Vehicle Tracking in Outdoor Environment Based on Curvelet Domain

Nguyen Thanh Binh(✉)

Faculty of Computer Science and Engineering,
Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam
ntbinh@cse.hcmut.edu.vn

Abstract. Vehicle tracking is a difficult part in intelligent traffic system. The images of vehicles on the streets, picked up from cameras, are usually in occlusion because of effecting outdoor environment such as lack light, weather, etc. Therefore, vehicle tracking is a challenging problem. This paper proposed a method for vehicle tracking in an outdoor environment. We use curvelet transform combined with object deformation of contour. The light of background may change from this frame to the other frame. The proposed algorithm has significantly improves the edge accuracy and reduces the wrong position of objects between the frames. For demonstrating the superiority of the proposed method, we have compared the results with the other methods.

Keywords: Curvelet transform · Vehicle tracking · Contour

1 Introduction

Vehicle tracking is a difficult part in intelligent traffic system, particularly for visual-based surveillance system. In the real world, an intelligent video surveillance system requires being fast and reliable. Tracking system designed consists of three function blocks: moving object detection, object classification and object tracking. The techniques of moving object detection used background subtraction, statistical models, temporal differencing, optical flow, etc [1]. The most popular methods to detect moving objects are based on background subtraction. This method is to create a background model that is quite similar to the real one. After that, they make differential operation with every frame of video and background image to set changing area as moving objects [2] such as: eigen backgrounds, median filter, mean filter, temporal median filter, Kalman filter and sequential kernel density approximation [3]. When the environment is crowded with scenes, the background is hard to model. The solution to this problem is to use optical flow method. In this method, the pixels compared in frame sequences for the pixel position are calculated based on the vector position [1]. However, its drawbacks are sensitive to noise and have high computational complexity.

For vehicle tracking, there are four main groups: feature-based tracking, model-based tracking, region-base tracking, contour-based tracking [5]. The commonly used

P.C. Vinh et al. (Eds.): ICTCC 2014, LNICST 144, pp. 360–369, 2015.
DOI: 10.1007/978-3-319-15392-6_34

methods are Kalman filter, Kanade-Lucas-Tomasi, mean-shift, particle filter, etc. However, most of these methods are complex, slow processing as the object is in occlusion. Vehicle tracking is to detect locating position of vehicles through consecutive frames [4] in the video. The video picked up from cameras, are usually in occlusion because of effecting outdoor environment such as lack light, weather, illumination variability, background noise, partial overlapping and occlusion, etc. In the last decades, although many different approaches have been proposed. Vehicle tracking is still a challenging problem.

In this paper, we propose a method to implement a vehicle tracking in outdoor environment using curvelet transform combined with object deformation of contour. For demonstrating the superiority of the proposed method, we have compared the results with the wavelet transform method and curvelet transform method. The rest of the paper is organized as follows: in section 2, we described the basic concepts on curvelet transform. Details of the proposed algorithm have been given in section 3. In section 4, the results of the proposed method for vehicle tracking have been shown and finally the conclusion in section 5.

2 Curvelet Transform

In this section we explain what curvelets are, how they are constructed, and what are their main properties are. Curvelets are basically 2D anisotropic extensions to wavelets that have a direction associated with them. Analogous to wavelets, curvelets can be translated and dilated. The dilation is given by a scale index j that controls the frequency content of the curvelet, while the translation is indexed by m_1 and m_2 in two dimensions.

The anisotropic scaling relation is a key difference between wavelets and curvelets. The parabolic scaling is also a key ingredient to prove that curvelets remain localized in phase-space (i.e., remain curvelet-like) under the action of the wave operator provided the medium is smoothed appropriately prior to propagation [6].

The idea of curvelets [7] is to represent a curve as a superposition of functions of various lengths and widths obeying the scaling law $width \approx length^2$. This can be done by first decomposing the image into subbands, i.e separating the object into a series of disjoint scales; then, each scale is analyzed by means of a local ridgelet transform.

