ROAD MONITORING AND OBSTACLE DETECTION SYSTEM BY IMAGE ANALYSIS AND MATHEMATICAL MORPHOLOGY

Serge Beucher, René Peyrard, Michel Bilodeau, Michel Gauthier

Centre de Morphologie Mathématique
Ecole des Mines de Paris
35, Rue Saint Honoré
77305 FONTAINEBLEAU CEDEX (France)
Email: beucher@cmm.ensmp.fr

Abstract

This paper presents the contribution of the Center of Mathematical Morphology to the design of a French demonstrator vehicle, named Prolab2, in the framework of the European PROMETHEUS project. Our main task was to set up an image processor able to perform road/lane segmentation and obstacle detection in real-time. To reach this goal, a multi-pipeline processor associated with dedicated algorithms have been designed.

Two different image processes have been implemented. The first one, road/lane segmentation, is used for selecting regions of interest (ROI) to which the second process, obstacle detection, is applied. The hardware processor must be able to perform these two tasks at the same time. To allow this double processing, the morphological processor is divided in two pipeline units working in parallel. Each pipeline is based on PIMM1, a dedicated morphological ASICs specially designed to perform binary and greytone morphological transformations.

The road tracking uses the watershed transformation. To speed up this transformation, an efficient algorithm, based on anamorphosis, has been used. The main processor, made of 8 pipelined PIMM1 can perform a watershed transformation in 80ms. The second pipeline processor is made of 4 PIMM1 to perform the on-request obstacle detection.

A multi-access image memory board including 32 greytone image frames supplies the two pipelines with image data.

1. Introduction

Along with other French research laboratories and with the car manufacturers PSA and Renault, the Center of Mathematical Morphology (CMM) has been involved in the European PROMETHEUS project for designing a car demonstrator named Prolab2. This demonstrator is equipped with various sensors: two front view, one rear view and two lateral video cameras, a telemeter coupled with one of the front view cameras and a pair of linear cameras used for stereovision. Our contribution to the demonstrator is the design of an on-board real-time image processor, the morphological sub-module (MSM), part of the whole embedded system. Its task is to analyze the images provided either by the front camera or by the rear one and to provide a list and position of obstacles in front of or behind the car transmitted to the system supervisor which, through a man-machine interface, delivers messages and warnings to the driver. Two main tasks are devoted to the MSM. The first one is the segmentation of the road and of the traffic lanes. This task is a background task performed continuously by the processor. The second one is the obstacle detection. This task may be either a background task (information is sent permanently) or a task performed on request by the supervisor.

Firstly, this paper briefly presents the algorithms used for achieving these tasks. A large number of morphological tools used in this project can be found in [3,6,7,13,14,15]. Special adjustments and modifications have been performed for increasing the speed of some algorithms without reducing too much their efficiency and robustness [4]. Secondly, a morphological multi-pipeline processor architecture is described, specially designed to fulfill the communication requirements with other modules and to perform obstacle detection and lane segmentation in real time [8,11]. This hardware is based on a pipeline architecture built with a mathematical morphology ASIC named PIMM1 [10]. It consists in an intelligent board with two system buses and equipped with a multi-tasking operating system and two separate image processors. Then, we describe the morphological pipeline processor board designed to speed up image segmentation.

2. The algorithms

2.1. Road and lane segmentation

Road and lane segmentations are of major importance for the following reasons:
- They provide useful information on the car position with respect to the lane boundaries. Moreover, the nature of the ground marking can be analyzed, with as a final result, an increase of the driver's safety.
- They allow to compute the coordinates of the regions of interest in the scene and in particular those where the obstacle detection algorithm is applied. The selection of regions for obstacle detection has many advantages. The size of the image to be processed is reduced and consequently the computation time. Moreover, the detected road or lanes can be used as a template for rejecting artifacts or false alarms produced by the obstacle detection task. Finally when the obstacle detection is performed on request, this process can be easily focused on a particular lane.
- The detection of the lane boundaries also provides the dynamic parameters allowing the calculation of the 3D real coordinates of objects on the road from their 2D position in the image.

Two lane segmentation algorithms are used. The first one is used during the initialization step and performs a first extraction of the current traffic lane from scratch. The second one is activated in the steady state and uses the previous segmentation to fit a computed model of the lane. This model produces a new segmentation which, in turn, is used to update the road model. Both processes use the same main steps and the same morphological segmentation tool: the watershed transformation. These main steps are the following:
- A morphological temporal filter [14] is applied over a number of successive images to remove noise and to close the white discontinuous (if any) ground marking. The number of successive images is determined by the speed of the car and by the acquisition rate. Although a temporal alternate sequential filter would be recommended, in order to reduce the computation time, a simple temporal dilation is used in practice. Moreover, the size of this dilation (that is, the number of successive images) is lower during the steady state segmentation (tracking mode) than it is during the initial step (acquisition mode). This variation of size leads to a better management of the possible changes of lane during the tracking mode. On the other hand, the acquisition mode needs a larger size of temporal dilation and consequently can only be done when there is no vehicle in the fore-ground (Figure 1).
- An edge detector made of a combination of a morphological gradient and of a top-hat transform is then applied to the previous image [16].
- Finally, a watershed transformation [1] of the gradient image is performed. This watershed transformation is slightly different depending on its use during the acquisition mode or during the tracking mode. In the former case, a simple watershed is performed. Provided that the temporal filter is sufficiently efficient, a simple catchment basin marks the current lane (Figure 2). Its right and left boundaries can then be used to compute a lane model (a straight lane model is used in practice). In the latter case, the previous lane model is used to build a marker of the lane which is injected to the watershed transform of the next image. This new watershed produces a simple catchment basin fitted to the new lane image. Then a new lane model is calculated and the entire process can be run again on the following image (Figure 3).

