The capture of moving object in video image

Weina Fu, *Changchun Institute of Technology, Changchun, China*

Zhiwen Xu, *Jilin University, Changchun, China*

Shuai Liu, *Changchun Institute of Technology and Jilin University, Changchun, China*

Xin Wang, *Changchun Institute of Technology and Jilin University, Changchun, China*

Article originally published in "Journal of Multimedia", V. 6 (December 2011), n. 6, pp. 518-525, http://ojs.academypublisher.com/index.php/jmm/article/view/jmm0606518525
Reprinted with permission

ABSTRACT. Nowadays, video is a primary information carrier in www (world wild web) and moving objects are often carrying more information. But it is hard to catch these objects in video quickly and correctly. In this paper, we put forward a method to catch moving object in video. Firstly, based on difference image method, we determine moving region in video image. To avoid hardness to build background, we build background with a new algorithm based on difference changed. Finally, we get the objects and denoise them with erosion and dilation. The experimental result is shown that the new method is feasibility and highquality.

KEYWORDS: *Applications, Modeling and Simulation, Sensor heterogeneity, Software Architecture, Structural Health Monitoring*

1. Introduction

An automatic video-based face recognition system includes human face detection part, face tracking part, facial feature capture part and the people face recognition part (Chellappa et al., 1995; Zhao et al., 2003; Li, Jain, 2005). Obviously, the premise is to locate the face. It is segment to two directions. One is how to locate human face in slack images (Liu et al., 2005), the other is how to locate human face in video (Srikantaswamy, Samuel, 2007). Moreover, it is well known that object recognition system has the samilar parts. An important method in video capture is to find and get changed regions of moving targets from background in series images of video. We can see the status of our work in video capture in algorithm a.

Algorithm a (Capture moving object in video)

Inp ut: a series video images
Output: the moving region (object).
Step1. Find difference between every two next images.
Step2. Judge if points are relative by relation of color and moving rules.
Step3. Catch the region (object) for next process.

We often call this method segmentation of moving objects. The effective segmentation of moving region is important in post-processing such as target classification, tracking and behaviour understanding. However, due to the dynamic changes of the background image, such as weather, light, shadow and other disturb factors, to make effective segmentation is still a difficult work.
In segmentation of moving objects, the main segmentation method is difference image method, time difference method and optical flow method (Sung, Poggio, 1998; Rowley et al., 1998).
Difference image method is a kind of technology to segment regions using difference of current frame and background frame. But it is too sensitive to dynamic scene that makes lots of mistakes. The main limitation of *time difference method* is that it can not get all pixels with general characters and it is often created a hole inside of moving task. Algorithm of *optical flow method* is too complex and too poor to resist noise. To compare these three methods comprehensively, the created new method is based on difference image method because it is simple and easy to implement in real time environment by the video image with general static background.
Nowadays, scholars improved object recognition to a new position. Tistarelli and partners (2009) provide an overview and some new insights on the use of dynamic visual information for face recognition. In their paper, not only physical features emerge in the face representation, but also behavioural features should be accounted. They give some experimental results obtained from real video image data to show the feasibility of the proposed approach.
Junius and partners (2010) demonstrate a three-dimensional (for location, time, and magnitude of body part movement) pattern representation of entire time-dependent front-view gait cycle that simultaneously displays the coupled kinetics of different body parts thereby revealing possible irregularities in the gait. Among the potential applications of their technique are improved diagnosis and treatment of gait pathologies in rehabilitation clinics and modeling

schools as well as development of more robust surveillance systems.
Tomokazu and partners (2010) propose an efficient method for estimating a depth map from long-baseline image sequences captured by a calibrated moving multi-camera system. The experiment verifies the validity and feasibility of their algorithm for both synthetic and real outdoor scenes.

In this paper, we segment moving object exactly by using the histogram with automatic threshold segmentation and mathematical morphology.

To consider the hardness to construct the background, we put forward a new algorithm. In this algorithm, we do not construct the background firstly, but we construct it in processing. Then, we trace moving object from background and catch them. Finally, we give some experiments to validate our algorithm's higher correctness and detect pace to the classic capture moving object algorithm.

2. Video image processing

A. Definition

Video image is also called dynamic image. It is made up of a series of images with a given or assumed relative order. We can get time interval series of every two next images by the order.

