COMPARISON OF IMAGE MATCHING ALGORITHMS ON SATELLITE IMAGES TAKEN IN DIFFERENT SEASONS

I. Yildirim [1,2]*, F. Demirtas [1,2], B. Gulmez [1,2], U.M. Leloglu [3], M. Yaman [1], E. T. Guneyi [1]

[1] ESEN Sistem Entegrasyon, Titanyum C Blok Kat 2, ODTU Teknokent, 06800, Çankaya, Ankara, Turkey
[2] Hacettepe University, Department of Electrical and Electronics Engineering, 06800 Çankaya, Ankara, Turkey
[3] Middle East Technical Univesity, Geodetic and Geographic Information Technologies Department, 06800 Çankaya, Ankara, Turkey
(irem.yildirim , fatih.demirtas , baran.gulmez, ugur.leloglu , mustafa.yaman , eylem.guneyi **)**@esensi.com.tr

KEY WORDS: Feature matching, invariance to seasons, robust matching

ABSTRACT:

Image matching, which aims to find the corresponding points in different images, is an important process which is used in various vision-based applications in military, industrial, remote sensing and security systems. Some applications require accurate matching across images taken at different times of the year to be reliable and reusable. Although many detection and description methods are used for image matching, it is important to correctly determine the most robust method for such changes. In this paper we investigate combination of SIFT (Scale Invariant Feature Transform), SURF (Speed Up Robust Features), KAZE, BRISK (Binary Robust Invariant Scalable), FAST (Features from Accelerated Segment Test) algorithms using satellite images that are taken at different times of the year in various seasons and weather conditions. Incorrect matches in the test results are eliminated by MLESAC (Maximum Likelihood Estimation SAmple and Consensus) method. As a result of these eliminations, the accuracy, propagation, changes in the number of the keypoints and the speed of detection of the keypoints are observed. At the end of these analyses, it is concluded that most reliable method in keypoint matching is FAST-SIFT despite the high cost of its computation time.

1. INTRODUCTION

Image matching is the process of finding correspondence between similar datasets on a given pair of images. It is a crucial task for many applications such as robot navigation, motion tracking, image stitching-, mosaicking (Santosh, Achar, & Jawahar, 2008), localization-mapping (Li, 2017), automatic surveillance (Fuentes & Velastin, 2006), visual odometry (Milford, McKinnon, Warren, & Wyeth, 2011) or pose estimation (Sminchisescu, Bo, Ionescu, & Kanaujia, 2011). To be able to get accurate results in these applications, we need robust outcomes from corresponding matches.

Especially images that are used in computer vision-photogrammetry combined applications, contain mountains, lakes, geographic structures or cities, fields etc. There are many algorithms to get strong features from images that can aid correct matching, however matching images from different seasons is difficult because as the time passes geographical structures vary depending on the natural events and seasons. These changes can be like defoliation, expansion of rivers, landslides etc. In the same way, man-made structures are affected by natural events, also they may be destroyed. Thus, structure of features is significantly affected from size, texture, illumination, neighbor changes and additional noise.

In this paper, we investigate the combination of well-known detection and description algorithms (SURF-SIFT, SURF KAZE, KAZE-KAZE, BRISK-SIFT, FAST-SIFT) that give the highest number of inlier points and the algorithms that are least affected from seasonal changes. We also investigate computation time and match number of these algorithms to be able to decide which is usable in real time applications.
The rest of this paper is organized as follows: Section 2 presents main steps of image matching, overview and principles of methods. In Section 3, experimental setup, datasets, experiments are shown. Finally, the paper is concluded in Section 4.

2. BACKGROUND AND LITERATURE REVIEW

Image matching basically consists of four stages: Feature detection; feature description; feature matching and outlier rejection (Hassaballah, Abdelmgeid, & Alshazly, 2016).