2.2. Obstacle detection

The second part of the treatment is the obstacle detection. This process is performed in parallel on the regions of interest (ROI) emphasized by the road segmentation. Starting from the position of the lane boundaries, it is easy to compute the coordinates of a window embedding the part of the image corresponding to a given range of distance where the obstacle detection algorithm can be applied.

Two main characteristics of the vehicles are used for their extraction:

- There always exists a darker region in front of the vehicle corresponding to its shadow, the wheels and the bumper. This region can be used as a marker in further segmentation algorithms. Another interesting point is that this region always touches the road.
- The geometry of the vehicles is rather simple. As most of the manufactured objects, vehicles are made of horizontal and vertical contours and their size is a rather constant parameter (between 1m and 1.5m).

The darker region is detected by means of a morphological transform called h-minima. The available markers are then filtered and only those which touch the road and whose size correspond to the dimension of a vehicle are kept (Figure 4). These markers can be used in watershed algorithms applied on directional gradient images or, more simply, used as such.

More refined obstacle segmentation algorithms can also be designed. They are based on hierarchical segmentations [2]. Their complexity comes from the fact that they need two successive watershed transformations.

2.3. Fast watershed algorithm

On non sequential computer hardware, the slowest part of the segmentation is the watershed transformation. This watershed is performed by means of geodesic skeletons of influence (SKIZ) [1]. The simplest way to achieve it consists in processing the successive gray levels of the image E. Let us denote $Z_i(f)$, the section of a function f at level i:

$$Z_i(f) = \{x \in E : f(x) \leq i\}$$

Given two sets X and Y, with $X \subset Y$, the geodesic SKIZ of X inside Y is denoted by SKIZ(X;Y). This set is made of the zones of influence in Y of the various connected components of X. A zone of influence of a connected component contains all the points of Y at a lower geodesic distance from that connected component than from any other component. The watershed transformation $WTS_M(f)$ of the function f controlled by a marker set M [5], can then be obtained by iterating the following formula for all the sections $Z_i(f)$:

$$W_{i+1}(f) = SKIZ(W_i(f); Z_{i+1}(f) \cup M)$$

with:

$$W_{-1}(f) = M$$

At the end of the procedure, when the highest gray level n is processed, we have:

$$WTS_M(f) = W_n(f)$$

When the watershed is not marker-controlled, the algorithm remains the same with M=m(f), minima of f.

Despite the fact that the hardware processor designed for this project is equipped with 8 PIMM1 chips, it remains too slow for achieving a real-time watershed. A possible solution for increasing the computation speed leads in the reduction of the gray levels. This reduction is made by means of an anamorphosis. An anamorphosis ϕ is a monotonous increasing mapping of f:

$$f \rightarrow \phi(f)$$

$$\forall x, y : f(x) < f(y) \Leftrightarrow \phi(f(x)) < \phi(f(y))$$

However, if we reduce the number of gray levels in an image, chances are, in the discrete case, that the watershed of $\phi(f)$ is dramatically different from the watershed of f. But, if the watershed is marker-controlled, the difference between $WTS_M(f)$ and $WTS_M(\phi(f))$ is not very important because, in both cases, the watershed lines take place on the highest zones separating the markers. Now, the highest zones of f correspond also to the highest zones of $\phi(f)$. But, in the case of a simple watershed, the minima m(f) will be very different from the minima $m(\phi(f))$. The solution for getting similar watershed transforms consists in performing a marker-controlled watershed of $\phi(f)$ with the minima of f:

$$W_{i+1}(\phi(f)) = SKIZ(W_i(\phi(f); Y_{i+1}))$$

with:

$$Y_{i+1} = Z_{i+1}(\phi(f)) \cup M$$

where M = m(f).

Among the various available anamorphoses, the following one is very efficient on gradient functions:

$$\phi(f) = \log_2(f + 1)$$

This transformation leaves the low values practically unchanged and decreases significantly the high values. An image defined with 256 gray levels is then reduced to eight gray levels. Thus the computation speed is multiplied by 32, with a slight degradation of the watershed transform.

