The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Low cost synchronized stereo aquisition system for single port camera controllers

Bennet Fischer, Hamid Reza Moballegh, Raul Rojas
Free University of Berlin

Abstract— **This paper presents a method for creating a low cost stereo vision system which fits perfecty to the needs of cheap mobile robots. As mass-produced low cost CMOS cameras have no direct means for taking synchronized images, we present a method for taking synchronous images with these devices by manipulating their system clock. We also show a method of merging the data of two or more cameras for interfacing all of them to a single camera controller.**

I. INTRODUCTION

The sense of the environment is an important aspect of intelligent autonomous systems. For making accurate actions, the system needs a detailed knowledge of its surroundings. One popular method is to sense the environment by using area cameras, also known as 2D cameras. As the name implies, these devices only deliver a two-dimensional projection of the real world. Under certain conditions of the evironment this permits to estimate the distance of objects. One precondition for such a method could be a fixed size of inspected objects which is well known. As the size of the projection is antiproportional to the distance of the object, its distance can be easily obtained by the help of the camera intrinsics.

In case of no knowledge about the environment this approach fails. One common way of solving this problem is by adding a second camera with a fixed displacement to the first camera. This approach is often referred to as “Stereo Vision” [1]. It allows to determine the distance of arbitrary objects by finding the same real world point in the two camera images and measure the pixel offset in the two images. This offset has a reciprocal relation to the distance of the point from the cameras.

For realizing such a stereo vision system it is necessary to capture images of two different cameras. Also if the cameras or the objects observed move, the two images have to be taken at the same point of time [4]. This condition is called synchronized image capture.

A. Computing system

In the area of autonomous robots the computing system often consists of an embedded system which is optimized for a small form factor and low power consumption. Recent developments led to systems powerful enough for doing stereo vision in realtime. Most of these systems are powered by an ARM CPU and have a rich set of pepherials, often including a camera controller. However, most of the embedded systems in nowadays have only one single camera controller. They are not able to interface with multiple cameras. Therefore stereo vision is not possible with off-the-shelf cameras.

B. Camera

Mass production offers the possibility of great cost savings. As nearly every mobile phone is nowadays equipped with CMOS camera sensors, these devices are available at a very low price. Thus they are generally well suited for creating a low cost stereo vision system. But due to their limited field of application they are kept as simple as possible. As synchronous image capture is not necessary for mobile phones, this feature is also omitted. Because of that these cameras cannot be used directly for synchronized image capture.

C. Our Method

Because of the shortcommings of low cost hardware for implementing a viable stereo vision system, we used a method for resolving the two problems mentioned above (single port camera controllers and unsynchronized cameras). By manipulating the cameras clock we archived to synchronize cheap mass produced cameras up to a single pixel accuracy. Further on we merged the data of two cameras for being able to interface them to a single camera controller. All these operations are made on a low end Complex Programmable Logic Device (CPLD). This leads to a Bill Of Material (BOM) for a VGA stereo vision system well below ten dollars, including the two cameras. Also any embedded system with a single camera controller can be used instead of specialized systems with dual port camera controllers.

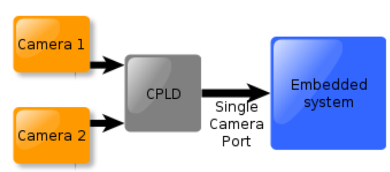

Fig. 1. System configuration

II. RELATED WORK

A system for multiplexing multiple video streams has been proposed in [5]. It uses programmable logic for merging the camera data. As this system is only designated to analog cameras, it is not possible to use cheap digital CMOS cameras. Also the synchronization of the cameras is not guaranteed as the system is not designed for stereo vision applications.

A system for stereo vision in autonomous robots was proposed in [6]. This approach aims at the use of mass produces CMOS cameras and interfaces them by using a programmable logic. However the problem of synchronized image capture was not addressed in this proposal. Thus their solution is only applicable for static scenarios.

III. Camera synchronization

Used in a mobile phone the cameras receive a continuous clock and record frames in a constant frequency but with an undefined offset. For the use in a stereo vision application we have to assure that both cameras have also a well defined offset. This is achieved by keeping the clock of one camera constant and manipulating the clock of the other cameras. The behaviour of this offset adaptation can be seen in Fig. 2. The first five signals show the signals between the CPLD and the embedded system following by the signals between the cameras and the CPLD. Important points are marked by red numbers.

