# Real-time Visual Measurements Using High-speed Vision

Yoshihiro Watanabe, Takashi Komuro, Shingo Kagami and Masatoshi Ishikawa

Department of Information Physics and Computing,
University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

## ABSTRACT

Real-time image processing at high frame rates could play an important role in various visual measurements. Such image processing can be realized by using a high-speed vision system imaging at high frame rates and having appropriate algorithms processed at high speed. In this paper, we describe two visual measurements using high-speed vision, target counting and rotation measurement. For those measurements, we propose methods utilizing unique features of our high-speed vision system. Both measurements have excellent measurement precision and high flexibility because of high-frame-rate visual observation achievable. Experimental results show the advantages of high-speed vision compared with conventional visual systems.

**Keywords:** visual measurement, real-time image processing, high-speed vision, vision chip, tracking

## 1. INTRODUCTION

Measurements using vision are useful in wide range of applications because visual systems enable contactless measurement and highly flexible operation. In conventional vision-based systems, however, which mainly consist of cameras for imaging and computers for processing, the frame rate is limited by the video signal, for example, 33 ms for NTSC, which is a serious constraint on the performance of visual measurements.

Recently, a system called high-speed vision has been developed to overcome this problem. High-speed vision can overcome the frame rate limitation in conventional visual systems and enables high-speed image processing with over 1000 frames/s. Using this device, some of the advantages of real-time visual measurements become evident. For example, targets moving at high speed and with irregular motion can be measured with high precision. We can also create mechanical systems for robot control and inspection that respond to the rapid changes in the motion of such objects.

In this paper, we use a so-called vision chip, which is one of the most commonly used components for implementing high-speed vision. The vision chip achieves high-speed visual feedback at 1 kHz by integrating sensors and processors together in one chip. Therefore, as the output of a vision chip, we can obtain only desired characteristics of an input image instead of the input image itself, which allows high-frame-rate imaging. Also, the vision chip has several benefits such as compactness, low cost, and low power consumption because it is a VLSI-based system. We have developed a general-purpose vision chip that consists of programmable processing elements. For this vision chip, we propose a multi-target tracking algorithm. This processing utilizes the unique features of the vision chip, such as the high frame rates and pixel parallel operation. Simultaneous observation of multiple targets at high frame rates can be realized by this algorithm.

Two real-time visual measurements using high-speed vision are described in this paper. Both of them are based on multi-target tracking processing in the vision chip. One of the applications is target counting. This application counts the cumulative number of targets coming into sight. This can also count only desired targets. At a frame rate of 0.5 kHz, we found experimentally that the maximum target speed was 20 km/h. This counting technique could be applied, for example, to high-speed inspection of small creatures and crops. Another possible application is the measurement of high-speed rotation. In this application, the rotation axis and rotation speed of a ball are measured in real-time. As a result of experiments at 1-kHz frame rate, the maximum rotation speed of the objects was 1200 rpm. We believe that such rotation measurement could find applications in ball games, such as soccer and baseball.

In section 2 of this paper, we briefly describe the vision chip and its applications. Also included in section 2 is a description of the multi-target tracking algorithm developed for the vision chip. In sections 3 and 4, we

explain the experimental system and some results of our target counting and rotation measurement algorithms, respectively. Section 5 concludes our paper.