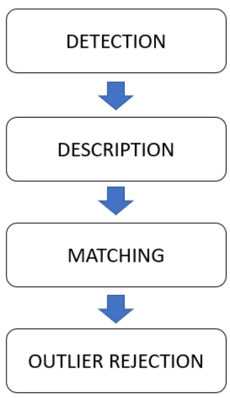


Figure 1. Stages of Image Matching Algorithms

Feature detection stage consists of extracting specific locations in images like corners, peaks, interestingly shaped patches that are dominant in the image. These specific locations are called keypoints or interest points. Detected points can be in the form of blobs, edges, corners, junctions, lines etc. ("Computer vision: algorithms and applications," 2013)In description stage, each keypoint is identified according to the neighbor pixels and therefore we can recognize them despite scale, rotation and illumination changes. After describing the features, we aim to find best correspondence between these images in the matching

[1]* Corresponding Author

stage. There are many kinds of approaches in feature-based image matching like, Hamming distance (for binary descriptors), (Friedman, Bentley, & Finkel, 1997), Voroni diagram (Friedman et al., 1997), k-d tree (Bentley, 2002), threshold-based matching, nearest neighbor-based matching, nearest neighbor distance ratio (Mikolajczyk & Schmid, 2005), L1-norm or L2-norm (Demirci, Osmanlioglu, Shokoufandeh, & Dickinson, 2011) (Demirci et al., 2011) and Fast Approximate Nearest Neighbors Approach (Muja & Lowe, 2014). Finally, wrong matches are rejected in outlier rejection part. Random Sample Consensus (RANSAC) (Cartography & Park, 1981), M-estimator Sample Consensus (MSAC) (Wang, Mirota, & Hager, 2010), Progressive Sample Consensus (PROSAC) (12) and Maximum Likelihood Estimation SAmple and Consensus (MLESAC) (Torr & Zisserman, 2000) are some robust approaches to reject incorrect matches.

2.1 Overview of Methods

The following methods are summarized in this subsection: SIFT, SURF, KAZE, FAST, BRISK.

2.1.1 SIFT

Lowe proposed Scale Invariant Feature Transform (SIFT) in 2004 (Lowe, 2004) which is cornerstone of many feature detection algorithms. It is significantly invariant to scale rotations and limited affine variations. Algorithm has four main steps; first stage is scale-space extrema detection. Lowe provides computation efficiency by using Difference-of-Gaussians (DoG) operator which is approximation of Laplacian-of-Gaussian (LoG). Secondly, a detailed model is determined at each candidate location. Thirdly, based on local image gradient directions, orientations are assigned and lastly for each keypoint, local image gradients are calculated at the selected scale that provides stability in local shape distortion and illumination changes.

2.1.2 SURF

Bay (Bay, Tuytelaars, & Van Gool, 2006) proposed SURF (Speeded Up Robust Features) which is speeded up version of SIFT. SURF uses a very basic Hessian matrix approximation for interest point detection. For a given point $\mathrm{x} = (x, y)$ at scale σ in image I Hessian matrix $H(x, \sigma)$ is defines as:

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \quad (1)$$

where $L_{xx}(x,\sigma)$, $L_{xx}(x,\sigma)$ $L_{yy}(x,\sigma)$ are the convolution of the Gaussian second order derivative. Even further Bay approximates Laplacian of Gaussian with box filters which can be done in parallel for different scales. With this approximation, Gaussian derivatives can be evaluated with low computational cost using integral images.

In description part, distribution of the intensity content within the interest point neighborhood is described. For each subregion Haar wavelet responses are taken and represented to get SURF feature descriptor of 64-D length that significantly effects computation and provides fast matching.