3. The multi-pipeline processor

The multi-pipeline processor is made of two parts. The main pipeline processor includes two boards in pipeline for speeding up background jobs and the second only one board: obstacle detection need less computing power. These processors come with a multi-access image memory board. This board enables data-transfer with the grab/display board, with the intelligent board and also supplies the two pipeline processors with image data.

3.1. The multi-pipeline architecture

The MSM exchanges data with the outside world via two buses. A VME bus permits to receive orders from the supervisor and also to transmit the processing results to the supervisor or to others sub-modules. Image data come from the camera through an acquisition board under Maxbus® video bus format. To satisfy the multi-tasking and the real-time requirements we have selected the architecture shown at Figure 5.

As mentioned above, the VME bus has been chosen for inter-module communication and data-exchange. On the other hand, programmable image processors working under iterative algorithms need to be configured very often, that is between each elementary step of the program. As the image size is 256x256 pixels or less during ROI image processing, the configuring phases occur every 3.3 milliseconds or less. So, not to clutter up the VME bus, inter-module communication and data-exchange on the one hand and morphological processor programmation on the other hand must be performed through separate buses. This particularity is an essential requisite in the choice of the intelligent board, a Radstone® RS41 board. It has a VME bus and a parallel sub system bus (VSB) with multiplexed address/data lines. Moreover, the VSB bus has block transfer capabilities which enable very fast image and bit data transfer to and from the memory board. The intelligent board manage several simultaneous processes, data-exchange with other sub-modules and management of the two separate image processing tasks on the two pipeline processors, by means of WxVorks®, a multi-tasking operating system.

As the road/lane segmentation algorithm is more time consuming than the obstacle detection algorithm, the power of the processor dedicated to the former function is doubled by pipelining two morphological processor boards. These two boards constitute the main pipeline processor. The second pipeline processor includes only one morphological processor board.

Each pipeline processor is supplied with image data by the memory board through two separate 60MB/s very high-speed video buses specially designed for morphological transformations. Moreover the memory board can exchange data with the acquisition/visualization board through the Maxbus video bus. The Intelligent board can read intermediate level data - such as a list of coordinates - and transmit them to the supervisor through the VSB bus.

With this architecture all the pre-cited requirements are satisfied. The MSM communicates with the outside world through the VME bus. The VSB bus, used for processor configuration, allows not to clutter up the VME bus. The use of the two pipeline processors and the fact that the VxWorks multi-tasking operating system runs on the intelligent board allow true multi-tasking image processing capabilities.

3.2. The morphological processor

As mentioned before, the most time consuming morphological transformation used in road/lane segmentation and obstacle detection algorithms is the watershed transformation. However, it is not the unique morphological operator used in this application. We can also mention minima detection, greytone and binary filters, particle reconstruction, greytone and binary geodesic transformations. Moreover, non-morphological operators such as point to point transformations, thresholds, are also used. These later operators must be implemented in a very efficient way: the ratio between speed and surface must be very high to obtain sufficient performances without having a too bulky structure.

Fortunately, to implement all these operators in a highly integrated way there exists a dedicated ASIC designed by the CMM, PIMM1 [9,10,12]. Figure 6 briefly shows its internal architecture.

PIMM1 works either on two-dimensional binary or 8-bit greytone images. The morphological transformations are supported, in binary mode, by eight 3x3 neighborhood processing units which can be organized either in pipeline or in parallel. In greytone mode, two 3 x 3 neighborhood processors works in parallel on the same data flow. Besides, some recursive transformations like particle reconstruction and distance function are directly implemented. Associated with external delay lines, PIMM1 may be considered as a pipeline: the image-data read sequentially in direct or reverse video scanning from memory go through PIMM1 at a 20MHz pixel frequency rate. Euclidean and geodesic transformations can also be implemented. Some point to point operations like numerical addition and subtraction, threshold and Boolean operators can be executed in the point processor. The major particularity of this ASIC lies in the possibility to directly pipeline several chips to adapt the number of connected PIMM1's to the required power of the image processor.

A large number of morphological operators can be divided in a series of 3x3 neighborhood operators and each morphological transformation can be built by iterating 3x3 elementary neighborhood operators. So, the greater the number of 3x3 neighborhood operators in pipeline is, the faster the morphological transformations are executed. However, when the number of pipelined 3x3 operators becomes too high, some problems occur for optimizing the pipeline utilization. Aware of this problem, the morphological processor board have been designed as shown at Figure 7.

To increase the computational power on the morphological processor, 4 PIMM1's have been pipelined. So we have either 64 binary neighborhood processors or 4 greytone neighborhood processors or any possible combination: a greytone transformation can be programmed on the first PIMM1 and a threshold plus some binary transformations on the last PIMM1 in the same scanning. In addition, an in-depth analysis of the application software allows to optimize the use of the various stages of the pipeline and to decrease the number of processing steps needed for achieving image treatment.

A look-up table (LUT) has been added to the input of the first PIMM1 for implementing the anamorphosis. This LUT acts on the main image flow. A second image flow is also sent to the pipeline. This second flow allows to execute binary geodesic operations [1,16].

