

# An Evaluation of Monocular Image Stabilization Algorithms for Automotive Applications

Luca Bombini, Pietro Cerri, Paolo Grisleri, Simone Scaffardi, Paolo Zani  
 VisLab - Dipartimento di Ingegneria dell'Informazione  
 Università di Parma  
 Parma, I-43100, Italy  
 {bombini, cerri, grisleri, scaffardi, zani}@ce.unipr.it

**Abstract**—The performance of many computer vision applications, especially in the automotive field, can be greatly increased if camera oscillations induced by movements of the acquisition devices are corrected by a stabilization system. An effective stabilizer should cope with different oscillation frequencies and amplitude intervals, and work in a wide range of environments (such as urban, extra-urban or even unstructured ones). In this paper we will analyze three different approaches, based on signature, feature, and correlation tracking respectively, that have been devised to face these problems.

Keywords: Video stabilization, Vision, Automotive applications.

## I. INTRODUCTION

Camera oscillations can be a serious problem in many computer vision applications, and this fact becomes particularly evident in the automotive field, where an acquisition system installed on a vehicle suffers from its movements. For instance, when driving on urban roads and highways, considerable pitch variations can be noticed, while in off-road and unstructured environments every camera orientation angle can be largely affected by the road roughness. Such kind of movements cause continuous variations of the calibration parameters which many algorithms (like those employed in pedestrian detection [1] [2], obstacle detection [3] or lane detection [4]) use to associate the real 3D world position to the features they extract. Figure 1 plots the error percentage that would affect distance estimations, if the pitch variation angle (with respect to the initial calibration) were neglected. Let  $p$  be the pixel corresponding to the projection of the real-world point  $(d, 0, 0)$  in a camera with an instant pitch of  $\theta' = \theta_{calib} + \Delta\theta$ , and height  $h$  from the ground; if no stabilization is performed, the use of the sole calibration pitch value to compute its distance leads to:

$$d_{estim} = \frac{h}{\tan(\text{atan}(\frac{h}{d}) + \Delta\theta)} \quad (1)$$

and thus to an error of:

$$error = \frac{|d_{estim} - d|}{d} \quad (2)$$

Many stabilization systems have been devised to cope with this problem, and while their goal is the same - i.e. to compensate for camera movements - the approaches used

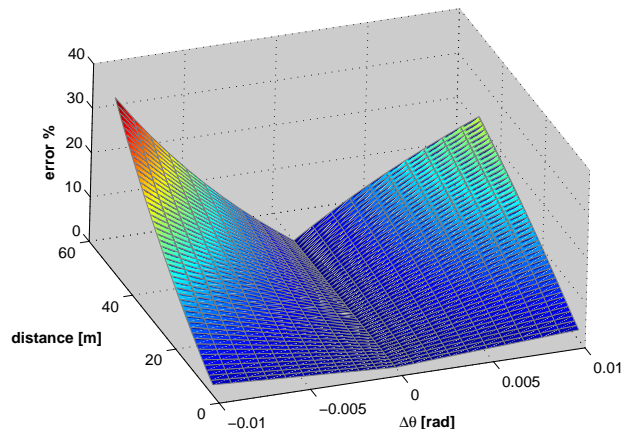


Fig. 1. Distance estimation error for a camera whose pitch has changed by  $\Delta\theta$ ; camera height is 1.8 m. The surface is lacking symmetry because when the image gets shifted up by  $p$  pixels (that is, for negative values of  $\Delta\theta$ ) real world points distance estimations are greatly increased, while the opposite movement makes the camera look farther, leading to smaller variations in the predicted values.

can be substantially different, depending on the application field they are targeted to.

## II. STATE OF THE ART IN VIDEO STABILIZATION

One way to reduce the acquisition system vibrations is to install it on an electromechanical stabilization platform [5]. This approach is quite effective in suppressing high frequency noise, but the performances are generally poor at lower rates, such as in the  $1 - 25$  Hz interval (which unfortunately is the one where most of the noise involved with vehicle driving is concentrated in). Besides this, the device itself is usually quite cumbersome and expensive, so other types of stabilizers are more commonly employed. Another viable solution is to exploit the movement of a set of lenses to achieve the desired stabilization [6]. While there is the clear advantage of having no image quality loss, since this process directly acts on light incidence angle on the camera sensor, still it must be taken into account the fact that broad movements cannot be dealt with, as the correction scope is limited by physical factors; this approach seems