The relative order is the time interval of next images. Moreover, to define a time series t_i (i=1, 2, … , n) and for every t_k next to t_{k-1}, we set the time interval as:

$$\Delta t_k = t_k - t_{k-1}, k = 1,2,...,n-1$$
$$\Delta t_k (k = 1,2,...,n-1) = \Delta t$$

It is to say that all the time intervals of image capture are equal to each other.

We call frame every piece of video images. Of course, space-position of every moving object is different to each other by different times. In other words, when space-position of a moving object is changed from one frame to the next, we call it moving object. When a point P of space-object moves from (x_{k-1}, y_{k-1}) in frame k-1 to (x_k, y_k) in frame k, we set the displacement as $(\Delta x_k, \Delta y_k)$. We also call it parallax when position is shifted from t_{k-1} to t_k of a point at the object' surface.

B. Construct the background

In classic algorithm, the construction of background is a key step. Based on the relative simple background of moving tracking, we construct the background image by using method based on CDM (Change Detection Mask). The method assumed is that the moving object can not cover all images. In other words, the background is must appeared in images. So when the object moves, we will see that the background changes in image.

We set the luminance component of image series $Ii\ (x,y)$. Point (x,y) is the pixel's position and integer i is the frame number $(i=1,\dots,N)$. Integer N is the total number of frames.

Then we use the formula below to define change direction mask. It reflects gray's changes in the next frames.

$$CDM_i(x,y) = \begin{cases} d, d \geq T \\ 0, d < T \end{cases},$$
$$d = \left| I_{i+1}(x,y) - I_i(x,y) \right|$$

In this formula, the threshold value T is used to control the removal of noise. For each position (x,y), $CDM_i\ (x,y)$ explains the changed curve which is along time axis of the pixel at position (x,y). Then we can segment the curve by compute whether $CDM_i\ (x,y)$ is larger than zero. Some stillness parts detected is expressed as set $\{S_j\ (x,y), 1 \leq j \leq M\}$. We can see them in Figure 1.

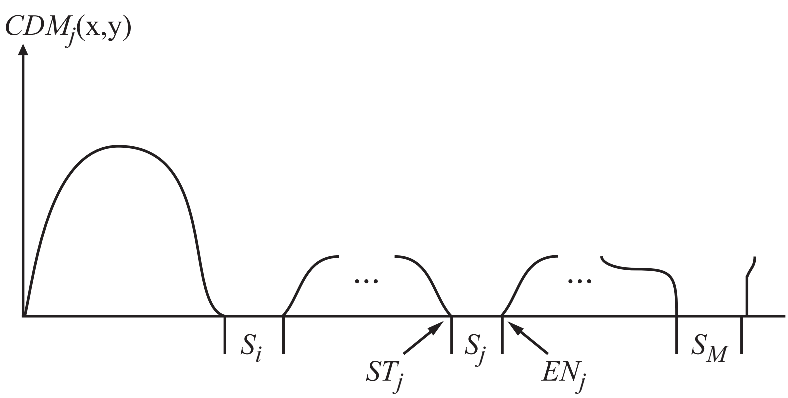


Figure.1. Curves to display the changes along time axis by difference of luminance frame

In Figure 1, the beginning and ending of S_j is ST_j and EN_j. We select the longest stillness part and register frame number of its midpoint as $M(x, y)$ in set of $\{S_j\}$ corresponding to position (x,y). Then we use points of frame with number $M(x, y)$ to fill corresponding position in video background. It is defined in the formula that follows.

$$M(x,y) = (ST(x,y) + EN(x,y))/2$$
$$B(x,y) = I(x,y,M(x,y))$$

In this formula, $ST(x, y)$ is the beginning of the longest stillness part and the ending is $EN(x, y)$. $B(x, y)$ is the rebuild video background.

C. A new think with no background structure

We can see that it is hard to construct background in video capture. Moreover, background structure is a key step in video capture. So if we find a new way to catch moving objects without background structure, we can get rid of bottleneck of capture of moving objects. In CDM, we find that it detects background by changes of images. As we know, it rebuilds background by using the region that has been covered by moving objects when moving objects move. So when moving objects cover all images and so on, we can not find the correct background. It means that we always catch incorrect objects in this case.

So we create a new think to find moving objects with no background rebuilt firstly. When objects move, we can detect them by the difference between two next series of images. Therefore, we know the movement region of images. It is to say the remained region is background when we remove the moving objects in images.

Then, we construct a background image. Of course, it may be a only small region of images. But we execute this step from the first image to last. When we get a piece of background, we change the original one to the union of the two. So we divide images to three kinds. The first is moving objects we have found. The second is background fragment. The last is moving object we have not found. We create a list to store them. We replace the older one to newer one when we find moving object we have found and link it to list when we find a new object.