Numerical geodesic transformations are not directly implemented, but it is possible to perform the elementary step of a geodesic dilation (erosion) in PIMM1. So, some delay lines have been added to re-synchronize a geodesic image flow with the main processed image flow. Four steps of geodesic greytone dilation (erosion) can then be executed in one cycle. These transformations are largely used in minima detection algorithms and in some greytone filters.

3.3. The image memory board

The image memory board has to manage numerous accesses. So it has been carefully designed to avoid dramatic decreasing of the performances of the MSM.

As acquisition, display and host transfer requests can occur at any time and in particular during a processing phase, some image memory buffers are allocated for each possible access (see Figure 8).

The memory board contains two identical processor image memories, each one dedicated to a pipeline processor. Each memory contains 16 256x256 9-bit image buffers which can be organized either in greytone images or in binary images. Each bank sends via the morphobus the image flows: the main one and a possible second one for geodesic and point to point transformations. The resulting data coming from the pipeline processor may be rewritten in any of the memory banks; this allows to exchange image data from one bank to the other. This functionality is very useful. Indeed, obstacle detection is performed on the ROI emphasized by the road segmentation algorithm. As the first and the second treatments are performed by different pipeline processors, the possibility to rewrite result flows in any processor image memory allows to transmit data from one processor to the other. Moreover, the results flows may be rewritten into the visualization and/or into the host interface memories.

The display buffer has two banks; this allows to send an image to the acquisition/display board and to write the next result to be displayed into the display buffer simultaneously. The same structure has been adopted for the acquisition frame buffer and for the host interface memory. In addition, the image data coming from the outside world by the maxbus and/or the VSB bus may be directly processed by one of the two pipeline processors.

Finally, the image memory board advises, through interrupt lines, the intelligent board that the processing and/or the acquisition jobs are ended. Thus, the VSB master knows when it can program the memory and the processor boards to begin a new processing step.

Such an architecture allows to perform simultaneously two processor image memory accesses, one acquisition frame memory writing and one visualization buffer writing. Therefore it exactly fits the pre-cited flexibility requirements.

4. Implementation of the watershed transformation and performances

The MSM was designed to speed up the lane/road segmentation and obstacle detection algorithms. In particular, the pipeline architecture of the morphological processor decreases the processing time of the watershed transformation.

The SKIZ used in computing the watershed transform may be implemented by applying a series of elementary binary thickening transformations. Each binary processor of PIMM1 was designed to directly support 3 x 3 thickening operations. In PIMM1, eight binary neighborhood processors can be pipelined: the thickening transformation by the 8 rotations of the structuring element may therefore be performed in one PIMM1. So with eight PIMM1's in pipeline, 64 rotations of the structuring element may be performed in one scanning. The processing time is theoretically divided by 8. As mentioned above, a $\log_2$ anamorphosis reduce the 256 grey levels of an image to 8. The result is slightly deteriorated but is not damaging for our application. The gain in speed is considerable, such a watershed runs in less than 80 milliseconds.

5. Conclusion

We have developed a MSM to implement the algorithms dedicated to road/lane segmentation and obstacle detection. To satisfy the requirements of real-time and multi-tasking image processing, we have designed two separate pipeline processors built around the powerful ASIC PIMM1. This pipeline architecture coupled with a fast watershed algorithm based on anamorphosis leads to robust, efficient and fast segmentation procedures. The multi-pipeline multi-tasking morphological sub-module is able to compute road segmentation and obstacle detection in less than 200 milliseconds. The Prolab2 demonstrator vehicle has been achieved in October 1994. Manu tests and public exhibits have already been performed. Regarding the MSM, experience has proved that its speed is compatible with real-time driving assistance.

6. References

[1] Beucher S. (1990). *Segmentation d'Images et Morphologie Mathématique*, Doctorate Thesis, School of Mines, Paris, June 1990.

[2] Beucher S. (1994). Watershed, Hierarchical Segmentation and Waterfall Algorithm, In: *Proc. of Intern. Symp. on Math. Morph. ISMM'94*, Fontainebleau, France, Sept. 1994.

[3] Beucher S., Bilodeau M., Yu X. (1990). Road Segmentation by Watershed Algorithms, In: *Prometheus / ProArt Workshop on Vision*, Sophia Antipolis, France, April 1990.

[4] Beucher S., Bilodeau M. (1994). Road segmentation and obstacle detection using a fast watershed transformation, In: *Intelligent Vehicles 94*, Paris, France, October 1994.

[5] Beucher S., Meyer F. (1992). The Morphological Approach to Segmentation: The Watershed Transformation. In: *Mathematical Morphology in Image Processing* (E.R. Dougherty, Ed.), Chap. 12, pp. 433-481. Marcel Dekker, Inc., New York.

[6] Beucher S., Yu X., Bilodeau M. (1990) Road Segmentation and Tracking by Mathematical Morphology, In: *Proc. of the 4th Prometheus Workshop*, Compiègne, France, Oct. 1990, pp. 64-75.