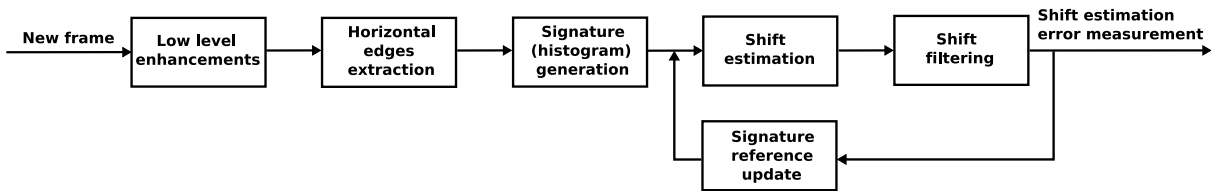


Fig. 2. Signature stabilization algorithm processing flowchart.

therefore more suitable for other kinds of applications, such as in handheld cameras.

Digital image stabilization algorithms do also exist, and have shown to be capable of providing good results in automotive applications. The big advantage with this class of approaches is that no extra machinery is needed besides the acquisition device itself, since all the desired corrections are performed within the image processing hardware, which can either be dedicated or general purpose (in which case the work is done at software level). This usually leads to a more compact and robust design, and possibly to lower costs in large scale production. The resulting system is also more customizable, as its behaviour can be modified to exactly fit the requirements of the application it is used in, while mechanical devices can hardly offer such flexibility. On the other hand these stabilizers have two main drawbacks: they cause information loss (only a limited region of the original image is available for further processing, and pixel interpolation might be used, thus reducing image quality), and introduce a certain delay.

A first example from this category of algorithms is stereovision, which can be exploited to estimate the vehicle pitch at the time of acquisition, as it has been shown in [7]. A pair of stereo images allows to estimate the ground coordinates with respect to the cameras at the time of acquisition, through the V-Disparity image approach, first presented in [8]. Once this information is known, and given the static camera orientation information, pitch detection becomes a trivial task. Stereovision has proven to be effective in both urban and unstructured environments, and represents a viable realtime solution whenever two or more synchronized cameras are available. The main problem arises from system calibration, whose accuracy becomes of paramount importance, especially with large baselines.

Algorithms based on motion vectors have also been proposed [9]. These algorithms try to determine global frame motion starting from local motion vectors, and deciding whether such movement has been produced by undesired oscillations or not. This approach has shown to be fast and effective, but it is quite hard to employ in the automotive field, where the presence of ego-motion leads to poor results.

In urban contexts lane markings can be successfully exploited to recover camera orientation [10]. Such kind of algorithms can produce accurate outputs; nonetheless, it must be noted that they make strong assumptions on environment structure, thus resulting not suitable for use in all those situations where their requirements are not met.

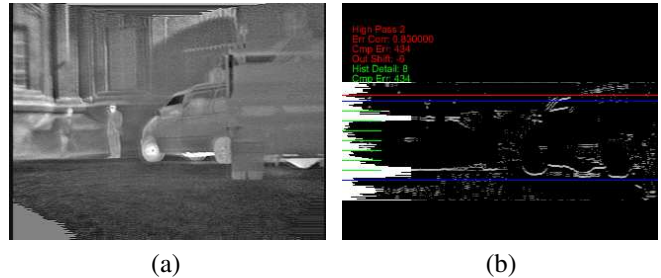


Fig. 3. A sample processing step of the signature stabilization algorithm. (a) Far infrared input image. (b) Output: blue lines indicate the region of interest, while green lines indicate features detection areas; the intensity histogram related to edge image is depicted on the left-hand side, computed values are reported in the top-left hand side.

### III. ALGORITHMS

In this section we will analyze and compare three more algorithms, expressly designed to be employed in automotive applications.

#### A. Signature-based stabilization

A signature, extracted from each acquired image, can be successfully used to estimate undesired image shifts caused by vehicle pitch: in [11] the horizontal edges histogram is used to generate a signature, whose changes are tracked over time, as shown in figure 2. This algorithm is able to correct shifts in the current acquired image up to 10% of its height, without introducing any frame delay in the processing pipeline; its reduced execution time makes it suitable for real time video applications. In figure 3 is presented a sample frame and the derived signature.

