The technology of image matching by the criterion of conformity of image fragments samples

V A Fursov¹,², A V Gavrilov¹, Ye V Goshin¹ and K G Pugachev¹

¹Samara National Research University, Moskovskoye shosse 34, Samara, Russia, 443086
²Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS, Molodogvardejskaya street 151, Samara, Russia, 443001

e-mail: fursov@ssau.ru

Abstract. In this paper we considered the image matching problem. The problem is solved in the framework of an area approach based on the comparison of image fragments in the neighborhood of the points being compared. We proposed a new criterion of image fragments similarity based on the principle of image fragments samples conformity. An important advantage of the method is the possibility of comparison of very small image fragments. However, computation of this criterion is very time-consuming. So, we proposed parallel algorithm of conforming image matching using MPI. We evaluated acceleration and efficiency of parallel algorithm. The results of experiments on the "Cones" test images show high accuracy of the method.

1. Introduction

The problem of image matching [1] is one of the most popular questions in technologies of image processing. Finding corresponding points is often an important stage in the technologies of localization and object recognition, model identification in image formation systems, and other tasks. At least three approaches to image matching problem solving are known: (1) area-based approach (image fragments are matched), (2) feature-based approach (values points, edges or angles are matched), and (3) relational approach (symbolic descriptions are used).

Area approach is the most widely used. In this approach, the fragment of the first image, named template, is searched at second image. Usually template sizes are odd values, because templates are centered in reference pixel. The template in the first image is compared with fragments of the same size in the second image. For every template position in the search area a value of similarity criterion is evaluated. The corresponding point is the point that minimizes or maximizes a value of similarity criterion.

In this paper we propose a new criterion of similarity of image fragments based on conformity estimation discussed in [2], [3].
2. Formulation of image matching problem

In this paper [4], the multistage taxonomy of image matching algorithms was presented by Scharstein and Szeliski. They used four main stages to describe image matching algorithms: matching cost computation, cost aggregation, disparity selection, and disparity refinement.

In general, image matching algorithms can be classified as local (area-based) and global approaches. Global methods treat disparity evaluation as a problem of global energy function minimization for all disparity values. They produce good results but are computationally expensive and cannot be used in real time systems. Global methods skip the second stage of taxonomy because they already use global information.

In contrast, local methods use only local information about intensity values within support windows and have low computational complexity. To assign disparity map values with local methods, one must use winner take all (WTA) optimization. Disparity value, that minimizes cost, is assigned to pixel. The matching cost is aggregated via a sum or an average value over the support window.

Usually area-based algorithms use absolute differences (AD) and squared differences (SD) to compute matching cost and the sum of absolute differences (SAD), sum of squared differences (SSD), normalized cross correlation (NCC), rank transformation (RT), and census transformation (CT) to aggregate matching cost. In detail, this matching cost computation and matching cost aggregation methods were described in paper [5].

Wang et al. [6] applied the AD algorithm to real-time image matching using GPU. The results of the AD algorithm show that it is not capable of producing a smooth disparity map in highly textured images. To overcome it, Min et al. [7] and Pham and Jeon [8] implemented truncated absolute differences (TAD) algorithm. The TAD algorithm uses the color and gradient in pixels matching to improve its robustness against variations in illumination.

Yang et al. [9] implemented the SD algorithm to subpixel image matching. They also used a bilateral filter to improve the flattening of edges and to smooth areas. In paper [10] Miron et al. researched various matching cost functions. They concluded that the SD algorithm provided the largest error.

Lee and Sharma [11] implemented real time disparity map estimation algorithm using the SAD algorithm to calculate matching costs via GPU. Gupta and Cho [12] implemented a technique using two different sizes of correlation windows in the SAD algorithm. As denoted in [5], the SAD algorithm is fast, but the quality of the initial disparity map is low because of the noise at object boundaries and in homogeneous regions.

In paper [13] Fusiello et al. tested the SSD algorithm on multiple fixed window blocks to reduce the incidence of occlusion errors. Yang and Pollefeys [14] implemented the SSD algorithm with multiple windows using GPU.

Satoh proposed low-dimensional image features matching method using NCC [15]. The NCC has achieved high accuracy, but a lot of computational resources are required. Cheng et al. [16] implemented matching cost calculation using a zero mean normalized cross correlation (ZNCC).

