A MEASURING SYSTEM FOR TRAFFIC FLOW OF PASSERS-BY BY PROCESSING ITV IMAGE IN REAL TIME

Hideo TAMAMOTO+, Yuichi NARITA+, Akira YANASE++, Futoshi SAITO+++ and Kazuto KOMATSU++++

+ Department of Information Engineering, Akita University, Akita City, Akita 010, Japan
++ Odate Technical High School, Odate City, Akita 017-01, Japan
+++ Goyo Electronics Co., Ltd., Minami Akita Gun, Akita 010-02, Japan,
++++ Ricoh System Development Co., Ltd., Akita City, Akita 010, Japan

Abstract

We have discussed the algorithm that such primitive traffic parameters as the number of passers-by, their moving direction and moving speed can be concurrently measured by processing ITV image in real time with a personal computer.

In this algorithm, the movement of a passer-by is displayed in the form of a locus line on a cross section image. Therefore, by counting the number of locus lines and calculating the gradient of the locus lines, we can measure the number of passers-by together with the moving direction and moving speed.

We implemented this algorithm on the system consisting of an ITV camera, an image processor and a personal computer. Using this system in a real situation, we could measure the above mentioned parameters within 5% and 20% of measuring error in case of a light traffic flow and a heavy traffic flow, respectively.

1. Introduction

It is one of the most important subjects in a heavily populated city to know the actual situation in the traffic flow of passers-by. To meet this subject, it is required to develop the automatic measuring system that can measure in real time such primitive traffic parameters as the number of passers-by, their moving direction and moving speed by using a computer. Some researchers have proposed the method where the traffic flow of passers-by is measured by means of image processing [1]-[4].

In these methods, an ITV camera is set such that its focusing direction is perpendicular to or slantingly above the road where passers-by are walking. Therefore, the quality of the taken image is much influenced by the shade, umbrellas and so on depending on the weather. The location where the camera can be set is restricted. Since in some proposed methods the traffic flow can be measured in real time using a mini-computer or a special-purpose processor, the system seems to be expensive. Also the above mentioned primitive parameters cannot be measured concurrently.

We discussed the algorithm that the number of passers-by, their moving direction and moving speed can be concurrently measured by processing ITV image in real time with a personal computer.

In this algorithm, the scene where passers-by are walking is taken by an ITV camera whose focusing direction is set horizontally to the road. One horizontal scanning line is appropriately selected on the scene. The density values of the pixels on this scanning line are gathered at some time interval. Arranging these data vertically in order of time, we can get the two-dimensional image. In this image, movement of passers-by can be observed in the form of a locus line. Therefore, by counting the number of locus lines and calculating their gradient, we can measure the number of passers-by together with the moving direction. Moreover, by measuring the period during which a passer-by appears and then disappears in the scene, we can calculate the moving speed of each passer-by

We implemented this algorithm on the system consisting of an ITV camera, an image processor and a personal computer. Using this system, we could measure in real time the number of passers-by, moving direction and moving speed within a acceptable error range in case of a light traffic flow.

2. Measuring system

2.1 System configuration

The configuration of the measuring system is shown in Fig.1. This system consists of an ITV camera, a video tape recorder, an image processor, a monitor display and a personal computer(PC). While developing the system, we recorded the image using a video tape recorder and then processed the replayed image. But after the development of the system will have been completed, the image taken by an ITV camera will be directly processed.

Table 1 shows the specification of the system. The image processor can get one frame of an image into an image memory every 1/60 seconds. The resolution of

Table 1. List of equipment

ITV Camera	SONY:BMC-100
Video Tape Recorder	SONY:SL-F7
Video Monitor	HITACHI:VM-129A
Image Processor	KEIO:IFM/PC-Jr
Personal Computer	NEC:PC9801RA

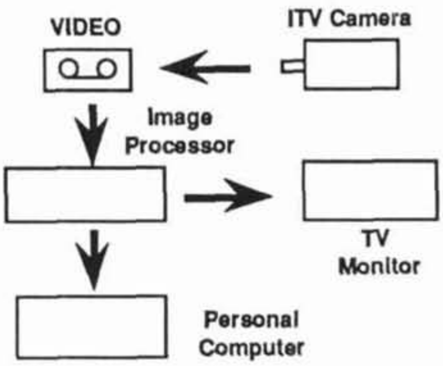

Fig.1 Measuring system

the image is 256x256, and the number of gray scale levels is 16(4 bits/pixel). Since the image memory is connected to the CPU bus of the PC, it can be directly accessed via the bus from the PC. All programs have been coded using C-language.

