Blur-robust image registration and stitching

Y Donon¹, R Paringer¹,², A Kupriyanov¹,² and Y Goshin¹,²

¹Samara National Research University, Moskovskoe Shosse 34A, 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: yegor.goshin@gmail.com

Abstract. This paper addresses the functioning of a new key point detection method for images. Such methods are useful in applications as image reconstructions, object signature recognition or video stabilization. The method, Blurred Image Matching (BIM), is based on the comparison of large areas of interest in images. BIM uses a process of blurring and shape comparison that makes the approach unique and in the vast majority of tested cases results in higher performances than other approaches. Among BIM’s key features are the amount of points used for comparison, 19 to 129 times fewer than other methods and of higher quality, allowing faster processing. Another of its key features is a high efficiency in the comparison of images, with a success rate higher than the average of existing methods by 13% to 36%, depending on the images’ characteristics. The method is still in a development stage and will significantly improve, however the results obtained so far are extremely promising, its task solving approach is new and makes a new range of applications necessitating key point detection possible. All this allows us to foresee BIM as a future standard among such methods.

1. Introduction

Human eyes are able to catch in a fraction of a second the subject of an image. Yet in that time we are unable to process its details. We focus on central points of interest, key points, and that is enough for us to identify what matters the most in that given image. We inherit that ability from millions of years of evolution and it has been a key feature for the survival of our species. In this article, we present a new method of key point detection inspired by this ability humans have not to use a vast quantity of detailed points but to understand general shapes, have a global vision.

Image reconstruction, object signature recognition, video stabilization, are all implementations necessitating detecting key elements in images [1], some approaches already exist to find these elements [2] but their most common struggle is dealing with low resolution, blurred, masked pictures, any “noised” picture [3]. A common approach is to run algorithms to attenuate those sources of dissimilarity, deburr pictures [4], sharpen angles [5] and highlight shapes [6, 7]. In this paper, we present a new method to detect key points on images. This method performs better than other existing ones, namely Harris, FREAK and SURF, in the vast majority of cases. In order to assess the method presented in this paper, we chose to apply it to picture stitching and assemble systematically pairs of partial pictures into a stitched one. The method “Blurred Image Matching” or “BIM” was first thought...
to deal with pictures presenting a degree of noise too high to use existing approaches and with the idea of; “If we can’t remove the blurring, let’s blur it further”. However, during our experiments, it performed above expectations, leading us to develop BIM further and push the experiments to a wider set of problems, as described in this article.

![Example of matching and stitching of two images using BIM.](image)

**Figure 1.** Example of matching and stitching of two images using BIM. (1) - left fragment, (2) – right fragment, (3) stitching of both fragments.

2. Proposed method

2.1. Method

BIM’s approach is based on the pre-processing of pictures using, grayscale transformation (1), Gaussian blur (2), thresholding (2), edge detection (2) and shape recognition (3; 4). This process’ main steps are the Gaussian blur, which allows blending imperfections into the rest of the picture and the thresholding, which allows distinguishing areas of interest. The efficiency of this process has already been assessed (5). The last phase of picture pre-treatment consists of identifying approximate polygons out of the shapes created through the process of edge detection (shape recognition).

Once polygons are identified, they are compared to each other in area, width and height, if two of the polygons are sufficiently close to each other, as described in 1.2.1, they are considered matching. Matching polygon’s edges coordinates’ are then added to the list of matching coordinates, later used for the correlation.

![Process of picture transformation:](image)

**Figure 2.** Process of picture transformation: the original picture (1), the grayscale transformation (2), Gaussian blurring (3), thresholding (4), edge detection (5) and shape recognition (6).
The result of this filtering is the shapes (elements) as highlighted in red on Fig. 2 - (6). The shapes are polygons of different but recognizable characteristics that are compared between each other from one picture to the next.

