# Statistical method for sub-pixel interpolation function estimation

I. Haller, C. Pantilie, T. Mariţa and S. Nedevschi Computer Science Department, Technical University of Cluj-Napoca, Romania

Abstract-Depth accuracy is one of the most important characteristics for sensors used in distance estimation. Stereovision systems employ sub-pixel interpolation to achieve such accuracy. Literature in this domain is usually dedicated to simple window based stereo solutions. There are currently several new stereo algorithms developed to counter pixel level errors, but they neglect sub-pixel results. We propose the use of function fitting to generate interpolation functions optimized for each algorithm type. Dedicated interpolation functions require the mathematical model of the algorithm. In the proposed methodology of generating the interpolation function the explicit model of the stereo algorithm is replaced by modeling the data distribution resulted from a pre-defined input. Several transformations are also proposed to reduce the dimensionality of the fitting data without loosing any information. The most accurate match for the fitting data-set was a sinusoidal function, a novel shape for sub-pixel interpolation. The function shows a significant improvement compared to legacy solutions, by reducing the error magnitude by several factor for both synthetic and real scenarios.

## I. INTRODUCTION

**R**ETRIEVING accurate depth information is critical part of the environment perception for automotive systems. A stereo-camera setup is a passive sensor solution for depth estimation. Such a system uses pixel-level correspondence between two images captured from different viewpoints. But pixel-level accuracy is not enough for long range systems because the pixel shift between the two images is inversely proportional to the distance. Sub-pixel accuracy is required to maintain high accuracy throughout the detection range.

The original taxonomy proposed by Scharstein and Szeliski [1] classifies stereo algorithms into two main groups, local and global methods. The group of local algorithms uses a finite support region around each point to calculate the disparities. The methods are based around the selected matching metric and usually apply some matching aggregation for smoothing. The window aggregation allows a local smoothing of the disparity values. Larger windows reduce the number of mismatches but also reduce the detection rate at object boundaries. The main advantage of local methods is the small computational complexity which allows for real-time implementations [2, 3]. The main disadvantage is that only local information is used at each step. As a result these methods are not able to handle featureless regions or repetitive patterns. The global methods have a very high computational complexity, thus they are not applicable for automotive systems.

In 2005 Hirschmüller proposed the Semi-global matching

(SGM) [4] stereo algorithm as an alternative to existing solutions which achieves high quality results while maintaining a reduced execution time. This algorithm cannot be classified using the original taxonomy, thus a new group was created, the group of semi-global algorithms. The method performs multiple 1D energy optimizations on the image. The different 1D paths run at different angles to approximate a 2D optimization. By using multiple paths instead of a single one, it can avoid a streaky behavior common with previous algorithms such as dynamic programming or scan-line optimizations. The energy optimization is based on a correlation-cost and a smoothness constraint. The smoothness is enforced by two components, a small penalty, P1, used for small disparity differences and a larger penalty, P2, used for disparity discontinuities. The larger penalty is adaptive and based on intensity changes to help with object borders. The form of the energy function is:

$$E(D) = \sum_{p} \begin{pmatrix} C(p, D_{p}) + \sum_{q \in N_{p}} PI * T[|D_{p} - D_{q}| = 1] + \\ \sum_{q \in N_{p}} P2 * T[|D_{p} - D_{q}| > 1] \end{pmatrix},$$

where D is the set of disparities, C is the cost function and Np is the neighborhood of the point p in all directions. The function T turns the values true and false into 1 and respectively 0. Dp and Dq represent the selected disparities in the points p and q. The Middlebury benchmark [5] shows the results achieved using this. The algorithm consistently achieves results similar to the computationally most expensive methods while clearly differentiating itself from other real-time solutions. Several real-time implementations were also proposed for smaller resolution images. These results show that the method represents a good compromise between speed and accuracy for real-time systems such as automotive applications.

Generally stereo algorithms use a simple parabola interpolation [2, 4]. The method uses the smallest matching value and its neighbors to interpolate a parabola around the three points [6, 7]. The location of the minimum point for this parabola will represent the sub-pixel shift. This solution is mathematically accurate if the matching function can be modeled at least locally as a  $2^{nd}$  degree polynomial. However in 2001 Shimitzu and Okutomi [8] have highlighted that this solution presents a serious issue for the simple window based stereo algorithm, namely the pixel-locking effect where given sub-pixel ranges are favored and large errors can accumulate.