While this algorithm has proven to be quite effective in many situations, still it must be observed that objects with a significant horizontal structure might induce it to apply unnecessary corrections, since they produce evident peaks inside the signature; slope changes are also hard to manage, as the signature extracted in such cases drifts in a preferred direction, making correct shift estimation impossible: this is an intrinsic weakness of vision-based stabilizers, which cannot know if such kind of oscillations should be ignored, unless inertial data is provided by other means. Best performances are obtained with input images grabbed from far infrared (FIR) cameras, as they have higher contrast if compared to visible ones, making the signature more easily trackable across frames.

### B. Features-based stabilization

The tracking of image features along the frames of a video sequence can also be employed for the detection of acquisition system movements [12]. A feature is a small region of the image characterized by some desired property, such as a high standard deviation in the spatial intensity profile, or a zero-crossing in the intensity function laplacian. Features have also been identified as lines, corners [13] or even through a wavelet-domain transform [14]. In [15] features are defined as image tiles which provide the best results with a given tracking algorithm; this approach is optimal by definition, and even if common sense would suggest that the former criteria should be preferred as they seem to be able to give reasonably good results, still the latter is more appropriate, as it makes no implicit assumption of independence between features localization and tracking.

Exploiting the features selection algorithm presented in [15] for automotive applications can lead to fairly good results, provided that some specific issues related to this kind of use are correctly addressed. First, the cross-correlation between two frames out of a sequence of images acquired from a moving vehicle can be low, because of the speeds involved with driving: this means that an image cannot be taken as a reference for many frames. Another problem is that, when grabbing frames from a camera placed onto a vehicle, depth information is not neglectable, making the feature tracking task harder to perform than it would have been if the whole scene could be approximated to a planar surface: the points of two successive frames cannot be related through a simple rotoranslation. An example of how the described issues impact on features selection and tracking can be seen in figure 4.

Processing time must also be kept under control, given the real-time nature of this class of systems. Following these considerations, the stabilization algorithm looks for features inside each image, avoiding an explicit tracking; instead it assumes that in two successive frames almost the same details are selected as features, and that they appear in similar columns. Figure 5 shows that features pairing is performed considering the  $x$ -axis position of each feature, so that a correspondence is considered to be found when for two of them (one from the frame taken at time  $T - 1$  and one at time  $T$ ) the abscissa difference is under a certain threshold.

This process creates a set of paired features, which is then used to estimate the current pitch, through the analysis of the vertical offset in each features pair. In a typical scenario many different values may appear, but usually one tends to prevail, thus the mode is used to determine the global frame shift, as in (3), where  $S$  is the set of all the admissible shifts, and  $frequency(s)$  is the number of paired features with a vertical offset  $s$ .

$$shift = \bar{s} \in S \mid f = \max_{s \in S}(frequency(s)) \quad (3)$$

In some situations software stabilizers output can be incorrect, with effects spanning across several subsequent frames; a very simple way to cope with this problem is to

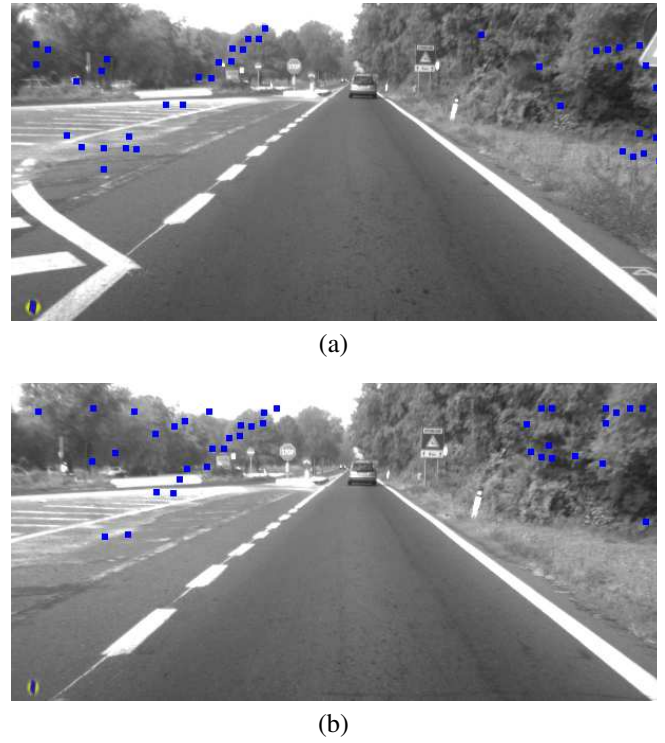


Fig. 4. Features tracking in a extraurban environment. (a) First acquired image. Selected features are marked with small blue squares (b) The following image. The set of selected features is quite different, due to the small amount of correctly identifiable details.