Then we give a new algorithm to catch moving objects from video images.

3. Tracking and processing of moving object

A. Algorithm based on difference image method

Difference image method is a method that judges existence of moving objects by difference subtract from two next frames. We call the difference as difference image. It is simple and active to use difference image to process globally and crudely. Moreover, it is also beneficial when we catch crude information of moving objects. Principle of the difference image method is to find the moving object by difference. When there is no movement of objects in monitoring region, the difference of grey level between next frames in image series is very small. On the contrary, when there is some movement of objects, the difference of grey level between next frames will be significantly increased. So when we choose a reasonable threshold value, we can determine whether there exist objects in image series or not. The mathematical formula is:

$$D(x,y)=\begin{cases}1\ when\left|f_1(x,y)-f2(x,y)\right|>T\\0\ otherwise\end{cases}$$

The $f_1(x, y)$ and $f_2(x, y)$ in formula are the images of background only and background with a moving object inside. $D(x, y)$ is a binary difference image of image at point (x, y). T is the gray threshold, with its size determines the sensitive degree of monitor. The difference may be produced by the movement of objects in the region or moving objects entering inside or leaving the region. It can also be produced by the lighting changes of region or noise.
We segment moving objects based on difference image method, which is shown in algorithm b.

Algorithm b (Segment moving object)

Input: video image series with pretreatment.
Output: the moving region (object).

Step 1. Use current image as the background image to compare.
IF (video image series are not null)
Goto Step 2.
ELSE
Goto Step 4.

Step 2. Get the next image as reading image;
Find difference between the background and the reading image;
If (difference> threshold)
{
Find moving objects as the difference in reading image;
Goto Step 3.
}
Set the reading image as the current image;
Goto Step 1.

Step 3. Judge the object is same to the objects in list or not.
If (the object has the same characters (for example, just like color, gray, et al) to an object in list)
{
Mark the selected moving object in video images.
Replace the selected object in the object with same character list.
}
If (the object has some characters to static part in list which is an item in list to store static part that calculated by whole image minus moving object)
{
Find union of static part and selected object and replace static part with the union in list.
}
If (the object has no same characters to both moving object and static part in list)
{
Mark the selected moving object in video images.
Link the selected object list.
If (item number is larger than the item number threshold)
Release the list items with moving objects till to n percent released, n is a selected number between 1 and 100.
}
Goto Step 2.

Step 4. End algorithm.

Of course, algorithm b is more effective than classic difference image method. We show it in theorem 1 and 2. Theorem 1 is prove the correctness of algorithm b and theorem 2 is prove the effective of algorithm b.

Theorem 1. The moving objects are same to catch both in algorithm b and classic algorithm when capture is correct.

Proof

In order to prove theorem 1, we segment all conditions into four cases. They are case I-IV.

Case I. When there is no moving object in images. It is easy to know that the image series are same to each other in this case because there is no active object in it. So we can not catch any moving object by both classic algorithm and algorithm b. It is to say that the results of both two algorithms are the same.

Case II. When there exist moving objects in image series and the objects exist in all images of series. (The moving object moves slowly.)

The classic algorithm compares all images to the background image. In this case, because the moving object moves slowly, to assume the created background image is correct by classic algorithm, we can find the correct moving object by classic algorithm.

In algorithm b, we will catch the same series moving objects in this case when the moving objects move slowly. So we also catch the correct object.

Case III. When there exist moving objects in image series and the objects are not in all images of series because they move too fast.

In this case, our algorithm will find that there is no object in images when the object moves out. And so is the classic algorithm when background is constructed correctly. In other words, the classic algorithm will find that the image is same to background when background is constructed correctly and our algorithm will catch the object irrelatively with the background. This is because we use static part to find objects.

Case IV. When there are more than one moving objects in image series.

It is the similar condition with case III. When the background is found correctly, the classic algorithm will get correct object. Nevertheless, it is hard to construct correct background when there is more than one object in background. Therefore, the capture rate is lower with more moving objects. To consider all above, we find that our algorithm always catches the correct moving objects. But the capture rate of classic algorithm depends on the correctness rate of background. So theorem 1 is proved.