Our aim is not to find elements matching each other completely, but to find several elements relatively similar offering a relatable translation from one frame to another. As displayed above, the thresholding unifies areas by diminishing the contrast of colours, then the Gaussian blur drowns imperfections, before the thresholding, that highlights shapes out of the image. Finally, edges are marked for the shape recognition, which searches for close shapes.

2.2. Shapes comparison

2.2.1. Shapes selection
As mentioned in the previous chapter, the elements used to compare images are shapes. Shapes are identified using a connected components labelling algorithm (6) available through Accord.NET’s imaging namespace.

Once selected, shapes are filtered, they should not be “too big” as this most likely means a false positive (such as a shape circling the whole image) nor too small, as they are then not significant for comparison (small shapes easily resemble each other according to the algorithm’s criteria). As such, shapes selected must have an area superior to 0.2‰ of the image’s total area. Moreover, their respective height and width must be inferior to 90% of the image’s height and width (Fig. 3, shapes selection parameters). The three criteria described above are sufficient for the RANSAC homography (7) as described in 1.2.2 to function optimally.

2.2.2. Tolerance
The tolerance designates the margin used during the analysis. Its main application is to determine acceptable differences between two shapes, from one image to another. The general tolerance degree has been empirically set to 8% during our experiments after a multitude of tests involving different values and it appeared as the most performant.

As such, the correlation between two shapes is made when the area, the width and the height of a shape all fit another, within the tolerance margin. All shapes may be correlated from zero to many other shapes. Once correlated, the points are run through a RANSAC homography method (8). The method selects 4 random 4 correspondences (points), transforms them into a given homography (9) and repeats the process for a number of iterations. The most successful iteration in terms of inliners is then selected as the best homography. This approach both filters the noise among points found and creates the homography for the pictures’ matching (Fig. 3, Homography – image stitching parameter).

2.2.3. Filtering parameters
Depending on an image’s characteristics the filters do not present the same results. For every new series of images, it is necessary to determine the ideal filtering parameters, as for now the determination is empirical, practice has shown that the results are best having on average six to eight shapes considered appropriate to work with. The steadiest results have been obtained using pre-processing parameters (Fig. 3): Gaussian blurring of sigma 8 and a kernel size 17 by 17, followed by a threshold degree of 100).

2.2.4. No-match
In case no matches are found from an image to another, the comparison is considered impossible, meaning that two pictures do not present sufficient similarities according to the software. In case of a collection of images, the next image is analysed to find a match, ignoring the unselected one.

3. Procedure
To assess the new method, it has been compared to several, existing algorithms. We selected a feature extraction, a corner detection and a key point detection algorithm. Namely SURF (10), FREAK (11) and Harris (12) having a reputedly performant implementation in Accord.net framework for C# (13).
The idea behind this selection was not only to select performant algorithms but also to have them run in a similar environment. In terms of process, after the selection of features and their correlation, all methods have seen their points filtered through the Accord.net implementation of RANSAC (8).

**Figure 3.** Process of image stitching representation, from the entry of new images to the resulting stitched image.

**Figure 4.** Comparison between two successfully matched pictures, with a Bhattacharyya difference index of 1.5%.
3.1. Output comparison

In order to assess the quality of our outcomes, we adopted the XnaFan ImageComparison method (14) based on the Bhattacharyya histogram algorithm (15), which has already proven itself reliable (16). All our datasets contained original pictures, or in other words pictures “as they should be” in case of ideal matching. We considered empirically a successful match two pictures presenting a Bhattacharyya difference index lower than 2%. Our reference for the selection of this index is the human perception of being in front of two similar pictures, as illustrated in Fig. 4 to Fig. 6.

![Figure 5. Comparison between two unsuccessfully matched pictures, with a Bhattacharyya difference index of 19.6%.](image)

![Figure 6. Demonstration of a failed match, with a Bhattacharyya difference index of 36.8%.](image)

3.2. Comparison to other methods

3.2.1. Regular set

The first set was meant to evaluate BIM’s performance on regular pictures. The sample is constituted of about 5000 images of different sources presenting various exposition and subjects.