A disparity map algorithm using the RT algorithm was proposed by Gac et al. [17]. Using careful selection of the window sizes, a reliable initial disparity map was efficiently obtained. In paper [18], a new RT algorithm was developed to reduce matching ambiguity using a Bayesian model. This model considers similarity between the first and second image pixels and the level of ambiguity within each image independently.

Some researchers tried to improve image matching accuracy using combinations of various cost functions. For instance, Mei et al. [19] implemented a combination of the AD and CT algorithms to compensate their limitations. In this research, the CT algorithms produced incorrect matches in regions with repetitive local structures, and the AD algorithm did not work well in large texture less regions. The combination of the SAD and CT algorithms led to higher performance, but computational complexity has become higher as well [20]. The SAD and CT cost measures are obtained individually. The final cost function is constructed as a linear combination of both cost measures based on a weighting factor. The accuracy improvement was achieved by Zhang et al. [21] using a combination
of SAD and arm length differences (ALD) methods. Lee et al. [22] used combination of CT and gradient difference methods. However, matching ambiguity can appear in certain regions as a result of similar or repetitive texture patterns.

Perez-Patricio et al. developed a fuzzy logic method [23] using various cost functions simultaneously and decision making based on membership functions. The method uses local and global information in order to reduce errors in homogeneous areas and to preserve edges near discontinuities.

Let us state that $F_1$ and $F_2$ are two multi-view images. We specify image fragments describing neighborhood for every pixel of interest with size of $K \times L$. We will call this neighborhood the search window and assume that $K$ and $L$ are odd. In general case, $K$ is equal to $L$. Fragments are described by descriptors vectors $f_1$ and $f_2$ for the first and second images, respectively. Vectors consist of $K \times L$ elements. Every pixel in the first image search area with the size of $N \times M$ is specified in second image.

Thus, the task is about finding corresponding points in search area in the second image for points in the first image using a criterion of similarity of vectors $f_1$ and $f_2(n,m)$, where $n$ and $m$ are points of rows and columns in the search area. Vectors are commonly compared using AD or SD criterions.

It is obvious that increasing the search area size leads to computational costs increasing. A preliminary image alignment allows us to reduce search area size. The maximum relative offset must be considered.

In addition, to obtain the correct selection of search area size, matching fragments size should be selected. It is assumed that the reliability of matching should be increased with increasing of image fragments matched. Due to projective distortion, fragments of the same parts of the scene at multi-view images can differ a lot. Therefore, in terms of invariance to projective distortion, fragment size must be small.

However, the probability of similar distributions of intensity in different image fragments is increasing, for example, in texture fragments. Thus, it becomes necessary to construct a new criterion for the similarity of smaller fragments that take into account the “finer” differences in the intensity of distribution functions.

3. Criterion of conformity

We propose a new criterion of image fragments similarity based on the principle of image fragments conformity. The task is to find a corresponding point in the search area of the second image for a reference point in first image.

For every vector $f_2(n,m)$, that describes points from the search area with a size of $N \times M$ difference of vectors, it can be computed:

$$\Delta f(n,m) = f_1 - f_2(n,m), \quad n = 1, \ldots, N, \ m = 1, \ldots, M.$$

We use criterion of image fragment similarity at position $(n,m)$:

$$W(n,m) = \sum_{i,j=1}^{S} (\Delta f_i(n,m) - \Delta f_j(n,m))^2,$$

where $\Delta f_i$ is $i^{th}$ element of vector (1), $S = K \times L$ is elements count of $f_1$ and $f_2$, vectors. The point in the search area that minimizes criterion (2) is selected as the corresponding point for point in fragment in first image with size of $K \times L$:

$$(\hat{n}, \hat{m}): \ W(\hat{n}, \hat{m}) = \min_{n,m} W(n,m).$$

The main idea of criterion (2) is that the summation of squared differences of compared vectors components using all possible pair combinations leads to a significant increase of summation elements count.
Therefore, we may conclude that even with small image fragments, reliability of results is higher. For example, using image fragments with a size of $5 \times 5$ with common per-pixel comparison gives only 25 elements, a very small sample. Use of criterion (2) gives 300 summation elements. The reliability is significantly higher because requirements of the law of large numbers are met [24].