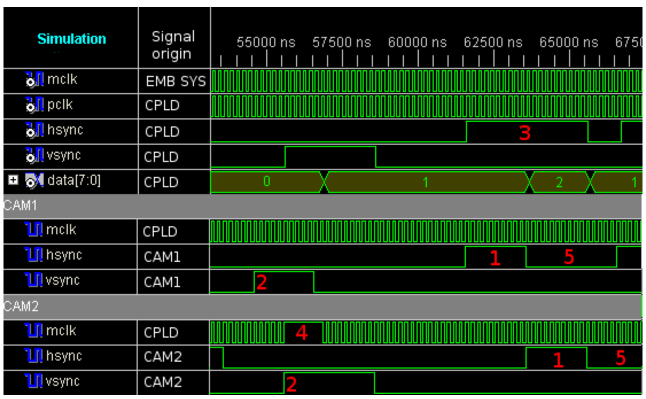

Fig. 2. Simulation of synchronization and merge of two cameras

The rising edge of the vertical synchronization signal of the two cameras at the beginning (2) shows that they have started with a different phase. Therefore the CPLD stops the clock of camera two (4). This state is held until a well defined offset between the two cameras is reached. The offset is the exact length of one camera line. This is necessary for the later merging. Once the system is synchronized this way, it keeps the synchronization implicitly for all the following frames. Additionally every frame a check of the synchronization is made. This is achieved by counting the number of clocks between the rising edges of the VSYNC signals of the two cameras. Due to the synchronization this number is supposed to be the length of one camera line. If this is not true the process seen in Fig. 2 is repeated until the synchronized state is restored again.

IV. Camera merging

For interfacing two seperate cameras to a single camera controller, the data of the cameras has to be merged. This behavior is shown in Fig. 2. As a preparation the blank periods (5) between two lines of the cameras are configured to be long enough to hold one line of the opposite camera. Due to the displacement in the synchronization of one line length mentioned in the previous section, a line from camera one is immediately followed by a line of camera two. Depending on which camera is currently active, its output is redirected to the embedded system. The two horizontal synchronization signals (1) are logically ORed which gives a horizontal synchronization signal to the embedded system (3) which covers the two camera lines. To the embedded system the two cameras appear in a single frame side by side.
By using this method the camera data is merged linewise. The two other obvious solutions, pixelwise and framewise merging are exceeded by this method. Framewise merging introduces a noticeable delay between the two captures and is therefore not suitable. Pixelwise merging combines the two frames in one framebuffer. Due to the structure of most memories an access to a single pixel would always transfer the pixels of both images. This introduces a memory bandwith usage which is twice as high as necessary. As memory bandwith is a crucial factor in vision systems this drawback is not desirable.
An important characteristic of our solution is that both cameras have an uninterrupted clock once they are synchronized. This is necessary because also the configuration interface of the camera is clocked by the master clock of the camera. Whithout this characteristic the camera would not be configurable.

V. Implementation

As stated before, camera 1 receives a continuous clock from the embedded system and acts as a master camera. All synchronization is done relative to this camera. The system is implemented as a finite state machine (FSM) as shown in Fig. 3. The text on the edges describes the condition for a state transition. This FSM assures that the cameras have a defined offset of exactly one camera line length. This condition is checked at the beginning of each frame.