## 2. VISION CHIP
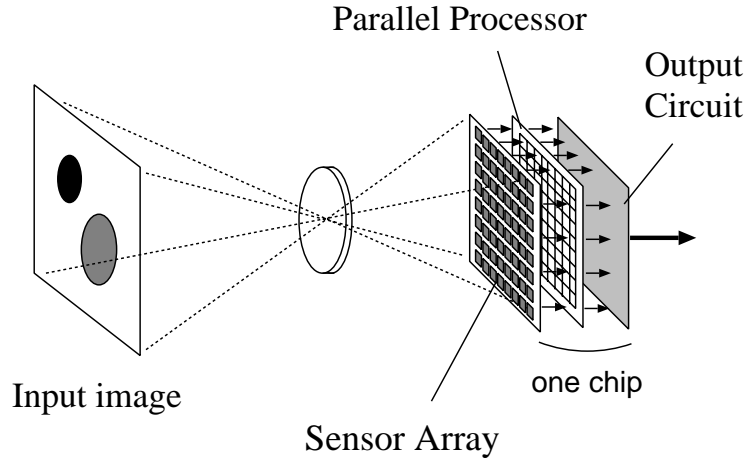
### 2.1. Basic architecture and applications



**Fig. 1.** Model of a vision chip

Fig. 1 shows a model of the vision chip. A vision chip is a device in which a photo detector and a processing element are provided in each pixel. Therefore, a vision chip integrates an imaging system and a processing system on one chip, systems which have been separated conventionally. This architecture provides some advantages in comparison to the traditional approach. These advantages include high-speed image processing, compactness, low power dissipation, and so on.

There are many examples of vision chips, most being made by analog circuits.[1] The pixel size of these analog vision chips is small because the number of necessary transistors is low. Therefore, the resolution of analog vision chips is generally high, but it is difficult to realize various processing operations on them. On the other hand, with the progress of semiconductor integration technology it has become possible to realize vision chips made by digital circuits. While digital vision chips require more area than analog ones, they are capable of various image processing operations. This versatility of digital vision chips is important in practical applications.

In this paper, a general-purpose digital vision chip system we have developed[2,3] is used as high-speed vision for real-time visual measurements. Fig. 2 shows a photograph of this vision chip system, which consists of a chip and its controller. The resolution is $64 \times 64$. The programmable processing elements enable various image processing operations controlled by software. Also, global features such as summation and the centroid of an entire image can be extracted at high-speed.

Vision chips allow precise observation of objects moving at high-speed and with irregular motion in real-time. Also, high-speed visual feedback enables rapid system response, even to rapid environmental changes. The vision chip has been applied to various fields, such as robot control[4] and microscope observation.[5] These applications utilize a tracking process but with tracking of only a single target, since they are based on simple matching of a tracked target. A labeling process, which is suitable for execution on a vision chip, is needed to track multiple targets. To solve this problem, in the next section we propose a multi-target tracking algorithm for a vision chip by adding a new labeling algorithm to the conventional tracking algorithm.

### 2.2. Multi-target tracking algorithm for vision chip

We propose a multi-target tracking algorithm utilizing the unique features of our vision chip, such as the high-frame rate and pixel parallel operation. As will be shown below, these features allow the algorithm to be simplified, thus realizing high-speed processing. The input images for this algorithm are assumed to be binary.
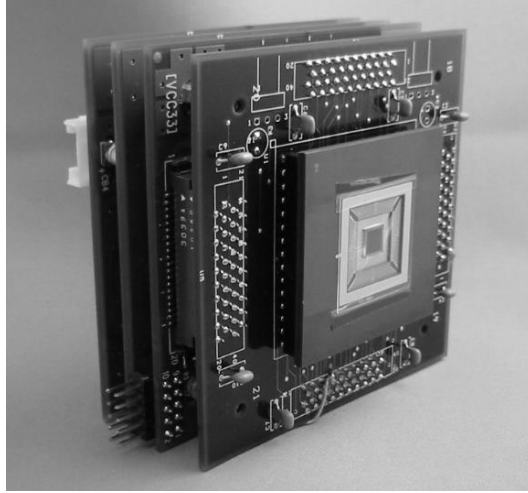
**Fig. 2.** General purpose vision chip system

The algorithm consists of a labeling process and a tracking process. Labeling is the initial process to generate new targets, and tracking is the process to carry out regional division and matching of each target between frames. This means that labeling is applied only to frames in which the number of targets changes and tracking is carried out for every frame, as long as the target remains in the image.

As the labeling algorithm, we use a method using binary search of an entire image. Binary search enables efficient labeling. Moreover, this labeling method is executed at high-speed by using pixel parallel operation. Fig. 3 shows the flow of this labeling algorithm, and the algorithm is described in detail below.