Let us now describe the image matching algorithm in details. The algorithm consists of three main stages:
1) Initialization of algorithm parameters and images preparing;
2) Finding corresponding points using criterion of conformity;
3) Disparity map construction.

At the first stage of algorithm search window, a size of $K \times L$ and a search area with the size of $N \times M$ are initialized. We used square fragments ($K = L$). We considered possible maximum disparity value, when we initialize a search area length $M$. In this case, when the rectified images are used, a search area height of $N$ is initialized equal to one.

Due to the changing camera position, some objects, which are visible in the first image, cannot be observed in second image. Thus, the area of disparity image calculation is specified based on search window size and search area size. This significantly decreases the execution time of the algorithm.

At the second stage of the algorithm, for every point of the first image in the area of disparity image calculation, a search of corresponding points within the second image is executed. Vectors $f_i(i, j)$ are formed for points in the first image. The search areas are specified for points intersecting with the first image at second image. In general, the central point of the search area is point $(i, j)$. For every point in the second image, vectors of $f_j$ are formed.

For every position of the search window vector of differences, $\Delta f_{ij}(n, m)$ is calculated. Then, values of similarity criterion $W(n, m)$ are calculated. The point with the lowest criterion value is selected as the corresponding point.

At the third stage of the algorithm, the values of relative offsets for every point are calculated, and the disparity map is constructed.

4. Parallel MPI-implementation of conforming image matching algorithm

Although the criterion of conformity works with small amount of data, the total computational complexity of algorithm is quite high. This is because even with a small size of search window a large count of comparisons is performed.

Let us now consider the most simple case and use a search window size of $3 \times 3$. Here the count of elements of vectors $f_i$ and $f_j$ will be equal to $S = K \times L = 9$. In this case, for every point in the search area of $C^2_S = C^2_9 = 36$ comparisons will be performed. If we use a search area size of $1 \times 150$, 5400 comparisons of elements of vector $\Delta f$ will be performed for the one point. If we consider 1000 points, $5.4 \times 10^6$ operations of comparison will be performed.

Ever for smaller sizes of the search window a count of comparison operations is large. So, to reduce execution time of our method, a parallel algorithm using MPI technology was constructed. The parallel algorithm consists of the same three main stages, as well as a sequential algorithm:
1) Initialization of algorithm parameters and images preparing;
2) Finding corresponding points using criterion of conformity;
3) Disparity map construction.

Every process has a copy of the search window size $K \times L$ and the search area size $N \times M$. To perform image matching for images of large size and allow using more calculation resources, calculations are distributed between MPI-processes. The entire area of the image of disparity values calculation is divided into several parts of approximately the same size, and calculations are executed separately. Then, parts of disparity maps are combined into full disparity map.
In this example, we will use an area of disparity image calculation that contains \( X \) rows and \( Y \) columns and uses 9 MPI processes. In this case, we can divide the disparity image calculations using \( I = 3 \) horizontally parts and \( J = 3 \) vertical parts. Figure 1 shows parallel algorithm in detail. The process calculation partitioning will have the following form.

![Disparity Map](image)

**Figure 1.** Use of the MPI-processes.

In this case, processes with numbers from 0 to 5 will perform calculations for \( X_{\text{div} 3} \) rows, while processes with numbers from 6 to 8 will perform calculations for \( X_{\text{div} 3} + X_{\text{mod} 3} \) rows. Operations \( \text{div} \) and \( \text{mod} \) are the integer division and remainder of division respectively. At the same time, processes with numbers 0, 1, 3, 4, 6, and 7 will perform calculations for \( Y_{\text{div} 3} \) columns, while processes with numbers 2, 5 and 8 will perform calculations for \( Y_{\text{div} 3} + Y_{\text{mod} 3} \) columns. It should be noted that the distribution of calculations can be used analogically for any count of processes and values of \( I \) and \( J \).

To implement the parallel algorithm, a program was written using programming language C++ and MPI. Values of acceleration and efficiency [25] of the parallel algorithm were obtained. Figure 2 shows the test images [26] that were used to construct the disparity maps.