Theorem 2. The magnitude of time complexity of algorithm b and classic algorithm are same to each other. The calculate time of algorithm b is less than classic algorithm plus $c \cdot s \cdot m$. *Number* n *is item number threshold of list and* m *is whole image number. Number* c *is a constant number.*

Proof

When we ignore the time of the structure background, we can easily find that the classic algorithm compute $s \cdot m$ times by s as whole pixels of every images. Compute time of Algorithm b can divide the time of step 2 and step 3.

Obviously, the compute time of step 2 is $s \cdot m$ because it compare s times for each image. And in step 3, we calculate every ‘if’ separately.

The mark of first ‘if’ is the same of the classic algorithm. It means that the calculation is also in classic algorithm. We can ignore it. The replace part take less than n calculate times.

It takes less than s calculate times by determining the second ‘if’. The union spends less than s calculates times because there are moving objects in it.

The last ‘if’ finds a new moving object in image. The mark part is ignored for the same reason of the first ‘if’. The link part spends one time and the item numbers detection is taken one time if we set an item to record the item number in list.

So when calculating the total times, we know that it spend less than $n+2s+2$ compute times for each image. It is to say that it spends $(n+2s+2) \cdot m$ times in step 3.

To plus compute times of step 2 and step 3, we conclude that algorithm b spend less than $(n+2s+2) \cdot m$ times in classic algorithm. As we know, n is usually small. So theorem 2 is proved.

Moreover, we ignore calculate time of background structure of classic algorithm. In fact, the background structure is an important process in classic algorithm and it is spent a lot of time to construct it. So when we consider this, we know that it is valuable that algorithm b spends some time to avoid constructing the classic background.

B. Capture of moving object

Using the difference between background and current frame image, we can gain moving objects to segment. We must segment the remaining points and moving objects because there may exist some remaining points of background in the segmented image. Of course, we can choose the suitable threshold based on distribution of images' gray. In this paper, we use an automatic threshold segment method to segment images because of histogram characters of these images.

We assume that gray scales of all images are $1, 2,..., L$ and the corresponding pixel number with gray i is n_i. So whole pixel number $N = n_1 + n_2 + ... + n_L$. It is to say that probability distribution of pixels with gray i is $P = n_i / N$.

In the formula, we set $P_i \geq 0$ and $\sum_{i=1}^{L} P_i = 1$.

When we divide the whole image to two kinds C_0 and C_1 with threshold gray k, in other words, we call a pixel is in C_0 when a pixel's gray is in $[1... k]$ and a pixel is in C_1 when a pixel's gray is in $[k + 1...L]$, we find that the probability value of the two kinds is $w_0=P_k (C_0)=\sum_{i=1}^{k} p_1 = 1$ and $w_1=P_k (C_0)=\sum_{i=k+1}^{L} p_1 = 1 - w_0$.

Then, we solve the average value of the two kinds is

$u_0=\sum_{i=1}^{k} iP_i /w_0=u(k)/w_0$ and $u_1=\sum_{i=k+1}^{L} iP_i /w_1=(u_1-u(k))/(1-w(k))$.

In the formulas, $w(k)\sum_{i=1}^{k}=P_1$ and $u(k)=\sum_{i=1}^{K} iP_i$.

Then, we can find that the average value of the whole image is $u_y=u(L)=\sum_{i=1}^{L} iP$, and it is easy to know that $w_0u_0+w_1u_1=u$ and $w_0w_1=1$. To study variances of these two kinds, we find that the variances are $\sigma_0^2 = \sum_{i=1}^{k} (i-u_0)^2 P_i/w_0$ and $\sigma_1^2 = \sum_{i=k+1}^{L} (i-u_1)^2 P_i / w_i$

To define variance between-kinds $\sigma_B^2=w_0(u_0-u_y)^2+w_1(u_1-u_y)^2=w_0w_1(u_0-u_1)^2$. Moreover, we define variance within-kind $\sigma_w^2 = w_0\sigma_0^2 + w_1\sigma_1^2$ and total-variance $\sigma_r^2 = \sum_{l=1}^{L} (l-u_y)^2 P_l$.

Then we can solve the best threshold $k^*=max\,\sigma_B^2, (1 \leq k \leq L)$ by the maximum of variance between-kinds. After that, we confirm moving region of initial video images.

C. Denoise in moving region

As we know, there would be errors in the caught object because the movement and environment are usually unpredictable. For example, intense movement can bring sharp or fuzzy border and strong light reflection and elastic deformation can bring bad-capture. In other words, when we get the moving object from video image, we will find the moving object always caught with sharp or fuzzy border, salt or pepper region.