[7] Beucher S., Yu X. (1994). Road recognition in complex traffic situations, In: *IFAC TS'94 Symposium*, Tianjin, China, August 1994.

[8] Bilodeau M., Peyrard R. (1992). Multi-pipeline Architecture for Real Time Road Segmentation by Mathematical Morphology, In: *Proc. of the 2rd Prometheus Collision Avoidance Workshop* (N. Clarke, Ed.), pp. 208-214, Nurtingen, RFA.

[9] Peyrard R. (1990). Pimm1: Mathematical morphology integrated processor: Data sheet. Technical report, Centre de Morphologie Mathématique, Ecole Nationale Superieure des Mines de Paris, February 1990.

[10] Peyrard R. (1992). *Conception et mise en oeuvre d'un ASIC de morphologie mathématique à architecture programmable*. Doctorate Thesis, School of Mines, July 1992.

[11] Peyrard R., Gauthier M., Klein JC. (1994). Real-Time Road Segmentation Using a Morphological Multi-Pipeline Processor, In: *Intelligent Vehicles Symposium '94*, Paris, France, Oct. 1994.

[12] Peyrard R., Klein JC. (1991). Speeding up Mathematical Morphology Processes by Using a New ASIC: PIMM1, In: *Proc. Computer Architecture for Machine Perception - CAMP91* (B. Zavidovique, PL. Wendel, Eds.), pp. 439-452, Paris, France.

[13] Yu X., Beucher S., Bilodeau M. (1992). Road Tracking, Lane Segmentation and Obstacle Recognition by Mathematical Morphology, In: *Proc. of Intelligent Vehicles '92 Symposium*, Detroit, USA, 29 June- 1st July 1992, pp. 166-172.

[14] Yu X. (1993). *Vision Dynamique et Morphologie Mathématique*, Doctorate Thesis, School of Mines, Paris, May 1993.

[15] Yu X., Beucher S. (1994). Vehicles Recognition by Video Camera, In: *IFAC TS'94 Symposium*, Tianjin, China, August 1994.

[16] Serra J. (1982). *Image Analysis and Mathematical Morphology*. Academic Press, New York, 1982.

Figure 1 - *Connection of the ground marking by a temporal filter: Original image (left), filtered image (right).*

Figure 2 - *Watershed of the gradient modulus of the filtered image.*

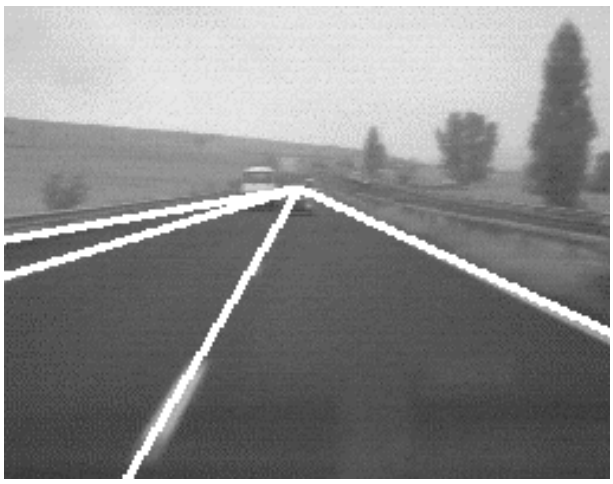

Figure 3 - *Segmentation in tracking mode, using a road model (rear view).*

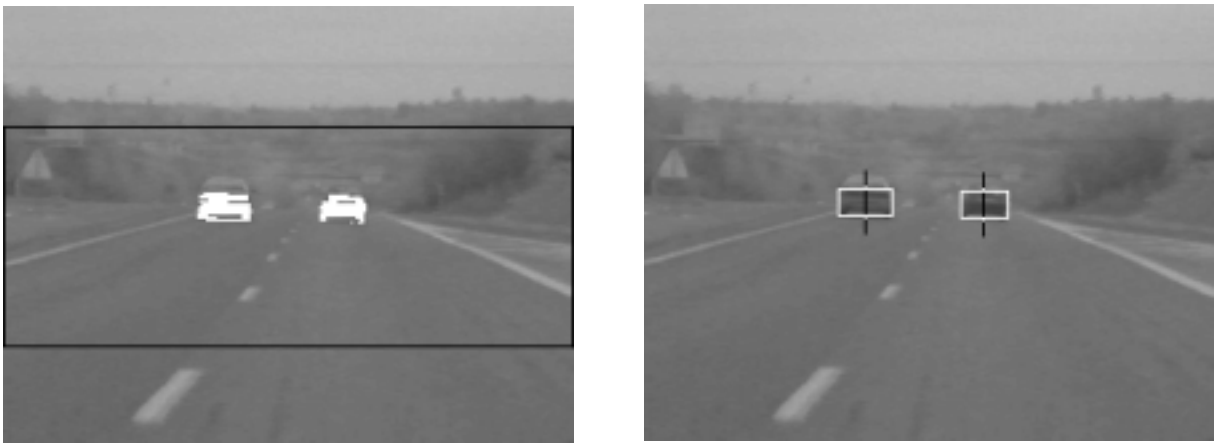

Figure 4 - *Detection of the dark zones in front of vehicles (left) and corresponding bounding boxes (right).*