2.2 Outline of processing

In this system, the scene where passers-by are walking is taken by an ITV camera whose focusing direction is set horizontally. One horizontal scanning line is appropriately selected on the scene. This line is called a sampled horizontal line. The density values of the pixels on the sampling line are gathered at some time interval. Arranging these data vertically in order of time, we can get a two-dimensional image. This image is called a cross section image. In the cross section image, movement of passer-by can be observed in the form of a locus line as shown in Fig.2. Therefore, by counting the number of locus lines and calculating the gradient of the locus lines, we can measure the number of passers-by together with their moving direction.

Moreover, by calculating the period during which a passer-by walks between two points whose distance is known, we can measure their moving speed.

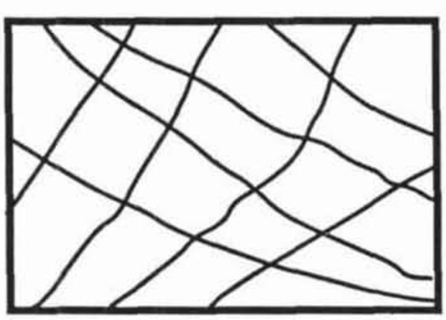

Fig.2 Locus line in cross section image

3. Extraction of locus lines

As described in **2.2**, we can measure the traffic flow of passers-by by counting the number of locus lines and calculating the gradient of them. But, in order to simplify the processing, we discussed some pre-processings explained below.

3.1 Pre-processing

3.1.1 Density value of background

Let the density value of the pixel on the i^{th} column of the sampled horizontal line that is sampled in the j^{th} sampling time be designated $s^{(j)}[i]$. Also let the same density value of the pixel in case a passer-by does not exist be designated $b^{(j)}[i]$. We call this density value a density value of the background. Since the density value of the background varies with time, its value is updated by averaging the past n density values of the corresponding pixels as shown in Eq.(1). But since there is scarcely a case where the density value of the background changes rapidly, if the difference between $b^{(j-1)}[i]$ and $s^{(j)}[i]$ is greater than Δb, we consider that this $s^{(j)}[i]$ is corresponded to a pixel of a passer-by or a noise. Then we substitute $b^{(j-1)}[i]$ for $s^{(j)}[i]$ in Eq.(1) and calculate $b^{(j)}[i]$.

$$b^{(j)}[i] = \sum_{k=0}^{n-1} s^{(j-k)}[i]/n \qquad (1)$$

3.1.2 Binarization

The sampled data are binarized such that if the difference between $s^{(j)}[i]$ and $b^{(j)}[i]$ exceeds Δ_t, the value of $s^{(j)}[i]$ is transformed into 1, otherwise 0. That is, letting the binarized value of the pixel on the i^{th} column be designated $x^{(j)}[i]$,

$$x^{(j)}[i]=\begin{cases} 0\ ;\ |s^{(j)}[i]-b^{(j)}[i]| < \Delta_t \\ 1\ ;\ |s^{(j)}[i]-b^{(j)}[i]| >= \Delta_t . \end{cases} \qquad (2)$$

3.1.3 Noise elimination

The data of one sampled horizontal line are only sufficient in principle to measure the traffic flow, but we cannot accurately measure it due to some noises. To eliminate the influence of the noises, each two horizontal lines above and below a sampled horizontal line respectively are referred to, and also the width of a passer-by is considered. We call each of these 4 lines a reference horizontal line.