![Test Images](image)

**Figure 2.** Dataset of test images: (a) first image; (b) second image.

Figure 3 shows the execution time of the algorithm with various sizes of search window. The computation was performed using nodes with processors Intel Xeon X5560 of supercomputer “Sergey Korolyov”.

Figures 4-5 show the execution time of a parallel algorithm with search window size \( 3 \times 3 \) and various count of MPI-processes varied from 1 to 4.

The figures show that the algorithm has high values of acceleration and efficiently. The algorithm is good scalable. This is due to the uniform distribution of the calculations between the processes.
5. Results of experiments

To evaluate the accuracy of the proposed criterion, a disparity map was constructed for two multi-view images presented in previous section using small search window size $3 \times 3$. Considering that images were rectified, a search area size of $1 \times 111$ was used. To reduce noise on the disparity map, we used a median filter using filter windows of sizes $5 \times 5$ and $7 \times 7$.

To compare the obtained results with the other methods and algorithms, we constructed a disparity map using semi-global block matching method. We used the implementation of this method from OpenCV library [27] with parameters: minDisparity=0, numberOfDisparities=80, disp12MaxDiff=1, preFilterCap=161, uniquenessRatio=10, speckleWindowSize=100, speckleRange=2.

To improve the readability of results, we changed the average intensity of disparity maps pixels to 128. Figure 6 shows obtained and ground true disparity maps with corrected average intensities.
Figure 6. Disparity maps: (a) ground true disparity map; (b) semi-global block matching method; (c) conforming matching with median filter with window size of 5 x 5; (d) conforming matching with median filter with window size of 7 x 7.

We calculated the matching error for every obtained disparity maps using equation 3:

$$\sigma = \sqrt{\sum_{i,j} (I_{i,j}^1 - I_{i,j}^2)^2},$$

where $I_{i,j}^1$ are values of pixels intensity of ground true disparity map, $I_{i,j}^2$ are values of pixels intensity of obtained disparity map, and $i$ and $j$ are indices of disparity maps pixels. The results of accuracy evaluation are presented in Table 1.

Table 1. Accuracy of image matching.

| Filter window size | Error value | Conforming image matching | Semi-global block matching |
|--------------------|-------------|---------------------------|----------------------------|
| 5 x 5              | 7.01        |                           | 7.18                       |
| 7 x 7              | 6.06        |                           |                            |

The use of median filtering allows us to reduce noise of disparity maps a lot. At the same time edges are still clear. A median filter with a window size of 5 x 5 gives error value equal to 7.01, that is less then error of semi-global block matching. A median filter with a window size of 7 x 7 allows for a reduction in error value, but the edges of the objects are slightly blurred. Utilizing a median filter with a window size of 9 x 9 and larger distorts the edges of objects significantly.

6. Conclusion

The results of our research show that the method of conforming image matching can give more accurate results than semi-global block matching. We have established that the optimal window size of a median filter is 7 x 7 because it reduces the error value measurably and preserves the clarity of objects’ edges. The error value of conforming matching method after filtering amounted to 6.06, at the same time while the error value of semi-global block matching amounted to 7.18. Thus, our research has allowed us to conclude that the conforming matching method can give a quality result even with a smaller sized search window.

In our future research, we will apply our method to simultaneously localisation and mapping systems.

7. References

[1] Gruzman I S, Kirichuk V S, Kosih V P, Peretyagin G I and Spektor A A 2002 Digital image processing in information systems (Novosibirsk: NGTU) p. 352
[2] Fursov V A, Gavrilov A V, Goshin Ye V, Pugachev K G 2017 Conforming identification of the fundamental matrix in the image matching problem *Computer Optic* 41(4) 559-563 DOI: 10.18287/2412-6179-2017-41-4-559-563

[3] Fursov V A, Gavrilov A V, Goshin Ye V, Pugachev K G 2017 Parallel algorithm of conforming fundamental matrix estimation in image matching problem *Russian Supercomputing Days: Proc. of the int. conf.* (Moscow: MGU) 683-691

[4] Scharstein D and Szeliski R 2002 A taxonomy and evaluation of dense two-frame stereo correspondence algorithms *Int. J. Comp. Vis.* 47 7-42

[5] Rostam A H, Haidi I 2016 Literature Survey on Stereo Vision Disparity Map Algorithms *J. Sens.* 8742920

