

OBSERVATIONAL DISCRETE LINES FOR THE DETECTION OF MOVING VEHICLES IN ROAD TRAFFIC SURVEILLANCE

Odil Mamaraufov

Department of Software of IT, Tashkent University of Information Technologies,
Tashkent, Uzbekistan

ABSTRACT

The paper deals with the development of mathematical models and algorithms for video processing in digital video surveillance systems to detect moving objects. The model and algorithm can be applied in video surveillance systems to identify moving objects on a surveillance area. The reduction of the calculations for the segmentation of video is considered and describes the algorithm of Observational discrete lines for the detection and tracking of moving objects is proposed in this article.

KEYWORDS

Video surveillance system, Video analysis, Visual monitoring, Motion detection, Tracing, Observation discrete lines

1. INTRODUCTION

This time the relevant tasks are the collection, analysis and processing of information on road safety, safety control, traffic on city streets and highways, road accidents and their study. Also relevant is the problem of determining the speed of traffic on motorways, registration of motor vehicles at intersections, posts and vehicle registration, car traffic and frequent road accidents. So important is the creation and implementation of video surveillance systems installed in roads and intersections.

For the video surveillance system is an actual resolution of contradictions between the quality of the generated image and hardware of existing channels of communication and data storage. In spite of the high capacity, the modern hard disks are not sufficient for storing large amounts of information for a long time, as it should be according to the specifications. Traditionally this contradiction is resolved by video compression with a noticeable decrease in their quality and loss of information. To improve the efficiency of video surveillance systems need to develop methods for video data compression without loss of information about the object of interest for the long-term storage and transfer in real time, high-quality images via communication channels with limited bandwidth [1].

Video moving object leads to the appearance of two phases-phase of adaptation to the current camera angle shooting and maintenance of objects of interest. The fixed camera shot scene with little changing background (relative sequence) with moving objects is of great practical use in observation systems (maintenance of vehicles, people), security systems, etc.

Analysis of video data in real time is possible, but a centralized information system Visual monitoring of video data processing problem as the stationary camera in the network. This problem is to reduce the amount of computation parallelization of calculations in the processing of video images. With the decline in computing for handling road traffic CCTV video images in real time, a method observational discrete line (ODL) for the detection of moving objects is proposed.

2. VIDEO ANALYSIS IN VIDEO SURVEILLANCE

Processing video data in video surveillance systems is essential significance of the system. The video information takes a lot of memory and processing support static frames.

Into the systems of video surveillance task of video analysis system the automatic acquisition of integrated information using algorithms for intelligent processing of video stream without operator security system. Video analysis is the development of the functions of video surveillance systems, allowing you to reduce the amount of recorded data. System identification and pattern recognition which has the ability to integrate with various components allows creating a structured database of events with dual video data. The main task of video analysis is to eliminate the human factor, minimizing the likelihood of a situation where an event may go unnoticed or misunderstood [2].

Computer vision methods are used in transmission protocols and standards for video storage. Recently, computer vision algorithms are used extensively in medicine. The main stage of video analysis solution is shown in Fig. 1.

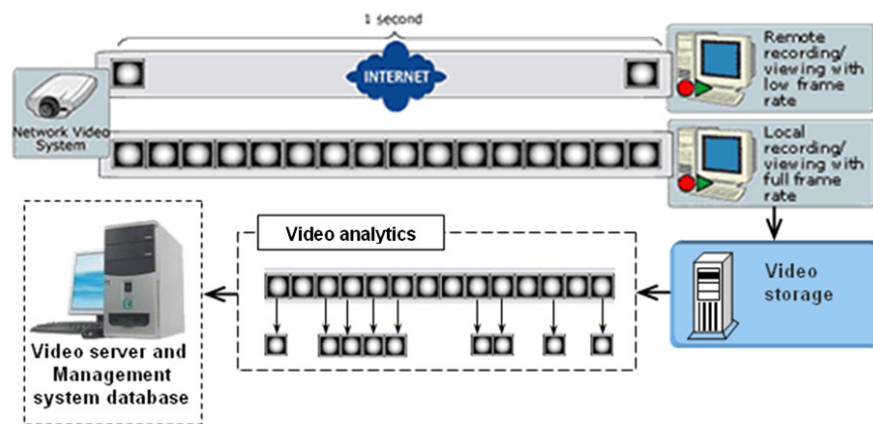


Figure 1. Process video analytics in CCTV

The video stream is characterized by a lot of information at relatively low cost of equipment. This explains the growing popularity of video cameras as a universal measuring instrument for the effective solution of many practical problems. Often shoot a moving vehicle (for example, the camera is mounted on a car, helicopter or other non-stationary platform). In such situations, image stabilization is needed. It is suitable for improving perception of the image operator and as a preliminary step for most computer vision algorithms, designed to work with the stationary frame.