2.1.3 KAZE

Alcantarilla et al. introduced KAZE algorithm (Alcantarilla, Bartoli, & Davison, 2012) that uses nonlinear diffusion filtering combined with a conductivity function (Perona & Malik, 1990) instead of gaussian scale space in in detection and description of 2D features. The idea behind the nonlinear diffusion filtering is decreasing natural boundary loss and reducing the noise in Gaussian blurring. By non-linear scale space, blurring can be made locally adaptive to the image data and it can provide superior localization accuracy and distinctiveness. KAZE detector computes the response of scale-normalized determinant of the Hessian (DoH) at multiple scale levels:

$$L_{Hessian} = \sigma^2 (L_{xx} L_{yy} - L_{xy}^2) \quad (2)$$

where (L_{xx}, L_{yy}) are the second order horizontal and vertical derivatives respectively, and L_{xy} is the second order cross derivative. Unlike SIFT, at orientation assignment the gradients are represented as points in a vector space, instead of orientation histogram. KAZE descriptor uses an adapted form of Modified SURF Keypoint Descriptor that finds the dominant orientation in a circular area for each interest point. And with expense of computation it obtains scale and rotation invariant features.

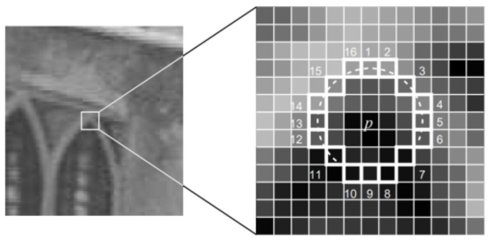


Figure 2. Fast Image showing the interest point under test and the 16 pixels on the circle (Rosten & Drummond, 2006)

2.1.4 FAST

Rosten and Drummond proposed the FAST algorithm (Rosten & Drummond, 2006) that is computationally faster than other detection methods (SIFT SURF KAZE). Like SUSAN (Smith & Brady, 1997)Smallest Univalue Segment Assimilating Nucleus), FAST algorithm uses a circle of 16 pixels (Figure 2) for each pixel to decide whether is it a keypoint or not. FAST detector first compares pixel intensity Ip with intensities of 9 1 5 13. If at least three of the four-pixel values (I1, I5, I9 and I13) are not above or below threshold, then P is not a corner. If it passes the threshold then check for all 16 pixels whether they are satisfying the criteria. In this way FAST algorithm has faster computation. But the algorithm has several weaknesses such as: for n<12, algorithm does not reject as many candidates, determination of the fast test pixels contains assumptions about the distribution of corner appearance, information from the first 4 tests is discarded, adjacent features are detected to one another. To overcome these issues a machine learning approach is used. In this approach for each pixel three state is defined as:

$$S_{p\to x} = \begin{cases} d, & I_{p\to x} \leq I_p - t \qquad (darker) \\ s, & I_p - t < I_{p\to x} < I_p + t \qquad (similar) \\ b, & I_p + t \leq I_{p\to x} \qquad (brighter) \end{cases} \quad (3)$$

From these states feature vector V is created. Vector is divided into 3 subsets: PS (similar points), PD (darker points) and PB (brighter points). Then a decision tree classifier (Quinlan, 1986) is performed to correctly classify all corners seen in the training set by using entropy minimization. Also, non-maximal suppression is used to eliminate interest points in adjacent locations by computing score function V.

2.1.5 BRISK

Leutenegger (Leutenegger, Chli, & Siegwart, 2011) proposed BRISK algorithm in 2011. The algorithm applies scale space that is used in SIFT algorithm to get advantage of scale invariance and extracts corner points by using AGAST (Mair, Hager, Burschka, Suppa, & Hirzinger, 2010). Then by looking at the gray scale relationship of random sample point pairs around each local image, binary feature descriptor is established and in matching part using hamming distance with EXOR operation instead of calculating Euclidian distance (like in SIFT, SURF or KAZE) has significant impact on performance of the algorithm.

3. EXPERIMENTS AND RESUTS

3.1 Experimental Setup

Matlab-2017b with OpenCV 3.4.1is used for performing the experiments. Specifications of the computer system are: Intel® Core i7-8700 CPU @ 3.19 GHz 8 GB RAM. Algorithm parameters are shown in Table 1.