[Horizontal direction]

Observing the binarized density values on the sampled horizontal line, we can find out the runs of the value 1 if a passer-by exists. Therefore, if the number of runs of the value 1 is greater than or equal to some N(N is equivalent to the width of a passer-by), the $x^{(j)}[i]$ is transformed to 1, otherwise to 0 as shown in Eq.(3). Here, let the modified values of $x^{(j)}[i]$ be designated $y^{(j)}[i]$, and call them horizontal noise eliminated data.

$$\cdot\ y^{(j)}[i]= 1\ ; \qquad (3\text{-}1)$$

$$\begin{cases} \prod_{k=-(N-1)/2}^{k=(N-1)/2} x^{(j)}[i+k]=N, \text{ or } \quad \prod_{k=-(N-1)}^{k=0} x^{(j)}[i+k]=N, \text{ or} \\ \prod_{k=0}^{k=(N-1)} x^{(j)}[i+k]=N \end{cases}$$

$$\cdot\ y^{(j)}[i]=0\ ; \text{ otherwise.} \qquad (3\text{-}2)$$

[Vertical direction]

With respect to the reference line, the binarization is performed using the same method shown in **3.1.2**, and then some noises are eliminated. Let the binarized density value of the pixel on the i^{th} column of the k^{th} reference line that is sampled at j^{th} sampling time and eliminated horizontal noises be designated $r_k^{(j)}[i]$ (k=1,2,3,4). In order to eliminate the vertical noises, we modify the $y^{(j)}[i]$ by taking the majority of the density values of the five pixels on the same columns of the sampled horizontal line and four reference lines as shown in Eq. (4). These modified data are called noise eliminated data.

$$z^{(j)}[i]=\begin{cases} 0 : \sum_{k=1}^{4} r_k^{(j)}[i] + y^{(j)}[i] < 3 \\ 1 : \sum_{k=1}^{4} r_k^{(j)}[i] + y^{(j)}[i] >= 3 \end{cases} \quad (4)$$

An example of noise elimination is shown in Fig.3 in case N=3.

0	0	0	1	0	1	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0
0	0	0	1	0	0	1	1	1	1	0	0
0	1	0	1	0	1	1	1	1	1	0	0
0	0	0	0	0	0	1	1	1	0	0	0

(a) Binarized density values

0	0	0	0	0	1	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	1	1	1	0	0	0

(b) Elimination of horizontal noises

0	0	0	0	0	0	1	1	1	1	0	0

(c) Elimination of vertical noises

Fig.3 An example of noise elimination

3.2 Extraction of moving direction

We consider two dimensional array p[j,i] where the value of [j,i]-element is equal to $z^{(j)}[i]$. Associating the value of [j,i]-element with the density value of the pixel on the j^{th} row of the i^{th} column and displaying p[j,i] on the monitor, we can get the locus line that represents the movement of a passer-by in the displayed image. Here, we call the p[j,i] a passer-by data.

Next, considering the two-dimensional array d[j,i], where the value of [j,i]-element is calculated as shown in Eq.(5).

$$d[j,i]=p[j-1,i] \times 5 + p[j,i] \times 3 \quad (5)$$

The value of each d[j,i] is either of 0,3,5 or 8. We substitute 1 and 2 for 5 and 8, respectively. This substitution does not have an essential meaning, but it is easy for us to recognize visually the moving direction of a passer-by when these data are arranged two-dimensionally. We call the d[j,i] a direction data. Fig.4 shows an example of the d[j,i] derived from some p[j,i]. As shown in Fig. 4, looking at the values of d[j,1]....d[j,n] sequentially, when a passer-by are moving from left to right and from right to left, we can find out the sequences

00...011..122..233..100..0 (6)

and

00...033..322..211..100..0 (7)

respectively. When a passer-by are standing, we can find out the sequence

00...022...200..0 (8)

When two passers-by are crossing each other, we can find out the sequence

00...011..122..233..322..211..100..100..0 (9)

or

00...033..322..211..122..233..300..100..0 (10)

Observing the direction data, we can recognize in which direction the passer-by is moving who is associated with each direction data.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(a) Passer-by data(p[j,i]).

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
1	2	2	2	3	0	0	0	0	0	0	0	3	3	3	2
1	1	1	2	3	3	3	0	0	3	3	3	2	1	1	1
0	0	0	1	1	1	2	3	3	2	1	1	1	0	0	0
0	0	0	3	3	3	2	1	1	2	3	3	3	0	0	0
3	3	3	2	1	1	1	0	0	1	1	1	2	3	3	3
2	1	1	1	0	0	0	0	0	0	0	0	1	1	1	2
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(b) Direction data(d[j,i]).