So in order to catch clear and complete moving object, we need to modify its body further more. In our work, we use mathematical morphology to modify the object. Specifically, we use erosion to fill small holes and dilation to remove isolated noise. To use them in special order, we get opening and closing operation derived from erosion and dilation.

When we call B^z as translation of structure of element B, we know that dilation $X \oplus B^z$ is a set construct with all points z, which makes B^z and X not empty set. Furthermore, dilation is an expanded process. The exchanged result is to make the target object expanded and empty holes reduced. So we can use dilation to fill holes in moving object and return it to initial connected domain.

Oppositely, erosion $X\Theta B^z$ is such a set construct with all points z, which makes B^z is a subset of X. Erosion is a shrink transformation because its result is a subset of X. Erosion is a process to remove boundary points. The exchange result can expand hole and shrink object. So it is an active method to remove isolated noise point.

Generally, both erosion and dilation are irreversible operations. It is to say that the result is usually not X when we transform X by erosion firstly and dilation secondly. We set the result X_B, so we know that $X_B=(X\Theta B^2)\oplus B=\cup\{B_z{:}B_z \subset X\}$. The new morphological transformation is called opening operation. We know that X_B is constructed with and set of B's translation B^z in X. So opening operation always smooth the object's border. Moreover, it can remove small sharp tips and isolation points. It also sharpen angles, disconnect limited gaps and remove thin tips.

Otherwise, closing operation $X_B=(X\oplus B^z)\Theta B=\cup\{B_z^c{:}B_z \subset X^c\}$ is opposed to opening. In other words, we transform X by dilation firstly and erosion secondly. In this case, X_B is an intersection of complementary set of all translation B^z of B outside X. Closing operation is smooth border same to opening. Similarly, it removes limited gaps and long thin blanks. It also removes small holes and fill ruptures of border.

To execute several erosions and dilations to caught object continually, we can get a complete moving object from video images. The capture object can be used in the further process.

4. Experiment

In this paper, we give an algorithm to catch moving object in series images of video. This is not only a part of algorithm of object recognition and judgment, but also an independent algorithm. So when we validate activity and correctness of our algorithm, we do not have to execute the experiment in a whole video capture algorithm. We can use difference method algorithms to catch moving object. Then we get important parameters of both difference method algorithm and our algorithm which conclude the caught object and computed time. By comparing the two algorithms with several different series images, it is known that our algorithm is better.

A. Capture object in video image

We can see the comparison of the two algorithms by catching normal moving object in figures below with Figure 2 that contains the original video images. We can also see the parameters in Table 1. In Figure 2, we can see some images of a traffic accident video from website www.youku.com. The images are not continuous considering the visualization factor. We choose some images with same interval from a part of video.

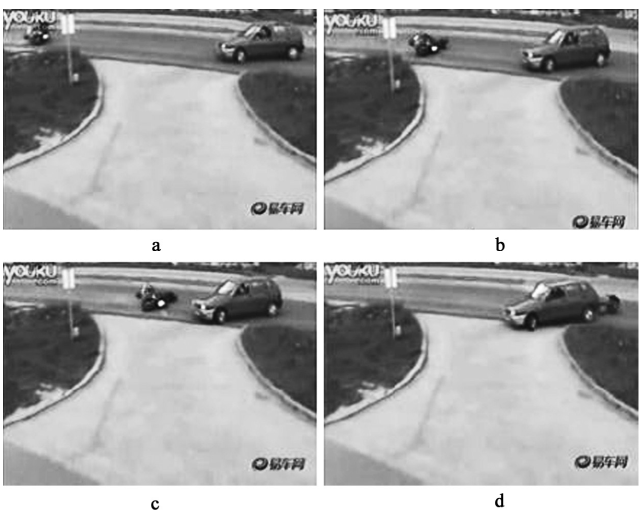


Figure 2. Original video images, the images order are from a to d

In Figure 3, we can see two moving objects. One is a motorbike, the other is a car. We use both classic algorithm and algorithm b to catch moving objects in this figure.

In fact, this video has whole background itself. We can see it in Figure 3. So classic algorithm has not to find background and get well detection result. The result is in Figure 4.

Figure 3. Background of the video images

Figure 4.x.1~4.x.4 are processes of figure 2.x (x is a~d). In these figures, figures 4.x.1 and 4.x.3 are processes of algorithm b, figure 4.x.2 and 4.x.4 are processes of classic algorithm. Both figures 4.x.1 and 4.x.2 are capture of moving objects. Figures 4.x.3 and 4.x.4 are extraction of moving objects.