Curvelets are based on multiscale ridgelets combined with a spatial bandpass filtering operation to isolate different scales. This spatial bandpass filter nearly kills all multiscale ridgelets which are not in the frequency range of the filter. In other words, a curvelet is a multiscale ridgelet which lives in a prescribed frequency band. The bandpass is set so that the curvelet length and width at fine scales are related by the scaling law $width \approx length^2$ and so the anisotropy increases with decreasing scale like a power law. There is a very special relationship between the depth of the multiscale pyramid and the index of the dyadic subbands. The side length of the localizing windows is doubled at every other dyadic subband, hence maintaining the fundamental property of the curvelet transform which says that elements of length about $2^{-j/2}$ serve for the analysis and synthesis of the jth subband $[2^j, 2^{j+1}]$.

Like ridgelets, curvelets occur at all scales, locations, and orientations as shown in Fig.1. However, while ridgelets have global length and variable widths, curvelets in addition to a variable width have a variable length and so a variable anisotropy does.

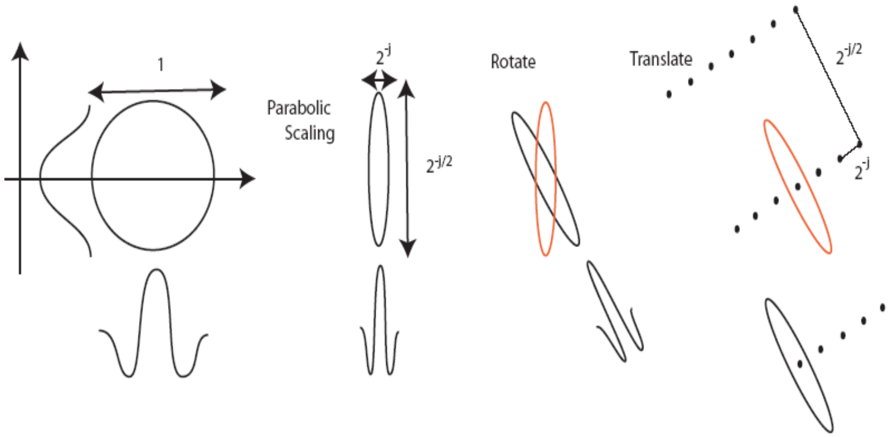


Fig. 1. Curvelets parameterized by scale, location, and orientation (source [8])

The length and width at fine scales are related by the scaling law *width* ≈ *length*^{2} and so the anisotropy increases with decreasing scale like a power law. Recent work [7] shows that the thresholding of discrete curvelet coefficients provided near optimal N-term representations of otherwise smooth objects with discontinuities C^2 along curves.

The curvelet dictionary is a subset of the multiscale ridgelet dictionary, which allows reconstruction. The "à trous" subband filtering algorithm [9] is especially well-adapted to the needs of the digital curvelet transform. The algorithm decomposes an n by n image $f(x, y)$ as a superposition of the form

$$f(x,y) = c_J(x,y) + \sum_{j=1}^{J} w_j(x,y) \tag{1}$$

where c_J is a coarse or smooth of the original image $f(x, y)$ and w_j represents the details of Im at scale 2^{-j}.

The discrete curvelet transform of a continuum function $f(x_1, x_2)$ makes use of a dyadic sequence of scales, and a bank of filters $(P_0 f, \Delta_1 f, \Delta_2 f, \ldots\ldots)$ with the property that the passband filter Δ_s is concentrated near the frequencies $[2^j, 2^{j+1}]$, e.g.,

$$\begin{aligned} \Delta_s &= \Psi_{2s} * f \\ \widehat{\Psi_{2s}}(\xi) &= \widehat{\Psi}(2^{-2s}\xi) \end{aligned} \tag{2}$$

In wavelet theory, one uses a decomposition into dyadic subbands $[2^j, 2^{j+1}]$. In contrast, the subbands used in the discrete curvelet transform of continuum functions have the nonstandard form $[2^j, 2^{j+1}]$. This is nonstandard feature of the discrete curvelet transform well worth remembering.