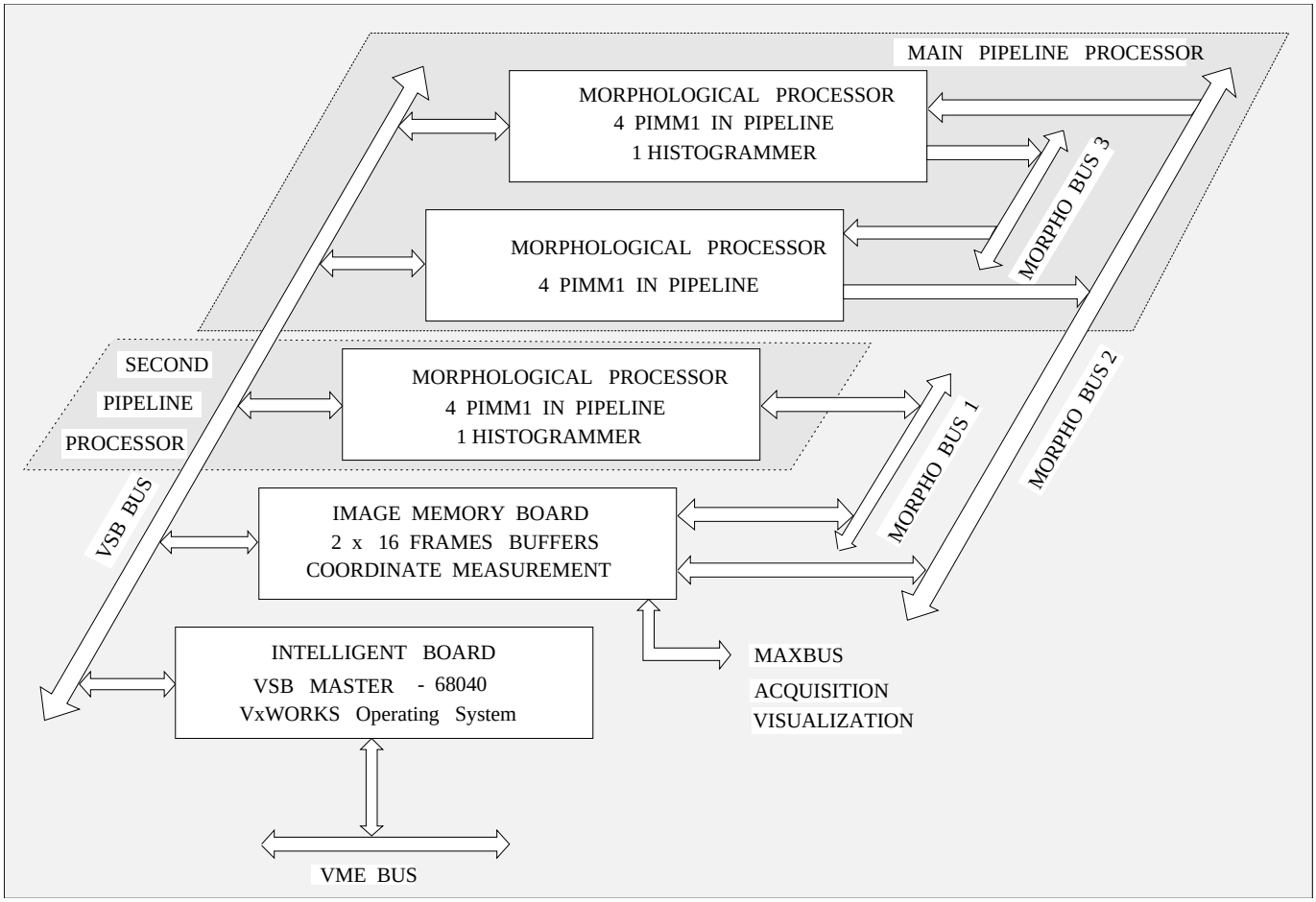

Figure 5 - *Architecture of the Morphological Sub-Module (MSM).*