We can find that both classic algorithm and algorithm b catch the correct moving objects in Figure 4. Otherwise, we find that both capture time and capture quality are better in figure 4.x.2 and 4.x.4. This means that classic algorithm is better than algorithm b. The advantage of algorithm b is that it spends less time than classic algorithm when we create the moving orbit of objects.

It is because that classic algorithm uses CDM to detect moving objects. All images are compared to the background. As we know, the background is easy to construct in this video. In fact, the first image of image series is the whole background. So the classic algorithm benefits of it in this case.

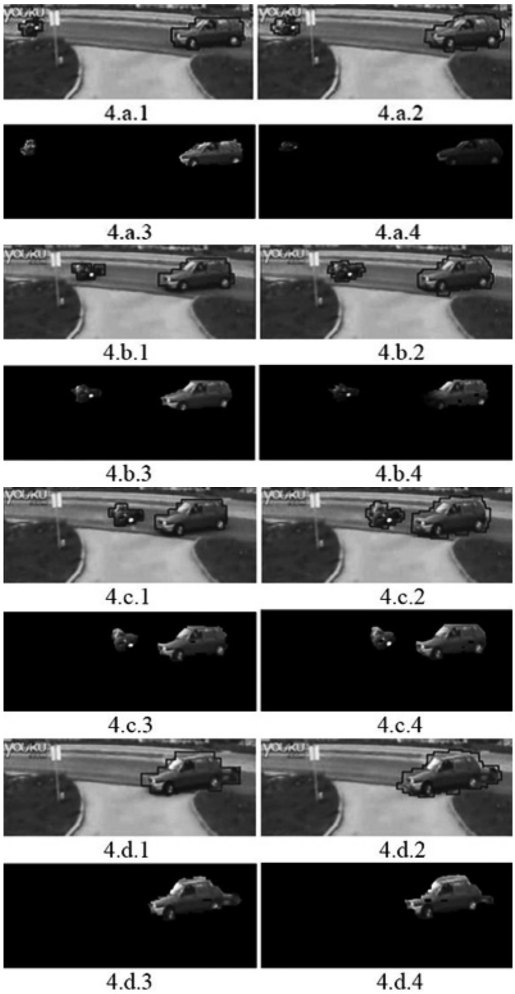


Figure 4. Capture and extraction of moving object

When we create the orbit of moving objects, algorithm b benefits. It is because moving objects in all images are compared in algorithm b. It is to say that we can easily find the moving orbit of objects. We show the parameters just like capture time and pixels number of moving object in Table 1. We can see the effective of the two algorithms when they catch moving objects.

Table 1. Parameters of classic algorithm and algorithm b

	Classic Algorithm		Algorithm b
Pixels	400/300=120000		
Pixels of the caught moving car with background	image a	10864	10913
	image b	10871	10912
	image c	10886	10927
	image d	10890	10938
Pixels of the caught moving car with background	image a	2706	2724
	image b	2731	2775
	image c	2749	2790
	image d	2282	2197
Time of the caught moving object with background	image a	1.28ms	1.31ms
	image b	1.29ms	1.31ms
	image c	0.54ms	1.30ms
	image d	0.55ms	1.31ms
Time of the caught moving object with background	image a	2.26ms	1.31ms
	image b	2.38ms	1.30ms
	image c	1.45ms	1.31ms
	image d	1.44ms	1.31ms

When we remove the background image from video, we can get the different results. It is also in Table 1. We do not show capture figures because it is similar to Figure 4.

Now we study parameters of this video capture in Table 1. We can see that there are some differences between classic algorithm and algorithm b.

At first, we can see that the pixels of the two moving objects are similar in the two algorithms. The pixels of four images are all increased from a to c. In Figure d, we see that parts of the two objects are superposed. Moreover, the motorbike is behind the car in figure. It means that a part of the motorbike can not be seen. This is why the pixels of motorbike decrease a lot in figure d. At the same time, we can see that the capture pixels of algorithm b are more than classic algorithm. It means that the classic algorithm is a little better than algorithm b in this condition.

Secondly, we find that the time of the caught moving object with background is also better than in our algorithm. Especially, with the video going, the time of classic algorithm is decreasing. It is because the background has been rebuilt. In algorithm b, we can see that all time in image a~d are similar. It is because we compare images to images in list and do not rebuild background. So algorithm b is a steady algorithm whether there is background or not. The classic algorithm is not a steady algorithm because the time with background is very different to the time without background. When we see the time of capture without background, we can see that the time of rebuild is increased. So is the compared time.