1. **Obtaining a starting point** $g_k^i = S(I_k)$
   –Search for a row including regions by dividing image $I_k$ at time $k$ into two.
   –Obtain a point $g_k^i$ in a region $i$ by searching for the column.
   –Let the point $g_k^i$ obtained be the starting point of a region.
2. **Restoration of labeled region**
   $$\begin{cases} {}^0C_k^i = g_k^i \\ {}^{j+1}C_k^i = D_1({}^jC_k^i) \cap I_k \quad (j = 1, 2, \ldots, n_l) \end{cases}$$
   –Restore the image $C_k^i$ of the region $i$ from the original image $I_k$ and the starting point $g_k^i$ by repeating 1-pixel dilation $D_1$ and logical multiplication $n_l$ times.
3. **Input image update** $I_k' = I_k \cap \overline{C_k^i}$
   –Update the image by deleting the labeled region $C_k^i$ to form a new input image $I_k'$.
4. **Repeat**
   –Repeat steps 1 to 3 $M$ times
   ($M$ is the number of regions that need to be labeled.)

In our vision chip, obtaining a starting point takes $40 \sim 50$ $\mu$s and restoration of the labeled region takes 3 $\mu$s each time.

New tracking target regions are generated by the labeling process. Then, after initializing the frame, the tracking process is applied. We use a previously proposed method[6] for high-speed vision to match the target regions between frames in the tracking process. This novel method is based on the high-frame-rate imaging of our vision chip, which allows us to assume that any moving region overlaps itself between two successive frames. Under this assumption, the matching algorithm can be described simply by logical multiplication and a dilation operation. Fig. 4 shows an outline of the tracking process, and actual algorithm is described in detail below.
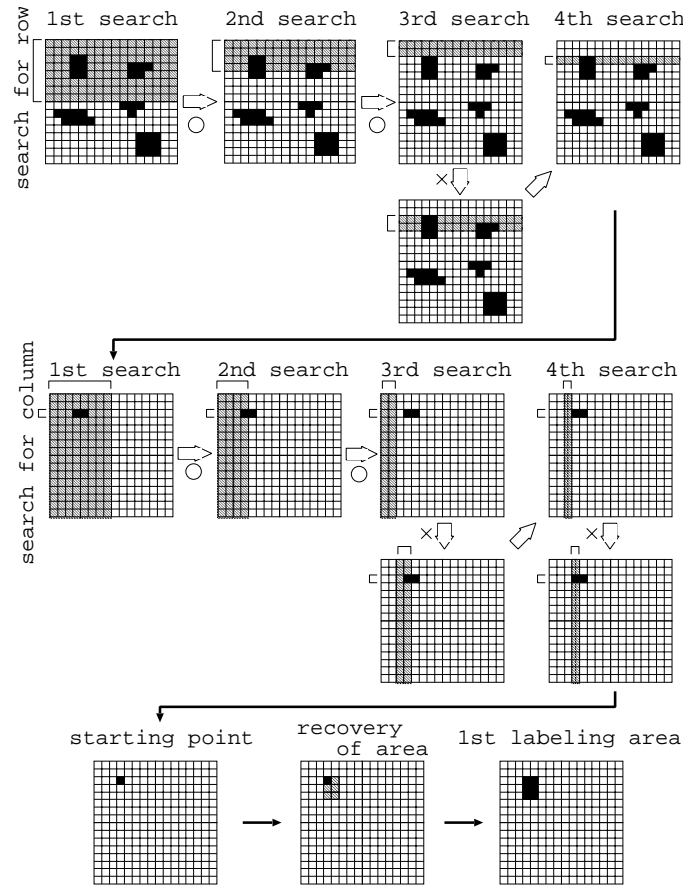
**Fig. 3.** Labling algorithm using binary search

## 1. Restoration of tracked region

$$\begin{cases} {}^{0}C_k^i = C_{k-1}^i \\ {}^{j+1}C_k^i = D_1({}^{j}C_k^i) \cap I_k \quad (j = 1, 2, \ldots, n_t) \end{cases}$$

–Restore the image $C_k^i$ of a tracked region $i$ at time $k$ from the input image $I_k$ and the image $C_{k-1}^i$ by repeating 1-pixel dilation $D_1$ and logical multiplication $n_t$ times.