Another important task for image processing, which is closely related to the task of stabilization is the task of tracking the selected objects. This problem is relevant for surveillance and monitoring, used to monitor the perimeter of police monitoring systems, etc.

Another application of computer vision algorithms - selection of moving targets. It is relevant to the aerospace and aviation industries to monitor and control the underlying surface.

3. COMPLEXITY

An important task is to develop fast algorithms for processing large amounts of video data and identification of a moving object in real time. Note that we will deal exclusively with digital video data, that is, those that are stored in computer memory and processed on its software or hardware.

At the present level of computer and video technology made possible the creation of systems that process video sequences in real time at a level of quality, which is appropriate to pay them much attention. In particular, the capacity of modern channels of the internet allows you to send them on video in real time.

Video data is a sequence of bitmap images in RGB, resulting from the digital video recording device (digital camera).

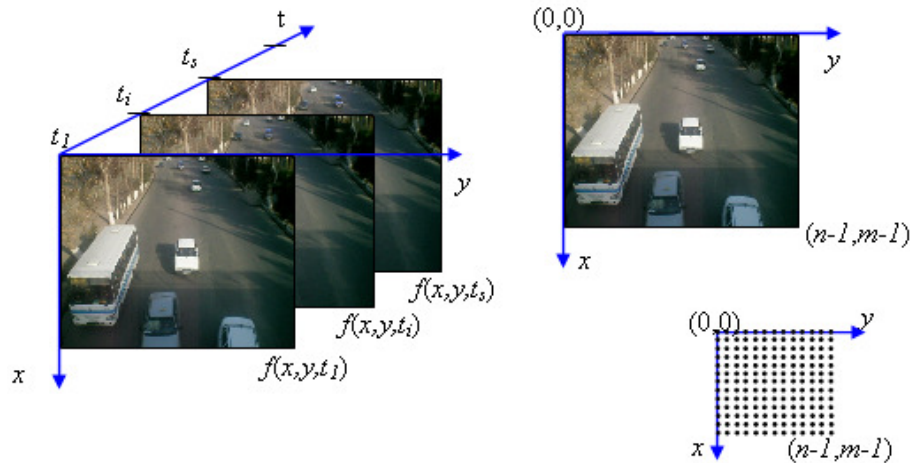


Figure 2. Model video image

Video data without sound can be represented as:

$$V = \{V(F, T) | F = f_v(x, y, t), x \in [0, N), y \in (0, M); N, M, v \in \mathbb{Z}, t \in T\}$$

where F is a frame, T is a time registration of the frame, $f_v(x, y, t)$ is bitmap image (frame) in F , v is the frequency of shoot camera ($C=15, 25, 30, 60, 120, \dots$), (x, y) are coordinates of the pixel raster, M is the horizontal number of pixels, N is the vertical number of pixels, t is the f frame time registration.

The volume of information available to the input of the algorithm for a second, in general, can be calculated by the formula

$$R = v \cdot N \cdot M \cdot d,$$

where, ν is the frequency of shoot camera (Hz), M is the width of the raster, N is the height of the raster, d is a color depth, or number of bytes per pixel, R is the total amount of information at the input of the algorithm per second (bytes).

Let's say, the camera operates at 30 Hz, image size: 640x480 pixel resolution, color depth 3 bytes per pixel. Total for the second algorithm has time to process 27648 KB of input data.

Let our processor running at 3 GHz, that is, does 3×10^9 cycles per second, so the formula

$$\omega = \Phi / \Omega,$$

where ω is the number of CPU cycles to process one byte of data per second, Φ is CPU frequency (cycles per second) to process one byte of input data per second will fall an average of 144 CPU cycles. It's not so much if you also take into account that most instructions of the CISC-processors which are practically all of the processors now manufactured by Intel and AMD, running away is not a single cycle, and for processing raster data is also necessary read/write operations in memory.