The basic process of the digital realization for curvelet transform is given as follows:

(1) *Subband Decomposition***.** We define a bank of filters P_0, $(\Delta_s, s \geq 0)$. The image f is filtered into subbands with à trous algorithm [9]

$$f \rightarrow (P_0 f, \Delta_1 f, \Delta_2 f, ...) \tag{3}$$

The different subbands $\Delta_s f$ contain details about 2^{-2s} wide.

(2) *Smooth Partitioning*. Each subband is smoothly windowed into "squares" of an appropriate scale.

$$\Delta_s f \rightarrow (w_Q \Delta_s f)_{Q \in Q_s} \tag{4}$$

where w_Q is a collection of smooth window localized around dyadic squares.

$$Q = [k_1 / 2^s, (k_1 + 1) / 2^s] x [k_2 / 2^s, (k_2 + 1) / 2^s] \tag{5}$$

(3) *Renormalization*. Each resulting square is renormalized to unit scale

$$g_Q = (T_Q)^{-1} (w_Q \Delta_s f), \qquad Q \in Q_s \tag{6}$$

where $(T_Q f)(x_1, x_2) = 2^s f(2^s x_1 - k_1, 2^s x_2 - k_2)$ is a renormalization operator.

(4) *Ridgelet Analysis***.** Each square is analyzed in the orthonormal ridgelet system. This is a system of basis elements p_λ making an orthonormal basis for $L^2(R^2)$:

$$\alpha_\mu = \langle g_Q, p_\lambda \rangle \tag{7}$$

We see that the performance of vehicle tracking will increase if the correct feature is selected for tracking algorithm. In our proposed work, we have used curvelet coefficients as a feature set.

3 The Vehicle Tracking Based on Curvelet Domain

In this section, we describe a method for moving vehicle tracking in outdoor environment using curvelet transform. The common approach for vehicles tracking consists of two periods: detecting vehicles and tracking vehicles as the following in figure 2. A video sequence contains a series of frames. Each frame can be considered as an image. The proposed method also consists of two periods. Firstly, curvelet coefficients are used for detection of vehicles. Secondly, we track vehicles in the sequence of frames. If an algorithm can track moving vehicles between two digital images, it should be able to track moving vehicles in a video sequence.

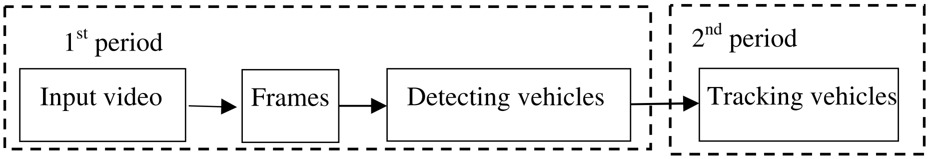


Fig. 2. The common approach for vehicles tracking system

The common approach for detection of vehicles consists of three steps: background modeling, foreground detection and data validation. We assume there are only two modes for each pixel in a single frame: background and foreground. The basic of background subtraction method is to compare the frame background with a threshold (T) which we are pre-defined. If the difference of a pixel is smaller than T, then it is background, otherwise, it is foreground. To detect objects, the curvelet coefficients and their statistical values were extracted as the features of object images. We define a discrete warped curvelet transform which goes across the region boundaries. We compute the image sample values in each region of the partition and also describe its implementation together with the inverse resampling. A warped wavelet transform with a sub-band filtering along the flow lines is implemented. At the boundaries, warped curvelet still have two vanishing moments. The curvelet coefficients of a discrete image are computed with a filter bank.