## 2. Repeat

–Carry out step 1 for multiple tracking images in turn.

The number $n_t$ is determined by the amount of the motion according to each application. The processing time for restoration of a tracked region is the same as that for the labeling process. We believe that collision and separation between some regions, which are common problems in tracking, can be detected from changes of trajectory and area based on high-frame rate observation, at least in simple cases.

Real-time observation of multiple targets in image sequences at high frame rates is possible with our multi-target tracking using our vision chip. The visual measurements described in this paper are based on this image processing. Two applications, target counting and rotation measurement, are described in the next two sections. Both of them realize contactless and high-speed measurements. Also, flexible measurements can be carried out by adding suitable image processing to our programmable vision chip.
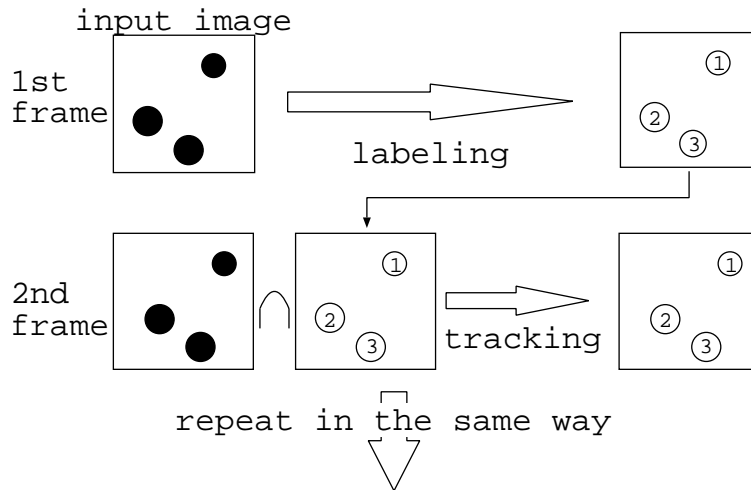
**Fig. 4.** Tracking process

## 3. TARGET COUNTING

### 3.1. Method

Target counting is a measurement that counts the number of regions that newly appear in an image. This counting is realized at high-frame rate by using multi-target tracking via the vision chip.
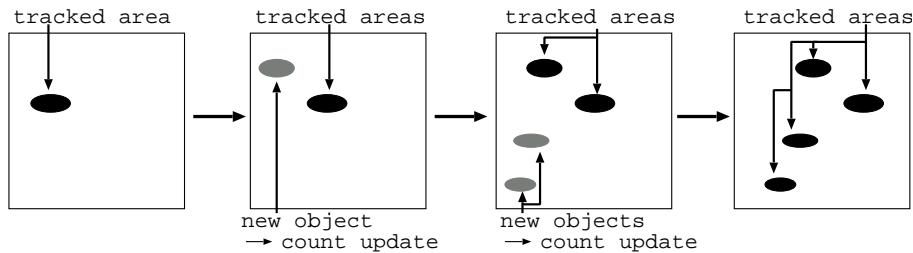


**Fig. 5.** Method of target counting

Fig. 5 explains how the cumulative number is obtained by the target counting method. In multi-target tracking, the regions in an input image are divided into two groups, new regions and tracked regions. New regions are obtained by deleting the tracked regions from the input image. Counting of the cumulative number is achieved by updating the result for each new region. A labeling process for generating new targets in multi-target tracking is used for updating the result.

The tracking process in target counting ensures the accuracy of the measurement. For example, the error that occurs when the same region is counted several times can be avoided even when many targets in the image sequences move freely. In addition, image processing on a vision chip enables the counting of targets with desired shape and size.