If, under our conditions to the input 27 648 KB of data per second, the algorithm with complexity of order $O(\omega)$ is needed 27.648 million CPU cycles to calculate, but the algorithm with complexity of order $O(\omega^2)$ is needed by about 7.64×10^{14} CPU cycles, what to several orders of magnitude increases the permissible value.

Thus, for video analysis solution for video surveillance is reasonable to consider only algorithms that have complexity less than $O(\omega^2)$.

4. CONSTRUCTION OBSERVATIONAL DISCRETE LINES

Usually the stage stationary traffic surveillance cameras filmed the video for car traffic flow directions. Reducing the amount of computation can be divided into two parts: the scenes flow intensive movements and area without movements (for example, fields, offices, etc.) (Fig.1-4).



Figure.1. Original and Frame with ODL



Figure 2. Original and Frame with ODL

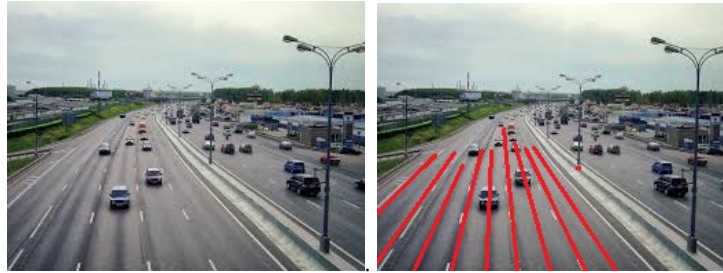


Figure 3. Original and Frame with ODL



Figure 4. Original and Frame with ODL

If the definition of a moving object by using the method of absolute inter frame is different than the processing pixels can be represented as Figure 5.

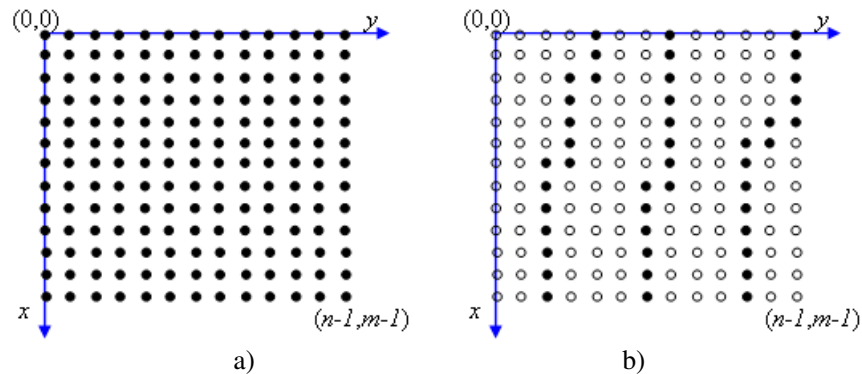


Figure 5. Processing and comparing pixels: a) the participation of all the pixels, and b) participation of pixels per line.

On stage video surveillance is allocated highway and to set the vertical direction (horizontal, curves, arcs) lines. To draw lines or curves you can use algorithms for rasterization of lines [3] and is formed by the coordinates of each ODL. The motion detector analyzes the pixels lie exactly these ODL. The ODL are set so that the stage of observation of the moving objects were on a last resort in a single line [5].

5. ALGORITHM ODL FOR MOTION DETECTION

The construction of the optical flow has traditionally been viewed as the evaluation procedure brightness-geometric changes between two adjacent frames. Motion of objects in front of a stationary camera lead to corresponding changes in the image [4]. The apparent motion of objects

relative to the camera, called optical flow. We define the field of motion, assigning each point of the image velocity vector. At some selected time point P_i the image corresponds to a point on the object surface P_0 . These two points are connected by the equations design. Point of the object relative to the camera P_0 moves at the speed of v_0 . This creates movement v_i corresponding image points P_i . During δt P_0 point moves away $v_0 \delta t$, and its image P_i - a distance $v_i \delta t$ (see. Fig.6)