[6] Wang L, Gong M, Gong M and Yang R 2006 How far can we go with local optimization in real-time stereo matching *Proc. 3rd Int. Symp. 3D Data Proc. Vis. Trans.* (Chapel Hill, NC, USA/ IEEE) 129-136

[7] Min D, Lu J and Do M N 2011 A revisit to cost aggregation in stereo matching: how far can we reduce its computational redundancy? *Proc. IEEE Int. Conf. Comp. Vis. (ICCV ’11)* (Barcelona, Spain/IEEE) 1567-1574

[8] Pham C C and Jeon J W 2013 Domain transformation-based efficient cost aggregation for local stereo matching *IEEE Trans. Circ. Sys. Vid. Tech.* 23 1119-1130

[9] Yang Q, Yang R, Davis J and Nistrier D 2007 Spatial-depth super resolution for range images *Proc. IEEE Conf. Comp. Vis. Pat. Rec. (CVPR ’07)* (Minneapolis, Minn, USA/IEEE) 1-8

[10] Miron A, Ainouz S, Rogozan A and Bensrhair A 2014 A robust cost function for stereomatching of road scenes *Pat. Rec. Let.* 38 70-77

[11] Lee S H and Sharma S 2011 Real-time disparity estimation algorithm for stereo camera systems *IEEE Trans. Cons. Electr.* 57 1018-1026

[12] Gupta R K and Cho S Y 2013 Window-based approach for fast stereo correspondence *IET Comp. Vis.* 7 123-134

[13] Fusiello A, Castellani U and Murino V 2001 Relaxing symmetric multiple windows stereo using Markov Random Fields *Computer Vision and Pattern Recognition* 2134 91-105

[14] Yang R and Pollefeys M 2003 Multi-resolution real-time stereo on commodity graphics hardware *Proc. IEEE Comp. Soc. Conf. Comp. Vis. Patt. Rec. 1* 1/211-1/217

[15] Satoh S 2011 Simple low-dimensional features approximating NCC-based image matching *Pat. Rec. Let.* 32 1902-1911

[16] Cheng F, Zhang H, Yuan D and Sun M 2014 Stereo matching by using the global edge constraint *Neurocomp.* 131 217-226

[17] Gac N, Mancini S, Desvignes M and Houzet D 2008 High speed 3D tomography on CPU, GPU, and FPGA *EURASIP J. Emb. Sys.* 930250

[18] Zhao G, Du Y K and Tang Y D 2011 Anew extension of the rank transform for stereo matching *Adv. Eng. Forum* 2-3 182-187

[19] Mei X, Sun X, Zhou M, Jiao S, H. Wang and Zhang X 2011 On building an accurate stereo matching system on graphics hardware *Proc. IEEE Int. Conf. Comp. Vis. Work. (ICCV Workshops ’11)* 467-474

[20] Cigla C and Alatan A A 2013 Information permeability for stereo matching *Sig. Proc.: Im. Com.* 28 1072-1088

[21] Zhang N, Wang H and Cr J 2013 A near real-time color stereo matching method for GPU *Proc. 3rd Int. Conf. on Adv. Comp. Comm.* 27-32

[22] Lee S, Lee J H, Lim J and Suh I H 2015 Robust stereo matching using adaptive random walk with restart algorithm *Im. Vis. Comp.* 37 1-11

[23] Pérez-Patricio M, Aguilar-González A, Camas-Anzueto J-L and Arias-Estrada M 2015 A Fuzzy Logic Approach for Stereo Matching Suited for Real-Time *Proc. Int. J. Comp. App.* 113 9

[24] Prohorov Yu V and Ponomarenko L S 2012 *Lectures on probability theory and mathematical statistics* (Moscow: Moscow University publishing house) p. 256
[25] Voevodin V V and Voevodin VI V  2002 *Parallel computing* (St. Petersburg: BHV-Peterburg) p. 608

[26] University of Tsukuba Computer Vision Laboratory (Access mode: http://www.cvlab.cs.tsukuba.ac.jp/dataset/tsukubastereo.php)

[27] OpenCV Library (Access mode: http://opencv.org)

**Acknowledgments**

The work was funded by the Russian Federation Ministry of Education and Science (project # 2.7891) and RFBR (projects ## 16-07-00729, and 17-29-03112).