Another solution proposed for sub-pixel interpolation is the use of a linear function [6, 7]. The linearity is motivated for simple stereo algorithms which are based on aggregation. The symmetric V interpolation proposed for the Tyzx DeepSea development system is one such solution [3]. This system shows high accuracy thanks to the synergy between the stereo algorithm and the sub-pixel interpolation function. Figure 1 shows the difference in shape for the two basic solutions, the parabola and the symmetric V.



Fig. 1. Parabola interpolation versus symmetric V

In this paper we present a new methodology for estimating sub-pixel interpolation functions for stereo pipelines. It allows the generation of specific functions for the stereo algorithms, depending on the real distribution of the matching costs. The second section presents a brief overview of existing solutions to the sub-pixel interpolation problem. The third section presents the stereo algorithm used for this paper. This is required because we believe that the sub-pixel performance is dependant on the selected algorithm. The description of our methodology and the resulting interpolation function is presented in the fourth section. The fifth section contains the evaluations which were performed using several functions on different benchmarks. The last part of the paper contains our conclusions.

#### II. RELATED WORK

#### A. Sub-pixel Estimation

The work presented by Shimitzu and Okutomi [8] handles the problem of pixel-locking by modeling the error and applying a correction through the use of the model. They observed that the error is symmetric and could be cancelled through the use of shifted images. The shifted images will have the error function inverted compared to the simple matching. Although this solution proved to be quite effective, its main disadvantage is that the stereo matching has to be performed 3 times resulting in a significant waste of computing resources.

## B. Stereo algorithm

Modern stereo methods such as the Semi-Global method [4] use multiple non-linear transforms due to which estimating a perfect mathematical model for the sub-pixel interpolation is almost impossible. Examples of such transformations are the census transform and also global and semi-global optimizations. The distribution of the matching values also varies between the solutions and as such it is important to mention the stereo algorithm for which we estimate the interpolation function.

The stereo algorithm selected for this paper is a variation

of the basic Semi-Global method [9]. These modifications concern both the running time and the sub-pixel accuracy. To reduce the running time our configuration uses only 4 optimization directions. The original description [4] specifies that the recommended number of directions is at least 8 to achieve quality, but our tests show that the difference is insignificant for automotive applications. Our system is optimized for automotive scenes where the object surfaces are usually tilted around the image axis. Consequently diagonal directions introduce no extra information. Besides reducing the computational task, this optimization has the advantage of simplifying memory accesses.

We also observed an issue with the original algorithm concerning sub-pixel accuracy. The P1 component affects the matching values used in sub-pixel interpolation. The values at the positions -1 and +1 may be shifted with the constant P1. As a result some of the sub-pixel values are corrupted and point scatter is increased. We proposed the elimination of this component from the equation. The new

equation is: 
$$E(D) = \sum_{p} \left( C(p, D_p) + \sum_{q \in N_p} P2 * T[D_p \neq D_q] \right).$$

For the correlation metric our solution uses the Census transform, a novelty for the Semi-Global method. This metric has the main advantage of being independent of luminosity and contrast differences between cameras. Other papers [10] evaluated the different metrics and the Census transform was consistently between the best solutions especially in the presence of radiometric errors. These features are important for an automotive system where the precise calibration of cameras is difficult. The original metrics proposed for the Semi-Global method were shown to be not effective in such systems. Another solution [11] proposed uses ZSAD, but in our tests the Census based solution presented a reduced number of errors. Figure 2 presents a comparison of the two solutions on a typical scenario.



Fig. 2. Intersection scene. Comparison of different solutions, left is SGM+ZSAD, and right is our solution using SGM + Census

#### III. SUB-PIXEL FUNCTION ESTIMATION

## A. Overview

In this paper we propose the use of function fitting in order estimate a new sub-pixel interpolation function with reduced errors. Instead of modeling the sub-pixel errors, our solution works with the distribution of the input values. For our model we choose to preserve the argument list from the parabola method to preserve simplicity. The formula for the sub-pixel disparity is:

$$d_{Final} = d + f\left(m_{d-1}, m_d, m_{d+1}\right)$$

where d is the integer disparity and m is the matching cost for the different disparity steps.

#### B. Data generation and transformations

The input data is generated using a rendered 3D scene of a vertical surface. The surface is textured with a non-repetitive pattern to reduce the stereo uncertainty. The stereo system is chosen to have similar parameters as a real system with a baseline of 44cm and a focal length of 6mm. The imaging resolution is 512x383. The position of the plane is set to distances corresponding to disparity values ranging from 3.5 to 4.5 pixels using a step of 0.05. This allows us to evaluate the behavior of the 3D points around the integer disparity 4. Although this value is arbitrarily selected, the sub-pixel interpolation is independent of the integer value. The value corresponds to a range beyond 40 meters and thus the errors are also highlighted in the metric space. The image on figure 3 is an example from the set. Even though it is a synthetic image we tried to simulate the real imaging conditions to get results as close as possible to reality.