The step of pre-processing stage raws input video. Background modeling is the current background scene. We can know that the background is to acquire a background image which does not include any moving objects. Foreground detection checks if the input pixels are background or foreground. Foreground pixels are calculated by the Euclidean norm at the time t:

$$\left|PI_t(x,y) - BG_t(x,y)\right| > T \tag{8}$$

where, PI_t is the pixel intensity value, BG_t is the background intensity value at time t and T is the foreground threshold.

$$PI_t = \left[PI_{1,t} \dots PI_{n,t}\right]^T$$
$$BG_t = \left[BG_{1,t} \dots BG_{n,t}\right]^T$$

where, n in the number of image channels. The foreground threshold T is determined experimentally.

The goal of tracking is find position of vehicle between two adjacent frames. The tracking algorithm searches the position of the vehicles in the next frame according to the value of object boundary energy, which is computed from the three previous frames and direction of movement. Our algorithm is capable of tracking an object whose size changes within a range in the various frames. The processing of vehicles tracking system as the following in figure 3.

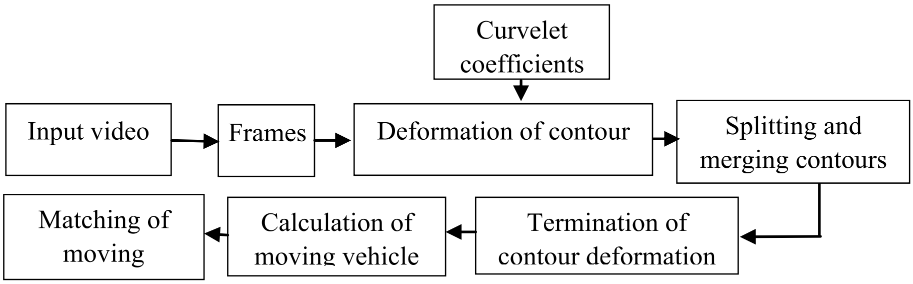


Fig. 3. The processing of vehicles tracking system

Firstly, the video input is divided into image sequence I(S), where S denotes a frame number, S =1, 2, 3, ….. Support $p_i(x_i(t), y_i(t))$ represent a contour model, where t is the number of iterations at each frame. If it is the first time, S is set to be 1. We set the number t = 0 at each frame. When S=1, we set an initial contour $p_i(x_i(0), y_i(0))$ for all moving objects [12].

Secondly, deformation of contour. In this step, we use the greedy algorithm [11] and move all contour points $p_i(x_i(t),y_i(t))$ (where i=1, 2,…,n) by minimizing a contour energy E_{snakes} and t = t + 1. The number moved points are stored in C_{move}. We detect object boundary by minimizing the following energy functional:

$$E_{snake} = E_{int}(p) + E_{image}(p) + E_{ext}(p)$$

where E_{int} is an internal energy associated with splines, E_{image} is an image energy such as edge potential and E_{ext} is an external energy associated with external forces.

Thirdly, splitting and merging contours. We divided a contour into multiple closed contours by detecting its self-crossings. The area E_{ext} of a contour model $p_i(x_i, y_i)$ (i=1, 2, 3,…,n) is defined as:

$$E_{ext} = \frac{1}{2}\sum_{i=1}^{n}\left[x_i(y_{i+1} - y_i) - (x_{i+1} - x_i)y_i\right] \tag{12}$$

where $p_{n+1}(x_{n+1}, y_{n+1}) = p_1(x_1,y_1)$. After that, we will be merging multiple contours. The process of merging two contours into a single one. To create new contour points, a new contour point between two adjacent points p_i and p_{i+1} must satisfy the condition $|p_{i+1} - p_i| > Dis_{TH}$ where Dis_{TH} is a threshold which maximums the distance between adjacent discrete points [12].

Fourthly, termination of contour deformation. If $C_{move} \le C_{TH}$ or $t_{\max} \le t$ then terminate the contour deformation at the image I(S) and proceed to step 5 else proceed to step 2. C_{TH} and t_{max} are predetermined thresholds.

Fifthly, calculation of moving vehicle. We calculate intensity histogram within the region surrounded by each contour model as the feature of moving objects.