Algorithm	OpenCV Object - Parameters
SIFT (128)	cv.SIFT(‘ContrastThreshold’,0.04,’Sigma’,1.6,’EdgeThreshold’,10,’NOctaveLayers,’3)
SURF (64D)	cv.SURF(‘HessianThreshold’,50,’Extended’false,’NOctaveLayers,3)
KAZE	cv.KAZE(‘Threshold’,0.0001,’NOctaveLayers’,4)
BRISK	cv.BRISK(‘DMax’,5.85,’DMin’,8.2)

Table 1. OpenCV parameters of algorithms

3.1.1 The Matching Strategy

In image matching, we use Fast Approximate Nearest Neighbors Approach (Muja & Lowe,2009). The algorithm uses hierarchical k-means trees that are created by splitting the data points at each level into different regions of K using k-means clustering. Using this priority queue provides efficiency in large data sets with minor loss in accuracy. Because of our data set is also extensive we use Fast Approximate Nearest Neighbors Approach in our research.

3.1.2 Outlier Rejection

In this study, we use Maximum Likelihood Estimation SAmple and Consensus with LLN approach (Grinstead & Snell, 1997) in outlier rejection. In that approach, previous random samples are used to benefit subsequent samples of RANSAC-like robust estimators. Because of its accurate results we used MLESAC (Zhang, Rastgar, Wang, & Vincent, 2009) in this paper.

3.2 The Datasets

Two datasets are used for this research. Urban (Figure 3) and rural area (Figure 4) images are taken from Google Earth Pro from different times of acquisition and seasons. Urban and rural area images are studied separately since the performance of the features in these areas are different. Urban areas contain more linear structures and sharp corners while rural areas have homologous view and have less significant features.
Before the experiment, resolution (GSD) of the matched images are set to the same width (0.49 meter for rural area images and 0.35 meter for residential area images).

(a)

(b)

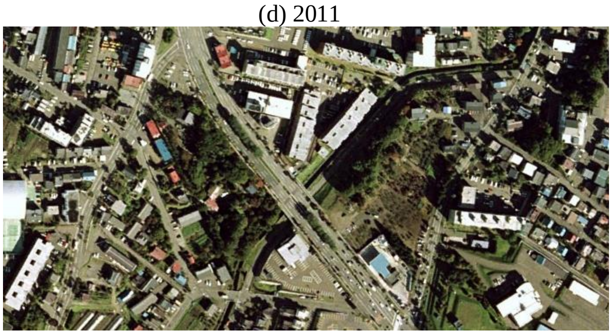
Figure 3. a) Urban Area Image
b) Rural Area Image

(a) 2016

(b) 2014

(c)2013

(d) 2011



(e) 2004

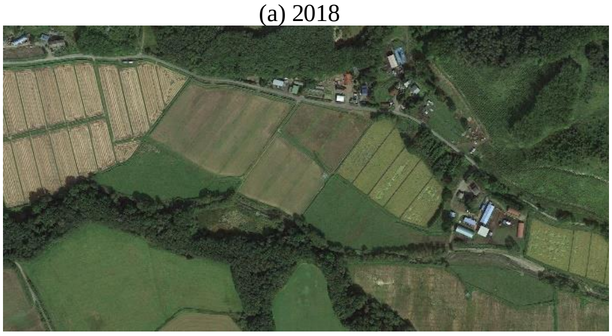
Figure 4. Urban Area Image Set

(d) 2012

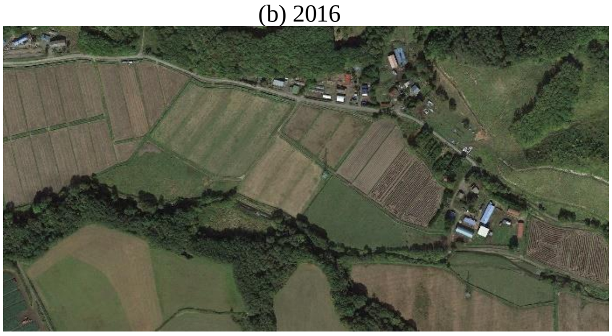
(e) 2003

Figure 5. Rural Area Image Set

(a) 2018



(b) 2016



(c) 2014

3.3 Results

3.3.1. Rural Area Images

The results of the experiments on rural area are summarized in Table 2, Table 3, Table 4:

NUMBER OF MATCHES		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2003	131	316	571	56	207
	2012	262	275	313	34	478
	2014	125	255	420	44	311
	2016	249	325	434	68	357
	2018	134	243	468	67	231
AVERAGE		180.2	282.8	438.5	53.8	316.8

Table 2. Detection-Description Algorithms Comparison in terms of number matched points (experiments on the rural area)

NUMBER OF INLIERS		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2003	6	5	7	7	6
	2012	175	19	149	23	326
	2014	69	6	40	25	203
	2016	90	11	20	33	199
	2018	41	6	39	28	90
AVERAGE		76.2	9.4	44.3	23.2	164.8

Table 3. Detection-Description Algorithms Comparison in terms of number of inliers (experiments on the rural area)

TIME (seconds)		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2003	9.40	5.74	24.74	4.82	7.20
	2012	9.38	5.63	31.47	4.53	6.48
	2014	9.75	5.89	19.16	4.81	8.29
	2016	9.68	5.81	14.62	4.90	8.45
	2018	9.47	5.60	7.95	4.54	6.70
AVERAGE		9.536	5.734	15.2	4.72	7.424

Table 4. Detection-Description Algorithms Comparison in terms of time (experiments on the rural area)

Quantitative Comparison for Rural area

Number of inliers: FAST-SIFT > SURF-SIFT > BRISK-SIFT >KAZE-KAZE >SURF-KAZE

From the results of the experiments on rural area FAST-SIFT detection and description algorithm gives the highest number of inliers. Our data set is not rotated, and corresponding images are adjusted same scale as we mentioned above. Under these conditions we observe FAST detector gives accurate results and SIFT descriptor creates distinctive features in satellite images.

Ratio of number of inliers to number of matched points: FAST-SIFT>BRISK-SIFT>SURF-SIFT>KAZE-KAZE>SURF-KAZE

The ratio of number of inliers to number of matched points allows us to make an inference about descriptor performance. If the match points are not rejected with outlier rejection we understand descriptor describe features distinctively. When descriptor performs accurately inlier number with respect to match points is high.

Total matching time: SURF-SIFT>KAZE-KAZE>FAST-SIFT>BRISK-SIFT

BRISK-SIFT provide fastest image matching. BRISK is a binary detector and combination of BRISK- SIFT can be useful when application needs high speed computation.

In summary, FAST-SIFT algorithm is preferable in rural satellite images to get high number of inliers with the cost of computation time in image matching algorithms.

3.3.2. Urban Area Images

The results of the experiments on Urban area are summarized in Table 5, Tables 6, Tables 7.

NUMBER OF MATCHES		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2004	242	472	1167	251	917
	2011	332	436	951	403	780
	2013	186	439	1079	164	970
	2014	333	635	1279	393	1365
	2016	343	593	868	291	1516
AVERAGE		287.2	515	1068.8	300.4	1109.6

Table 5. Detection-Description Algorithms Comparison in terms of number of matched points (experiments on the urban area

NUMBER OF INLIERS		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2004	126	12	255	121	456
	2011	197	17	207	225	438
	2013	115	53	54	61	579
	2014	218	121	343	192	809
	2016	235	33	284	138	1039
AVERAGE		178.2	47.2	228.6	147.4	664.2

Table 6. Detection-Description Algorithms Comparison in terms of number of inliers (experiments on the urban area)

TIME (seconds)		Methods				
		SURF-SIFT	SURF-KAZE	KAZE-KAZE	BRISK-SIFT	FAST-SIFT
Year of Image Acquisition	2004	22.65	8.85	15.55	10.78	19.890
	2011	23.05	9.13	15.94	11.97	22.94
	2013	21.98	8.78	15.38	10.47	18.024
	2014	23.25	9.39	16.08	12.78	25.72
	2016	23.22	9.26	14.63	10.73	23.844
AVERAGE		22.51	8.98	15.35	10.87	20.34

Table 7. Detection-Description Algorithms Comparison in terms of time (experiments on the urban area)

Quantitative Comparison for Urban Area

From the results of the experiments on urban area we observe that:

Number of inliers: FAST-SIFT > SURF-SIFT > BRISK-SIFT >KAZE-KAZE >SURF-KAZE

FAST- SIFT gives the highest number of inliers. FAST-SIFT works better under no rotation and scale differences.