At last, we compare the time with and the time without background. We know that classic algorithm depends on the background much more because the capture time increases a lot in classic algorithm. When the background is removed, the rebuilt time is increased much and larger than algorithm b. So when we catch moving objects in a video with tiny background, we find that classic algorithm is unsuitable. Therefore, we execute an experiment to catch moving objects in a video with tiny background. The result of experiment is shown in Figure 5-6.

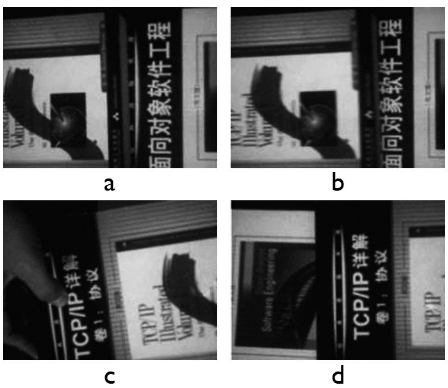


Figure 5. Original video images, the images order are from a to d

In this video, we use two books to execute our experiment. Book a is *TCP/IP Illustrated. Volume 1. The Protocols* written by W. Richard Stevens which is published by China Machine Press and Addison Wesley. Book b is *Object-Oriented Software Engineering* written by Stephen R. Schach which is published by China Machine Press and McGraw Hill.

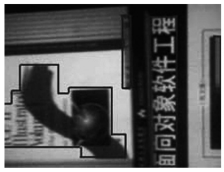
We video-record the video by used book a cross to book b. In this process, the video shows only a little background.

Figure 5 shows some images from the video. We choose some images with same interval just like in Figure 2. We put book a onto book b and move book a across book b.

Figure 6. Capture and extraction of moving objects

6.a.1

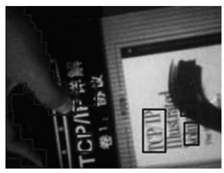

6.a.2



6.b.1


6.b.2

6.c.1


6.c.2

6.d.1


6.d.2

Then we can see the capture objects by both classic algorithm and algorithm b in Figure 6. Figures 6.*.1 are captures by algorithm b and 6.*.2 are captures by classic algorithm. In Figure 6, the advantage of algorithm b is obvious because classic algorithm can not determine the background. When it determines the incorrect background, the following captures of moving objects will be all jumbled in the upcoming images. It is all in Figures 6.*.2.

In Figures 6.*.1, we know that the effect of background is immune in algorithm b. Though it may catch some incorrect objects in some of the beginning images, it may change itself in the following images. So when the background is tiny, algorithm b is more effective than the classic algorithm.

5. Conclusion and further work

We give a method to catch a moving object in series images of video in this paper. In this method, we give an algorithm to catch moving objects. This algorithm avoids background structure. At first, we determine the moving region by difference between next images in series. Secondly, we track the moving objects and catch them. Finally, we catch the moving object to de-noise it. The experiment shows that our method is more effective than classic algorithm by similar compute time.
In next step, we will work in two directions. The first is that we will extend the capture method to distribute system in order to improve the capture time. Secondly we will use this method in some detection system to catch the selected object.

Acknowledgment

The authors wish to thank the anonymous reviewers for their helpful comments in reviewing this paper. This work was supported by National Natural Science Foundation of China (No. 60973091).

References

Chellappa Ramalingam, Wilson Charles, Sirohey Saad (1995), *Human and machine recognition of faces: A survey*. Proceedings of the IEEE, 1995, V. 83, n. 5, pp. 705-740

Junius André F. Balista, Maricor N. Soriano, Caesar A. Saloma (2010), *Compact time-independent pattern representation of entire human gait cycle for tracking of gait irregularities*, "Pattern Recognition Letters", V. 31, n. 1, pp. 20-27

Li Stan Z., Jain Anil K. (2005), *Handbook of Face Recognition*, New York, Springer

Liu Xiaomin M., Zhang Yujin J., Than Huachun C. (2005), *A new Hausdorff distance based approach for face localization*, "Sciencepaper Online", pp. 1-9

Rowley Henry A., Baluja Shumeet, Kanade Takeo (1998), *Neural network-based face detection*, "IEEE Transactions On Pattern Analysis and Machine Intelligence", V. 20, n. 1, pp. 23-38