Fig. 3. Example image (right camera, distance is 62.17m)

The rendering results in 21 pairs of left-right images used for the stereo vision algorithm. For the sub-pixel interpolation function we need to log for each point the three matching values used by the function. To reduce the dimensionality of the problem, we consider only the relative position of the 3 points. These can be described using only the following two parameters  $leftDif = m_{d-1} - m_d$  and  $rightDif = m_{d+1} - m_d$ . This allows us to model the interpolation function as a 3D surface while maintaining the shape of the matching function.

The point distribution is presented in figures 4-7. Each point is a tuple of (leftDif, rightDif, expected sub-pixel value). The sub-pixel value is mapped to the range 0-1 starting from 3.5 to 4.5. When looking at the different viewpoints, we can observe a pattern in the XY view. The sub-pixel value shows a direct dependence on the polar angle of the points. The 3D points draw a surface similar to a generalized helicoid. This is further evidence for the observed correlation. Applying a function fitting in 3D is quite difficult, but our observation allows us to reduce the dimensionality of the problem once more.



Fig. 4. Perspective view of point distribution. X and Y are the horizontal axis, while Z is the vertical axis. (X,Y,Z) represents the tuple (leftDif, rightDif, expected sub-pixel value).



Fig. 5. XY view of point distribution.



Fig. 6. XZ view of the point distribution.



Fig. 7. YZ view of the point distribution.

The polar angle of the points in the XY coordinate system is :

polarAngle = arctan(leftDif / rightDif).

In our evaluation we use a simplified variable without the

arctangent function, var = leftDif / rightDif. We take into consideration the symmetricity of the problem around the value 0.5 to eliminate the issue of extremely large values  $(rightDif \rightarrow 0)$ . The distribution will be modeled using the one dimensional function in the following way:

 $subpixel = \begin{cases} function(leftDif / rightDif), leftDif \leq rightDif \\ 1 - function(rightDif / leftDif), leftDif > rightDif \\ function: [0,1] \rightarrow [0,0.5] \end{cases}$ 

## C. Managing the fitting process

Due to the large spread of the point distribution we are not able to perform perfect fitting. Our solution is to use average values for each sub-pixel step. The basis for this solution lies in the fact that each step is represented by a dedicated stereo image pair. For each image the average distance of the points is required to match the expected distance to the surface. The images are also independent from each other, thus we can average the values only across a single image.

The first option is to apply the averaging on the input data before fitting. The advantage of this solution is that the amount of data is significantly reduced. The function can also be visualized helping to establish the main components. These components will be used in the function fitting phase. The averaged points are presented on figure 8 and we can already observe that the final solution should contain a sinusoidal component. The first solution was to use a spline which fits perfectly to the averaged data points, but we observed some residual errors when performing the evaluation with the whole data-set. The evaluation estimates the distance to the surfaces as the average of the individual points. The reason for the difference between fitting and the evaluation is the non-linearity of the function. As consequence this solution is not accurate enough because we cannot estimate the final error at the fitting phase.



Fig. 8. Averaged data points (leftDif/rightDif, expected sub-pixel)

We solved this problem by applying the function fitting to all points and averaging only the results. The target is to minimize the distance of the average values from the expected ones. The problem with this solution is the complexity of the fitting and the lack of visual feed-back. Even though these issues made the process harder, this solution provided the most accurate results as the fitting estimates matched the evaluation results.

## D. Optimization metric

Before we can estimate a function, we need to define the optimization metric which needs to be minimized. The basic metric used for fitting is the sum of errors. This solution may result in non-uniform optimization. A low sum cannot guarantee the lack of error peaks for the final estimate. For a robust system it is much more important to consider the worst case error. In conclusion our methodology is focused on optimizing the maximum error.

#### IV. RESULTS

## A. Resulting function

Our solution uses a model based fitting for simplicity. The model uses component functions, for example polynomials and trigonometric functions. The components were identified using the shape presented in figure 8. The best fit was achieved when a sinusoidal component represented 99% of the final function. A first and a second degree polynomial represented the remaining part. We consider that the polynomial components are too small to take into account because they are within the error margin of the imaging process. The sinusoidal function has the following formula:  $function(x) = \sin(x*\pi/2 - \pi/2)/2 + 0.5$ .