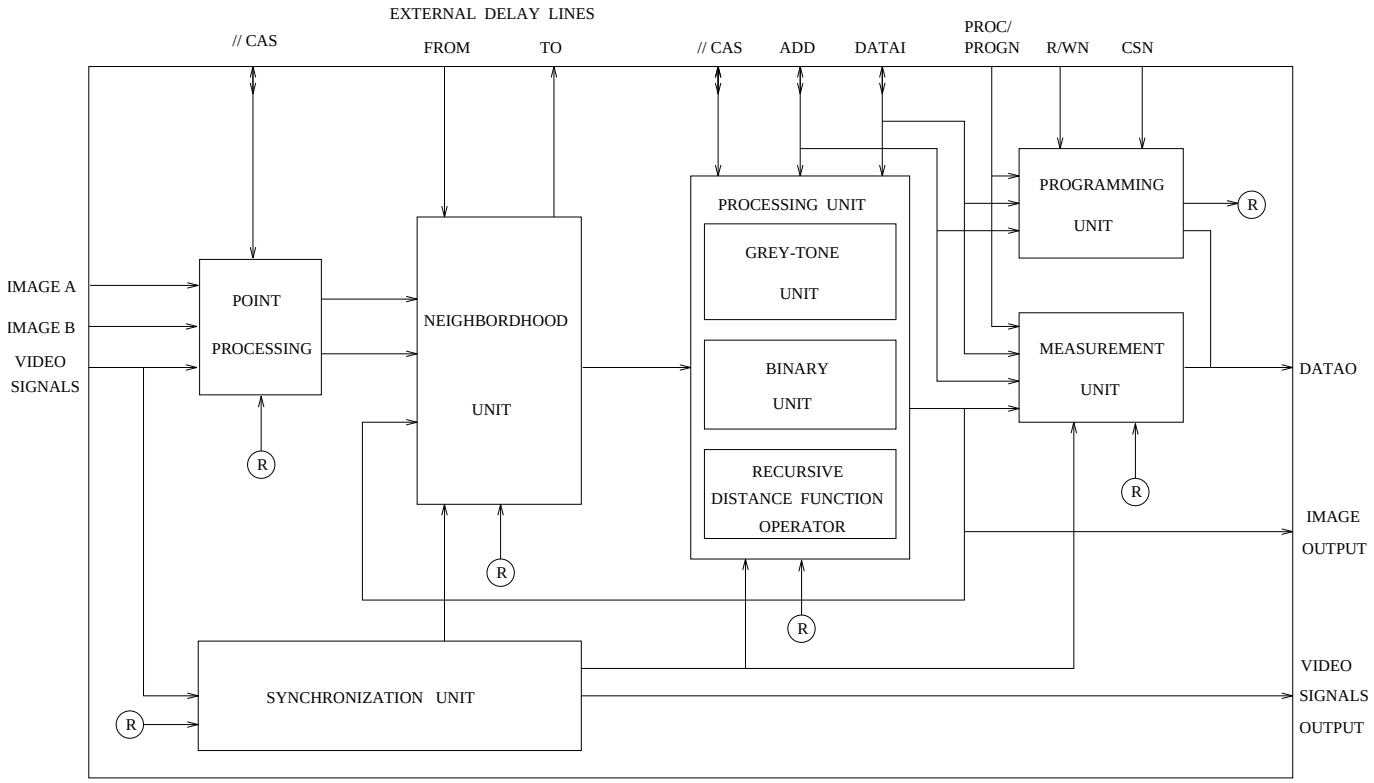

Figure 6 - *PIMM1 block diagram.*

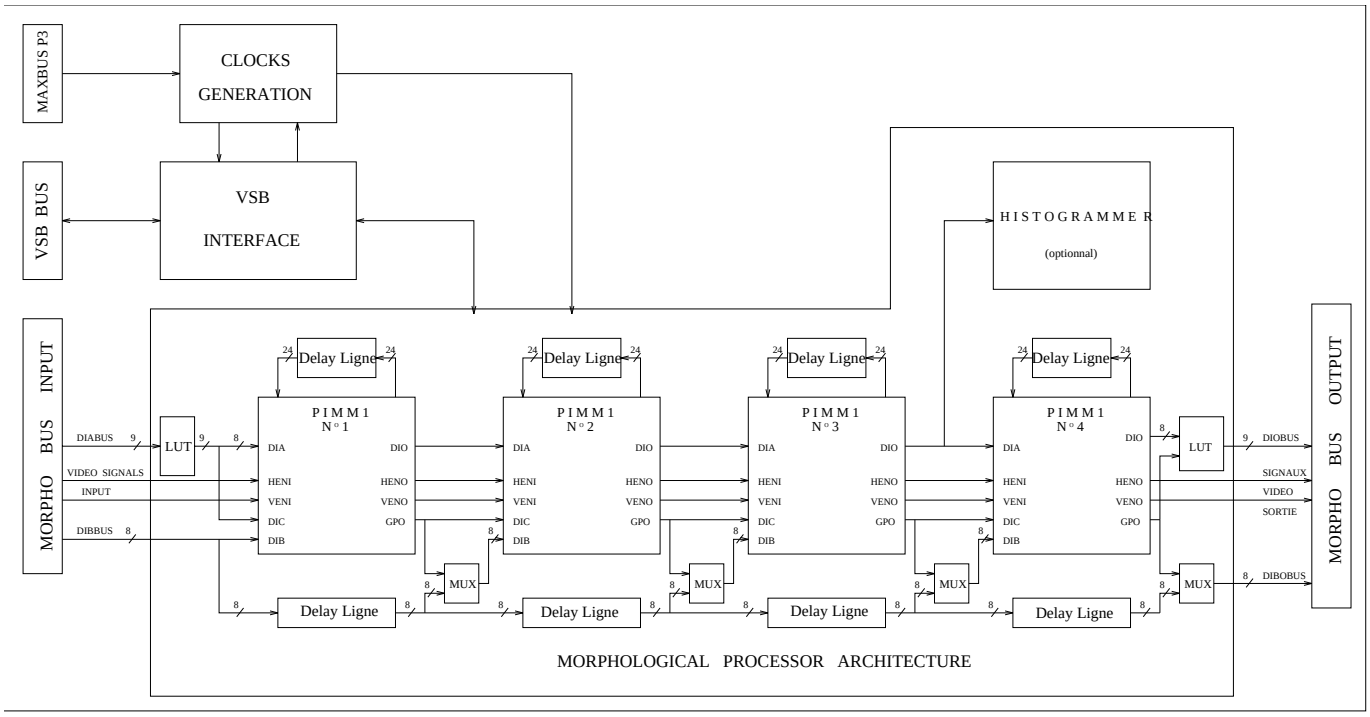

Figure 7 - *Pipeline