Fig. 4 An example of direction data

3.3 Extraction of locus line

In Eq.(6), all the numbers of 1, 2 and 3 are transformed to 1. In Eq.(7), all the numbers of 1, 2 and 3 are transformed to 2. In Eq.(8), all the numbers of 1 and 2 on the left of the run of number 3 are transformed to 1, and all the numbers of 2 and 1 on the right of the run of number 3 are transformed to 2. In Eq.(9), all the numbers of 3 and 2 on the left of the run of number 1 are transformed to 2, and all the numbers of 2 and 3 on the right of the run of number 1 are transformed to 1. All the other numbers of d[i,j] are transformed to 0. Let the modified d[i,j] be designated l[i,j] and we call it a locus data.

Each value of l[i,j] has the information about the direction in which the associated passer-by are walking. If a passer-by are walking from left to right, the locus data are equal to 1. If a passer-by are walking from right to left, the locus data are equal to 2. Fig. 5 shows the result where this transformation is applied to the example of Fig. 4.

In this way, we can extract a locus line that is associated with each passer-by.

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0 0 0 2 2 2 2
1 1 1 1 1 1 1 0 0 2 2 2 2 2 2 2
0 0 0 1 1 1 1 0 0 2 2 2 2 0 0 0
0 0 0 2 2 2 2 0 0 1 1 1 1 0 0 0
2 2 2 2 2 2 2 0 0 1 1 1 1 1 1 1
2 2 2 2 0 0 0 0 0 0 0 0 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Fig. 5 An example of extracting locus lines

4 Measurement of parameters

4.1 Number of passers-by and moving direction

The locus data include the locus line associated with a passer-by. By generating the label every time when the locus line occurs and propagating the label along the locus line, we can recognize each passer-by. If the locus line is disconnected as shown in Fig. 5, we can connect this disconnected locus line by considering its location and direction.

By counting the number of the generated labels, we can measure the number of passers-by. Since the each locus data have the information about the moving direction, we can concurrently measure the moving direction of each passer-by.

4.2.Moving speed

The gradient of the locus line is reflected by the moving speed of a passer-by. But the relationship between the moving speed and the gradient of the locus line depends on the distance between the passer-by and the ITV camera. Since it is difficult to calculate this relationship in a simple way, we discussed the other method in the following[5].

We define some parameters in the scene taken by the ITV camera.

h: the number of pixels that exist between the bottom and the center of the scene(=128),
k: the number of pixels that exist between the bottom of the scene and the foot of a passer-by,
l: the distance between the ITV camera and the place that is displayed at the bottom of the scene,
x: the distance between the ITV camera and a passer-by,
θ: Camera angle /2.

Let the moving distance of a passer-by be designated y,

$$y=2hl\,tan\theta/(h-k) \qquad (10)$$

If l is measured in advance, we can calculate the moving distance while a passer-by appears and then disappears in the scene. Since the period t during which a passer-by appears and then disappears in the scene can be easily measured, we can calculate the moving speed s of a passer-by as shown in Eq.(12).

$$s=y/s. \qquad (12)$$

5. Experimental results

When performing the experiment, we assumed that Δ_t in Eq.(2) is equal to 3 and N in Eq.(3) is equal to 3.

5.1 Number of passers-by and moving direction

We considered two cases where the traffic flow is heavy and light respectively, and performed the experiment in each case.

The results in the heavy traffic flow and light traffic flow are shown in Table 2 and Table 3, respectively.

Table 2 Results of measurement(heavy traffic)

Direction :	Right --> Left	Left --> Right
Real	130	22
Measured	107	16
Difference	-23	-6

Table 3 Results of measurement(light traffic)

Direction :	Right --> Left	Left --> Right
Real	19	23
Measured	19	21
Difference	0	-2

5.2 Moving speed

Table 4 shows the results where the moving speed of randomly selected 5 passers-by was measured. The error rates of the period, moving distance and moving speed are from 0 to 7.5%, from -2.4% to 16.9% and from -8.1% to 16.7%, respectively.