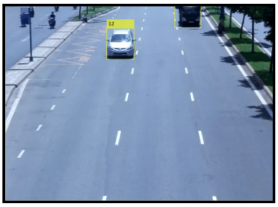
Finally, matching of moving vehicle. In each frame, cumulative intensity histogram $H_m(k)$ is computed within a region extracted by a converged contour model as a moving object. Set S = S+1 and proceed to step 1. The test cases will present in section 4.

4 Experiments and Evaluation

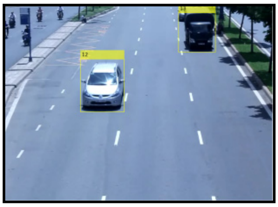
In this section, we applied the procedure described in section 3 to track the vehicles in a video. We apply hard thresholding coefficients after decomposition in curvelet domain. For the tracking period, the vehicle area is determined in the first frame and we find the vehicles in each frame of the video and from frame to frame. The proposed method has been done on many videos in PEST2001 dataset and the other videos picked up from cameras on the streets. Here, we report the results on some video clips. Our experimental approach is as follows. For demonstrating the superiority of the proposed method, we have compared the results with the wavelet transform (WT) method, curvelet transform (CT) method.

Our experiments are on vehicle video clips with the frame size 254 by 254. Most of videos are fuzzy videos. The proposed method processes this video clip at 24 frames/second. We have experimented on the video up to 3000 frames. Here, we report the results up to 2000 frames. Some results achieved as shown in figure 4 and figure 5.

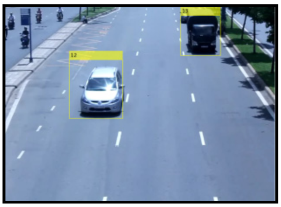
Frame 200



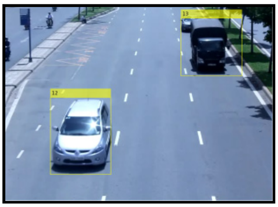
Frame 300



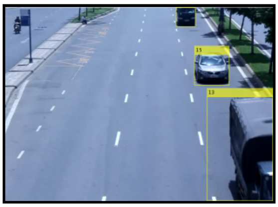
Frame 400



Frame 500



Frame 600



Frame 800

Fig. 4. Tracking in car video clips up to 800 frames

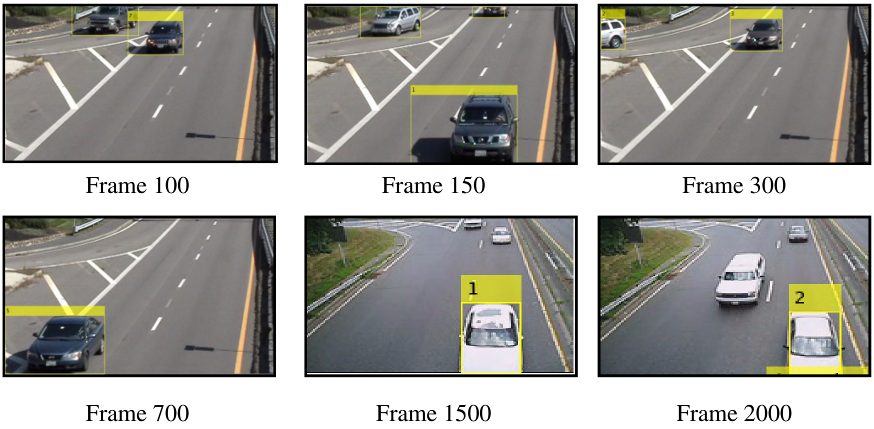


Fig. 5. Tracking in the other car video clips up to 2000 frames

Table 1. Comparing the vehicle tracking error of WT, CT and proposed method

Frame	**WT method**	**CT method**	**Proposed method**
50			
100			
150	false	false	
200			
250	false		
300			
350			
400			
450			
600			
650	false		
700			
750	false	false	
800		false	
850	false		
900			
950	false	false	false
1000	false		

In these figures, we observe that the proposed method performs well. Other experiments also show that the proposed method works well and better than the other ones. The proposed method is also more accurate.