3.2.2. Noised set

One of the developed method core functions is to allow dealing with data presenting a high degree of noise. For this purpose a set of 14 pictures has been treated with different noise, combined or not, for a total of about 4000 samples. The images are of different sizes and kind, from maps to paintings, landscapes or buildings. The noises imposed on the pictures have been generated uniformly using OpenCV and the parameters described in this chapter. Some noises have also been combined following the same parameters, as showed in Fig. 7 to Fig. 11 (all the presented samples were successfully recognized by the method described in this paper):

- Fig. 7. Blurring has been done using a normalized box filter (17), with a kernel size of \((x; x)\), \(x\) comprised between 5 and 200 incremented by steps of 5.
- Fig. 8. A salt-and-pepper layer has been added with a probability of the noise varying linearly from 0.01 to 0.4.
- Fig. 9. A Multiplicative noise (speckle noise) has been added depending on picture’s height, width and depth, all with an intensity between 500 and 20'000.
- Fig. 10. An artificial perspective has been added with a deformity from the point of origin of 1/4 of the image’s size.
- Fig. 11. A filter presenting fixed defaults and adding a white layer has been added to represent the picture taken in a situation of extreme cloudiness with a damaged objective. The filter has been overlaid with an intensity varying from 2.5% to 97.5%, by steps of 2.5%.
Figure 7. Comparison between a picture before (left) and after a blurring operation with a kernel size of 200 by 200 (right).

Figure 8. Comparison between a picture before (left) and after a salt-and-pepper noise application with a probability of 0.4 (right).

Figure 9. Comparison between a picture before (left) and after a multiplicative and blurring noise operation with a kernel size of 150 by 150 and an intensity of 7'500 (right).

Figure 10. Comparison between a picture before (left) and after a perspective noise application with an inclination of 200px (right).

Figure 11. Comparison between a picture before (left) and after a filter application with an intensity of 37.5% (right).
4. Results

4.1. Computation time
The four methods’ computation time has been assessed using the regular set as described in 2.2.1. The solution was ran through the whole set using similar pictures cut to different sizes, ranging from 250 thousand pixels to 4 million. As shown in Fig. 12, BIM takes significantly less time than the other methods - on average 76.8%.

The experiment has been performed on a computer using an Intel Core i3-8100 at 3.60GHz.

![Figure 12](image1.png)

**Figure 12.** Computation time comparison between the different methods. In this figure BIM stitching takes significantly less time than with other methods, independently from image sizes.

4.2. Precision
In terms of success rate, as defined in chapter 2.1, on the regular set, BIM arrives second to Harris however, when compared to the noised set, BIM clearly over-takes all others as show in Fig 13.

In other terms, one of BIM’s main characteristics is to perform well despite situations presenting a high noising degree, when other methods are significantly weaker.

![Figure 13](image2.png)

**Figure 13.** Methods’ success rate comparison in percent, on both sets. In this figure BIM presents the second highest performance on the regular set and the highest on the blurred set.

4.2.1. Matching through different noises
It is possible through the method presented in this document to match pictures presenting drastically different noise intensities. Although minor shapes blend into their environment, significant ones remains and allows the matching of two pictures otherwise extremely different as shown in Fig. 14.

4.3. Difference in noise types
Not all methods show similar performances depending on the kind of noise imposed on the picture. Fig. 15 shows the results of the different methods when applied to specific noises.
Figure 14. Matching comparison with different noise levels. On the top, the same images with different noise levels. On the bottom, the resulting shapes of both images.

Figure 15. Efficiency comparison between methods on given noises. In this figure BIM shows a high performance in the treatment of noised images, only second to SURF to match “filtered” and “filtered and blurred” images and to Harris to handle “perspective” noise.