This target counting enables precise measurement of various objects at high-speed. For example, as an application of this target counting we consider the counting of small fish swimming over a line. This kind of application is very difficult to automate because the targets move very quickly and irregularly.

### 3.2. Experimental results

Fig. 6 shows the target counting system used for this experiment. The system consists of a vision chip, its controller, and a computer. The computer obtains the counting result held in the vision chip and displays it.
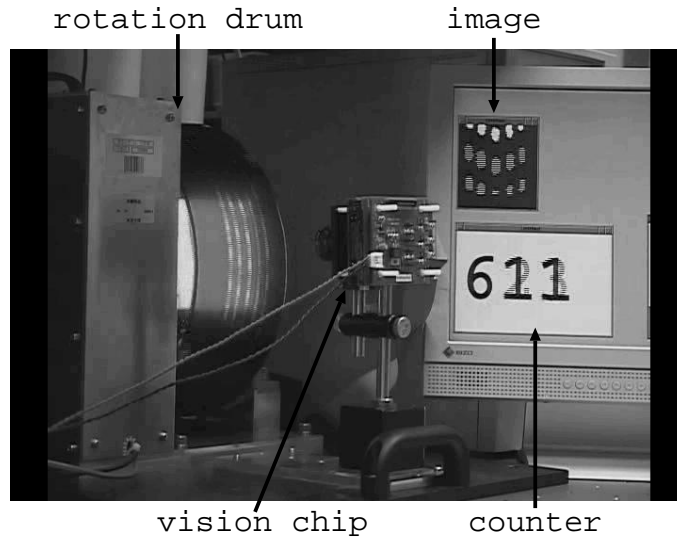
**Fig. 6.** Target counting system for experiment

In this experiment, there is no synchronization of the data passing between the vision chip and the computer. As targets, 100 regions are set on a rotating drum with a radius of 15 cm. The size of the targets is about 1.0 cm (10 pixels) in the direction of the movement. The cumulative number increases in cycles because the target movements are controlled by the drum rotating at a constant speed. The frame rate is 0.5 kHz. This frame rate is determined by the number of targets that newly appear in one frame. In this experiment, the number is 15. In addition, as a de-noising condition in this experiment, only targets larger than a fixed size are counted.

Fig. 7 shows the experimental results of target counting. Fig. 7(a) is the result of counting the cumulative number when the drum rotates at 254 rpm. The result increasing periodically shows that the counting was carried out correctly.
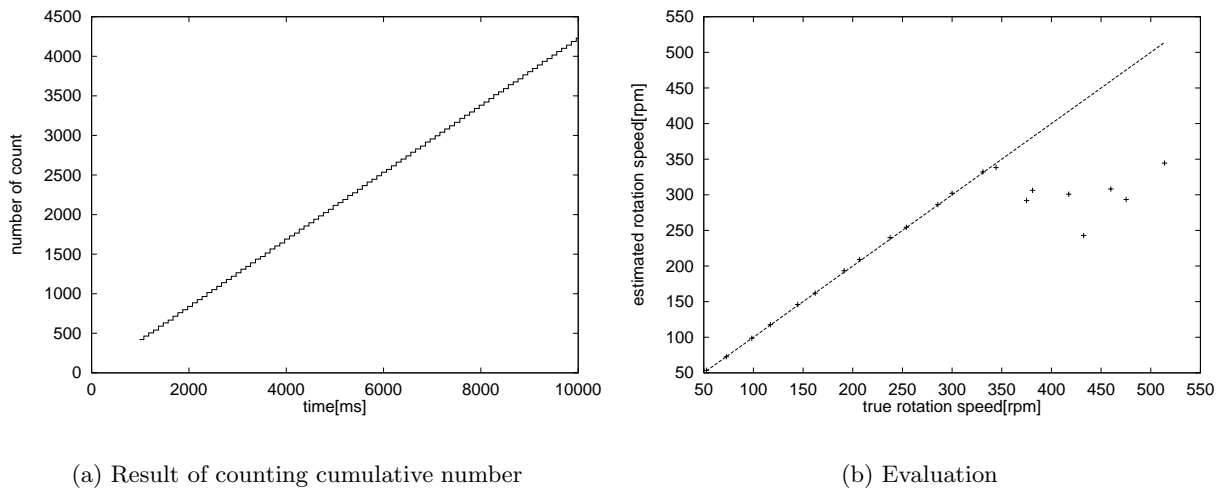