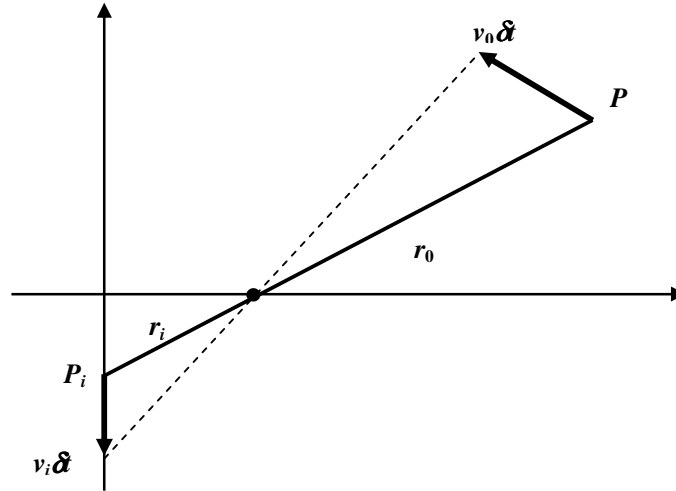


Figure 6. Moving the environment, causing the displacement of a point on the image.

Real-time distribution move together with observed objects. Optical flow, as mentioned earlier, called the apparent movement of the light pattern. Ideally optical flow corresponds to the previous field in the movement, however, in practice this is not always the case. In the settings of the camera and the software module on the stage set of Visual discrete lines. Now let $f(x,y,t)$ is the brightness of the pixel in the point images (x,y) at time t , L_1, L_2, \dots, L_k a plurality pixels respectively k ODL's at the $[P_i, Q_i]$ intervals and the variety does not change the number of pixels of each frame comparisons.

$$L_i = \{(x, y), x \in P_i, y \in Q_i\}, i = \overline{1, k}, \quad (1)$$

$$L_1 \cup L_2 \cup \dots \cup L_k \in f(x, y), \quad (2)$$

$$L_i \cap L_j = \emptyset, i \neq j. \quad (3)$$

Because the background must be separated from the individual pixels of the ODL at the scope level considering the mutual relationship between the pixels. Moving evenly under ODL always surrounded by pixel objects with pixels identical properties. Parallel algorithm of identification and tracking of moving objects through the ODL is presented below. We calculate the difference between two frames and compared against the threshold:

$$M_i(x, y, t) = \begin{cases} 1, & |I_{L_i}(x, y, t) - I_{L_i}(x, y, t - 1)| > \delta_m \\ 0, & \text{other} \end{cases} \quad (4)$$

The threshold value δ_m depends on the properties of the camera [3] and has a value in the interval [10,30].

The lower limit of fragments L_k seems many elements

$$H_j(L_k) = \{l_{ij} | l_{ij} \in L; i = k, j = \overline{1, m}\}. \quad (5)$$

Let denote the neighborhood $N_{\eta}(l_{ij})$ element $l_{ij} \in L$ following way

$$\begin{array}{|c|c|c|} \hline i-1,j-1 & i-1,j & i-1,j+1 \\ \hline i-1,j & i,j & i,j+1 \\ \hline i+1,j-1 & i+1,j & i+1,j+1 \\ \hline \end{array} N_8(l) = \{\vec{\eta}_1, \vec{\eta}_2, \dots, \vec{\eta}_8\}$$

Clockwise direction $\vec{\eta}_i$ vectors are designated:

$$\begin{aligned}
 \vec{\eta}_8 &= l_1(i-1, j-1), \quad \vec{\eta}_1 = s_1(i-1, j), \quad \vec{\eta}_2 = l_2(i-1, j+1), \\
 \vec{\eta}_7 &= l_7(i-1, j), \quad l(i,j) \quad \vec{\eta}_3 = l_3(i, j+1), \\
 \vec{\eta}_6 &= l_6(i+1, j-1), \quad \vec{\eta}_5 = l_5(i+1, j), \quad \vec{\eta}_4 = l_4(i+1, j+1).
 \end{aligned}$$

$$\begin{array}{|c|c|c|} \hline & i-1,j & \\ \hline i-1,j & i,j & i,j+1 \\ \hline & i+1,j & \\ \hline \end{array} N_D(l) = \{\vec{\eta}_{2i-1}\}, i = 1,2,3,4$$

$$\begin{array}{|c|c|c|} \hline i-1,j-1 & & i-1,j+1 \\ \hline & i,j & \\ \hline i+1,j-1 & & i+1,j+1 \\ \hline \end{array} N_I(l) = \{\vec{\eta}_{2i}\}, i = 1,2,3,4.$$