Srikantaswamy R., Sudhaker Samuel R.D. (2007), *A novel face segmentation algorithm from a cideo sequence for realtime face recognition*, "EURASIP Journal on Advances in Signal Processing", V. 2007, pp. 1-6

Sung Kah-Kay, Poggio Tomaso (1998), *Example-Based Learning for View_Based Human Face Detection*, "IEEE Transactions On Pattern Analysis and Machine Intelligence", V. 20, n. 1, pp. 39-51

Tistarelli Massimo, Bicego Manuele, Grosso Enrico (2009), *Dynamic face recognition: From human to machine vision*, "Image and Vision Computing", V. 27, n. 3, pp. 222-232

Tomokazu Sato, Naokazu Yokoya. (2010), *Efficient hundreds-baseline stereo by counting interest points for moving omni-directional multi-camera system*, "Journal of Visual Communication and Image Representation", V.21, n. 5-6, pp. 416-426

Zhao Wenyi, Chellappa Ramalingan, Rosenfeld Azriel, Phillips P. Jonathon (2003), *Face recognition: A literature survey*, "ACM Computation Survey", V. 35, n. 4, pp. 399-458

Sintesi

Il riconoscimento automatico di oggetti, da immagini fisse o in movimento, è da sempre legato a questioni di sicurezza, siano esse relative ai sistemi militari di difesa, o a quelli di sorveglianza. Con il perfezionamento e la diffusione delle tecnologie, tale settore sta, tuttavia, avendo ricadute anche in applicazioni meno strategiche: ne è un esempio il face recognition *che, sviluppato per l'individuazione dei visi di soggetti potenzialmente pericolosi a partire dai filmati dei sistemi di controllo degli aeroporti, così come di altre aree ad alto rischio, è attualmente ampiamente utilizzato anche*

nei social network.

Questa utilità in ambiti così diversi è di sprone sia per il perfezionamento delle tecnologie attualmente presenti, sia nella ricerca e nella sperimentazione di nuove idee.

La collaborazione tra l'Università cinese di Jilin e l'Institute of Technology di Changchun si inquadra nel secondo ambito, quindi nella indagine di nuove vie per la determinazione di oggetti in movimento, senza che queste comportino l'introduzione di nuove tecnologie. Nel caso particolare, la novità è costituita da un nuovo algoritmo.

In generale, gli algoritmi di localizzazione sono divisi in due passi: nel primo, si evidenziano gli oggetti che si muovono rispetto allo sfondo; nel secondo, si identificano gli oggetti evidenziati. Fondamentale, in questo schema, è che lo sfondo *sia conosciuto. Esistono due possibilità: è noto* a priori *ovvero è costruito a partire dalle immagini, dando vita così a un passo zero. Il primo caso è attuabile in una casistica che, pur essendo molto ampia, non copre il complesso dei sistemi. Nel secondo invece i costi macchina totali risultano elevati, portando così a una bassa efficienza.*

Il presente algoritmo risolve questo problema evitando entrambe le possibilità, introducendo, di fatto, un'idea innovativa: lo sfondo, infatti, non è dato a priori né è costruito inizialmente, ma viene individuato contestualmente *agli oggetti in movimento. Mentre questi sono determinati dall'analisi di fotogrammi successivi, tenendo conto di diverse caratteristiche che variano tra le posizioni di aree che si modificano in modo coerente e solidale, così come dei possibili legami tra queste, suggeriti dai colori delle stesse, ciò che non subisce modifiche viene identificato come sfondo. Inoltre, poiché l'oggetto in movimento, con lo scorrere dei fotogrammi, copre e scopre parti diverse, lo sfondo viene a formarsi come un mosaico e non istantaneamente. Questa strategia, questo utilizzo contemporaneo degli stessi fotogrammi per raggiungere due obiettivi differenti, da una parte svincola dalla necessità della conoscenza a priori, dall'altra aumenta l'efficienza rispetto agli algoritmi analoghi precedentemente progettati.*

I futuri sviluppi già pianificati per questo algoritmo che, già nei suoi primi passi, porta in nuce un ampio spettro di applicazioni, dalla diagnosi e il trattamento di patologie all'interno di cliniche riabilitative, allo sviluppo di sistemi di sorveglianza maggiormente robusti, fino ad arrivare all'individuazione di uno stesso oggetto su sorgenti video differenti, non potranno che migliorare l'efficacia e la diffusione dell'algoritmo.