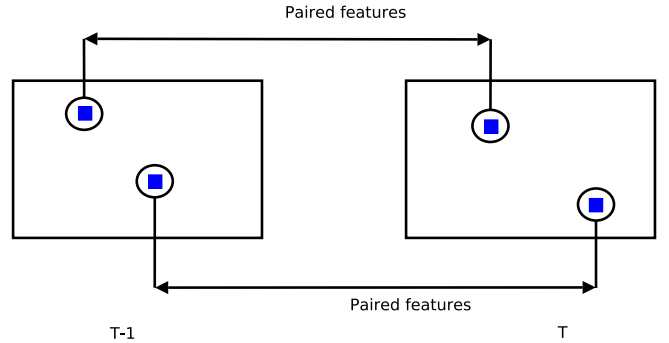


Fig. 5. Features pairing. The frames are acquired at times  $T - 1$  and  $T$ , then their details selected as features are paired according to their  $x$ -axis position

insert in the processing pipeline a PID controller, that brings the shift back to zero when its running sum becomes greater than a certain threshold (thus avoiding absolute value drifts), or when the estimated image offset remains constant for too many frames, as sometimes happens after a sharp bump, when the algorithm might stabilize its corrective action on a small - but still non-zero - output shift.

The complete features-based stabilization algorithm processing flowchart is the one depicted in figure 6.

Urban environments are the ideal field of application for the described algorithm, as they are usually rich of easily-trackable details, while on extra-urban roads the performance is lower, given the presence of large plain-textured areas;

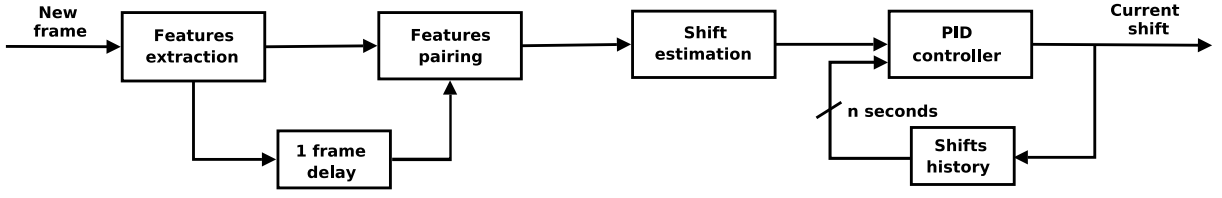


Fig. 6. Features-based stabilization algorithm processing flowchart.

sharp turns can also be hard to manage, since the acquired images content changes very quickly. Features-based stabilization has also another disadvantage: it is computationally expensive, with a processing time that heavily depends on the number of tracked features. While some fast implementations of features-based algorithms exist [14], in this case the analysis of a lower number of details seems more suitable to obtain a faster execution, albeit at the cost of slightly deteriorated results: a more reliable stabilization still needs higher time consumption. A good compromise for images with a resolution ranging from  $750 \times 400$  up to  $640 \times 480$  pixels in both urban and extra-urban contexts is of about 10 features per image.

### C. Difference-based stabilization

The inherent complexity of features-based stabilization, with a performance bound to the environment characteristics and a high processing time, makes it clear that a simpler and more robust approach is necessary to achieve better results.