Overall, BIM is systematically first or second to other methods. Interestingly, Harris performs remarkably well in dealing with perspective.

BIM shows a weakness in the comparison of filtered images due to the important diminution of contrast existing in those kind of images, which makes distinguishing shapes more difficult.

BIM detects significantly less points than any of the methods it was compared to as only shapes of significant size are used. Fig. 16 illustrates the average amount of points found per 10 thousand pixels.

This could present a problem for particularly small pictures, however it is not significant as BIM eliminates small objects in an image depending on its size, making the average of points found non-linear. According to our experiments, BIM is able to deal with pictures down to 160x120 pixels making it usable in the vast majority of situations.

On the other hand, this presents a serious advantage when processing large amounts of pictures. In our experiments we only processed images by pairs, but a common problem (18) is having to match
large amounts of pictures together, possibly in no given order. In that case, the processing time grows exponentially as each picture has to be compared to the rest of the set. In this situation, it is a serious advantage to limit the amount of points detected. As highlighted in Fig. 16 BIM uses on average 19 to 129 times fewer points than its counterparts do, thus reducing significantly the necessary processing time. The processing time issue is recurrent (19; 20) for such methods and this approach is a new and very efficient way to address it.

![Figure 16. Average points found by method on 10k pixels images. This figure highlights how fewer points are selected by BIM compared to other methods.](image)

If it takes fewer points than other methods, BIM has the points with the highest quality, understand the points necessitating the smallest noise filtering. Fig. 17 presents the percentage that are matched and then filtered through the RANSAC algorithm, a hundred percent being the total of points detected.

![Figure 17. Percentage of points kept after the matching and filtering operations. This figure shows BIM’s higher ratio of points usable for the homography. A higher ration means a higher point’s quality.](image)

5. Future research
This method performed extremely well and opens doors for further research in order to improve its functioning. Points to be researched in the future are a sorting method more appropriate to the amount of BIM’s matched points (21), the use of new sets of features to match key points, an adaptive system
to find the optimal parameters relative to an image, or methods meant to normalize sets of picture before treatment (22)

6. Conclusion
In this article we presented a method inspired the human’s natural ability of understanding the content of a scene by observing it globally. BIM allows the detection key points in images with a success rate, in the framework of our experiments, higher than similar methods and presents specificities in its function allowing it to deal with new sets of problems. In terms of performance, BIM was faster and presents a generally higher tolerance to the images’ noises. In terms of functioning, the method uses a much smaller quantity of high quality points for its matching process, making it particularly good for the comparison of large and non-continuous sets.

On a final word, by publishing these results we hope to generate new inputs and contributions for the evolution of this research, helping us to make this method reach a potential that we foresee, given the results obtained, to be considerable. The first paragraph after a heading is not indented (Bodytext style).

