time visual tracking using compressive sensing," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, pp. 1305-1312.

Power System Technology
ISSN:1000-3673
Received: 15-03-2024 ***Revised: 20-04-2024*** ***Accepted: 13-06-2024***

[12] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *European conference on computer vision*, 2012, pp. 864-877.
[13] W.-Z. Zhang, J.-G. Ji, Z.-Z. Jing, W.-F. Jing, and Y. Zhang, "Adaptive real-time compressive tracking," in *Network and Information Systems for Computers (ICNISC), 2015 International Conference on*, 2015, pp. 236-240.
[14] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, pp. 2002-2015, 2014.
[15] P. Zhao, S. Cui, M. Gao, and D. Fang, "Scale adaptive compressive tracking," *SpringerPlus*, vol. 5, p. 849, 2016.
[16] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM transactions on Intelligent Systems and Technology (TIST)*, vol. 4, p. 58, 2013.
[17] Available: https://www.cs.utah.edu/~arul/report/node13.html
[18] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern recognition*, vol. 25, pp. 1479-1494, 1992.
[19] T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Efficient Gabor filter design for texture segmentation," *Pattern recognition*, vol. 29, pp. 2005-2015, 1996.
[20] J. R. Movellan, "Tutorial on Gabor filters," *Open Source Document*, 2002.
[21] A. K. Jain, N. K. Ratha, and S. Lakshmanan, "Object detection using Gabor filters," *Pattern recognition*, vol. 30, pp. 295-309, 1997.