Total matching time: SURF-SIFT> FAST-SIFT >KAZE-KAZE> BRISK-SIFT> SURF-KAZE

SURF-KAZE provide fastest image matching in urban area images while having least number of inliers.

The ratio of number of inliers to number of matched points: FAST-SIFT>SURF-SIFT>BRISK-SIFT>SURF-KAZE>KAZE-KAZE

3.4 Comparison Between Urban and Rural Area Image Matchings

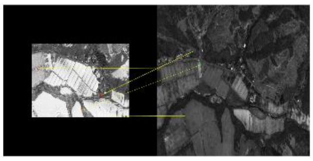
It is observed that FAST-SIFT detector and descriptor combination algorithm find more inliers at urban images than rural area images. This is because urban area images have more corners and distinctive points than rural area images. Therefore, in the urban area images, FAST-SIFT algorithm detects more distinctive keypoints. However, when we compared other methods with FAST-SIFT algorithms it still finds the best number of inliers in both rural area and urban area. Also see from (Figure 7 - 2003 Rural Image - Global Mapper Image) Fast algorithms suffers to detect features at snowy weather because of losing texture and sharpness.

3.4.1 Seasonal Comparison

Image matching performance of detection and description algorithms is varying with respect to seasons. Especially in winter and summer (Figure 6 - 2003) we observe noticeable change in number of inliers and matching accuracy. That shows change in texture, illumination significantly effects all algorithms in both rural area and urban area. FAST- SIFT algorithm still gives better performance under seasonal changes in satellite images.

4. CONCLUSION

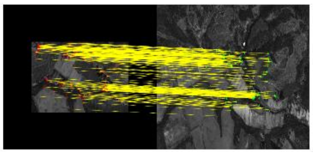
In this paper, we investigated well-known feature detection and description algorithm pairs to find most accurate and robust pair of algorithms against seasonal changes and geographical differences, growing or demolition of texture, and manmade construction etc... In the experiments, 3 image sets are used: Urban Area (Figure 4), Rural Area (Figure 5), Winter-Summer (Figure 4c-Figure5e). Images are selected as same scale, and same rotation. Under these conditions, in rural areas, FAST-SIFT detection and description pair is the best pair according to finding high number of Inlier points. FAST-SIFT combination has also more advantages over other detection-description pairs for finding higher number of Inlier points in the Urban areas as well. But one of the drawbacks of the FAST-SIFT pair is the higher computational cost against other pairs.