(a) Result of counting cumulative number

(b) Evaluation

**Fig. 7.** Experimental result of target counting

The counting result at various speeds is evaluated in Fig. 7(b). This figure compares the rotation speed of the drum, estimated from the result of the cumulative number, with the true rotation speed obtained by a motor encoder. The vertical axis in the graph is the estimated speed and the horizontal axis is the true speed. The straight line in the graph passes through the origin and the gradient is 1. Therefore, the points on the line are

accurate. As shown in Fig. 7(b), the target counting is accurate up to about 350 rpm.

In this experiment, the rotation speed limit of the drum is about 350 rpm, which corresponds to a target speed limit of about 5.5 m/s (5500 pixels/s). This shows that our target counting can be applied to quickly moving targets. The limit occurs because the assumption in the tracking process that the same region overlaps with itself between frames cannot be fulfilled.

## 4. ROTATION MEASUREMENT

### 4.1. Basic concept

This rotation measurement obtains rotation information on a rotating object in real-time. The rotation information is the rotation axis and rotation speed. High-frame-rate observation enables precise measurement even when an object changes its motion quickly and randomly. In this paper, the shape of the object is spherical.

As an application of this rotation measurement, we consider ball games. For example, the rotation speed of a soccer ball in a free kick reaches about 600 rpm and a baseball at pitching reaches about 1800 rpm. It is very difficult for conventional vision systems to realize the performance required for these applications. We also consider robot control applications based on similar measurement.

This measurement estimates the rotation information from the distribution of speed vectors on a sphere. The vectors are obtained by multi-target tracking. The method includes three processing steps as follows: (1) Obtaining two-dimensional trajectories of some regions on a sphere. (2) Determining three-dimensional speed vectors. (3) Estimating rotation information. Each processing step is described in the next section.

The measurement system consists of three devices, a vision chip, its controller, and a computer, like target counting. Processing (1) is carried out in the vision chip and its controller. Processing (2) and (3) are carried out in an external computer.

### 4.2. Processing for measurement

In processing (1), trajectories of $m$ target regions for $n$ frames are obtained by multi-target tracking. The positions of each target are given by the centroids of the regions. Therefore, the positions can be obtained to sub-pixel precision.

In processing (2), three-dimensional speed vectors are determined from two-dimensional trajectories approximated by $n$ positions. The calculation uses orthogonal projections and the assumed spherical shape.

In processing (3), rotation information is estimated based on the following equations. The center of the sphere is assumed to be the origin. The origin corresponds to the center of an image. The imaging area is parallel to the $xy$ plane. A sphere is imaged from the positive $z$ direction.

Rotation axis $\boldsymbol{l}$ is obtained from equation (1) by using the least squares method. Speed vector $\boldsymbol{v}$, position vector $\boldsymbol{r}$ of each target, and radius $R$ of the sphere in an image are needed for this equation. The radius and correction value for the center of the sphere are obtained from the input image.

$$l_x v_x + l_y v_y - \frac{r_x v_x + r_y v_y}{\sqrt{R^2 - r_x^2 - r_y^2}} = 0 \tag{1}$$

Rotation speed $\omega$ is approximately calculated by the equation (2). This equation needs a speed vector $\boldsymbol{v_0}$ at position $\boldsymbol{r_0}$ and distance $h$ between the origin and the plane passing through the position $\boldsymbol{r_0}$ and orthogonal to the rotation axis $\boldsymbol{l}$. Also the equation is based on an assumption that $\boldsymbol{v_0}$ is small enough.