The N_D and N_8 is empty D and 8 neighborhood of the elements $l_{ij} \in G$. Consider the sequence of elements

$$\ell_1^k, \ell_2^k, \dots, \ell_r^k, \ell_r^k \neq 0. \tag{6}$$

If $N_8(\ell_i^k) \cap N_8(\ell_{i+1}^k) \neq 0$ and $P_i=4$ for ℓ_i^k and ℓ_{i+1}^k , then (6) sequences called discrete line. Tracking of all objects at the same time is to find the elements in a sequence (6) starting ℓ_1^k and ℓ_r^k . To find the start and end of the image pixel is scanned under the mask 3x3 and checks the value of P_k for elements of the $N_8(\ell_{r1})$. If $P_k=2$ for $\ell_{r1} \neq 0$, then ℓ_{r1} element specifies either the starting or ending pixel moving objects under the ODL. If does not exist pixel image $\ell_{ij} \neq 0$ on $N_D(\ell_i^k)$, then it means $\ell_i^k = \ell_r^k$ finite element (6) sequences, and follow-up is complete.

3. CONCLUSIONS

The software for this algorithm in C++, and tested on single-core computer chip on Intel (3 Hz). If necessary, track multiple objects, use the above algorithm independently several times for each object of interest. Experiments have shown that on a personal computer for tracking a single object algorithm can handle up to 100 frames per second, which is 4 times the flow of real-time video. At the core of the algorithm is the union of the three main existing approaches to date with constant quality control decision making and accuracy of the estimates at each step of the algorithm is allowed to create algorithms that ensure high robustness in a wide class of video sequences, real-time on modern non-specialized computing devices.

When detecting and tracking moving objects over ODL there different cases. For example, an object can be detected simultaneously two ODL, the direction of motion can be changed in the direction crossing between ODL. In such a case, consider the following studies.

ACKNOWLEDGEMENTS

The author would like to thank everyone, just everyone!

REFERENCES

- [1] V.Damjanovski. CCTV. Second Edition: Networking and Digital Technology. Butterworth-Heinemann; 2 edition (April 19, 2005).-584 pages. ISBN-10:0750678003.
- [2] A. Konushin: "Geometric properties of multiple images", Computer graphics and multimedia. Issue 4 (3) / 2006 - <http://cgm.computergraphics.ru/content/view/141>
- [3] S.V.Chirikov. Computer graphics algorithms (methods of rasterizing curves). Tutorial – St. Petersburg: St. Petersburg SIFMO (TU), 2001.- 121 pages.
- [4] A.A. Lukyanitsa, A.G. Shishkin. Digital processing of video images. AG – M.: ISS Press, 2009.
- [5] Sh.X.Fazilov, O.A.Mamaraufov, X.A. Xasanov. Fast algorithm to detection the moving objects in video surveillance. Information Technology Promotion in Asia (ITPA2011), September 26-27, 2011, Tashkent University of IT, Tashkent, UZBEKISTAN.

AUTHOR

ODIL ABDIXAMITOVICH MAMARAUF OV

Place of birth: UZBEKISTAN, Samarkand region, Bulungur district.

Location: 100125. UZBEKISTAN, Tashkent, the Mirzo Ulugbek district, Durman Yuli Steet, 51/36.

Date of Birth: March 16, 1975.

Scientific sphere: Digital imaging, Detection and identification of Dynamic objects in the Videostream, Image recognition.



Working skills

Borland Delphi, Visual Basic, Matlab, MathCAD — professional level.

C ++, Java — average level.

Windows, Unix— professional level

Education:

- Student of the Samarkand State University. Specialty: Applied Mathematics. (1992-1998);
- Post-graduate student, Institute of Cybernetics, Academy of Sciences of Uzbekistan, Tashkent. (1999-2000);
- Researcher, Institute of Cybernetics, Academy of Sciences of Uzbekistan, Tashkent. (2005-2006);
- Independent competitor. Senior lecturer of the department "General professional disciplines" of the Samarkand branch of the Tashkent University of Information Technologies. (2007-2013);
- Today Senior Researcher of the department "Software IT" of the Tashkent University of Information Technologies. (from 2014).

Executant projects:

- Executor of the project Development of methods, algorithms and computer image analysis system and pattern recognition. (2005-2006);
- Executor of the project Creation of educational and methodical complex on the subject of intelligent systems in the direction of" Computer Science and Information Technology "and" Vocational training (information technology). (2012-2013)
- The main executor of the project Development of algorithms and image processing software of dynamic objects in the video. (2015-2017)