7. References
[1] Smith S W 1999 The Scientist and Engineer's Guide to Digital Signal Processing, Second Edition (San Diego, California: California Technical Publishing)
[2] Shapiro L and Stockman G 2001 Computer Vision (Seattle, Washington: Prentice Hall)
[3] Gomez S 2011 Shape recognition using machine learning (Manizales, Colombia: IEEE)
[4] Ngo T T, Nagahara H, Nishino K, Taniguchi R I and Yagi Y 2019 Reflectance and shape estimation with a light field camera under natural illumination Journal of Computer Vision 1-16
[5] NitinTaran and Kamra V A 2015 Review: Image Edge Unmasking by Applying Renovated and Colony Optimization Technique International Journal of Advanced Research in Computer and Communication Engineering 4 35-38
[6] Accord.NET BlobCounter Class URL: http://accord-framework.net/docs/html/T_Accord_Imaging_BlobCounter.htm
[7] Urbancic T, Fras M K, Stopar B and Koler B 2014 The Influence of the Input Parameters Selection on the RANSAC Results Journal of Simulation Modelling 13 159-170
[8] Accord.NET RansacHomographyEstimator Class URL: http://accord-framework.net/docs/html/T_Accord_Imaging_RansacHomographyEstimator.htm
[9] Dubrofsky E 2007 Homography Estimation (Vancouver: The University of British Columbia)
[10] Accord.NET SpeededUpRobustFeaturesDetector Class URL: http://accord-framework.net/docs/html/T_Accord_Imaging_SpeededUpRobustFeaturesDetector.htm
[11] Accord.NET FastRetinaKeypointDetector Class URL: http://accord-framework.net/docs/html/T_Accord_Imaging_FastRetinaKeypointDetector.htm
[12] Accord.NET HarrisCornersDetector Class URL: http://accord-framework.net/docs/html/T_Accord_Imaging_HarrisCornersDetector.htm
[13] de Souza C 2014 Automatic Image Stitching with Accord.net
[14] Krarup J 2019 ImageComparison URL: https://www.nuget.org/packages/ImageComparison/
[15] Bhattacharyya A 1943 On a measure of divergence between two statistical populations defined by their probability distributions Bulletin of the Calcutta Mathematical Society 35 99-109
[16] Soe N N 2015 Image Matching Schemebyusing Bhattacharyya Coefficient Algorithm Journal of Innovative Research in Computer and Communication Engineering 25 6364-6370
[17] OpenCV 2014 Image Filtering. OpenCV documentation URL: https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=gaussianblur#gaussianblur
[18] Cao S and Snavely N 2012 Learning to Match Images in Large-Scale Collections (Ithaca, New York: Cornell University)
[19] Goshin Ye V 2018 Estimating Intrinsic Camera Parameters Using the Sum of Cosine Distances Journal of Physics: Conference Series 1096
[20] Fursov V A, Goshin Ye V and Kotov A P 2016 The hybrid CPU/GPU implementation of the computational procedure for digital terrain models generation from satellite images Computer
Optics 40(5) 721-728 DOI: 10.18287/2412-6179-2016-40-5-721-728

[21] Kotov A P, Goshin Ye V, Gavrilov A V and Fursov V A 2017 DEM generation based on RPC model using relative conforming estimate criterion Procedia Engineering 201 708-717

[22] Finlayson G, Gong H and Fisher R B 2019 Color Homography: Theory and Applications IEEE Transactions on Pattern Analysis and Machine Intelligence 41(1) 20-33

[23] Rosenthaler L, Heitger F, Kübler O and von der Heydt R 1992 Detection of general edges and keypoints European Conference on Computer Vision 77-78

[24] Lindeberg T 2009 Encyclopedia of Computer Science and Engineering (Hoboken, New Jersey: John Wiley and Sons)

[25] Wang Y, Wei X, Ding L, Tang X and Zhang H 2019 A robust visual tracking method via local feature extraction and saliency detection The Visual Computer 1-18

[26] Banner R and Staelin C 2012 Blurred digital image deblurring US 8.249,377 B1 USA

[27] Berg C A, Berg L T and Malik J 2005 Shape Matching and Object Recognition Using Low Distortion Correspondences IEEE Computer Society Conference on Computer Vision and Pattern Recognition 26-33

[28] Ancuti C, Ancuti C O and Bekaert P 2009 Deblurring by Matching Eurographics 28 619-628

[29] Song Y, Zhang J, Gong L, He S, Bao L, Pan J, Yang W and Yang M-H 2019 Joint Face Hallucination and Deblurring via Structure Generation and Detail Enhancement Journal of Computer Vision 127(6-7) 785-800

Acknowledgments

This work was financially supported by the Russian Foundation for Basic Research under grants # 19-29-01135, # 19-29-01235, # 17-29-03112, # 17-01-00972 and by the Ministry of Science and Higher Education within the State assignment to the FSRC “Crystallography and Photonics” RAS No. 007-GZ/Ch3363/26 (theoretical results).