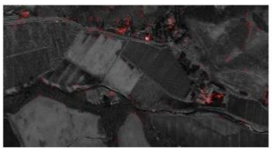
2003 Rural Image - Global Mapper Image

Matched keypoints 2003 Rural Image

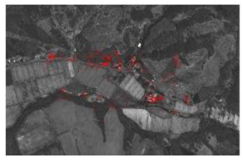
Matched keypoints Global Mapper Image



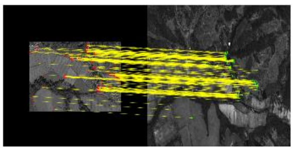
2012 Rural Image - Global Mapper Image



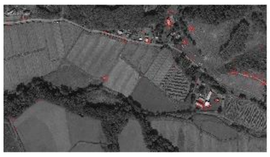
Matched keypoints 2012 Rural Image



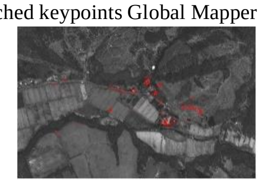
Matched keypoints Global Mapper Image



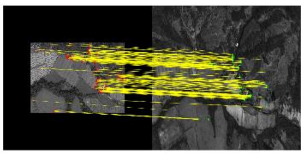
2014 Rural Image - Global Mapper Image



Matched keypoints 2014 Rural Image

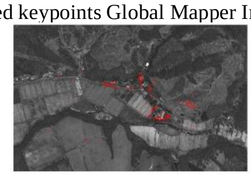


Matched keypoints Global Mapper Image

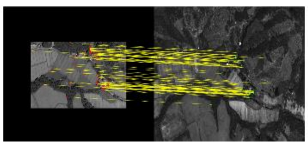


2016 Rural Image - Global Mapper Image

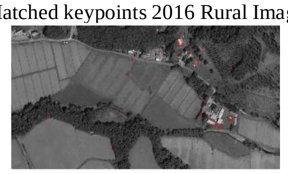
Matched keypoints 2016 Rural Image



Matched keypoints Global Mapper Image



2018 Rural Image - Global Mapper Image



Matched keypoints 2018 Rural Image

Matched keypoints Global Mapper Image

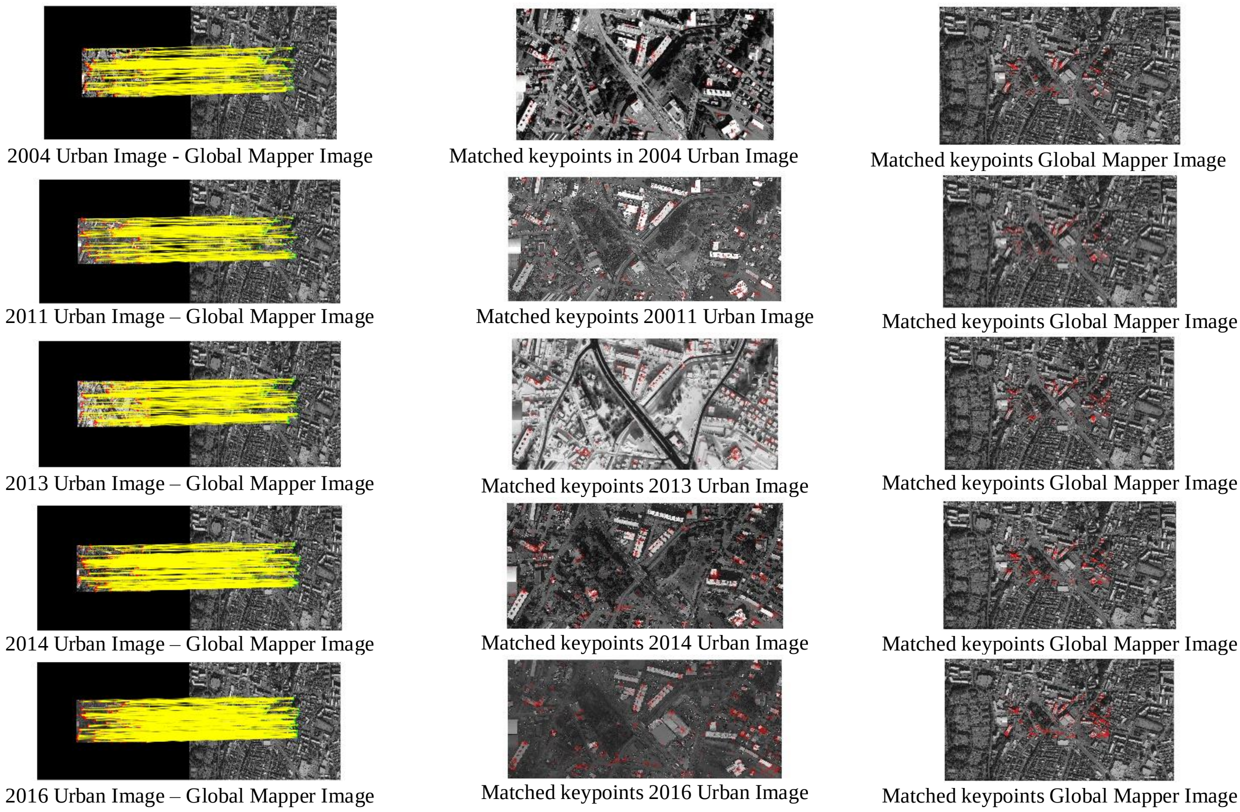


Figure 7. Experiments on Urban Area Images using FAST-SIFT algorithms

ACKNOWLEDGEMENTS

This study was partially supported in the framework of TUBITAK-TEYDEB 1511 program project no: 1170179.

REFERENCES

Alcantarilla, P. F., Bartoli, A., & Davison, A. J. (2012). KAZE features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-642-33783-3_16

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/11744023_32

Bentley, J. L. (2002). Multidimensional binary search trees used for associative searching. *Communications of the ACM*. https://doi.org/10.1145/361002.361007

Cartography, A., & Park, M. (1981). Random Sample Consensus : A Paradigm for Model Fitting with. *Communications of the ACM*.

Demirci, M. F., Osmanlioglu, Y., Shokoufandeh, A., & Dickinson, S. (2011). Efficient many-to-many feature matching under the l1 norm. *Computer Vision and Image Understanding*. https://doi.org/10.1016/j.cviu.2010.12.012

Friedman, J. H., Bentley, J. L., & Finkel, R. A. (1997). An algorithm for finding best matches in logarithmic expected time. *ACM Trans. On Mathematical Software*.

Fuentes, L. M., & Velastin, S. A. (2006). People tracking in surveillance applications. *Image and Vision Computing*. https://doi.org/10.1016/j.imavis.2005.06.006