architecture of the morphological board.*

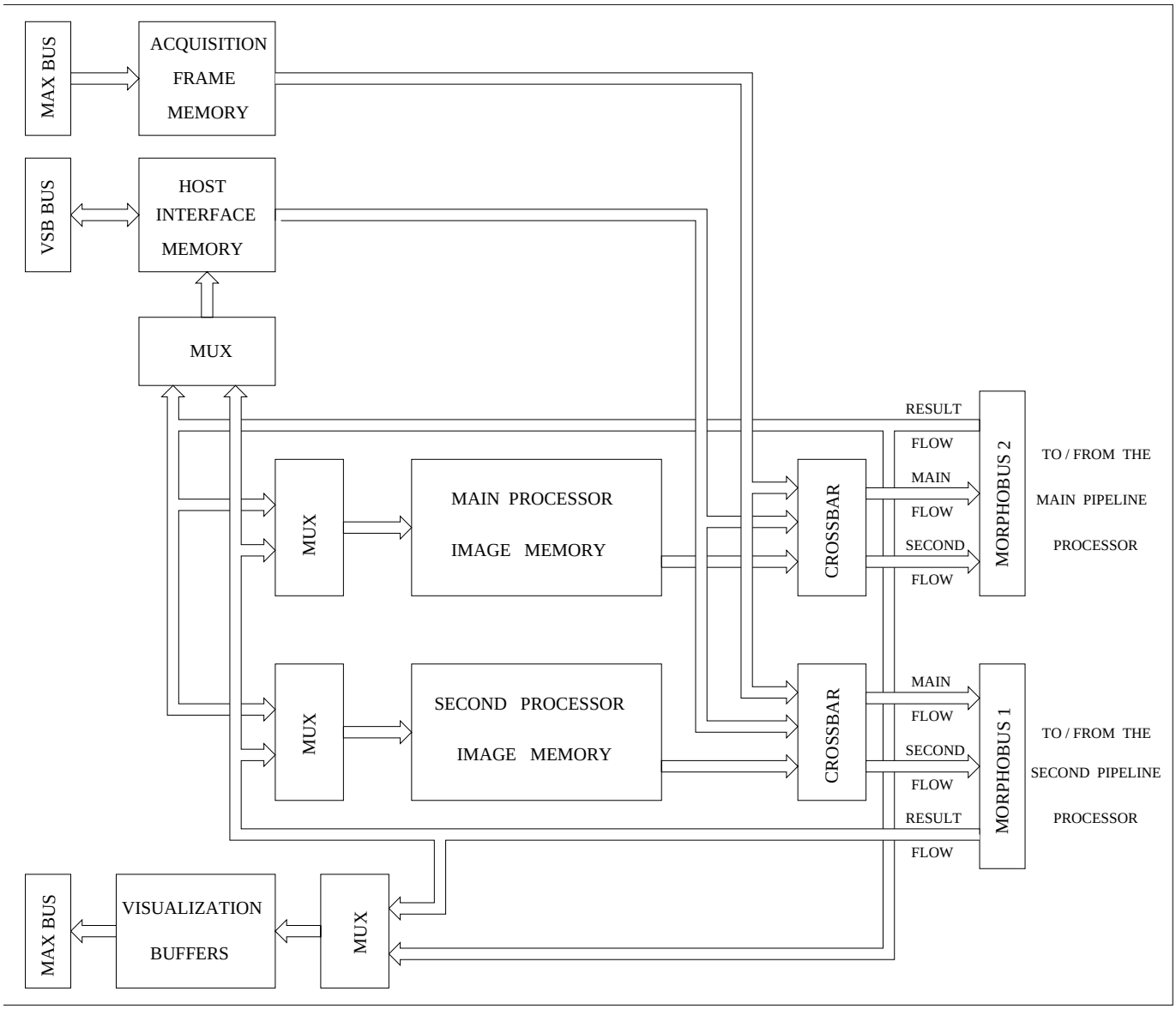

Figure 8 - *Structure of the memory board.*