Table 1 compares the vehicle tracking error between WT method, CT method and the proposed method. In table 1, we put false in the frames which vehicle tracking is not exactly. We have the percentage of the vehicle tracking error in the frames is 35% with WT method, 20% with CT and 5% with proposed method. As the above mention, the proposed method detecting object boundary is better than the other methods. Therefore, the results of the proposed method are good. As above mentioned, a dyadic segmentation of curvelet coefficients and choice of a polynomial flow inside each square define a curvelet B(T). Curvelets provide optimally sparse representations of objects. The representations are as sparse as if the object were turn out to be far more sparse than the decomposition wavelet of the object.

5 Conclusion and Future Work

Vehicle tracking is a difficult part in intelligent traffic system. The images of vehicles on the streets, picked up from cameras, are usually in occlusion because of effecting outdoor environment such as lack light, weather, etc. Vehicle tracking in these cases is not easy. In this paper, we have constructed a method for detecting and tracking of vehicles for outdoor environment. We use curvelet transform combined with object deformation of contour for tracking objects in outdoor environment. The proposed algorithm significantly improves the edge accuracy and reduces the wrong position of objects between the frames. However, if the quality of the frames in videos is very bad then the estimation ability is reduced. In the future work, we will compare the proposed method with the other methods and improve it in case of light change.

References

1. Shuigen, W., Zhen, C., Hua, D.: Motion Detection Based on Temporal Difference Method and Optical Flow Field. Second International Symposium on Electronic Commerce and Security **2**, 85–88 (2009)
2. Zhang, R., Ding, J.: Object Tracking and Detecting Based on Adaptive Background Subtraction. International Workshop on Information and Electronics Engineering **29**, 1351–1355 (2012)
3. Masafumi, S., Thi, T.Z., Takashi, T., Shigeyoshi, N.: Robust Rule-Based Method for Human Activity Recognition. International Journal of Computer Science and Network Security **11**(4), 37–43 (2011)
4. Ritika, G.S.S.: Moving Object Analysis Techniques in Videos - A Review. IOSR Journal of Computer Engineering **1**(2), 07–12 (2012). ISSN : 2278-0661
5. Zang, Q., Klette, R.: Object Classification and Tracking in Video Surveillance. In: Petkov, N., Westenberg, M.A. (eds.) CAIP 2003. LNCS, vol. 2756, pp. 198–205. Springer, Heidelberg (2003)
6. Smith, H.F.: A Parametrix Construction for Wave Equations with C1,1 Coefficients. Annales de l'institut Fourier, tome. **48**(3), 797–835 (1998)

7. Cand`Es, E., Donoho, D.: Curvelets: A Surprisingly Effective Nonadaptive Representation for Objects with Edges. In: Schumarker, L.L., et al. (eds) Curve and Surface Fitting: Saint-Malo 1999, pp. 105–120, Vanderbilt University Press, Nashville, TN (2000)
8. http://www.crm.umontreal.ca/imagerie06/pdf/montreallecture2.pdf. (last accessed: July 30, 2014)
9. Starck, J.L., Candès, E.J., Donoho, D.L.: The Curvelet Transform for Image Denoising. IEEE Transactions on Image Processing **11**(6), 670–684 (2002)
10. Araki, S., Yokoya, N., Iwasa, H., Takemura, H.: Real-time Tracking of Multiple Moving Objects using Split-and-Merge Contour Models Based on Crossing Detection. Systems and Computers in Japan **30**(9), 25–33 (1999)
11. Williams, D.J., Shah, M.: A Fast Algorithm for Active Contours. Proceedings of Third International Conference on Computer Vision, pp. 592–595 (1990)
12. Hoa, D.T.T., Binh, N.T.: Adaptive Object Tracking Technique based on Bandelet Domain in Outdoor Environment. In: Proceedings of International Conference on Advanced Computing and Applications, Vietnam (2014)