#### B. Evaluation using the generating data-set

Our first evaluation uses the fitting data-set to compare the new function with the parabola and linear interpolations. This test shows the best case results for our solution, but it helps to highlight the problems with the other solutions and provided a basis for comparison.

The figure 9 presents the deviation between the measured distance of the synthetic surface and the expected one. We use the relative deviation to normalize the results with respect to the distance. The first observation is the dependence of the error on the sub-pixel location. This is highlighted mostly in the case of the parabola and linear interpolation. The symmetric behavior of the interpolation functions is also observable in the figures. There are few locations where this behavior is violated. The locations match in all three graph and are caused by the imaging errors.

Both the parabola and linear interpolations show a very strong pixel locking tendency with error rates up to 5%, and 3% respectively. These values are very large for an ideal test case and resulted from the mismatch between the stereo algorithm and the sub-pixel interpolation. Although the two methods have shown good or even excellent results in case of window based stereo solutions, they are not appropriate for complex solutions such as the Semi-Global method.

Our solution shows a significant reduction in the error magnitude. The maximum error is reduced to 0.6% and for 50% of the region it is under 0.1%. At these error rates, the components independent of the sub-pixel interpolation represent a significant portion of the total error. For example

the deviation from symmetricity can reach 0.4% (at 58.81m).



Fig. 9. Error for parabola (blue), linear (red) and sinusoidal (green).

## C. Evaluation using a tilted surface

In order to validate our results concerning accuracy we used another synthetic test case. The main advantage of the synthetic image compared to real ones is the availability of an environmental model. This model allows us to accurately estimate the errors for each solution.

The model used for this test contains a surface tilted at exactly 45 degrees. The surface stretches from 35 meters to 45 meters in depth and from -5 to 5 meters in height.

For the evaluation we decided to average the values across each image row. This solution allows us to eliminate the effect of the point spread from the total error. The result is a set of average Y and Z coordinates belonging to each row. The values are presented on figure 10. The expected Z values are calculated using the measured Y values and the 3D model.



Fig. 10. Estimated synthetic surface at 45 degrees. Axis are in mm.

TABLE I						
DISTANCE ERRORS FOR TILTED SURFACE						
Method	AVERAGE (ABS)	AVERAGE (REL)	Max (ABS)	Max (REL)		
Parabola	693.8 mm	1.75743 %	1317 mm	3.23962 %		
Linear	373.5 mm	0.94699 %	807.5 mm	1.94809 %		
Proposed	97.13 mm	0.24274 %	283.5 mm	0.65971 %		
ABS – Absolute values : REL – Relative values						

ABS – Absolute values ; REL – Relative value

Table 1 presents a detailed report of the error values. The values are comparable with those presented in the previous example. The maximum values for the parabola and linear solution dropped by 35% while our solution remained the same. But even with the drop in efficiency the maximum value is better by a factor of 5 respectively 3. The biggest

difference is in the average error which is improved with a factor of 7 respectively 4. These results prove that our solution is not only optimized for a single test case but can also perform similarly in other scenarios.

#### D. Evaluation using real images

We also performed a brief evaluation of the three solutions using real images. The methodology is similar with the previous evaluation, but instead of a tilted surface we use two vertical surface located at 23.57 meters and 28.53 meters respectively. The surface is textured using the same texture as the one used for synthetic tests. The real distances were measured using a laser rangefinder for maximum accuracy. An example image is presented in figure 11. The evaluation used a region of interest from the patterned surface located near the center of the image.



Fig. 11. Real scene with pattern located in middle at 23.57 meters.

The results are summarized in table 2 and are similar with the synthetic ones. Figure 12 and 13 shows the distances for the different image rows. The figures show that these results are not due to some constant deviation in distance because the errors for legacy solutions change in sign between the two measurements. Interestingly the shape of the surface is preserved between the different solutions. This shape is due to imaging issues. For the test performed at 23.57 meters these errors have an increased magnitude and as a result the difference between the three solutions is lowered, but our solution is still better by several factors.







Fig. 13. Estimated real vertical surface at 23.57 meters. Y axis is distance in millimeters, X axis is image row index in selected region.

TABLE II							
DISTANCE ERRORS FOR REAL IMAGES							
Method	Average (28.53m)	Average (23.57)	Max (28.53m)	MAX (23.57)			
Parabola	724.6 mm	480.6 mm	823.7 mm	558.5 mm			
Linear	403.1 mm	300 mm	518.9 mm	405.4 mm			
Proposed	57.76 mm	86.18 mm	147.8 mm	213.3 mm			

The reduced depth error also improves the performance of environment perception for driver assistance systems. The results of the previous two tests are applicable for object distance estimation improving the accuracy of the environment model. The removal of the pixel locking effect also helps the elevation map algorithm [12] to generate a more refined classified occupancy grid (figure 14). The most visible result is the extension of the side-walk detection range. The road surface is also smoother on the depth map image.