Hassaballah, M., Abdelmgeid, A. A., & Alshazly, H. A. (2016). Image features detection, description and matching. *Studies in Computational Intelligence*. https://doi.org/10.1007/978-3-319-28854-3_2

Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). BRISK: Binary Robust invariant scalable keypoints. *Proceedings of the IEEE International Conference on Computer Vision*. https://doi.org/10.1109/ICCV.2011.6126542

Li, S. (2017). A review of feature detection and match algorithms for localization and mapping. *IOP Conference Series: Materials Science and Engineering*. https://doi.org/10.1088/1757-899X/231/1/012003

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*. https://doi.org/10.1023/B:VISI.0000029664.99615.94

Mair, E., Hager, G. D., Burschka, D., Suppa, M., & Hirzinger, G. (2010). Adaptive and generic corner detection based on the accelerated segment test. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-642-15552-9_14

Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2005.188

Milford, M., McKinnon, D., Warren, M., & Wyeth, G. (2011). Feature-based Visual Odometry and Featureless Place Recognition for SLAM in 2.5 D Environments. *Ssl.Linklings.Net*.

Muja, M., & Lowe, D. G. (2014). Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2014.2321376

Muja, M., & Lowe, D. G. (2009). 09 Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. International Conference on Computer Vision Theory and Applicati (VISAPP '09). https://doi.org/10.1.1.160.1721

Perona, P., & Malik, J. (1990). Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/34.56205

Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/11744023_34

Santosh, D., Achar, S., & Jawahar, C. V. (2008). Autonomous image-based exploration for mobile robot navigation. *Proceedings - IEEE International Conference on Robotics and Automation*. https://doi.org/10.1109/ROBOT.2008.4543622

Sminchisescu, C., Bo, L., Ionescu, C., & Kanaujia, A. (2011). Feature-Based Pose Estimation. In *Visual Analysis of Humans*. https://doi.org/10.1007/978-0-85729-997-0_12

Torr, P. H. S., & Zisserman, A. (2000). MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*. https://doi.org/10.1006/cviu.1999.0832

Wang, H., Mirota, D., & Hager, G. D. (2010). A generalized kernel consensus-based robust estimator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2009.148