$$\omega = \frac{|\boldsymbol{v_0}|}{\sqrt{R^2 - h^2}} \tag{2}$$
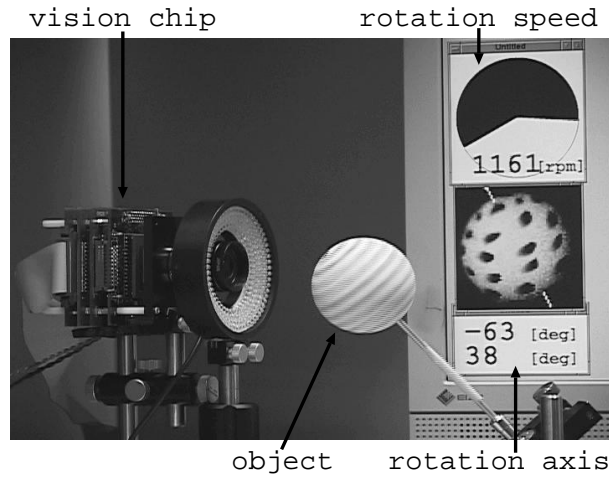
**Fig. 8.** Rotation measurement system for experiment

## 4.3. Experimental results

Fig. 8 shows the rotation measurement system used in the experiment. In this experiment, the number of tracked targets is 10 and the number of positions for obtaining the trajectory is 10. These numbers are equivalent to $m$ and $n$ in the section 4.2, respectively. The values of both $m$ and $n$ must be at least 2 in order to obtain a solution. Greater accuracy is achieved with higher values. Also, the frame rate is 1 kHz, and a pattern is applied to the sphere for easy extraction of regions from an image.
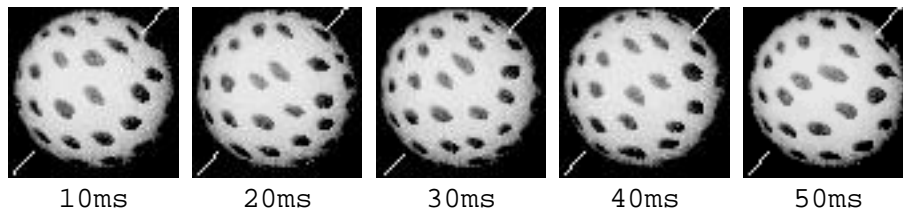


| 10ms | 20ms | 30ms | 40ms | 50ms |

**Fig. 9.** Image sequences of measured rotation axis

The results obtained when the rotation axis is defined by $\theta \approx 45°$, $\gamma \approx 30°$ and the rotation speed is fixed at 335rpm are shown in Figs. 9, 10(a). $\theta$ is the angle of the rotation axis in the $xy$ plain. $\gamma$ is the angle made by the rotation axis and the $xy$ plain. Fig. 9 shows image sequences of the measured targets; the rotation axes are drawn in the images.

Fig. 10(a) shows the distribution of the rotation axis measured 200 times. The histogram drawn in the solid line is the result for $\theta$ and the one in dashed line is for $\gamma$. The experimental conditions were the same as in Fig. 9. The results are distributed centered on a fixed angular value. As shown in this figure, the rotation axis was measured almost correctly, considering that the axis can only be fixed roughly and the rotation axis of the observed object is not completely stable because of the experimental environment.

Fig. 10(b) shows the evaluation of the measured rotation speed. This figure compares the result with the true speed. The vertical axis is the measured value and the horizontal axis is the true one. The plotted points are the experimental results. The straight line in the graph passes through the origin and the gradient is 1. As shown in Fig. 10(b), the rotation measurement is accurate up to about 1200 rpm.