Fig. 14. Urban scene, left column is the classified occupancy grid, right column is the depth map generated by the SGM. Top row uses parabola interpolation, while lower one the new function.

## V. CONCLUSIONS

In conclusion we have show that the problem of sub-pixel interpolation needs new solutions for accuracy critical systems. Many solutions were proposed for the simpler window based stereo algorithms, but they are not appropriate to newer solutions such as the Semi-Global method.

In this paper we propose the use of function fitting to estimate new functions explicitly designed for a given stereo algorithm. Through the use of 3D modeling and rendering a set of synthetic stereo image pairs can be created as a source of input data.

We also proposed a set of transformations to reduce the dimensionality of the problem without loosing information. This process is important because multi-dimensional function fitting can be a very difficult problem. We believe that these transformations can be reused for other stereo solutions as well.

Our solution uses a model based function fitting to determine the interpolation function. We first generated a rough model based on reduced data and used it to estimate the main components of the final model. The resulting function is a sinusoidal, a novelty for sub-pixel interpolation. It was able to match the data while also preserving a simple shape without the need for weird constants.

To verify the validity of our solution we used two synthetic and two real data sets. At this phase we wanted to use an image set for which the 3D model is known. This helps to eliminate any errors resulting from model inaccuracies. More complex real images do not allow this level of knowledge, but will be used in further work to validate our findings. The results for all four evaluations are consistent and prove that our approach can significantly reduce sub-pixel errors for the Semi-Global matching stereo algorithm. The same function fitting methodology is not limited to any given stereo solution and can be used to model even the most complex algorithms.

The results are also applicable to driving assistance systems, both for structured and unstructured environment description.

#### ACKNOWLEDGMENT

This work was supported by CNCSIS – UEFISCSU, project number PNII – IDEI 1522/2008.

#### REFERENCES

- Scharstein, D., Szeliski, R., "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", International Journal of Computer Vision, vol. 47 no.1-3, pp. 7-42, April-June 2002.
- [2] Hirschmüller, H., Innocent, P. R., and Garibaldi, J. "Real-Time Correlation-Based Stereo Vision with Reduced Border Errors" International Journal of Computer Vision vol. 47, no.1-3, pp. 229-246, April-June 2002.
- [3] Woodfill, J.I., et al., "The Tyzx DeepSea G2 Vision System, A Taskable, Embedded Stereo Camera", Embedded Computer Vision Workshop, pp. 126–132, 2006.
- [4] Hirschmüller, H., "Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information" IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR'05, vol. 2, pp. 807-814, June 2005.
- [5] Scharstein, D., Szeliski, R.: Middlebury stereo vision and evaluation page, <u>http://vision.middlebury.edu/stereo</u>
- [6] Fisher, R.B. Naidu, D.K., "A Comparison of Algorithms for Subpixel Peak Detection", Image Technology, Advances in Image Processign, Multimedia and Machine Visio, Springer-Verlag, 1996, pp. 385-404.
- [7] Bailey, D.G., "Sub-pixel estimation of local extrema", Image and Vision Computing, New Zealand, 2003, pp. 408-413.
- [8] Shimizu, M., Okutomi, M., "Precise sub-pixel estimation on areabased matching", Eighth IEEE International Conference on Computer Vision, ICCV 2001, Vancouver, Canada, pp. 90-97, 2001.
- [9] Haller, I., Pantilie, C., Oniga, F., Nedevschi, S., "Real-time semiglobal dense stereo solution with improved sub-pixel accuracy", Proceedings of the IEEE Intelligent Vehicles Symposium IV 2010, pp. 369-376, June 2010.
- [10] Hirschmüller, H. Scharstein, D., "Evaluation of Stereo Matching Costs on Images with Radiometric Differences," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 9, pp. 1582-1599, September, 2009.
- [11] Gehrig, S., Eberli, F., Meyer, T., "A Real-Time Low-Power Stereo Vision Engine Using Semi-Global Matching", Lecture Notes in Computer Science, Computer Vision Systems, vol. 5815, pp. 134-143, 2009.
- [12] Oniga, F., Nedevschi, S., "Processing Dense Stereo Data Using Elevation Maps: Road Surface, Traffic Isle and Obstacle Detection", IEEE Transactions on Vehicular Technologies, vol. 59, issue 3, pp. 1172-1182, 2010