Starting from the usual consideration that the content of two successive frames doesn't normally change too abruptly, it follows that even in presence of significant camera oscillations it is still possible to find, for some vertical shift, a high correlation between the acquired images, as it has been shown in [13]. A good correlation criterion is to compute the sum of absolute values of the differences among the corresponding pixels between a reference image (named  $r$ ), taken at time  $T-1$ , and the current one (named  $c$ ), taken at time  $T$ , shifted vertically by  $s$  pixels; this procedure is summarized in (4), where  $d(s)$  denotes the difference at a given shift,  $h_{min} = \max(s, 0)$ ,  $h_{max} = \min(\text{height}, \text{height} - s)$ , and  $n = h_{max} - h_{min}$ :

$$d(s) = \frac{1}{n} \sum_{y=h_{min}}^{h_{max}} \sum_{x=0}^{\text{width}} |r[x][y-s] - c[x][y]| \quad (4)$$

The selected shift will be the one which minimizes the difference between the two images, thus producing the best possible match: this leads to (5), where  $S$  denotes the set of all admissible shift values:

$$\text{shift} = \bar{s} \in S \mid \bar{d} = \min_{s \in S} (d(s)) \quad (5)$$

The described operations are illustrated in figure 7.

It is possible to effectively compute the desired shift even applying some subsampling to the input images (for example processing one pixel every four), obtaining fast

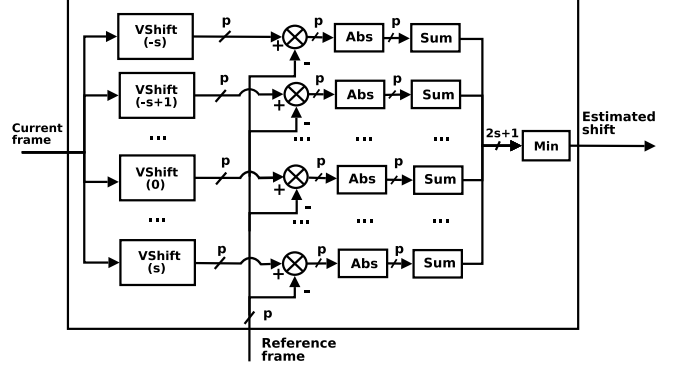


Fig. 7. Details on how the shift estimation is performed;  $p$  is the number of analyzed pixels, while  $s$  is the maximum offset applied.

execution times, which make this algorithm suitable for real-time applications; results stability is granted, as it happens for the features-based stabilizer, by a PID controller.

An horizontal Sobel filter can be inserted at the very beginning of the processing pipeline to enhance the results in urban contexts, which are rich of details; this step is nevertheless optional, and can even cause some information loss in certain situations, such as the one depicted in figure 8, where the frame has been grabbed at night: in these conditions the preprocessing step is useful to remove the visual artifacts introduced by lens flare, but it also completely eliminates the other car beams, that would have been an easy to match region.

A big advantage of the difference-based stabilizer over the features-based one is that the type of environment does not affect its performance, both in terms of results and processing time; moreover, no particular thresholds are needed to fine-tune its behaviour, and the only conditions that must be met in order to have it working correctly are:

- **good frames correlation:** sharp turns or crossing vehicles can seriously impair the stabilization outcomes, since they produce images with loose vertical correlation; these events can however be spotted out, as they are usually associated with good correlation for some horizontal shift. Moreover, the processing overhead of this sanity check is neglectable, since it is effective even when performed on very few pixel rows.
- **a few objects inside the field of view:** this ensures that enough information is available to perform the matching; large, textureless areas - such can be wide extraurban roads where no lane markings are present -

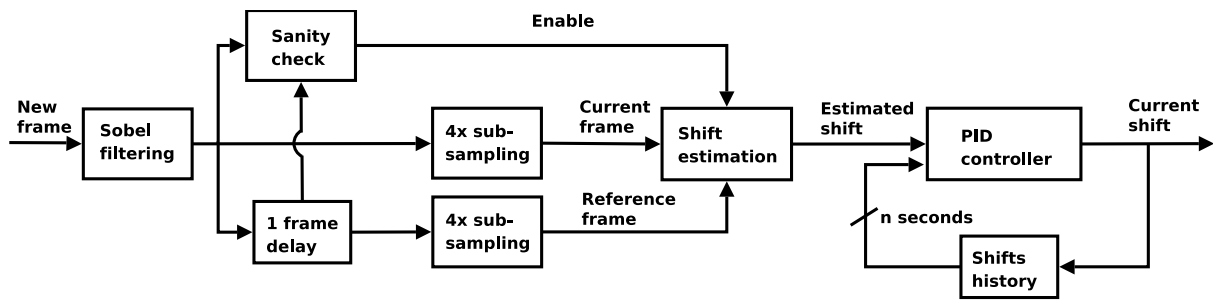
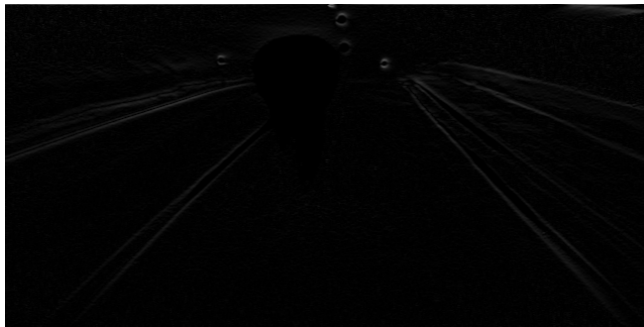