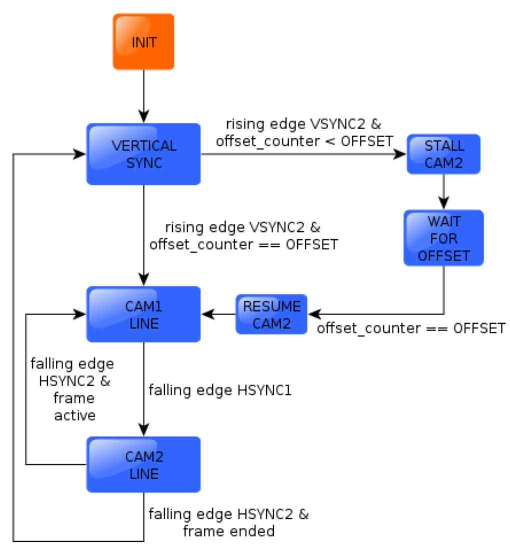

Fig. 3. FSM of the synchronization

The offset counter is set to zero at the rising edge of the vertical synchonization of camera 1. It is incremented every clock cycle.
The end of a frame is detected by counting the image lines and compare that value with the image height.
The merge of the camera data from the implementational point of view is a simple multiplexer dependent of the FSM states in Fig. 3. The following pseudo-code describes the multiplexers for the data and clock and the logic for the synchronization signals.

```
DATA_OUT <=
  CAM1_DATA when fsm_state = CAM1_LINE else
  CAM2_DATA when fsm_state = CAM2_LINE else
  0;

CAM2_MCLK <=
  1 when fsm_state = WAIT_FOR_OFFSET else
  MCLK_IN;

HSYNC_OUT <=
  CAM1_HSYNC OR CAM2_HSYNC;

VSYNC_OUT <= CAM2_VSYNC;
```

This pseudo code describes the behaviour seen in Fig. 2. The data is multiplexed between the two cameras whereas the two horizontal synchronization signals are combined by a logical OR. The clock of camera two is multiplexed between a logical one and the master clock of the embedded system. The constant logical one value to the camera acts as a stall which preserves the current camera status. This is used to stop the camera and restart it at a given offset. For the vertical synchronization simply the VSYNC signal of camera 2 is passed through to the embedded system.

VI. Conclusion and Future Work

In this paper a method for creating a stereo vision system out of standard parts at a very low price level is shown. The system features very low power consumption, low weight and small physical dimensions. It is therefore ideally suited for mobile application in autonomous systems like humanoid robots.
An realization of the described system shown in Fig. 4 has proved its relieable operation on a humanoid robot at several Robocup competitions [3].

In the future we plan to integrate several image processing operations inside the configurable logic of the camera system by using a logic device with a higher capacity. We already designed a logic which removes the lens distortion and projects the camera images to a common plane and plan to integrate it into our stereo camera system. This step greatly reduces the necessary computing power for doing online stereo vision calculation.

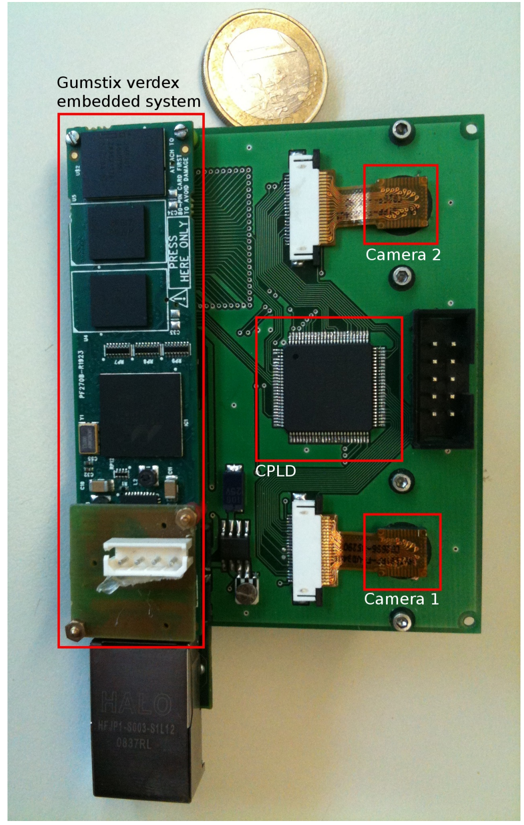

Fig. 4. Realization of the presented system

References

[1] Barnard, S. T. and M. A. Fischler, Computational stereo, *Computing Surveys*, vol. 14, 1982, pp 553-572

[2] K. Sabe, M. Fukuchi, J.-S. Gutmann, T. Ohashi, K. Kawamoto, and T. Yoshigahara, Obstacle avoidance and path planning for humanoid robots using stereo vision, *In Int. Conf. on Robotics and Automation (ICRA), New Orleans*, 2004

[3] http://www.fumanoids.de

[4] Coppin, B., Artificial Intelligence Illuminated, *John and Bartlett Publishers*, 1 ed., 2004

[5] Herbland, A. Sotudeh, R., Real-time digital video multiplexer synchronisation implementation with CPLD , *5th International Conference on ASIC Proceedings.*, vol. 2, 2003, pp 914 - 917

[6] Kei Okada, Takeshi Morishita, M. Hayashi, Masayuki Inaba, Hirochika Inoue, Design and Development of Small Stereo Vision Sensor Module for Small Self-Contained Autonomous Robots, *Journal of Robotics and Mechatronics*, Vol. 17, No. 3, 2005, pp. 248-254