The 1200-rpm limit occurs for the same reason as the limit in the target counting. The result shows that this system can measure the rotation information even of an object rotating at high speed. Considering the application to ball games, in particular, soccer, this system enables measurement of a ball during a free kick,
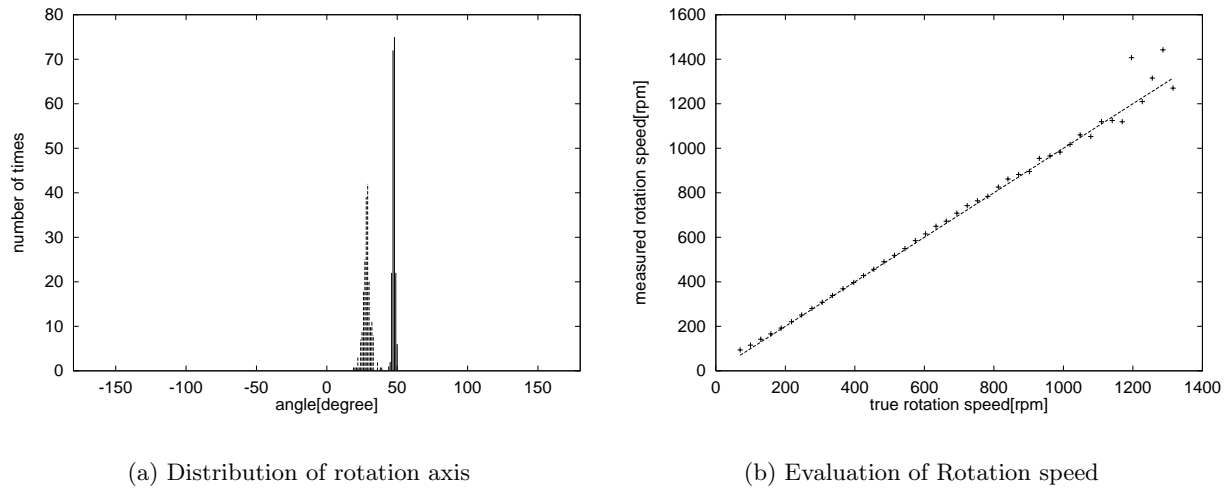
(a) Distribution of rotation axis

(b) Evaluation of Rotation speed

**Fig. 10.** Experimental result of rotation measurement

whose rotation speed is about 600 rpm. On the other hand, in baseball, when the ball is pitched, its rotation speed is about 1800 rpm, which is difficult to be measured in this experimental condition. However, such measurement is possible in our system because the frame rate can be improved by reducing the number of tracked regions $m$. We also can improve the frame rate by designing the processing elements of the vision chip so as to be specialized for multi-target tracking.

## 5. CONCLUSION

We describe the usefulness of real-time visual measurements using a high-speed vision chip that overcomes the frame-rate limit of conventional systems. We describe target counting and rotation measurement based on multi-target tracking. The experimental results using the vision chip we have developed show that both algorithms excel in the points, high-precision, high-frame-rate observation and flexibility. These visual measurements, providing such advantages, are expected to be applied in various fields, such as inspection, industrial applications, sport measurement, robot control, and so on.

## REFERENCES

1. A. Moini, VISION CHIPS, Kluwer Academic Publishers, 2000.
2. T. Komuro, S. Kagami,and M. Ishikawa: "A New Architecture of Programmable Digital Vision Chip," Proceeding of 2002 Symposium on VLSI Circuits, pp. 266-269, 2002.
3. S. Kagami, T. Komuro, I. Ishii,and M. Ishikawa: "A Real-Time Visual Processing System using a General-Purpose Vision Chip," Proceedings of 2002 IEEE International Conference on Robotics and Automation, pp. 1229-1234, 2002.
4. Y. Imai, A. Namiki, K. Hashimoto,and M. Ishikawa: "Dynamic Active Catching Using a High-speed Multifingered Hand and a High-speed Vision System," Proceedings of 2004 IEEE International Conference on Robotics and Automation, pp.1849-1854, 2004.
5. N. Ogawa, H. Oku, K. Hashimoto,and M. Ishikawa: "Motile Cell Galvanotaxis Control using High-Speed Tracking System," Proceedings of 2004 IEEE International Conference on Robotics and Automation, pp. 1646-1651, 2004.
6. I. Ishii and M. Ishikawa: "Self Windowing for high speed vision," Proceedings of 1999 IEEE International Conference on Robotics and Automation, pp. 1916-1921, 1999.