Fig. 9. Difference-based stabilization algorithm processing flowchart.



(a)



(b)

Fig. 8. Night stabilization. (a) An image acquired at night on an extraurban road. (b) The same image, after the application of a Sobel filter. Easily recognizable details, such as the car beams, have been removed by this kind of preprocessing.

can prevent the algorithm from working correctly.

This minimal set of requirements allows a wide range of applications, with both visible and infrared cameras; the complete processing flowchart for this algorithm is presented in figure 9.

#### IV. RESULTS AND CONCLUSIONS

The presented algorithms have been tested on sequences acquired both in urban and extra-urban environments, using three cameras simultaneously, in order to build a common evaluation set. The cameras were FIR, NIR and visible ones, with resolutions of  $320 \times 240$ ,  $768 \times 288$  and  $640 \times 480$  pixels respectively. Figure 10 contains a sample frame for each sequence, grabbed by the visible camera.



(a)



(b)



(c)

Fig. 10. Stabilization test sequences samples. The images have been acquired by a visible camera on (a) The close proximity of a bumper on a urban road. (b) A gravel road (c) A country road. Note the sign used in all the sequences as a reference for manual annotation

During the acquisitions, cameras have been set up with optical axes laying all on the same plane, parallel to the ground. A sign, mounted on a pole and aligned with the optical axes of all the cameras, has been used as a reference for performance evaluation: a perfect stabilization would keep it on the central row of the image, so the gap between its actual position and the ideal one has been manually determined for each frame. The results are presented in table I, where each cell contains error mean and variance for the given combination of camera and algorithm.

TABLE I  
STABILIZATION ALGORITHMS PERFORMANCE

Camera	Algorithm			
	None $\mu, \sigma^2$	Histograms $\mu, \sigma^2$	Features $\mu, \sigma^2$	Difference $\mu, \sigma^2$
Urban Road				
FIR	-0.012, 75.91	-1.457, 26.72	0.931, 111.48	-0.815, 55.56
NIR	-5.337, 181.37	-8.723, 151.98	-8.224, 222.03	-4.733, 165.60
Visible	-2.608, 28.40	-5.026, 20.55	-3.487, 36.09	-6.173, 10.51
Gravel Road				
FIR	3.997, 43.31	-1.342, 30.84	2.838, 22.70	-0.245, 31.19
NIR	9.907, 92.08	1.604, 57.30	8.645, 101.37	4.040, 83.18
Visible	2.885, 12.66	1.474, 7.23	1.368, 21.32	1.493, 7.78
Country Road				
FIR	5.110, 57.36	6.530, 41.88	4.768, 62.96	4.453, 46.82
NIR	18.117, 149.99	19.078, 118.14	6.216, 222.00	15.513, 139.26
Visible	0.287, 10.28	-0.642, 5.62	4.967, 56.03	-0.693, 8.33

The signature-based algorithm and the features-based one have shown to provide the best results when employed to face specific problems, such as the stabilization of images in urban contexts; while the former can more easily manage frames grabbed with a FIR camera, the latter is more suitable to compensate the undesired oscillations in visible and NIR images. The difference-based stabilizer appears to be much more versatile, as it can handle both urban and extra-urban environments without significant performance variations. Execution times for all the described algorithms have been compared on a commercial Pentium 4 2.8 GHz machine, equipped with 1 Gb RAM; the results are summarized in table II.