5.3 discussions

5.3.1 Number of passers-by

We can find out from Table 3 and Table 4 that there never exists the case where the measured number of passers-by was greater than the real number of passers-by. The error rate in case of light traffic is small, but the error rate in case of heavy traffic is large. This is why when some passers-by are overlapped, our system cannot recognize each passer-by separately and must count one for some passers-by.

Table 4 Result of measurement

No.		Time(s)	Distance(m)	Speed(km/s)
	Real	2.3	4.1	6.4
1	Measured	2.4	4.1	6.2
	Difference	0.1	0.0	-0.2
	Real	4.0	4.1	3.7
2	Measured	4.3	4.0	3.4
	Difference	0.3	-0.1	-0.3
	Real	2.4	4.1	6.2
3	Measured	2.5	4.1	5.9
	Difference	0.1	0.0	-0.3
	Real	7.8	7.7	3.6
4	Measured	7.9	7.9	3.6
	Difference	0.1	0.2	0.0
	Real	2.2	7.7	12.6
5	Measured	2.2	9.0	14.7
	Difference	0.0	1.3	2.1

To meet this problem, we are discussing the methods that (1) the density values of the legs of passers-by are taken, and the number of legs is counted, and (2) two images are taken by using two ITV cameras and then these images are analyzed three-dimensionally.

5.3.2 Moving speed

We can find out from Table 5 that the moving speed of all passers-by could be measured within an acceptable error range except for one passer-by(passer-by number 5 in Table 4). Since this passer-by was running and strode over checking points, walking distance could not be measured preciously. To meet this problem, more checking points have to be placed in the scene taken by the ITV camera.

There existed two case where the moving speed could not be measured. In one case, since a passer-by was running, the locus line was disconnected so long that it could not be correctly traced. In the other case, since a passer-by was outstepped by the other passer-by, the locus line could not be correctly traced. To meet the former problem, we have to introduce a more powerful PC. To meet the latter problem, we have to devise a method that can correctly trace the locus line even if the passers-by are overlapped.

6. Conclusion

In this research, we studied the measurement system where such primitive traffic parameters as the number of passers-by, their moving direction and moving speed can be measured by processing a ITV image with a personal computer.

In this method, the small and restricted area of the scene taken by the ITV camera is processed, and the movement of passers-by is observed in the form of a locus line. By counting the number of locus lines and calculating the gradient of the locus lines, the number of passers-by is measured together with the moving direction. The moving speed of a passer-by is calculated by measuring the period during which the passer-by appears and then disappears in the scene. Since it is very easy to implement these algorithms, the above mentioned parameters can be measured in real time and concurrently.

Using this system, we could measure in real time the number of passers-by, their moving direction and moving speed within an acceptable error range in a light traffic flow.

In the next research step, we have to discuss how we can improve this algorithm to measure with less error rate in case of a heavy traffic flow. Besides, we are discussing the method that the traffic flow of vehicles can be measured using a transputer equipped in a personal computer.

References

[1] W. Hwang and S. Takaba,"Real-time measurement of pedestrian flow using processing of ITV images," The Transactions of the Institute of Electronics and Communi-cation Engineers of Japan, Vol. J66-D,No. 8, pp.917-924(Aug. 1983).

[2] Y. Kudo,"Traffic flow measurement system using image processing," The Transactions of the Institute of Electronics and Communication Engineers of Japan, Vol. J68-D, No. 3, pp.308-315(Mar. 1985).

[3] M. Yamamoto,"Motion analysis by visualized locus method," The Transactions of the Information Processing Society, Vol. 22, No.5, pp.442-449(Sept. 1981).

[4] H. Yoda, J. Motoike, M. Ejiri and T. Yuminaka,"A measurement method of the number of passengers using real-time TV image processing techniques," The Transactions of the Institute of Electronics and Communication Engineers of Japan, Vol. J69-D, No. 11, pp.1679-1688(Nov. 1986).

[5] M. Yamamoto,"Processing of continuous image," The Journal of the Institute of Electrics Engineers of Japan, Vol. 105, No. 5, pp.417-420(May 1985).

[6] T. Agui and M. Nakajima,"Introduction to image engineering," Shoko-do(1988).