TABLE II  
EXECUTION TIMES FOR THE DESCRIBED ALGORITHMS  
ON A PENTIUM 4 2.8 GHZ MACHINE

Algorithm	Execution time [ms]		
	Min	Avg	Max
Signature	19	20	23
Features	28	30	39
Difference	6	6	6

The most time-consuming algorithm is clearly the features-based one, which is also the less suitable for real time tasks, since an upper bound to its execution time cannot be easily determined; signature-based stabilization worst case execution time can instead be determined, albeit it can be quite different from the average one. Difference-based stabilization seems to be the best solution for real-time applications, since its running time is constant across

frames, making it a good choice when it comes to deal with the constraints of the automotive field.

#### ACKNOWLEDGMENTS

The authors gratefully thank Alberto Broggi for his support and review of this project.

#### REFERENCES

- [1] M. Bertozzi, A. Broggi, P. Grisleri, T. Graf, M. Meinecke, *Pedestrian detection in infrared images*, Intelligent Vehicles Symposium, 2003. Proceedings IEEE, 9-11 June 2003, Pages: 662 - 667.
- [2] M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, A. Tibaldi, *Shape-based pedestrian detection and localization*, Intelligent Transportation Systems, 2003. Proceedings IEEE, Volume: 1, Pages: 328 - 333.
- [3] S. Nedeveschi, R. Danescu, R. Schmidt, T. Graf, *High accuracy stereovision system for far distance obstacle detection*, Intelligent Vehicles Symposium, 2004. Proceedings IEEE, 14-17 June 2004.
- [4] M. Bertozzi, A. Broggi *GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection*, Image Processing, IEEE Transactions on, Volume: 7, Number: 1, January 1998, Pages: 62 - 81.
- [5] J. Schiehlen, E.D. Dickmanns, *Design and control of a camera platform for machine vision*, International Conference on Intelligent Robots and Systems, Proceedings IEEE, Volume: 3, 12-16 September 1994, Pages: 2058 - 2063.
- [6] B. Cardani, *Optical image stabilization for digital cameras*, IEEE Control Systems Magazine, Number: 2, Volume: 26, April 2006, Pages: 21 - 22.
- [7] A. Broggi, C. Caraffi, R. I. Fedriga, P. Grisleri, *Obstacle Detection with Stereo Vision for off-road Vehicle Navigation*, Procs. Intl. IEEE Wks. on Machine Vision for Intelligent Vehicles, June 2005, San Diego, USA.
- [8] R. Labayrade and D. Aubert, *A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision*, Intelligent Vehicles Symposium, 2003. Proceedings IEEE, 9-11 June 2003.
- [9] L. K. Ko S.J., S. H. Lee, *Digital image stabilization algorithms based on bit-plane matching*, Consumer Electronics, IEEE Transactions on, Volume: 3, August 1998, Pages: 617 - 622.
- [10] Y.-M. Liang, H.-R. Tyan, S.-L. Chang, H.-Y.M. Liao, S.-W. Chen, *Video stabilization for a camcorder mounted on a moving vehicle*, Vehicular Technology, IEEE Transactions on, Volume: 53, Issue: 6, November, 2004, Pages: 1636 - 1648.
- [11] A. Broggi, P. Grisleri, T. Graf, M. Meinecke, *A software video stabilization system for automotive oriented applications*, Vehicular Technology Conference, May 30 - June 1, 2005, Stockholm, Sweden.
- [12] A. Censi, A. Fusiello, V. Roberto, *Image stabilization by features tracking*, International Conference on Image Analysis and Processing, Proceedings IEEE, 27-29 September 1999, Venice, Italy, Pages: 665 - 667.
- [13] L. Marcenaro, G. Vernazza, C.S. Regazzoni, *Image stabilization algorithms for video-surveillance applications*, International Conference on Image Processing, 2001. Proceedings IEEE, Volume: 1, 7-10 October 2001, Pages: 349 - 352.
- [14] J.P. Derutin, F. Dias, L. Domez, L. Allezard, *SIMD, SMP and MIMD-DM parallel approaches for real-time 2D image stabilization*, International Workshop on Computer Architecture for Machine Perception, 2005, Proceedings IEEE, 4-6 July 2005, Pages: 73 - 80.
- [15] C. Tomasi, T. Kanade, *Detection and tracking of point features*, Carnegie Mellon University, April 1991, Pittsburgh, USA.